1. Introduction

Objective:

A laryngoscope is a medical tool used for examining the larynx during a medical procedure called a laryngoscopy. When performing a direct laryngoscopy, a doctor has about a 10 minute window while the patient is under general anesthetic to insert the laryngoscope and cut and remove blockages that they were looking for [4]. This is a complicated procedure that requires a decent amount of practice to ensure high success rates. Consequently, performing this procedure repeatedly on different cadavers is not a viable option for training.

We propose to ameliorate this issue by taking advantage of recent improvements in virtual reality and sensor technology. We propose to create a virtual environment for doctors to train performing this procedure so that they can get the practice they need without burning through resources each trial. Our plan is to develop a laryngoscope with an IMU to track orientation, and a linear motion sensor to track how far the tube has been inserted in the throat. These sensors would collect information about how the trainee is performing the surgery, and would send this data through a microcontroller to the computer connected to our VR headset. The simulation environment would adapt and change in real time in response to the inputs provided by the trainee. We believe that this would provide medical professionals with adequate training while saving thousands of dollars on cadavers.

Background:

The healthcare industry is one of the largest and fastest growing industries in the world. It is projected to rise from 7.077 trillion dollars to 8.734 billion dollars by 2020 [1]. Currently, a major issue that medical professionals encounter while training is the lack of resources available to actually simulate complex and delicate procedures. A synthetic human cadaver costs about 40 thousand dollars each, and fewer than 20,000 real cadavers are donated to science each year [2-3]. This pretty much matches the number of medical students in all of the US. With the growing demand for improved health care, doctors must be adequately trained to perform procedures. Clearly, there is a mismatch between the number of doctors and training resources available for training physicians. This leads to a rise in malpractice and botched medical procedures.

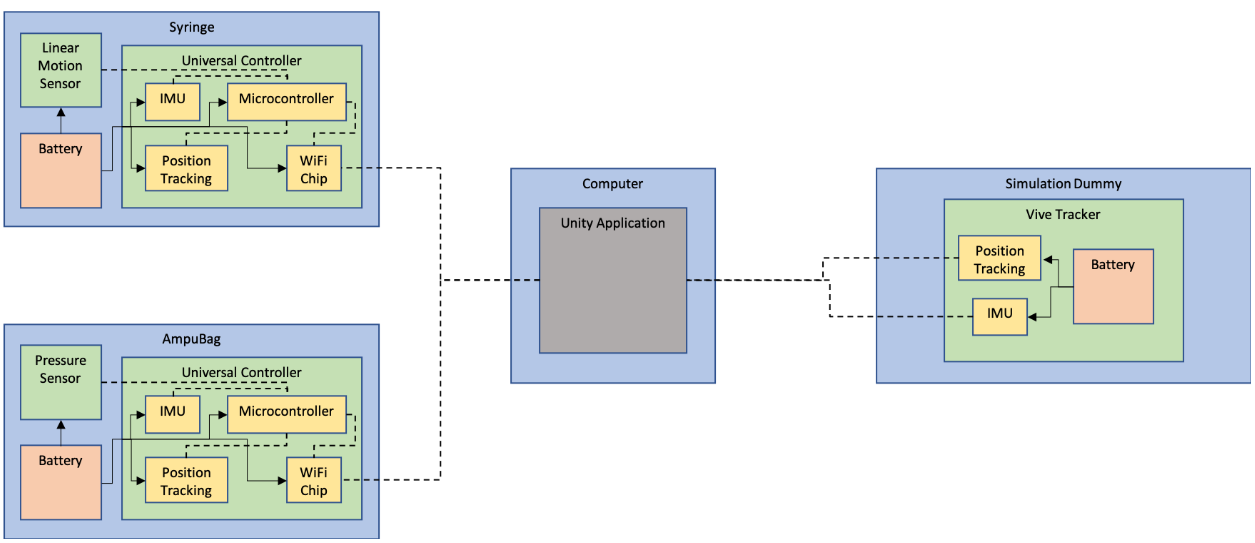
The need for improved medical tools and devices is a growing industry. Deloitte projected that the medical technology sector would experience a compound annual growth rate (CAGR) of about 15.9% between 2016 and 2021 [1]. While a majority of this industry focuses on devices for real procedures, because of the rising costs allocated towards training physicians, there is an increasing demand for simulated training for medical students. Recently, virtual reality training has been tested for testing the competency of EMTs. Mcgrath [5] concluded that simulation softwares are indeed effective for training and testing. However, he noted that the rapidly changing VR industry has made it hard for healthcare professionals to handle the various platforms available. Also, many proposed solutions are relatively high cost, so it is important to ensure that sensors and controllers are used efficiently. We plan to design a cost-efficient solution that would be easy for any general medical professional to use and train on.

High-Level Requirements List:

1) Sensors should relay correct measurements (distance and angle) with high fidelity to WiFi unit.
2) Virtual reality environment must respond correctly to sensor information with low latency from WiFi unit
3) Sensors and microcontrollers used should be low cost (total should be less than $50) as to not exceed expectations set by [5].

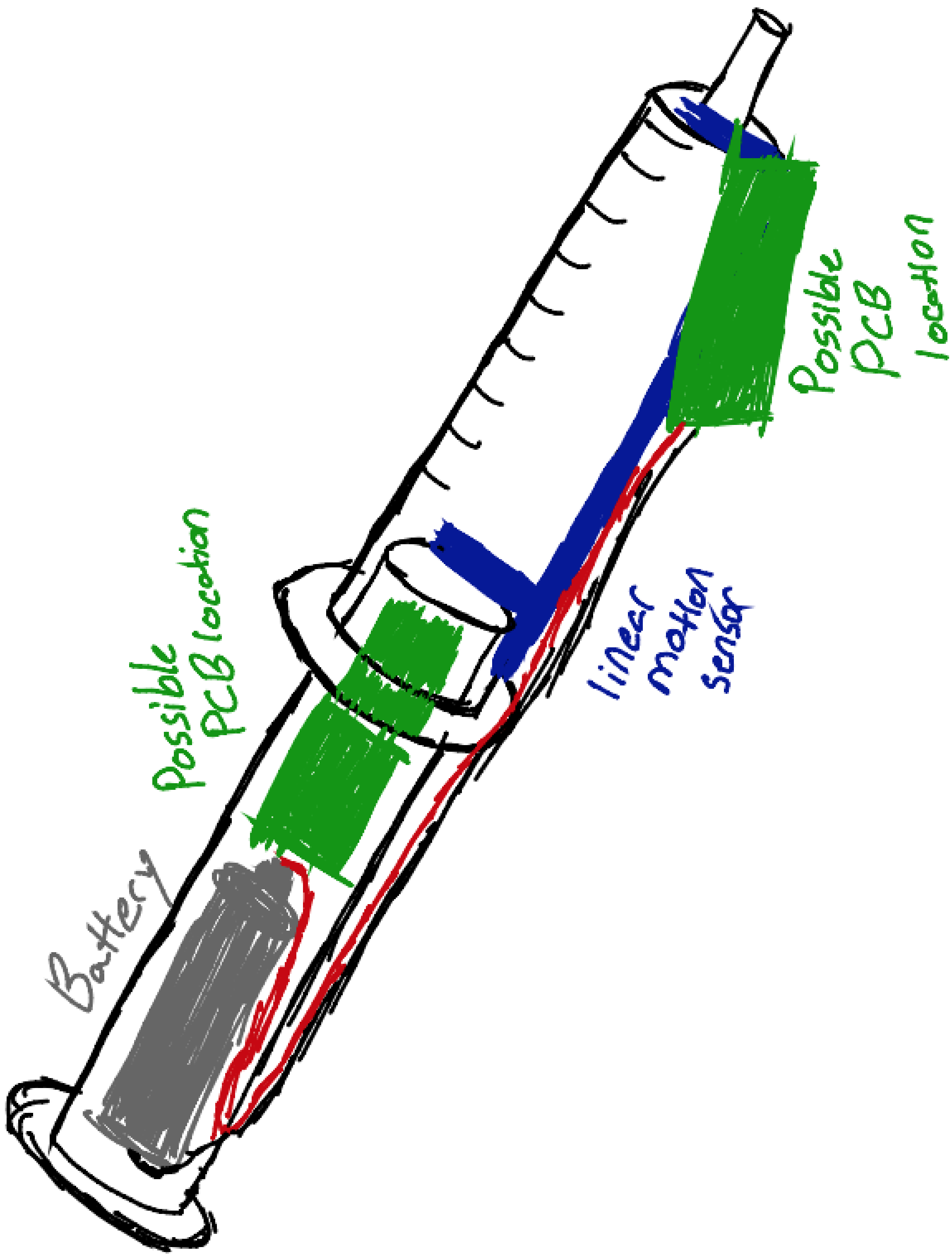
2. Design

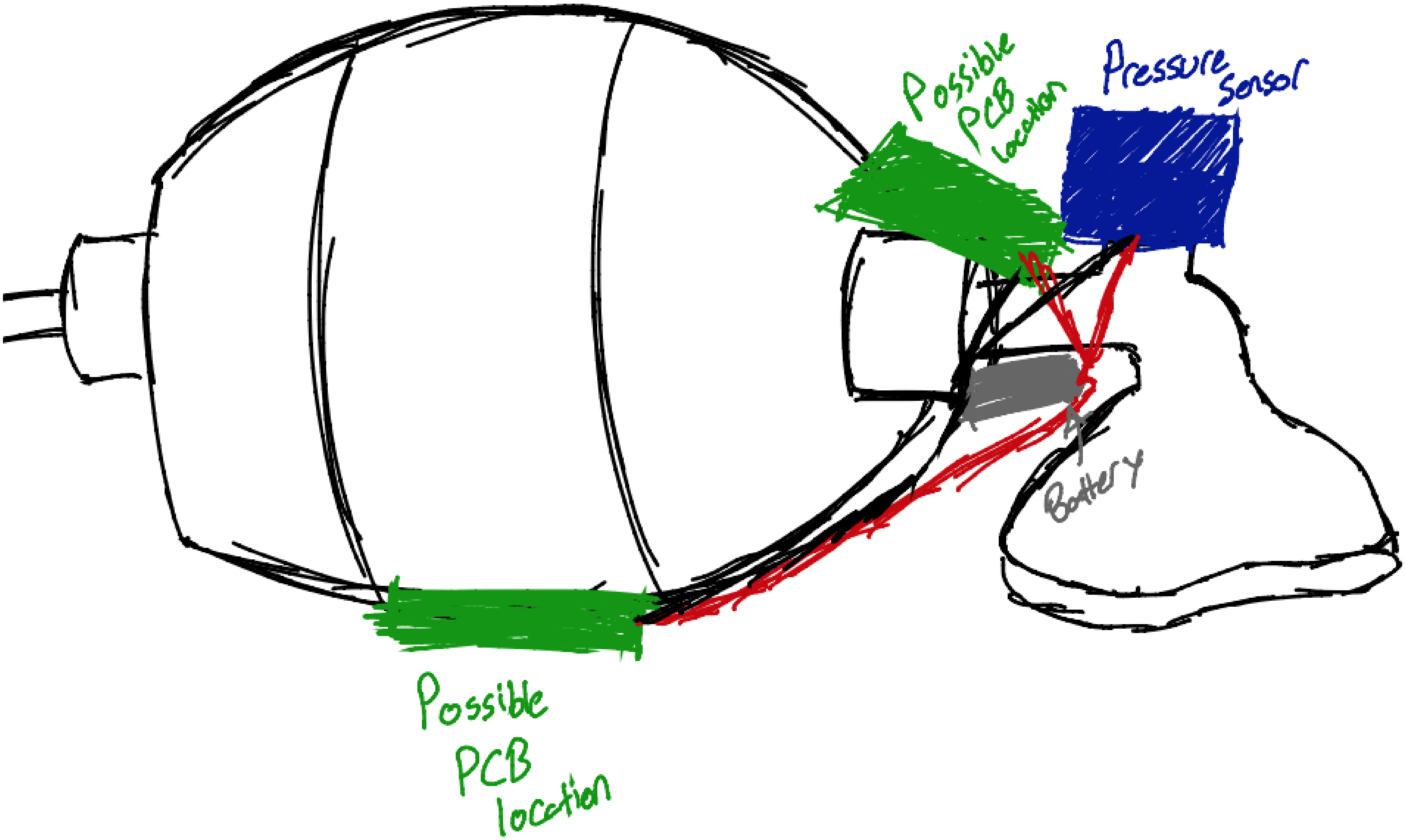
Block Diagram:



The 4 main blocks of the diagram are the Syringe, the Ambu Bag, the Computer, and the Simulation Dummy. Most of our physical design will focus around the Syringe and the Ambu Bag. WIth the Dummy and Computer there to show off how the controllers work. The Syringe and Ambu Bag will have their own built in sensors specific to their uses and then attached to each will be a universal controller that will measure more common parameters as well as reading the output from the specific sensor and transfering the data to the computer over wifi. In the diagram you can see the sensors all transferring data to the microcontroller which will then send it to the computer via the WiFi Chip. This fulfills our first requirement of the design. The second requirement is again fulfilled by the microcontroller and WiFi chip but also focuses more on the actual Unity Application that we will build. While the application is part of the block diagram it is really more of a black box of software that will be created with the requirement kept in mind. The third requirement will be met by the fact that our actual design doesn't require that much in the way of hardware. Batteries, Microcontrollers, IMUs, and most other sensors are cheap. The only expensive components will be the LEAP motion for position tracking and the Vive tracker simulation dummy which will be borrowed from HCESC so we won't have to factor in paying for those.

Physical Design:




Possible PCB location
Pressure sensor
Battery
Possible PCB location

In each of the diagrams I've drawn the physical devices in black, the external sensors in blue, the possible locations for PCBs in green and the battery locations in gray. I made sure to draw on multiple PCB locations in case either one of them doesn't work ergonomically, or if the PCB ends up being too large and we need to end up splitting it into multiple parts in order to get it to fit well. The goal will to be to hide all of our added equipment away from the common touch points of the medical devices, and where that is not possible make them as small and unobtrusive as possible.

Functional Overview:

Linear Motion Sensor: This will be built into the actual syringe itself and will output its reading to the I/O ports on the universal controller which will then go onto the microcontroller and transferred to the unity application. This is required to get a full readout of all the important parameters of the syringe. The linear motion sensor will be attached to the plunger and it will measure how much the syringe has been pressed. Then in the application we can measure how much fluid would be injected into the patient and at what rate and we will be able to tell if this is the correct amount, as well as giving visual feedback to the user on what exactly they are doing.

In order to actually create this we will have to either find a linear motion sensor online that fits into our syringe and can run at the voltages we use, or we will have to modify some other device like a small digital caliper. It is likely that we will have to cut open the syringe and have a hole that the linear motion sensor can stick out of so that it doesn't get in the way of the plunger. We will most likely be running this sensor off of the built in battery hidden in the syringe itself.

Pressure Sensor: This will be attached to the Ambu Bag itself and will output its readings to the I/O ports on the universal controller much the same way as the linear motion sensor does in the syringe. This will then read how much air pressure is being exerted into the patient's lungs, some actual Ambu Bags actually have pressure monitors attached to the bag which can show the doctor if they are sending the correct amount of pressure, but these won't work in VR so a digital solution must be created. The actual integration of the pressure sensor should be easy enough as long as we are able to source an Ambu Bag with a built in location for the standard pressure monitor to be attached. This sensor will allow us to track both how much air the user is putting into the patient's lungs as well at what intervals they are squeezing the bag.

IMU: This will be built into the universal controller itself, it will be powered by the battery attached to the medical device and will send its data to the microcontroller. It will be comprised of a built in accelerometer, gyroscope, and magnetometer in order to consistently track the orientation of the chip and its attached device. IMUs are a crucial part of tracking the device and is used in pretty much all VR controllers so they will be used in all of our controllers as well.

Position Tracking: We aren't 100% sure what we are going to use for position tracking, there are a couple options on the table and we need to do some hands on tests on a couple before we come to our final decision. The current plan is to use LEAP motion to track the controllers. This is nice because it removes the need for extra sensors on the controller itself and instead puts them on the VR headset. Otherwise we will have to find a way to track the controllers using the built in IR camera system built into the VIVE. Then we would have to create a chip and some IR LEDS and power them with the inbuilt battery and then send the data to do the microcontroller. Position Tracking is also an important requirement as it allows us to know exactly where our controller is in 3D space and will be needed to accurately map our controllers into our application.

Microcontroller: This will be powered by the built in battery and attached to the universal controller. It will have several inputs from the I/O ports, the position tracking, and the IMU and will output data to the WiFi chip. It will read all of these signals and do any necessary computations on them and then tell the WiFi chip to send them to the computer where they will be used in the Unity application. The microcontroller will be in charge of controlling the chip itself along with all its peripherals and signals going to and from the device.

WiFi Chip: This will be attached to the microcontroller and will be powered by the built in battery. Its job will be to take whatever the microcontroller is outputting and send it to the computer so that it can be used in the unity application. This will be the main form of communication between our controllers and the unity application.

Universal Controller: This will either be one large chip or several smaller chips and will contain all the components necessary to make a medical tool into a VR controller. It will attach to the controllers battery and will track all of the default parameters that every VR controller needs and will process and send that information to the application. It will also have several I/O ports that can be attached to external sensors if needed. The idea is that we will be able to design one chip that will cover most of the functionality necessary for creating a VR application, then we will be able to easily reproduce several copies of that chip and reuse them for several devices. The universal controller currently refers to the IMU, I/O ports, Microcontroller, and WiFi chip, and may refer to the position sensor depending on which method we end up using.

Vive Tracker: This is an off the shelf stand alone unit that can track position and rotation and send its data to the unity application. It is rather large and expensive so we will most likely not be using it on our controllers themselves but instead borrowing a few examples to attach to the simulation dummy in order to give the user some haptic feedback while our controllers. They will be attached to a specific point on the dummy and will be mapped to a model in VR the model will then follow the dummy so that your visual and touch senses will line up properly.

Unity Application: This is a piece of software instead of a physical device like the rest of the blocks. Within it we will create a small virtual simulation tutorial that will walk the user through a simple medical procedure that uses our controllers and dummy. It will be required to accurately display the locations of the devices as their data is sent to them via the microcontrollers and wifi chips. This will be a large part of our demo and will be used to make sure that we actually meet our requirements. People will be able to use the application to test out if the controllers work properly and track accurately in VR.

Block Requirements:

Linear Motion Sensor:

- Can track simulated volume to within 10% of the actual value (actual syringes can be as high as 5%).
- Doesn’t drift outside of the simulated range after multiple uses or prolonged usage.

Pressure Sensor:

- Must be able to accurately signal when the user squeezes the bag so that interval can be tracked.
- Must measure a limit of at least 0-60 cmH2O (range of an actual bag) and an accuracy of at least 2cmH20 (based off actual gauge).

IMU:

- Should be able to correctly measure angle of device with respect to all 3 axes.
- Data should be connected correctly to controller.

Position Tracking:

- Must accurately track position to where the user cannot perceive inaccuracies.
- Must not drift while being held still.

Microcontroller:

- Must consistently keep a low latency of <20ms this includes Wifi chip.

WiFi Chip:

- Must be able to collect data from sensors correctly.
- Must be able to send data to receiver at VR headset.

Universal Controller:

- Must be easily replicable in order to make several copies.
- Must be cost efficient <$50.

Vive Tracker:

- Should follow position of device both laterally, vertically, and horizontally with high fidelity.

Unity Application:

- Must follow well accepted 3D models for various anatomical components.
- Must have low latency response from sensors end to end latency of around 20ms.

Risk Analysis:

For Linear Sensors, they are used in many applications, including gear measurement tension testers, coordinate measuring machines, laser scanners, and calipers. They can also be used in servo-controlled motion systems to provide accurate information. Position sensors are also deployed in hazardous areas such as oil exploration rigs, natural gas utilities, chemical and pharmaceutical manufacturing facilities, wastewater treatment plants, and large-scale dry cleaning operations. All of these have in common that there is a flammable gas or liquid. Linear sensors, as electromechanical devices, have the basic risk of sparking and can cause explosions at these locations.
For other products like Microcontroller and IMU, Products used in their final product applications can suffer from system or random failures. Functional safety standards are designed to help influence the potential risk of reducing physical injury and property damage to people caused by such failures. But first, these risks need to be identified for final product applications and the impact of those risks analysed in order to take appropriate measures to reduce the likelihood of their occurrence and to reduce their impact where those risks do occur. Overall security perspective.

3. Ethics and Safety

We must ensure that the real environment that the trainee is in completely safe as the user would not be able to see what is actually around them. For this, we think that the best course of action is to not use blunted tips on the laryngoscope device (this is fine because the actual device does not need to cut into anything). Also, we plan on having the actual devices tethered to the work station that we develop. This way, the user would not be able to move far away from the dummy while they are essentially blinded to their actual environment. Overall, the design does not have too much room for safety hazards, but we find it important to address the ones that do exist.

Ethically, VR in medicine has had a history of contentious points. Historically, many medical professionals have raised concerns regarding the accuracy and verisimilitude of simulations such as ours [6]. Also, many professionals have raised concerns regarding biases in VR research applications stating that researchers would be motivated to make advancements in their simulations at the expense of patients that would be later impacted by this technology. We believe that our project will not violate any of these ethical concerns as our device will be used as ancillary training tool for students who will have to perform the procedure on cadavers regardless. Essentially, this tool would not be a replacement for practicing on an actual body, but would rather be a training tool used before actually carrying out the procedure. Also, we plan on using well-accepted 3D models of a human larynx/throat in our simulation to ensure adequate verisimilitude.

References:

[1] S. Burrill, “2019 US and Global Health Care Industry Outlook,” *Deloitte United States*, 25-Jan-2019. [Online]. Available: https://www2.deloitte.com/us/en/pages/life-sciences-and-health-care/articles/us-and-global-health-care-industry-trends-outlook.html. [Accessed: 07-Feb-2019].

[2] S. Bachai, “Synthetic Human Cadaver To Be Used By Medical Students Instead Of Actual Humans,” 13-Nov-2013. [Online]. Available: https://www.medicaldaily.com/synthetic-human-cadaver-be-used-medical-students-university-arizona-instead-actual-humans-262768. [Accessed: 07-Feb-2019].

[3] M. Maccall, “The Secret Lives of Cadavers,” *National Geographic*, 29-Jul-2016. [Online]. Available: https://news.nationalgeographic.com/2016/07/body-donation-cadavers-anatomy-medical-education/. [Accessed: 07-Feb-2019].

[4] “Laryngoscopy: Purpose, Procedure, Types, And Complications.” [Online]. Available: https://www.webmd.com/oral-health/what-is-laryngoscopy. [Accessed: 07-Feb-2019].

[5] J. L. Mcgrath, J. M. Taekman, P. Dev, D. R. Danforth, D. Mohan, N. Kman, A. Crichlow, and W. F. Bond, “Using Virtual Reality Simulation Environments to Assess Competence for Emergency Medicine Learners,” *Academic Emergency Medicine*, vol. 25, no. 2, pp. 186–195, 2017.

[6] L. Whalley, “Ethical Issues in the Application of Virtual Reality to the Treatment of Mental Disorders,” *Science Direct*, pp. 273–287, Mar. 1995.