STORY LINK

CART360

Written by Mihaela Eftene
Prototype Progress Report

2020-11-13

INFORMATION PROJECT

Website Documentation:
https://emiha.com/CART360/pdfs/M_EFT-ENE_PrototypeProgress.pdf

All Links are interactive.

Written Response which addresses and relates the implicit concerns of Why Do We Prototype? & Fidelity Levels to the development process of your Physical Prototype

From the choice of materials to how to implement different sensors, Story Link has its own hurdles. According to the article "Why We Prototype" by Kathryn McElroy, prototyping is a good way to figure out the challenges faced by a product without having to spend a large amount of money or manufacturing a final product and having to go back to the drawing board to fix its inherent issues. My journey in prototyping started with brainstorming using post-its, sketches, circuit planning, and submitting a proposal (fig. 01-02). This first non-physical low-fidelity prototype helped me envision my product and discover new questions that I wasn't aware of at first. For instance:

- What is my main goal in this project?
- Who is my audience?
- What shape should I go for that would make the object user friendly?
- How am I going to hold my circuit inside (do I need a platform)?
- In case of transportation, how can I safeguard my circuit?
- Are there any ways for me to make my circuit work without the wire hooked up to my computer / arduino software (yes there is, but at first I didn't know that you could use a 9V battery after sending the code into the arduino)
- What electrical components should I use?

And so on.

The next step was to figure out what electrical components, sensors, and materials would improve the user experience. For the sensors, I created a brief storyboard on how the user would interact with the specified sensors (fig. 03-06) to give an idea on what the user will input as data and what the output (outcome) will be. Then from these sketches, I started doing my research on what sensors will help me accomplish my goals : record, listen, amplify, and overlap. For this part, another low-fidelity prototype was created for each one of the individual interactions by building small circuits and Arduino sketches with the goal of making sure that each sensor functions properly. The main focus was on the recording module which allows the user to record and playback a story and on the pressure sensor that triggers the output (view the section on sensors).

Without those, continuing further would be extremely difficult. Later in this project, all of those individual sensors and code will be merged for a medium fidelity prototype.

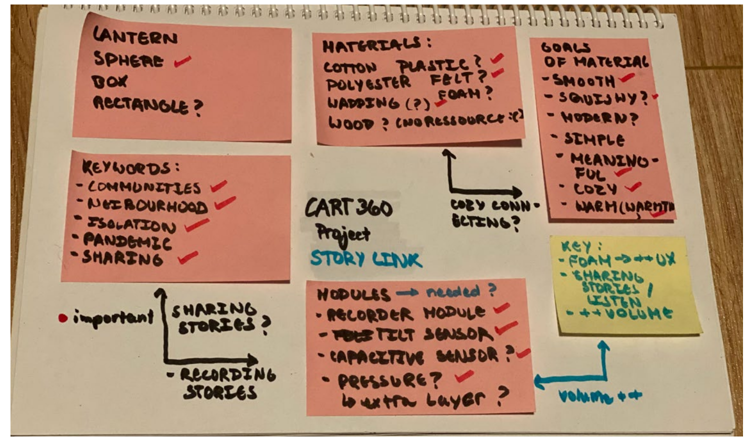


FIG.01 : Brainstorming with post-its

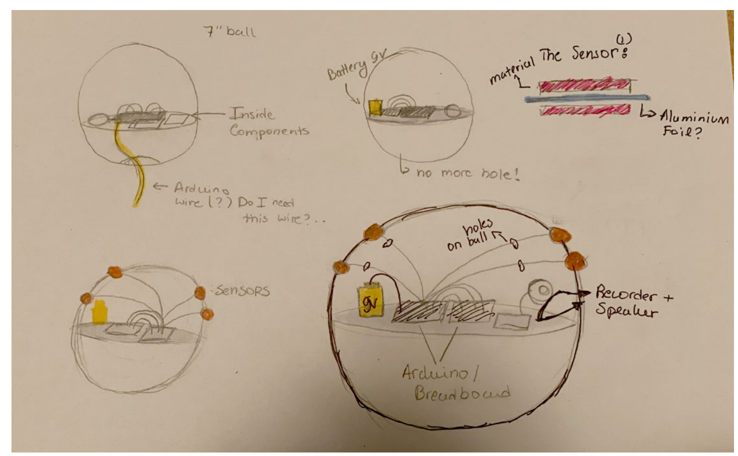
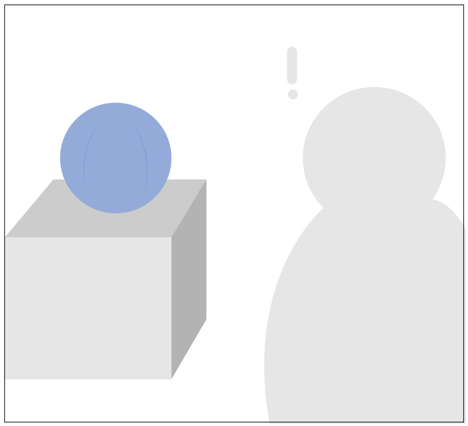
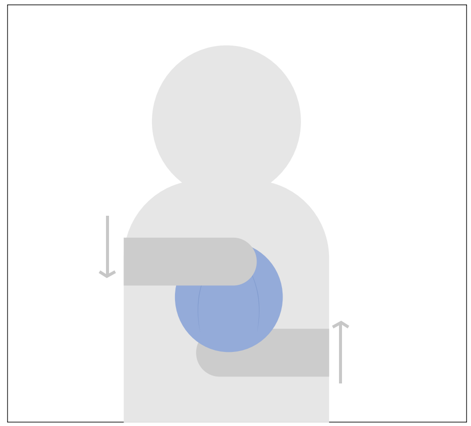


FIG.02 : Basic Sketches of Story Link's core

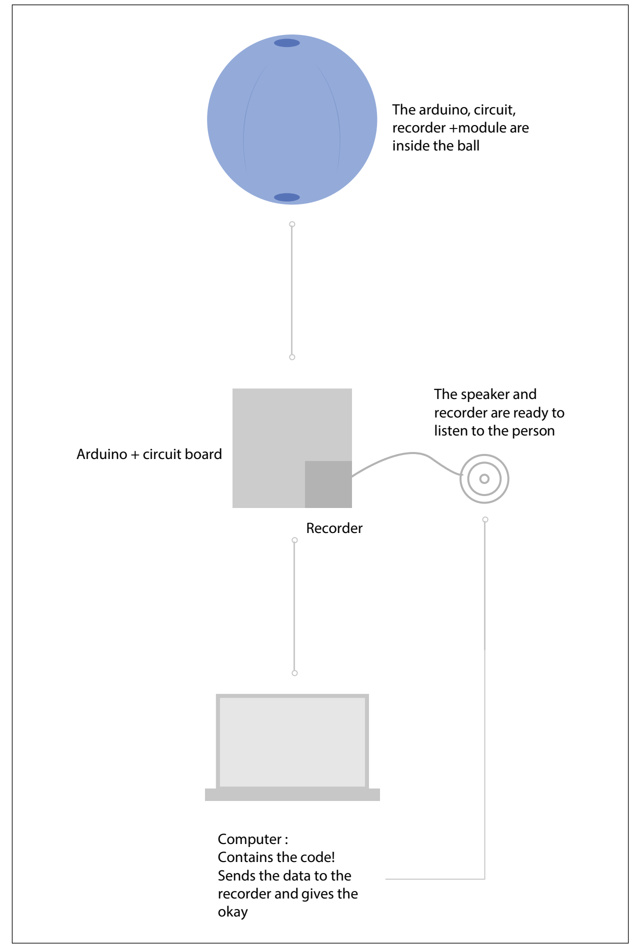
Storyboards - When the user wants to record himself



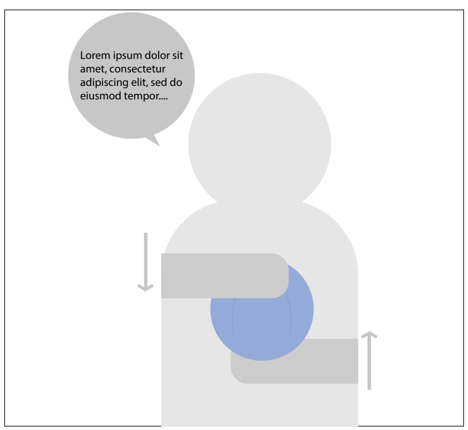
1. The user notices the ball. He wants to try to record his story.



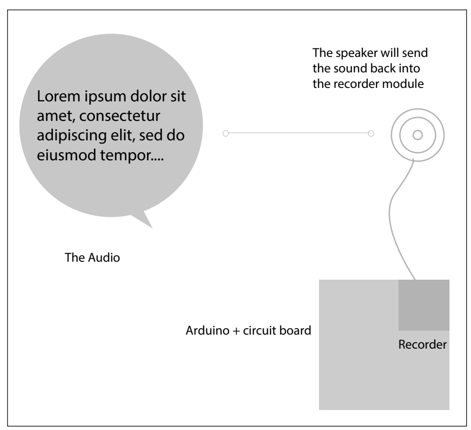
2. Applying pressure with both hands on the top and bottom (same time) will trigger the recording module



3. The sensor will send data to the arduino then to the computer that contains the code. The computer will allow the triggering to be done then send data back to the recorder module to start the record.



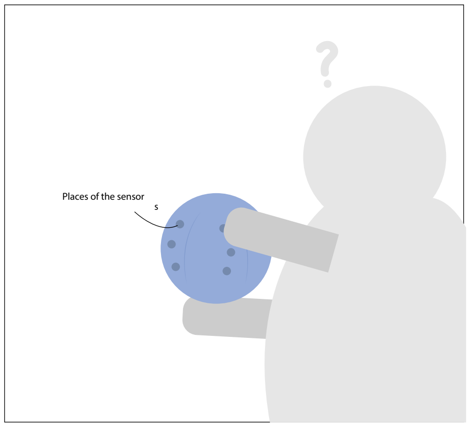
4. The user will then be able to record himself.



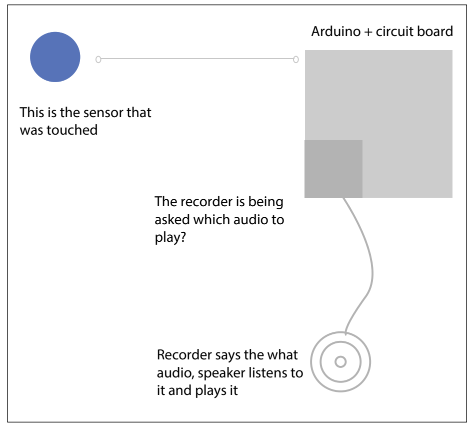
5. After finishing, the speaker will send the data to the recorder and store it inside the module.

FIG.03 : Reminder on how the recording Module works

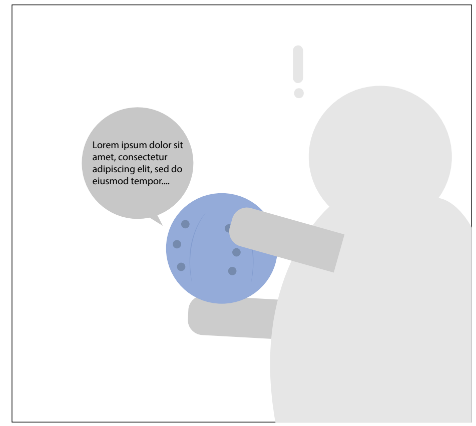
Storyboards - When the user wants to listen to the audio (Other people's stories)



1. The user is curious and would like to listen to others people's stories. Touch one of the capactive sensors on the sides.



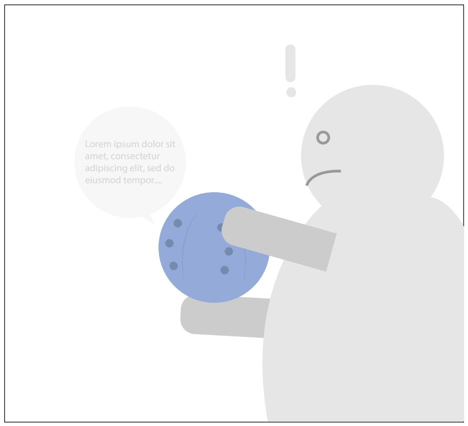
2. Upon touch, the sensor sends data to the recording module and asks which recording (from the storage) should the sensor play. The speaker listens and plays.



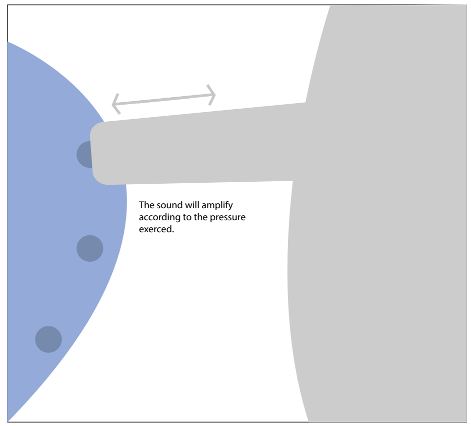
3. The speaker (inside the ball) listens to the recorder and plays what was asked. The user listen to it. Relates to that person's storiy or not.

FIG.04 : Reminder on how the listening interaction works

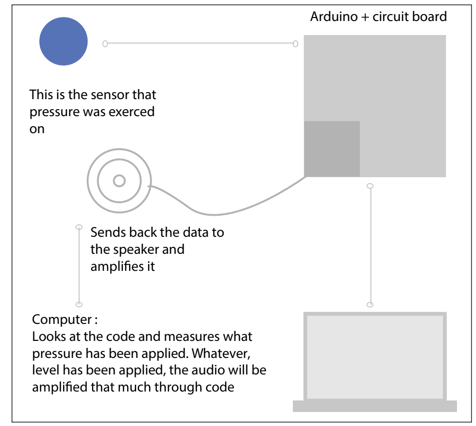
Storyboards - When the user wants to amplify the audio of one story



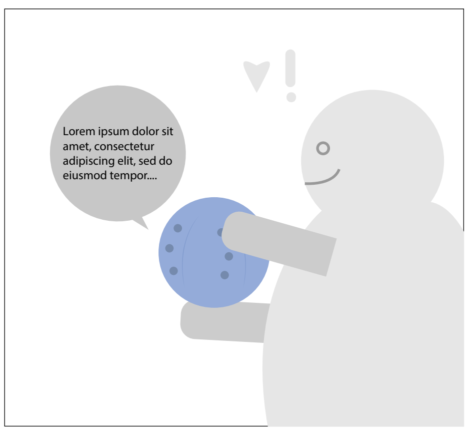
1. The user wants to amplify the sound to feel more closeness to that person when listening to that story.



2. The sound will be amplified according to the pressure excersed on the sensor.



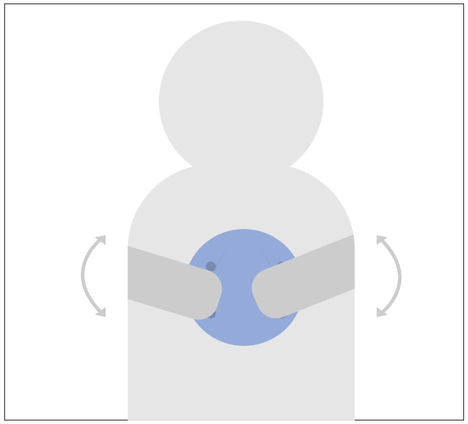
3. The pressure sensor will send the data to the arduino and to the compter. The computer will look at the code and measure what level of sound amplification matches with the pressure that the user used. The computer will send back the data back to the speaker.



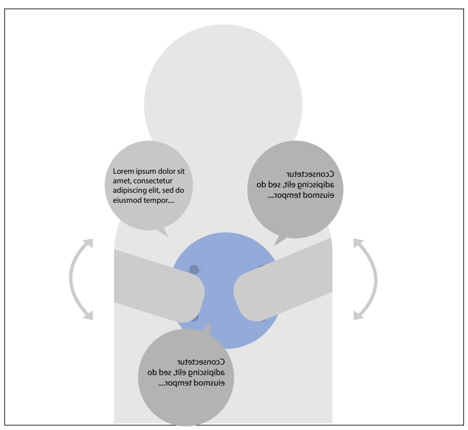
4. The speaker then projects the amplified sound! Now the user is happy.

FIG.05 : Reminder on how the amplifying volume works

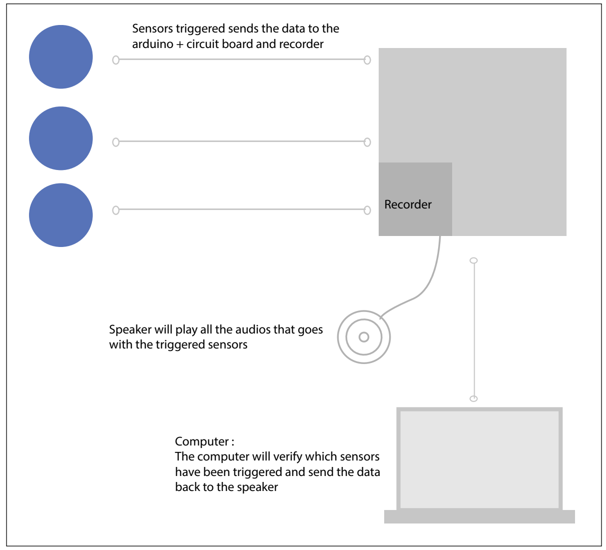
Storyboards - When the user wants to overlap everyone's recorded audios



1. If the user feels creative, sliding his hands (up and down) on the sides on multiple sensors, stories will overlap.



3. The user will be able to hear all stories at the same time and immersive himself in them.



2. The problem with this interaction is that there might be a limit at how much data at a time the sensors can send...For this one multiple inputs would be triggered at the same time and the information will be then sent to the recording module, arduino and circuits and finally the computer. The computer will verify which sensors have been triggered and send the data back to the speaker.

FIG.06 : Reminder on how the user overlaps stories

Materials Development (Continuation of Written Response (p.3))

In the proposal, the ball will be travelling from one neighborhood to another so a few questions were asked:

- How to keep the inner components from being displaced during travel?
- What materials should be used to provide good user experience?
- What material should the core be made out of?

A hamster ball made out of plastic will be used as the core housing all the microcontrollers and sensors (fig. 07). In order to keep the components in place inside the ball, a platform made out of cardboard will be fixed to the hamster ball (fig. 08). Then, once the sensors have been placed onto the core, a foam-like material will act as a protective layer that will be placed over the ball. With it, Story Link will now become a squishy storytelling ball that will intrigue users. (Fig. 09). Finally, the surface layer made out of thin wadding will be placed on top of the protective layer for aesthetic purposes (still possible to be changed but I'm looking for something soft that brings that familiar / cozy feeling) (fig. 10). Grey Thread will be used to decorate the surface layer in circular motion (fig. 11). These layers still need to be added to the prototype. As mentioned in my proposal, to match the sustainability aspect, the core will eventually be replaced with wood and the materials of each layer will become plant-based ones. However, due to the circumstances and budget issues, normal materials will be used as placeholders for the time being.

To conclude, I still consider my prototype to be of a low-fidelity level. I could say that it has advanced through individual sensor testing, however there are still issues with some sensors that need to be fixed before being able to merge all the tests together. Nonetheless, it is getting close to the mid-fidelity stage. Thanks to the multiple challenges that I have faced, I have a better understanding of what should be used in terms of materials and sensors; I could say that I have progressed in this domain. Yet the project isn't at the stage where multiple prototypes are being produced and user testing has yet to begin.

For more information on the sensors read section "sensors"

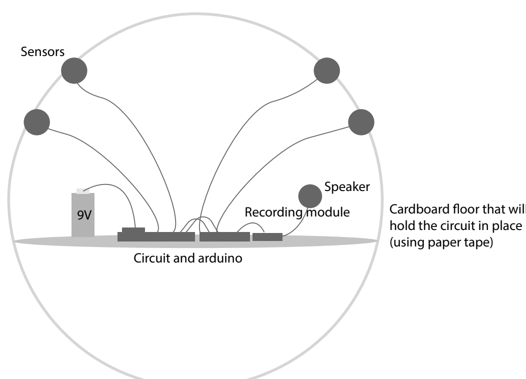


FIG.07 a : Illustration of the inside core of Story Link

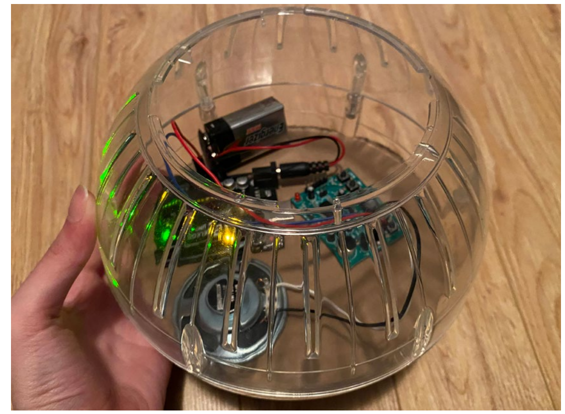


FIG.07 b : Hamster ball with platform housing the electrical components (example with the recorder)

FIG.08 : Cardboard platform inside the core done!

Materials Developement (Continuation of Written Response (p.3))

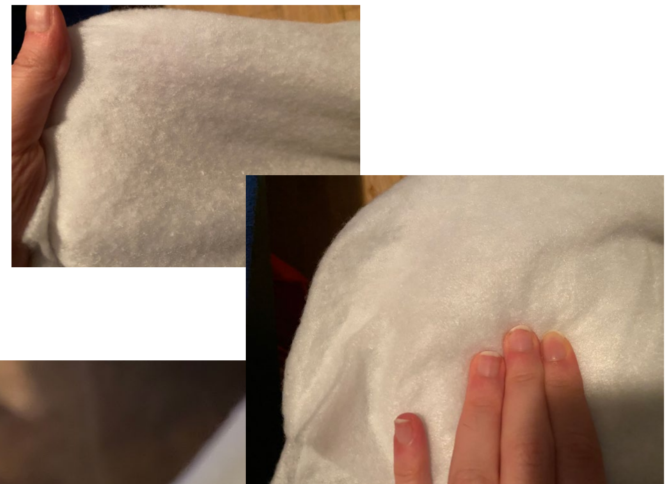


FIG. 10 : Wadding material that will be used as the surface of Story Link

FIG.09 : Thin Foam Protective layer that will give that squishy feeling to the ball

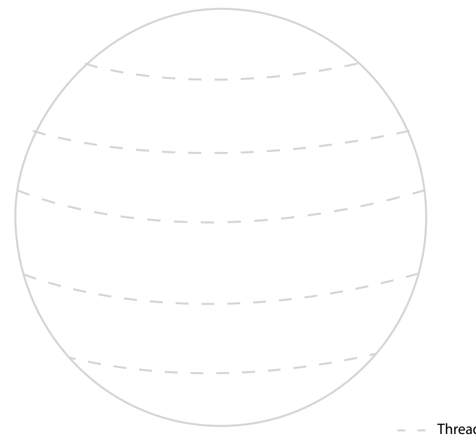
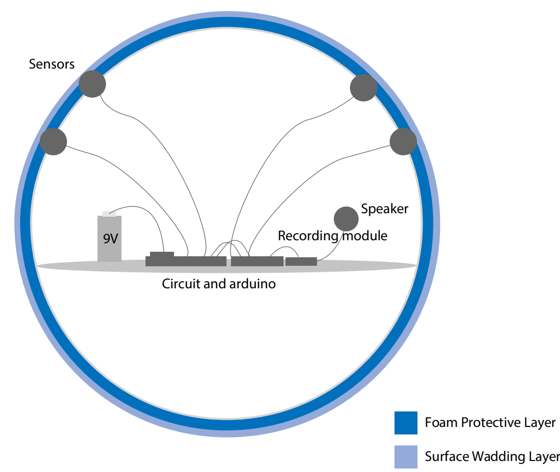


FIG.11 : Thread sewn on the ball (360 degrees)



Example of all layers on the ball and their material

Sensors

As mentioned in my last proposal, the Story Link project will provide a continuous feedback-loop relationship between the user and the artefact. The tactile sensing and audio interactions will allow the user to relate with other people through storytelling. At this stage in the project, I have a better grasp of what sensors and materials I should be using to provide a comfortable and fun experience. Here is a breakdown of the sensors:

Touch sensor

In order to detect when the person either touches or slides their hands onto the ball, which would trigger a specific output (either recording or listening), a capacitive sensor is required. Thus this type of sensor is vital for the project. In my proposal I suggested either creating my own sensor or use one called Flora. After realizing that Flora wasn't meant to be used as a capacitive sensor but rather to only trigger lights (which is not what I want), creating my own was the only option. Getting inspired by the PDF "Handcrafting Textile Sensors from Scratch" from Adafruit and a YouTuber called Nathan Villicaña-Shaw, who shows different DIY pressure-sensing techniques, I knew what path I should undertake. Therefore, the technique that I will be using is the "sandwich" (fig. 01) method. Velostat, a highly pressure-sensitive material that reduces its electrical resistance upon compression will be in the middle of the sandwich (fig. 02). Then, at the top and bottom of the Velostat, 3M aluminium foil tape, an electrically conductible material will be placed. The reason why it is placed at the top and bottom is because two wires (positive and negative) will be soldered but both sides are not allowed to directly touch each other which is why the Velostat is used. Finally, Felt material will be used to wrap the sensor and protect it. To improve the user experience upon touch, foam will be added to provide that squishiness and fun experience. The foam wasent added yet because the material that I had didnt work out (not a pleasant user experience feel).

In terms of the tactile sensors, some testing code has been made which shows values on the serial monitor (fig. 04), but the final assembly has yet to be made. However, the sensors themselves have been completed. Just like McElroy said, it is important to test the sensors separately because it is much easier to debug if there are issues (McElroy).

FIG. 01 b: Pressure Sensor Sandwich technique DIY

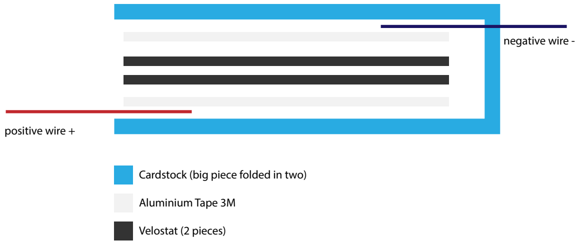


FIG. 01 b: Pressure Sensor sandwich illustration technique

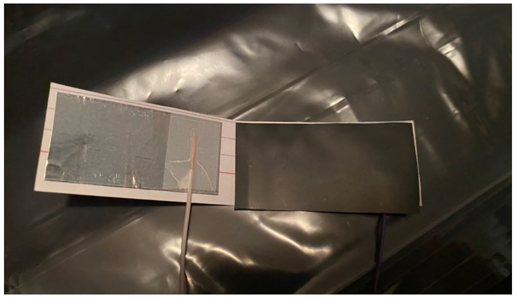
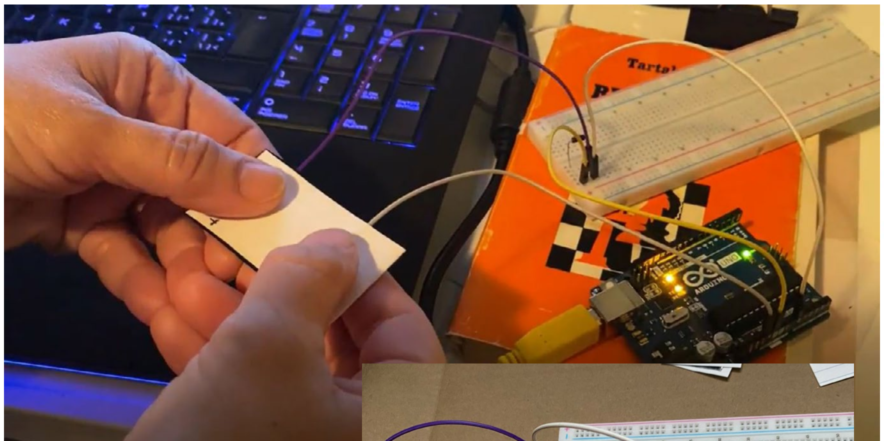
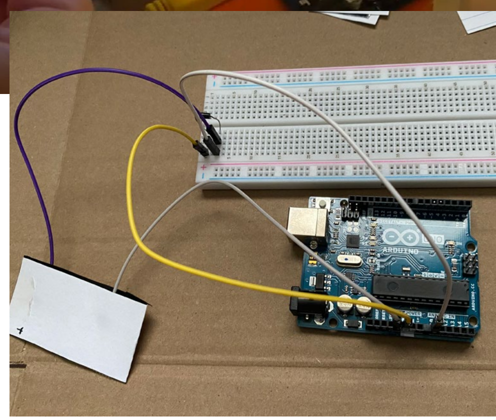


FIG. 02: Pressure Sensor sandwich with Velostat added

Touch - Pressure Sensor





```
Lightly Tap
Lightly Tap
Lightly Tap
Lightly Tap
Lightly Tap
Lightly Start Touch
Lightly Tap
Lightly Tap
Lightly Start Touch
Lightly Tap
Lightly Tap
Lightly Tap
Lightly Start Touch
Lightly Tap
Lightly Tap
Lightly Start Touch
Lightly Tap
Lightly Tap
Lightly Tap
Lightly Start Touch
```

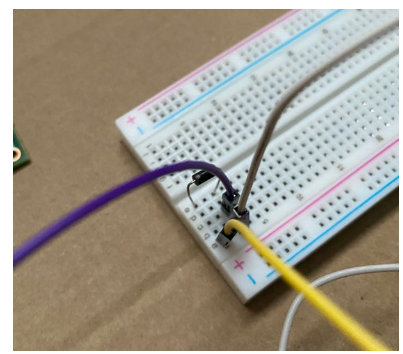
Pressed lightly

```
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
Strongly Hold
```

When the person presses it hardly

FIG 04: Serial monitor that recognizes when it has been touched or not and intensity.

```
//learned from https://www.youtube.com/watch?v=aXS2blhIyGA
#include <math.h>

int sensorOneR = 0;  //sensor one is 0
int touching = false; //you are not touch it
int touchingCount = 0; //count the touch

void setup() {
  Serial.begin(9600);
}

// the loop function runs over and over again forever
void loop() {
  int sensorValue = analogRead(sensorOneR); //read sensor One
  String amount = "Start Touch";

  //if the sensor value is more than 90 consider it as being touched so add +1 to the counter
  if (sensorValue > 90) {
    touching = true;
    touchingCount++; //add one to the count
  } else {
    touching = false;
    touchingCount = 0; //if not touching, reset
  }

  //if touch AND touch count is smaller than 20, then show "tap
  if (touching && touchingCount < 20) {
    amount = "Tap";
  //if its only "touch", show hold
  } else if (touching) {
    amount = "Hold";
  }

  //sensor value is smaller than 90, print "not touched"
  if (sensorValue < 90) {
   //if the value is smaller than 120 show lightly touched
   //    Serial.println("Not touched");
  } else if (sensorValue < 120) {
    Serial.println("Lightly " + amount);
  //if the value is smaller than 160 show strongly touched
  } else if (sensorValue < 160) {
    Serial.println("Strongly " + amount);
   //if the value is smaller than 190 show hardly touched
  } else if (sensorValue < 190) {
    Serial.println("Hardly " + amount);
  }
```



Code inspired from an Arduino class on Youtube on how to make pressure sensors. The "Diode" is used as a sort of protection.

Recording / Playback module

Just as mentioned in my proposal "touching the top and bottom of the sphere with both hands at the same time enables the recording feature". My first suggestion of recording module was using the "ISD1820 Voice recorder Module" (Fig. 05). However, this recorder only records 20 seconds which is not a lot. So my second try was the "Velleman MK195 Voice recording module" (fig. 06). This one had a recording time of 60 seconds which was a great option but had to be assembled. During the soldering process, the iron burned one of the pads of the PCB so a new alternative had to be found. Finally, I opted for the ISD240 Recording module, much smaller size-wise (which is good) that allows the user to record themself for 240 seconds and was already assembled (fig. 07.) The mic sends input to the board and the VCC sends to 5V on Arduino. Then GND from the module sends the data to GND into the Arduino. This data is sent back into the speaker. Using the speaker that came with the "Velleman MK195 Voice recording module", I was able to make the module work during testing. The only thing left to do is to extend the mic from the recording board so it can be much louder when it will be into the ball (better UX).

**Update: This part will be the most challenging coding. The recorder works along with the speaker, but now we need to find a way to control it through Arduino. ISP library would be an answer, but I'm still trying to understand how it works. However, upon writing this documentation, I found this:

https://create.arduino.cc/projecthub/tsaritsynskyyaa/software-for-recording-audio-files-into-isd1700-chips-fad583

Useful Links that will be good for ISP research:

https://forum.arduino.cc/index.php?topic=37348.0 https://forum.arduino.cc/index.php?topic=37348.0

https://robojax.com/learn/arduino/?vid=robojax_ISD1700_sound_recorder

https://github.com/georgepatterson/ISD17xx

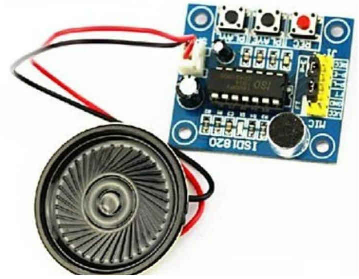


FIG.05 : Image of : ISD1820 Voice Recorder Module with a speaker (Didn't work out..)
source: https://www.instructables.com/HOW-TO-USE-ISD1820-VOICE-RECORDER-AND-PLAYER/

FIG.06 : Image of :"Velleman MK195 Voice recording module" with a speaker (Solder iron burned it...)
source: https://www.velleman.eu/products/view?id=412506

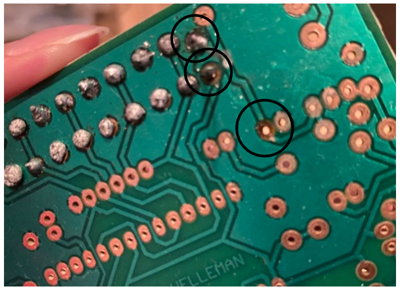


Image of :"Velleman MK195 Voice recording module" (Solder iron burned three pads)
source: https://www.velleman.eu/products/view?id=412506

ISD240 Recording module

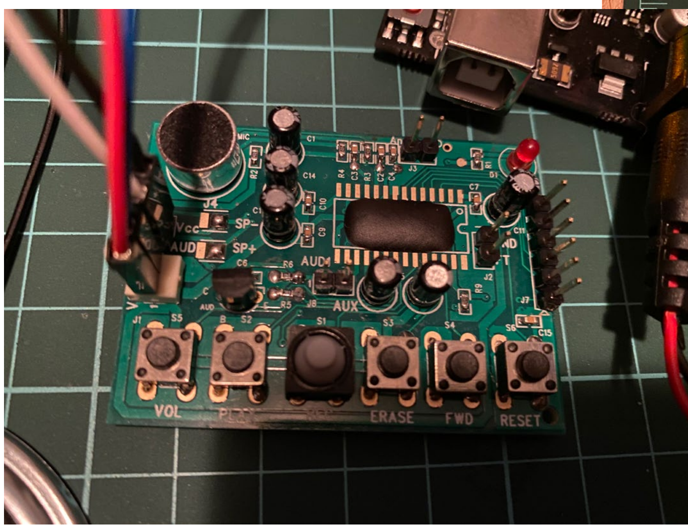
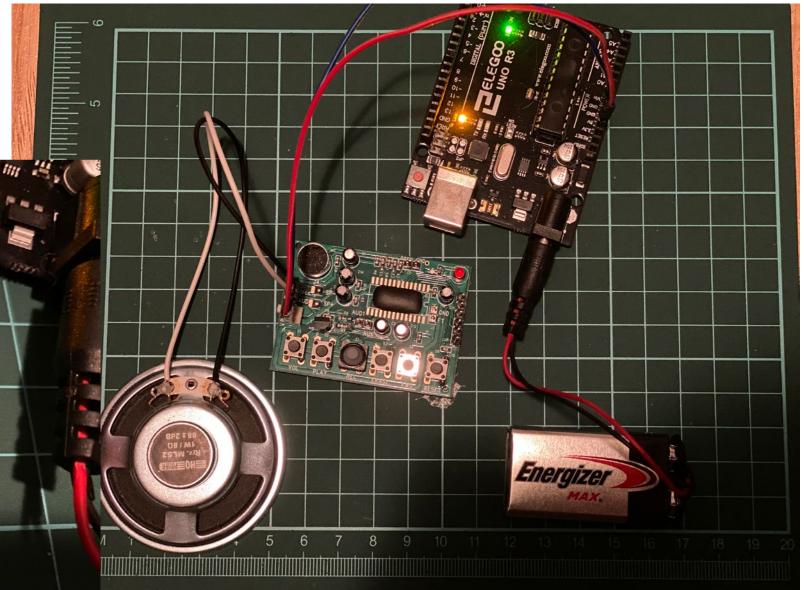
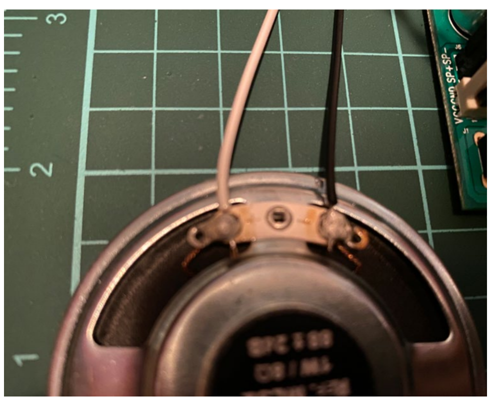


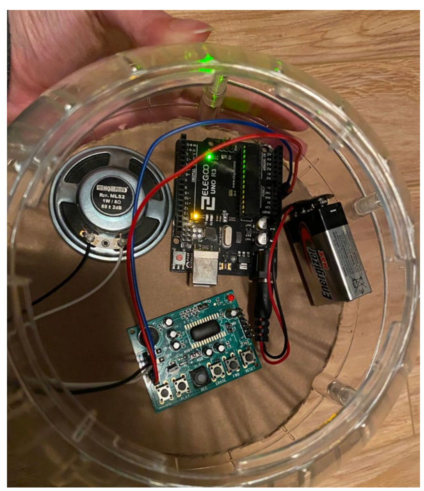
FIG.07: ISD17240 Recording module that records 240 seconds



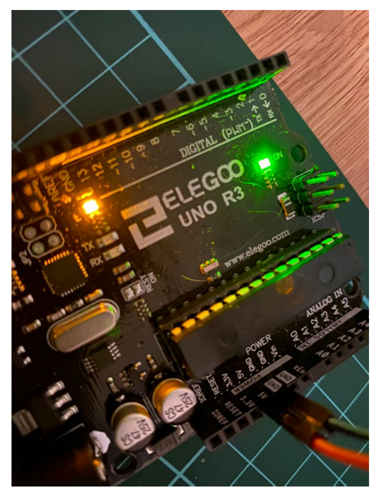
Full Image of the Recording module connected to a 9V battery



Speaker with two wires soldered



Test of the Recording Module in the ball. Seems pretty crowded. The speaker will be glued to one of the sides vertically.



Amplifying the playback volume

In correspondence to the pressure applied onto the sensors, the user will be able to increase or decrease the volume of the audio playback. This part made me think that maybe I could merge the volume variation with the pressure sensors that I created (fig. 08). Due to the amount of time that it took me to get my Velostat and other materials, this would be the next step for the pressure sensor. Here is an idea of the code:

-It check values of each sensors (~90 when untouched , ~150 pressed hard)

-It applies a threshold and conditions (If the sensor value is bigger than 200, then apply the amplification property. Else, act as a normal sensor).

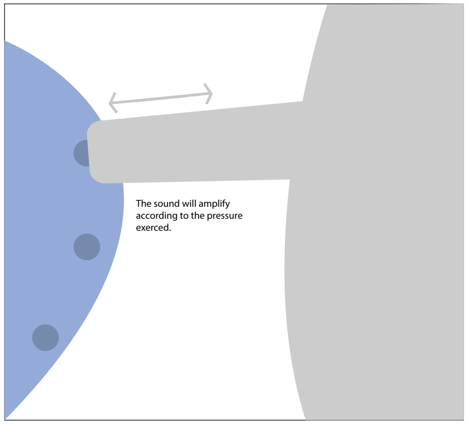


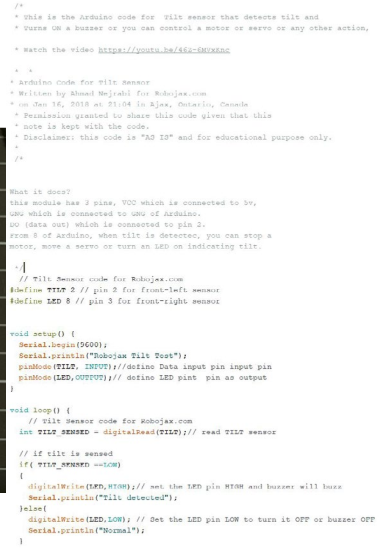
FIG. 08: The sound will be amplified according to the pressure excersed on the sensor.

Overlapping / tilt

The overlapping and tilting experience would be purely experimental but will allow the user to play with stories just as mentioned in the proposal. For instance, if the user tilts the ball to the right, all the stories on the right side of the ball will start playing and overlap each other. The sensor that will allow me to do that is a “tilting sensor” (fig. 09) that will react to certain conditions described in the code. The sensor’s metal cylinder that detects which side the user has tilted the sensor will send data to the module and to the code. Based on the angle and commands in the code it will output a specific outcome (In my case stories on the tilted side will start playing). The good part is that it’s also tiny which is good because the ball isn’t super big. Some testing with code has been done (fig. 10 - 11) but I believe that this part still needs to be explored and perhaps will be tested in-depth once the other interactions are fully implemented. However, it makes me think that instead of allowing the user to just slide their hands up and down onto the ball, titling it from left to right will make it a funner experience.

Note: This code is meant for testing and a tutorial has been followed. I will have to make the tilt sensor work once I will be having one pressure sensor on each one of the ball. Its a good step though.

FIG. 11: Test Arduino code



```
/*
 * This is the Arduino code for  Tilt sensor that detects tilt and
 * Turns ON a buzzer or you can control a motor or servo or any other action,

 * Watch the video https://youtu.be/46Z-6MVxKnc

 *  *
 * Arduino Code for Tilt Sensor
 * Written by Ahmad Nejrabi for Robojax.com
 * on Jan 16, 2018 at 21:04 in Ajax, Ontario, Canada
 * Permission granted to share this code given that this
 * note is kept with the code.
 * Disclaimer: this code is "AS IS" and for educational purpose only.
 *
 /*


What it does?
this module has 3 pins, VCC which is connected to 5v,
GNG which is connected to GNG of Arduino.
DO (data out) which is connected to pin 2.
From 8 of Arduino, when tilt is detectec, you can stop a
motor, move a servo or turn an LED on indicating tilt.

 */
  // Tilt Sensor code for Robojax.com
#define TILT 2 // pin 2 for front-left sensor
#define LED 8 // pin 3 for front-right sensor


void setup() {
  Serial.begin(9600);
  Serial.println("Robojax Tilt Test");
  pinMode(TILT, INPUT);//define Data input pin input pin
  pinMode(LED,OUTPUT);// define LED pint  pin as output
}

void loop() {
    // Tilt Sensor code for Robojax.com
  int TILT_SENSED = digitalRead(TILT);// read TILT sensor

  // if tilt is sensed
  if( TILT_SENSED ==LOW)
  {
    digitalWrite(LED,HIGH);// set the LED pin HIGH and buzzer will buzz
    Serial.println("Tilt detected");
  }else{
    digitalWrite(LED,LOW); // Set the LED pin LOW to turn it OFF or buzzer OFF
    Serial.println("Normal");
  }
```

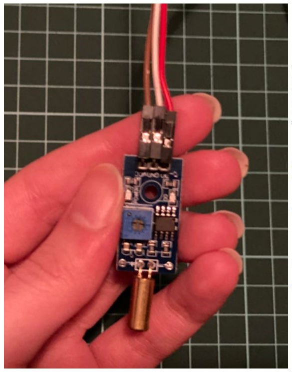


FIG. 09: Tilt Sensor

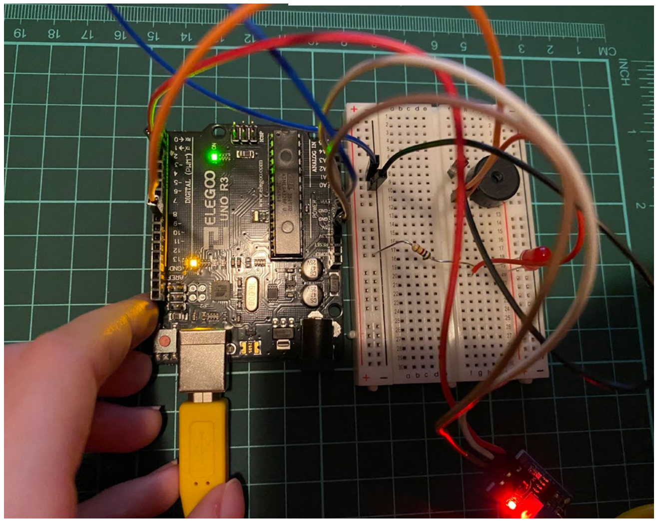
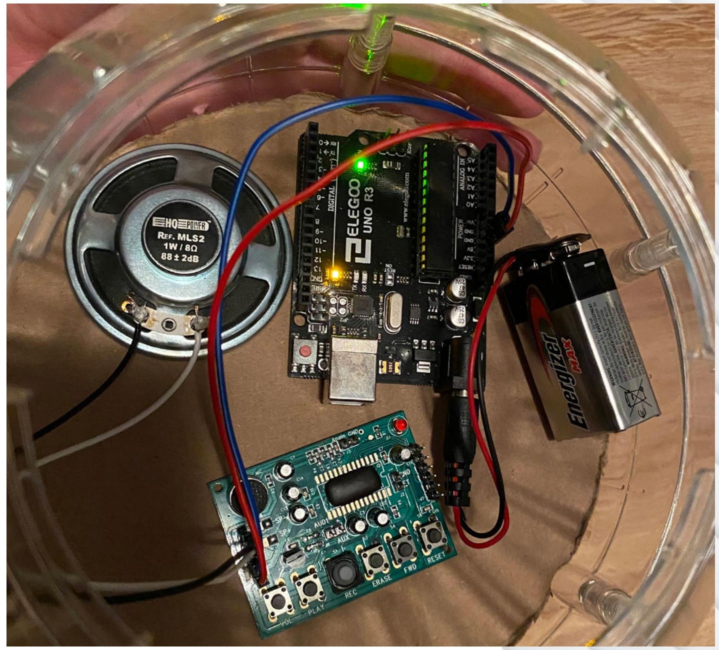


FIG. 10: Test made with an LED. When the sensor detects a tilt, it will buzz and light up.

Has your Project's initial intention or supposed meaning changed over the course of researching and implementing the Physical Prototype? If YES or NO – Explain why?

My intention remains unchanged from when I started writing this proposal. The goal of this project is to make a device that challenges the traditional ways of journaling while consuming no paper. With this device, people can tell their own stories through the recording device or listen to other people's stories to feel connected to their community. As a reminder, Story Link will be travelling from a neighborhood to another so people can hear about other people's experiences. The use of this device will empower communities, bring neighbours together, and perhaps even break the walls of isolation for people living alone. The only part that has changed are the sensors, materials, and the tilting interaction that might replace the action of sliding one's hands up and down. Also, understanding what materials I should be using made me realize that perhaps I should create Story Link with a squishy surface to improve the user's feeling of closeness and warmth.



Ressources (That links are not mentioned above):

DIY Pressure Sensor and Code:

https://youtu.be/SLRYX879Py0

https://youtu.be/aXS2b1hlyGA

https://cdn-shop.adafruit.com/datasheets/HandcraftingSensors.pdf

Recording Module 1SD17240

https://forum.arduino.cc/index.php?topic=37348.0 https://forum.arduino.cc/index.php?topic=37348.0

https://robojax.com/learn/arduino/?vid=robojax_ISD1700_sound_recorder

https://github.com/georgepatterson/ISD17xx

https://create.arduino.cc/projecthub/tsaritsynskyyaa/software-for-recording-audio-files-into-isd1700-chips-fad583

Tilt Sensor:

https://www.youtube.com/watch?v=46Z-6MVxKnc

Why We Prototype

https://learning.oreilly.com/library/view/prototyping-for-physical/9781492042440/ch01.html