Interactive Self-Standing Training Bag

Critical Design Review

UCF CECS, Spring 2021

ECE Senior Design Group 22

Hannah Clarke
Joseph De La Pascua
Nicole Karam Pannaci
Natesha Ramdhani
Team Members

(listed clockwise from top left)

Nicole Karam Pannaci (EE):
Sensors and Crafting Lead

Hannah Clarke (EE):
Software and Communications Lead

Joseph De La Pascua (EE):
Power and PCB Lead

Natesha Ramdhani (EE):
Design Concept and Indicator Lead
Overview

- Motivation and Goals
- Specifications, Requirements, Standards, and Constraints
- Design Concept
- Selected Components
- Software Design
- Administrative Content
  - Current Progress
  - Next Steps
Project Motivation

Train Wherever User Needs To
- Train without a partner
- Train without needing to physically be at a special location
- Especially relevant due to CoVID-19 Quarantining!

Create Workouts and Track User Progress
- Enable specific workout goals for user
- Flexibility within workout structure
- Focus on workout instead of tracking own performance

Cost of Existing Products
- No monthly cost needed to use on its own
  - Gym
  - Dojo
  - Personal Trainer
**Existing Products**

**PADIPATA**
- Hanging bag
- Covered in sensor material
- Retails for $25,000+
- New product, not many reviews
  - Recently completed Kickstarter campaign
  - Not much knowledge about actual functionality as told by real users

**FightClub**
- Standing bag
- Must be used with connected boxing gloves or hand wraps
  - Only senses punches
- Package containing gloves retails for $1,219+
- Works with app
  - Separate paid membership needed
Goals and Objectives

**Cost**
Make device cost effective to produce and affordable for the consumer.

**Interaction**
Make device interactive for the user.
- Use indicators and sensors.
- Provide feedback to user.

**Experience**
Include multiple training modes to diversify the user's workout experience.
Specifications, Requirements, Standards, and Constraints
Overall Functional Specifications

Downtime Modes
- Idle Mode
- Ambient Mode
- Off

Active Modes
- Combination Generator (Side A)
- Cardio Mode (Side B)
- Reaction Mode (Side B)
- Accuracy Mode (Side B)
Marketing Requirements for device

DURABLE
AFFORDABLE
PORTABLE
INTUITIVE
VERSATILE
LONG-LASTING
Technical Requirements for components

- **DURABLE** (can take impact of 120 hits/minute)
- **EFFICIENT** (device power < 1.1 kW)
- **COMPATIBLE** (with other device components)
- **AFFORDABLE** (within $800 budget)
- **SIMPLE TO IMPLEMENT** (within 12 weeks)
- **LONG-LASTING** (device life ≥ 5 years)

Interactive Self-Standing Training Bag | Group 22
**Constraints**

**Economic**
- Budget of $800
- Self-funded
- Running cost of electricity

**Time**
- Finished product by end of Spring 2021 semester
- Mid-Project deadlines

**Environment and Safety**
- Low-power operation design
- Keeping weight as low as possible
- Heat of electric equipment
• **Power Supply**
  - IEC 60906-2:2011
  - 3-prong ground plug (NEMA 5-15R and 5-15P)
  - Provides grounding and electrical noise immunity

• **Regulation Workout Bag Standards**
  - ASTM F2276-10(2015)
  - Age Restriction, Documentation on assembly and build, adequate warning labels, indoor setting

• **PCB Standards**
  - IPC-221B: Generic design requirements for Printed Circuit Boards and Component Mounting

• **IEEE 802.15.4**
  - Physical Layer and MAC Sublayer
  - RF Parameters
Design Concept
Overall Design

Interactive Self-Standing Training Bag | Group 22

Side A

Side B

Will be added to Side B if time permits
Base Design Concept Sketches
Physical Bag Design

Side B

Side A
Remote (UI) Design Concept Sketch

Interactive Self-Standing Training Bag | Group 22

Welcome Press Mode!

ON OFF

Mode Diff

Cancel Select
Physical Remote (UI) Design
# Device Modes

<table>
<thead>
<tr>
<th>Side Used</th>
<th>Cardio Mode</th>
<th>Reaction Time Mode</th>
<th>Accuracy Mode</th>
<th>Combination Generator Mode</th>
<th>Ambient Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Side Used</td>
<td>Side B</td>
<td>Side B</td>
<td>Side B</td>
<td>Side A</td>
<td>Both</td>
</tr>
</tbody>
</table>

## User’s Goal

<table>
<thead>
<tr>
<th>Difficulty Options</th>
<th>Cardio Mode</th>
<th>Reaction Time Mode</th>
<th>Accuracy Mode</th>
<th>Combination Generator Mode</th>
<th>Ambient Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hit target as many times as possible within specified time period</td>
<td>Hit target as close to center marking as possible</td>
<td>Hit target ASAP once indicator goes off</td>
<td>Hit designated target of grid within specified time period</td>
<td>Enjoy the LED display 😊</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stats Reported to User</th>
<th>Cardio Mode</th>
<th>Reaction Time Mode</th>
<th>Accuracy Mode</th>
<th>Combination Generator Mode</th>
<th>Ambient Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total hits</td>
<td>Total hits</td>
<td># of hits landed</td>
<td>M/L accurate hit</td>
<td>Total hits</td>
<td></td>
</tr>
<tr>
<td>Session time</td>
<td>Session time</td>
<td># of hits landed</td>
<td>L/S hit time</td>
<td>Total possible hits</td>
<td></td>
</tr>
<tr>
<td>Avg. time per hit</td>
<td>Avg. time per hit</td>
<td># of hits landed</td>
<td>Avg. time per hit</td>
<td>Hit success ratio</td>
<td></td>
</tr>
<tr>
<td>Hits per second</td>
<td>Hits per second</td>
<td># of hits landed</td>
<td>Avg. time per hit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Components and Hardware Implementation
Sensor Specifications

Sensors must be able to:

1. Withstand the force of many punches and kicks.
2. Detect rapid hits in multiple locations.
3. Count sequential hits in the same area.
4. Detect location of hits relative to designated target.
5. Consume minimal amount of power.
6. Cover entire sensing area with no more than four units.
Textile Pressure Sensor Materials

- **Velostat**: Pressure-Sensitive Conductive Sheet (Adafruit | 4.95$)
  - Sheet Dimensions: 11" x 11" (28cm x 28cm) before shaping
  - Surface Resistivity: < 31,000 ohms/sq.cm, changes with pressure
  - Temperature Limits: -45°C to 65°C

- **Shapeable Foam Sheet** (Joann’s | 12.99$)

- **Conductive Thread** (Stainless Steel) (Adafruit | 6.95$)

- **Conductive Fabric** (Adafruit | 4.95$)

- **Standard Needle and Thread**

- **Paper Clips**

- **Soldering Materials**

- **Stencils for Sensor Shapes**
Textile Pressure Sensor Construction

Interactive Self-Standing Training Bag | Group 22
Gathering and Using Sensor Data

- Resting vs Being Pressed
  - 5 quick hits
- Measuring 220Ω resistor in series with sensor
- Threshold is generally about 0.65 V
Requirements for Indicators:
• Multicolor Programmable Units
• Cut and Connect in Various Shapes
• Durable and Water Resistant
• Cost-Effective

WS2811 Pixel Lights:
• 50 LEDs, 3.5m / unit
• IP68 Waterproof Rated
  • suitable for outside use
• Seller: WESIRI (via Amazon)
• Price: $18.99 / unit

Prominent LED Testing Considerations:
• Power LEDs sufficiently so they will be stable and respond accurately
• Avoid interference with LED data signal wire
Indicator: Testing

- Power Supply (Voltage Testing)
- Current Testing
- Data Lines
- Simple Code
  - FastLED Library functions
  - RGB Testing
Power Generation

• **Base System (Amazon| 32.88$)**
  - 120 V 60 Hz AC / 12 V DC LED driver transformer.
  - 12V DC, 8.3 A Output Signal

• **UI Handheld**
  - 6V battery holder from LampVPath. (Amazon| 7.99$)
  - Holds 4 AA batteries
  - Generates a 6V DC signal.
Power Conversion

• **AC/DC**
  • Included with the base LED power supply

• **DC/DC: Breadboard Power Supply**
  • Provides regulated 5V and 3.3V for all the PCB components
  • Originally designed the PCB with onboard buck DC/DC converters (TPS560430)
    • Learned too late that these provided insufficient current to the MCU and other components
    • Bucks were replaced with the breadboard power supply connected directly to the MCU in order to use the MCU design.
UI Components

1) LCD

- HD44780: 20x04 display
- PCF8574T: 8-bit I/O expander for I2C bus
- 2 wire I2C serial communication
- 1 hot and 1 ground
- Operates at 5V
- Small enough to fit on remote
- Large enough to fit user results on one screen
- Price: $12.15 (Amazon)
UI Components

2) XBee

- XBee S2 - chip or wire antenna
- XBee USB adapters: required for configuration
- Price: 26.95$ each Antenna (Amazon), and 7.99$ each adapter (Digikey)

- **Pros**
  - Low power consumption (2mW while transmitting)
  - Plenty of range (indoor at 40m)
  - Sufficient data rate to keep low power (40kbps)

- **Cons**
  - Requires manual configuration
  - Not compatible with previous generations of XBee chips
UI Components

3) User Input

• **Push Buttons:** push to close circuit, release to open circuit, waterproof, large enough for finger, small enough to fit 4 on the remote

• **On/Off slide switch:** provides stability to remain in the on position while user is interacting with base (remote will experience movement)

4) Power Supply

• **4xAA batteries:** provides 6V with 9600mAh, enough to withstand larger current drawn from LCD backlight; provides enough voltage for LCD, processor chip, and XBee antenna
MCU Selection

Standards for MCU Selection

• Have sufficient input lines for all sensors and communication necessary.
• Have sufficient output lines for all indicators and communication necessary.
• Communicate with the UI system to provide raw data.
• Require internal ROM and RAM.
• If possible, use MCU for both base and UI

Main Contenders

• FPGA
• TI MCU
• AVR MCU
• DSP MCU
• (Raspberry Pi)
MCU Selection Top Two

TI MCU (MSP430F6459):
- RAM: 66KB
- Pins: 74
- I2C: 3
- UART: 3
- SPI: 6

AVR MCU (ATmega2560):
- RAM: 64KB
- Pins: 100
- I2C: 1
- UART: 4
- SPI: 5
- Timers: 6
- Input voltage: 1.8V to 5.5V

Supports function libraries that facilitate the software coding!
Hardware Connection Block Diagram

**Base Device**
Body of physical training bag and its attachments

**Side A: Grid**
- **Output:** LED Indicators for Sensors
- **Input:** 4 Sensor Grid with opponent graphic overlay

**Side B: Target**
- **Output:** LED Indicators for Sensors
- **Input:** Multi-zone Sensor Target to detect hit location

**User Interface**
Takes prompts from user, relays information to user

- **Power Supply**
- **Input:** Switch and Buttons
- **Output:** LCD Screen

**MCU**
- **XBees**

Interactive Self-Standing Training Bag | Group 22
Base PCB
Version 1

Interactive Self-Standing Training Bag | Group 22
Base PCB
Version 2
UI Remote PCB
Version 2
Bag PCB Rework

Interactive Self-Standing Training Bag | Group 22
PCB Changes

Bag PCB
- Changed surface mount resistors to through hole
- Added an ISP header to the board
- Added labels to all pins
- Added 16 MHz oscillator and associated filter capacitors
- Added one more buck converter (ultimately unused)

UI PCB
- Changed surface mount resistors to through hole
- Added an ISP header to the board
- Added labels to all pins
- Added 16 MHz oscillator and associated filter capacitors
- Added one more buck converter (ultimately unused)
Software Design
Interconnected web of interrupt service routines to handle actions.
Software Design Approach

• User Interface System
  • Heavily dependent on user input through buttons
  • Relies on button ISR’s to handle actions:
    • Enable/Disable specific button interrupts
    • Reset Timer values
    • Display instructions to User
    • Display Mode and Difficulty selection changes
    • Display correct screen when Cancel button is pressed
    • Trigger UART TX interrupt
  • Relies on UART RX ISR to display results at the end of the session
  • Relies on Timer ISR to prevent getting “stuck”
Software Design Approach

• **Base System**
  - Heavily dependent on pressure sensor ISRs and timer ISRs to handle actions:
    - Keeping score
    - Continue through the session, even without user interaction
    - Enable appropriate indicators for the mode of the session
    - End the session
  - Relies on UART TX ISR to send data back to the UI system
Software Testing

• External Devices
  • Individual code is prepared to test and
  • Integrate each external device.
    • LCD
    • XBee
    • LEDs
    • Sensors

• Libraries for devices
  • `#include <Wire.h>`
  • `#include <LiquidCrystal_I2C.h>`
  • `#include <SoftwareSerial.h>`
  • `#include <FastLED.h>`
UI System Software Flowchart

Button/Switch Names:
ON/OFF = On/Off (Toggle Switch)
MODE BTN = Mode
DIFF BTN = Difficulty
SEL BTN = Select
CXL BTN = Cancel

User Begins Training: next interactions will be controlled by Base System

Enter Base Active Mode
- Enter LPM
- Send Mode
- Data to Base
- Disable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN
- Disable MODE BTN
- Disable ON/OFF BTN

Enter Welcome
- Display Welcome Message
- Start Fail Safe Timer 0
- Enable MODE BTN
- Enable ON/OFF
- Enable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN

Mode Select
- Display Mode Select Message
- Start Fail Safe Timer 0
- Enable SEL BTN
- Enable CXL BTN
- Increment Select Count

Difficulty Select and Confirm
- Display Difficult Mode Select and Confirm Message
- Display Difficult Confirm Message
- Cycle through options

System On
System Off
**Button/Switch Names:**
- ON/OFF = On/Off (Toggle Switch)
- MODE BTN = Mode
- DIFF BTN = Difficulty
- SEL BTN = Select
- CXL BTN = Cancel

**Enter Base Active Mode**
- Enter LPM
- Send Mode Data to Base
- Disable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN
- Disable MODE BTN
- Disable ON/OFF BTN

**User Begins Training:**
Interactions will be controlled by Base System

**System On**
- Welcome
  - Display Welcome Message
  - Start Failsafe Timer 0
  - Enable MODE BTN
  - Enable ON/OFF
- Enable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN

**Mode Select**
- Display Mode Select Message
- Start Failsafe Timer 0
- Enable SEL BTN
- Enable CXL BTN

**Cycle through options**
- Has timer run out?
- Which button pressed?

**System Off**
- Return to Welcome

**Difficulty Select and Confirm**
- Which mode selected?
- Display Difficulty Select and Confirm Message
- Display Mode Confirm Message
- Enable DIFF BTN
- Increment Select Count
- Start Failsafe Timer 0
- Enable DIFF BTN
- Increment Select Count

**Which button pressed?**
- Yes
  - Cycle through options
- No
  - MODE
- OFF
- SELECT
- CXL
- OFF
- SELECT
- CXL
- OFF
- SELECT
- CXL
- OFF
- SELECT
- CXL

**Has timer run out?**
- Yes
  - Cycle through options
- No
  - Which button pressed?

**Interactive Self-Standing Training Bag | Group 22**
UI System Software Flowchart

Button/Switch Names:
ON/OFF = On/Off (Toggle Switch)
MODE BTN = Mode
DIFF BTN = Difficulty
SEL BTN = Select
CXL BTN = Cancel

User Begins Training: next interactions will be controlled by Base System

Enter Base Active Mode
- Enter LPM
- Disable Timer 0
- Send Mode Data to Base
- Disable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN
- Disable MODE BTN
- Disable ON/OFF BTN

System On

Welcome
- Display Welcome Message
- Start Failsafe Timer 0
- Enable MODE BTN
- Enable ON/OFF BTN
- Disable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN

Mode Select
- Display Mode Select Message
- Start Failsafe Timer 0
- Enable SEL BTN
- Enable CXL BTN
- Increment Select Count

Difficulty Select and Confirm
- Display Mode Confirm Message
- Display Difficulty Select and Confirm Message

Which mode selected?
- Cycle through options
- Cycle through options 1-3
- Has timer run out?
- OFF
- MODE
- Which button pressed?
- Yes
- OFF
- System Off
- NO
- Return to Welcome
- CXL
- SEL
- DIFF
- OFF
- Which button pressed?
- Yes
- Select count = 2?
- Yes
- Return to Welcome
- No
- Which button pressed?
- Yes
- OFF
- MODE
- Which button pressed?
- No
- Has timer run out?
- OFF
- MODE
UI System Software Flowchart

Button/Switch Names:
ON/OFF = On/Off (Toggle Switch)
MODE BTN = Mode
DIFF BTN = Difficulty
SEL BTN = Select
CXL BTN = Cancel

User Begins Training: next interactions will be controlled by Base System

Enter Base Active Mode
- Enter LPM
- Disable Timer 0

Send Mode
- Data to Base
- Disable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN
- Disable MODE BTN
- Disable ON/OFF BTN

Enter Active Mode

System On

Welcome
- Display Welcome Message
- Start Failsafe Timer 0
- Enable MODE BTN
- Enable ON/OFF
- Disable SEL BTN
- Disable CXL BTN
- Disable DIFF BTN

Mode Select
- Display Mode Select Message
- Start Failsafe Timer 0
- Enable SEL BTN
- Enable CXL BTN
- Increment Select Count

Difficulty Select and Confirm
- Disable MODE BTN
- Start Failsafe Timer 0
- Enable DIFF BTN
- Increment Select Count

Which mode selected?
- Display Mode Confirm Message
- Display Difficulty Select and Confirm Message

Has timer run out?
- No
- Which button pressed?
  - Yes
  - OFF
- System Off
  - OFF
  - SEL
  - CXL
  - Return to Welcome

Which button pressed?
- Yes
  - Cycle through options
- No
  - Cycle through options 1-3
  - System Off
  - OFF
  - SEL
  - CXL
  - Return to Welcome

Select count = 2?
- Yes
- Return to Welcome
- No
Base System Software Flowchart

Initialization
- Initialize Communication
- Initialize Sensors
- Initialize buttons
- Initialize LEDs

Welcome
- Disable Start BTN
- Initialize results registers A-F to 0
- Enable Early Exit BTN
- Enter LPM (wait for UI signal)
- Wait here until UI data signal received

Active Mode
- Enable and Initialize Mode Timer 1
- Disable Start BTN
- Execute Mode

NOTE: Specifics for mode detailed in individual flow charts

T1 Expires
- Which signal received?
- Early Exit

Result Computation
- Send Results Data to UI (to display for the user)
- Disable Early Exit
- Disable all LEDs

NOTE: Specifics for mode detailed in individual flow charts

Prepare for Active Mode
- Exit LPM
- Store MODE
- Store DIFFICULTY
- Enable Start BTN
- Disable UI Timer 0
- Enable LED Ready to Start sequence
- Wait here until Start BTN pushed

Start
- Early Exit or T0 Expires
- None (Yet)
Base System Software Flowchart

**Initialization**
- Initialize Communication
- Initialize Sensors
- Initialize buttons
- Initialize LEDs

**Active Mode**
- **Mode Sequence**
  - Enable and Initialize Mode Timer 1
  - Disable Start BTN
  - Execute Mode

  **NOTE:** Specifics for mode detailed in individual flow charts

- **Result Computation**
  - Send Results Data to UI (to display for the user)
  - Disable Early Exit
  - Disable all LEDs

  **NOTE:** Specifics for mode detailed in individual flow charts

- **Prepare for Active Mode**
  - Exit LPM
  - Store MODE
  - Store DIFFICULTY
  - Exit Start BTN
  - Disable UI Timer 0
  - Enable LED Ready to Start sequence

  **Which signal received?**
  - Early Exit
  - Start

- **Welcome**
  - Disable Start BTN
  - Initialize results registers A-F to 0
  - Enable Early Exit BTN
  - Enter LPM (wait for UI signal)
  - Wait here until UI data signal received

  **Which signal received?**
  - Early Exit
  - None (Yet)
Base System Software Flowchart

Active Mode

Mode Sequence
- Enable and Initialize Mode Timer 1
- Disable BTN
- Execute Mode

NOTE: Specifics for mode detailed in individual flow charts

Which signal received?
- T1 Expires
- Early Exit

Result Computation
- Send Results Data to UI (to display for the user)
- Disable Early Exit
- Disable all LEDs

NOTE: Specifics for mode detailed in individual flow charts

Prepare for Active Mode
- Exit LPM
- Store MODE
- Store DIFFICULTY

Which signal received?
- Early Exit or T0 Expires
- None (Yet)

Prepare for Active Mode
- Enable Start BTN
- Disable UI Timer 0
- Enable LED Ready to Start sequence

Wait here until Start BTN pushed

Welcome
- Disable Start BTN
- Initialize results registers A-F to 0
- Enable Early Exit BTN
- Enter LPM (wait for UI signal)

Which signal received?
- None (Yet)
- Early Exit

Wait here until UI data signal received
Base System Software Flowchart

**Initialization**
- Initialize Communication
- Initialize Sensors
- Initialize buttons
- Initialize LEDs

**Welcome**
- Disable Start BTN
- Initialize results registers A-F to 0
- Enable Early Exit BTN
- Enter LPM (wait for UI signal)
- Wait here until UI data signal received

**Active Mode**

**Mode Sequence**
- Enable and Initialize Mode Timer 1
- Disable Start BTN
- Execute Mode

**Result Computation**
- Send Results Data to UI (to display for the user)
- Disable Early Exit
- Disable all LEDs

**Prepare for Active Mode**
- Exit LPM
- Store MODE
- Store DIFFICULTY
- Enable Start BTN
- Disable UI Timer 0
- Enable LED Ready to Start sequence
- Wait here until Start BTN pushed

**Note:** Specifics for mode detailed in individual flow charts.

Interactive Self-Standing Training Bag | Group 22
Administrative Content
## Budget and Financing

<table>
<thead>
<tr>
<th>Component</th>
<th>Price (USD)</th>
<th>Quantity</th>
<th>Total (USD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conductive Thread</td>
<td>$5.99</td>
<td>2</td>
<td>$11.98</td>
</tr>
<tr>
<td>Velostat/Linqstat</td>
<td>$4.95</td>
<td>5</td>
<td>$24.75</td>
</tr>
<tr>
<td>Conductive Fabric</td>
<td>$4.95</td>
<td>2</td>
<td>$9.90</td>
</tr>
<tr>
<td>Foam Sheet</td>
<td>$12.99</td>
<td>2</td>
<td>$25.98</td>
</tr>
<tr>
<td>LCD Screen</td>
<td>$12.25</td>
<td>1</td>
<td>$12.25</td>
</tr>
<tr>
<td>LED Strips</td>
<td>$25.99</td>
<td>2</td>
<td>$51.98</td>
</tr>
<tr>
<td>LED Pixel Lights</td>
<td>$18.99</td>
<td>2</td>
<td>$37.98</td>
</tr>
<tr>
<td>Arduino Uno</td>
<td>$23.00</td>
<td>2</td>
<td>$23.00</td>
</tr>
<tr>
<td>Training Bag</td>
<td>$120.00</td>
<td>1</td>
<td>$120.00</td>
</tr>
<tr>
<td>Xbee Antenna</td>
<td>$26.00</td>
<td>2</td>
<td>$52.00</td>
</tr>
<tr>
<td>Xbee Explorers</td>
<td>$12.00</td>
<td>2</td>
<td>$50.00</td>
</tr>
<tr>
<td><strong>Total Cost</strong></td>
<td></td>
<td></td>
<td><strong>$736</strong></td>
</tr>
</tbody>
</table>

**Expected Cost** ≈ $800 to $900
# Work Distribution

**GREEN (1): Lead**

**BLUE (2): Primary Assistant(s)**

<table>
<thead>
<tr>
<th>Task</th>
<th>Hannah</th>
<th>Joseph</th>
<th>Nicole</th>
<th>Natesha</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Sensors</td>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>UI</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Power</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Indicators</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Software</td>
<td>1</td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>MCU</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCB</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sewing and Crafting</td>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Video Editing</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Questions?
Group 22 appreciates your time!