ECE 198 Project: Design Document

REAXION

Written by:

Lucy Sun,
1A Electrical Engineering, University of Waterloo
Alex Wang,
1A Electrical Engineering, University of Waterloo
Lakshya Rao,
1A Computer Engineering, University of Waterloo

Section 1: Operational Concept Description

Section 1.1 Required Components

The ***Reaxion*** project is built upon the STM32 Nucleo 64 development board with supporting components. The list below shows all the components that one would need to build this project for themselves along with their approximate prices and web links for purchase.

Table 1.1 - The required components

Component Name	Quantity	Price ($)	URL
STM32 Nucleo 64 Development Board	1	41.59	*https://amzn.to/3ChvkkN*
Multi-color LEDs	6	17.98	*https://amzn.to/3GtUyio*
Current limiting resistors	10	16.89	*https://amzn.to/3bocLPT*
TFT LCD Display	1	44.67	*https://bit.ly/3CGHiEH*
Jumper Wires	20	12.99	*https://amzn.to/APHL9E*
Breadboard	1	14.99	*https://amzn.to/3GCjqV3*
Buttons	6	12.09	*https://amzn.to/3mqNZoV*

Section 1.2 Operational Concept Description

Reaxion is a physical game designed to test the reaction times of the player, which is based primarily on the STM32 microcontroller. The product uses multiple other components such as multi-color LEDs, an LED display, a combination of LEDs and buttons. As *Diagram 1.1* below shows, the logic of the game is not very complex; the LED matrix displays the name of a color (C_n), the user then presses the button corresponding to that color on the keypad (B_m), all in under 0.5 seconds (t is the time the user presses the button). If the user presses B_n in the required time, one of the six LEDs turns on - these LEDs act as a scoreboard, with S representing the score of the player (where $S \in [0,6]$). If the user fails to press B_m in $t \leq 0.5s$ and $S \geq 1$, then S is reset to 0. And another variable R is to record how many times the game restarts. Every time that the game restarts, R will add one on it. If the number of R is over 3, the game is over. However, if S successfully reaches 6 without R passes 3, the game wins. Diagram 1.1 (bottom) displays the logic.

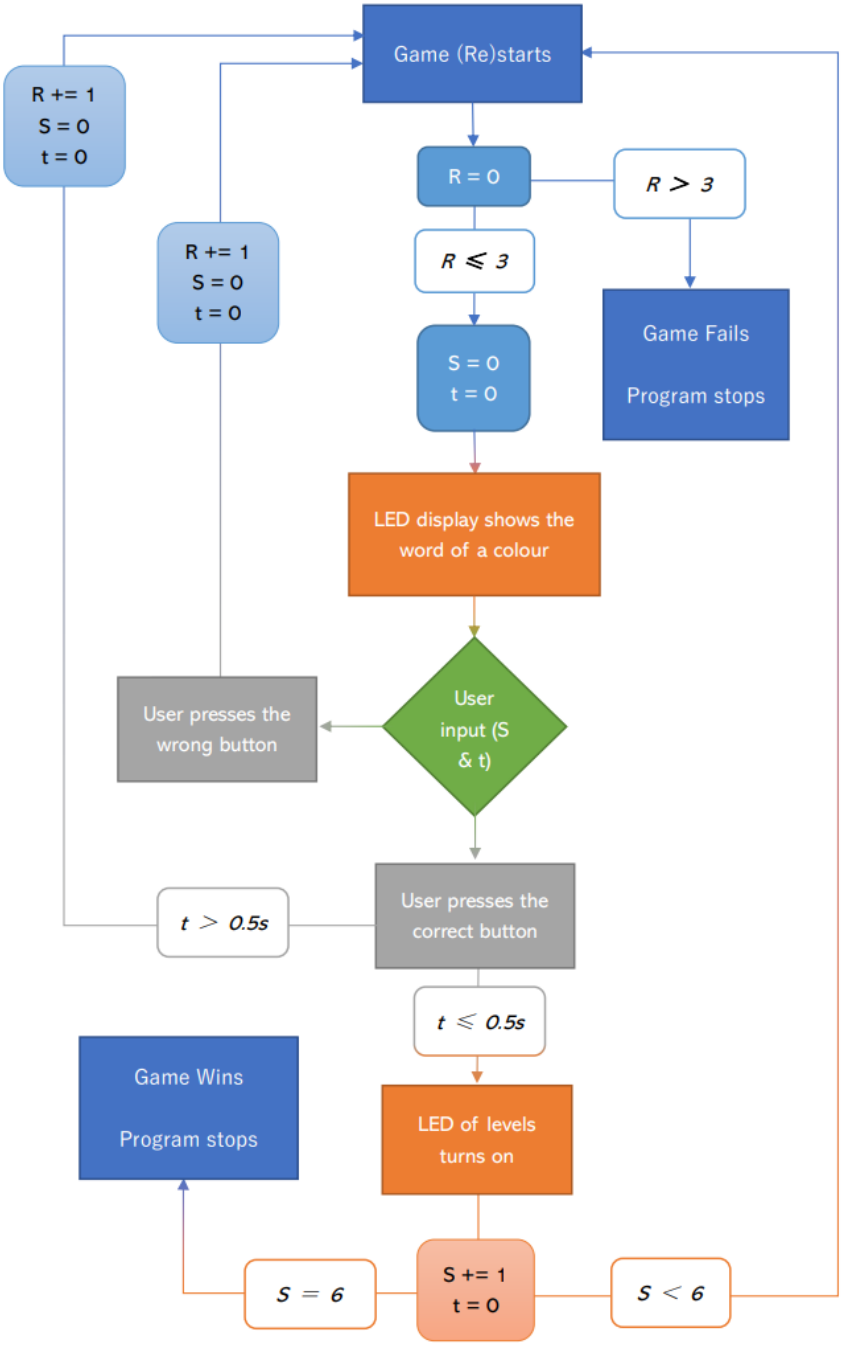

Diagram 1.1 - The logic of operation

Section 2: Design Specifications

2.1 Design description

2.1.1 Components description and connection

In our design, appropriate components are required for implementation. Because this is an escape puzzle, we need to use some LEDs to show the level of the game, and LEDs to display the colors. To obtain the reaction time, we aim to use push buttons. All the code will be downloaded on the development board for the game to run. The current limiting resistors are to control the current that flows in the circuit in a safe amount to avoid damage to the LEDs. We decided to use the TFT LCD display according to our design, this is the best choice to show the words in different colors with colorful backgrounds. The components will be assembled on a breadboard and will be connected using jumper wires.

For the container of our product, we plan to use 3-D printing technology for aesthetic and ease of use purposes. To make the product easy to store, we add a bracket and a reversible part of the container so that the LED display can withdraw and stay oblique for better sight. In addition, somewhere below the reversible container and the connection points between the bracket and the reversible container, we need to add axes of rotation to make sure they can reverse. Moreover, to ensure the stability of the inclined display, there are some anti-slip cushions added. *Diagram 2.1* and *Table 2.1* exhibit the components and the way they will be connected and place

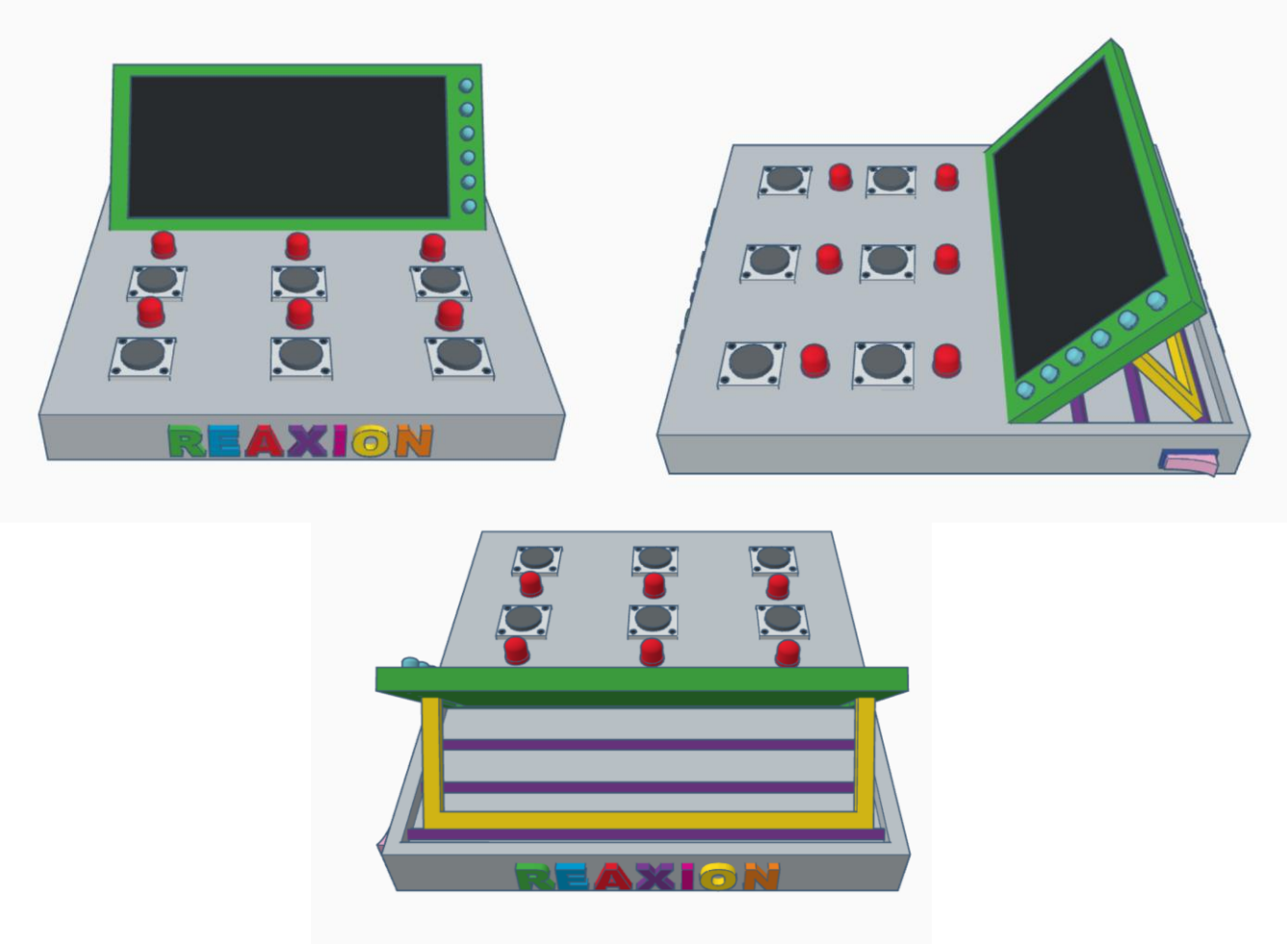

Diagram 2.1 - Three viewing angles of the model product (Build with TinkerCAD by Lucy)

Table 2.1 - The match of components and connection in the model

Components Color	Components Name
Black	TFT LED display
Green	Reversible container
Light Blue	LEDs (show the level)
Red	Multi-color LEDs (the color is changeable)
Dark Grey	Buttons
Light Grey	Main container
Pink	Start switch
Yellow	Bracket
Purple	Anti-slip cushion
Colorful Text	Reaxion (name of our product)

2.1.2 Software description

The working process has been shown in the logic of operation. There are two variables: the time the user took to react and the times that they correctly press the button with the right color. We are going to code a program (Program #1) to run logic (the data collection and data comparison), and another program (Program #2) to randomly show colored words with colorful backgrounds will be coded. For example, if the word shown on display is “Green”, but with a yellow color and blue background, the user ought to press the button with the green LED on within 0.5 seconds to successfully pass the level. Program #1 will collect and check the color the user chose (condition #1) and the time the user took (condition #2). The correctness of condition #1 is determined by the output word of Program #2 which decides if the game should restart or keep checking condition #2. After passing condition #1, by comparing condition #2 with the default length of time (0.5s), the program will either keep running or restart the game. After the game restarts 3 times, it means the user has failed the game. Otherwise, if the user successfully passes all 6 levels within 3 restart chances, he/she wins the game. *Diagram 2.2* below represents the software’s working process.

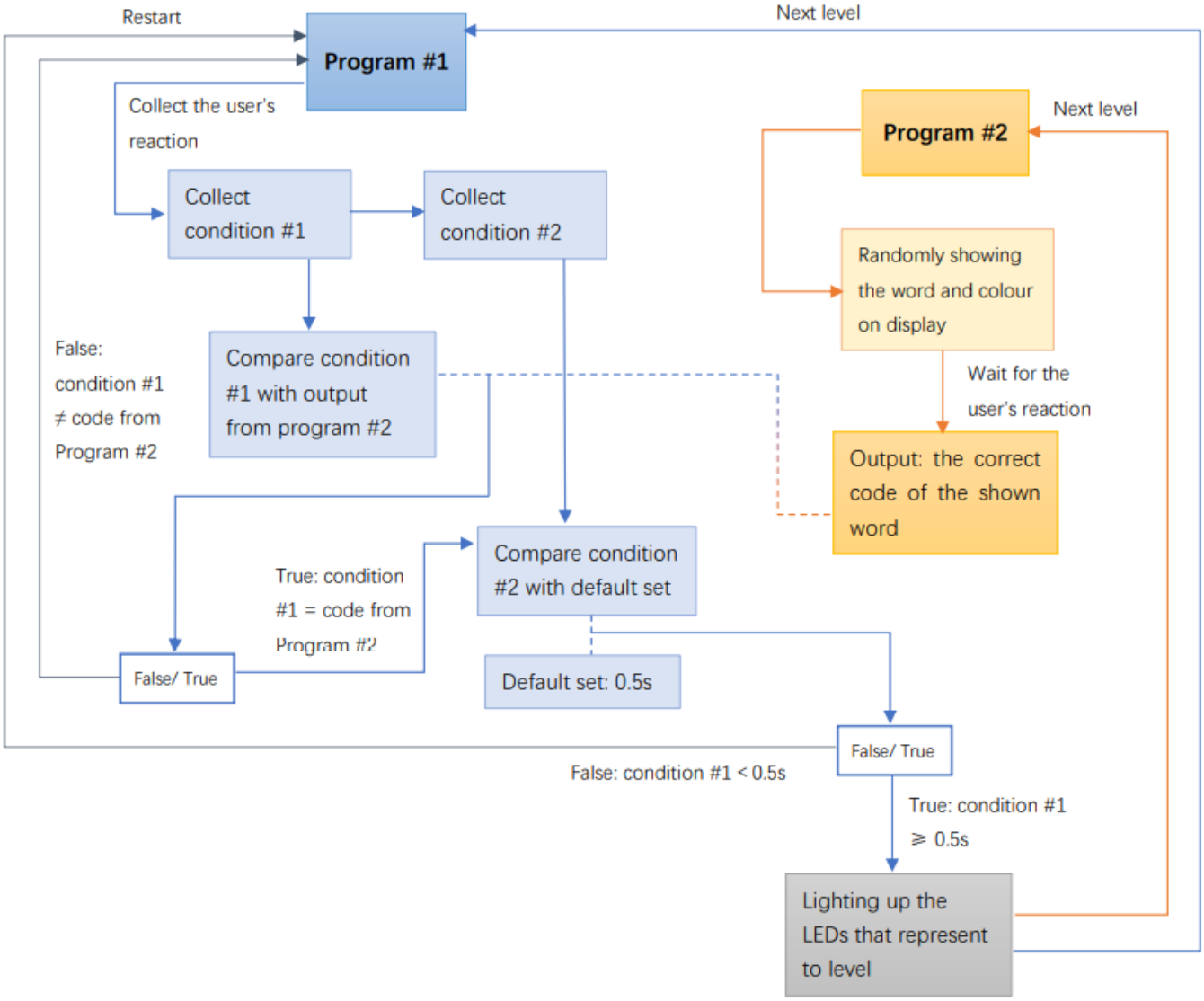

Diagram 2.2 - The software working process

2.2 Budget

The Required Components section shows the components and their respective prices. Relatively expensive components like the STM32 MCU itself and the TFT LCD display have only a required quantity of one and were bought singularly. Relatively inexpensive items such as the multi-color LEDs and push buttons were bought in bulk.

Table 2.2 - The budget list

Component Name	Quantity	Price ($)
STM32 Nucleo 64 Development Board	1	41.59
Multi-color LEDs	6	17.98
Current limiting resistors	10	16.89
TFT LCD Display	1	44.67
Jumper Wires	20	12.99
Breadboard	1	14.99
Buttons	6	12.09

Total Cost ($)	161.20

2.3 Project Timeline

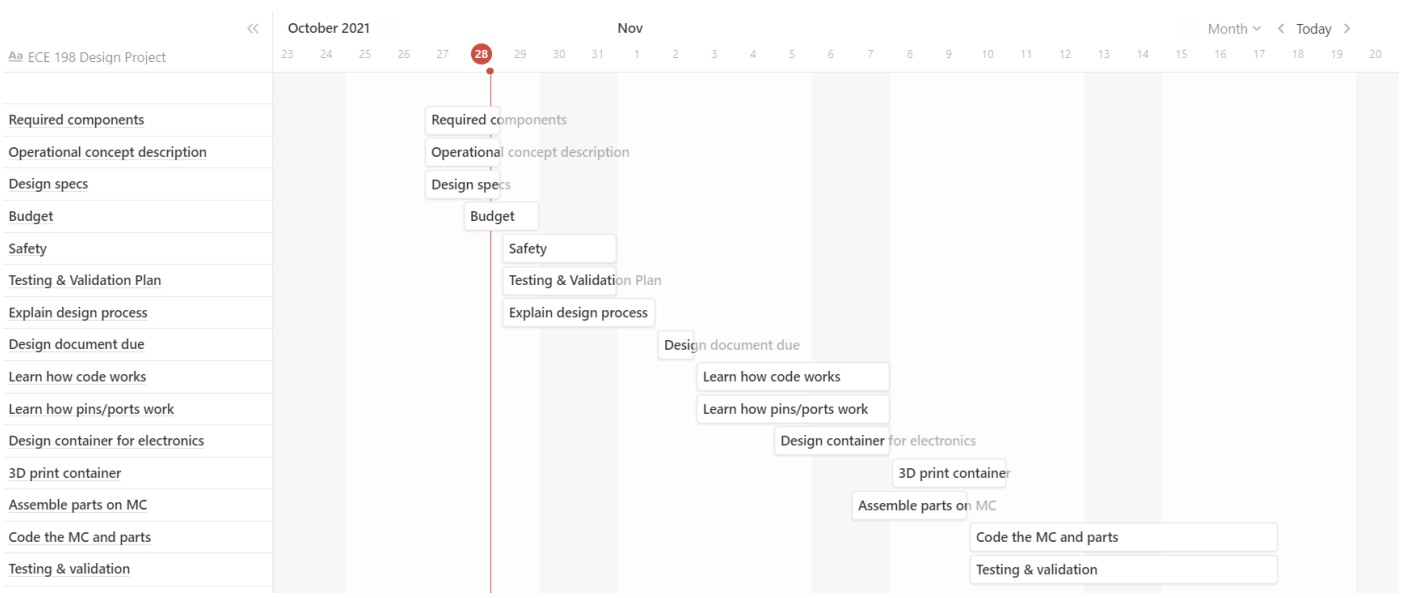

Diagram 2.3 - The Gantt Chart of project tasks

2.4 Safety

Working with electronics can get dangerous (due to the presence of electricity) if proper precautions are not followed. Even though we do not have a separate power source (like a battery pack) in this project, there are still other ways we can come in contact with electricity. Due to the presence of several hazards which can lead to severe risks, we have developed a table (*Table 2.3*) showing the hazard, risk, and a possible solution. *Table 2.4* shows the actions to be taken in case of an incident.

Table 2.3 - Hazards, risks, and potential solutions when working with electronics

Hazard	Risk	Solution
Bare cardboard, paper, or flammable material/liquid near the electronics	Possibility of fire if a flammable substance comes in contact with electricity from the wires	All combustible materials and liquids should be kept away from the working area. There should also always be a fire extinguisher near the working area
Bare wire (with no rubberized insulation)	Possibility of a fire or mild electric shock, and severe electric if bare wire penetrates the skin	All wires and components will go through a process of quality check where we look for cut-up wires, exposed wire, and any exposed metal which has the ability to conduct electricity. Wear rubberized gloves to avoid penetration of the skin
Poor connections and/or loose components	Loose components can cause poor connections which can cause a short-circuit	Check all component connections by physical testing (moving it around) and solder wires which are too loose
Water and fluids	Short circuit if electronics come in contact with the liquid	Keep all liquids and fluids away from the working area

Table 2.4 - Actions to respond to an incident

Incident	Action
Fire	Use a fire extinguisher, fire blanket if available, or sand/dirt to cut out the oxygen supply
Smoke and burning smell	Disconnect MCU from the computer (cutting out the power) and examine the circuit for what went wrong
Electric shock	Disconnect MCU from the computer and remove contact of the person from the circuit using a non-conducting object (eg. a wooden stick)

2.5 Testing and validation plan

We determined the default time limit for the reaction to be *0.5s* and 6 levels. To avoid directly failing, we decided to add 3 lives for the user, which means the game can restart if the user exceeded the time or pressed the wrong button. Those are the testing criteria.

Except for our own tests, we are planning to ask several volunteers (no less than 6) to help us test the product. The volunteers will be given three tries to try their best to finish the levels. We will record the result using *Table 2.5* below, and divide the successful press of the right button by the total times they press the button. Then, we need to decide if we need to change the setup based on the average rate of the correct response. We hope the average rate will be 50% in order to achieve the purpose of training. After we make changes to the prototype, the real device test will start over again to achieve the ideal rate of correctness as shown in *Diagram 2.4*.

Table 2.5 - Sample record table for the device test with volunteers

Volunteer number	Trial 1 (s)	Trial 2 (s)	Trial 3 (s)
#1			
#2			
…			

Average correct response (%)	

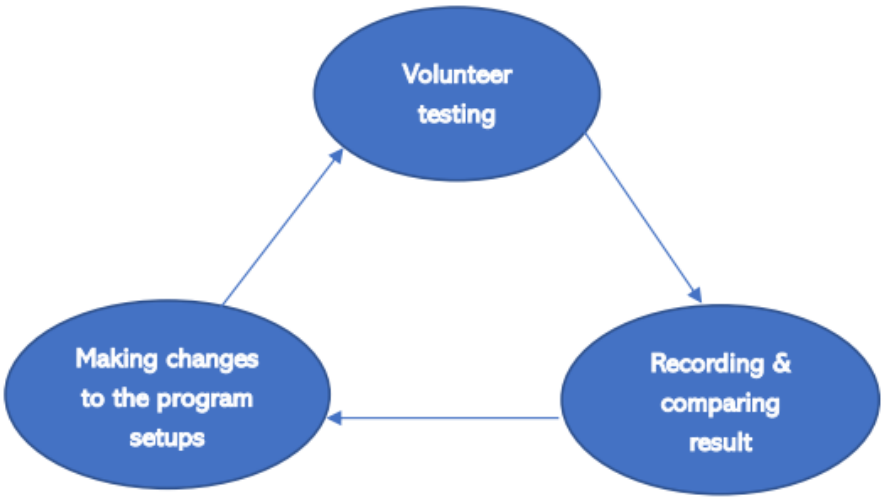

Diagram 2.4 - Testing cycle

2.6 Design process

2.6.1 Target audiences and technical choices

Although our product does not have any predetermined target audiences, it is, however, developed to be used in a playground-themed escape room. There are age rules on some escape rooms and estimately, the playground-themed escape room should have an age rating of 8+. Our product's target audience is anyone above and including the age of 8.

When we were selecting the TFT LCDs, we had considered 3 different types of it, which are "*DSD TECH 1.8 Inch TFT LCD Display Module with SPI Interface for Arduino and MCU*", "*1.8 inch SPI TFT ST7735 LCD Display Module 128x160 High Color Display Screen Board for Arduino 51/AVR/STM32/ARM*", and "*ILI9163C LCD 1.8 Display Arduino Platform Evaluation Expansion Board*". After a discussion with our instructors, we decided to go with the third option. The shipment of the first two was after November 25th 2021, which is after the deadline for our prototype.

2.6.2 Requirements and constraints

Table 2.6 - List of requirements and constraints

Requirements	Constraints
Components listed in *Table 1.1*	Budget
Coding programs in Visual Studio code	Time
Creating model TinkerCAD	Technical issues
Right to use 3-D printer	Physical limitations

References

1. Abasolo, Louvil. “Working with Electronics Safely.” *Circuit Basics*, 13 June 2020, https://www.circuitbasics.com/working-with-electronics-safely/.
2. Sharpe, Arnold. “SAFETY PRECAUTIONS TO TAKE WHEN WORKING WITH ELECTRONIC EQUIPMENT.” *Blog4safety.Com*, 6 Feb. 2017, https://bit.ly/3CApUBJ.