Piano Simulator

Alyona Radkevich

Project Description

The project is a small 24-key physical version of a piano. Combined with a piece of code, that will be created specifically for this piano, it can teach users basic musical notation and sound correlation. LEDs and a scoring system will tell users whether the note they hit was right or wrong. The application will provide visual feedback as well. I've created a simple 7-key version in MTEC 2280 class, and now I want to extend it to 23 keys and improve the functionality as well. It will develop my technical skills in wiring and coding. It will give me a better understanding of how Arduino pieces work together and on their own.

The outcome of the project is to have a working mini piano that will help to learn basic notation and develop a simple understanding of piano keys. Each key will have its own sound. The software will provide the right feedback to users and give them tasks to complete.

The code will be written in JavaScript and Arduino. The piano will use an Arduino motherboard. Each piano key acts as a push button and when pressed, it connects two wires together and sends a signal detecting which key is pressed, allowing the sound to be played without delay. Piano keys are made out of plastic, and the outside shape of the piano is made out of wood.

The project is aimed to teach users how to play basic piano keys, where they are located, and how to read notes. There are two parts: a piano keyboard and a JavaScript application that is as well linked to Arduino. Once the application is launched, a user sees the main menu with a few buttons. A user can then choose what he wants to learn and click on the right option.

A virtual keyboard on the screen helps to locate the right key on a physical keyboard. JavaScript generates a random key to play and displays it on the screen. Once any key is pressed, Arduino sends a value for that key to JavaScript that, in turn, compares the received value of the key with the expected one. If they match, JavaScript tells Arduino to light up the LEDs that are under the kyes. The code also corresponds to the pressed key by coloring that key on the screen. When the right key is pressed, the application randomizes a new value for the key to play, and the process repeats.

Deliverables

1. Physical piano
2. Completed JavaScript based application
3. Video showing working project
4. Poster
5. OpenLab Portfolio

Schedule/Calendar & Methods

Piano Simulator
Alyona Radkevich

Task	Start Date	End Date	Duration
Pick topic	10/26/22	11/1/22	7
Decide the scope	11/2/22	11/5/22	4
Set up deadlines	11/6/22	11/7/22	2
List materials	11/8/22	11/10/22	3
Order materials	11/11/22	11/20/22	10
Build a prototype	11/21/22	12/7/22	17
Application prototype	12/8/22	12/20/22	13
Test and update	12/21/22	12/25/22	5
Sketch keys and piano	1/10/23	1/17/23	8
Prepare files for laser cutting	1/18/23	1/25/23	8
Prepare for wood cutting	1/18/23	1/25/23	8
Laser cutting	2/20/23	2/28/23	9
Wood cutting	3/1/23	3/20/23	20
Make keys	3/1/23	3/10/23	10
Application coding	3/7/23	5/2/23	57
Wire up and test keys	3/11/23	3/31/23	21
Make piano (outer shape)	3/11/23	3/31/23	21
Install keys	4/1/23	4/20/23	20
Test piano and adjust	4/21/23	4/30/23	10
Test (by someone else)	5/2/23	5/3/23	2
Finishing adjustments	5/4/23	5/7/23	4
Record a video	5/7/23	5/7/23	1
Prepare presentation	5/7/23	5/8/23	2
Present	5/9/23	5/9/23	1

Required resources

1. Workshop
2. Laser cut
3. Soldering
4. Code editor
5. Arduino

Piano Simulator

Alyona Radkevich

Budget

Item	Estimate amount	Total spent
Materials	$200.00	$0.00
ELEGOO mega		$19.99
Springs		$29.97
Cooper Tape		$6.88
10K Resistors		$1.54
LEDs		$23.79
Paint		$6.48
Hot Glue Gun		$24.95
Jumper Wires		$14.99
Wire Stripper		$8.59
8.875% TAX		$12.17
TOTAL	**$200.00**	**$149.35**

Proposed table of contents

1. Introduction
2. Methods
3. Project Budget – estimate vs. actual
4. Project Calendar – estimate vs. actual
5. Completed Project
    a. Pictures of work in progress
    b. Inserts of code
    c. Video of the project
6. Conclusion