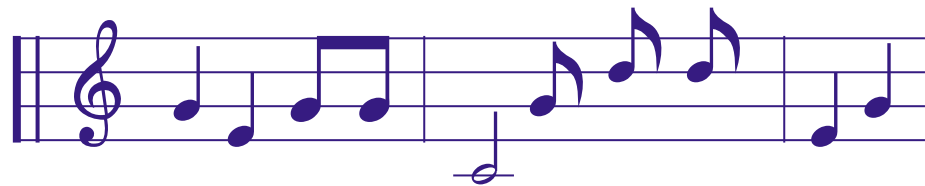


Self-Playing Bass Guitar

Intro to Engineering Design ENGS 452
Spring 2025



Will Lamm, Carter Mann
Andnet DeBoer, Amy Drelicharz





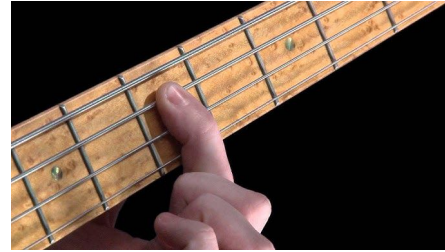
Project Goals

Investigate the ability of incorporating modern robotics into the creative and technical space of making live music.



How to play the Bass Guitar

Fretting Notes



Plucking String
(using fingers or pick)



Damping String
(with plucking hand or
fretting hand)



Requirements

Notes Per Second $>$ 5 notes

Force to pluck string $>$ 4.5N

Force to press fret $>$ 70N

Force to damp string $>$ 4.5N

Cost $<$ \$500

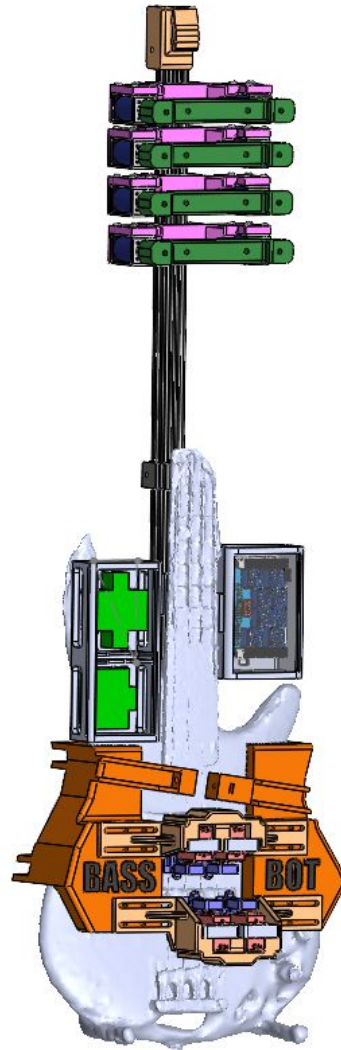
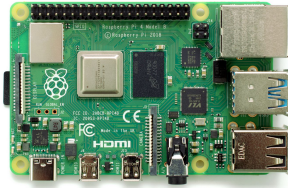
of Notes $=$ 20



Final Design

Flow Diagram

Bass Bot App

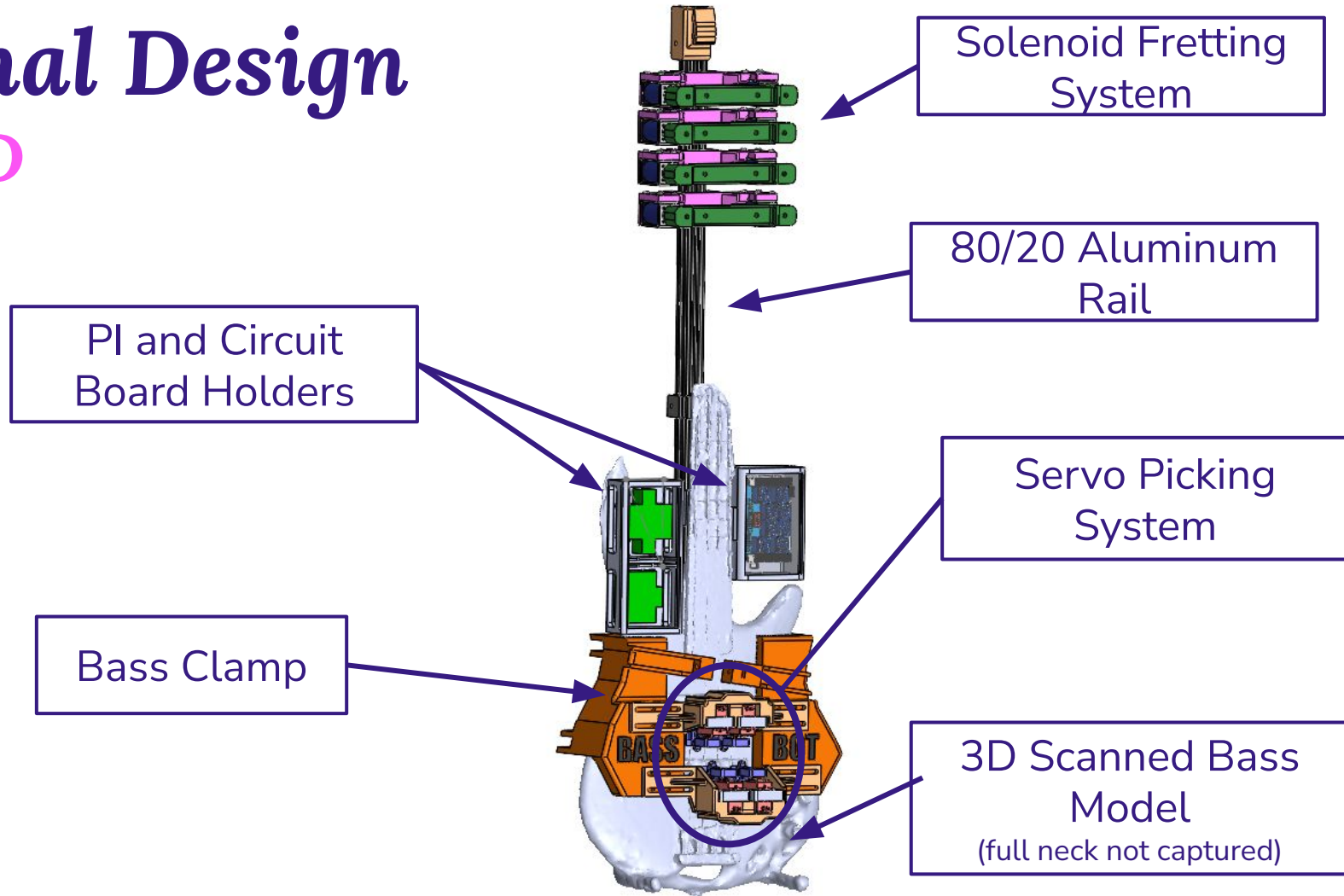


Amplifier



Final Design

CAD



Final Design

PI and Circuit
Board Holders

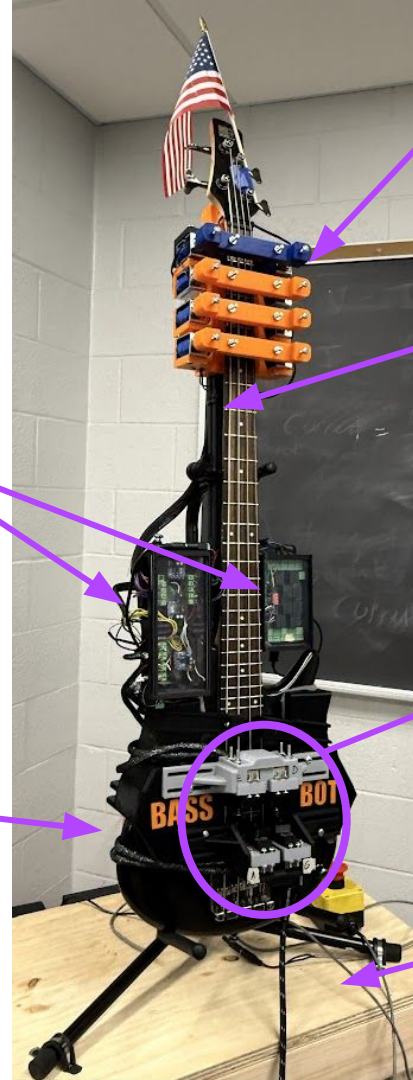
Bass Clamp

Solenoid Fretting
System

80/20 Aluminum
Rail

Servo Picking
System

Stand and Table

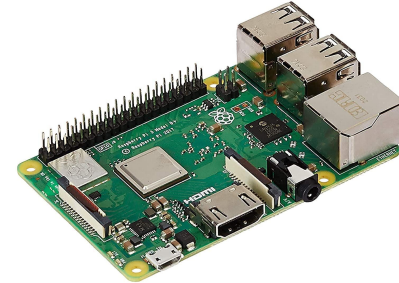
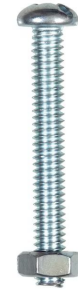


Design Process | Tenth times the charm!



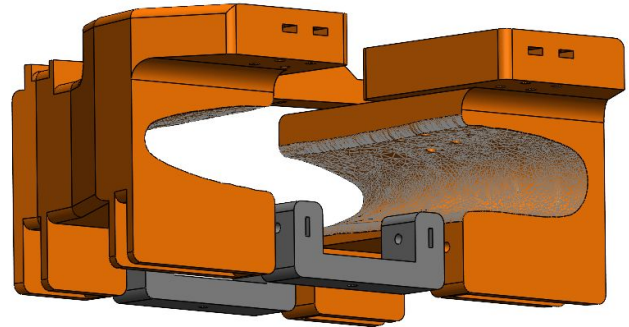
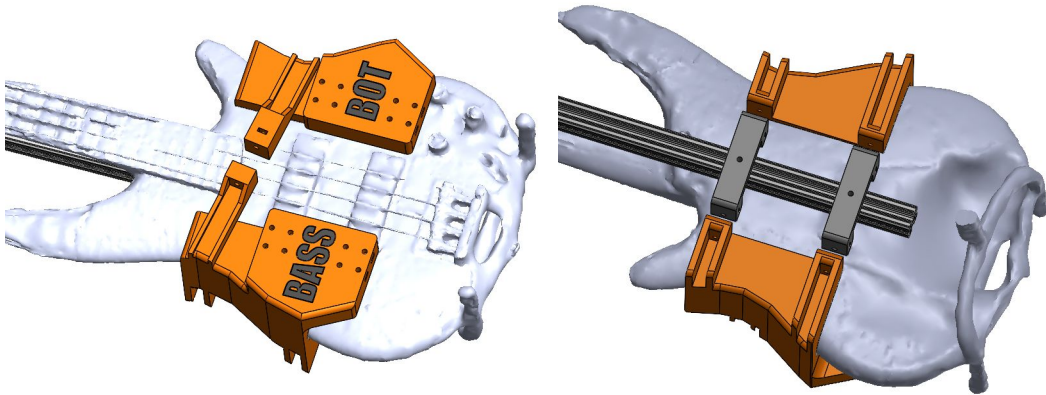
Materials Used

- 3D printed PLA:
 - Bass clamp
 - Fretting brackets
 - Servo brackets and damping system
 - PI and circuit board holders
- Purchased:
 - 80/20 aluminum extrusion
 - Picking servo motors
 - Fretting solenoids
 - Fasteners
 - Bass stand
 - Relay Board
- Misc:
 - Already owned bass guitar and amplifier
 - Built wooden table
 - Found Raspberry PI
 - Custom built circuit boards and wiring systems



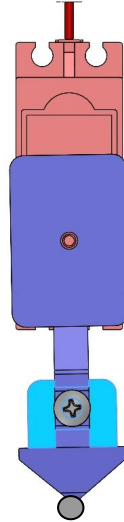
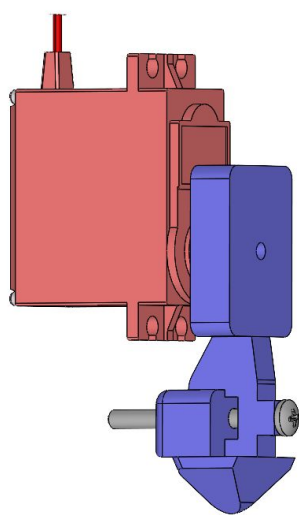
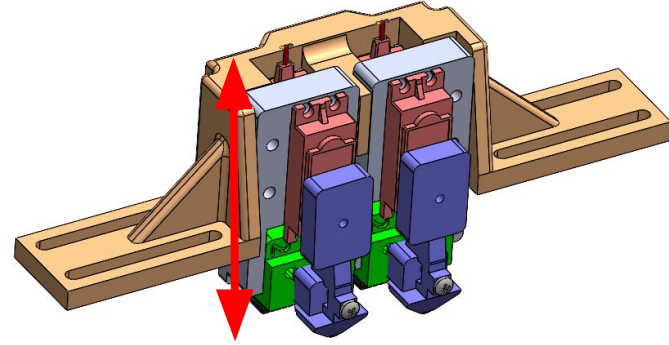
Bass Clamp

- “Clamps” around contours of bass body
- Foundation for all components
- Uses 3D scanned bass to perfectly match surface and clamp effectively
- Holes for attaching picking system
- Linkage piece attaches to 80/20 rail

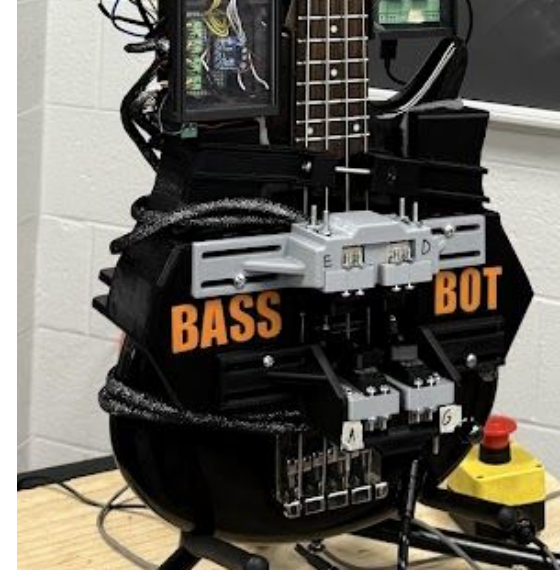


Picking System

- Servo motors pick each string (4)
- Slotted holes allows for left/right adjustability to place picks over strings
- Additional bracket (gray) is height adjustable to precisely set picking action

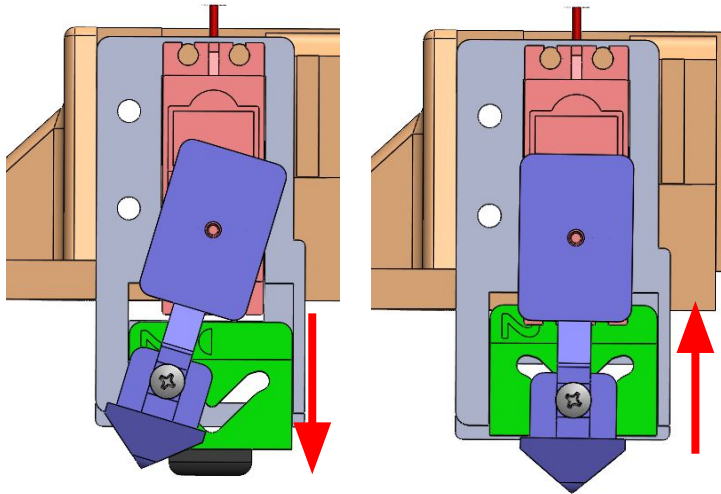


← Bass String



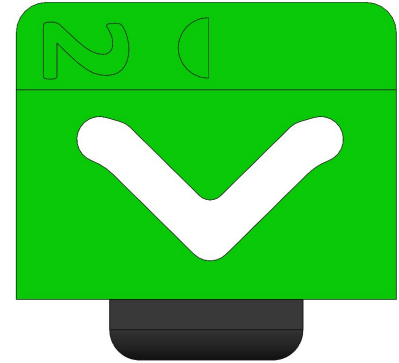
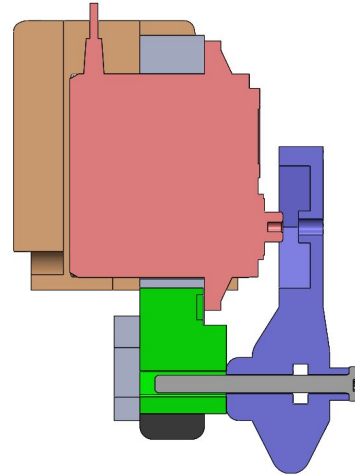
Picking cont. - Damping

- Screw in pick slides through slot to give vertical motion to “damping block”
- Foam damps strings when pick reaches maximum angle



Pick at maximum angle
= damper down

Picking string =
damper up

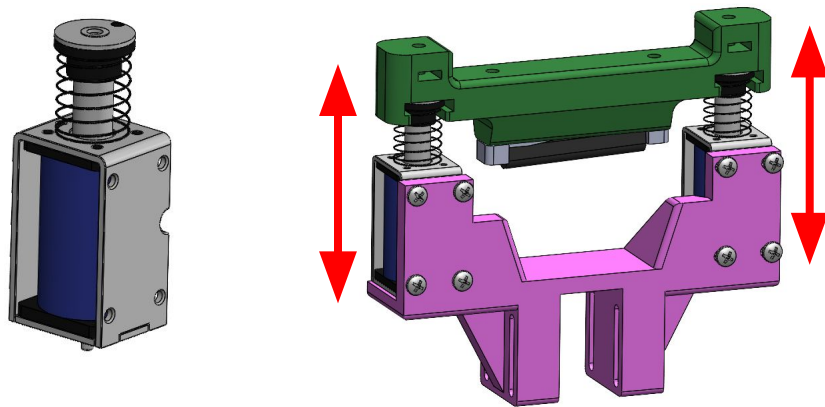


Foam damping
material



Fretting System

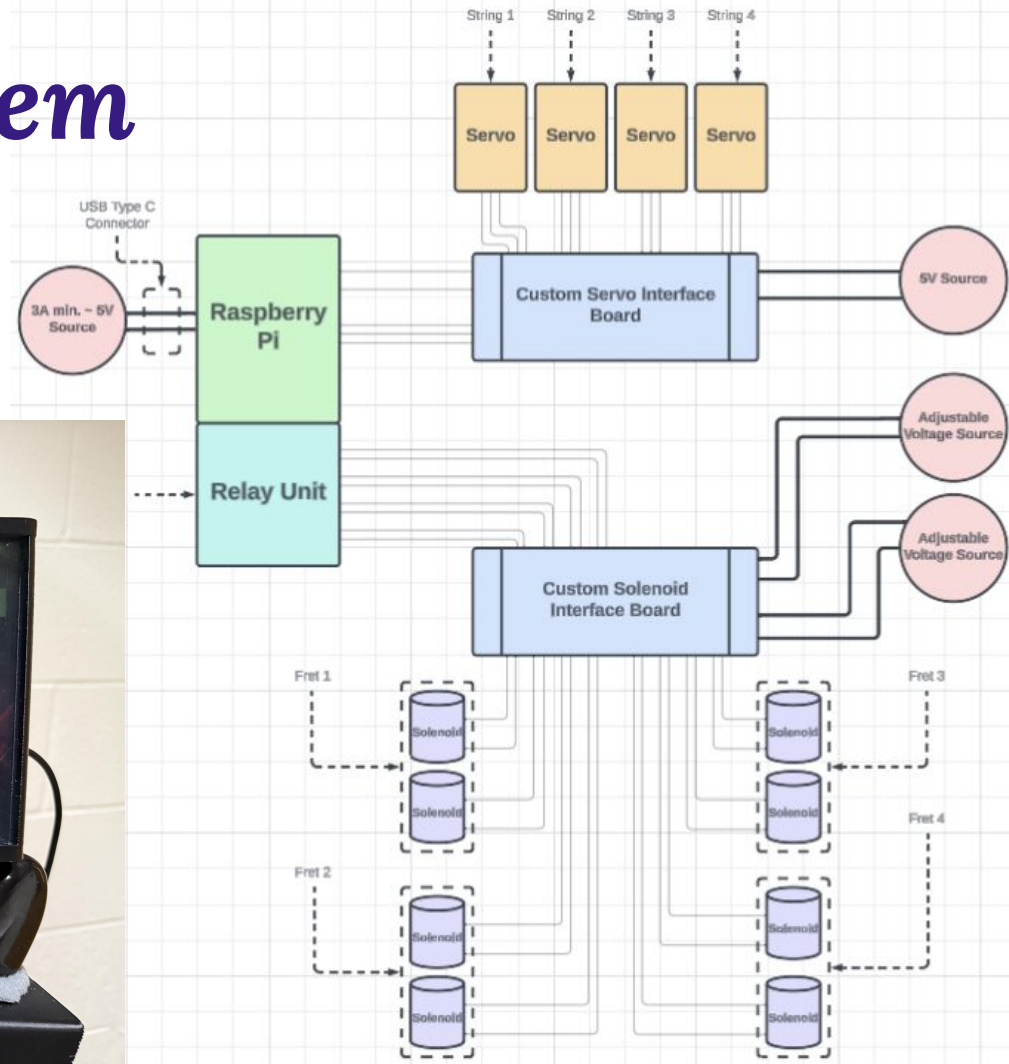
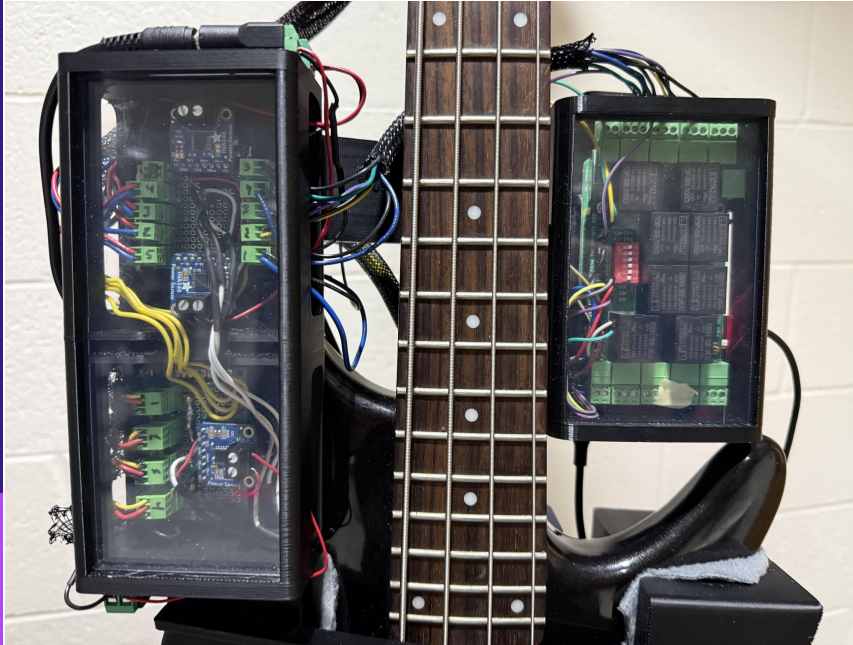
- Bracket uses two solenoids to press capo rubber across all four strings
- Four frets used to access a wide range of notes - 8 solenoids total
- Fixture attaches to 80/20 rail
- Angle of rubber precisely chosen to fret all strings effectively



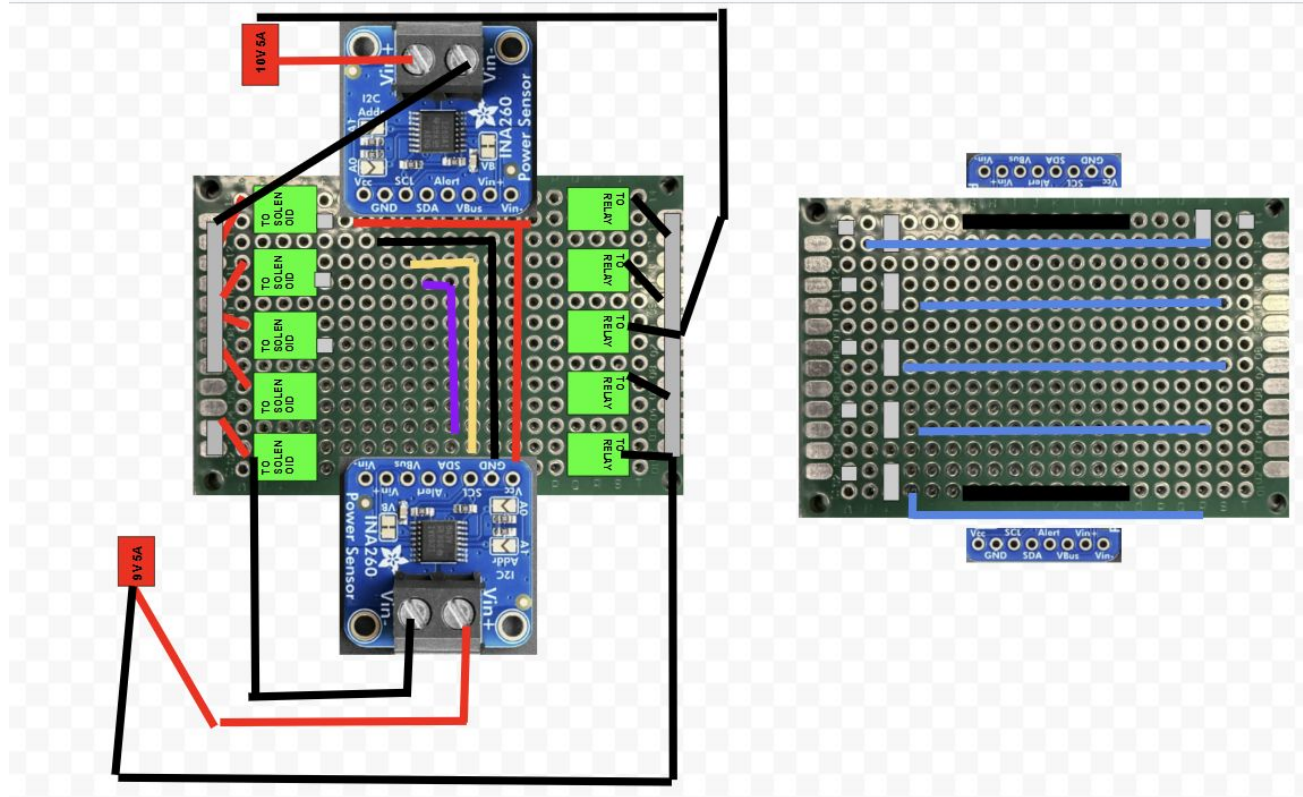
Angled fretting rubber



Electrical System



Electrical System

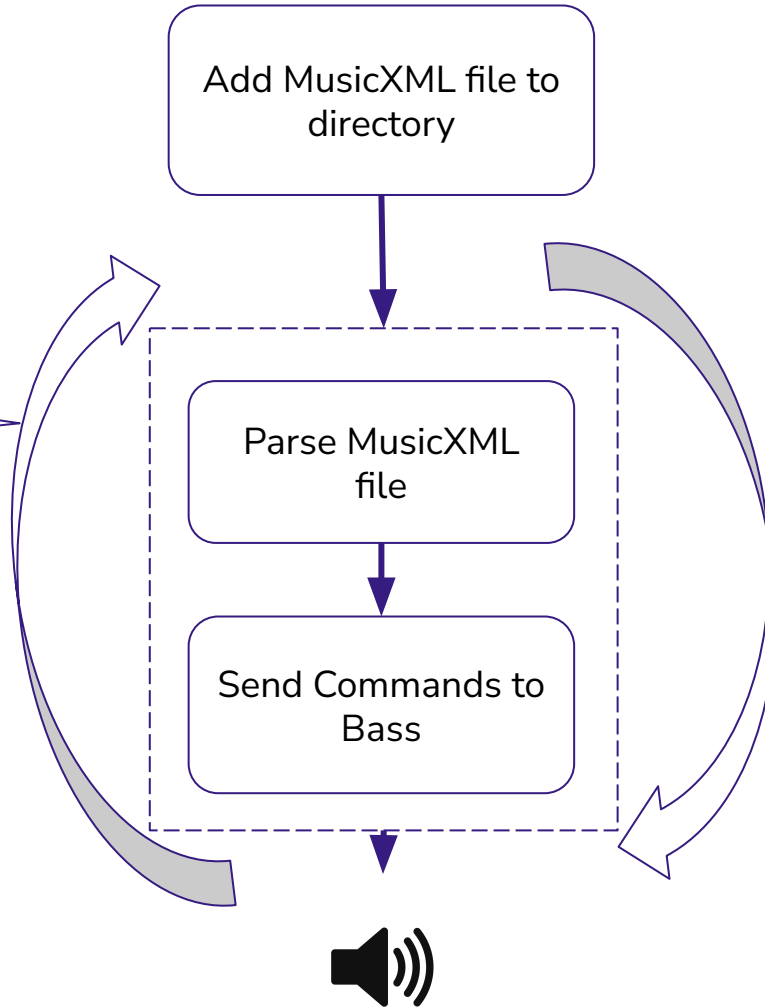


Electrical Design Adjustments



Software Flow

REPEAT
For each note in the
musicXML file



GUI



Hardware

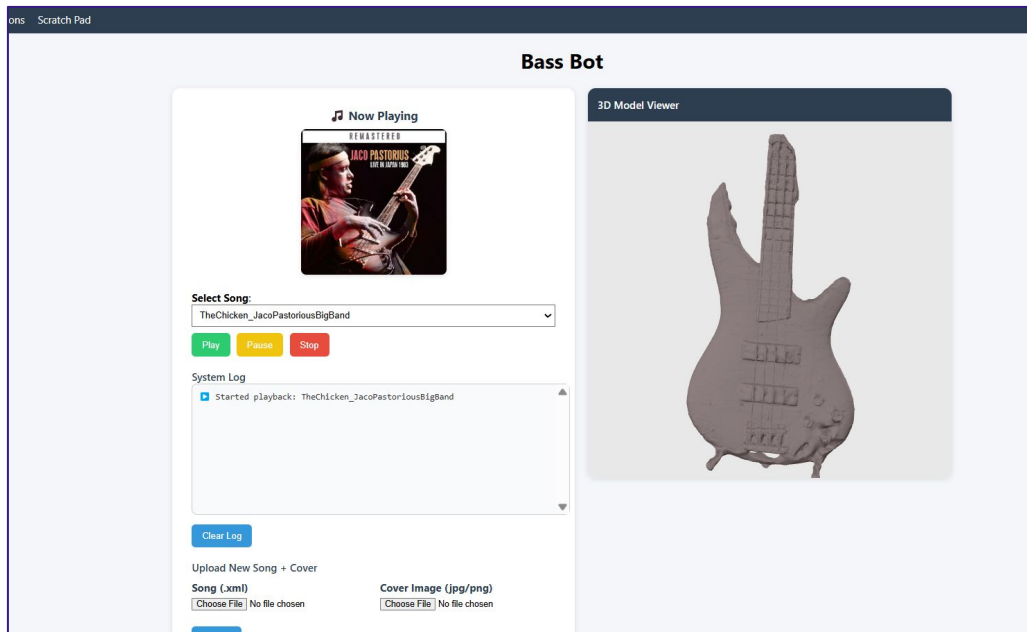


Flask

HTTP Requests



Graphics
Interface Logic



GUI

[Home](#) [Configuration](#) [Scratch Pad](#)

Configuration Settings

String	Servo GPIO	State	Sustain Factor	Low	High	Zero Offset
E	10	☐	1.3	-10	40	-15
A	18	☒	1.44	-40	30	0
D	24	☐	1.3	-30	30	15
G	27	☒	1.5	-35	30	0
Fret	Relay Number					
fret1	5					
fret2	3					
fret3	2					
fret4	4					
damper	1					

Save Settings



Scratch Pad

Friendly for Non-programmers

Google



Blockly



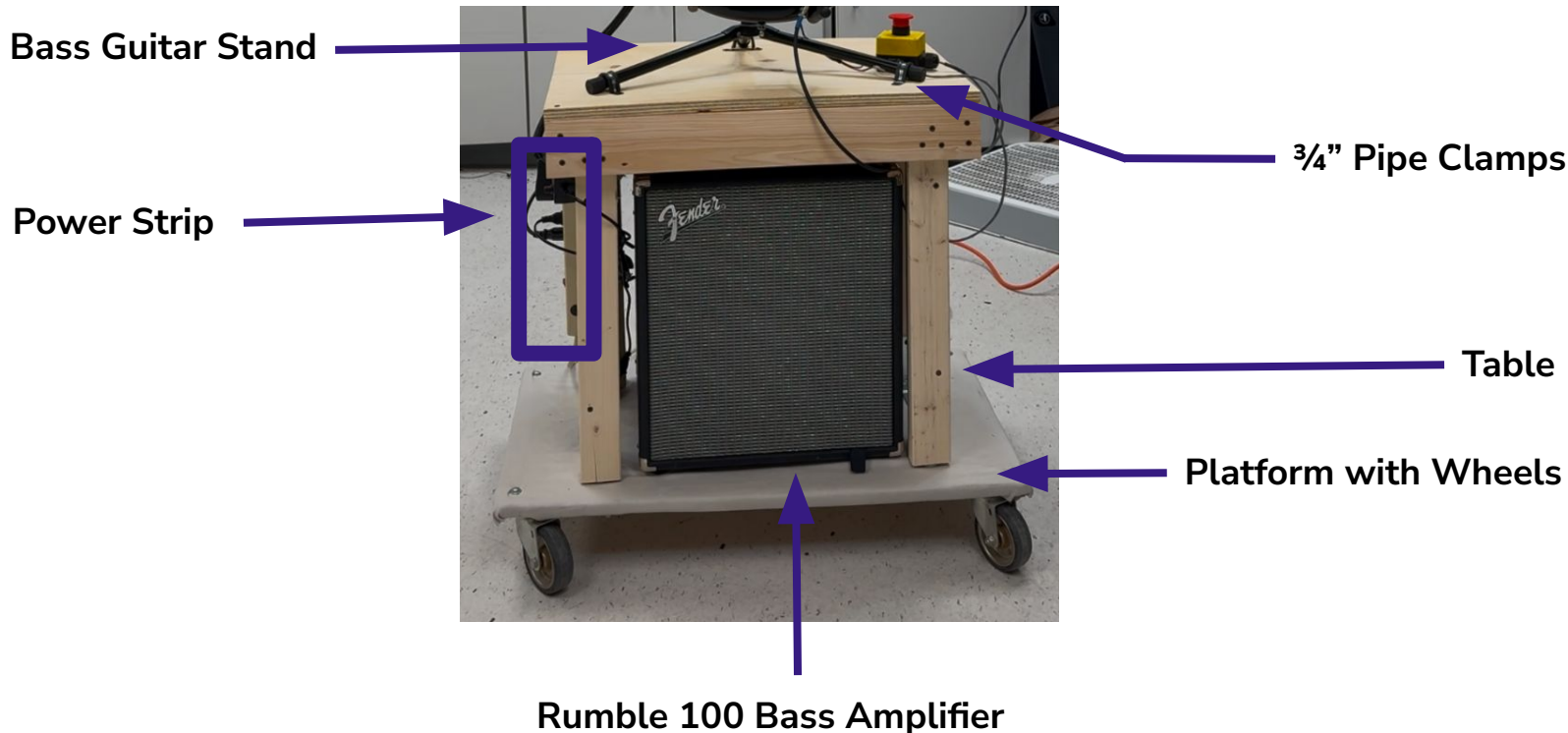
Software

The screenshot shows a GitHub repository page for 'ENG5-451---Senior-Capstone-Project'. The repository is public and has 1 branch, 0 tags, and 0 stars. The main branch is 'main'. The repository contains a file named 'andnet-deboer' with 31 commits. The file is a configuration file for a project. The repository also contains a directory named 'Adafruit_CircuitPython_INA260' and a file named 'Adafruit_INA260'. The repository is described as 'Self-playing bass guitar an exploration of robotics in live music'. The repository has 0 releases, 0 packages, and 0 languages. The repository is created by 'andnet-deboer'.

File	Commit	Time
Adafruit_CircuitPython_INA260	Relay testing	last month
Adafruit_INA260	Relay testing	last month
BassBot	Configurations	3 days ago
Songs	Relay testing	last month
Testing	Modularizing code	last month
AudioSpectrum.py	Updated audio driver index	2 months ago
AudioSpectrum2.py	Relay testing	last month
MusicXML_parser.py	Merge branch 'main' into actuatorTesting	2 months ago
README.md	Update README.md	4 months ago
servoControl.py	Relay testing	last month
servoMusic.py	Updated project	2 months ago
temp.py	Tracking existing test code in repo	5 months ago
webDisplay.py	web display and multithreading test	5 months ago



Showcase Display Set-up

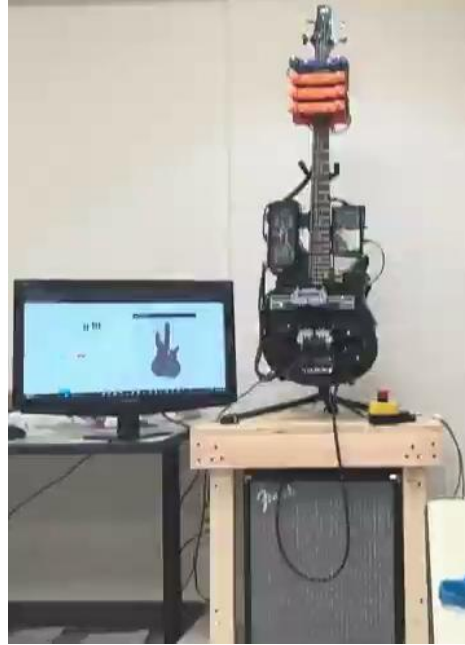


Results

- ❖ \$584.47
- ❖ > 5 Notes/sec
- ❖ Removable



Play Many Different Notes



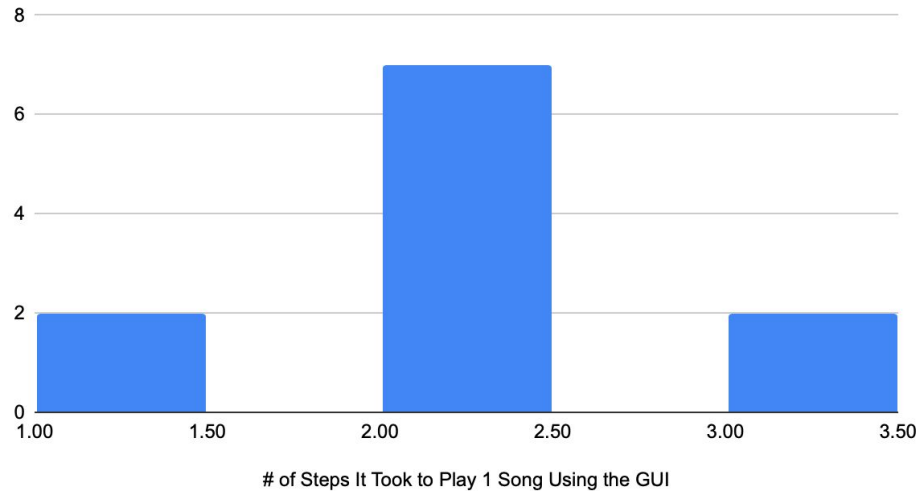
← Chromatic Scale

of Notes = 20



Testing GUI Simplicity

Histogram of Number of Steps



Average Time to
Play 1 Song =
9.43 seconds

Steps

<

5

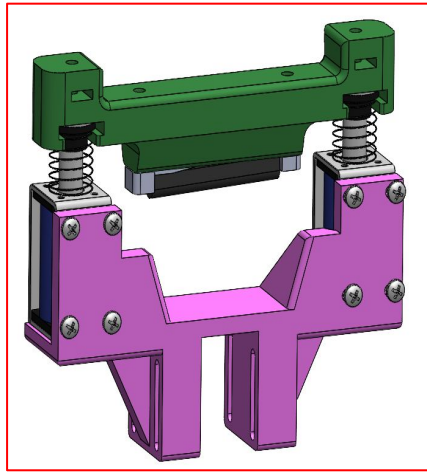
Time

<

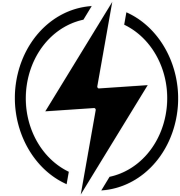
2 min.



Next Steps



Servo Improvements



+ Encoder



Acknowledgments



Dr. Cuthbert,
Python Music21 Library



Dr. Miguel Abrahantes



Dr. Roger Veldman

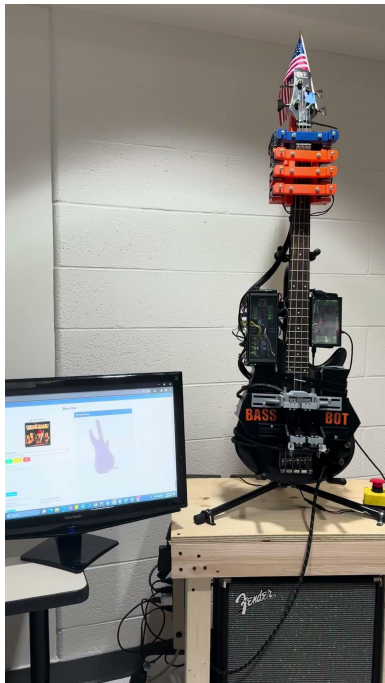


Dr. Matthew DeJongh



Hans Veldman

Thank You



Questions?