Hardware & Physical Computing 2018-2019
Project Documentation - Group 10

Anna Tol   Jeannette Shijie Ma   Marissa Memelink

January 2019

1 Initial plan

We designed an interactive pendulum which creates different patterns based on the input from the audience. It contains 3 main parts: the input register, the pendulum, and the output presenter (the paint and canvas). The input register will collect information from a certain sensor. The collected information will be translated to the movement of pendulum. The movement of pendulum will be visualized by an output.

1.1 Idea variations and final idea

Various ideas for the three part of this project are listed in Table 1. The decision on what idea will be executed has been made based on pilot trails. This depended on feasibility and outcome quality.

Not listed in this table, is the input idea of an ultrasonic sensor. This idea has emerged during the process and has become the final form of input. The reason for the decision was that an ultrasonic sensor is more precise compared to for example a sound sensor, since the sound sensor can also pick up noise.

Parts	Ideas
Input	Shadow of people walking around (light sensor) People singing or clapping (sound sensor) Mouse/keyboard activity (connect to computer) Output from other projects (collaboration)
Pendulum	It needs to be triggered (pulled and released), or start automatically Only pendulum moves, or both pendulum and canvas need to move
Output	Ink/paint on paper canvas Dyed isomalt on metal canvas (lollipop) UV light on photo-luminescent painted canvas

Table 1: Various ideas for the 3 parts

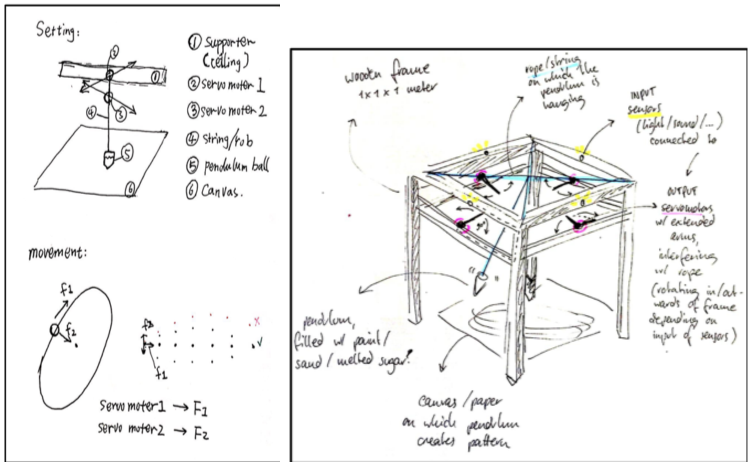

Figure 1: Plan A (Left); Plan B(Right)

An ultrasonic sensor also allows for a more natural feel than giving input to for example a computer through a keyboard or mouse. Finally, we were interested in doing the project in the dark. This would eliminate the idea of shadows, since these are absent in complete darkness.
From the idea to work with darkness also came the idea to work with uv light as an output. As found online (see the link in table 1 after 'UV light on photoluminescent painted canvas'), UV light leaves traces on a canvas covered glow in the dark paint. This principle has formed the base of the project. However, the project still had to include the original marble, the reason being the weight that created a better swing. Therefore, the UV light had to be attached to the marble. With an ultrasonic sensor as an input and a UV LED on a marble to form the pendulum, there still had to be decided on a frame. To possible frame ideas can be found in Figures 1.

Both frames have been built in order to test the physics. From the trials emerged the idea to build a rectangular frame. This frame only needs three servo motors, three sensors and has as a benefit compared to four of each that there is no direct opposite force to pull on the string. The prototype can be found in Figure 2. This teepee-like structure worked very well, so the next step was to expand and enlarge.
Instead of creating a larger teepee, we decided to create a triangle that can be hung up to the ceiling. This concept gives people more room to walk around the structure. The final idea on what the structure should look like can be found in Figure 3. The triangle that you see in this figure will be mounted to the wall with a string hanging in the middle. Each servo motor on the sides will have a string attached to it that is connected to the string in the middle.

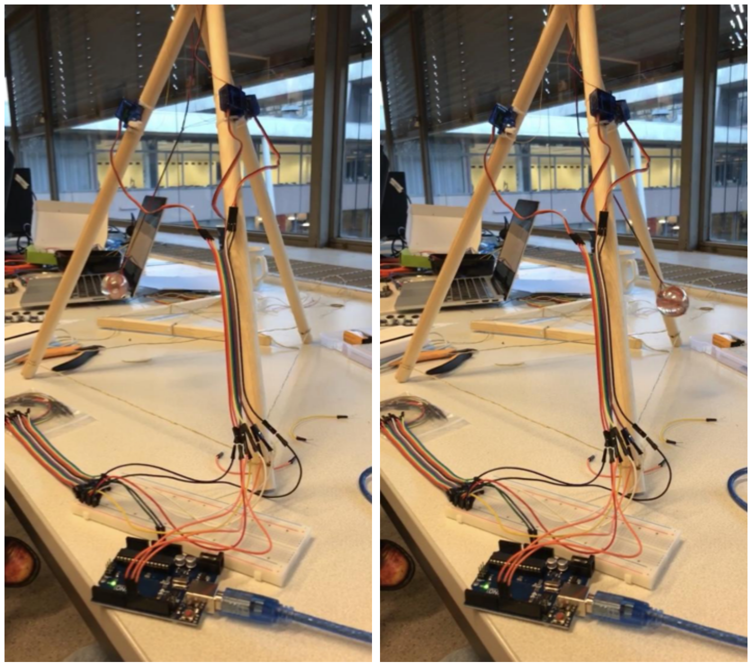

Figure 2: Teepee shaped prototype

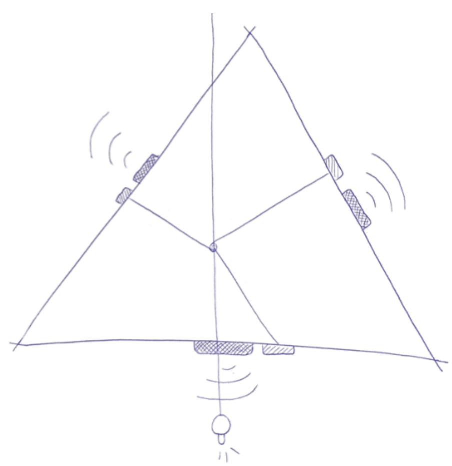

Figure 3: Final frame idea

2 Who did what

We have done a fairly great deal all together, which we are very proud of. Several thing we did together, however we have also done some tasks apart. The following section will give an overview of the division of tasks. Figure 4 demonstrates the materials used in this project.

Together:

- Concept development
- Deciding on and ordering of required materials, sensors and motors
- Testing and debugging

Anna:

- Fabrication of the net
- Coding
- Soldering
- Wiring

Jeannette:

- Knotting of pendulum net
- Arranging UV light glow in the dark paint
- Knotting/coating of the conductive wire

Marissa:

- Production of the wooden frame and elements
- Mounting of the frame
- Soldering
- Wiring

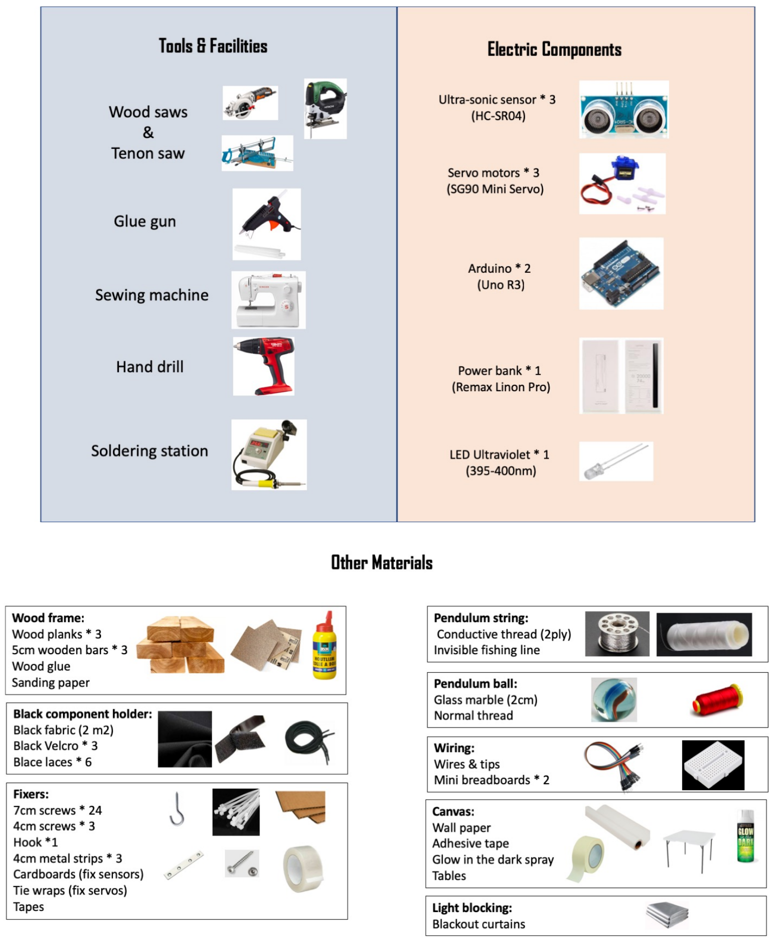

Figure 4: Materials used in this project

3 Problems encountered & solutions

We tested the physics of pulling the rope of the pendulum intensively (on different scales, and with different angles and leverages of the servomotors), to make sure it worked as we expected. Below some problems we encountered and how we solved them.

Physics: What is the optimal angle and leverage for maximum pull force? We tested multiple settings, heights, angles and leverages to find the optimal force.

Shape and size of the frame: After testing on smaller scale, we recalculated all dimensions to convert them to bigger scale. And tested again. And again.

Wiring: Multiple times our structure stopped working because of bad wiring. The solution was quite obvious; make sure that everything is connected in the right way.

Sticks on the servo's: The servo motors needed extension of their arms, for more leverage in pulling the pendulum. To find the appropriate robustness and lightness for the arms, we used small wooden sticks, drilled a small hole in the tip (first consolidated them with extra tape to prevent the wood from splitting), attached them with tape to the arms of the sermotors, and attached the fishing line through it.

String for the UV light: We wanted a most invisible rope as possible. How to still light a LED at the end of the marble? We bought very thin conductive wire, running from above in the Arduino all the way down along the rope of the pendulum, to the UV LED under the marble, all the way up again to the Arduino. To not let the wire conduct when touching other, we coated one side of the wire with small knots of fishing line. The long wire simultaneously functioned as a resistor for the UV LED.

Arduinos: One arduino had a hard time running both sensors and servo's. Two Arduino's, but how can they communicate? We used one master Arduino, sending values to a "slave" Arduino.

One servo motor broke: we still don't know if we fried this, but we had to arrange a new one last minute.

Pulling too hard: We implemented a delay in the code, so the servomotors were now rotating more slowly and facilitated a more controlled pull to the fishing line.

Height of the sensors: How to know what is the most optimal height, when the frame is not in its final place yet? We attached the sensors to small wooden bars, that were adjustable/bendable with a metal strip. This gave us freedom to lower/heighten them manually once the frame was mounted to the ceiling.

Pendulum was hanging a bit too high: Folded paper underneath the table legs.

Last: Sometimes everything just stopped working without a reason and started working again without a reason. Sometimes the sensors measured distances of 20+ meters, which was not possible in the room. We adjusted the code on this multiple times (to only trigger the servo's between 170 meter and 280 meter), but in the end this wasn't necessary anymore. We did found out that the sensors responded better to harder surfaces (phones/card boards/laptops) than softer ones (fabric/clothes).

4 Result

The result contains an isosceles triangular wooden frame (each side around 1 meter long, and 4 cm high), mounted to the ceiling, with a fabric triangular net to keep the Arduino's, wiring and power bank in place. From the center of the frame and net, a rope of thin fishing line with a marble is hanging from the ceiling mounted with a little screw hook. Under the marble a UV LED light is attached, hanging just above the flat surface of three tables shoved together, forming a square. On the tables lies a canvas made of three strokes of white wallpaper spray painted with glow in the dark paint, which - once the room is darkened - will light up the path the UV LED light makes when the pendulum is swinging. This way, the light "drawing" is created by the people walking around the installation. Their interaction with the sensors decides when the servo motors pull the pendulum, and can thus change its path accordingly.

4.1 Schematic graphs

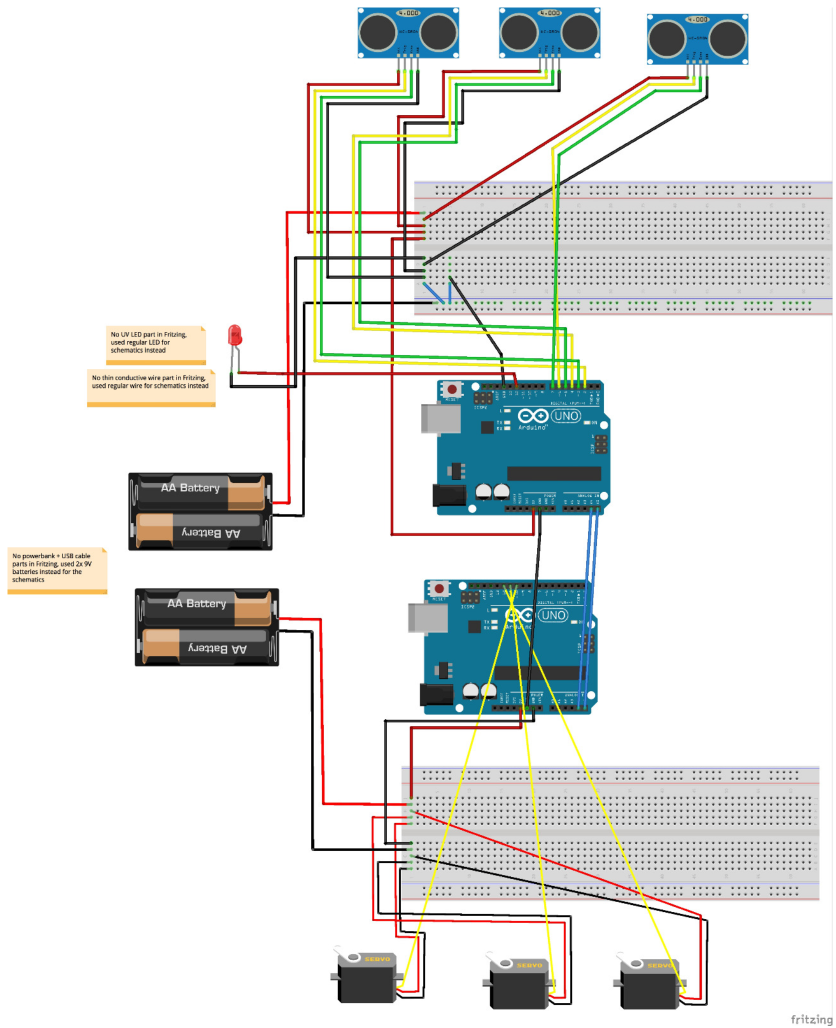

Figure 5: Schematic graph 1

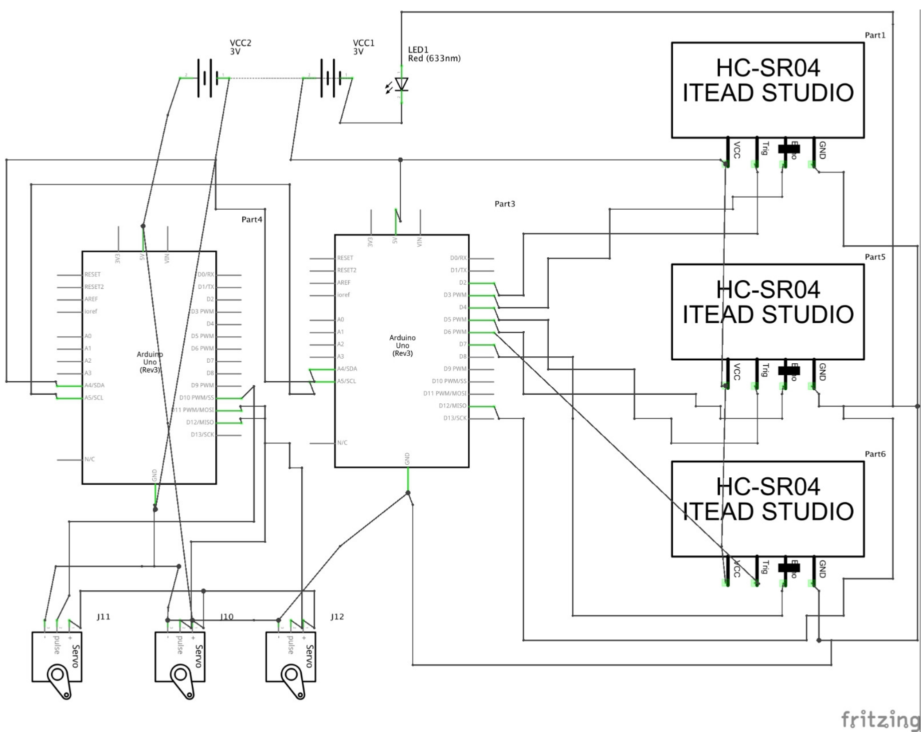

Figure 6: Schematic graph 2

4.2 Demonstration graphs

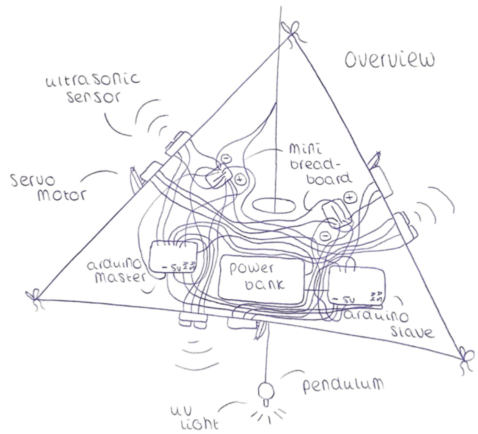

Figure 7: Overview of the structure

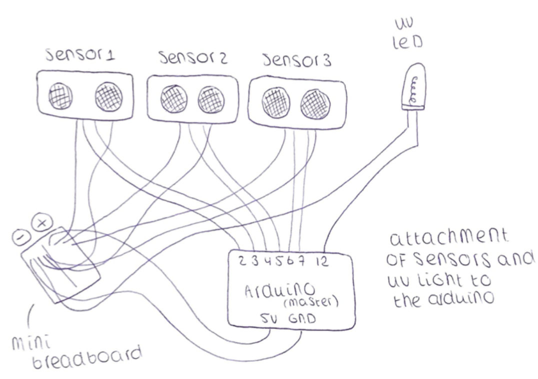

Figure 8: Overview of the attachment of the sensors and UV light

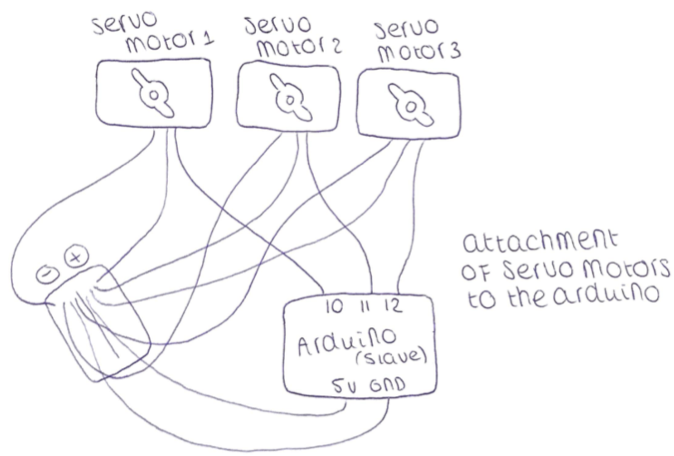

Figure 9: Overview of the attachment of the servo motors

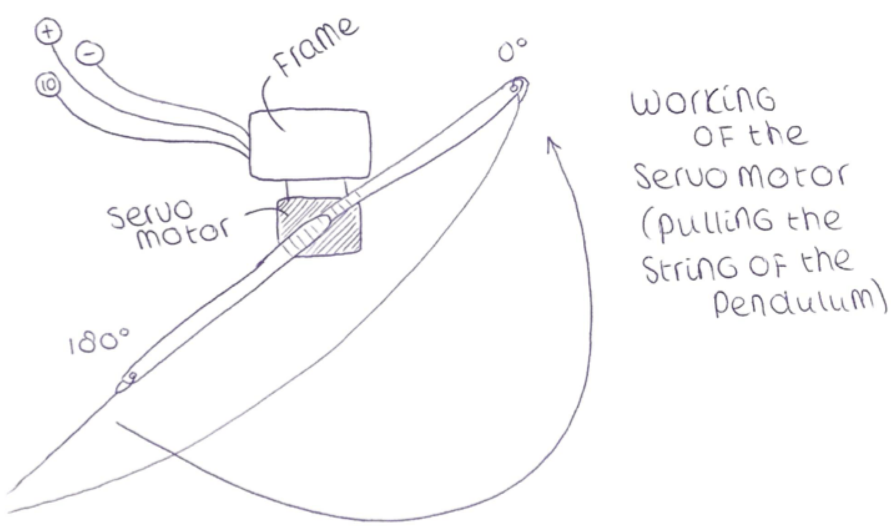

Figure 10: Functioning of the servo motors

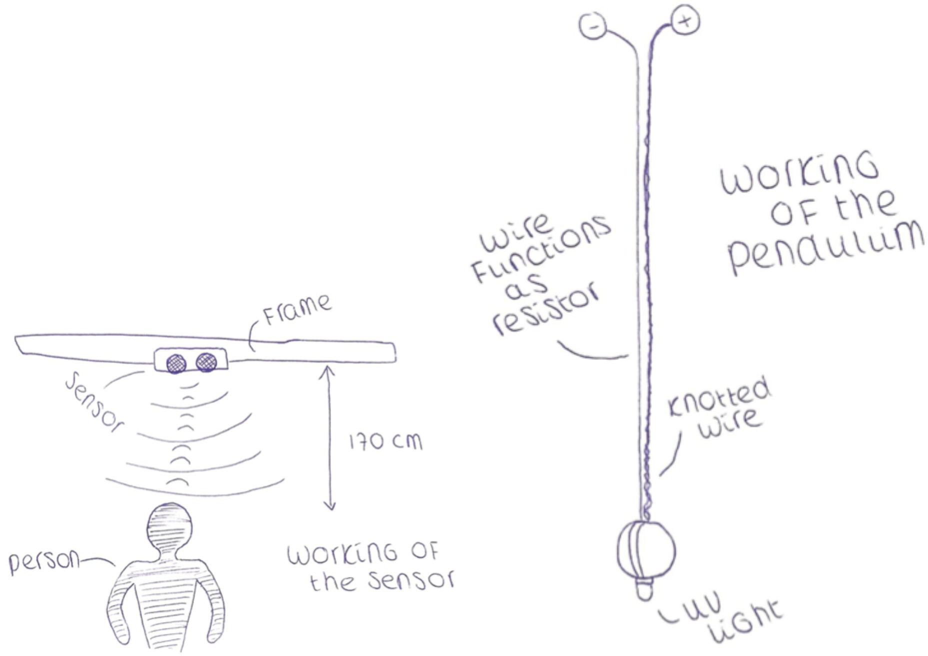

Figure 11: Functioning of the ultrasonic sensors(left) and UV light(right)

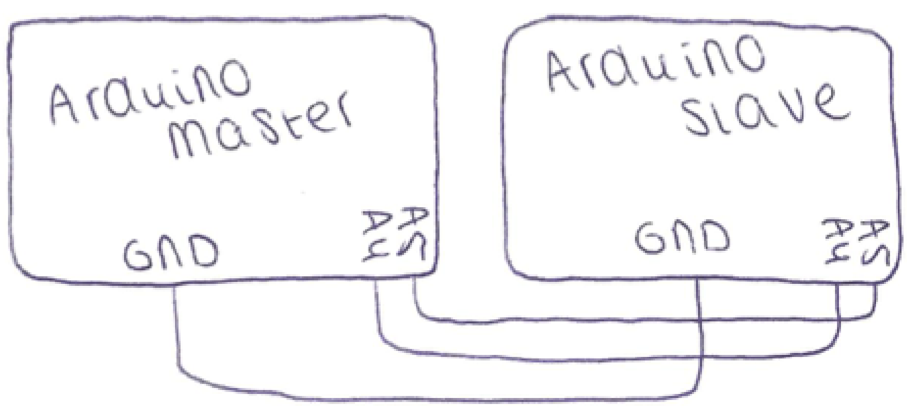

Figure 12: Communication between the two Arduinos

5 Licensing

We selected the Attribution-ShareAlike 4.0 International (CC BY-SA 4.0) license. This license allowed free sharing and adaptation under the same license while crediting us. The sharing can be used commercially.

Reasons of choice:

- We would give everyone the inspiration of our work and the possibility to improve it. Therefore we would allow adaptation and share under our credit.
- Although this project is non-purpose oriented, it can be applied to many cases (entertainment park; toys; public installation etc.). Therefore we would like to make it open for commercial usages.
- We would also like to promote creative sharing in the society. Therefore we decided the sharing of any adaptation should be under the same type of license (Attribution-ShareAlike).

Figure 13: License

6 Arduino Code

Our project included two arduino's; one for the sensors and the uv light, and one for the servo-motors. As researched online, one master arduino can hold multiple slave arduino's. In our case, the arduino with the sensors and uv light was the master and the arduino with the servo motors the slave. Code for both arduino's can be found in Figure 14 and 15.

```
// connection to slave arduino
#include <Wire.h>

#define LED 12

// sensor 1
#define trigPin1 2
#define echoPin1 3

// sensor 2
#define trigPin2 4
#define echoPin2 5

// sensort 3
#define trigPin3 6
#define echoPin3 7

long duration, distance, sensor1, sensor3, sensor2;

boolean s1_trigger = false;
boolean s2_trigger = false;
boolean s3_trigger = false;

void setup() {
  Serial.begin(9600);
  Wire.begin();
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
  pinMode(trigPin3, OUTPUT);
  pinMode(echoPin3, INPUT);
  pinMode(LED, OUTPUT);
  digitalWrite(LED, HIGH);
}

void loop() {
  digitalWrite(LED, HIGH);

  // calculate the distance for every sensor
  SonarSensor(trigPin1, echoPin1);
  sensor1 = distance;
  SonarSensor(trigPin2, echoPin2);
  sensor2 = distance;
  SonarSensor(trigPin3, echoPin3);
  sensor3 = distance;

  /* For troubleshooting:
  Serial.print(sensor1);
  Serial.print(" - ");
  Serial.print(sensor2);
  Serial.print(" - ");
  Serial.println(sensor3);
  */

  // if person close to sensor, wire number 1 to the slave
  if(sensor1 < 170 && sensor1 != 0 && s1_trigger == false){
    s1_trigger = true;
    Wire.beginTransmission(9);
    Wire.write(1);
    Wire.endTransmission();
    // if person close to sensor but sensor is already triggered; wire 0
  } else if(sensor1 >= 170 && s1_trigger == true) {
    s1_trigger = false;
    Wire.beginTransmission(9);
    Wire.write(0);
    Wire.endTransmission();
  } else {
    Wire.beginTransmission(9);
    Wire.write(0);
    Wire.endTransmission();
  }

  // if person close to sensor, wire number 2 to the slave
  if(sensor2 < 170 && sensor2 != 0 && s2_trigger == false){
    s2_trigger = true;
    Wire.beginTransmission(9);
    Wire.write(2);
    Wire.endTransmission();
    // if person close to sensor but sensor is already triggered; wire 0
  } else if(sensor2 >= 170 && s2_trigger == true) {
    s2_trigger = false;
    Wire.beginTransmission(9);
    Wire.write(0);
    Wire.endTransmission();
  } else {
    Wire.beginTransmission(9);
    Wire.write(0);
    Wire.endTransmission();
  }

  // if person close to sensor, wire number 3 to the slave
  if(sensor3 < 170 && sensor3 != 0 && s3_trigger == false){
    s3_trigger = true;
    Wire.beginTransmission(9);
    Wire.write(3);
    Wire.endTransmission();
    // if person close to sensor but sensor is already triggered; wire 0
  } else if(sensor3 >= 170 && s3_trigger == true) {
    s3_trigger = false;
    Wire.beginTransmission(9);
    Wire.write(0);
    Wire.endTransmission();
  } else {
    Wire.beginTransmission(9);
    Wire.write(0);
    Wire.endTransmission();
  }
}

// function for calculating distance to sensor
void SonarSensor(int trigPin,int echoPin) {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration / 2) / 29.1;
}
```

Figure 14: Code of the Arduino master

```
// connection to master arduino
#include <Wire.h>

#include <Servo.h>
Servo jeanette;
Servo marissa;
Servo anna;

int wire = 0;
int speed_forward = 1000;
int speed_backward = 360;

void setup(){
  // receive number (0, 1, 2 or 3) from master
  Wire.begin(9);
  Wire.onReceive(receiveEvent);

  // attach servo's
  jeanette.attach(10);
  marissa.attach(11);
  anna.attach(12);
  jeanette.write(180);
  marissa.write(180);
  anna.write(180);
}

void receiveEvent(int bytes){
  wire = Wire.read();
}

void loop() {
  // if sensor 1 is triggered, pull servo 1
  if (wire == 1){
    jeanette.write(0);
    delay(speed_forward);
    jeanette.write(180);
    delay(speed_backward);
    // if sensor 2 is triggered, pull servo 2
  } else if (wire == 2){
    marissa.write(0);
    delay(speed_forward);
    marissa.write(180);
    delay(speed_backward);
    // if sensor 3 is triggered, pull servo 3
  } else if (wire == 3){
    anna.write(0);
    delay(speed_forward);
    anna.write(180);
    delay(speed_backward);
  }
}
```

Figure 15: Code of the Arduino slave

7 Video and pictures

A video of the working system can be accessed HERE.

Figure 16: Pictures of the finished system 1

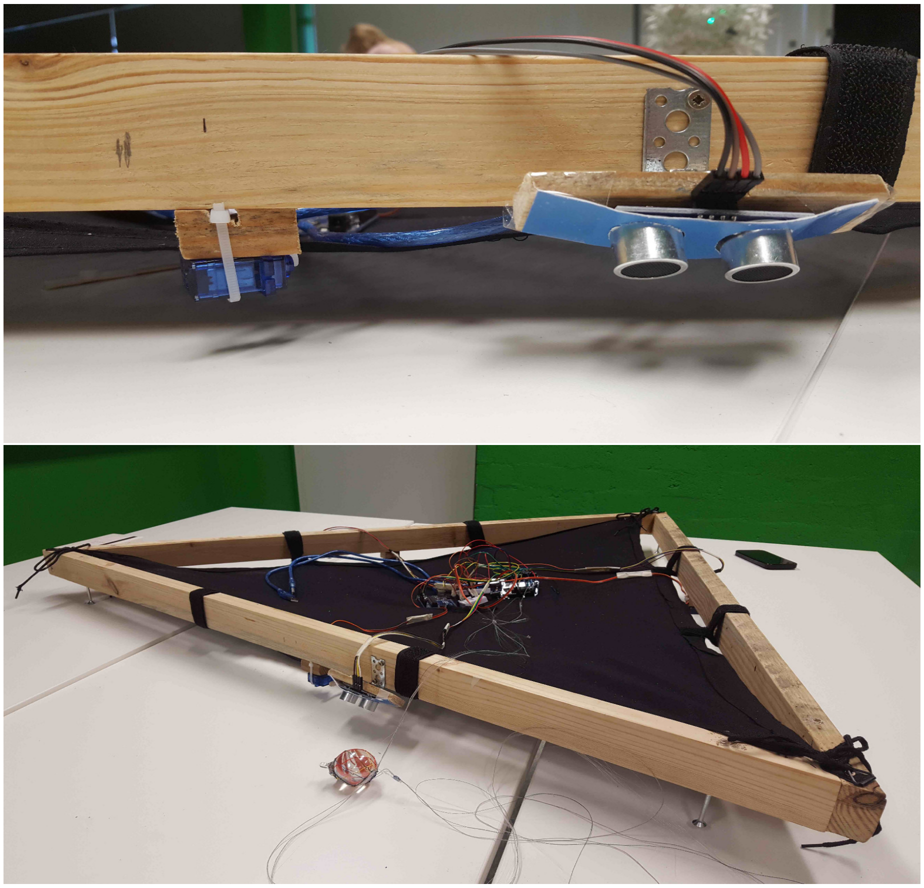

Figure 17: Pictures of the finished system 2