

FINAL PROJECT FOR INTRO TO IM

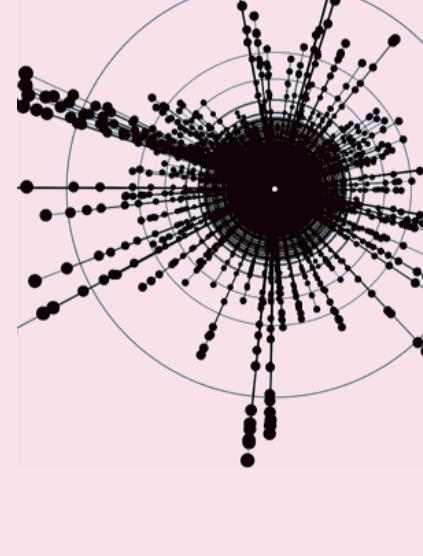
LI CHENG



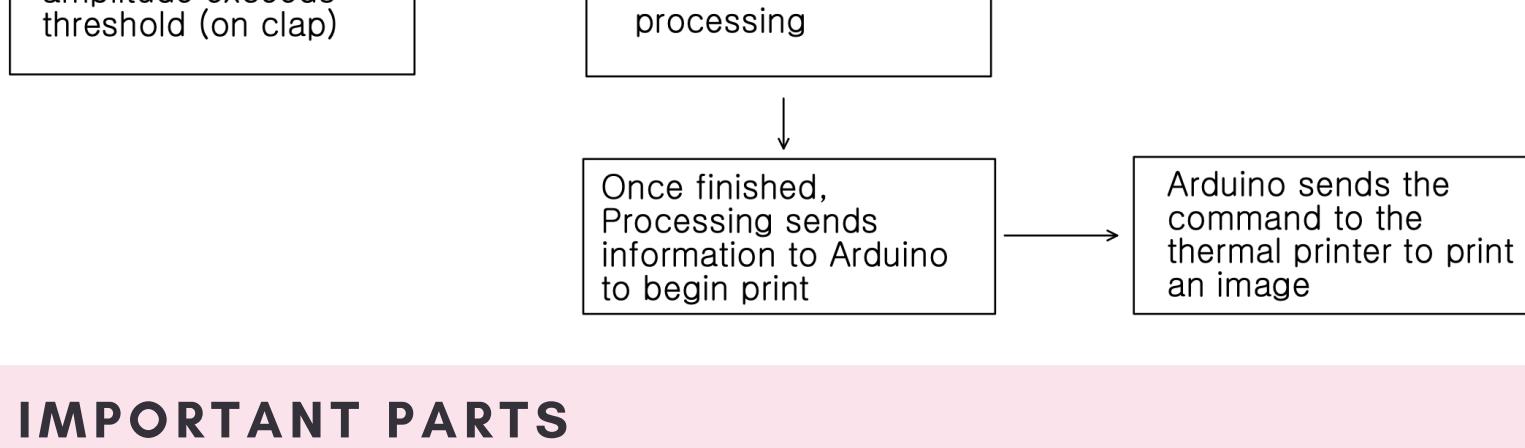
THE CONCEPT

This project uses audio visualization to create a tangible manifestation of a conversation between two friends. By prompting reflective questions within a peaceful space, it aims to spark an honest and appreciative exchange, resulting in a snapshot of a friendship.

Different from photo booths, which print a single moment to be kept as a memory, the friendship booth requires an entire conversation to occur, which is instantly processed on the screen and finally printed on receipt paper.



SYSTEM DIAGRAM



IMPORTANT PARTS

- frame made of acrylic
- IPad used as display
- adafruit thermal receipt printer
- laptop for running processing and Arduino

CHALLENGES AND SOLUTIONS

The main challenge was to send the image from processing through Arduino and to the printer. This was difficult for two reasons: the Arduino can only store a limited amount of data, barely enough for one image. Even using an Arduino Mega could not have been enough for a single larger image. Furthermore, the program memory cannot be used to store dynamically allocated data, so the entire array would have to be created at once. Moreover, the built-in print function is also designed for predefined data arrays and not for serial streams. These circumstances made it very difficult to send a new image to the printer in each round. In the end, I simply stored two compressed bitmaps in the Arduino program which are printed in alternating order, rather than printing the actual image from the conversation.

Another challenge was in building the frame. Inspired by retro photo booths, I planned to make the frame out of wood. However, with little knowledge in woodworking, this proved to be quite more difficult than I expected. For reasons of time efficiency, I decided to use black acrylic (as it matched best with the display) and used a laser cutter to create the box.

HOW IT WORKS

The processing program measures the amplitude constantly. When the amplitude exceeds a threshold, triggered by clapping, the drawing function runs. It draws the friendship circle on the display, and depending on the amplitude different patterns are created. After three questions are displayed and the time is over, processing sends a signal to Arduino via serial communication. Depending on the signal, Arduino sends a command to the receipt printer to print out one of the two stored files. The guests can easily take this receipt from the box and keep it as a memory.

USER TESTING

I user testing to see if the questions "worked", meaning if they actually sparked the kind of honest conversation I had in mind, as well as to see set the time lengths for each question. The interaction itself worked well; people intuitively understood that what they saw on the screen was related to their conversation.

I measured how long it took people to answer the prompts without a set time limit. The results differed quite significantly as each friendship is unique and will have more or less extensive answers to these questions. The times for each answer were between 20 seconds and five minutes. In the end, I settled for a time of a little less than a minute per question, because I wanted to prevent people from running out of things to say. If people want to give longer answers, of course it can be done outside of the friendship booth as well, so it can be provide the starting point for a more extensive conversation.

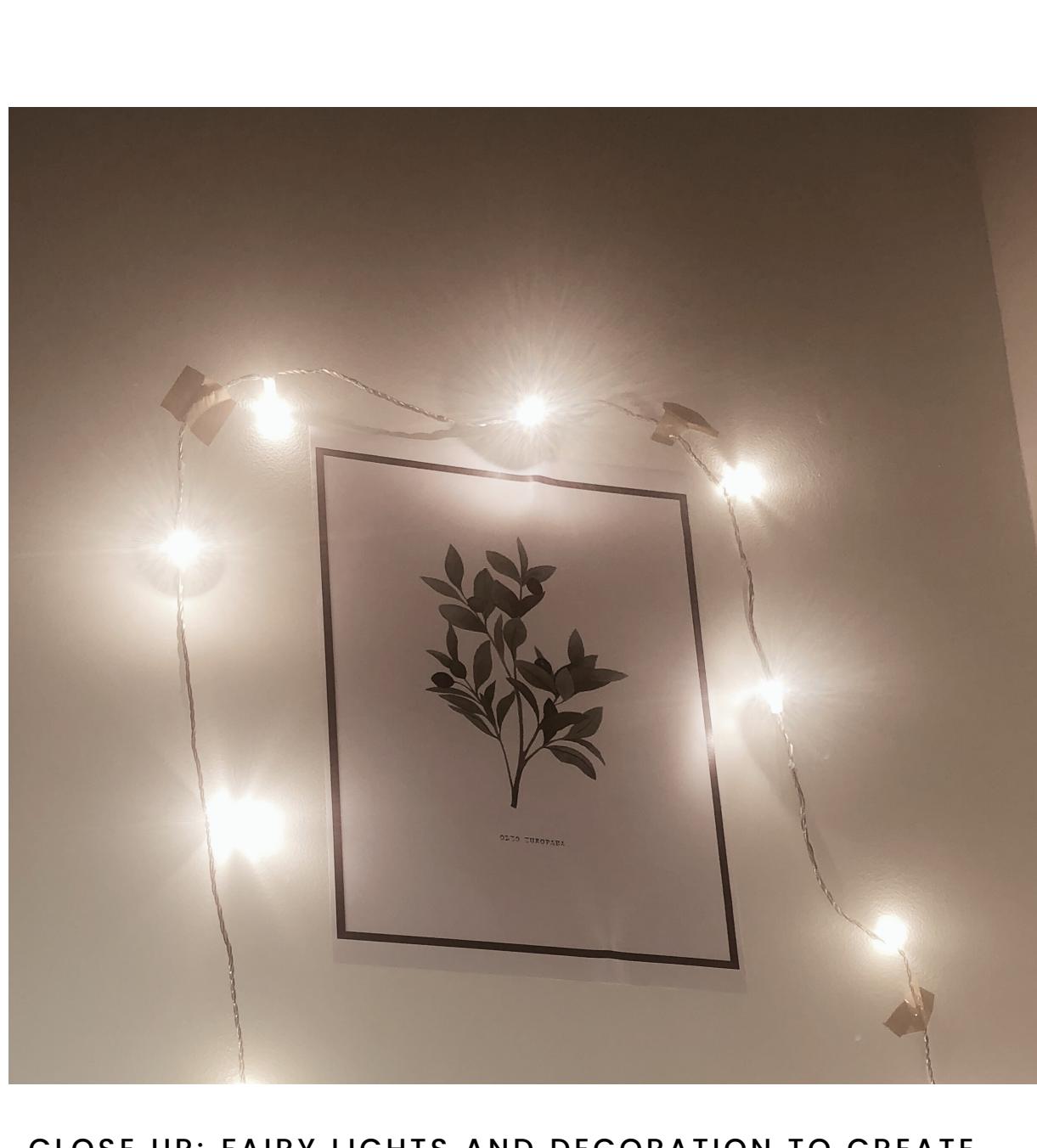
CODE

The code for this project can be found at https://github.com/introCSLi/homework-IM/tree/master/final_documentation.

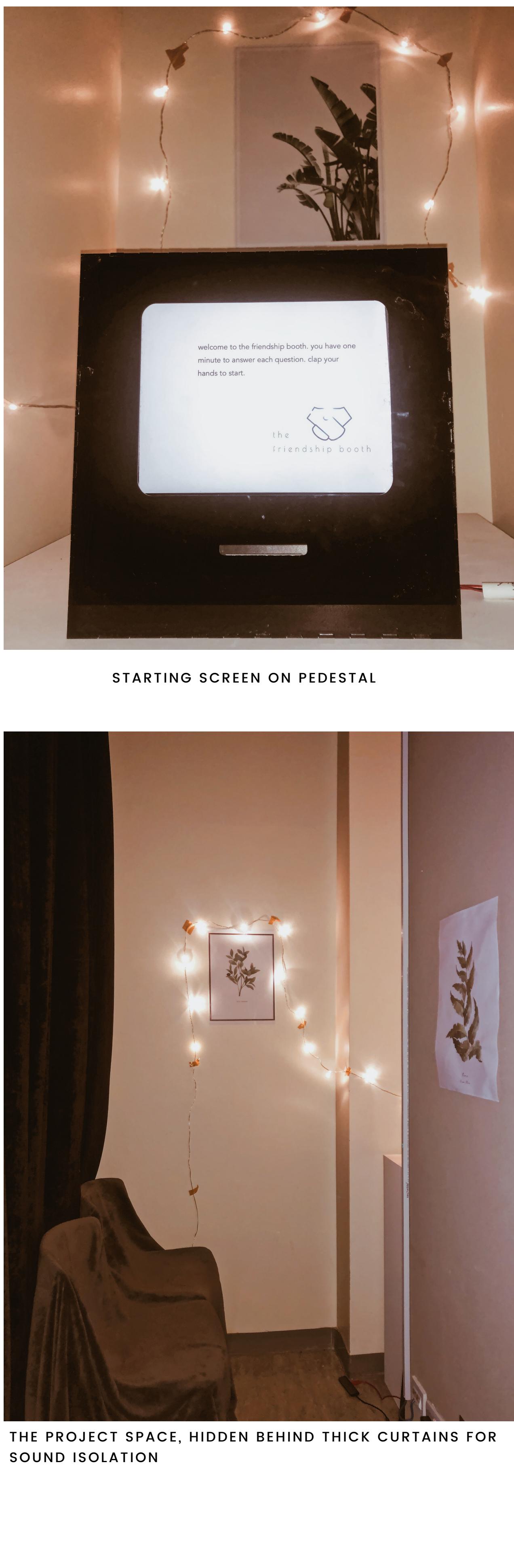
IMAGES



TWO GUESTS TESTING OUT THE FRIENDSHIP BOOTH



CLOSE UP: FAIRY LIGHTS AND DECORATION TO CREATE WELCOMING ATMOSPHERE



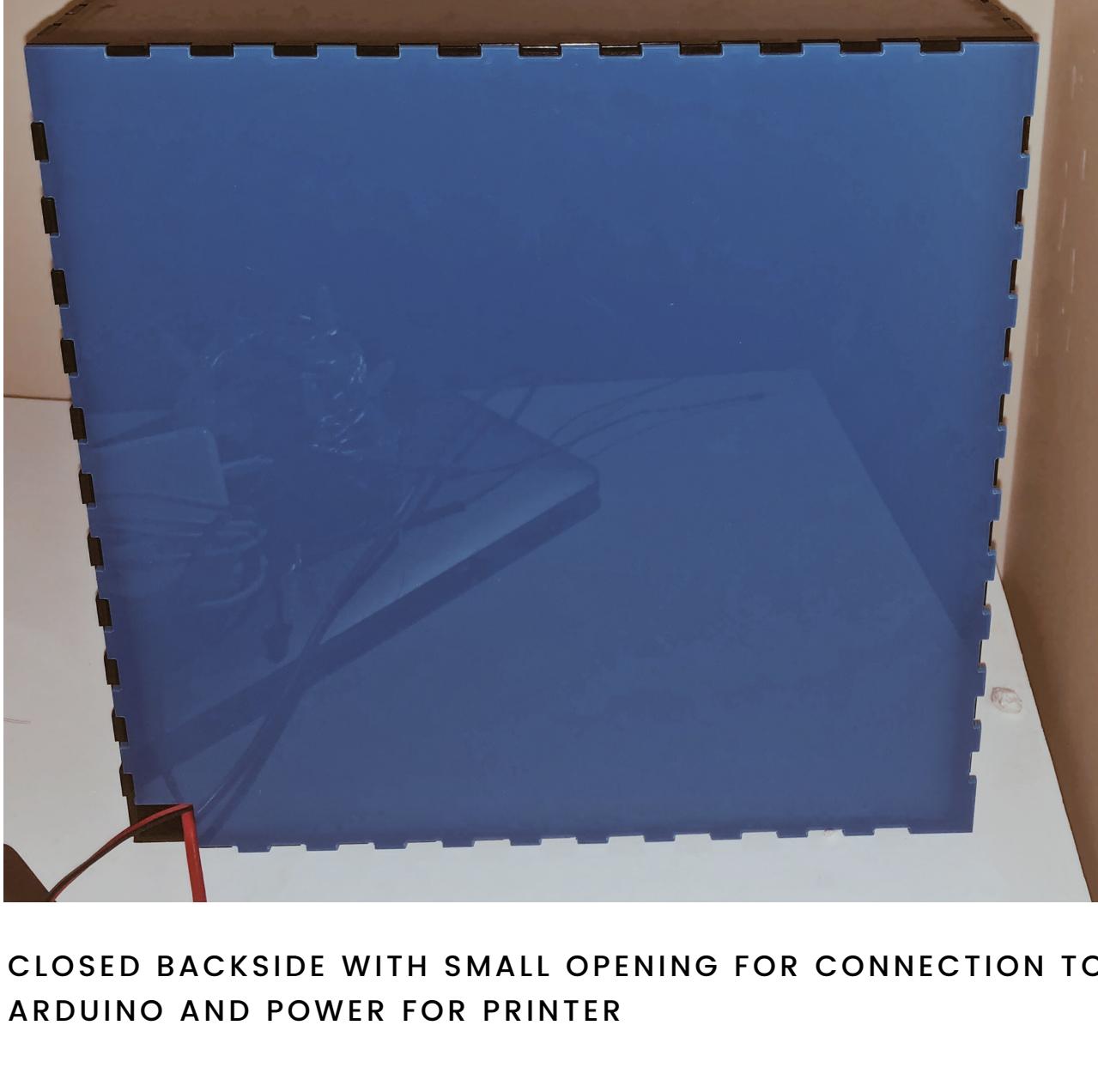
STARTING SCREEN ON PEDESTAL



THE PROJECT SPACE, HIDDEN BEHIND THICK CURTAINS FOR SOUND ISOLATION



INSIDE THE FRAME: I PAD MOUNTED USING SUPPORTING PLATFORM AND VELCRO, RECEIPT PRINTER AND ARDUINO FIXED USING DOUBLE SIDED TAPE



CLOSED BACKSIDE WITH SMALL OPENING FOR CONNECTION TO ARDUINO AND POWER FOR PRINTER