Interactive Self-Standing Training Bag

University of Central Florida

College of Engineering and Computer Science

Department of Electrical and Computer Engineering

Dr. Samuel Richie

EEL4914: Senior Design I

Initial Project and Group Identification Document

(Divide and Conquer 2.0)

Group 22

Hannah Clarke: Electrical Engineering

Joseph De La Pascua: Electrical Engineering

Nicole Karam Pannaci: Electrical Engineering

Natesha Ramdhani: Electrical Engineering

Table of Contents

Introduction: Project Statement/Narrative

For our Senior Design project, we are creating an interactive boxing/martial arts stand up workout bag, where the athlete can choose different types of workout patterns, and receive results to view progress. Our motivation behind this project stems from the idea of creating an easy yet engaging way for a trainee of any level, to exercise, train, and track their progress in various perspectives of fight training without the need for a training partner. The Coronavirus pandemic has made this even more relevant since most training locations closed for an extended time, leaving many people without a method or physical partner to train at home with.

Another aspect of our project is that our design will be at a lower cost to those on the market, which can do some, but not all, of the actions as our interactive design. Currently on the market, interactive standup punching bags usually sell for about $25,000. Because our design can provide feedback/results after a session as well as provide interesting/fun workout patterns, the user can practice their skills in an engaging manner without paying the $40 to $70 average hourly cost of a personal trainer. Additionally, our design can be used at home which saves the user the monthly cost of a gym membership necessary for the correct equipment: a cost that ranges from $10 to $100 per month.

In order to capture as many different aspects of training as possible, the bag will have a dual-sided design that accommodates various modes that the user can train with. “Side A” will have various zones with binary touch sensors that will detect whether a certain area was hit. These zones will be loosely based on different practical locations that a fighter would be making contact with in a boxing, kickboxing, or martial arts environment against a real opponent, so that the user can have the most immersive and accurate training possible. On “Side A”, the device should be able to generate random sequences of techniques landing in various areas; this can be extended to using multi-color indicators to signal which side or technique type should be used to attack the opponent. “Side B” will contain a target zone with a sensor/cluster of multiple sensors arranged within this target so that more focused training can also take place and create a complete, rounded training session. These sensors will gather data on the training session and report it back to the user so that the user can quantify their performance in different types of workouts and track their progress. Some modes that can be implemented by “Side B” include a cardio mode where users try to make as many hits as possible within a time limit, a reaction mode where users aim to land their hits based on a stimulus by the indicator, and an accuracy mode where the user practices hitting a specific area as precisely as possible.

A regular standup punching bag alone does not have the capability to improve a user’s performance, since it does not provide feedback or results without a boxing/martial arts trainer. With an interactive design, the user would be able to improve performance in precision, reaction time and speed. This design should be easy to use for first time athletes as well as experienced athletes because the user is able to select different mode levels that are most suitable for them, and increment their levels as needed to continue to receive better training.

Project Specifications

General Specifications

This device should:

- Be able to implement downtime modes
  - Idle Mode
  - Display Mode
  - Off Mode
- Be able to implement exercise modes
  - Combination Generator (Side A)
    - Uses set zones on opponent
    - Binary sensors count correct hits and save data for user
    - Practice training against an "opponent" to improve the user's skill
  - Cardio Mode (Side B)
    - Uses sensor target to count hits
    - User goal: make as many hits as possible in a certain time period
    - more hits = higher score
  - Reaction Mode (Side B)
    - Uses sensor to detect time for user to hit target after illuminating indicator
    - User goal: hit target as soon as possible after it lights up
  - Accuracy Mode (Side B)
    - Uses sensor to detect how close to center mark target is hit
    - User goal: hit center as accurately as possible to increase score

Marketing Requirements

- **Durability:** Device will be able to withstand impact from regular use. (+)
- **Cost:** Device must be cost efficient for a typical consumer; priced reasonably compared to other multimode home-gym fitness devices. (-)
- **Portability:** Device must be able to be moved if necessary without large risk of damage, and must be able to be used in a variety of different environments. (+)
- **Intuitive:** The user interface will be easy to understand for and welcoming to experts and enthusiasts alike. (+)
- **Versatile:** The device will have multiple modes of operation utilizing both sides and orientation of sensors for a well-rounded user experience. (+)
- **Device life:** The device should last as long as possible; at least 5 years for the user. (+)

Technical Requirements

- **Durability:** Components will be able to stand up to impact at a rate of over 120 hits/minute. (+)
- **Power:** Components will be power efficient; no more than 1.1kW. (-)
- **Compatibility:** Components will integrate well with other components used in the device. (+)
- **Component Costs:** Total cost of type of components of good quality will be within our budget, but as low as possible. (-)

- **Implementation Time:** The time taken for the design and implementation of modes using the sensors should not be excessive. Our goal is 12 weeks or less to build and develop. (-)
- **Longevity:** Components will not die out quickly, providing approximately 5 of years of use. (+)

House of Quality

Double Symbol = **Strong** Correlation | Single Symbol = **Weak** Correlation

Technical Requirements vs Technical Requirements

	Durability (+)	Power (-)	Compatibility (+)	Component Cost (-)	Implementation Time (-)	Longevity (+)
Implementation Time (-)						
Component Cost (-)						↓
Compatibility (+)				↑	↓	
Power (-)						↑↑
Durability (+)				↓		↑↑

Technical Requirements (Green) vs Marketing Requirements (Blue)

	Durability (+)	Power (-)	Compatibility (+)	Component Cost (-)	Implementation Time (-)	Longevity (+)
Durability (+)	↑↑			↓		↑
Cost (-)	↓		↓	↑↑		↓
Portability (+)	↑			↓		↑
Easy to Use (+)			↑		↓	
Versatile (+)		↓	↓	↓	↓	↑
Device Life (+)	↑↑			↑		↑↑

Table 1: House of Quality Specification Matrix

Component Specifications

Sensors (Nicole)

Technical Standards

Sensors used in this device should:

1.1	Be durable enough to withstand the force of a punch/kick.
1.2a	Be able to detect distinct, rapid, successive hits in different locations (Side A).
1.2b	Be able to detect distinct, rapid, successive hits in the same location (Side B).
1.3	Be able to detect the location of a hit relative to a target point (Side B).
1.4	Consume a low amount of power.
1.5	Cover a zone with no more than four individual sensor units.

Table 2: Technical Standards for Sensors

Design Matrix and Analysis

Standard:	1.1	1.2a	1.2b	1.3	1.4	1.5	Total
Accelerometers	2	5	4	1	5	5	22
IMU	1	4	5	5	4	4	23
MIMU	3	3	3	4	1	1	15
Strain Gauge	5	1	2	3	2	2	15
Rosette Strain Gauge	4	2	1	2	3	3	15

Table 3: Design Matrix for Sensors

In this case, we will be discussing accelerometers and IMU since these had high ranking among all the sensors. In **1.1**, these have low ranking since without any protective layer, these sensors will not survive any damages done by any hits/kicks by the user. In **1.2b**, IMU is ranked higher since it can differentiate each force of each hit/kick with the direction of the impact, and Accelerometers only will measure the force of the hit/kick of the impact. In **1.2a** if it is just hitting in one area, accelerometers will do the job fine since it only measures the force applied to the hit/kick , while the IMU will collect the direction applied to the location as well. In **1.3**, IMU will measure the direction of the hit, so it will say how close it was to the target point. The accelerometer is the least desirable here since it only measures the force than according to the hit from the target area. In **1.4,** Accelerometers only measures the force and we can find one that can consume a lower amount of power for it to work. The IMU will not consume as much power as the accelerometer but since it measures more than one sensor, it could consume more power than the accelerometer. In **1.5**, The Accelerometer contains just one sensor and that one sole sensor measures the force of the hit and IMU possess different types of sensors, so it might just be one device that measures everything, but, in this ranking, it was considered as well all the sensors that are inside of the device.

Indicators (Natesha)

Technical Standards

The indicators used on this device should:

2.1	produce light with high enough intensity to be visible behind a translucent cover
2.2a	(each individual unit should) be durable enough to withstand repeated hits
2.2b	(connections between each unit should) be durable enough to withstand repeated hits
2.3	be able to easily shape as needed without excessive strain
2.4	sufficiently indicate zone in multiple colors with as few physical units as possible
2.5	have a lifetime comparable to that of the device itself.

Table 4: Technical Standards for Indicators

Design Matrix and Analysis

Standard:	2.1	2.2a	2.2b	2.3	2.4	2.5	Total
Addressable LED Strip	3	3	2	1	3	3	**15**
Electroluminescent Wire	2	1	3	3	2	2	**13**
Individual LEDs	1	2	1	2	1	1	**8**

Table 5: Design Matrix for Indicators

The best scoring option for our project's indicators are the addressable LED strips. These are strips of many LEDs, which a controller is able to individually address to control brightness and color. They offer more than enough brightness **(2.1)** and lighting control **(2.4)** for our project, and are relatively more durable **(2.2)** and long-lasting **(2.5)** than our other options. One potential shortcoming of these strips may be their stiffness; the thickness of some LED strips can make them difficult to bend into tight shapes **(2.3)**, but the benefits of using this indicator far outstrip this inconvenience.

We can easily rule out individual LEDs, our lowest scoring option. While this option can offer the same potential as LED strips, we would need to hand-solder and wire each individual indicator, making this choice significantly more fragile and difficult to shape **(2.2, 2.3)** than other indicators. Their brightness **(2.1)** and ability to indicate in different colors **(2.4)** is limited by this as well, since we would like to avoid using too many physical units, and their individual brightness cannot be pushed too high without risk of burnout **(2.5)**.

Electroluminescent wires came close in score to addressable LED strips. These devices are made of a phosphor-covered core wire and two thin outer wires, and produce light (the color is chosen by its plastic sheath) when AC voltage is applied. Ultimately, they are limited in their brightness **(2.1)**, especially over long periods of time, as they lose 50% of their brightness in 3000 hours **(2.5)**. They are quite durable **(2.2)** and easy to shape to our needs **(2.3)**, but we would need to pack multiple runs of EL wire in an area to use multiple colors **(2.4)**, while LED strips are able to show multiple colors and effects with one unit.

Processor & Memory (Joseph)

Technical Standards

The processor used by the device should:

3.1	Have sufficient input lines for all sensors and communication necessary.
3.2	Have sufficient output lines for all indicators and communication necessary.
3.3	Communicate with the UI system to provide raw data.
3.4	Require internal ROM and RAM.
3.5	Work with a volatile memory large enough to store data from multiple sensors.

Table 6: Technical Standards for Processor and Memory

Design Matrix and Analysis

Standard:	3.1	3.2	3.3	3.4	3.5	Total
FPGA	1	4	1	3	9	**9**
DSP	4	1	4	1	10	**10**
MCU (MSP)	3	2	4	4	13	**13**
Raspberry Pi	2	3	2	4	11	**11**

Table 7: Design Matrix for Processor and Memory

3.1 is the power analysis. For the point selection, the highest level was given to the processor with the lowest input power necessary to run. This is the DSP MCU. The MSP MCU was the second lowest input voltage requirement. This makes the MCU and DSP more attractive for total consumption of power and therefore cheaper operating costs. The raspberry pi runs at 5V making it a contender for second place with the MSP. The FPGA board requires the most input voltage at 7V-15V, making it the most expensive operating cost.

3.2 is the compatibility analysis. The main components to compatibility for this project is speed of the processors and the I/O options to connect to the sensors and screens. FPGA has several features that make it very compatible with this project. It has the second fastest speed, It has ample RAM (256MB) and flash memory (16MB), and has SPI and UART interface for connection. Another unique feature of the FPGA is the ability to change both the software circuitry and the hardware circuitry on the same board. This will allow us to code a different hardware implementation for each mode of operation, allowing for much greater flexibility of operation. The Raspberry pi is another contender for this mode due to its fastest speed and the amount of I/O that is available both on the baseboard and with extensions to the controller. The I/O is USB and standard GIPO headers so it has greater compatibility with both USB and GPIO sensors. It has 4GB of SDRAM. The MSP MCU has many options in terms of I/O, particularly UART, I2C, and SPI are the three main connections. But the total processing speed is only up to 25MHz, making it the slowest processor, but it edges out the DSP due to many different options of I/O. It also comes with 128KB of flash memory and 8KB of RAM. The DSP controller has middle road speed, but only works with I2C connection, making the connection type limited. The DSP has 2MB of flash memory.

3.3 is the cost analysis. At $120-$200 the FPGA is the most expensive board, followed closely by the raspberry pi at $100. The MSP and DSP MCUs are each relatively the same price at around $15.

3.4 is the implementation time analysis. The implementation time for the MCU and Rasp Pi are very similar due to both being already compatible microcontrollers with set hardware circuitry. This allows us for a more "plug and play" method for these processors. The FPGA board has programmable hardware circuits, which increases the implementation time because we have to code both hardware and software through the board. The FPGA does have I/O already part of the controller however which decreases the implementation time. The DSP MCU does not come with the same I/O board as the other three options, soldering would be necessary to implement the DSP MCU, making it have the worst implementation time. After considering all these options in the table above, the MSP MCU was determined to have the highest total score after the analysis.

UI System (Hannah)

Technical Standards

The device's user interface should:

4.1	Provide completed results to the user including number of total targets, number of total successful hits, total time per mode run, and average time per hit.
4.2	Receive user inputs such as turn on/off, select mode, and select level of difficulty.
4.3	Be capable of communication between itself and the base.

Table 8: Technical Standards for UI System

Design Matrix and Analysis

Category/Standard	Options	Ranking		
4.1: Display Results	LCD Screen (segments)	3		
	LED Screen	1		
	USB Port(.csv or .txt file export)	2		
4.2: Receive Input	Buttons w/ LED to show selection		3	
	Toggle Buttons w/ LED to show selection		2	
	Display w/ buttons		1	
4.3: Communication Device ↔ UI	TCP			1
	UDP			2
	Wired (SPI)			3

Table 9: Design Matrix for UI System

Standard **4.1** (provide results to user)'s highest rank was given to the LCD screen, as it is compatible with many processors, and can immediately display results to the user. However, this is not a finalized parameter of the design, as the display will be the last component that is configured as its dependent on the processor selection and type of communication used.

Standard **4.2** (receive user inputs) was won by the push buttons w/ LED lights to show selection option. Push buttons are relatively easy to implement and to program a debounce method. As push buttons and LED lights are relatively inexpensive, it is an easy and cheap way to be able to cycle through modes, while showing which sections have been made. Since push buttons can be easily programmed to cycle through modes with repetitive pushes, fewer buttons are necessary. The next best option would be to use toggle buttons instead of push buttons, as they do not have a bounding effect, although it would require more toggle selections. It may be easier for the user to interpret what selection they chose or change selections with toggle buttons.

Standard **4.3** (communication between base and UI system) would be best implemented with the wired connection, as it is the easiest to implement, time wise, and the cheapest. This selection may change depending on the type of processors used within the base and UI system. The second best option (if the processor was an embedded system & if a wired connection isn't ideal), would be to have a SPI as it's faster than I2C but only works in closer ranges. If the processor would be more advanced (like a raspberry pi), then a UDP connection would work nicely as it's faster than TCP and can communicate through its own local area network.

Power Supply (Joseph)

Technical Standards

The device's power supply should:

5.1	Connect to a standard US AC 110-120V input.
5.2	Use an AC-DC voltage converter (such as a 20V 60Hz AC to 12V DC converter) to provide power to the system's DC components.
5.3	Provide AC voltage of necessary magnitude to any components that require an AC input.

Table 10: Technical Standards for Power Supply

Design Matrix and Analysis

Standard:	5.1	5.2	5.3	Total
FPGA	1	3	1	**9**
DSP	3	1	4	**10**
MCU (MSP)	2	2	4	**13**

Table 11: Design Matrix for Power Supply

The power supply selection is highly dependent on which processor is ultimately chosen. If the raspberry pi is chosen, the included power supply is what will be used. The Multi output wall adapter has a wide selection of different output voltages and can be used with the FPGA and MCU. The 3.3V wall adapter will be used if the DSP or MCU is chosen to be the processor. The total is the same for all three power supplies as they all have very similar implementations. They all are wall outlet adapters that will plug directly into the processor unit. The Multi output PS has a micro USB outlet. Due to several of the processors use of USB-A input, no matter which PS is chosen (apart from the raspberry pi PS) a micro USB to USB-A adapter will be needed.

Programming (Hannah)

Technical Standards and Analysis

The programming implemented by this device should:

6.1	Work with both the base system and the UI system.
6.2	Incorporate receiving user inputs, communications, and computational mathematics.
6.3	Include low power modes while the system waits for input.
6.4	Utilize multiple timers to control the indicators and the sequence.
6.5	Detect when a sensor registers a hit and continue the sequence.

Table 12: Technical Standards for UI System

As shown by these standards, the programming language selection is highly dependent on which processor is ultimately chosen. The language in the FPGA is Verilog. The IDE for the MCU can compile embedded C. DSP programming uses C/C++. The language used for a Raspberry Pi is Java/Python/C/C++/C#.

Project Diagrams

Mode Software Flowchart

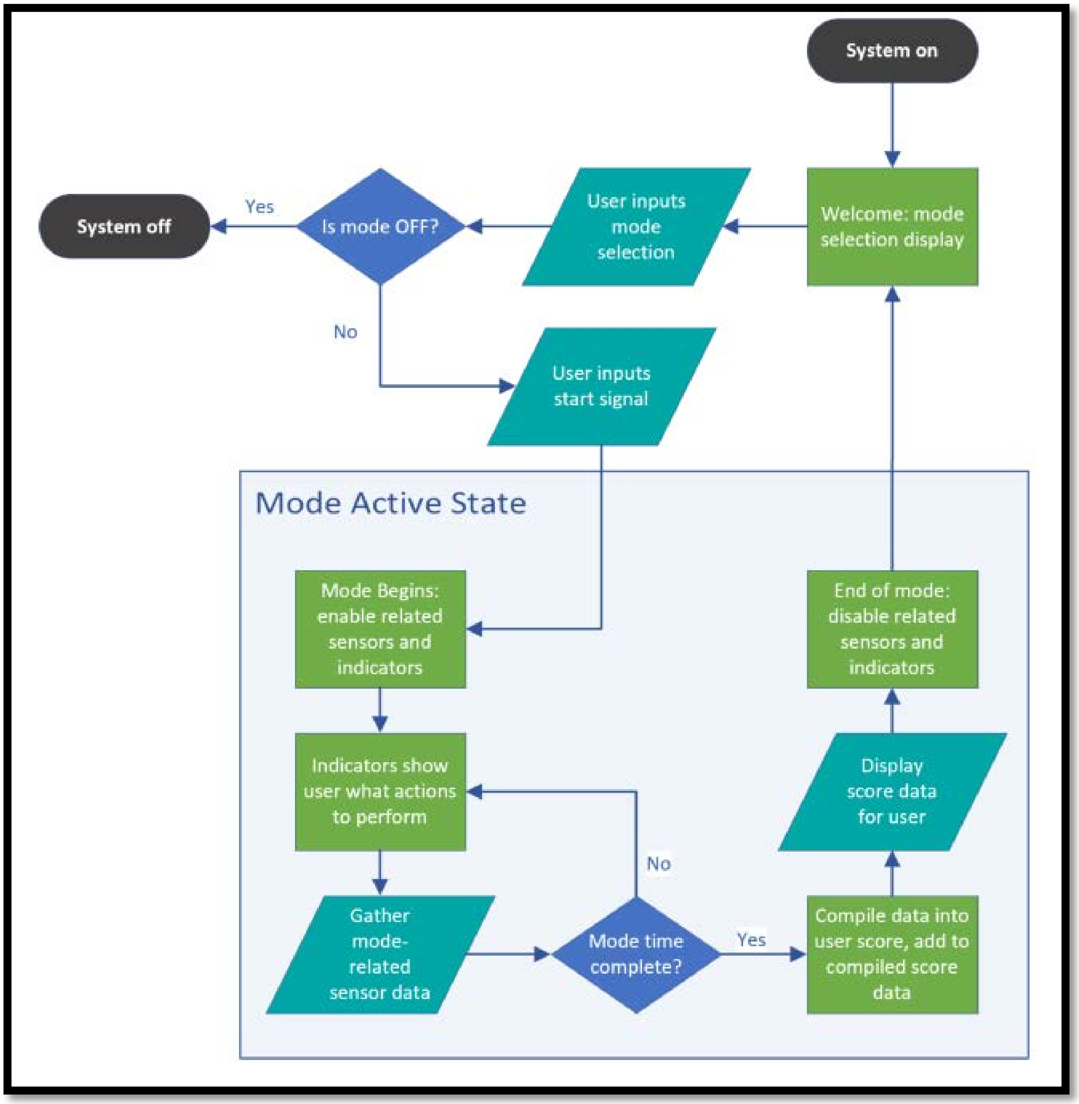


Image 1: Potential Software Flow Diagram

Hardware Block Diagram

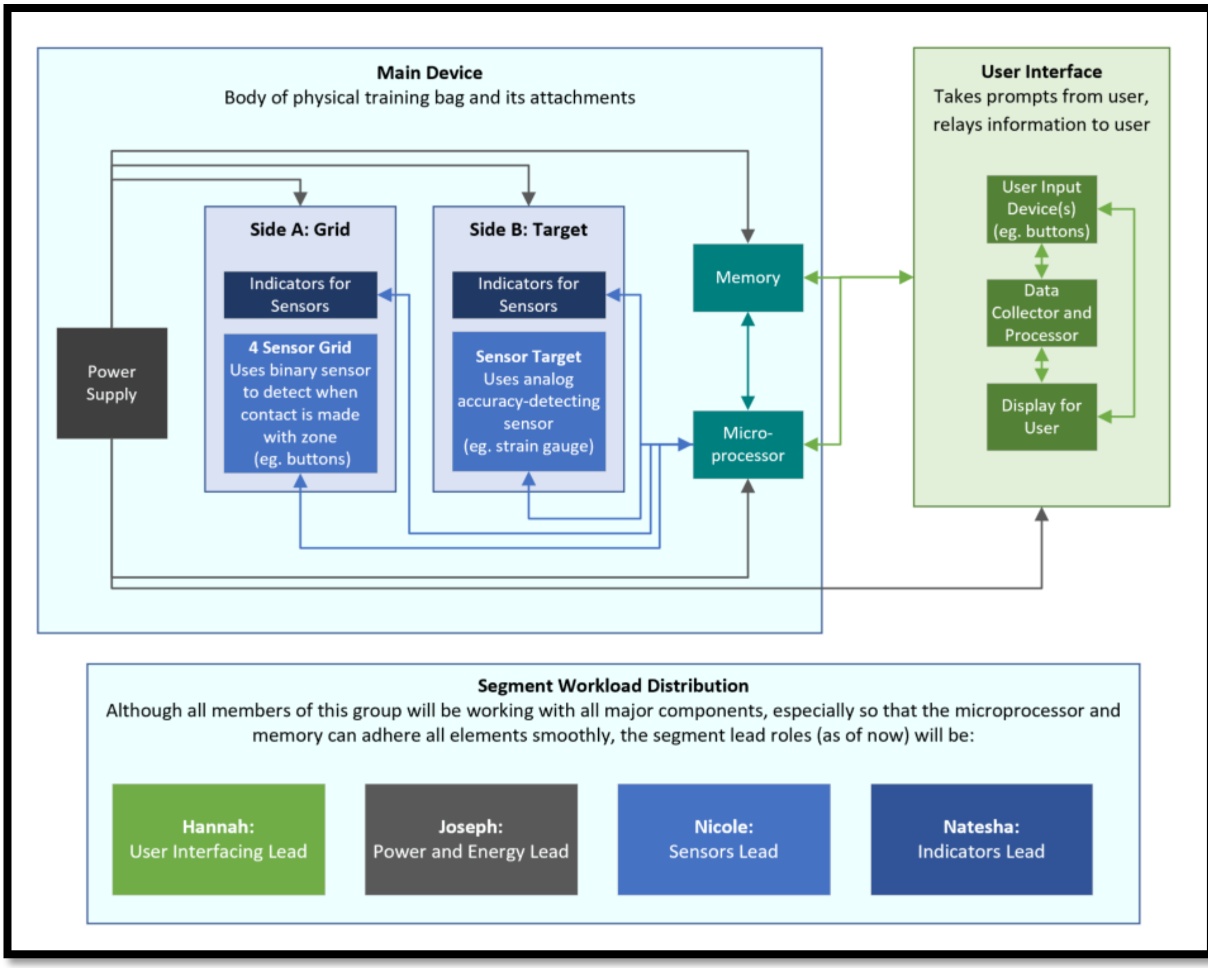
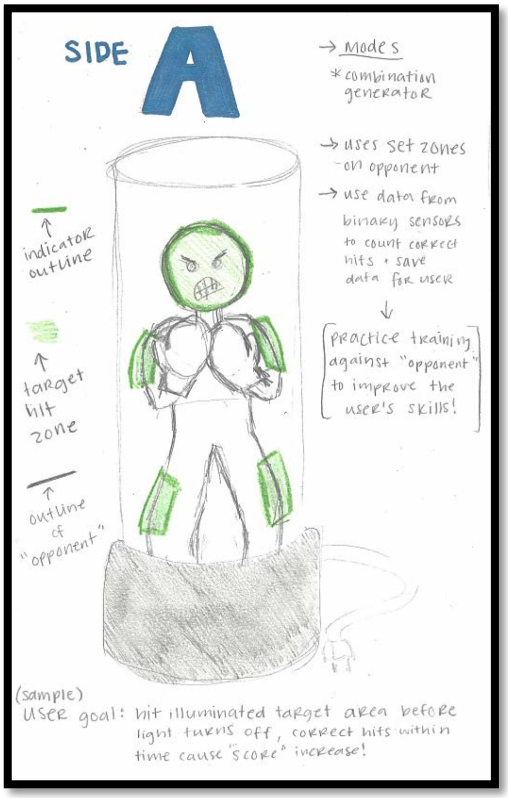


Image 2: Hardware connection block diagram; description of current distribution of group roles

Prototype Concept Illustrations



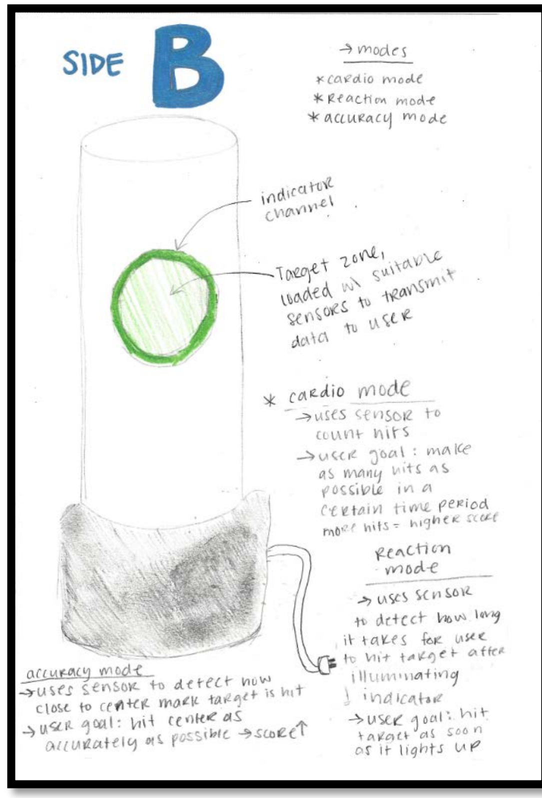


Image 3 (Left): "Side A" of the device **Image 4 (Right):** "Side B" of the device

Note: the descriptions of modes in these diagrams are available to be read in the General Specifications section on page 4.

Design Considerations and Decisions

Initial Milestones and Timing for Senior Design 1 and 2

Components: Decide Processor, Sensors, Indicators, UI System

The prototype should be done by the end of December, and the prototype UI System should be finished by the end of January, to give us enough time to debug the project, and move our components over to the final design. Temperature sensors are optional to the very end to measure the temperature of the sensor grid and to make sure it is not overheating.

Construction: Complete Sensor Grid (Side A) and Target (Side B)

This project will need multiple areas with sensors, so these will need to be completed as soon as possible to be able to incorporate the electrical components, indicators, and the software which depend on the sensors. The deadline for the first sensor grid prototype is beginning of January, and the other three are no later than the end of January.

Final Touches: PCB and Interface Completion

The PCB components should be completed no later than the beginning of March. This will give us time to finalize the project and be sure that all modes that are included are functioning as intended. Once this has been verified, we would like to begin working on additions that will make this device something that will be worth the cost for a customer. One possibility of this is making an extra display on the body of the bag to interface with the user and add another element of interactivity. This will be challenging due to the frequent impacts the bag will face that could damage the screen, but it would allow more elements to be added to diversify the training possibilities from this device alone.

Group Member Familiarity with Project Elements

- **Hannah** is familiar with software since she works in a company with software engineers. She also likes writing code and programming in her free time.
- **Joseph** is familiar with processors and microcontrollers. He has knowledge of the process of ensuring a system is powered sufficiently.
- **Nicole** took an in-depth sensors class, so she is familiar with various types of sensors that we could use for this project.
- In addition to knowledge of typical electrical engineering curriculum, **Natesha** has many years of experience with and regularly uses a self-standing kicking bag since she is a fourth-degree black belt in karate and works as a sensei in her dojo.

With the group’s combination of knowledge of electrical engineering principles and real-world experience with all aspects of this project, we should be able to successfully create and develop this device.

Project Challenges

Most challenges that our group will be facing are a result of the CoVID-19 pandemic that we are currently living through. Due to social distancing, we will need to work on our sections separately; as of now we will not be able to get together unless it is to test the device. As of now we are doing most of our meetings and communication via Discord, but once sensor/indicator testing begins, we will likely face difficulty in being sure everyone has what they need to individually test components.

In general, being able to optimize the parts used to cater to all specifications as well as being low cost, accessible, and realistic for the scope of this design prototype will be a challenge. The pandemic has also slowed electrical component production around the world, meaning that we will have to be sure about our designs before ordering many items and begin attaining and assembling components as soon as possible in attempts to avoid roadblocks.

One non-pandemic related obstacle is the size and scale of this project. A self-standing training bag is a large and heavy piece of boxing equipment, so this component should be in a location accessible to all members with sufficient space to work on device assembly and construction.

Budget and Financing

Our group will self-fund this project. We are budgeting to contribute a maximum of $500 USD, divided equally among our teammates. As of now, no equipment aside from standard circuit components (eg. wires, resistors) has been acquired. All parts of this project must be attained at some later point. Table 1 below depicts estimated costs of each component.

The individual component that would cost the most money is a self-standing kicking bag since it is a big and technical instrument aimed at people who are experienced in this area. However, finding a used bag will lower the cost of this item, and will not make a difference in the quality of the bag as it will need to be changed to incorporate our components regardless of its condition.

Although sensors are generally inexpensive, we would need enough components to detect contact in the large target area so that the device can test accuracy in addition multiple sensors to detect contact in a binary "hit or no hit" manner in all areas of the sensor grid.

Component	Price (USD)	Quantity	Total (USD)
Sensors	$0.4 to $5.00	20	$6.40 to $100.00
Processor	$10.00 to $15.00	2	$20.00 to $30.00
RAM	$50.00	2	$100.00
Miscellaneous Buttons	$1.00 to $5.00	4	$4.00 to $20.00
Standup Punching Bag	$130.00	1	$130.00
Indicators	$10.00 to $30.00	1	$10.00 to $30.00
AC to DC converter	$2.36 to $20.00	1	$2.36 to $20.00
Miscellaneous Hardware	$20.00 to $30.00	n/a	$20.00 to $30.00
Total			$292.76 to $430.00

Table 13: Expected costs for this project