Carnegie Mellon University

Team A2: SuperFret

Owen Ball, Ashwin Godura, Tushaar Jain

Use Case

Beginners face obstacles such as:

- Building muscle memory
- Developing rhythm
- Maintaining tempo
- Memorizing melodies
- Understanding music theory

Traditional solutions, such as hiring a guitar teacher

- Are expensive
- Lack scheduling flexibility

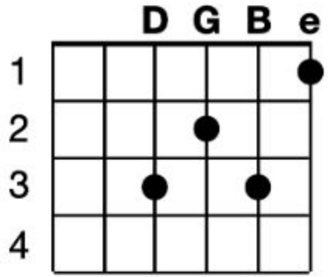


Image Source


Image Source

Carnegie Mellon University

The Solution

Meet *SuperFret*, an innovative guitar training solution featuring

- An LED fretboard to indicate finger positions
- Detection of user finger placement
- A web interface to receive feedback and select training modes

Image Source

SuperFret will allow beginners to

- Quickly master basic notes and scales
- Visualize musical patterns
- Rapidly build expertise with new melodies

Covers: Circuits, Hardware and Firmware, and Software Systems

Carnegie Mellon University

Requirements

Support music

- As fast as 100 beats per minute
- Featuring 1/8th notes and longer

Hardware

- Accommodate 14 frets with 4 strings per fret
    - Allows for the majority of bass guitar notes to be played
- 50 ms max. delay between playing note and seeing LED feedback
    - Delay between perceivable audio and visual stimulus
- Detect finger placement and strums with
    - 99% accuracy (~1 incorrect reading / minute)
    - <40ms latency (to give LEDs time to provide rapid feedback within 50ms)

Carnegie Mellon University

Requirements

Web Interface

- Intuitive UI for selecting training mode
- Includes at least one basic song and scale for training
- 250ms max. delay between playing a note and the UI reacting

Safety

- Exposed contacts do not exceed 5V making dielectric breakdown of skin impossible
- <1mA through body in worst case (wet fingers)
- No perceptible sensation from touching the strings and frets

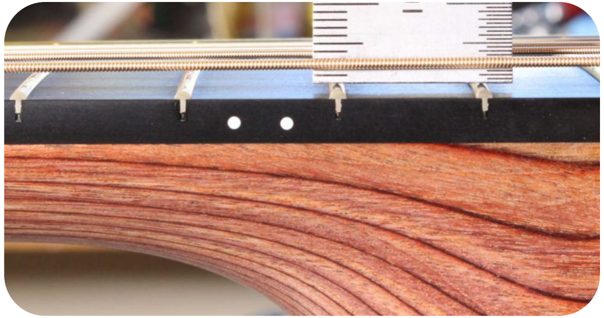
Current Levels (Milliamps)	Probable Effect on Human Body
1mA	Slight tingling sensation. (Still dangerous under some conditions.)
5mA	Slight shock felt. Disturbing but not painful. Average person can let go.
6mA – 16mA	Painful shock causing some loss of muscle control. Commonly termed “let go” range or freezing current.

Image Source

Carnegie Mellon University

Technical Challenges

Image Source



Electronics on fretboard do not interfere with playing

- Frets stand 1-2mm above fretboard
- Fretboard is cambered, making a rigid PCB difficult

Electronics should not drastically change sound or ease of use of the guitar

- Electronics should not interfere with effectiveness of resonant chamber

Detecting strumming from ambient background noise

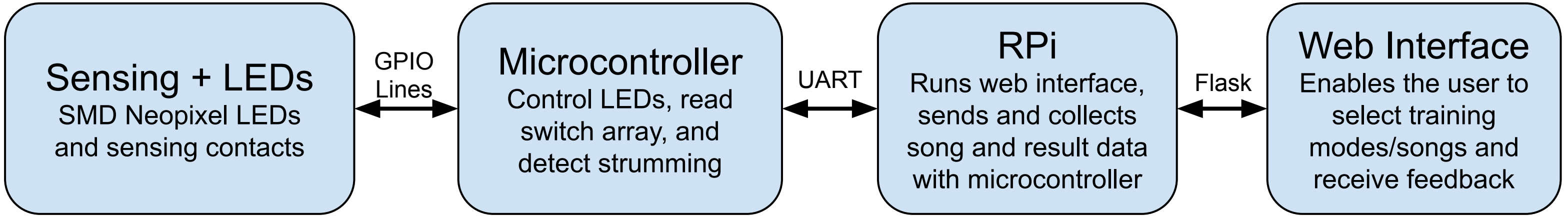
Web UI development

- None of us have experience developing an interactive web interface

Carnegie Mellon University

Solution Implementation

General Overview



Sensing + LED PCB: A series custom PCBs with low profile SMD components that will be placed into carved out channels next to each fret. Will take in a clock and data line and send outputs via the guitar strings

Microcontroller: Teensy 4.1's high clock speed enables rapid I/O with LEDs and switch array. Offloads real-time work from RPi. Will run the NeoPixel library to control LEDs at >20Hz and detect strumming using an electret microphone

Raspberry Pi: RPi to handle high-level coordination of the system and connect to web interface. Will transmit song data to the the Teensy and receive results via UART

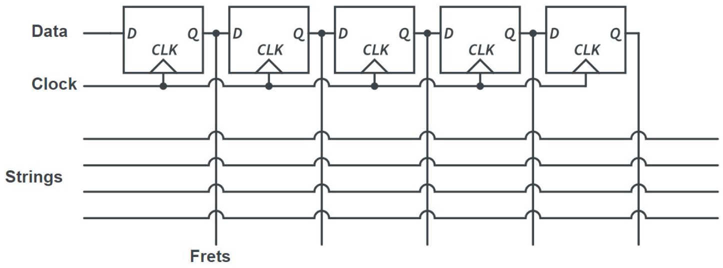
Web Interface: A GUI run using Flask to allow general users to control the actions of the RPi

Carnegie Mellon University

Solution Implementation

Finger Placement Sensing

- Use a shift register design to drive each fret high individually while the string voltages are read
  - Lowers wire count and reduces cost by allowing for tileable PCB design
- Voltage will be limited to 3.3V, which is too low to cause dielectric breakdown of the skin
  - Current will be limited to <0.1mA, although current limiting resistors can also be used



Strum Detection

- Read amplified electret microphone data using Teensy and detect spikes in sound amplitude

Carnegie Mellon University

Testing, Verification, and Metrics

Hardware Testing

- Use multimeter to verify safe current levels through body
- Strum detection tested by performing a series of 1/8th note strums and comparing number recorded to actual count
- Test LEDs and finger placement by lighting up the LEDs where the user's fingers are. Measure percentage correct.

Latency Testing

- Use oscilloscope to measure time delay between various microcontroller stimuli
  - Microphone response, LED data line, fret-string switches closing, debugging GPIOs
- Send series of web app requests and measure time for responses to return
- Full end-to-end latency testing by using high frame rate video to record delay between strum and update of web app

Testing, Verification, and Metrics

User Experience

- A user should be able to understand the web interface and begin playing notes within 5 minutes
- Users should not notice the latency of the LEDs and strum detection
- Our system should not impede fingering notes and strumming

Metrics

- Latency for strum detection, fingering detection, and LED control are hard limits
- Overall device experience and training usefulness are inherently subjective and will be broken down into various subcategories tested by volunteers.
    - Rated on a scale of 1-5 and combined to form quantitative scores for experience and usefulness

Carnegie Mellon University

Tasks and Division of Labor

Ashwin:

- Creating web interface hosted on a RPi to control device
- Develop framework for communicating song data to microcontroller

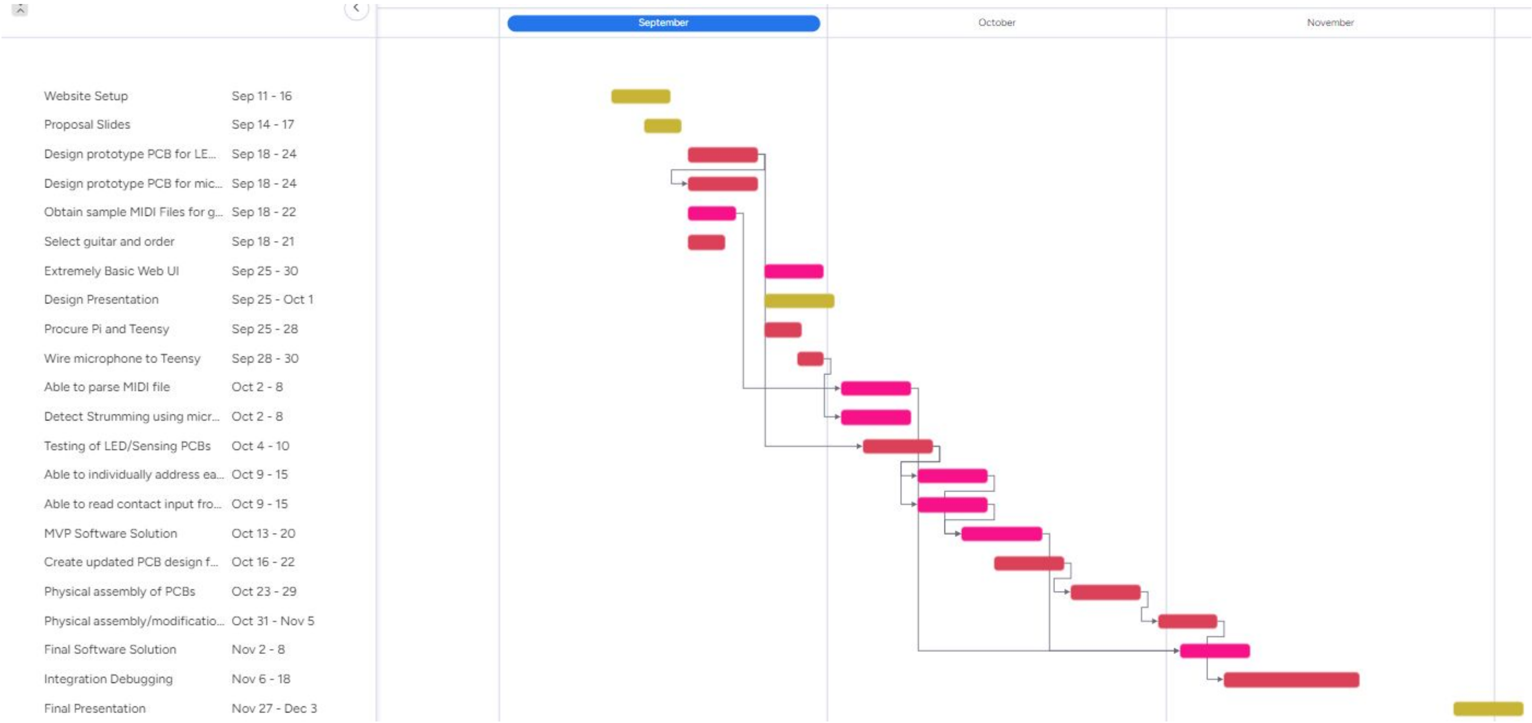
Owen:

- LED PCB design
- Power distribution and data lines for microcontroller, RPi, LEDs, finger placement sensors
- Mechanical modifications to guitar and mounting of electronics

Tushaar:

- Establish bi-directional communication system between microcontroller and RPi
- Read finger placement sensors and independently control all the LEDs
- Strum detection via processing audio data from microphone

Schedule



Carnegie Mellon University