Senior Project Proposal

Kristen Argyle, Michael Anderson, Aundrea Hargroder

***Abstract*—Martial sports are a paradox – a compromise between safety and accuracy, especially when weapons-based martial arts are the basis for a martial sport. Weapons by their nature are dangerous and every step taken to make them more safe changes their essential characteristics. By design, safe weapon simulators make it less obvious when a fight-ending injury would have occurred. Thus, it is difficult to assess in martial sport how much the martial art is being applied and practiced. Using technology, this project hopes to address that problem. This project will take a European Longsword simulator and integrate an embedded system that will detect and report martially sound strikes. This system will filter out parries and contact too light to be considered valid while assessing the quality of a true strike. As such our system must be rugged and prepared for taking impacts. Our goal is to contain the entire system within the sword simulator while disturbing its handling as little as possible.**

***Index Terms*—force sensor, accelerometer, martial arts, martial sports, sabertron, contact assessment, rotational dynamics, force threshold, force sensor array, impact efficiency, sweet spot**

I. INTRODUCTION

> IT'S A SWORD, said the Hogfather. THEY'RE NOT MEANT TO BE SAFE.[1] [1]

Use of electronics in martial sport is nothing new. Electric scoring was adopted for Olympic Fencing competition in 1936 [2]. Though fencing is virtually alone in its use of electronic scoring, numerous small studies and applications have been used in assessment of martial sport. Accelerometers and force sensors are an obvious choice to get a more objective estimate of an impact's imparted force. Using such methods as a judging mechanism would be unnecessary for sports like boxing and mixed martial arts, since the winner is determined by who is left standing, roughly speaking. Spectators and judges don't need a readout of the force behind a punch to determine if it did damage – the damage is there to see and the competitor will either carry on or succumb. With fencing, on the other hand, this option is not available as it is imitating a fight with weapons far more deadly than fists.

Some martial sports scale back the potential damage by imposing restrictions on competitions, allowed techniques, etc., but weapons-based martial sports are always bound by the safety imperative to introduce simulators for the weapons they train. In a competitive environment even weapon simulators can be dangerous, so safety equipment is also used. This is common sense preservation of life and limb, evident even historically in the use of wooden training weapons.

Enter modern European longsword competition and training [3]. Judges are still needed for competitions while honesty is necessary for friendly free-bouting. Both methods are prone to problems of subjectivity and human error, not to mention the constant monitoring a competitor must impose on herself. A competitor has a twofold challenge in two directions: 1) did they receive a hit, 2) did they deliver a hit, and then both are multiplied by the question: was the hit seen/felt and counted? Our project aims to use technology to alleviate some of this problem.

[1]RIP, Sir Terry

II. BACKGROUND

A company, LevelUp Inc., has already created a solution for counting sword hits. It is called "Sabertron," and it caters to the more casual field of foam sword fighting, a prototype of which can be seen in Fig.1. Using components found in cell phones, the product discards detected hits to the opposing sword and counts hits to the body. All of this is contained in the sword [4]. For our project this is a convincing proof-of-concept that we'd like to build our own version of using more rigorous sword simulators.

Fig. 1. A one pound foam sword with electronic scoring using accelerometers, Bluetooth, and a touchscreen [4].

We will also rely heavily on accelerometers not only for hit detection but for quality assessment. We're looking to a study on cricket bats as a practical test of accelerometers for this purpose.

The point on a bat at which the most energy is transferred is often referred to colloquially as the "sweet spot." The less one's hands feel rattled by an impact, the more efficient the energy transfer – and the less discomfort for the user. In a series of experiments with cricket bats, engineers used 3-axis accelerometers to detect the amount of lost energy, or "jarring," upon an impact in order to identify the sweet spot on a cricket bat [5]. Accelerometers were placed on both the wrists of the batter and on the bat, as pictured in Fig.2.The data revealed that different parts of the bat indeed caused different wrist accelerations, and they were able to estimate the location of the sweet spot [5].

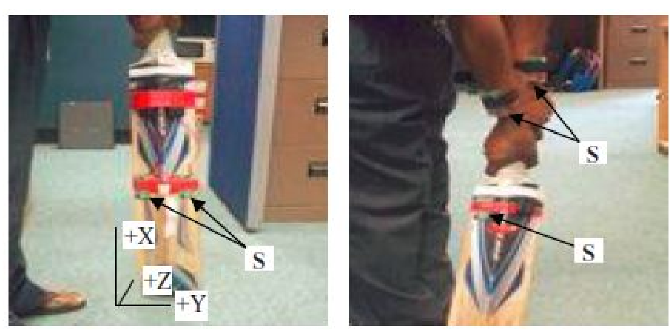

Fig. 2. Accelerometer locations for cricket bat "sweet spot" tests [5].

Based on this study, we know more than one accelerometer in the right area can give us information on the quality of the hit. Quality for us indicates how efficient the strike was. A hit on the point of percussion will result in the most force imparted to the target and the least kickback to the hands [6]. Likely places for these accelerometers will be discussed in "Proposed Work" under "Advanced Functionality" below.

III. Proposed Work

In ECE 5780 Embedded Systems, we have implemented the following basic requirements for our smart-sword system as a proof of concept:

1) Hit detection
2) Outward indication of hit detection

More complicated requirements that we will implement for the Senior Project are:

1) Assessment of quality of hit
2) Discarding hits to opposing sword
3) Upload and display of information off-sword

In the baseline design, we plan on improving the Sabertron design by maintaining the integrity of historical sword fighting. We will do this in two ways. First, we will make the physical modifications to the sword minimal. The mounted system components will affect the weight and feel of the sword in the slightest amount. Second, we will implement precise real-time analysis of hits. The user must be informed of the success of their hit in the same amount of time as they would in a real fight. For this reason, the real-time demands of precision and speed will be our first priority. Other analysis and features can be considered post-event and do not have the same real-time constraints. Certain strike analysis features like discriminating what kind of hit was made – cut, thrust, slice – will have to be vetted for the possibility of real-time analysis. If they are not up to scratch, they will have to be slated for post-event analysis.

Post-event analysis introduces the challenge of maintaining our baseline functionality while logging or transmitting data somewhere else to be logged.

Further features will be described in the "Additional Features" section below.

For hit detection, we plan on using one or more accelerometers. We can apply the concept from the cricket-bat study to our application, using multiple accelerometers to determine the quality of a hit.

Outward visual indication of hits can be achieved using LEDs. These LEDs can be used to indicate the amount of times the user has hit his or her opponent and the amount of times the opponent has hit the user of the smart-sword. Hit indication could be enhanced by using audio indications from a speaker placed in the pommel of the sword.

Discarding hits to the opposing sword will require some form of wireless communication between swords. RFID has been initially suggested as a solution. The Sabertron project used radio for wireless communication initially as well, but found that it caused a significant power drain in their setup. They opted to switch to Bluetooth [7], which is a possibility we'll explore in the course of the project. An overview of the spart-sword design is shown in Fig.3 below.

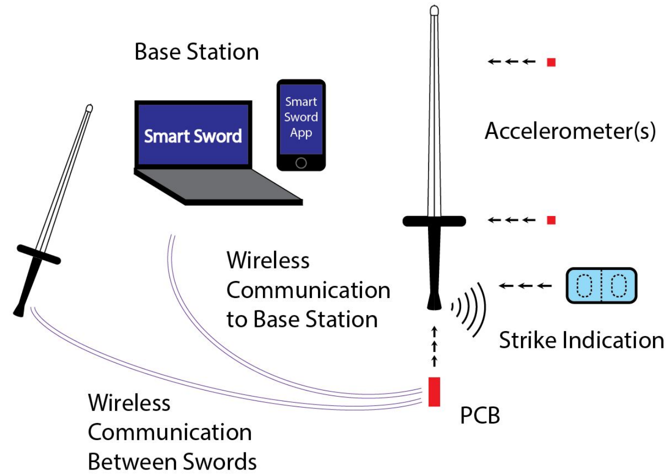

Fig. 3. The major components of the smart-sword design.

For the ECE 5780 proof-of-concept our plan is to use an already established discovery board with an integrated accelerometer. This will give us a good platform to begin selection and development of more custom parts to be used in our Senior Project.

The measurements this project will prioritize are:

1) impact force
2) character of impact

The smart-sword capabilities this project will prioritize are:

1) safety features preserved
2) handling features preserved
3) ruggedness of measurement equipment
4) sensitivity of measurement equipment

If we initially used a stick for testing and calibration and then moved to a sword, our numbers would be off due to the fact that a sword is weighted and mass distributed differently than a hegemonic stick. Therefore, we will do our testing and development on a nylon sword simulator of comparable weight and mass distribution to a real sword from Purpleheart Armory [8]. This simulator will initially be in the same series used for HEMA tournaments [8] [9]. Later, we may need some custom sizing or shaping to integrate the hardware properly.

A. Base Simulator

An example of the specific model we've chosen can be seen in Fig.4. We've chosen our simulator based on a few criteria. Already mentioned was weight and mass distribution as we want to preserve the integrity of a real sword's handling, which lead us to the Pentti Type III series.

Fig. 4. The base simulator we will use to calibrate and test our smart sword design. This model was originally designed for use in hot, dry climates, as the other models had a tendency to break [8].

Additionally, we chose a one-piece model that does not have a steel crossbar going through the base of the blade. Instead this model has an attachable crossguard, most of which is hollow. This allows us to do two things: first, we may run wiring from the blade to the hilt through the nylon; second, we may mount components inside the removable hilt piece, where they are both accessible and protected.

B. Basic Functionality Proof-of-Concept

As a proof-of-concept, we have created a basic implementation of our smart-sword system for our 5780 Embedded System Design course. Our basic implementation consists of two seperate systems, each with an accelerometer, an STM32F4 Discovery Board, an Arduino, a Piezo speaker, LEDs, and an RF tranceiver. This system lights a green LED when a hit is detected and plays a sound through the speaker[2]. Additionally, the hit detection is communicated through the RF tranceiver to the other system. This allows for the detection of simultaneous hits, which mimics the action of two swords striking each other. When a simultaneous hit occurs, each system detects a hit and communicates it to the other, and the simultaneous hit is indicated with a red LED and a different sound through the speaker. This is a raw proof-of-concept for the most important parts of the project and is not the entirety of the base functionality we want for our Senior Project. The functional block diagram for our proof-of-concept design is shown in Fig.5.

[2]This is our ECE 5780: Integrated Circuits project.

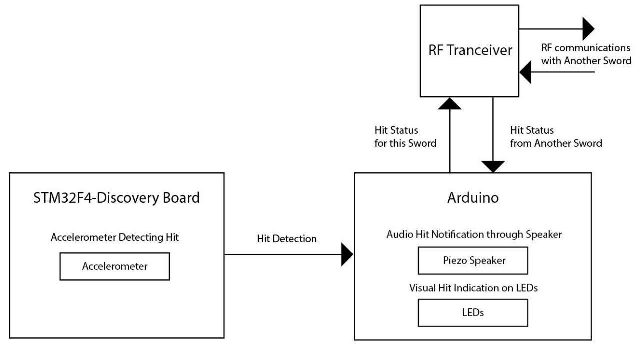

Fig. 5. The functional block diagram is a higher-level depiction of the proof-of-concept design for 5780 Embedded Systems.

1) Hit Detection: We have been successful in detecting if a hit has occurred using an accelerometer. We found that to detect the large changes in acceleration that characterize a sword strike, only a single accelerometer is required. This is insufficient for calculating the efficiency of a strike, but sufficient for detecting an impact. We chose the accelerometer because it is a small sensor which can easily fit within the body of a practice sword. The accelerometer is also very cost effective. We feel that the accelerometer will provide the best data for performing hit detection analysis.

In the proof-of-concept design this hit detection was done with one accelerometer located on the STM32F4 Discovery board. Later, we hope to have at least two dedicated accelerometers at different locations to more sophisticated acceleration readings.

2) Hit Indication: The first form of hit indication performed by the smart-sword system was through LEDs placed on the hilt of the sword. The final placement of the LEDs will be such that the user can read the amount of LEDs that are lit while wielding the sword. The placement should also minimize the chances of the LEDs being damaged during combat. The simplest form of hit indication will be LEDs that display the amount of times that the user of the sword has struck his or her opponent. An advanced implementation will include LEDs to indicate the opponent's score as well. We plan on using LED strips of ten to twenty small LEDs for this purpose.

Since LEDs on the hilt of the sword will be difficult to view a fast-paced sparring session, we also implemented a speaker to indicate the result of the strike. In the proof-of-concept design, the speaker used was located on an Arduino board. In our future designs we would like the speaker to be embedded in in the pommel of the smart-sword where it would be unlikely to get hit.

3) Hit Assessment: In order for hits to have meaning, we must know if the sword hit another sword or if it hit a person. Our solution is to first have both swords communicate wirelessly with one another. Should one sword register a hit, it will first confirm with a second sword. If the second sword also detected a hit, it will be registered as sword-to-sword contact. Otherwise, it will be counted as a hit to a person.

This leaves open a gap – what if two people hit each other at the same time? The accelerometer data should spike differently for sword-to-sword contact than for sword-to-person contact. Swords are very rigid compared to people, so our application could analyze the accelerometer data to get further information about the character of the spike. What this would or would not "count" as would be somewhat arbitrary since counting both hits or neither are both valid ways of measuring success or failure. This could be an interesting choice to leave up to users to configure later with the use of an app. Clearly it is more intensive to analyze spikes than to just detect that the force went over a threshold, so not counting a "double hit" is an easy default choice.

C. Advanced Functionality

In order to take our smart-sword to Senior Project level, we need more sophistication in our data gathering and display of that data. To do this we will break out and add an additional accelerometer, as well as choose a microprocessor that fits our needs. Additional hardware will be integrated into the system as we add features. While we will use pre-built parts initially, we may design a custom PCB for our system.

1) Hit Analysis: On the surface, in order to analyze a hit we need at least one accelerometer and to know precisely where it is placed. For example, an accelerometer placed at the node, *B*, where the hand controls the pivot as seen in Fig.6 would tell us the kickback effects of a strike. If we registered little or no acceleration, the hit might have been close to the center of percussion *A* in Fig.6 or it might have bee a light hit. This means we cannot rely on one accelerometer to tell us how much force was imparted to the target. Therefore, in order to analyze hit efficiency, we need at least two accelerometers: one to tell us the acceleration change measured at the center of percussion and another at the hand pivot node. By comparing these, we will be able to know the difference between an efficient strike and a light tap and get a rough estimate of the actual efficiency of the strike.

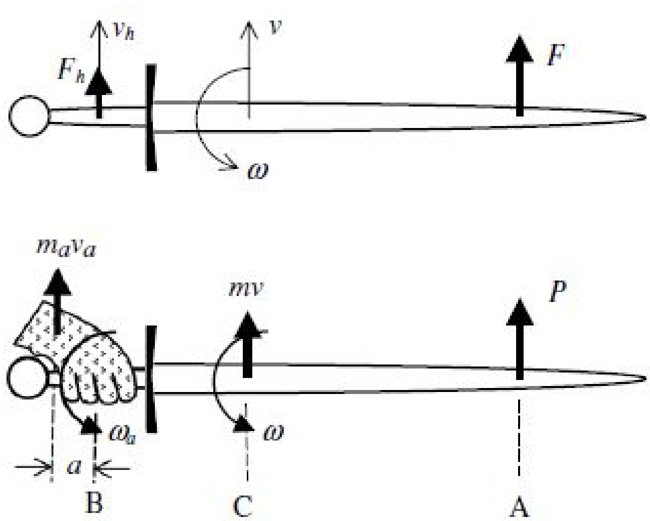

Fig. 6. A diagram showing important locations and forces working on a sword. In the top image, F is an impulse force on the blade at the center of percussion and F_h is a reaction force, while v_h is a reaction velocity on the grip of the blade. The center of mass's movement after the impulse force is represented by its velocity v and its angular speed ω. The bottom image shows an analysis relative to the rotating arm [6].

For example, a deceleration spike at point *A* with very little jarring at point *B* would be consistent with a good hit to the center of percussion. Two smaller spikes at both points would indicate a less optimal strike. The more accelerometers we add to specific nodes the better we can analyze the strike and filter out any false-positives.

The analysis of the data from the accelerometer and assessment of the strike will likely be carried out by software on a processor. We will have to gather data from strikes in order to determine where our thresholds should be and how categorize efficiency calculations.

A simple way we can indicate strike efficiency is through our LEDs. Number or color of LEDs would be a simple, recognizable indication that would give instant feedback to the user.

2) Interface: During implementation of our advanced features we will need a basic interface for moving data off the device. For this purpose we have a 2.4GHz wireless module that we will use to create an interface for the device. The data will be moved from the device into a Microsoft SQL database for long term storage where it can be analyzed.This interface will also be used so that the code running on the device can send information such as the number of hits detected, hit type, quality, etc.

As mentioned, one of the proposed methods for determining if a strike was parried is inter-sword communication. The device will need to be able to view data from the sword which was struck to determine if it had struck the opponents sword or something else.

A simple interface which provides us access to this data should be sufficient for implementing all of our advanced functionality. Viewing this data in a meaningful way is the subject of one of our proposed features. In the event that we are unable to implement this interface using the 2.4GHz wireless module or it doesn't suit our needs, we would investigate Bluetooth as an alternative wireless interface.

3) Custom Boards: The discovery board is temporary as we design our own solution. Due to its size, it would be unwieldy and prone to being struck. Ideally, we would want to place components where we want them on the smart-sword. This means processors and non-sensor hardware needs to be in a place subjected to less impact stress, while sensors need to be at specific nodes where the physics are easily calculated, known, and helpful.

Depending on the magnitude of the forces our smart-sword will be subject to, we may need to explore different accelerometers with different ranges and sensitivities. These accelerometers may not be available with analog-to-digital converters, in which case we'll have to design our own.

A custom shaped board (or boards) would also be helpful in order to embed the system in the sword simulator without disturbing the sword's shape or handling. This is mostly an advantage for the idea of mass modification or manufacture of the smart-sword system.

D. Additional Features

Once base functionality is established, additional features can be added.

1) Charger: The sword will require a USB or 5V power supply connection until we are able to implement a battery. In the event that we meet our project deadlines we feel that rechargable battery would be a good additional feature to implement. To determine the battery size that would be required we need to analyze the wattage used by the device after creating the prototype.

The battery could simply be an embedded portable USB battery which is commonly used for charging cellphones. If we used one of these chargers to implement our embedded battery, we would simply need to fabricate our prototype to fit the battery and place a USB port on the outside of the prototype to charge the portable USB battery. If we determine that we want to use an OEM battery, we would need to find one that has it's own DC charger, a compatible charger, or we would need to develop our own DC charger.

Designing our own charger would be useful for learning more about the electrical engineering aspects of our project. However, this may not be a feasible plan of action due to time constraints on the project and the large number of features which we have planned to implement.

2) GPS: An additional feature which we feel might be useful is a GPS module. The GPS module would allow us to very accurately synchronize time on the device so that it is able to better correlate data with other devices. We had the idea that we might want to track teams of users simultaneously so the GPS data could tell us which users are in immediate proximity of each other so that we can analyze which combatants are currently engaged. The GPS data can also be used to view the movement on the field of combat. It wouldn't be difficult to have the sword record GPS coordinates of the wielder, transmit these coordinates to a database, and then overlay these coordinates onto a map.

3) Advanced Interface: Using data delivered by the sword, we hoped to create a more user friendly way to analyze the data. Using the SQL database, we hope to eventually create an application or website which will allow the user to view information from the database in a meaningful way.

A simple example would be a page that allows the user to view a table or time line of all sword strikes and information about the strikes such as type, quality, etc. If GPS is added to the device it would also be possible to overlay different events that have occured (e.g. such as a parry of strike) onto a map of where they occured and where the device has been.

4) The Three Wounders: There are three ways to injure someone with an edged weapon. The strike, cut, or "hew" is obvious as a percussive, hacking or chopping motion. A thrust relies on the point penetrating through a target. Finally, the less obvious but used daily method is a simple slice using the edge with no percussive element.

We've dealt mainly with the strike in this project description, but using multi-axis accelerometers impacts from thrusts can be differentiated from those of strikes. The difficulty lies in the tendency of thrusts to deflect off of safety equipment, therefore robbing the deceleration impact. While this means safety equipment is working as designed, it presents a challenge to our design.

Slices are the trickiest prospect as they rely on smooth velocity not sudden acceleration. We may be able to leverage this fact in some way by detecting little or no acceleration in the perpendicular to the edge but a modest acceleration parallel to the edge. However, it may be better to explore another hardware solution such as light detectors instead.

With the translucent nylon simulators, light detectors could be placed below the surface at intervals and the condition for the slice could be that one or more light detectors detect darkness (due to contact) which then appears to the light detectors to moves to other light detectors.

E. Team Responsibilities

We've broken up responsibilities into major sections. The responsibilities are separated base on what we feel are the major tasks of implementing each milestone. Each task has two assigned individuals from our team. One individual holds the primary responsibility and the other holds secondary responsibility for the task. Each person holds one primary responsibility and one secondary responsibility.

1) Application: For the basic functionality the application task will involve writing a high level application which will process data received through the accelerometer to determine if a hit has occured and based on this information generate some external indication. This would initially be simply turning on the LEDs when a hit is detected. For implementing the advanced functionality the application will need to add additional functionality for advanced hit analysis and support for the simple RF interface. For implementing the additional features the application task would include writing the logic for performing advanced hit detection and recording the GPS coordinates at regular intervals.

The primary individual assigned for this task is Kristen. The secondary assigned individual for this task is Michael.

2) Drivers and Interfaces: The application will need to make high level calls to a number of hardware devices. Implementing the basic functionality will require writing interfaces for the accelerometer and the GPIO. For implementing the features outlined as advanced functionality the driver task will include writing a low level driver interface for the RF communication using SPI. For implementing the additional features a GPS driver would need to be written for the application to use. This task also includes implementing the basic and advanced interfaces used for viewing the data generated by the device.

The primary individual assigned for this task is Michael. The secondary assigned individual for this task is Aundrea.

3) Documentation and Testing: The documentation task will include updating functional and component diagrams as the project expands and changes, and maintaining the project information that will be used in the final report. The testing task will include testing each individual component of the project separately as it is completed before integrating it into the project with the rest of the components to ensure quality in each step.

The primary individual assigned for this task is Aundrea. The secondary assigned individual for this task is Kristen.

F. Risk Assessment

The most obvious risk of our project is the physical strain placed on the electronics. Sabertron limited this strain by using foam swords. We would like to improve on the practicality of their design by using nylon swords that are wielded more similarly to real swords. Since the material our swords are built from is less forgiving than foam, the physical strain on our electronic components will be more significant. Accelerometers are designed to tolerate a certain level of impact stress, but given the somewhat chaotic environment of competition they may be subjected to more stress than intended. For example, an accelerometer in the hilt of the sword that is meant to detect a less wide range of accelerations might suddenly be subject to an impact from the end of another sword – which must have the widest range available. To mitigate this, we may use accelerometers of our highest expected range for all areas. This risk is primarily financial in nature.

The risk of breaking parts places constraints on the mechanical design of our project. Parts need to be readily accessible and replaced should the worst happen. This means parts need to be embedded with a method of extraction that doesn't damage the sword or attached to the outside.

Another area of our project with obvious risks is our choice of the 2.4GHz RF wireless module for communication between devices. When developing their product, Sabertron opted for Bluetooth instead of this approach. We would like to implement the RF wireless module option because it is easy to modify the protocol to fit our specific needs. This would allow us to implement a more robust means of communication. However, as the range between devices increases in RF communications the power consumption increases as well. Should we find that the 2.4GHz RF wireless module takes up too much power like the Sabertron team did, we will switch to Bluetooth communication between devices.

Observing Sabertron's development also gives us another important risk to consider: time and expertise. The development of Sabertron has taken years with experienced Computer Engineers, and we have less than a year. While we don't have to consider mass manufacturing, marketing, or other product-development concerns, Sabertron has overhauled the product firmware at least once during development [10]. What this means for us as amateurs is a need for significant devotion to our own firmware. We run the very real risk of failing to create drivers and software up to the rigors and constraints of our application. As this is a combat simulator aid, actions happen in less than the blink of an eye and we must catch those actions.

Lastly, since our project goes beyond all-real-time functionality, we need to consider the risks of creating additional data logging and analysis systems for post-event assessments. We run the risk of bogging down the speed of the real-time system by preserving information. Navigating this hazard may end up taking much time and effort and potential overhauls of our system. One way to mitigate this risk would be to have parallel systems. That is, putting in more hardware to work in parallel: one to handle all the real-time demands and another to log the data for later analysis.

IV. Schedule

The project schedule can be seen in Fig.6. It has been sectioned into three parts as outlined previously: Implementing Basic Functionality, Implementing Advanced Functionality, and Implementing Features. We expect to have the ability to detect hits using the discovery board by the end of the current semester. Our team has a agreed to spend time on implementing advanced functionality during the summer break and by that time we hope to have our product in a state that can be demonstrated to the public. With the additional time before demonstration day, we will attempt to add features which we feel aren't necessary to be demonstrate the device. The deadline for implementing basic functionality is May 6th, 2015. The deadline for implementing the advanced functionality is August 23rd, 2015. The deadline for implementing additional features is December 11th, 2015.

We planned the schedule in this tiered fashion so that we can allocate additional time to the most important features in the event that it takes more time than we currently have allocated. We can neglect our optional features in favor of implementing core functionality if the need arises.

Fig. 7. Project time line calendar.

V. Required Resources

The required resources will depend on how much functionality is ultimately implemented. We may require additional accelerometers or other devices to help perform advanced hit analysis. We may also determine that Bluetooth is a better alternative to our 2.4GHz wireless module.

In the event that we need to fabricate our own custom board(s) the part list will need to change to include items for the custom board. We hope to obtain a good understanding of what parts will be required for a custom board after implementing our basic functionality using the discovery board.

.

References

[1] T. Pratchett, *Hogfather*. Victor Gollancz, 1996.

Parts required for project:
STMicroelectronics STM32F4DISCOVERY
NRF24L01+ 2.4GHz Antenna Wireless Transceiver Module
Nylon Practice Longsword
Additional accelerometers (optional)
5v Lithium ion battery and charger (optional)
GPS module (optional)
Bluetooth module (optional)

[2] (2015) Fencing. [Online]. Available: http://www.britannica.com/EBchecked/topic/204172/fencing/2263/Organized-sport

[3] M. W. Bishop, "Medieval weapon finds modern appeal," *New York Times*, 2014, 15 September. [Online]. Available: http://nyti.ms/1uBcsWD

[4] T. R. D. Lynch, J. Davidson. (2015). [Online]. Available: http://sabertron.com/

[5] A. K. Sarkar, D. A. James, A. W. Busch, and D. V. Thiel, "Cricket bat acceleration profile from sweet-spot impacts," *Procedia Engineering*, vol. 34, no. 0, pp. 467 – 472, 2012, {ENGINEERING} {OF} {SPORT} {CONFERENCE} 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877705812016931

[6] M. Denny, "Swordplay: an exercise in rotational dynamics," *European Journal of Physics*, vol. 27, no. 4, p. 943, 2006. [Online]. Available: http://stacks.iop.org/0143-0807/27/i=4/a=025

[7] D. Lynch. (2015, 12 January) Sheath and blackout edition revealed. [Online]. Available: http://sabertron.com/sheath-blackout-edition-revealed/

[8] (2015) Type iii federschwert longsword plastic guard 1. [Online]. Available: http://www.woodenswords.com/product_p/type-iii-f-pg1.htmr

[9] (2015, 18 April) Study in steel. [Online]. Available: https://www.facebook.com/events/381670271994783/r

[10] T. R. D. Lynch, J. Davidson. (2014, 4 December). [Online]. Available: http://sabertron.com/sabertron-smooth-new-firmware/