The Traveling Musician

A Senior Project Report

presented to

the Faculty of California Polytechnic State University

San Luis Obispo

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Computer Engineering

By

Brian Pease

December 2021

Table of Contents

Abstract

"The Traveling Musician" is the name of the device that travels up and down any piano and plays music. After setting it up, the traveler will start playing the song programmed into it after a button press until the song has been completed. Pressing the button again will have it repeat.

Introduction

Musical instruments such as pianos, have periods of time of unuse. Whether this is because of work, family, or time, pianos have downtime. This product serves to utilize the piano when it's not in use, to act more as background noise. There are pianos that can self-play, though this product would be a cheaper alternative aimed for pianos that do not have this function.

Stakeholders

Customers that would be potentially interested in this would be people who have mostly unused pianos and would like to have background music without having to spend a lot for their piano to be converted to a self-playing piano. The product could be used in smaller restaurants or within family households. These customers would benefit from a cheaper alternative to automated play and that is not difficult to set up.

Project Goals and Objectives

The goal would be to play a song:

- Objective 1: Move to the correct key for the note
- Objective 2: Activate the correct solenoid for the note (white or black key)
- Objective 3: Play a simple song that requires little movement
- Objective 4: Play a normal song

Project Deliverables

The main deliverable would be a device that goes up and down any piano in time to a song and plays the proper notes.

Project Outcomes

- More pianos being used in a casual environment
- Piano furniture (never played piano) becomes a feature for a house.
- Could inspire more piano players
- Create live piano music for movie usage without a pianist knowing how to play.

Background

There has been work in the past that makes pianos “self-playable”, such as Edelweiss’s pianos. This was accomplished by having solenoids with varying pressure under the striking lever, mimicking a strike to the key and thus to the piano string. Having these solenoids allows for an electronic means of activating them such as through an ipad. Prior to the Edelweiss piano, there was also the pianola, a vacuum based self-playing piano popular in the late 19 century. The pianola was a much more complex device since it required the use of specifically made paper rolls for the music and required the mechanisms to be under a slight vacuum. A less extensive and simpler way is using solenoids.

Formal Project Definition

Customer Requirements

- Be reasonably priced
- Easily placed and removed
- Plays a song
- Generally easy to use

Engineering Requirements

Product Requirements Table					
Spec Number	Parameter Description	Requirements	Tolerance	Risk	Compliance
1	Seconds Off Beat	1 sec	Max	H	T
2	Note Travel Error Accuracy	0.5 mm	Max	H	T
3	Power	6 W	Max	L	A, T
4	Key Range	7 Keys	Min	L	I

5	Runner Height Size	50 mm	Max	L	I
6	Cost	$60	Max	L	I
7	Weight	200 g	Target	L	T
8	Set up time	10 sec	Max	L	I

Customer Personas

Nathan, a piano enthusiast, wants to explore capabilities not previously thought of in live piano music. Effectively playing with three hands, the product will act as support to the piano player Nathan. His students would also use it to ease them into playing with both hands. The robot acting as the other hand.

Diego, a small business owner, has a piano for his customers but he finds that not many partake in playing. He has considered hiring a part time musician but due to the covid situation he does not want to eagerly spend. The device would allow him to have his piano be more in use, and create a more vibrant environment for his customers.

Samantha, A piano owner, has frequent hangouts with friends, family members, and coworkers. She loves the presence of a piano but never actually learned to play it. With the device she can add more flavor to her house without breaking the bank.

Use Case

The user would place the device on the piano. Situate the arduino and breadboard in front of the base, ideally on the piano (off of the keys). Plug in the device to a wall outlet and situate the device on top of a C key with at least the next octave under the wooden base. The user would then simply press the button on the breadboard after they pre-programmed their songs prior in order to hear their song.

Design

The design process was broken up into three different parts. The first part pertained to the design of the base or structure that held the device; The second was the design of the actual “runner” or the piece that holds the solenoids and parts; The third was the software design.

Base

The base was only difficult to design mainly because it had to work for any piano of any size. I had access to three different pianos during the summer when I was working on this part. A regularly sized grand piano, electric piano, and a medium sized digital piano (33 keys). The size and structure differed from each piano drastically. Originally, the first design had some type of clamping mechanism and was fixed above the piano keys.

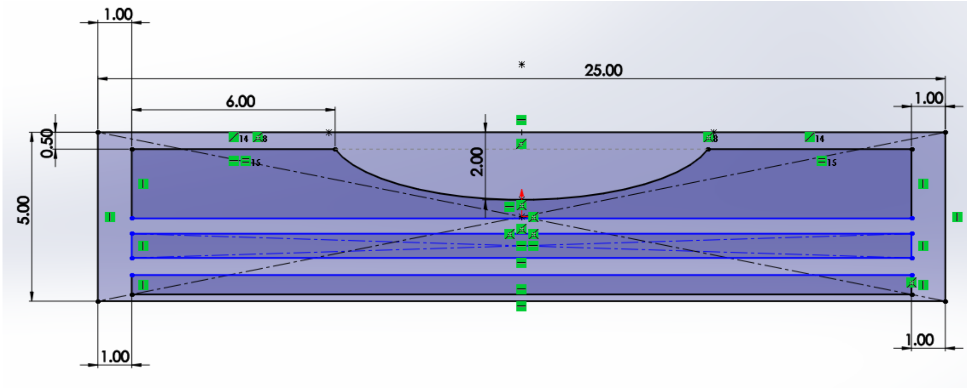

Figure 1: Original Base Design

Figure 1 was the design of this without the clamping idea. Because I was too focused on having a physical device rolling up and down the piano I neglected the other, and likely better, ideas such as having a type of belt system. I primarily didn't understand how a belt system would work but looking at a 3d printer would have helped in that endeavor. Eventually the best idea turned out to be just the rail system, such a simple block of wood, resting on top of the piano keys. As long as the device wasn't heavy and the board was long enough, the weight would distribute to the other keys. The final board is seen on Figure 2.

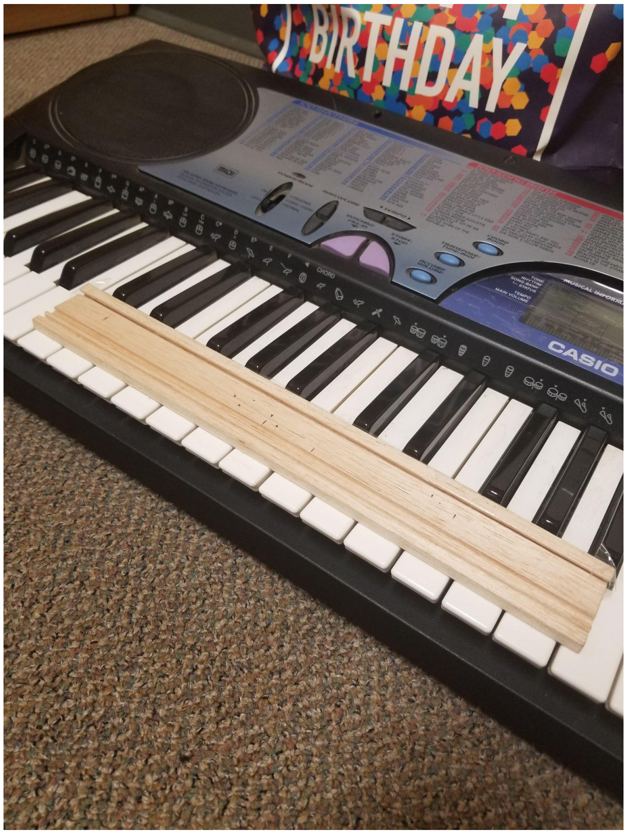

Figure 2: Final Board

Runner

The “runner” or the device that moves and holds the solenoids, had multiple iterations. When designing the runner I arbitrarily added the constraint that the max width of the runner would not exceed 22 mm, to allow for another possible runner to play the key right next to it. This turned out to be a bit of a difficult condition to meet considering that the servo on it’s side was 22 mm. The main idea of the first two iterations was to create a makeshift electromagnet that would attract a metal that would be attached to some type of lever to press the key in Figure 3, or act as a regular solenoid in Figure 4.

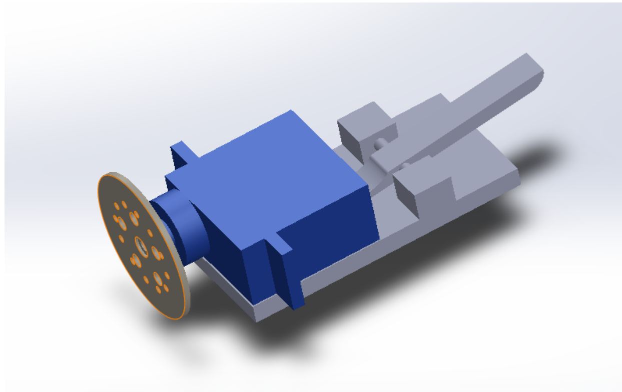

Figure 3: Runner Iteration 1

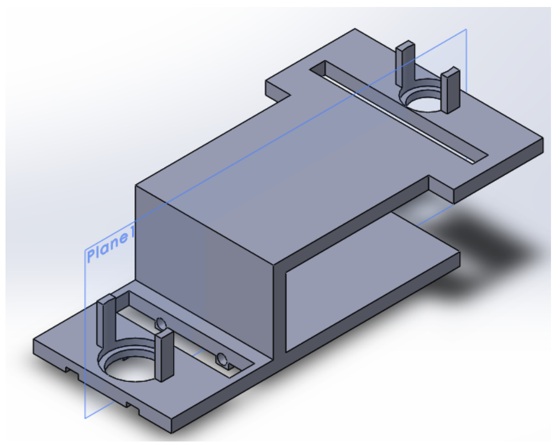

Figure 4: Runner Iteration 2

The main faults of these two designs was that it did not properly account for the size of the solenoid and it's wires to create enough pushing force.

From here, future iterations used actual pre-build solenoids as I did not want my lack of tools at home to come into consideration.

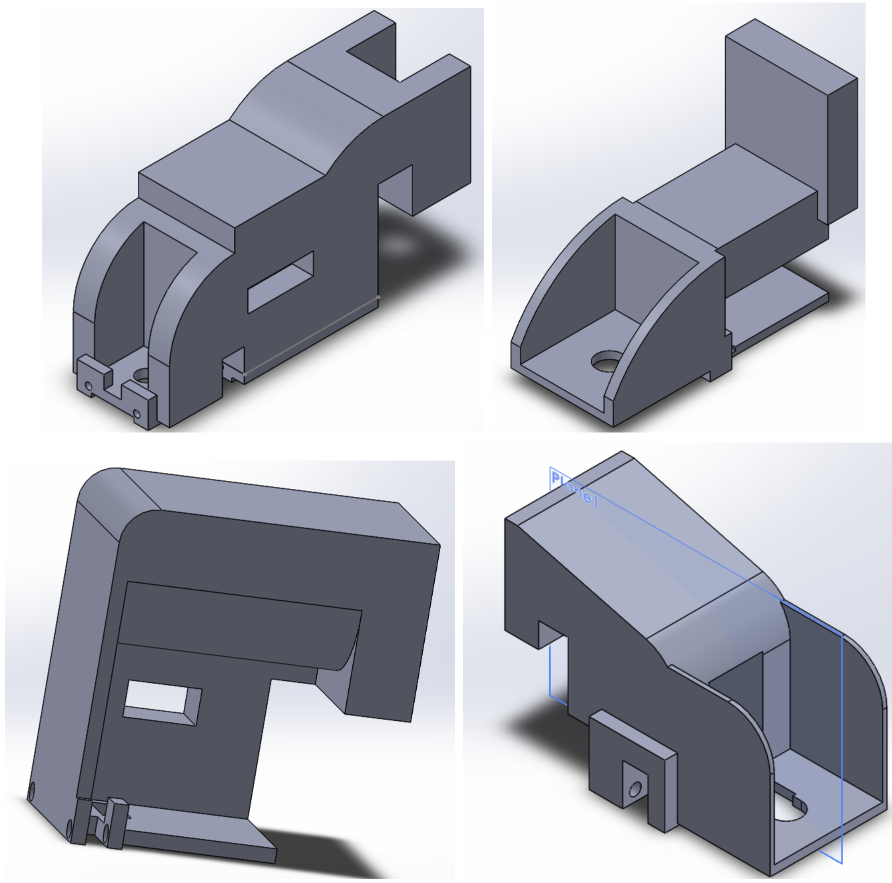

Figure 5: Runner Iterations 3 (top left), 4 (top right), 5 (bot left), and 6 (bot right)

From Figure 5, iterations 3 (top left), 4 (top right), 5 (bot left), and 6 (bot right) had adjustments and experimented with new ideas between iterations such as iteration 4 being too long and housing an insufficient solenoid. This iterative process was the result of trial and error up until the last iteration, iteration 6 seen in figure 5.

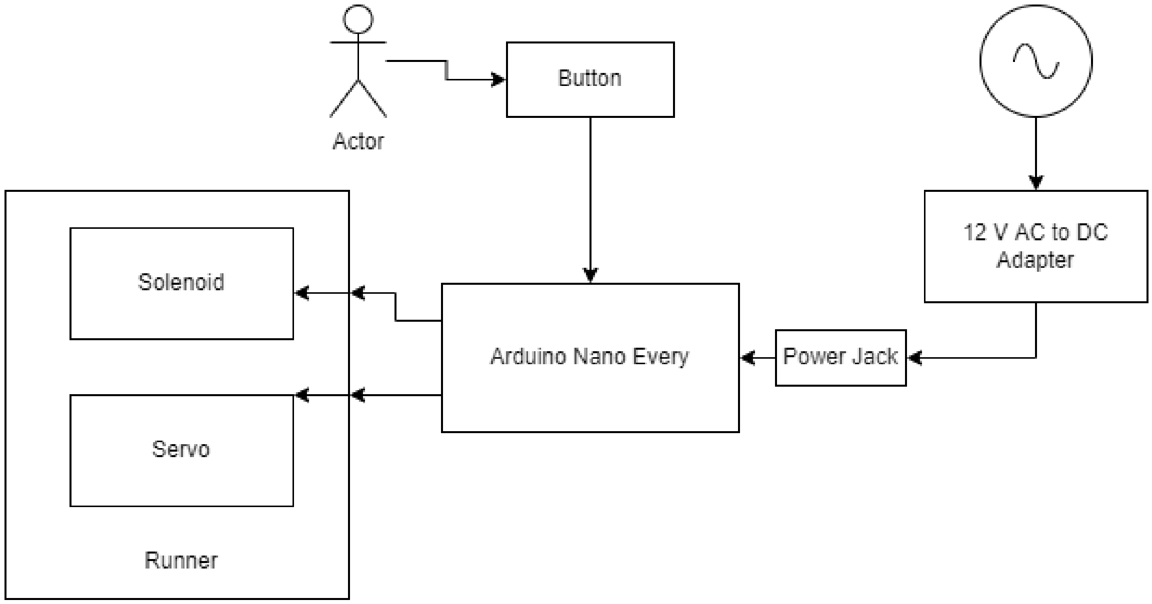

Figure 6: Hardware Functional Block Design

The hardware functional block design is depicted in Figure 6. Power is supplied through an AC to DC adapter reducing the voltage to a 12 V DC and this in turn is connected to a power jack that powers everything including the Arduino Nano Every. The user uses an external button to start the algorithm and the Arduino signals the servo and solenoid when to turn or deploy. This continues until the song finishes.

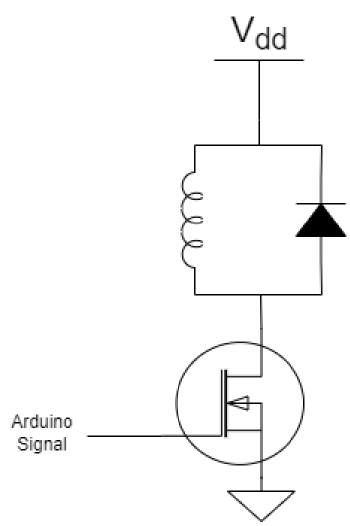

Figure 7: Solenoid Driver

On top of the runner lies a small breadboard in charge of holding all of the electronics for the runner such as the n-mosfet, diodes, and power lines. The circuitry (Figure 7) is rather simple so a small breadboard with 8 lines will suffice. The diode in the illustration acts as a flyback diode and provides a path back to the source voltage when switching occurs to avoid frying the mosfet.

Software

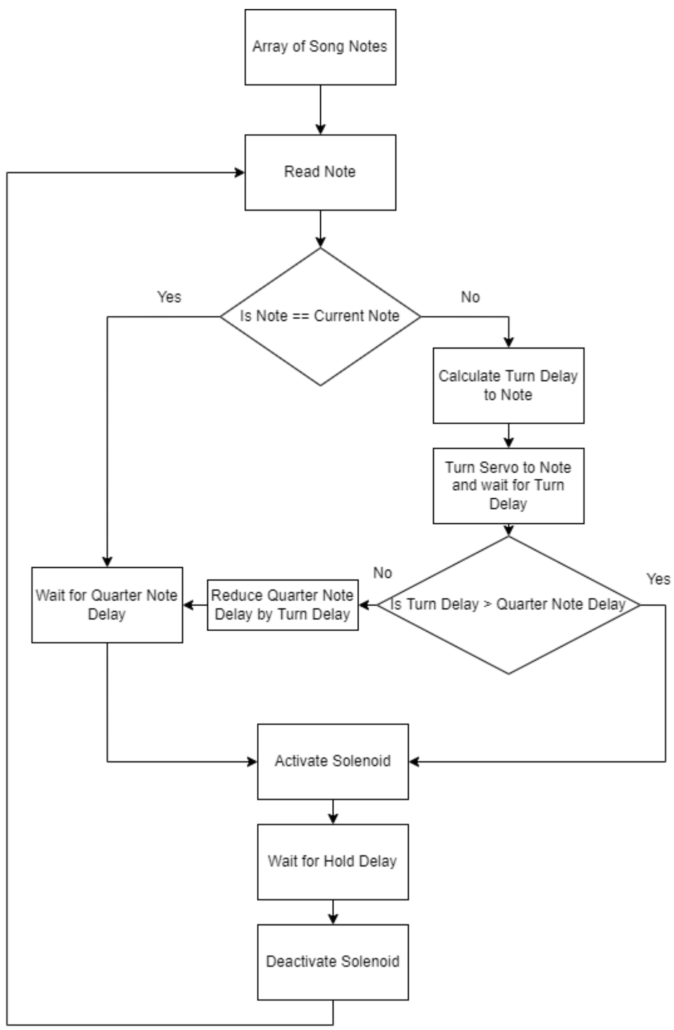

Figure 8: Software Flowchart

The general path of the software is seen in Figure 8 and is executed after the button is pressed. The “Quarter Note Delay” and “Hold Delay” are both constants pre-programmed before use. This software path was originally for the manual option. Since it is hard to read and produce notes with these delays present, it was designed to be constant so the user only needs to concern themselves with the notes and quarter notes. The alternative would be an array formatted to something similar to a song track in .midi files. The design is also scalable. In the event of an alternative way of movement, replacing the servo, the code will still attempt to adhere to the timing of the song. The solenoid only activates after the Quarter Note Delay or Turn Delay depending on which one is larger.

Notes in the song array are formatted as indexes. ‘0’ corresponds to the starting C key and other positive numbers correspond to the number of keys away from the starting C key. This is so a person could intuitively write the key they wish to play for a quarter note by simply counting the number of keys from the start.

The original design was to have the device repeat the songs it was given until told otherwise but given that the device’s variability error in where it ends up. This requires frequent intervention thus a continuous play would not be possible for the given setup..

System Testing and Analysis

Generally, testing was done in stages. The first stage was to adjust the delay times and servo speeds to a distance consistently with little error. This was a stage that was never really completed since very slight alterations were eventually found to alter the results. The wires for instance, if not set right would affect the distance traveled for one side resulting in two different velocities, left and right velocity. This was likely because I neglected weight distribution. The center of gravity would be roughly around the helper wheels and not the servo, resulting in little friction and a stationary device. A piece of wood was attached to the other end of the servo with slots for more weight to draw the center of gravity closer to the servo but this ended up to be meaningless as testing found. Trial and error eventually led to an error that would still be on the target key for a small song by tilting the device roughly 2 mm toward the servo.

The second stage was solenoid testing. After ordering the wrong solenoids, the two solenoids that I was going to use for the black keys and white keys separately met all of the pre-designed conditions. Their functionality was tested by having the solenoids deploy in tune to a midi formatted song. This was done by essentially reducing the notes to mean on and off.

The third stage was to combine all of the code and hardware to test for the final product. As mentioned in the first stage, the biggest issues were variable error in movement and weight distribution. Unfortunately the solution to the weight distribution and a tolerable movement, being tilting the device approximately 2 mm required the black key solenoid to push the extra weight from the tilting effectively making it unable to press the key. However since there is still an issue with movement, even if the solenoid was able to press the key it would not have been able to. The black piano keys are essentially smaller targets and would require more precise movement.

Testing the device essentially came down to trial and error. Especially the 3d printed runner as it required 6 iterations before it had some form of success.

The final demo was only achievable after extending the range of the white key solenoid, securing the base to the piano, lowering the black keys with tape slightly to account for the new height of the solenoid (from the tilting), and by raising the ground that the helper wheels were on by taping the railing.

Conclusion and Future Work

Overall, the project could have been better. Although it is able to play a simple melody like “Mary Had a Little Lamb”, it accomplishes it minimally. The software couldn’t be expanded because of the issues present within the hardware, such as variable error in movement and the weight imbalance. If the design of having a rc like car were to be kept and improved on, a separate arduino for bluetooth communications to remove wires, changing the servo to a motor, and wheel / weight reconsideration would be beneficial to remove some of the issues found during building the device. Ideally some type of sensor would also be beneficial to help recenter and ensure keys are properly pressed.

However it would probably be best to completely redesign the device. One that doesn’t require 1-3 mm altercations and be more consistent. The best idea is to have a belt system, one for the white keys and another for the black keys. Although it would not have the benefit of having multiple “runners” like the rc car could possibly have, it would be faster, more accurate, and would not require long wires or another arduino.

Reflection

My first response to my mistakes was to start the trial and error process earlier so I could find the issues that arose with my design such as the movement and weight issue. However, I don’t think that might have helped. Most of my development in regard to the hardware was during the fall where I had access to the university's tools. I did not have most tools back at home during the summer and development could only be done digitally. Instead, the best thing I could have done more extensively was to ask other people's opinions on my designs since they were the designs I was most unsure of. I also think this project would have benefited the most being in a team project since hardware and building were the areas I was least experienced in. It would have allowed me to focus more on the software side while getting more experience in the other two.

An aspect I realized about my thinking is that I wanted everything to be perfect in terms of measurement. This is fine to strive for, but I believe this is one of the reasons the performance of the final product was affected by some error in fabrication down to 1-3 mm. Another aspect I realized about myself is that I tended to apply constraints to myself, such as the width limit of 22 mm. These constraints only slowed me from getting to the prototype process, where larger issues would become more apparent such as the movement. Identifying these potential issues early allows for more designs to be considered.

In regards to other points, I am glad I got more experience in hardware. I had created solidwork CAD files and grew familiar with the machinery in Mustang 60 which would help in the future if I ever would need to use them again.

Overall, I learned a lot from the project. Mainly to get help and bounce ideas off people to find apparent issues like the weight distribution along with prototyping small things earlier like movement. That and to apply not so critical conditions after a prototype is made to try to improve it. This way if there were any issues with the design, it would become known before more time was spent improving a faulty design.

Bibliography

"Edelweiss Pianos." Edelweiss Pianos – Self Playing Pianos | Bespoke Player Pianos, 13 Aug. 2021, https://edelweisspianos.com/self-play/.

"Pianola." Wikipedia, Wikimedia Foundation, 19 May 2014, https://simple.wikipedia.org/wiki/Pianola.

Appendix

Analysis of Senior Project Design

Please provide the following information regarding your Senior Project and submit to your advisor along with your final report. Attach additional sheets for your responses to the questions below.

Project Title: The Travelling Musician Quarter / Year Submitted: Fall / 2021

Student: (Print Name): Brian Pease (Sign): BP

Advisor: (Print Name): John Oliver (Initial): BP Date: 12/10/2021

Summary of Functional Requirements

Play a small song on the piano that was manually programmed into the arduino prior.

Primary Constraints

One constraint was not having any tools available for prototyping in my first quarter of the project in Summer 2021. A common issue in the project was something to do with hardware. A 3d print that seemingly works on paper proved to not work once actually printed, which is mostly due to my lack of experience with the medium. I also ran into trouble obtaining the right solenoid. Finding the right sized solenoid for what I needed also proved to be difficult since none of the well known vendors seemed to not have small solenoids that had a decent throw. I had to resort to non-engineering websites like Amazon that had less information about a product and didn't have any datasheets.

Economic

Original Estimated Cost

Part	Part Cost	Manufacture ID
Arduino UNO	$20.12	A000066
Motor	$3.28	M1N10FB11G
Button	$1.00	54-700-R
Wood	Vary	-
Phone Charger (5V Source) Or 5V Battery	Vary	-

This was the original estimated cost table before the start of the project when most of its operations were unknown. The solenoid wasn't decided on this stage so it was omitted. None of which are the parts actually used, aside from the button.

Final Component Cost

Part	Part Cost	Actual Cost (Shipping & Tax)	Vendor	Manufacture ID
Arduino Nano Every	$ 13.90	$ 19.37	Arduino	ABX00033
FS90R Continuous Servo	$ 4.95	$ 10.08	Pololu	2850
DC Barrel jack	$ 0.95	$ 6.42	Adafruit	PJ-102AH
n-ch Mosfet TO220-3 x2	$ 3.28	$ 8.92	Digi-Key	FQP30N06L
12V Adapter Power Supply	$ 8.59	9.47	Amazon	AEP2EA-A5
120 pcs Assorted 50cm Dupont Wire	$ 13.99			B07GD3KDG9
10mm Micro Solenoid	$ 11.29			ADD06190484
8 mm Micro Solenoid	$ 12.30	$ 40.87	Amazon	B00R5CICX8
1N4001 Diode x2	$ 1.00	$ 1.00	Cal Poly	-
			Total	$ 96.13

Final Bill For All Components

Date	Item	Vendor	Cost	Documents
11/06	SD Reader	Amazon	$ 7.50	Invoice
11/06	Micro SD	Amazon	$ 7.50	Invoice
10/20	Continuous Servo x1	Pololu	$ 10.08	Paypal / Invoice
10/20	DC Barrel jack	AdaFruit	$ 6.42	Paypal / Invoice
10/20	nmos x2	Digi-Key	$ 8.92	Paypal / Invoice
10/20	12V Adapter Power Supply	Amazon	$ 9.47	Invoice
10/13	nmos x2	Digi-Key	$ 8.92	Invoice
10/13	Wires/10mm & 8mm Solenoi	Amazon	$ 40.87	Invoice
09/19	Solenoid 5V small	SparkFun Electroni	$ 25.01	Paypal / Invoice
09/08	Enamled Copper Wire	Amazon	$ 7.70	Invoice
08/20	Arduino Nano Every	Arduino	$ 19.37	Paypal / Invoice
08/19	Continuous Servo x2	Pololu	$ 15.17	Paypal / Invoice
			Total	$ 166.93

Original Estimated Development Time

Week	Task
1	Have a CAD implementation of the base that will hold all the parts and attach to the piano.
2	
3	Have a final tested base CAD design.
4	Design and prototype the “runner” and it’s pressing mechanic.
5	
6	Have a final tested runner CAD design.
7	Setup / fit Arduino to base and connect runner to Arduino.
8	Code “keyboard setup” module and test effectiveness of moving across and mapping
9	
10	Catch up for any unforeseen issues / roadblocks
11	
12	Break unless behind.
13	Code the conductor and test with one note. Debug any issues with hardware and software.
14	Test with two notes near and far from each other to test coordination of the system and debug if needed.
15	
16	Full system test with different pianos and debug
17	
18	Implement another way to add songs or other features only if on schedule.
19	Product Finish
20	Senior Project Paper

Actual Development Time

Week	Task
1	
2	
3	Brainstorm designs for Base / Order Solenoid Parts
4	
5	Learn SolidWorks and create a base CAD
6	
7	Design and 3D print the Runner
8	Wind Solenoid and Check ability to press things
9	
10	Redesign Runner design for pre-made solenoids / 3D Print Runner
11	Solenoid and Servo drivers done
12	Try to finalize final design for runner / Get more parts for runner / 3D Print Runner
13	
14	Design code for Nano / 3D Print Runner
15	Write code for nano auto function (.midi) / 3D Print Runner
16	Testing Servo interrupts and timing debug
17	Mapping distance traveled by servo
18	Soldering wires to breadboard and coding for manual function
19	More sanding wheels / Attaching wooden block for extra weight / More coding for manual function(s)
20	Senior Project Paper / Fix Runner

If manufactured on a commercial basis

Given the state of the project, the product would not sell in an actual commercial environment.

Manufacturability

Issues with the 3d printing were fitting issues such as wrong tolerances but that was due to inexperience in the medium. Other issues were trying to get exact measurements. The base for example with the final CAD model needed to be 6mm high and have a railing with a depth of 3 mm and a width of around 2 mm (had to settle with 3 mm). I also did not know how to create something with a hatch to secure a part like the servo so I needed to work to create a design that allowed the devices to slip or be screwed on.

Sustainability

The device was struggling to stay together since the solenoid and diode wires barely fit into the small breadboard. It was soldered on but the solder started to come off after moving it around in boxes and resting it sideways. Additionally the solder that was used was solder with lead because otherwise it would melt the plastic around the pin. Aside from the lead solder, a tight fixture that houses the electronics would benefit the lifetime of the product. Upgrading the design shouldn't be difficult since most of the components are removable aside from the wooden piece attached to one of the solenoids. The project tries to use the least amount of components so it's not an overuse of resources.

Ethical

The device can't really be misused unethically.

Health and Safety

There are a couple of issues in regards to safety. Most of the components are in open view and could be re-wired to be dangerous especially since there is a 12v adapter that could go up to 2A. The second issue is less drastic. Since the device has no sensors, it cannot detect issues and can possibly collide with other objects. The wires could also push objects (if they are light enough) too. Because of this there is some amount of risk with the project.

Social and Political

There can be issues with copyright with this device. If the device was acting as it was meant to, then it is likely (if it got popular) restaurant owners would have to pay for the rights to use it. Other than that there are no other social and political considerations.

Development

I learned about the machinery in Mustang 60 and obtained my red card for the project. However, I primarily used the table saw, drill press, and belt sander for the project. I learned and understood .midi files since I wanted to implement them originally. There is also Solidworks and Cura for 3d printing. I designed all of my 3d prints in solidworks and got used to working with the software when I did not know how prior. I didn’t learn too much in Cura since I mainly used it to verify my design and see what support structure would form and what would manually have to be removed after printing. Most of what I learned was in regards to hardware build and design.