Daniel Taylor, Jake Burton, Winson Tao, Yamil Tarabin

Installation Development Project Documentation

Available at: [GitHub Release](https://github.com/Daniel-Luke-Taylor/Installation-Development-Project/releases/tag/1.0)

Demonstration at: [Prototype Video](https://drive.google.com/open?id=1Y8H00pvV_kOKPruEqJIzgvLQkkqwZNnR)

Contents

[Concept 2](#_Toc5714705)

[Vision 2](#_Toc5714706)

[Ideation 2](#_Toc5714707)

[Designs 3](#_Toc5714708)

[Wave Machine designs (old) 3](#_Toc5714709)

[Blueprints 4](#_Toc5714710)

[Ping Pong designs (final) 4](#_Toc5714711)

[Blueprint for laser cutting 5](#_Toc5714712)

[Equipment List: 6](#_Toc5714713)

[Water tank 6](#_Toc5714714)

[Water tank requirements: 6](#_Toc5714715)

[Access to a water source 7](#_Toc5714716)

[Arduino 7](#_Toc5714717)

[Actuators 7](#_Toc5714718)

[Sensors 7](#_Toc5714719)

[Paddle 7](#_Toc5714720)

[Technical Architecture 8](#_Toc5714721)

[APIs, IDEs, Libraries and SDKs Required 8](#_Toc5714722)

[Kinect for Windows SDK 2.0 8](#_Toc5714723)

[Arduino IDE 8](#_Toc5714724)

[Firmata 8](#_Toc5714725)

[Processing 8](#_Toc5714726)

[Arduino Firmata (processing library) 8](#_Toc5714727)

[KinectPV2 8](#_Toc5714728)

[Prototypes 9](#_Toc5714729)

[Making Waves 9](#_Toc5714730)

[Materials used: 9](#_Toc5714731)

[Propeller: 10](#_Toc5714732)

[Paddles: 11](#_Toc5714733)

[References 16](#_Toc5714734)

# Concept

## Vision

# Ideation

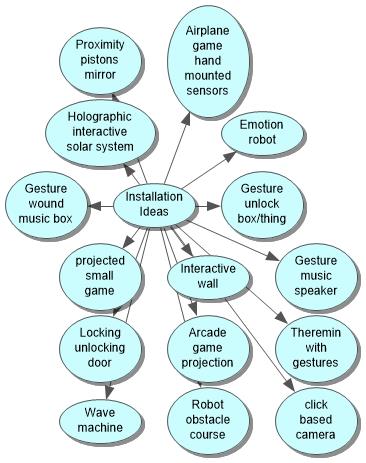


Figure 1 - Mind map showing possible project ideas

## Designs

### Wave Machine designs (old)

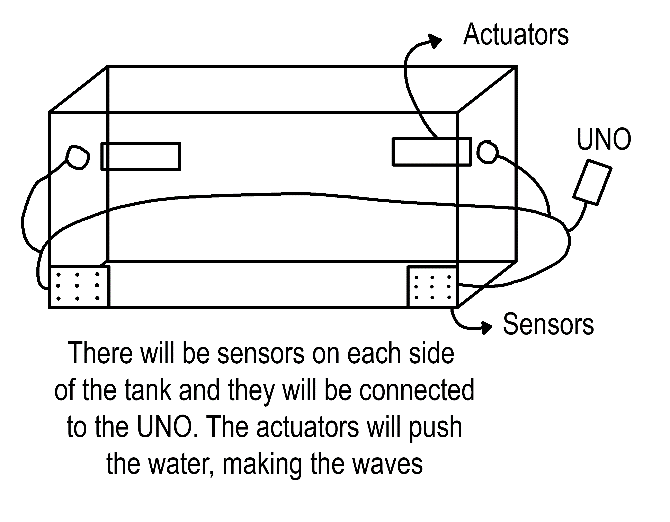
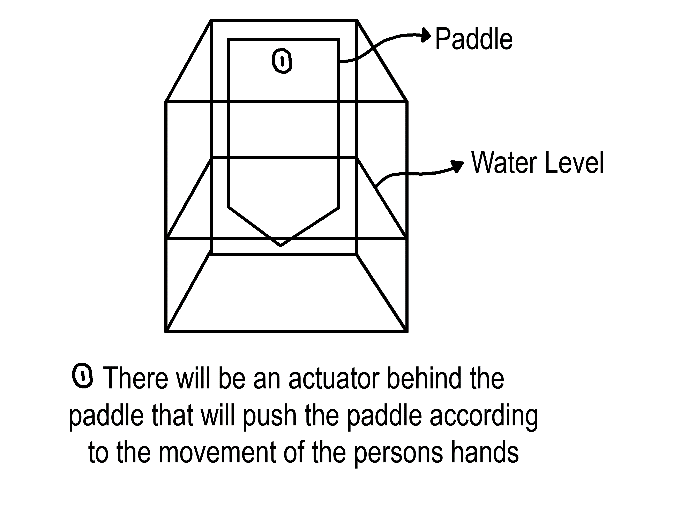
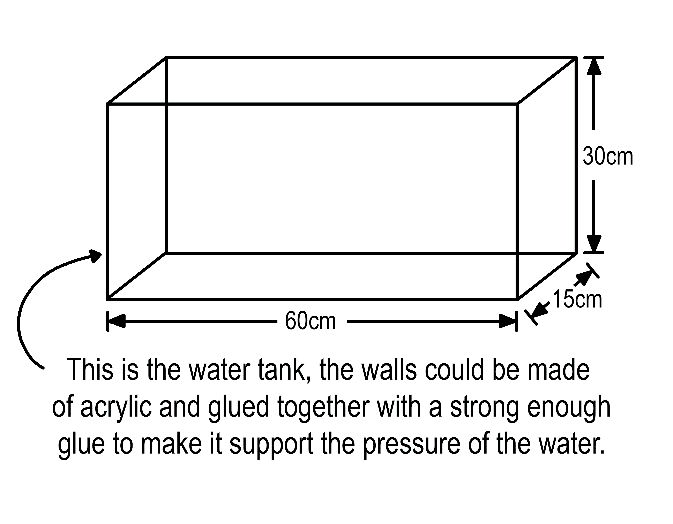
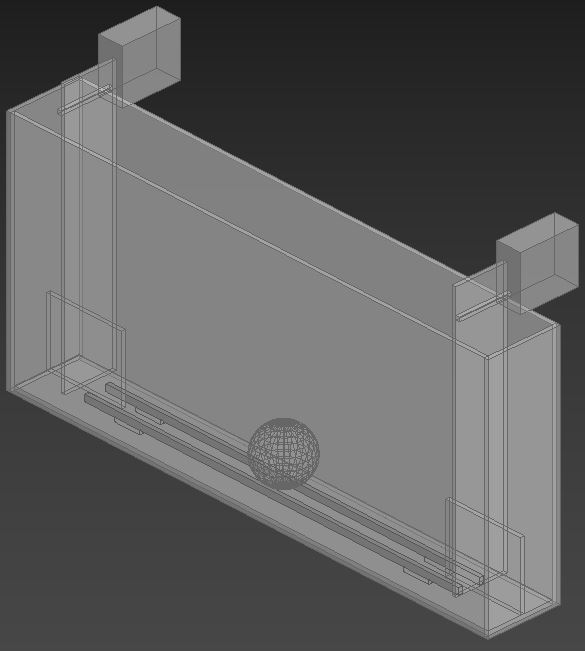


Figure 2 - Old project designs

## Blueprints

### Ping Pong designs (final)



56mm

380mm

180mm

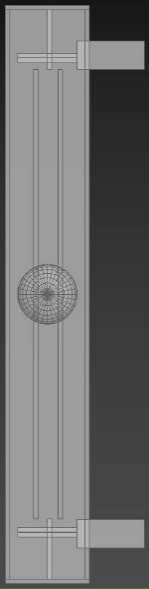
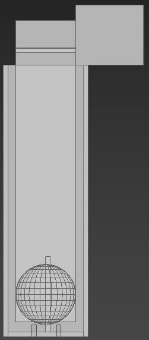
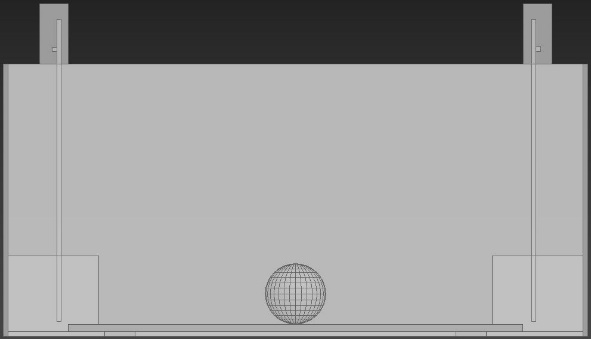
Ball

Rails

Paddles

Stoppers

Servos



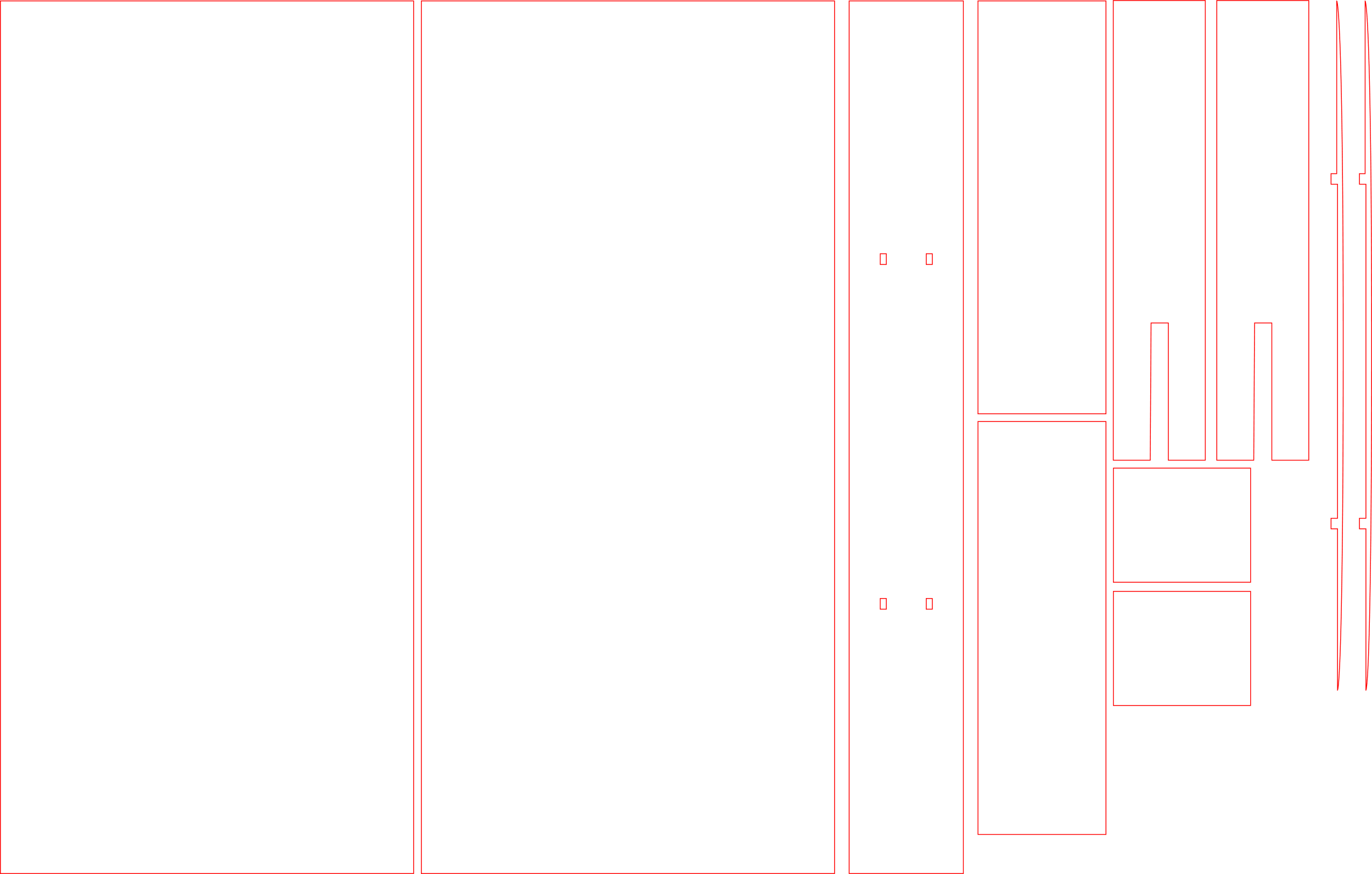
Front View

Top View

Side View

Figure 3 - 3D rendition of project output

### Blueprint for laser cutting



Front

Back

Base

Right

Left

Paddles

Stoppers

Rails (not final changed in laser cutter)

Figure 4 - Illustrator file used to laser cut container parts

# Equipment List:

## Water tank

The project revolves around using gestures to manipulate water, with water being one of the essential components it will require a container for it to be used.

### Water tank requirements:

* **Square or rectangular shaped water tank** – A square or rectangular shaped water tank will allow for easier manipulation over the water flow when it is being pushed in a single direction. A circular or round water tank has more space for the water to move around, dispersing all around towards the edges causing the water to seem more stagnant compared to a square or rectangular water tank.
* **Water tank** – The water tank must have enough room to fit in equipment such as the paddles and actuators along with the water itself. There also must be enough room for the water to flow from initial impact from being hit by a paddle to reaching towards the edge of the water tank without causing the water to immediate crash. However, it is also ideal to reduce the size of the water tank and the volume of the water to increase the efficiency to power the actuators. By keeping the water volume and tank size to a minimum, the chances that a relay or alternative power source as well as more powerful and expensive actuators to be used for the artefact are reduced, which will lower the cost of production as well as making it simpler to plan and power the equipment.

Plexiglass

Epoxy sealant

* **Water tank with an open top –** An open top water tank will enable equipment such as paddles and actuators to be fitted inside the water tank.
* **Water tank lid –** Because the water inside the tank will be moving a lot during its use, a lid will be required to contain the water inside its tank. Reason for why this is important are: to avoid water loss for prolonged usage, to avoid damaging the electrical equipment outside of the tank, to avoid water coming into contact with the user and ruining their clothes, and to avoid creating an electrical hazard for when water meets electricity.

## Access to a water source

The artefact will require water for it function. Tap water is the preferred option as it is more easily accessible from local toiletries as well as being the most cost-effective option compared to other liquids such as purchasing and using bottled mineral water.

## Arduino

The Arduino is another essential component for the artefact to function. The Arduino is a cheaper alternative to the raspberry pi, with another advantage being a larger source of innate power available to be used for external hardware. Whilst the Arduino is not as powerful as the raspberry pi, the programming used for the artefact should be simple enough for the Arduino to follow through.

## Actuators

An linear servo that can push and pull water with a paddle attached to it is required for the artefact to work.

## Sensors

Xbox One Kinect Camera Sensor – This sensor will be used to track the user’s hands and will be able output a number based on the horizontal distance of the hands. This will be the ideal user experience and is the first choice for input.

Ultrasonic Distance Sensor – By using this type of sensor, it can tell the distance between a user’s hand from the water tank, this sensor will be a backup if the Kinect sensor ends up failing.

## Paddle

A paddle is required as it will be attached to an actuator. The increase in surface mass will allow for easier water manipulation compared to using an actuator on its own.

# Technical Architecture

The user will be able to wave their hands in front of a Kinect sensor and when the user moves their hands to the side the servo mounted paddle connected to the Arduino will fire and hit a ping pong ball.

The solution will start with using a Kinect sensor for input. This will be connected to a computer powerful enough to process the data from the sensor and get hand coordinates. The processed data will then be sent to an Arduino which will use that data to move the servos.

## APIs, IDEs, Libraries and SDKs Required

### Kinect for Windows SDK 2.0

(Microsoft Corporation, 2014)

Used to connect the Kinect device to a computer and access the data from the steam.

### Arduino IDE

(Free Software Foundation Inc, 2018)

IDE used to interface with the Arduino and provides the Firmata code for external programs to connect to the Arduino.

### Firmata

(Free Software Foundation Inc, 2018)

Library used so the Arduino can receive commands from external programs.

### Processing

(Fry & Reas, 2019)

IDE used as a connector for the Kinect and Arduino technology, and to provide a visualisation of the implementation for debug purposes.

### Arduino Firmata (processing library)

(Free Software Foundation Inc, 2018)

Library used to access the Arduino from inside of the processing sketch.

### KinectPV2

(Lengeling, 2016)

Library that can access and process the data from the Kinect sensor.

# Prototypes

## Making Waves

### Materials used:

ب 
فين.. «موووم..،ن لاب 
وو 
٢٠٠١) ن 

Figure 5 – Resources for making waves prototype

* A large black rectangular shaped plastic container
* Transparent tape
* Rubber bands
* Craft sticks
* Card paper

The wave that needs to be created must be big, as the aim is to imitate as close as possible the motions of the hand. If the hand motion over the sensor is slow, the wave should also go slow; if the hand motion is fast, the wave motion should also be fast.

### Propeller:

The first prototype we made was a propeller, we used two craft sticks as the base and rubber bands to keep them together, the card paper was cut into a rectangular shape to then be attached to the craft sticks with tape.



Figure 6 - ConstructedPpropeller

With another pair of craft sticks, rubber bands and tape a rotational arm was made, the idea was that this would be attached to the motor and as it rotates the propeller would also rotate.

When the propeller was tested, we came to the realization that the type of waves created were not the desired type as the propeller made many small waves in a quick succession, while the desired waves were big and slow imitating the hand motions.



Figure 7 - Propeller Testing

### Paddles:

The second prototype we made were a pair of paddles, to make them we used the card paper and drew the shape of the container, after cutting it we used craft sticks to strengthen the paddle to increase its firmness and with tape we completely wrapped it to make it resistant to water.

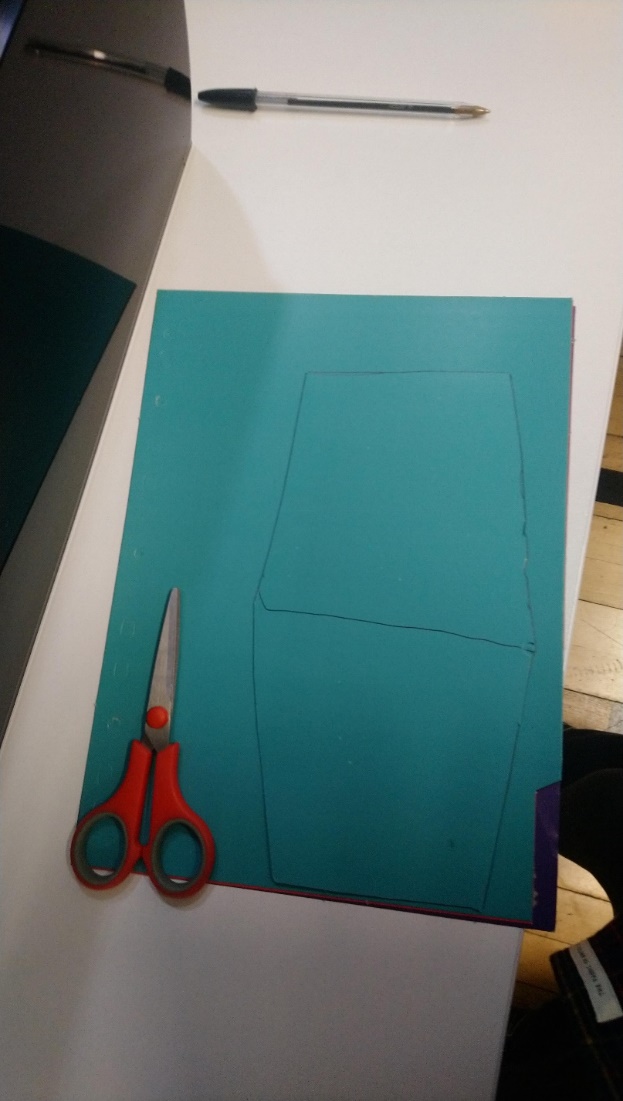




Figure 8 – Images Showing Paddle Construction



Figure 9 - Constructed paddles

When the paddles were tested, we came to the realization that this is the way forward, the paddles created one big wave that move according to the motions of the hand

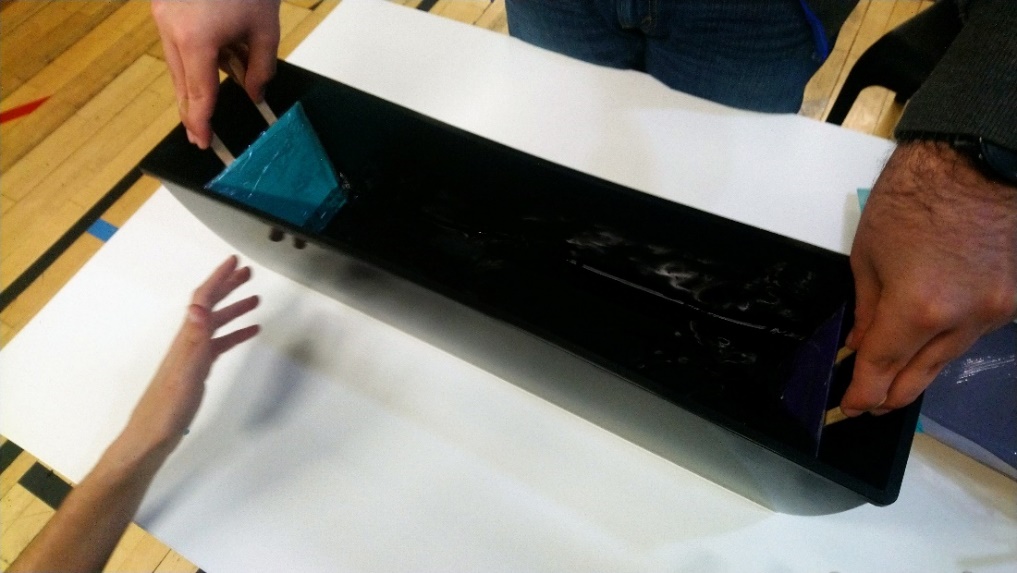


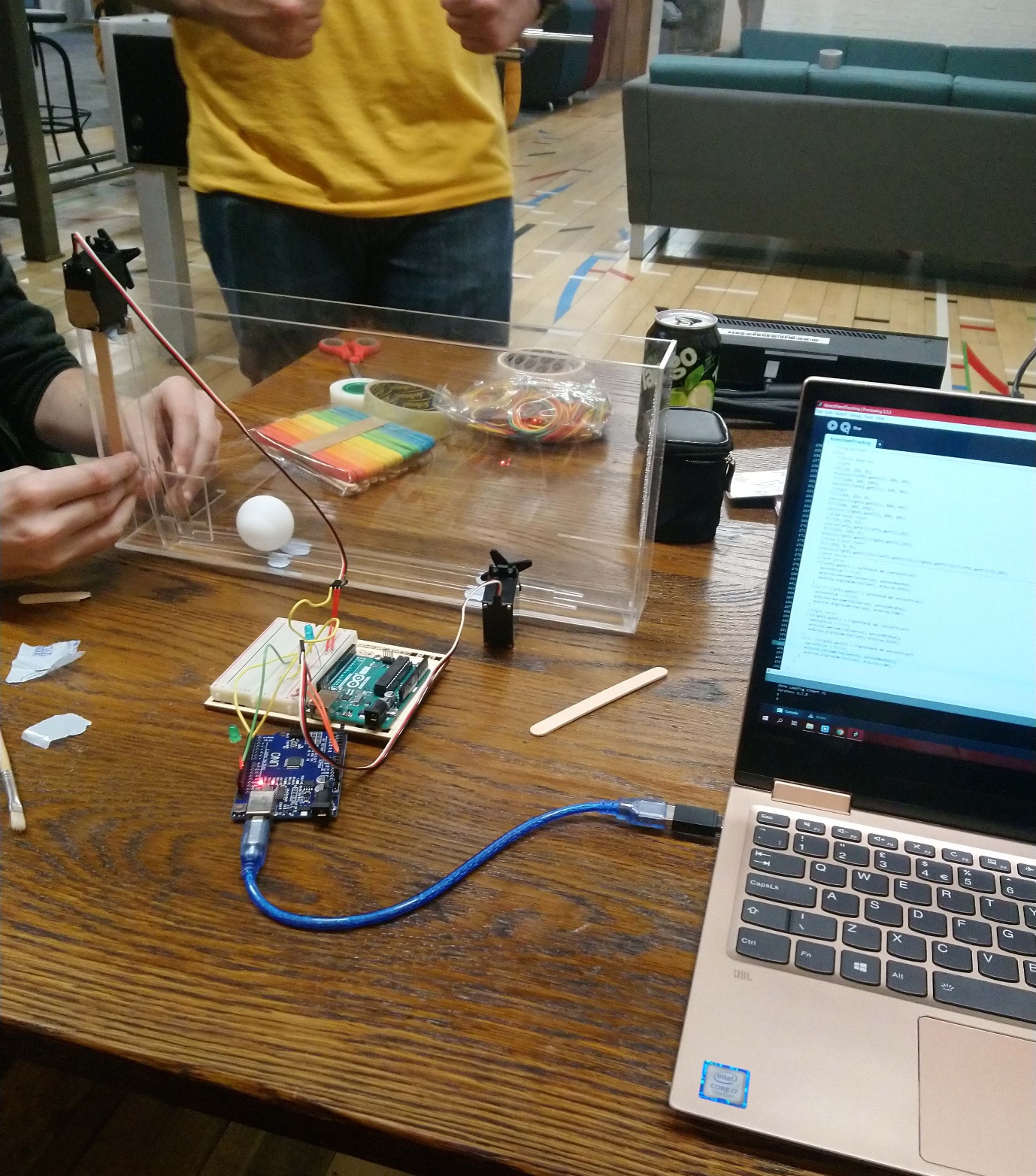


Figure 10 – Images Showing Paddle in Use

Thermin prototype

Final prototype writeup

Final prototype images



GitHub Release: <https://github.com/Daniel-Luke-Taylor/Installation-Development-Project/releases/tag/1.0>

Prototype Demonstration Video: <https://drive.google.com/open?id=1Y8H00pvV_kOKPruEqJIzgvLQkkqwZNnR>

Minutes

# References

Free Software Foundation Inc, 2018. *Arduino IDE.* [Online]   
Available at: https://www.arduino.cc/en/main/software

Free Software Foundation Inc, 2018. *Fimata 2.5.8.* [Online]   
Available at: https://github.com/firmata/arduino/releases

Fry, B. & Reas, C., 2019. *Processing 3.5.3.* [Online]   
Available at: https://processing.org/

Lengeling, T., 2016. *KinectPV2 0.7.8.* [Online]   
Available at: https://github.com/ThomasLengeling/KinectPV2

Microsoft Corporation, 2014. *Kinect for Windows SDK 2.0.* [Online]   
Available at: https://www.microsoft.com/en-gb/download/details.aspx