Music and Tone Sequencer

Annie Zhang
Cornell University
Ithaca, NY, USA
zz229@cornell.edu

Boris Chan
Cornell University
Ithaca, NY, USA
bkc38@cornell.edu

Cindy Wang
Cornell University
Ithaca, NY, USA
cw653@cornell.edu

Ryan Kim
Cornell University
Ithaca, NY, USA
rk494@cornell.edu

INTRODUCTION

The goal of this project is to design a music sequencer device that can be used to generate musical patterns and rhythms of varying tones and sounds. The device will be able to generate a musical sequence whose notes and rhythm are determined by a pattern of buttons activated by the user; users can choose which notes to activate by interacting with a button pad present on the top of the device. The purpose of this project to twofold: firstly it is meant to help individuals experience music creation without needing a background in music, and secondly it can cultivate interaction and social bonding between individuals by having them work together to generate music without much effort. For example, the device can be used to teach young children about the fundamentals of chords and rhythm by having them see how buttons activate different sounds, which would encourage them to see how these sounds play together and/or how they create rhythms as the system loops through their activated buttons. The device ultimately can help users gain a wider understanding of how music operates, thereby engendering interest in the study of music and music theory.

The system operates mainly with the use of buttons that are organized in a X-Y, 16x8 grid arrangement. The X-axis, represented by the 16 columns of buttons in the button grid, determines when a button should be played when the system reads through all the buttons; the Y-axis, represented by the 8 rows of buttons in the button grid, is tied to an anhemitonic pentatonic scale wherein there exist five notes within an octave: G, A, B, D, and E. In other words, a button's row will determine which of these notes to play, and the Y-axis is organized such that the lowest rows are the lower notes of the pentatonic scale while the higher rows are the higher notes in the pentatonic scale. Since a anhemitonic pentatonic scale only has five notes within an octave, the 8 rows of the button grid are set up such that the lower five rows represent one full octave while the three remaining higher rows represent half an octave. By pressing any of the buttons of their choice, users can effectively create patterns of buttons that correlated to rhythms of notes within a pentatonic scale.

Our ultimate design goals consisted of developing an intuitive device that would allow individuals unfamiliar with music theory to still create music. We aimed to develop a neat, fully functional prototype that can correctly process button input, emit sounds, and accept volume control.

RELATED WORK

The inspiration of our project stemmed from an online pentatonic step sequencer [8]. While this online tool is great, we wanted to see if making it physical was a possibility, which led to us finding a video of a physical music sequencer table [2]. This has since inspired our design for indicator lights. Other physical productions of a musical sequence inspired our use of buttons and our display in general [3][4][5][11]. As we began to consider the musicality of our device, we concluded from a few sources that using a pentatonic scale would be best for easily creating harmonious sounds for beginners in music theory [8][10]. Finally, to increase the interactivity of our product, we found some sources that suggest displaying inviting messages and encouraging collaboration [3][9]. We consider these to be optional features for future iterations. To elaborate more on our related work, we have divided our research into three different sections, which will each be further explained below.

Psychological

Before designing our device, we looked into two scientific studies that could be useful to our designs. We first looked into synesthesia, a neurological condition that unifies an individual's senses [1]. This means that in some cases, affected individuals can literally see music as various colors in their mind. We are exploring this condition and relating it to how we want our users to interpret music and color. We also found a video game named "Journey" to be very interesting [9]. The premise of "Journey" is that people are on journeys and interact with other people, but they cannot communicate through speech or text, only a musical chime. The game explores how people build connections through this interaction. The soundtrack also changes dynamically depending on the players' actions. This is similar to the effect we are trying to create in the sense that we want to explore interactions using music, and our display will also change dynamically depending on the users' actions.

Music Sequencing

The most important aspect of our design involves music sequencing. To ensure that we could design the most intuitive, harmonious, and interactive device possible, we looked at a few examples. Our project was originally inspired by an online pentatonic step sequencer [8]. However, we wanted to make this physical. Thus, a music sequencer table we found on YouTube has had the greater influence on our design [2]. This project makes good use of indicator lights, which we have adapted into our design. Harmony-wise, we settled on using a pentatonic scale. Many pieces of music use this scale, and it is commonly used to introduce beginners to music [8]. Since a large part of our audience consists of those who are unfamiliar with music, it makes sense that the pitches used in the project should be simple and sound good together. Finally, we looked into DJ launchpads for layout inspiration [11]. DJ launchpads are structured in a very neat, grid-like pattern. We would like to use similar visual lights and buttons to create our interactive display. Additionally, we are intrigued by the sounds of DJ launchpads and would like to incorporate that into our designs as well.

Interactive Art

In order to incorporate interactive components into our design, we specifically conducted research on interactive art. The first resource we stumbled upon was Google's Anypixel.sj library [3]. This library is open source and free. Thus, it could be useful in implementing the visual aspect of our display. We also found this interactive sound art exhibition, named LINES, especially interesting [4]. This exhibition implemented user input with sensors instead of buttons, which we took into consideration for our design. After seeing this exhibition, we wanted to look more into interactive wall displays, and we found a YouTube video that showcases quite a few [5]. Having a vertical display would allow for increased interactivity and more collaboration; our display could also be more of an artistic piece. To further explore the role our senses play in interactive art, we researched The Museum of Feelings [6]. This was a pop-up interactive museum designed specifically to stimulate the visitors using all five senses. Space and resource limitations will likely prevent our display from being as immersive as the Museum of Feelings, but engaging our audience's senses (specifically sight, touch, and sound) is an important part of the user experience. In terms of how we wanted our users to interact with our product, we have discussed a tabletop implementation and a vertical implementation. This project uses the idea of music visualization on a tabletop in a slightly different way than we intend to, but exploring different ways to connect sound and light may help develop a better, more unique user experience [7].

KEY FEATURES AND GOALS

We have six main features in our final prototype. We will go into more detail about each below, but these features are also further explained in our video.

Button Input

We first implemented button input with a small 4 x 4 number keypad we found in our kits. This keypad was not ideal, as it was very unclear as to which button corresponded to which LED. We evaluated its performance by rating its appeal; it was not very intuitive or professional. We modified this by exchanging the keypad with an Adafruit Trellis, a backlight keypad driver system (Fig. 1). These boards come completely bare, meaning we needed to solder on the button LEDs ourselves (more on that in the next section). For button input, all we needed was this board and a corresponding silicone button pad to to take in signals; these signals were then able to be processed via our Teensy microcontroller, which we wired to our button board. Integration-wise, we had some issues because our initial prototype involved using a RedBear instead of a Teensy; this led to some coding issues when we switched over to our Teensy, but we were able to resolve the issues. This feature performed beautifully in our final prototype; all button inputs worked perfectly, we were able to process multiple buttons at a time, and we never encountered any errors or malfunctions when we tested or presented our final product.

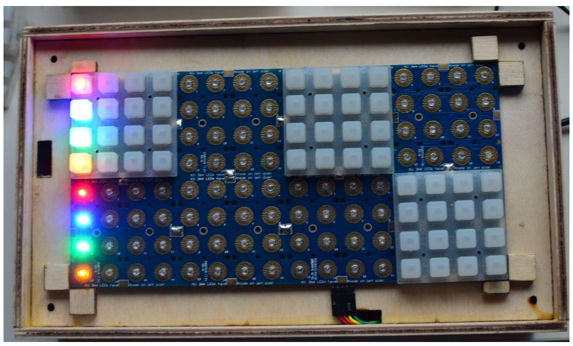

Fig. 1: Button input boards and silicone buttons.

Button LEDs

Originally we wanted our button LEDs to be all white colored LEDs, as it was closer to the pentatonic step sequencer that inspired us. But after the decision that we would like an extra row of LEDs as a time indicator, we thought that all of them being white color would be kind of dull and in order to make the device more attractive, we decided to use different colors LEDs. Although the indicator LEDs feature was scrapped in a later phase, the colorful LEDs decision was kept. We had decided that since each row represents a certain note, it would be more indicative if each row was made of the same color, instead of each column. Also, for the convenience of color-blind people, we made sure the red and green color was not next to each other to avoid causing confusion. There was also a original design where we would like our button LEDs to be turned on dimly if they were pressed and set to the ready state and only glow brightest when the sound it represents was being produced. This idea was not implemented due to the hardware did not have the capability to control the brightness of each LED. During the integration phase, the soldering of the LEDs took a lot of time and effort as we had 128 LEDs in total in our final design. Several LEDs had also burned off due to all the test runs that we had done and we had to replace those LEDs. However, we had run out of the yellow LEDs at a time so we had to use a white LED to replace in its place in our final product even though it was not our original intended design.

Fig. 2: Button LEDs.

Variability

In our initial prototype, we implemented a small 2 x 3 grid of buttons using the keypad as a proof of concept. We expanded this to an 8 x 8 grid using four Adafruit Trellis PCBs. This allowed us to have a full octave of a pentatonic scale, plus three more tones, giving users a variety of tone options. But after learning that we could connect up to eight of these boards, we decided to expand it to a 16 x 8 grid of buttons.. We chose to expand the x-axis over the y-axis to allow for a longer sequence and more diverse rhythms.

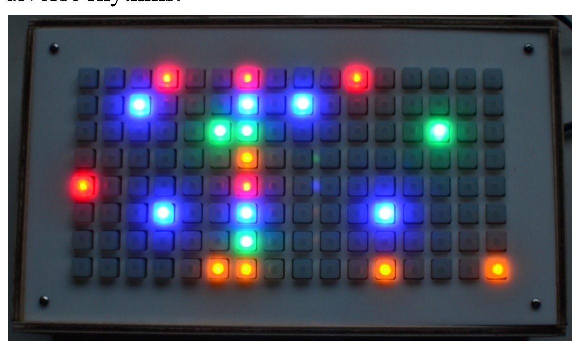

Fig. 3: 16 x 8 button grid.

Sound Output

Sound was first implemented in our first prototype, in the form of a passive buzzer that outputted tones of different frequency. The system was set up so that the buzzer would play a tone at the frequency corresponding to the bottommost activated button in a column; it did not play a combination of frequencies or different notes of any kind. This system was merely intended to replicate the feeling of changing sound output based on user input.

The first prototype's sound system was successful in that it properly responded to user inputs and detected when users activated or deactivated buttons on the button pad. If a user activated a button in a column, then the buzzer would play a tone at a frequency corresponding to that button; if multiple buttons in a column were activated, the buzzer would play a tone with a frequency corresponding to the bottommost activated button in the column. After fully realizing this in our first prototype, we believed that having different sounds corresponding to each row in the button pad was possible.

Our next design made use of the Teensy Audio Shield, which could output at most four channels of audio and could use a microSD card to store sounds and other data (Fig. 5). We altered the system so that instead of playing a single tone, the system would select which sounds to play from the microSD card, depending on which buttons in the column were activated, and output a combination of sound files. This was more in-line with our finalized goal as we wished for the system to be able to output chords of notes instead of single tones. However, after evaluating the system and using dummy sound files to check combinations of sounds, we realized that the audio shield was not powerful enough to output to four channels of audio; only three channels were possible without audio becoming warped. Furthermore, with eight rows of buttons, we had 8! possible combinations of sound we had to generate from 8 rows of buttons.

To circumvent this problem, we had to divide the eight rows of buttons into 3 groups; each group was attached to a particular channel on the audio shield. Using a binary system, each row in a group was assigned to a number either 1, 2, or 4. In each group, the combination of activated buttons would produce a binary sum that ranged between 1 to 7; by doing so, 7 combinations of sound were possible each each channel, circumventing the need to create 8! separate sound files. For example, in a group of three rows, if the first and third rows were activated, they would add to a sum of 1 + 4 = 5; this sum of 5 corresponds to a sound file where G and A are played simultaneously. By dividing all eight rows into three groups, we take full advantage of having multiple channels possible on the audio shield and reduce the complexity of combining sounds together at the same time.

This change was very successful: there were no more complications with warped sound that we had originally witnessed, and the code was able to process the information from the button pads to the sound output much faster than before. Furthermore, with the binary system in place within each group of rows, only 19 sound files had to be created for this assignment.

Fig. 4: External speaker.

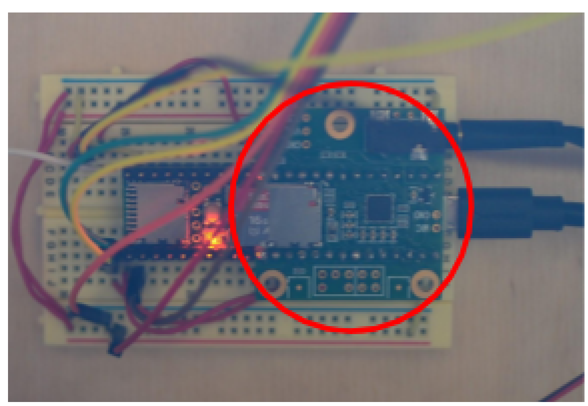

Fig. 5: Audio shield with microSD card.

Volume Control

Volume control was not introduced initially or in our first prototype. After we completed most of our final prototype, we realized that our audio shield can be plugged into a speaker or a set of headphones. Thus, we wanted to introduce a volume control knob in order to allow for users to adjust the volume if they are using either option, especially for the headphone option. Since we introduced this feature very late in our design cycle, we needed to solder on the wires needed for the potentiometer very last minute. Additionally, our box was already laser cut and half put together by the time this feature was introduced; as a result, we drilled the hole for the knob ourselves, so it's not very precise or clean (Fig. 6). However, this feature performed very well and proved to be a valuable addition to our product. At the Sciencenter, volume control came in handy many times, and it was also helpful to be able to turn the volume down so that we could talk over the music.

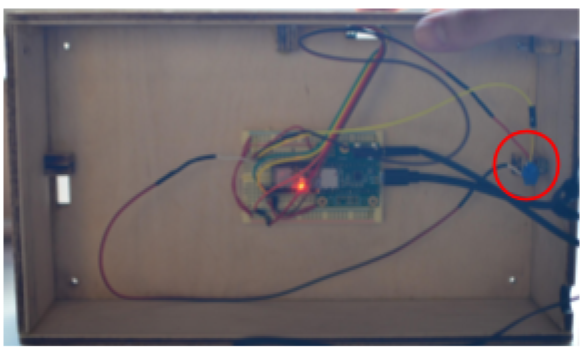

Fig. 6: Volume potentiometer.

Enclosed Hardware

The box was designed to keep the enclosed hardware easily accessible and the outside clean and user friendly. From the top down, there were several layers: a white acrylic sheet, the silicone buttons and PCBs, and the wooden top.

The top acrylic sheet was laser cut to keep the buttons in place. A 440 screw was placed at each corner in order to attach it to the wooden layer below it. We secured the screws with nuts on the underside. Screws allow the top to be easily removed or reattached; thus, we could easily remove the board.

As seen in Fig. 1, there were wooden blocks placed at the corners of the wooden layer below in order to keep the PCBs in place but still removable in case additional modifications to the boards were necessary. The wooden blocks also serve as a support to keep the acrylic sheet from crushing or applying too much pressure to the PCBs. A hole was placed on the side of the wooden layer to allow for wiring to reach underneath.

The breadboard was placed upside down on the underside of the wooden layer. The box had no bottom to keep the breadboard easily accessible. It also allowed flexibility in choosing power sources and audio outputs. The Teensy 3.5 was powered by a small portable battery during the demonstration, but by keeping the bottom of the box open, a computer could be used to power it during tests. The audio output was connected by an AUX cable, so external speakers or even headphones could be used to output sound. It was then necessary to include a potentiometer for adjusting volume so that headphone users could use the device comfortably and external speakers would still be loud enough to hear. The potentiometer protrudes from a hole in the side of the box.

Because of the open bottom, when the device is in use, the external speaker and portable battery can both be placed underneath the box. Alternatively, there was a small hole on the side of the box allowing for wires to exit if the equipment did not fit inside.

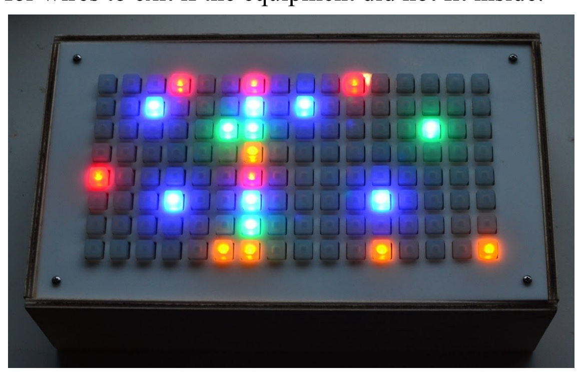

Fig. 7: All hardware can be fit under the box.

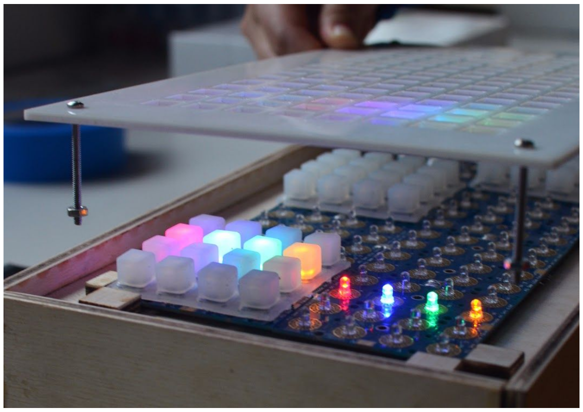

Fig. 8: The top is removable and can be screwed in.

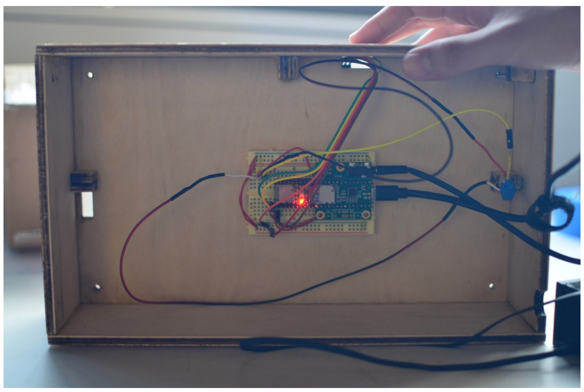

Fig. 9: The breadboard's location inside the box.

FINAL PRODUCT FEATURE INTEGRATION

Our final prototype of our music sequencer will consist of a 29cm * 16.5cm * 9cm frame that consists of a 8*16 silicone button pad that operates in tandem with an audio shield and volume potentiometer to output rhythms of notes; these notes are part of an anhemitonic pentatonic scale wherein there exist five notes within an octave.

A video of the device and the explanations for each feature are given in the link below:
https://youtu.be/X9uepFLdJ9k

8*16 Silicone Button Grid System

An 8*16 silicone button grid with LED's contained within each button will act as the primary input for the user. The grid is designed such that the y-axis (the rows) represents certain frequencies of sound corresponding to notes in the pentatonic scale, while the x-axis (the columns) represents time. Essentially, all the buttons on the same y-value position will have the same frequency; as (f) increases, the higher the frequency for that row of buttons. The frequencies increment in such a way that correspond to the pentatonic scale. Meanwhile, as the x-value increases over time, all the buttons with that x-value who have been activated will have their sounds played at the same time. Our buttons will have two states: inactive and active. A button's initial state is inactive, in which it is dark. After being pressed, it is active, and is lit up. When it's a column's turn to generate noise, all buttons in that column will light up, and the active buttons within that column will play their corresponding sounds. When an active button is pressed once more, it returns to the inactive state.

Sound Output Based on User Input

The system reads the button grid like a graph of frequency vs. time (Fig. 10). It "moves" from left to right; the leftmost columns emit sounds first, then the next, then so on and so on until the end is reached. Upon reaching the end, the system loops back to the beginning of the graph.

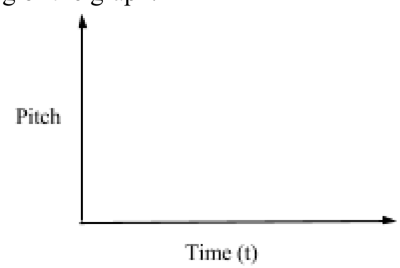

Fig. 10: *Graph displaying our button grid setup.*

As "t" increases, the system will read the next column of buttons; each incremental increase of "t" is equivalent to one beat. Upon reaching a column, all buttons in that column will have their LEDs shine at 100%, and the system will output, using our speaker, a combination of the sounds associated with each activated button. When (t) increases, the lights in the previous column will return back to their inactive (dark) or active (lit) status, and the next column in time will light up. As you can see in our diagram (Fig. 10), the pitch increases as y increases, and the time increases as x increases. Each unit of time represents one beat, and after 16 beats have passed, the system resets to t = 0.

Sound output volume is controlled via a potentiometer that allows the user to control the volume between 0% and 100%. Headphone users should maintain volume between 0% and 50%, while speaker volumes must maintain volume between 50% and 100%. By turning the potentiometer, users can control the volume of the sounds outputted by the Tone Sequencer.

Wiring

A diagram of the wiring of the device is provided in Fig. 11.

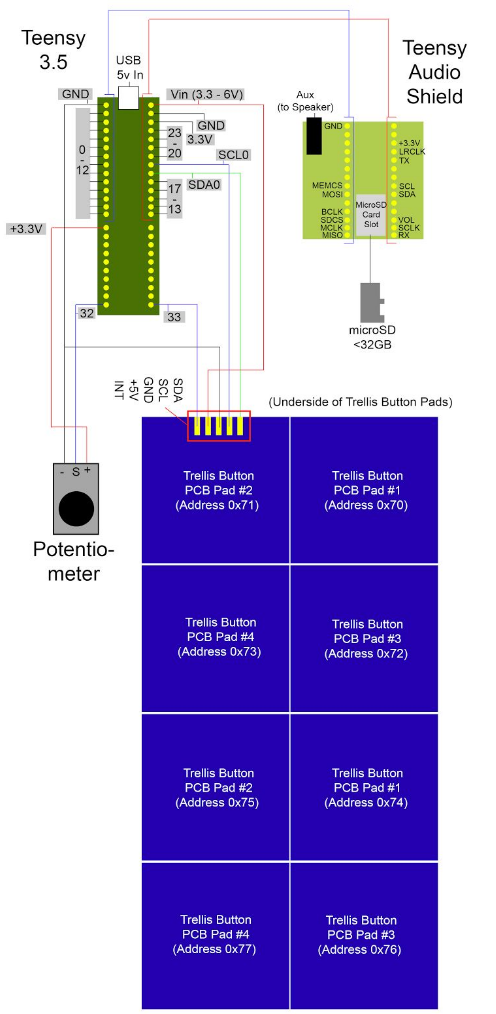

Fig. 11: *Wiring schematic for the Tone Sequencer*

FUTURE WORK

The only feature we had planned to implement but did not involved indicator lights. In the future, we would like to place 16 white LEDs in a single row above our main button grid, one above each column of buttons. A single white LED will light up when its corresponding column is active. This will increase clarity for users and allow for our product to be more intuitive.

REFERENCES

1. Barnes, Tom. 2015. This is What's Happening in the Brains of People Who "See" Music. Retrieved September 12, 2017 from https://mic.com/articles/120435/this-is-what-s-happening-in-the-brains-of-people-who-see-music#.ZHswyIGYb
2. Cater Clips. 2016. Music Sequencer Table Plays Michael Jackson. Video. (21 July 2016). Retrieved from https://www.youtube.com/watch?v=mIvc6uBDGzU
3. Google Developers. 2016. AnyPixel.js - a web friendly way for anyone to make unusual displays. Video. (2 June 2016). Retrieved September 9, 2017 from https://www.youtube.com/watch?v=58pxJ8z1Vow
4. LINES. LINES is an interactive sound art exhibition created by Swedish composer Anders Lind in 2016. Lines attached to the wall, on the floor and hanging from the ceiling in combination with sensors and electronics are forming three novel music instruments. Retrieved from https://www.youtube.com/watch?v=hP36xoPXDnM
5. Nambiar, Krishnan. 2016. Interactive wall collection. Video. (20 July 2016). Retrieved from https://www.youtube.com/watch?v=hqCeFdPoQLc
6. Ogilvy. 2016. Museum of Feelings. Video. (19 June 2016). Retrieved from https://www.youtube.com/watch?v=wzO_vEdemyA
7. Robert Scales. 2010. ReacTable - interactive table top music visualization. Video. (7 February 2010). Retrieved from https://www.youtube.com/watch?v=yqJTnX3ey2k

8. Tonematrix. u.d. Audiotool. Retrieved August 30, 2017 from http://tonematrix.audiotool.com/
9. Wikipedia. 2017. Journey (2012 video game). Retrieved September 9, 2017 from https://en.wikipedia.org/wiki/Journey_(2012_video_game)
10. Wikipedia. 2017. Pentatonic scale. Retrieved September 12, 2017 from https://en.wikipedia.org/wiki/Pentatonic_scale
11. zBlurryz - Launchpad PRO Covers. 2016. What is a Launchpad and How Does it Work? Retreived September 9, 2017 from https://www.youtube.com/watch?v=vFAzD5amZDo