SIMPLE
instruments

March 14th, 2019

Dr. Craig Scratchley
School of Engineering Science
Simon Fraser University
Burnaby, BC, Canada
V5A 1S6

Re: Design Specifications for Luminotes

Dear Dr. Scratchley,

The enclosed document provides design specifications for our Capstone project, Luminotes. The goal is to create a projection device that will be mounted over a piano,which will illuminate the piano keys using a laser for the users to follow along. Users will be able to select a song to play by choosing a MIDI file using an accompanying mobile application. Luminotes looks to provide a solution to users who wish to learn piano but struggle to read sheet music or prefer to play songs immediately. This device and application can also be a cost-effective introduction to learning piano and more powerful to use compared to equivalent learning methods, like YouTube videos.

This document looks to present the design specifications of Luminotes through multiple stages of design. It will delve into the high level design before individually examining the hardware, firmware, and structural designs.

Simple Instruments is a company composed of six undergraduate engineering science students: Shahira A. Azhar, Caitlin Finnigan, Hsiao Chen Eldon Lin, Dayton Pukanich, Adnan Syed, and (Mike) Wei Hao Xu. This diverse group of students looks to utilize their skills in computer, electronics, and systems engineering to create the Luminotes.

Thank you for your time in reviewing the enclosed design specifications documentation. Please feel free to contact us through my email (wxa17@sfu.ca) regarding any questions and concerns.

Sincerely,

(Mike) Wei Hao Xu
Chief Communications Officer
Simple Instruments

SIMPLE instruments

Design Specifications

ENSC 405W

Simon Fraser University
Faculty of Applied Science
School of Engineering Science

Team 14 Members:

Shahira A. Azhar
Caitlin Finnigan
Hsiao Chen Eldon Lin
Dayton Pukanich
Adnan Syed
(Mike) Wei Hao Xu

Abstract

Luminotes serves as a tool for casual hobbyists and beginners to quickly learn how to play new songs on the piano. It does away with the need for a large monetary and time commitment, that comes with having to learn how to read sheet music. Luminotes achieves this by simply projecting lasers onto the individual keys that the user will follow to play the chosen song. Songs are chosen using a mobile application, which connects to the projector device via bluetooth. The projector device is small, and lightweight enough to be easily mounted to the piano without distraction to the pianist.

This document provides in-depth detail about Luminotes through a system overview and a listing of all functional and structural design. It serves as a guide for the design and development of Luminotes.

Table of Contents

SIMPLE instruments

List of Figures

List of Tables

Table 10: Environmental Standards

Glossary

ADC	Analog to Digital Converter
App	Luminotes mobile application
BLE	Bluetooth Low Energy
Chassis	Structural framework of Luminotes device
Device	Luminotes device
DSP	Digital Signal Processing
Flash	An electronic non-volatile computer storage medium
Galvo	Galvanometer
ISO	International Organization for Standardization
LED	Light Emitting Diode
MCB	Microcontroller Board
Microcontroller	Small computer on a single integrated circuit
MIDI	Musical Instrument Digital Interface; a protocol for digital music recording and playback
NOOBS	New Out Of the Box Software
Octave	8 consecutive white keys on the piano keyboard
OS	Operating System
PoC	Proof of concept
RAM	Random Access Memory; a volatile computer storage medium
Rx	Receiver
SoC	System on Chip; integrated circuit for MCB
Tx	Transmitter

1 Introduction

Formally learning how to read sheet music in order to play specific songs on the piano can often be a barrier for casual musicians, as it is unintuitive and time consuming. The monetary and time commitment involved with doing so is much too steep for someone who simply wants to learn their favourite song to show off to friends. This road bump is often enough to discourage many hopeful pianists from learning more.

While alternatives exist in the form of electronic keyboards with fairly easy to follow tutorials using light up keys, they are just not the same as a traditional piano. The lack of string resonance in these keyboards produces a lacklustre tune when compared to a piano. Moreover, having to buy this sort of electronic keyboard is still quite the monetary commitment; especially for the casual hobbyist who already has free access to a piano through friends, family or school.

With all this in mind we offer Luminotes as a solution. Our device acts as a low-cost, and small form factor accessory that makes learning new music easy. One just mounts it to their piano, selects a song from the mobile application that is connected via bluetooth, and follows along to the keys lit up by the device projector. *Figure 1* outlines this functionality with a high-level behavioural diagram. Achieving this functionality demands a detailed specificification of design, which this document aims to provide.

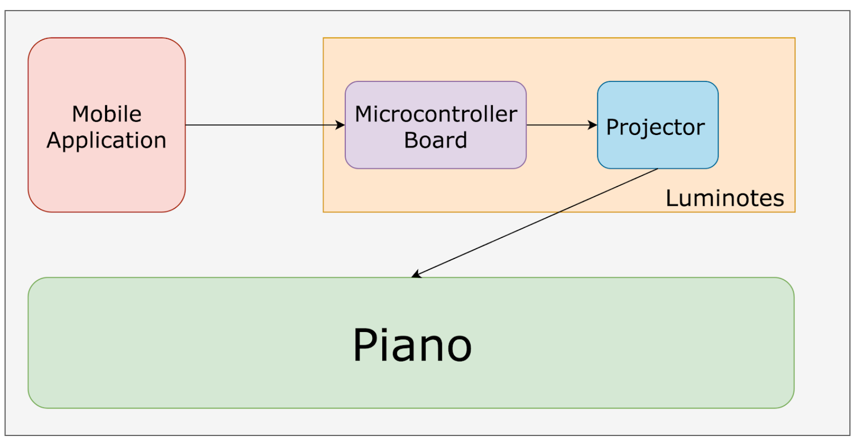

Figure 1: *High Level System Overview*

1.1 Scope

This document provides design specifications, as well as a system overview of the Luminotes application and device. The specifications are categorized as High Level, Hardware, Firmware, and Structural. These categories are further broken down into subcategories according to the specific functionalities and components involved.

1.2 Intended Audience

This document acts as the main point of reference for specifications of Luminotes functional and structural design. Use of this document is targeted towards Simple Instruments Inc. engineers, testers and project managers. This document will be used as a baseline for future revisions of Luminotes. Furthermore, this document is also intended for potential partners, Dr. William Craig Scratchley, Dr. Andrew Rawicz, and teaching assistants.

1.3 Background

Musical Instrument Digital Interface (MIDI) is a file format created in the 1980's, which is used to encode musical information. The file consists of parallel tracks of note sequences as well as timing information. Since MIDI is digital, its file size is significantly smaller than an audio file, which allows for quicker information transfer. Because of MIDI's simplicity, it is easy to integrate into many systems. MIDI files of copyright-free music are widely available; these can be found on websites such as the Petrucci Music Library.

2 High Level Design

Luminotes consists of two main components; a mobile application and a projection device. The application is the primary point of user interface with the device. Through it the user can access various settings such as adjusting the metronome, and selecting what music to play. The other point of user interface is on the actual device; a button to power it on and a set of dials to calibrate the projector. Bluetooth is used to connect the app to the projector device. Selected music is a MIDI file that is transmitted wirelessly to the microcontroller board. The board processes the file and forwards it to the Projector unit which then projects it onto the piano keys for the user to follow along to. In this Proof of Concept (PoC) design, we will be implementing the firmware, hardware, and structural components. The software will be a part of the prototype.

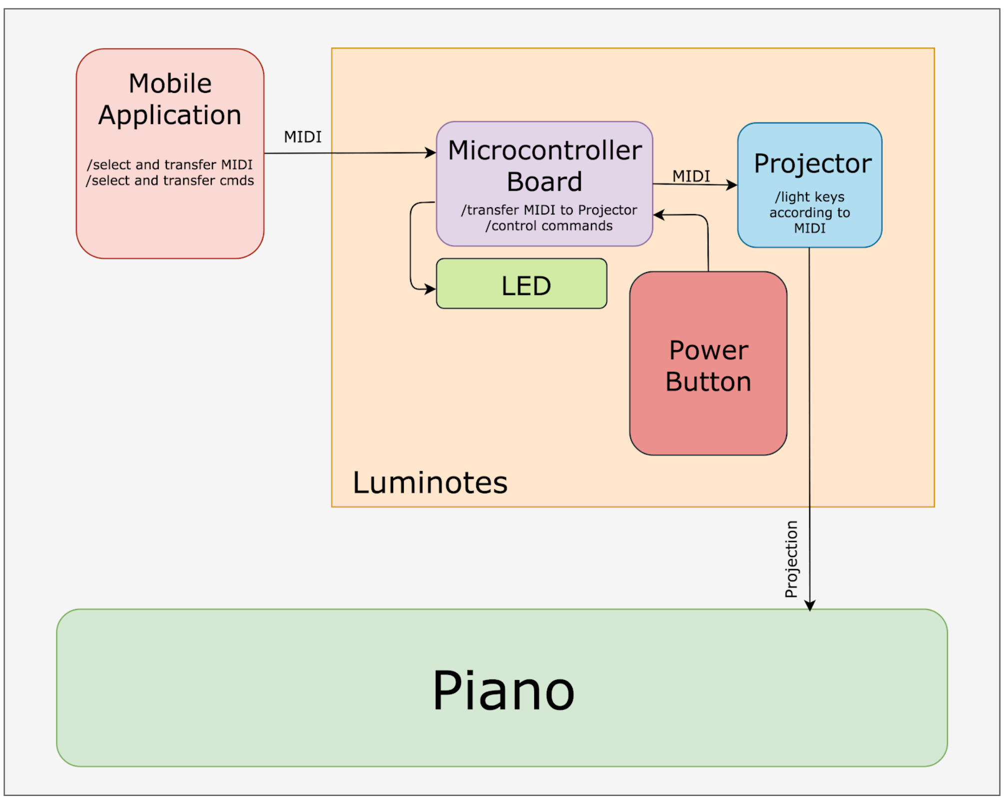

Figure 2: *Detailed System Overview*

3 Hardware Specifications

Luminotes hardware is split into two main components; the base and the head. The base contains the main bulk of the system, including scanner system, MCB and other peripherals. A high level layout of the base is shown in the following figure.

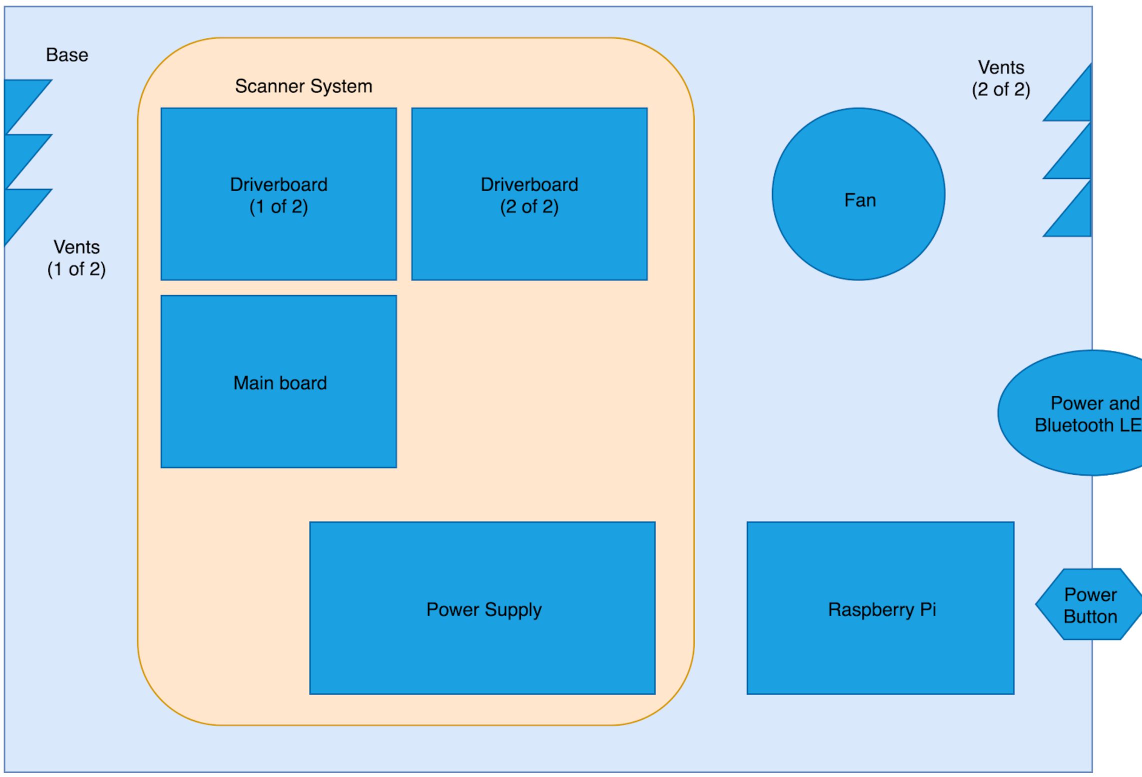

Figure 3: *High level layout components of the base*

The head contains the components needed for projecting onto the piano. It includes the galvo, laser and twin mirrors. A high level layout of these components can be seen below.

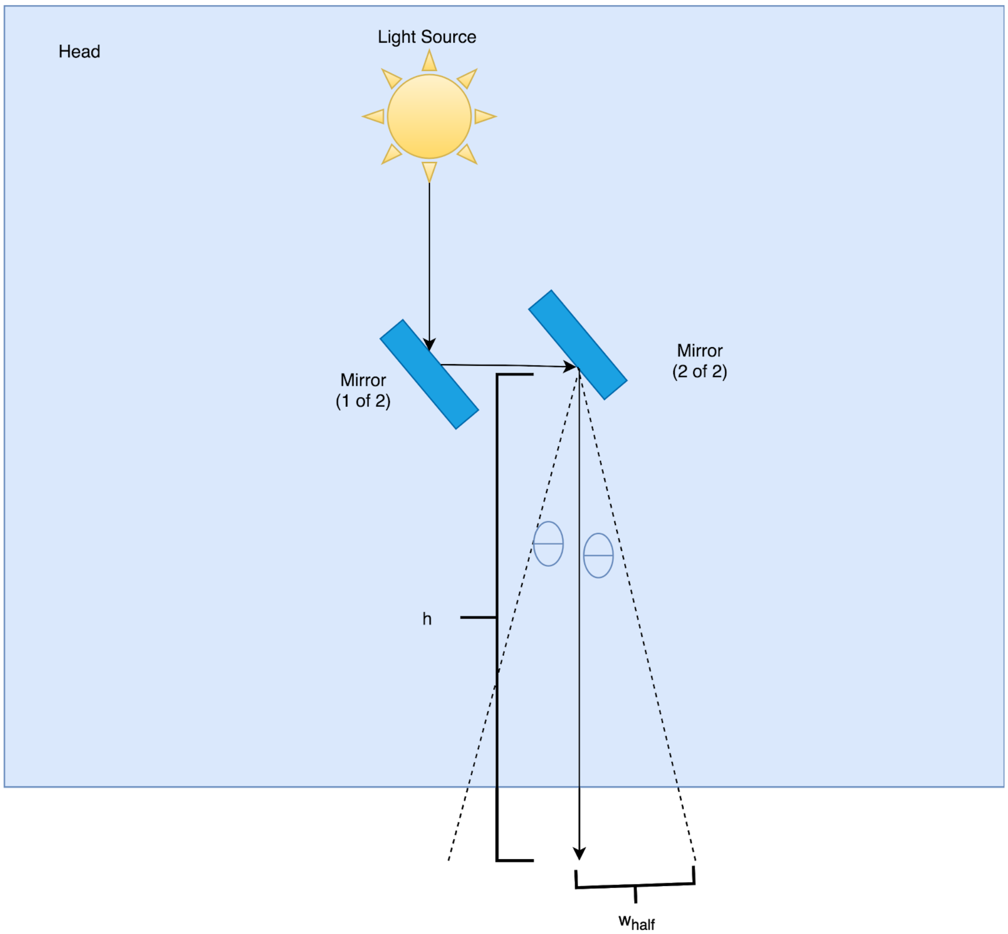

Figure 4: *High level layout components of the head*

3.1 Microcontroller Board

The Microcontroller board is a Raspberry Pi 3 B+; chosen for its processing capabilities and ease of development. The Raspberry Pi's quad-core SoC is plenty powerful enough to support Luminotes firmware. It comes equipped with a Bluetooth module; allowing for communication with the mobile app. Storage is in the form of a MicroSD card which offers flexibility in flash memory size. A 64GB Silicon Power MicroSD is chosen, as per [**Req.5.1.13-a**]. Further hardware specifications of the MCB, as well as the requirement specification they fulfill, can be found in *Table 1*.

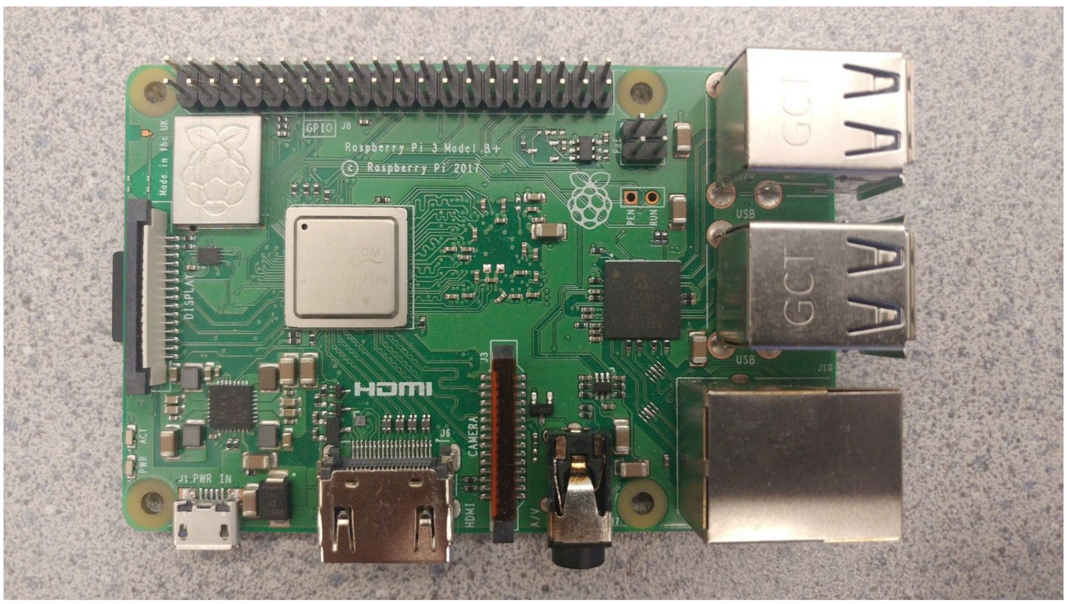

Figure 5: *Raspberry Pi 3 B+*

Table 1: *Raspberry Pi 3 B+ Hardware Specifications [2]*

Specification	Description	Requirement
SoC	Broadcom BCM2837B0 quad-core A53 (ARMv8) 64-bit @ 1.4GHz	**Req5.1.9-a**
Bluetooth	Bluetooth 4.2, BLE	**Req5.1.1-a**
Storage	Micro-SD	**Req5.1.13-a** **Req5.1.14-a**
RAM	1GB LPDDR2 SDRAM	**Req5.1.12-a**
GPIO	40-pin	**Req5.1.16-a**
Dimensions	82mm x 56mm x 19.5mm, 50g	**Req5.1.15-a**

3.2 Laser

The laser used will be Class 2. However, Class 1 is safer which is defined as non-hazardous. Class 1 must be used to avoid possible fire risk, eye damage caused by reflections from piano keys, or damage to eyes to due additional optical instruments used. [3] Therefore, we will reduce the voltage to reduce to the power to allow make the Class 2 laser a Class 1 laser. [7]

The laser will be externally powered by the MCB (Raspberry Pi) as per [**Req.5.2.4-a**]. As the Raspberry Pi can supply either 3.3V or 5V, the laser specifications must comply with one of these voltages. By powering the light source this way, it will reduce the need for batteries, which will be a more sustainable and reliable way to power the laser. The laser will also be controlled by the MCB to turn it on and off. It will be on only during times when the piano keys need to be illuminated, while during times when the laser is being positioned to light up different piano keys it will be kept off.

3.2.1 Laser length calculations

The piano keys has a certain width, therefore, the laser beam diameter must be limited to a certain size. By figuring out the rayleigh length, the placement distance from the laser to the mirror can be determined. As shown in the figure below. The near field (rayleigh length) desired is right before the beam expands.

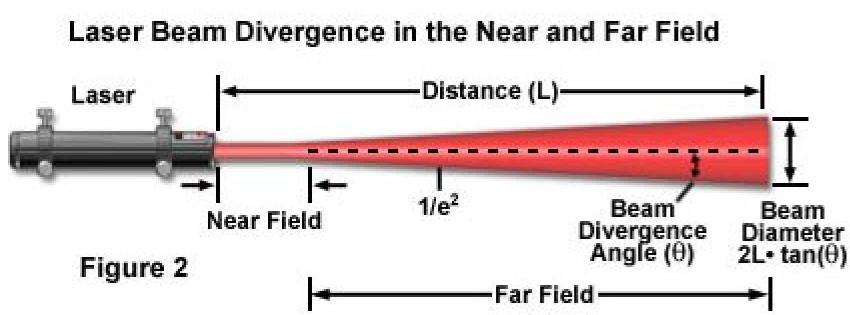

Figure 6: *Laser near field and far field [4]*

The rayleigh length can be calculated using the following equation

$$Z_R = \frac{\pi\omega_0^2}{\lambda}$$

Where λ is the wavelength and ω_0 is the beam waist. If we zoom in on to the laser shown in the figure below we see the the rayleigh length is the “distance along the propagation direction of the beam from the waist to the place where the area of the cross section is doubled” [5]

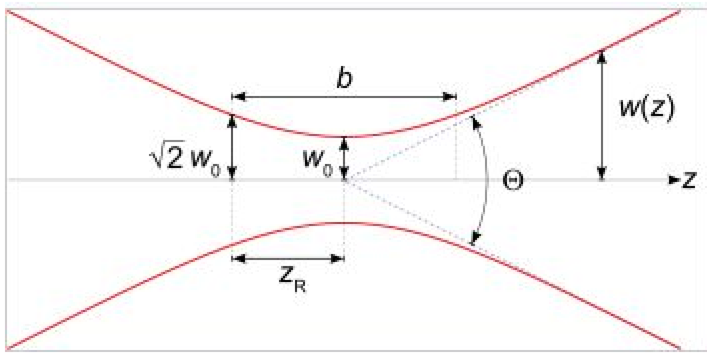

Figure 7: *Beam width calculations [5]*

The laser used is red which has a wavelength 650nm.

3.2.1 Mirror Calculation

To calculate how lasers reflect off the mirrors, we use the law of reflection which states that the light returns to the same medium that it came from when the lights hits the surface. Our surface will be the mirror. [6]The image below shows how the light will be reflected:

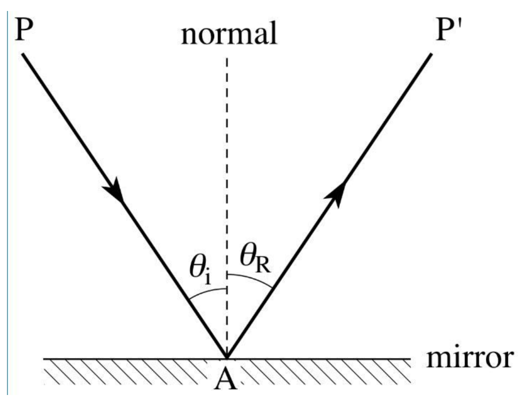

Figure 8: *Law of reflection. Light reflected from a plane mirror [6]*

As the image shows, at any point P, when the incoming ray hits the mirror at point of incidence A, the reflected ray will pass through point P'. The angle of incidence θ_i is equal to angle of reflection θ_R [6]

3.2.3 PoC Prototype

During the proof-of-concept prototype there will be initially only one laser to control the notes for a single hand.

3.2.4 Final Prototype

For the final prototype there will be two different coloured lasers, each one to control the notes for one hand.

3.3 Galvanometer Scanner

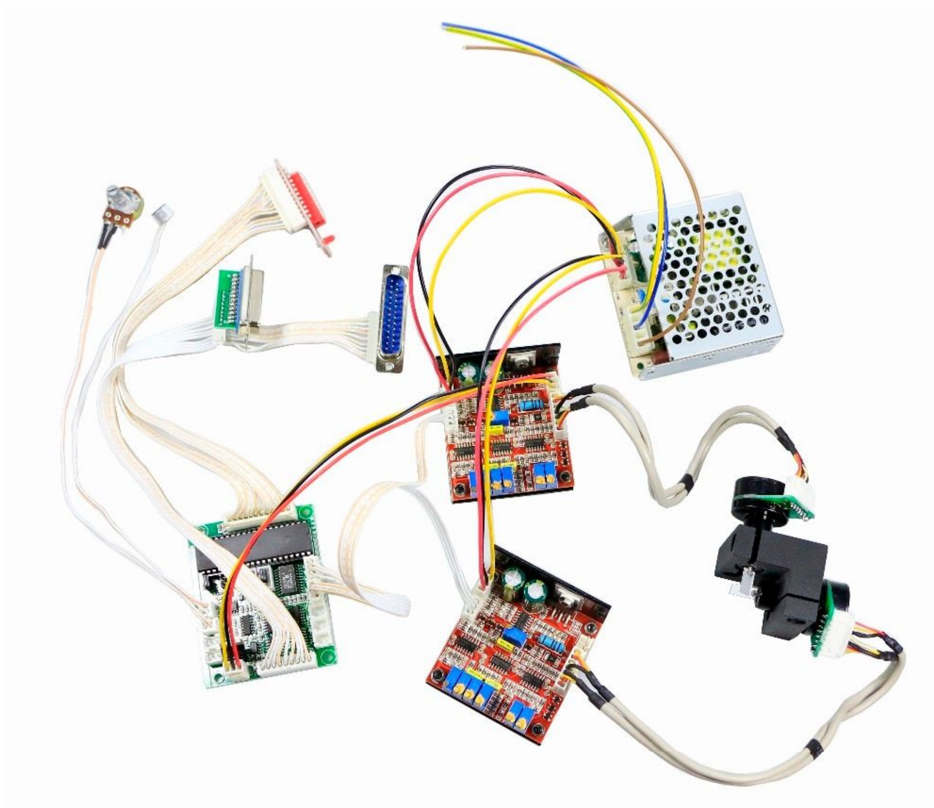

Figure 9: *WonSung Galvanometer [12]*

The red laser that will be shined on to the keys, to indicate where the user should place their fingers, need to be precisely directed onto the correct piano keys. To do this, the projector will need to be able to rotate an attached mirror precisely. After considering the galvanometer and motors such as stepper and servo, it was concluded that all three choices could be precise enough to direct the laser to the designated keys. However, when considering the requirement to displace upto 10 notes at once **[Req5.4.5 - b]**, it would be more cost effective to use low cost galvanometer instead of using higher quality stepper and servo motors that may cost more. The galvo will be controlled using the MCB, Raspberry Pi 3 B+. This meets the requirement **[Req5.4.1 - a]** set forth in our requirements document. The axes of the keyboard is defined in *Figure 10* below.

Table 2: *Specifications of WonSung galvanometer [12]*

Specification	Description
Signal input voltage	$\pm 5V$
Input voltage requirements	$+Vcc: \ +15V/1.0A$ $-Vcc: \ -15V/0.6A$
Operating temperature range	$0-50$ ℃
Maximum optical angle	$\pm 20°$
Scanner speed	At $\pm 20°$: 18Kpps At $\pm 15°$: 25Kpps At $\pm 5°$: 30Kpps
Mirror dimensions ($W \times L \times \lambda\ width$)	$7mm \times 11mm \times 0.6mm$
Weight (including all boards, power supply, and scanning mirrors)	$\leq 800g$

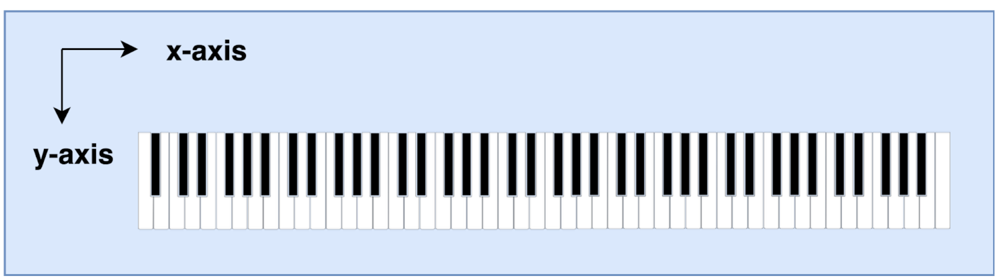

Figure 10: *Definition of axes for piano keyboard [13]*

3.3.1 PoC Prototype

For the initial PoC prototype an one octave range will be controlled as stated in the requirement **[Req5.4.2 - a]**, specifically the octave with the middle C. Additionally, only one key would be lighted at a time to meet the requirement **[Req5.4.3 - a]**. Based on the width of an average white piano key being about 23.5mm [14], an octave would be approximately 188mm wide. Using this width and the $\pm 20°$ maximum optical angle of the galvanometer purchased, the height placement of the galvo is determined using trigonometry.

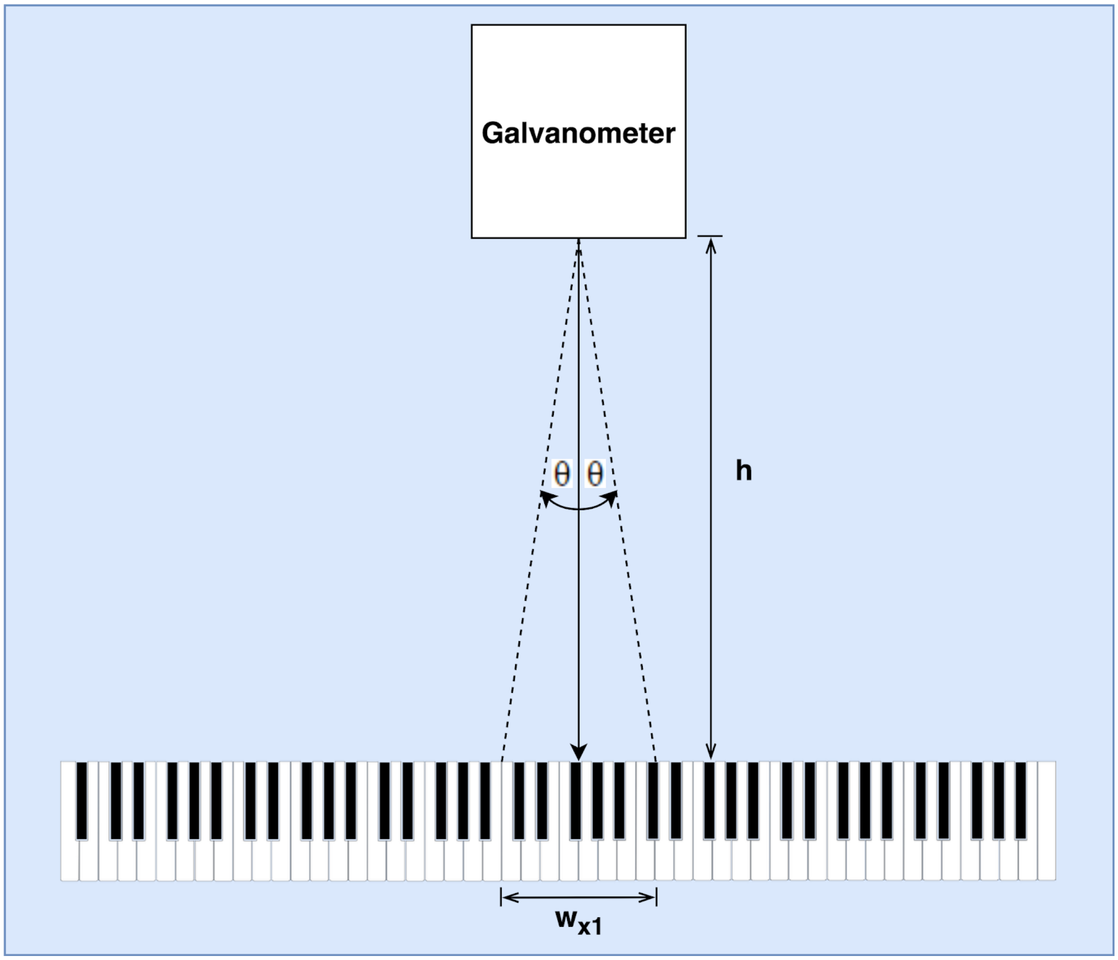

Figure 11: *X-axis projection angle diagram for illuminating one octave on the piano*

X-Axis One Octave Projection Angle Calculation:

Let w_{x1} be the width of an octave, and h the height of the galvanometer from the piano keys. We are given $\theta = 20°$ as the maximum angle for the scanner to reach. Then the following calculations give the minimum height, as height increases when θ decreases.

$$tan\,(\theta) = \frac{opposite}{adjacent}$$

$$\frac{w_{x1}}{2} = \frac{188mm}{2} = 94mm$$

$$tan\,(20°) = \frac{\frac{w_{x1}}{2}}{h} = \frac{94mm}{h}$$

$$h = \frac{94mm}{tan\,(20°)} \simeq 258.2628774mm \approx \mathbf{25.8cm}$$

This value of the height is the minimum value required to illuminate at least one octave on the piano, which is what will be completed in the POC.

3.3.2 Final Prototype

From *Table 2* above, it is noted that the weight is approximately 800g and this would meet our requirement **[Req5.4.6 - b]** to be under 1kg.

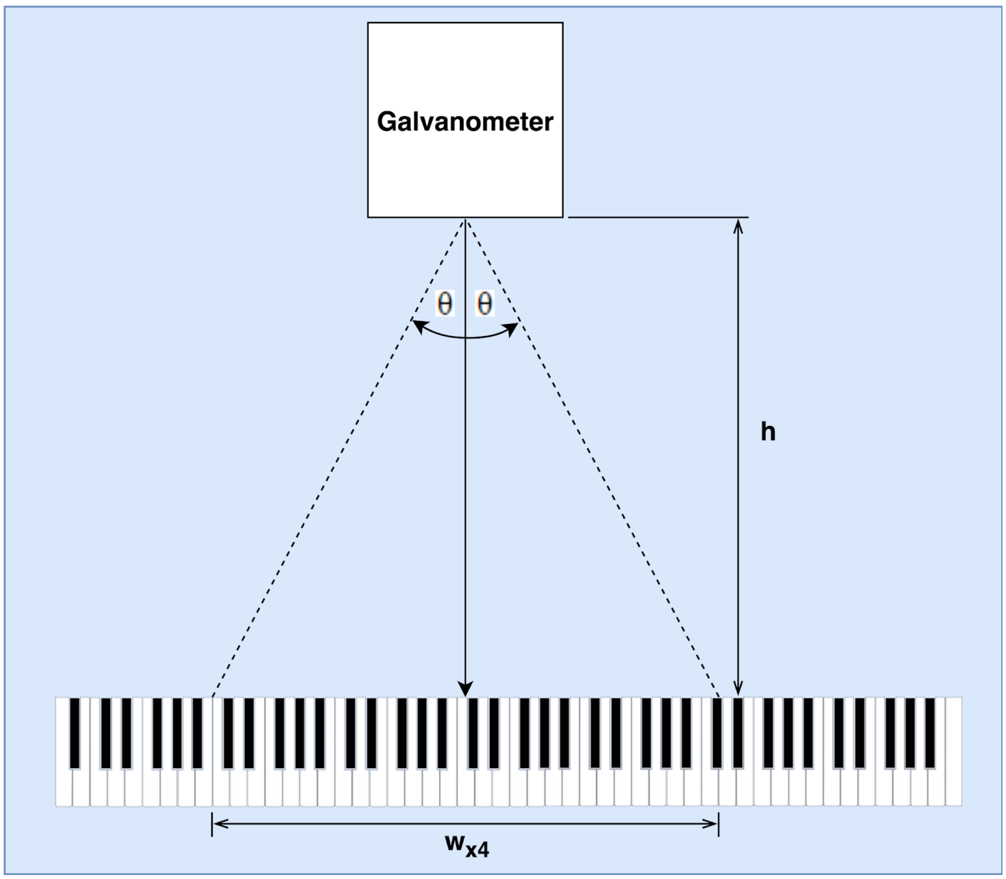

Figure 12: *X-Axis projection angle diagram for illuminating four octaves on the piano*

X-Axis Four Octave Projection Angle Calculation:
The same calculation produces the minimum height for a four octave scanner, which will be delivered in the final prototype. This time, let w_{x4} be the length of four piano octaves, and θ remains the same.

$$w_{x4} = 29\ keys \times \frac{23.5mm}{key} = 681.5mm$$
$$\frac{w_{x4}}{2} = \frac{681.5mm}{2} = 340.75mm$$

$$tan\,(20°) \;=\; \frac{\frac{w_{x4}}{2}}{h} \;=\; \frac{340.75mm}{h}$$
$$h \;=\; \frac{340.75mm}{tan\,(20°)} \;\approx\; 936.2029307mm \;\approx \mathbf{93.6cm}$$

Using the same calculations for the final prototype, the Luminotes device must be at least 93.6cm above the keyboards in order to illuminate four octaves. This would meet the requirement **[Req5.4.4 - b]** [1]

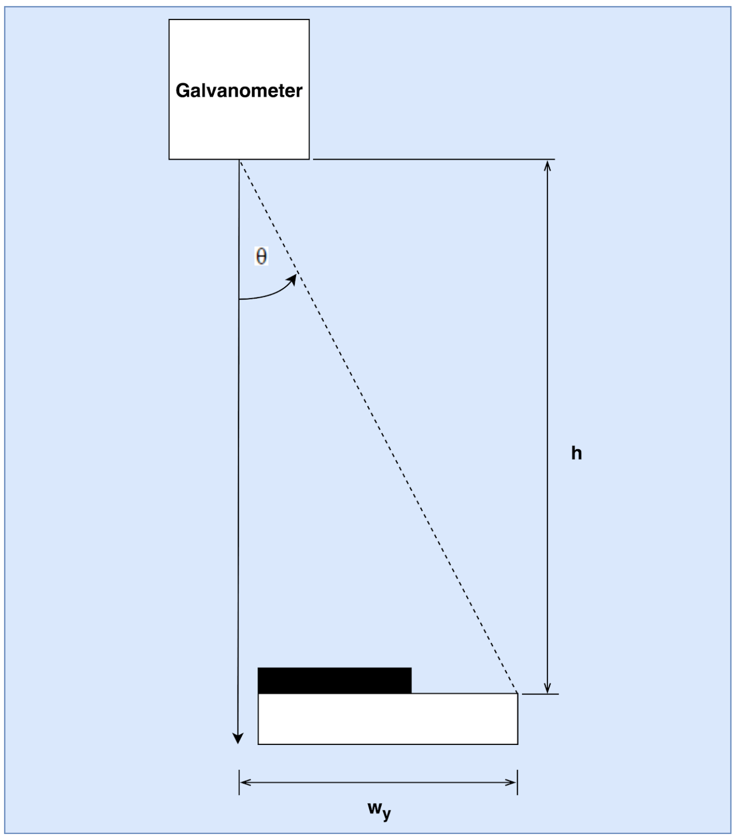

Figure 13: *Y-Axis projection angle diagram*

Y-Axis Projection Angle Calculation:

The galvanometer specs allow for a maximum angle $\theta = 20°$ along the y axis. Here, w_y is the length of a key.

$$tan\,(20°) \;=\; \frac{w_y}{h} \;=\; \frac{w_y}{93.6cm}$$
$$w_y = 93.6cm \times tan\,(20°) \approx 34.06761393cm \;\approx \mathbf{34.1cm}$$

Based on the height of 93.6cm that was calculated for the final prototype height of the galvo from the surface of the piano keys, the y-axis maximum distance that the galvo needs to be from the edge of the piano keys must be 34.1cm. This distance will ensure that the galvo will be able to scan over the full length of all piano keys, which will be necessary for projecting on both the white keys and the shorter black keys. From these calculations, this height can also be applied to the PoC prototype since the minimum height required for the galvo above the keys, to project over an one octave range, is 25.8cm.

5 Firmware Specifications

When designing firmware for our PoC, it is necessary to first understand how data flows across the high level system. Luminotes firmware exists within the MCB. Data travels into it from the mobile application via Bluetooth, and out to the the Galvo Scanner and Laser for projection. Thus fulfilling **[Req. 2.6.7-a]** of the Luminotes Requirements Specification.

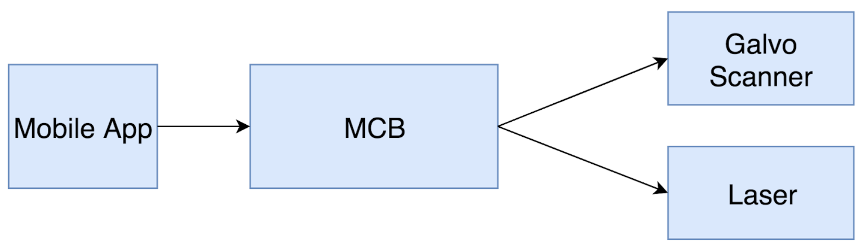

Figure 14: *High Level System Data Flow*

5.1 Operating System

The MCB will run Raspbian, the official OS for Raspberry Pi. Raspbian was chosen because it is based off of Debian, an OS quite familiar to our developers. This allows for the Luminotes firmware source code to be built into a standard binary package (.deb) and integrated with the system. Doing so greatly simplifies installation and uninstallation of the firmware, which in turn greatly simplifies development.

5.2 MCB Dataflow

Within the MCB data travels along one of two paths; either it is MIDI which needs to be transformed into data readable by the Galvo and Laser for projection, or it is a command to control the state of Luminotes. This flow be seen in *Figure 13* below.

If the data is a command, it goes into Command Control which is tasked with state changing. If it is MIDI, the data is stored on the file system, where it is then decoded note by note. The decoded node is fed through a process which extracts angle, timing, and note state information. This data is saved into a transmission buffer. Once all the angles and timings have been saved in the buffer, Luminotes can begin to project onto the piano as data is forwarded to the Galvo and Laser, note by note, in synchrony.

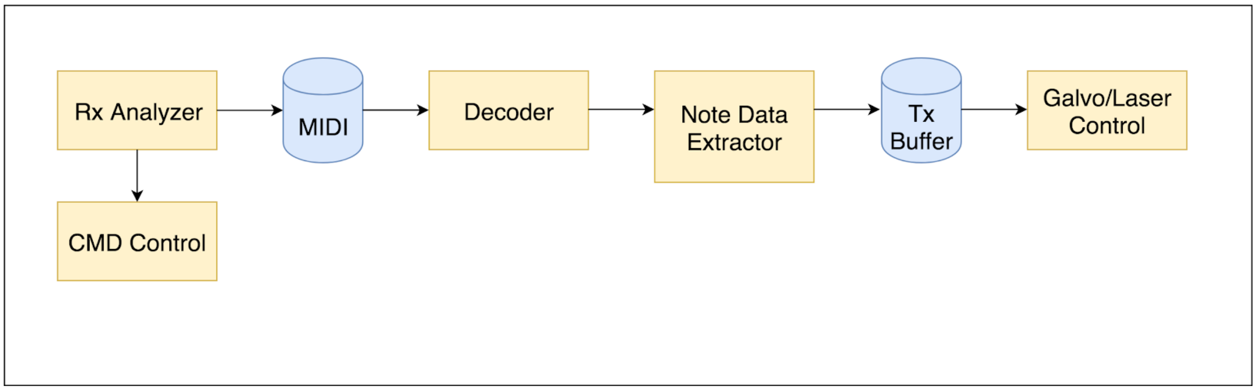

Figure 13: *MCB Dataflow*

5.3 Received File Format

As per **[Req 2.6.8-a]**, the MCB will receive files over Bluetooth, to be processed by the Rx Analyzer. The files received will contain a header, a timestamp, and data. The header will be an ASCII encoding of either “CMD” or “MIDI”. If the file has neither of these headers, it will be discarded . The size of the command packet is 32 bytes, as this is a standard data size and is easy to parse in lower level languages.

Table 3: *MIDI Packet*

4 bytes	4 bytes	Remainder of bytes
Header	Timestamp	Midi file

Table 4: *Command Packet*

4 bytes	4 bytes	24 bytes
Header	Timestamp	Command

5.4 MIDI Decoder

Decoding of MIDI data will be done using the open source Arduino MIDI library. This C++ code simply needs to be slightly transformed to support the Raspberry Pi. Since MIDI data will input via bluetooth rather than a MIDI connector, not many changes to the source code are required. The decoder will be installed as an isolated package, and called upon by the Luminotes main firmware.

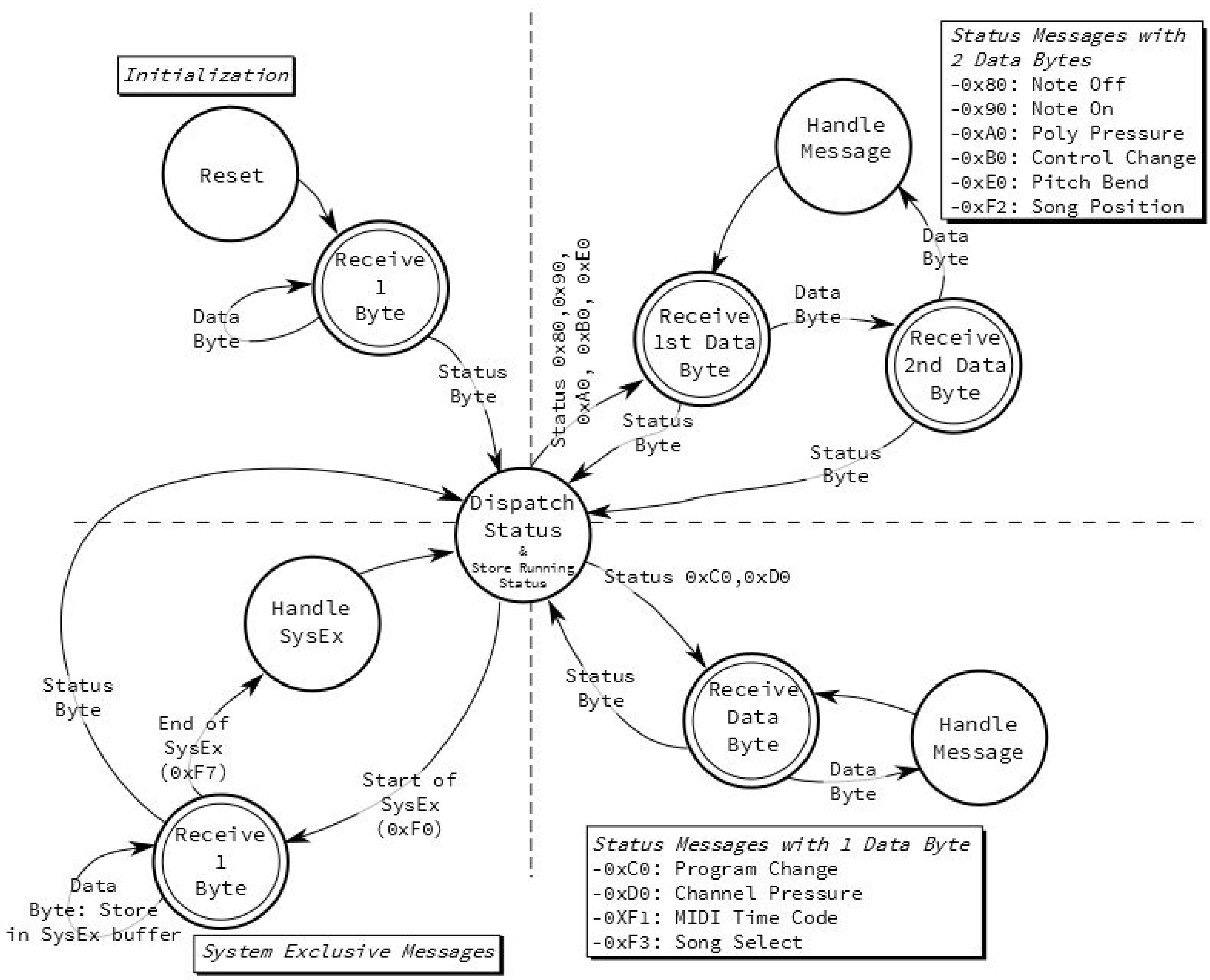

Figure 15: *MIDI Decoding FSM [15]*

5.5 Note Data Extractor

The file created by the MIDI decoder has the format shown in *Table 5* below. The possible note states are on and off. The notes can range from 1 to 88. The note duration is from 0 to 2^32 ticks. These ticks are later converted to time duration, depending on the inputted tempo.

Table 5: *MIDI Data Format*

Note (1 byte)	Note state (1 byte)	Note duration (2 bytes)

After the MIDI file has been read and decoded, the required angles for the galvanometer are calculated by the algorithm shown in *Figure 16*. It is based on the calculations from sections 5.3.1 and 5.3.2. The height is set during calibration, which allows the correct scaling of the projection dimensions, fulfilling requirement **[Req6.2.6 - a]**, and the note is read from the decoded MIDI file. The height and note are passed into the note calculator. The function outputs the x axis and y axis angles for the galvanometers, and stores them in a buffer. This function satisfies the requirement **[Req6.2.5 - a]**.

```
def angle_calculations(note,height):

    unit_distance = distance_LUT(note)
    scale_factor = f(height)
    length = unit_distance*scaling_factor
    width = unit_width*scaling_factor
    x_angle = arctan(length/height)
    y_angle = arctan(width/height)
    return x_angle, y_angle
```

Figure 16: *Galvo Angle Calculation Algorithm*

5.6 Galvo and Laser Controller

This process is tasked with extracting angle data, timing, and note state for each note, and then using that information to control the Galvo and laser. The angles are used to set the Galvo motors, while the note state turns the laser on or off. The laser is turned off during a *rest*; a short period of time where no keys are played. The process then sleeps for the specified time, to allow for the Galvo and laser to transmit the note.

```
def galvo_and_laser_control(file):
    while(txbuf_ready):
        line = file.read_line()
        x_angle = get_x_angle(line)
        y_angle = get_y_angle(line)
        time = get_time(line)
        turn_laser_on = is_laser_on(line)
        set_galvo_x_motor(x_angle)
        set_galvo_y_motor(y_angle)
        if turn_on_laser == True:
            turn_on_laser()
        else turn_on_laser == False:
            turn_off_laser()
        sleep(time)
```

Figure 17: *Galvo and Laser Controller Algorithm*

SIMPLE instruments

5.7 Command Control

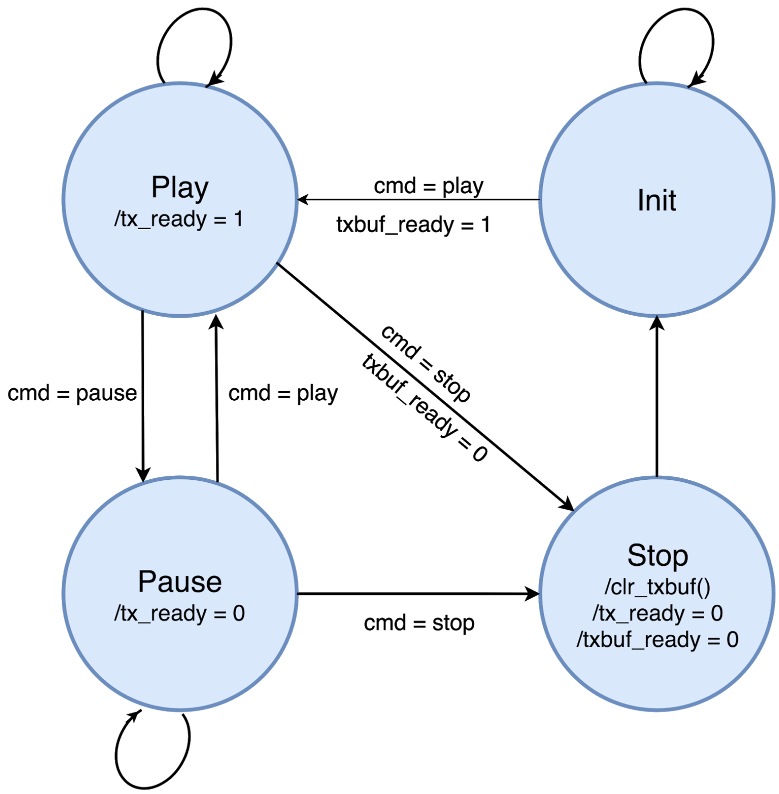

Figure 18: *Command Control State Machine*

When Luminotes boots, it enters the initialization state. Once the transmission buffer for the Galvo and scanner is filled with all the note data, the flag *txbuf_ready* is set. The MCB then continues waiting until it receives a command to start playing. It cannot change state unless *txbuf_ready* is set.

Within the Play state the flag *tx_ready* is set; which tells Luminotes that it is safe to forward data from the buffer to the Galvo and scanner. It stays there until either all data has been forwarded from the buffer and *txbuf_ready* is unset, or a command to pause or stop has been received. In the case that data has reached completion or *cmd = stop*, then Luminotes enters the Stop state. For *cmd = pause* it enters the Pause state.

During Pause, the flag *tx_ready* is unset. Data will no longer be forwarded from the buffer, and nothing will be projected onto the piano. The system is paused until a command to start playing again, or to stop all together, is received.

At Stop, Luminotes is done playing. The buffer is cleared, and both transmission flags are unset. The system returns to the Init state where it waits to be used again.

6 Structural Requirements

6.1 Dimensions

6.1.1 Components

The dimensions of the two driver boards are 74mm x 50mm x 25mm. We utilized these dimensions alongside pixel measurements to approximate the size of the other components and thus the chassis requirements. It should be known that due to a lack of information about *Figure 9*, our dimensions will be rough estimates as they do not include depth of field and thus Parallax considerations.

Table 5: *The estimated dimensions for the Luminotes device*

Component	Length (mm)	Width (mm)	Height (mm)
Main Board	74	50	9
Driver Board[1]	74	50	25
Power Source	84	78	14
Galvo	81	81	30
Raspberry Pi[1]	82	56	19.5

[1] The measurements for these components have been given by the manufacturer.

6.1.2 Components

By applying the dimensions in *Table 5*, we are able to predict a structure for the PoC prototype structure. *Figure 17* demonstrates an orthographic drawing of the PoC prototype head, arm, and body as explained in subsequent sections.

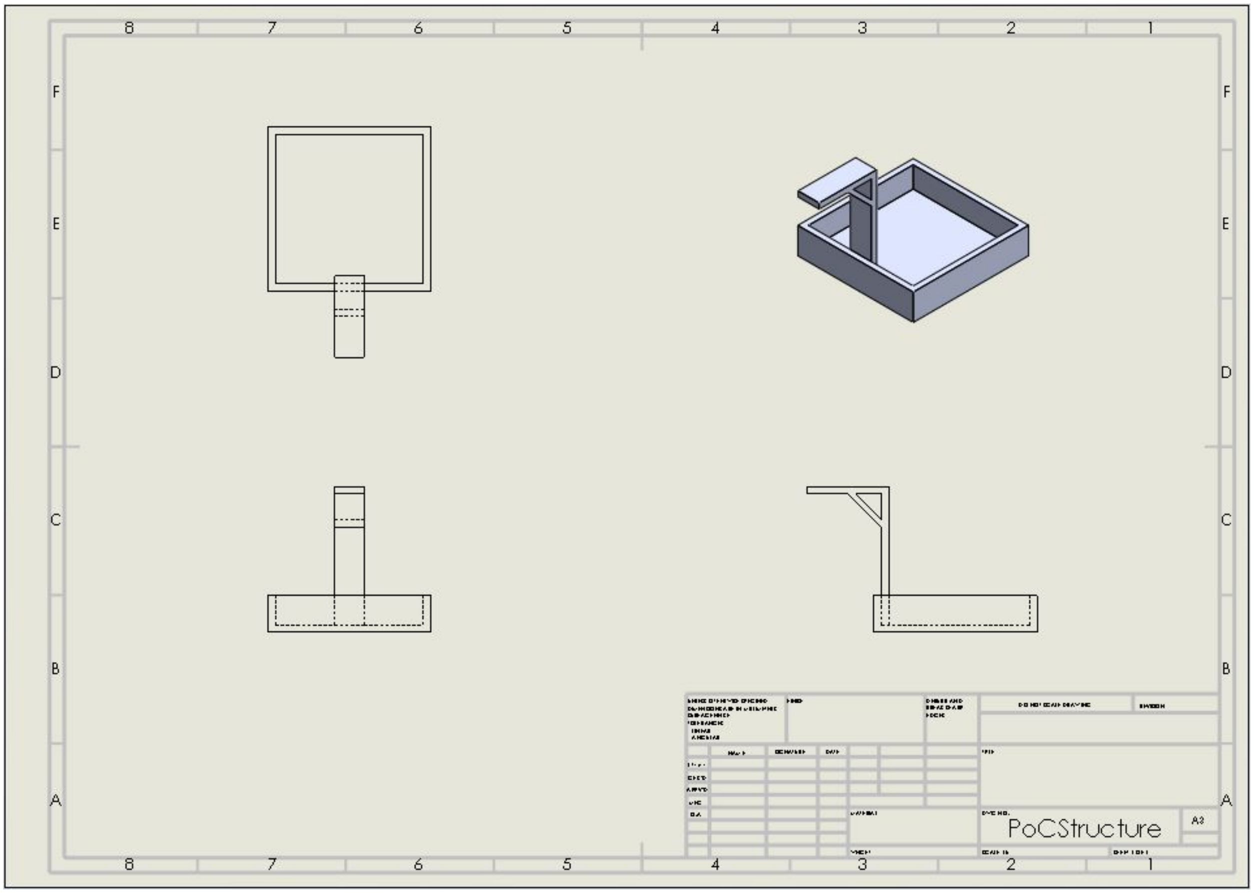

Figure 19: *PoC prototype structure Solidworks design*

6.2 Materials

6.2.1 PoC Prototype

Our PoC prototype will be made from wood, glue, screws, and L brackets due to time constraints and the concentrating on hardware and firmware. This is also a cost-effective solution that will allow us to take advantage of scrap materials. The wood and glue will act as additional reinforcement to the combination of screws and L brackets to hold together the head, arm, and body.

6.2.2 Final Prototype

Our final prototype will be 3D-printed using PETG filament, which combines the best of ABS and PLA, allowing for both easability and durability. There are a number of options available for smoothing the chassis, ranging from sanding and applying to an epoxy resin to vapor smoothing. We can use plastic solvent or cement to combine the different components, which will be important due to us splitting the head and body in half in order to install the hardware components prior to assembly [17].

6.3 Head

6.3.1 PoC Prototype

The head will comprise of a piece of wood that is large enough to hold the optics scanner and the laser that we are using [**Req7.2.5 - a]**.

6.3.2 Final Prototype

The optical scanners length and width are equal, allowing for us to use a sphere that contains a slit where the laser will project [**Req7.2.3 - a]**. For safety purposes, the front and side of the sphere will extrude outwards like a shield, blocking the laser's initial output, as stated in **Req7.1.1 - a**. We chose this design to prevent eyesight damage and distractions to those within line of sight of the device.

6.3 Arm

6.3.1 PoC Prototype

The arm will connect both the head and the body, and look like an L. The L will include two pieces of wood: the first of which extrudes from the body, and the second of which extends far enough to be above the piano keys. The arm will need to be sturdy enough to withstand vibrations, which we believe will be possible given our chosen materials. To further reduce the risk of vibration, as well as the chance of the arm falling off the body, we will reinforce the base with another piece of wood acting as a bracket.

6.3.2 Final Prototype

We chose an arm that is capable of changing its height at the user's will. To ensure the arm retains its position, we opted for a hollow heavy-duty flex arm from Moffatt Products [**Req7.2.1 - b]**. This allows us to place the boards and power source in the body, and use the arm for holding the wires alone.

Both the head and body will contain the corresponding female head to allow for the arm to attach to. This will allow for seamless integration between the head and arm, and prevent detachment in the case of a fall.

Figure 20: *H/D Flex Arm, hollow ⅝-27 UNF male fittings on both ends*

Table 6: H/D Flex Arm, hollow ⅝-27 UNF male fittings on both ends specifications

Component	**Description**	**Requirement**
Length	12”	**[Req7.2.4 - b]**
Top Fitting	5/8-27 UNF female thread	
Base Fitting	5/8-27 x .500″ UNF male thread	
Outside Diameter	.695″ covered diameter	
Inside Diameter	.325″	
Max Load Capacity	3 lbs	**[Req7.1.2 - b], [Req7.2.2 - b]**

6.4 Body

6.4.1 PoC Prototype

The body will consist of wood with wooden walls on the side. The walls will prevent any damage to the components and wiring during construction.

6.4.2 Final Prototype

Given the use of four boards and one power source of similar size, we chose a cylinder as our base shape to maximize the space we use.

The four boards will be attached to the walls of the device, allowing for closer connections at the cost of more complex construction. To prevent the device from being topheavy and thus falling over, we will distribute the weight in the back. The main two areas we will add weight is in the bottom, with some weight in the top if needed. As the power source is heavier, it will remain on the bottom to act as a weight. The goal will be to always have a centre of gravity that is within the body, regardless of the arm's horizontal displacement [**Req7.2.7 - b]**.

Since we plan to encase all the components within a chassis [**Req7.2.1 - a]**, we investigated whether or not a cooling system—from vents to a heat sink fan—is needed. As we do not have access to the components currently, we looked at the requirements of the individual components that could use cooling. The Raspberry Pi operates between 24°C and 40°C, for the unused ports and the quad-core CPU respectively. If the CPU exceeds 85°C the CPU will stabilize the temperature by throttling the frequency [16]. Similarly, the galvanometer does not surpass a temperature of 50°C [12]. As these two are the main components that could cause overheating, we do not believe that a cooling system is necessary for our final prototype. However, we plan to include ventilation to allow for the heat to diffuse into the surrounding area.

7 Conclusion

Luminotes is a device to assist beginners to learn to play songs on the piano. The main components are the MCB, the mobile application, and the projection device. The user interacts with the device via the mobile application. The design specification for the proof of concept is divided into three sections: firmware, hardware, and structural. For the prototype this will also include a software section.

The firmware decodes a MIDI file and uses height calibration information to control the galvanometric scanning mirrors and the laser. Decisions related to processes, algorithms, and data formats were chosen due to constraints from the galvo and the MCB.

The hardware consists of an MCB, a laser, and a 2-axis galvanometer. Safety was a big consideration in choosing a laser. The galvanometer was chosen to be precise and fast enough to scan angles of a fraction of a degree. The MCB was chosen to support multiprocessing.

The structural component discusses the PoC and final prototype; goes into detail about the implementation of the materials used, head, arm, and body.

All of these design specifications allow our product to be easy to use, and how not to interfere with regular piano use. Furthermore, these design specifications show how we will satisfy our requirements specifications.

8 References

[1] S. A. Azhar, C. Finnigan, H. C. E. Lin, D. Pukanich, A. Syed, W. H. Xu, "Simple Instruments Requirements and Functional Specifications," Burnaby, BC, 2019

[2] www.raspberrypi.org. (2019). *Raspberry Pi 3B+ Specs Benchmarks.* [Online] Available at: https://www.raspberrypi.org/magpi/raspberry-pi-3bplus-specs-benchmarks/ [Accessed 14 Mar-2019]

[3] "Fundamentals of Lasers," *Edumund optics* [Online]. Available at https://www.edmundoptics.com/resources/application-notes/lasers/fundamentals-of-lasers/ [Accessed 14-Mar-2019]

[4] "Laser Systems for Optical Microscopy" *Microscopy Resource Center* [Online]. Available at http://olympus.magnet.fsu.edu/primer/techniques/microscopylasers.html [Accessed 14-Mar-2019]

[5] "Rayleigh Length" *Wikipedia* [Online] Available at https://en.wikipedia.org/wiki/Rayleigh_length [Accessed 14 Mar-2019]

[6] "Phys 6.2: Rays and geometrical optics" *The Open University. [Online]* http://www.met.reading.ac.uk/pplato2/h-flap/phys6_2.html [Accessed 14 Mar-2019]

[7] "VLM-635/650-01 Series" *Infiniter* [Online] Available at https://www.quarton.com/download/9/ [Accessed 14 Mar-2019]

[8] "Safety of machinery -- Laser processing machines -- Part 2: Safety requirements for hand-held laser processing devices, "ISO 11553-2:2007, *ISO,* 2007 [Accessed 6 Feb. 2019].

[9] "Lasers and laser-related equipment -- Laser device -- Minimum requirements for documentation," ISO 11252:2013, *ISO,* 2013 [Accessed 6 Feb. 2019].

[10] "Functional safety of electrical/electronic/programmable electronic safety related systems — Part 1: General requirements" *Standard Council of Canada*. [Accessed 6 Feb. 2019]

[11] S. Michael. "Human Factors & Usability Engineering." SFU,Burnaby. 2018

[12] "Faster shipment:30Kpps laser Galvo Galvanometer Based Optical Scanner (including Show Card) for Stage Laser 3D Printer" *AliExpress* [online] Available at: https://www.aliexpress.com/item/Hot-30Kpps-laser-Galvo-Galvanometer-Based-Optical-Scanner-including-Show-Card/1905934723.html?spm=2114.search0104.3.23.27b55fc9vKNMsH&ws_a

b_test=searchweb0_0,searchweb201602_4_10065_10068_10130_10547_319_10059_10884_317_10548_10887_10696_321_322_10084_453_10083_454_10103_10618_10307_537_536_10902,searchweb201603_61,ppcSwitch_0&algo_expid=82314ccf-8fee-42d3-ae46-3baa3e2fde23-3&algo_pvid=82314ccf-8fee-42d3-ae46-3baa3e2fde23 [Accessed 14 Mar. 2019].

[13] OpenClipart-Vectors / 27429 images, *Pixabay* [online] Available at: https://pixabay.com/vectors/full-size-keyboard-music-piano-2024898/ [Accessed 14 Mar. 2019].

[14] “Musical keyboard” *Wikipedia* [online] Available at: https://en.wikipedia.org/wiki/Musical_keyboard [Accessed 14 Mar. 2019].

[15] Learn.sparkfun.com. (2019). *Implementing Midi.* [Online] Available at: https://learn.sparkfun.com/tutorials/midi-tutorial/implementing-midi [Accessed 14 Mar-2019]

[16] “Does the Raspberry Pi need a cooling system?” *Raspberry Pi.* [Online] Available at: https://raspberrypi.stackexchange.com/questions/22928/does-the-raspberry-pi-need-a-cooling-system [Accessed 14 Mar-2019]

[17]
“How To: Post-Processing PETG 3D Filament” *MatterHackers.* [Online] Available at: https://www.matterhackers.com/articles/how-to-post-processing-petg-3d-filament [Accessed 14 Mar-2019]

Appendix: User Interface and Appearance Design

Introduction

The User Interface Design and Appearance Design document will go more in depth on the aforementioned topics in regard to Luminotes. The main purpose of this document is to focus on user interactions, which we strive to keep simple and intuitive.

Background

Readers will benefit from knowledge of understanding human factors and usability, as we use such jargon to explain how we will make the user's experience positive.

User Analysis

The device's target audience are causal piano enthusiast who want to learn to play the piano without learning the theory required to understand sheet music. We are hoping to maximize our market by creating a device that is easy-to-use and affordable for piano players. One way we plan to do this is through the use of a smartphone. The app design will use Android UI practices to ensure ease of use, and adhere to the following requirements:

- User must be able to have hand motor function to play the piano
- User must be able to use a smartphone

The user is expected to meet all the requirements above.

Technical Analysis

Discoverability

Hardware: There hardware is as simple as possible. The stand is similar to that of a lamp. A power switch will start the device, and a LED will show the system status. There will be a power switch to start the device. User will adjust the laser scanner the same way the user will adjust a lamp for light. All the user will need to do is plugin the device, turn it on, and then adjust the scanner like a lamp.

Software: The software will untitive as well. As soon as they start, a list will be shown will all the songs available to play and some information about the songs. Once the user select the songs, the page will look very similar to a music player that all smartphones have. The mentioned pictures are below.

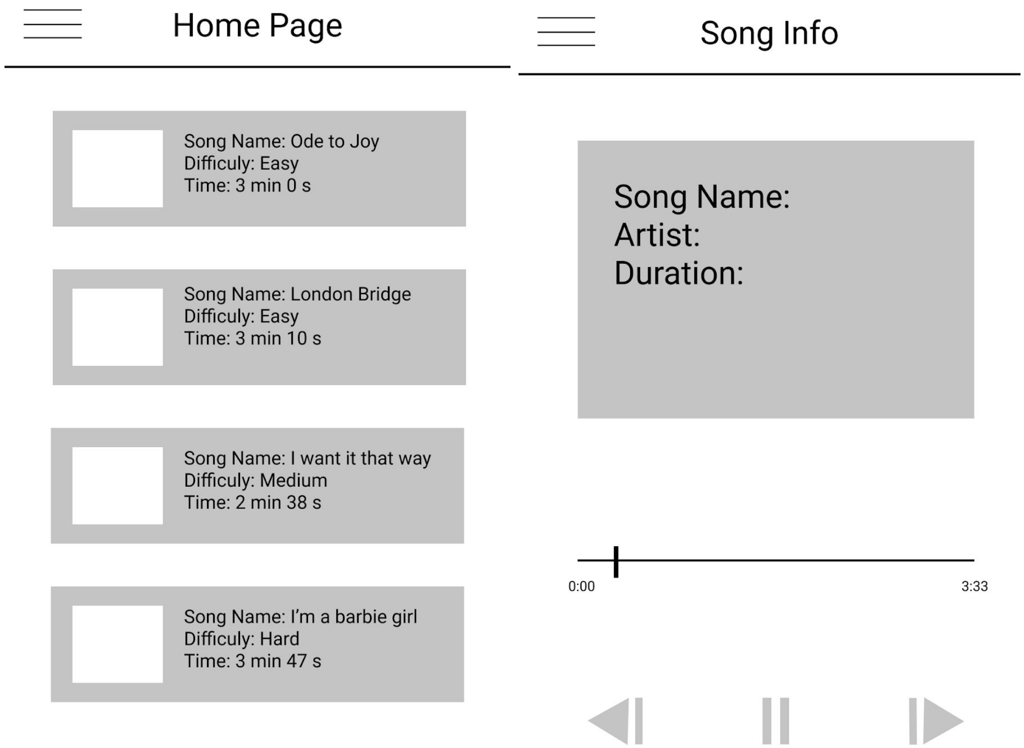

Figure 21: *Home page (left) and Song Screen (right)*

Feedback

Hardware: The device will have an LED to showcase the state of the device. Here is the color coding and its meaning:

- Red: Error
- Blue: Bluetooth on and ready to pair
- Green: Bluetooth Paired and ready to play song

Software: It will show let the user play the song again or go to a different full song list as shown in figure 20.

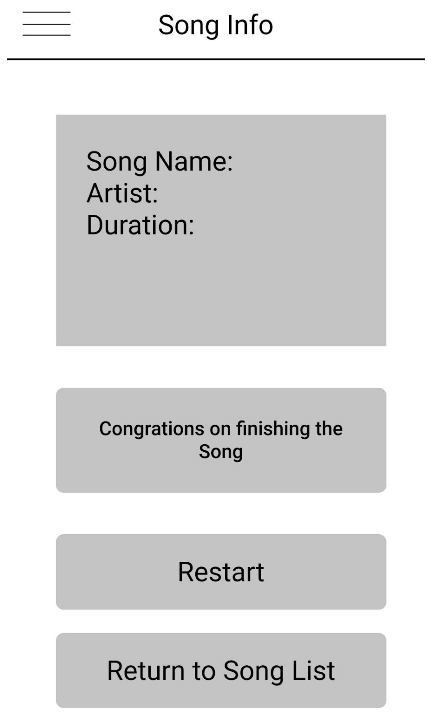

Figure 22: End Song Screen

Conceptual Models

Hardware: The LED light will help conceptually see the status the device. The colors corresponding accordingly.

Software: The headings of the different pages will help provide certain features.

Affordances

The affordances for will be done in such a way that it will consider every aspect of the device. The laser, we considered safety to in order to prevent the laser ever being shined on the eyes directly. The different size of keys, where our device considers this when the user is calibrating the device. The base of the holder is angled in such a way that you know which way is forward and you know where the scanner is based on the arm.

Signifiers

The LED indicator to showcase the status of the device to show if its ready to play a song.
The app will also indicate the how much time is remaining in the song and will notify the user when the song is finished so the user know exactly what state the device is in.

Mapping

Hardware: There is no mapping for the hardware. The power button will be the only button, which will have a label to indicate its power button. The laser will have a one to one map to each of the keys.

Software: The software mapping will be such that the heading is at the top, the back button will be at the top. This follows the standard that Google has set for Material Design. For more pages, the user can open the menu on the top left. Action buttons will near the bottom of the page.

Constraints

Hardware: The is constraints of the currently will be for one octave for the piano. The other craints will be the adjustment that the arm can provide. The arm will be used to adjust the laser as it needs to be matched to line up the keys as shown in the figure below. We will not provide any more adjustments to keep a limited number actions for the user. We will only have one note at a time for the user to play for phase 1 and increase it for phase 2.

Software: The songs list will be specifically catered to only have songs that only have a piano component to not allow confusion for the user of what songs are playable for the user. The UI will be forced to be portrait mode to not allow more confusion for the user.

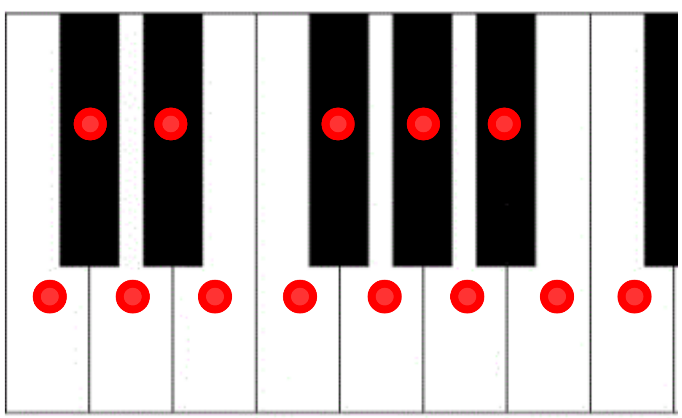

Figure 23: *Showing lasers projection on the piano*

Engineering Standards

The Luminotes device and app will adhere to the following standards.

Laser

Table 7: *Laser Standards*

ISO 11553-2:2007	Safety of machinery -- Laser processing machines -- Part 2: Safety requirements for hand-held laser processing devices [8]
ISO 11252:2013	Lasers and laser-related equipment -- Laser device -- Minimum requirements for documentation [9]

Electrical

Table 8: *Electrical Standards*

CAN/CSA-C22.2 NO. 61508-1:17	Functional safety of electrical/electronic/programmable electronic safety-related systems — Part 1: General requirements [10]
CAN/CSA-C22.2 NO. 0-10	General Requirements - Canadian Electrical Code, Part II
UL 60950-1	Information Technology Equipment - Safety - Part 1: General Requirements
UL 60065	Standard for Audio, Video and Similar Electronic Apparatus - Safety Requirements [10]
IEC 60065:2014	Audio, video and similar electronic apparatus - Safety requirements

Mobile Device and MicroController

Table 9: *Mobile Device and Microcontroller Standards*

IEEE 1003.13-2003	IEEE Standard for Information Technology - Standardized Application Environment Profile (AEP) - POSIX(R) Realtime and Embedded Application Support
IEEE 1625-2008	IEEE Standard for Rechargeable Batteries for Multi-Cell Mobile Computing Devices
IEEE 802.15.1-2002	IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN - Specific Requirements - Part 15:

	Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)

Environmental

Table 10: *Environmental Standards*

CAN/CSA-ISO 14040-06 (R2016)	Environmental Management - Life Cycle Assessment - Principles and Framework

Technical Analysis

Testings

Both testings will follow a numbering system to decide how bad the problem is [11]:

0 = not a problem
1 = cosmetic
2 = minor, low priority to fix
3 = major problem
4 = catastrophe, must be fixed before product is released

Analytical Usability Testing

Simple Instruments team will test the usability of our product. Before getting a user to test it, the team will test for learnability, efficiency, memorability, errors and satisfaction.[11]

Mobile:

1. The app UI is easy to navigate
2. The app displays warning clearly
3. The app displays errors clearly
4. The app explained instructions clearly

Laser:

1. The laser projects on to the piano clearly
2. The laser projects on to the piano safely
3. The laser will not injure skin/eyes
4. Easy to understand which key to play based on the projection from the laser

Mount:

1. Easy to put the Luminote product on to the piano
2. The Luminote product will not tip over

Empirical Usability Testing

It is very important must that the device is gone through thorough user testing to ensure quality of our product. We want to make sure the device isn't frustrating to use, and it safe on the eyes even for long period of time. During phase 1, our functionality will be very limited. How ever we will be doing user to testing our prototype to get initial reaction from casual pianist or want to be pianists to see what changes are required to meet our goals. We will document the experiences and opinions. For our prototype, we will also make sure to talk to the professor as well.

Some tasks we will ask users to do:

Task 1

Turn on the device, align the laser by adjusting the arm, and then play a note

Questions:

1. What is turning on the device easy and fast?
2. Was aligning the laser easy and intuitive to do?
3. Was it difficult to move the arm?
4. Was the laser on the note you played easy to see?

Task 2

Play the easy song that comes after setup

Questions:

1. Was the song tempo to fast or two slow?
2. Was it easy to follow the laser to play the note that needs to be played?
3. Did playing this song feel your eyes are strained from the reflection of the laser?

After collecting this feedback, Simple Instruments will see what adjustments and modifications will need to be done.

Summary

A good UI is crucial for the success of a product. It must be enticing and easy for users. As such we implement various measures such as LED lights to indicate system status, to increase user experience. Future versions of Luminotes will include the capability to adjust the laser to work with different piano types based upon the width of the device. Arm movement will be similar to that of the average office lamp; making adjustment easier. Regardless all products have constraints. Ours will initially be constrained to a single octave and key at a time. This number will be increased for the final prototype.

The app contain songs specifically catered towards learning to play piano at different stages of experience. This prototype will be tested by users, allowing for modifications for the final prototype. The following will be considered in our testing; ease of use, learning and lack of eye strain.

Test Plan for Phase 1

HIGH LEVEL	
Mobile App	**Comment**
1.The app must provide a method for the user to select a MIDI song file from their phone Yes ( ) No ( )	
2. The app must be able to send instructions, over Bluetooth, on note timing to the microcontroller based on the user's selection of song and tempo. Yes ( ) No ( )	
Microcontroller	**Comment**
1.The microcontroller must control the laser to light up the correct piano key at the correct time and duration. Yes ( ) No ( )	
2. The microphone must record the piano audio in a suitable quality and output this to the microcontroller. Yes ( ) No ( )	
SOFTWARE	
Mobile App	**Comment**
1.The app will take less than 50 MB of storage space on the mobile device. Yes ( ) No ( )	
2.The app will support operating systems between Android 5.0 and Android 8.1. Yes ( ) No ( )	
3.The app will detect that the mobile device is paired with the device. Yes ( ) No ( )	

4.The app will instruct the user on how to connect to the device if it is not already paired. Yes ( ) No ( )	
5.The app will send instructions to the device one file at a time. Yes ( ) No ( )	
6.The app will send the instructions to the device over Bluetooth in less than 10 seconds. Yes ( ) No ( )	
7. The app will display warnings to the user with less than 1 second of delay. Yes ( ) No ( )	
8.The app will display errors to the user with less than 2 seconds of delay. Yes ( ) No ( )	
9.The app will successfully connect to the device 90% of the time. Yes ( ) No ( )	
10. The app will successfully send the instructions 90% of the time. Yes ( ) No ( )	
11.The app will notify the user if the instructions fail to send to the device. Yes ( ) No ( )	
12.The app will notify the user if the pairing to the device fails. Yes ( ) No ( )	
13.The app will allow the user to play, pause, stop, restart, and repeat the song. Yes ( ) No ( )	
14.The app will allow the user to enable or disable the metronome on the device. Yes ( ) No ( )	
15.The app will allow the user to change the tempo of the note projection.	

Yes ( ) No ( )	
16.The app will have at least one MIDI file as part of the app. Yes ( ) No ( )	
HARDWARE	
Microcontroller	**Comment**
1.The MCB will support Bluetooth connection. Yes ( ) No ( )	
2.The power LED will be enabled when in use. Yes ( ) No ( )	
3.The power LED will blink slowly when on less than 20% of its charge. Yes ( ) No ( )	
4. The MCB will enabled and disabled by the power button. Yes ( ) No ( )	
5.The connection LED will be enabled for a successful connection to the app. Yes ( ) No ( )	
6.The microphone LED will be enabled when the microphone in use. Yes ( ) No ( )	
7.The MCB will receive input from the microphone. Yes ( ) No ( )	
8.The MCB will be ADC capable. Yes ( ) No ( )	
9. The MCB will send instructions to the projector based on the tempo chosen by the user in the app. Yes ( ) No ( )	
10.The MCB will have enough processing capabilities to support onboard DSP. Yes ( ) No ( )	

11.The MCB will have at least 32-bit SoC architecture. Yes ( ) No ( )	
12. The MCB will have at least 1 MB of RAM. Yes ( ) No ( )	
13.The MCB will have at least 2 MB of Flash. Yes ( ) No ( )	
14.The MCB will have Flash with read-while-write support. Yes ( ) No ( )	
15.The MCB will have a small form-factor no more than 80g Yes ( ) No ( )	
16.The MCB will support GPIO connections Yes ( ) No ( )	
Laser	**Comments**
1.The lasers will be replaceable. Yes ( ) No ( )	
2.The lasers will be easy to take out and put into the projector. Yes ( ) No ( )	
3.Laser will be externally powered by the MCB (Raspberry Pi) Yes ( ) No ( )	
Projector	**Comments**
1.The projector will display lasers on the corresponding keys based on the instructions from the MCB. Yes ( ) No ( )	
2.The projector will be able to project light across a range 1 octave of keys on the keyboard. Yes ( ) No ( )	
3.The projector will be able to project 1 note at a time.	

Yes ( ) No ( )	
FIRMWARE	
Micocontroller	**Comments**
1.The firmware will process the music within 5 seconds per 1 minute sample. Yes ( ) No ( )	
2. The firmware will correctly map a given note to the x and y galvo motor angles, to hit the corresponding key. Yes ( ) No ( )	
3. The firmware will will correctly scale the projection, depending on the height that the user calibrates the device to. Yes ( ) No ( )	
4.The firmware will receive from the mobile application and transmit to the galvo scanner and laser Yes ( ) No ( )	
5.The firmware will receive MIDI and Command data via Bluetooth Yes ( ) No ( )	
STRUCTURAL	
Head	**Comments**
1.The mount will be long enough to not obscure the user's ability to use the piano. Yes ( ) No ( )	
2.The mount will remain still regardless of common exterior movement. Yes ( ) No ( )	
3.The head will allow the projector to display the lasers without self-interference. Yes ( ) No ( )	

4.The head will shield the user and onlookers from the eye damage. Yes ( ) No ( )	
5.The head will only contain the components for laser projection. Yes ( ) No ( )	
Arm	**Comments**
1.The mount will be flexible enough to be moved with a single hand. Yes ( ) No ( )	
2.The mount will be able to be adjusted while retaining its height. Yes ( ) No ( )	
Body	**Comments**
1.The chassis will contain a slot for the power LED. Yes ( ) No ( )	
2.The chassis will contain a slot for the pairing LED. Yes ( ) No ( )	
3.The chassis will display labels for the interactable components. Yes ( ) No ( )	
4.The chassis will contain a slot for an external power cord. Yes ( ) No ( )	
5.The body will firmly hold its contents in place when assembled and moved. Yes ( ) No ( )	
6.The body will not fall over, regardless of the arm position. Yes ( ) No ( )	