FANtastique Inc.

Design Specifications Document

For

FANTOM

AIRFLOW OF THE FUTURE

Team Members:

Khartanovich Karina	CEO
Jehaan Jacob Joseph	CCO
Ardavan Mohseni-Javid	CFO
Shafin Rehman	CTO
Roy Zhao	CIO

Issue Date: July 7, 2019

Revision Number: 1.0.0

Abstract

FANtastique Inc.'s product, FANTOM, is an automated fan that is capable of tracking any individual near its vicinity automatically and provide cooling constantly by changing its direction toward the individual whether they are stationary or moving. We at FANtastique Inc. believe that our fan product will provide an affordable and environmentally conscious alternative in the fan industry with a low power consumption compared to other fan products.

The fan will be fully automated with features such as tracking individuals within its perimeter through thermal sensor, providing different levels of cooling intensity based on the user preference, ability to turn off when no one is present inside the room to save energy, and an accompanying app that will provide the user with more control of the fan and its custom features.

The information presented in this document will address the details of the design of the fan in terms of its computational, Electrical, Mechanical and Structural components. Furthermore, the requirements that were addressed in the Requirement Specification document will be discussed as to how they will be incorporated into our design of the prototype. In addition, an appendix for our testing procedure and validation as well as an appendix for user interface design is attached to this document. These will delve deeper into our test plan for the device and its user interface features.

Table of Contents

List of Figures **4**

List of Tables **5**

Glossary **6**

1. Introduction **7**

1.1 Background 7

1.2 Scope 8

1.3 Intended Audience 8

1.4 Design Classification 8

2. System Overview **10**

3. Computational System Design **12**

3.1 Hardware Design 13

3.1.1 Stepper Motor Control 13

3.1.2 DC Motor Control 15

3.1.3 Thermal Sensor Module 18

3.2 Software Design 19

3.2.1 Object Recognition & Tracking 19

3.2.2 Motor Control 22

4. Electrical System Design **25**

4.1 Power Supply 26

4.2 Power Distribution 26

5.Mechanical and Structural System Design **28**

5.1 Mechanical Design 29

5.1.1 Fan Blades with DC Motor 29

5.1.2 Oscillation Unit (Stepper Motor) 30

5.2 Structural Design 32

5.2.1 Fan Mounting Structure 32

6. Conclusion **34**

7. References **35**

8. Appendix A: Test Plan **36**

8.1 Introduction 36
8.2 Test: User 36
8.3 Test: Electronics 37
8.4 Test: Software 38
8.5 Test: Mechanical 39

9. Appendix B: User Interface 42
9.1 Introduction 42
9.2 User Analysis 42
9.3 Graphical Presentation 43
9.3 Technical Analysis 47
9.5 Engineering Standards 49
9.6 Analytical Usability Testing 49
9.7 Empirical Usability Testing 50
9.8 Conclusion 51
9.9 References 52

List of Figures

- Figure 1.
- Figure 2.
- Figure 3.
- Figure 4.
- Figure 5.
- Figure 6.
- Figure 7.
- Figure 8.
- Figure 9.
- Figure 10.
- Figure 11.
- Figure 12.
- Figure 13.
- Figure 14.
- Figure 15.
- Figure 16.
- Figure 17.
- Figure 18.
- Figure 19.
- Figure 20.
- Figure 21.
- Figure 22.
- Figure 23.
- Figure 24.
- Figure 25.
- Figure 26.

List of Tables

- Table 1.
- Table 2.
- Table 3.
- Table 4.
- Table 5.
- Table 6.
- Table 7.
- Table 8.
- Table 9.
- Table 10.
- Table 11.
- Table 12.
- Table 13.
- Table 14.
- Table 15.
- Table 16.
- Table 17.
- Table 18.
- Table 19.
- Table 20.

Glossary

Name	Description
PoC	Proof-of-Concept
PWM	Pulse Width Modulation

1. Introduction

1.1 Background

The effect of global warming on our planet has grown to be an issue that has many people concerned and looking to reduce their carbon footprint. Many cooling and air conditioning systems that are commercially available consume a considerable amount of energy. This problem is amplified in places where the climate tends to be both hotter and drier throughout most of the year. In addition, most of the cooling systems require physical user interaction and input in order to perform basic functions such as turning on or changing the intensity, which often leads to people leaving their systems turned on throughout the day out of convenience. All of these functions can be automated however, through the use of sensors where the device will only operate if a person or multiple people are present in a room. A cooling system of this nature will be more efficient in power consumption and be both more environmentally friendly and convenient to use.

We at FANtastique Inc. believe that our tracking fan (referred to as the device from here on) can be the next step in improving the cooling systems in the industry. Our device is an automated fan that is capable of turning on/off by sensing people's presence within its perimeter or in a room where it is installed. As the individual moves around in between rooms the fan in the room where the person is not present will turn off, and the fan where the person just entered will turn on. This feature is implemented to optimize energy consumption, and prevent fans from wasting power when cooling in a room is not needed. In addition, the fan will also be able to track the person in the room and rotate to provide constant cooling without the user manually changing its direction as they move in the room. Accompanying the fan is an app that can provide the user the ability to control the fan wirelessly, as well as change modes or intensities. Through the app, the user can control the speed of the fan, or its position, or simply turn it on and off.

1.2 Scope

This document will be detailing the design requirements that will be present in the PoC demo. The features and functions needed to realize the basic operation of this device will be covered under software and hardware design as well as systems for electrical, mechanical and structural design. The specifics of the software will explain how the hardware of the apparatus will communicate with one another. The electrical system will detail how power distribution between the parts. The mechanical system will incorporate aspects such as the motor and fan blades as well as the structural design of the fan. The sustainability of the product will be considered throughout the design process and safety features applicable to each system will be outlined. This document will not be going over the more advanced features that may be implemented in the future, nor will it explain the technology or the methodology that will be used to implement these said features.

1.3 Intended Audience

The target demographic for our device is for those who are conscious about their impact on the environment and would like a cooling solution that minimizes that concern. The intended use of our device is to be positioned in either a room or multiple different rooms. The installation will be simple and quick, and the set up can be completed through the companion application wirelessing. If multiple fans are to be used, they will all be able to communicate with one another. The user can then customize its operation with different modes and profiles, optimizing the airflow and also consuming less energy.

1.4 Design Classification

The project will be divided into two phases: the PoC Phase and the Prototype Phase. The PoC Phase will involve the design, testing, redesign and implementation of a working device that will showcase all of the required functionality in a basic or limited form. The reason for this is to prove the viability of the system before taking it further into the prototype stage.

The Prototyping Phase will implement all it's final features and create a marketable product for showcasing.

In order to accommodate the multiple redesigns that will inevitably occur during the PoC Phase, the system will be designed to be modular so that components and subsystems can we swapped out with relative ease. The following flowchart in Figure 1 will detail the design and implementation process for the PoC Phase:

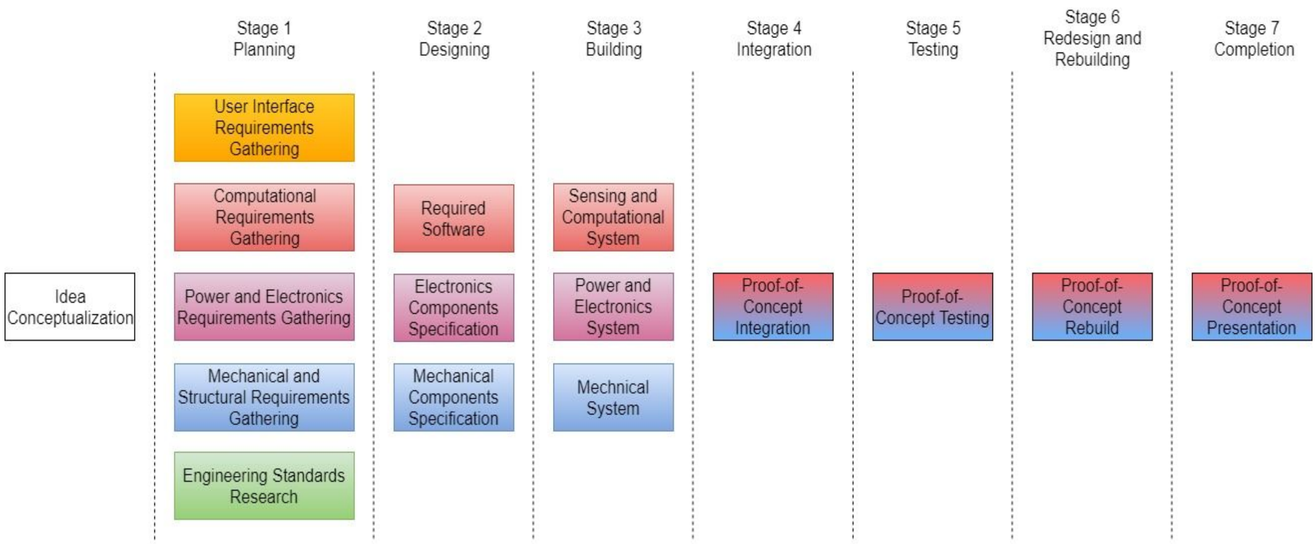

Figure 1.

We are currently in the final steps of Stage 2 (Designing) with this document serving as the culmination of that stage. Shortly, we will be preparing to move onto Stage 3 (Building). The materials and parts that will be incorporated in our PoC have already been gathered, and will be assembled in accordance to the designs and schematics that will be outlined in the next sections of this document. As a precaution, all the parts have been tested to ensure that do perform their own respective functions prior to any modifications being implemented.

2. System Overview

Our entire system can be simply visualized with the overview shown in Figure 2. While it shows the system in its entirety, only the features that showcase the basic functionality of our device will be implemented into the PoC phase.

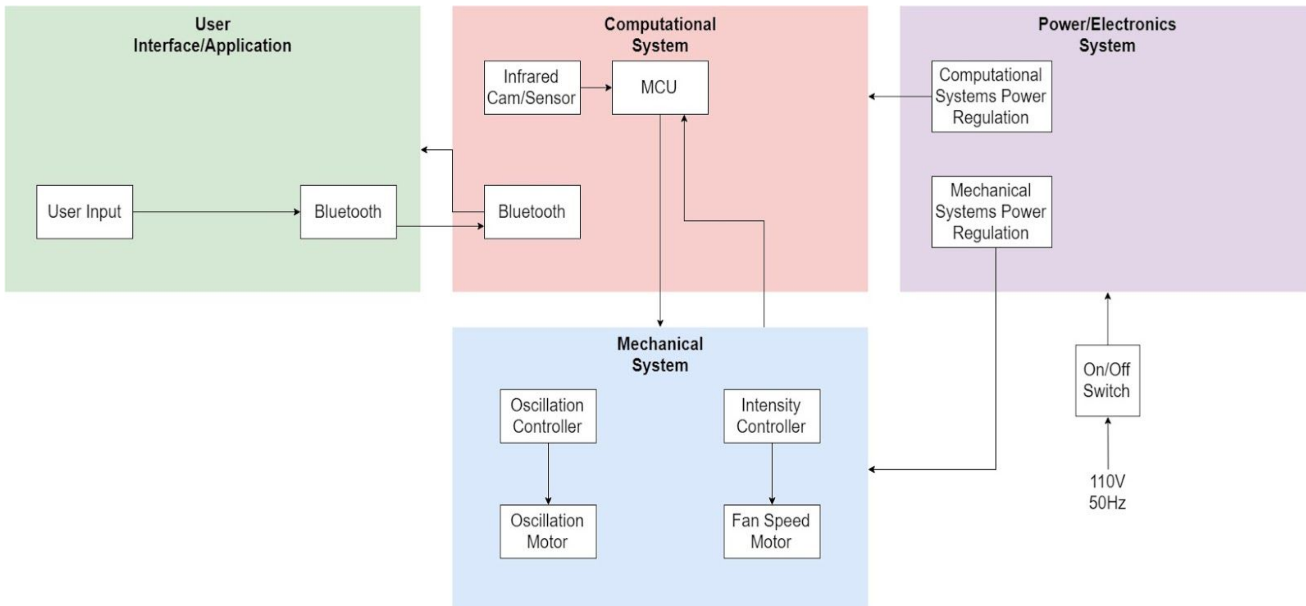

Figure 2.

These features include the ability to track people and adjust airflow to aim at the person, adjust the intensity of airflow depending on how close or far the person is, and having a minimalistic on-device user interface. The PoC device will be barebones and have both motors to show movement of the motor. It will be mounted on a wooden base for stability, which will also house the user interface. Computing electronics will be mounted on a breadboard.

The final prototype will have all the above features, but it's computing electronics will be on PCB, and it will be encased in a plastic cover with a grill over it's blades. It will also have the additional feature of an app that will allow the user to control the fan wirelessly. The device will also have additional power saving features, such as being able to turn on and off or slow down fan speeds depending on whether someone is in the room or not. It will also have different states that will turn off various electronics and functions based on what is needed and what isn't.

3. Computational System Design

The computational system includes all the software and hardware that is required to achieve object discovery, tracking, dynamic fan speed, and manual control.

Design ID	**Design Requirement**
Design Req 3.1	The computational unit must be able to control/ communicate with the IR sensor, the stepper motor and the fan motor.
Design Req 3.2	The fan must be able to identify objects that have temperature with in 28 °C and 34 °C.
Design Req 3.3	The fan must be able to find objects that have temperature with in 28 °C and 34 °C when scanning the environment.
Design Req 3.4	The fan must be able to track objects that have temperature with in 28 °C and 34 °C if that is the only object within frame.
Design Req 3.5	The fan must be able to switch between tracking and non-tracking mode.
Design Req 3.6	The fan must be able to respond to fan speed change within operating range.
Design Req 3.7	When the feature is enabled, the fan must be able to dynamically change speed based on the distance between itself and the target.

Table 1.

3.1 Hardware Design

3.1.1 Stepper Motor Control

List of Components used for the hardware design:

- Stepper Motor (12V, 0.4A)
- EasyDriver (7-30V, 0.15-0.75A)
- Power Supply (12V, 0.4A)
- Arduino Mega 2560

An Arduino Mega 2560 board is used to control the motor which rotates the fan base. In Figure 3 below the schematic for the Hardware Design is shown including Arduino Mega, Stepper Motor and the driver for the motor. The motor can not be directly connected to the power supply and Arduino. Hence, the driver is used to get the best performance, as if the Stepper Motor connected directly to the Arduino it will not be able to draw enough power.

From Figure 3, it can be seen that the Stepper Motor has four connections. Each coil has two connections, therefore we connect coil A to A+ and A-, and coil B to B+ and B-.

The EasyDriver has an adjustable current control from 150mA/phase to 750 mA/phase. Since the Stepper Motor has a limit of 0.4 A we will set the current limit on the driver to 0.4 A to protect the motor.

Arduino will be programmed to control the speed of the Stepper Motor, so that the sensor has enough time to scan the environment and capture any users. Arduino will also control the DC Motor speed, which is responsible for the fan blades rotation speed. In beta phase of the project, the speed will be changing depending on how close the user is to the fan (air flow intensity). User will be able to change the intensity manually as well using the phone application implemented in beta phase of the project.

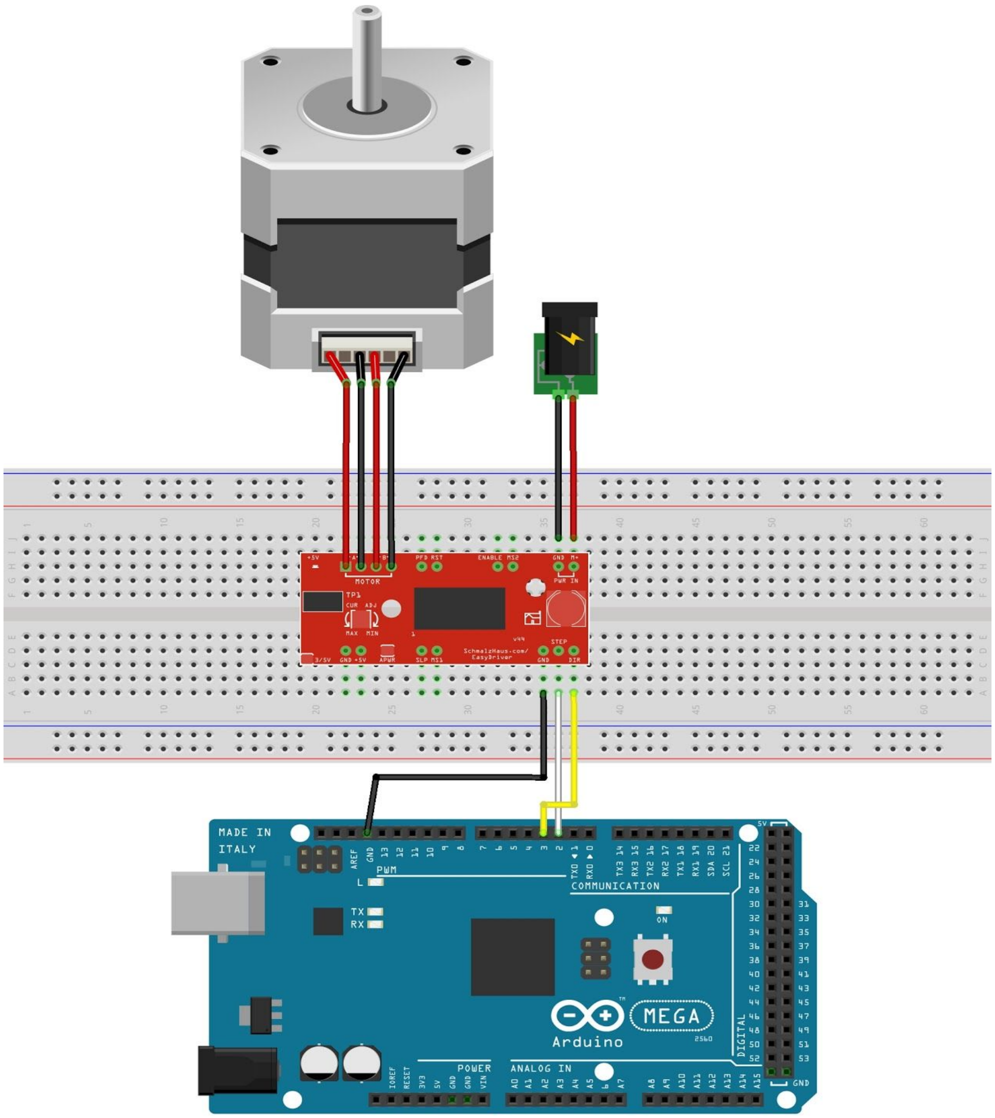
MADE IN
ITALY
COMMUNICATION
POWER
ANALOG IN
DIGITAL
MEGA
Arduino
MOTOR
STEP
DIR

Figure 3.

3.1.2 DC Motor Control

For our PoC, we opted to go with a 5V and max. 5W DC fan. In addition, the stock fan has a built-in PWM circuit to control the intensity of the fan. This built-in circuit has a pushbutton that increases the current supplied to the motor with each push, up to 3 increasing speeds. The current speed is stored as a state in it's built in memory module. The schematic of this circuit is shown below in Figure 4:

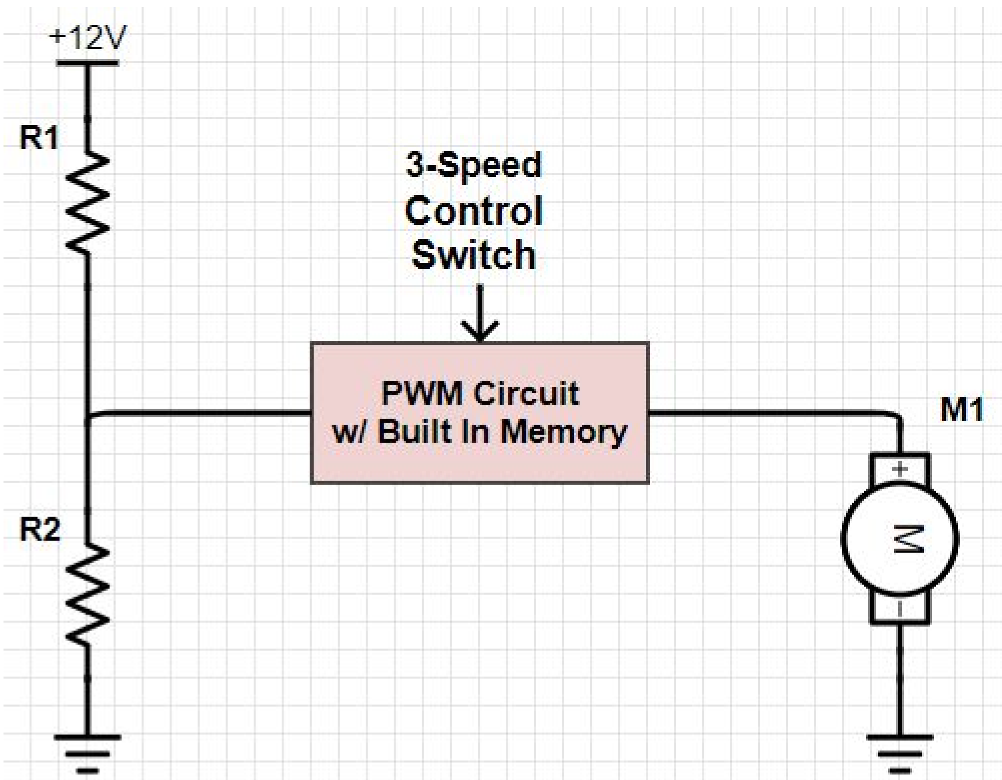

Figure 4.

However, given that our fan uses a slider-type feature to be able to smoothly increase or decrease its speed, this circuit will be modified for our device to allow the user to smoothly increase or decrease the speed of the fan.

It will be connected to the Arduino MCU through an NPN MOSFET to act as a PWM, with the Arduino controlling the pulse duty cycle of the current supplied to the MOSFET, as seen in Figure 5 below:

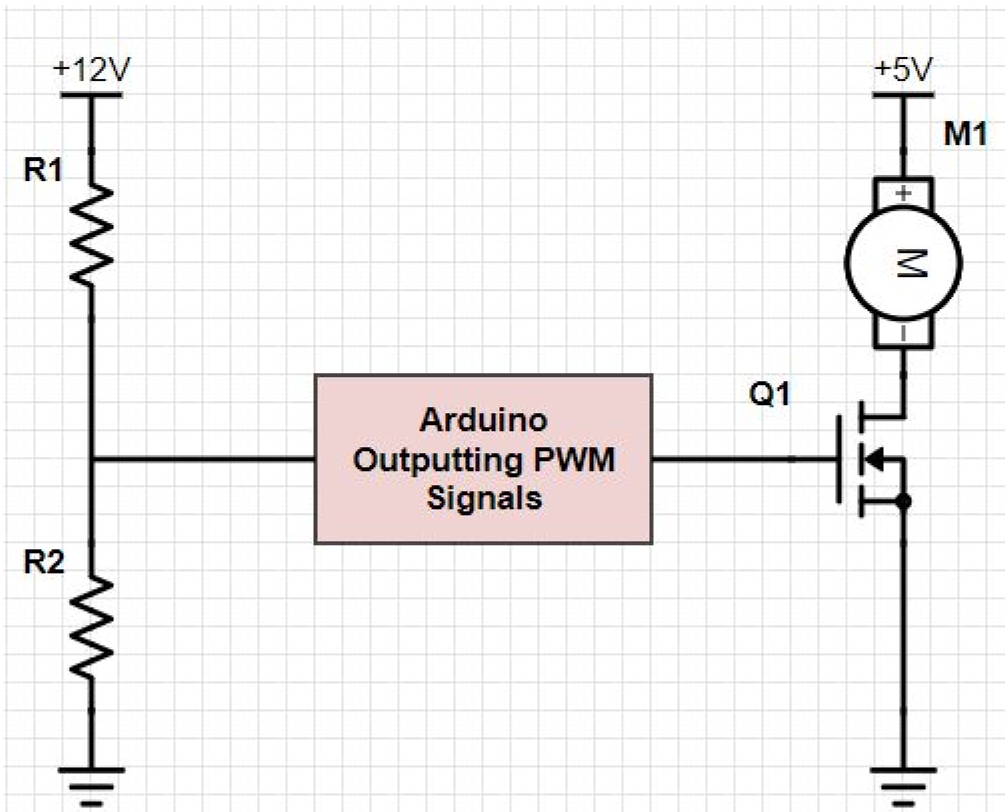

Figure 5.

The maximum fan speed will be set by the user. If Dynamic Speed mode is turned on, the fan will scale it's intensity to the temperature of the room and the distance of the person by modulating the duty cycle of the Arduino output.

The current supplied to the fan motor will be 1A, with the minimum intensity being at a 10% duty cycle and the maximum intensity at a 100% duty cycle. This feature will further save power used by the device, as it has been shown that using a PWM is a much more power efficient method of controlling intensity that the method used by a regular fan.

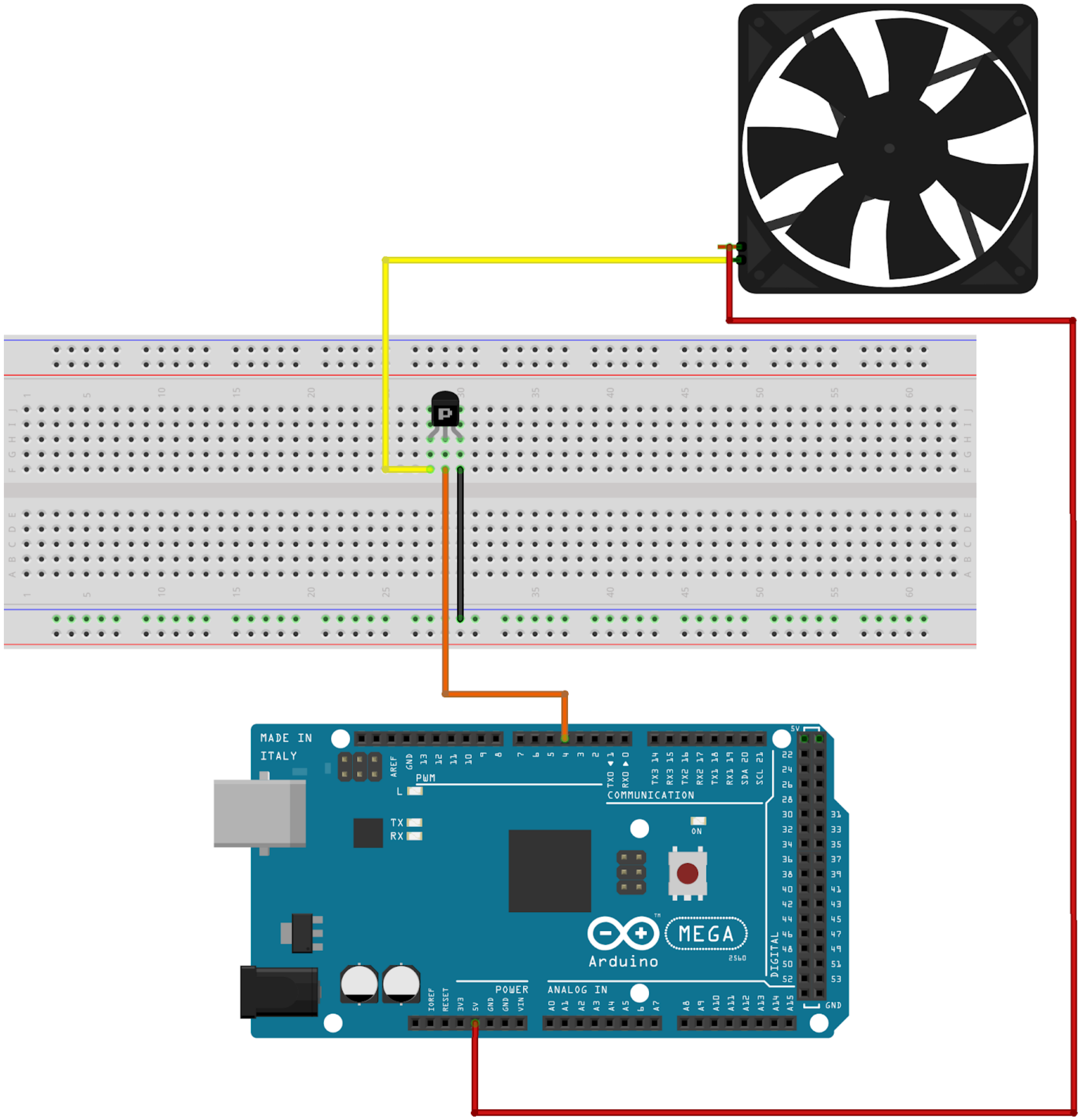

Figure 6.

Figure 6 shows the layout we will be using to assemble the circuit shown in Figure 5 (Isolated from all other components for clarity).

3.1.3 Thermal Sensor Module

For the alpha phase of the project, a Thermal Sensor (MLX90640) was chosen. The following sensor will detect users and send feedback to the Arduino board.

The chosen sensor will get power (3.3 V) from Arduino. Thermal sensor has four pins that need to be connected to the Arduino board. The first one is power and is connected to the 3.3V pin on the Arduino board. Second one is GND, connected to any GND pin on the board. Third and fourth are Clock and Data connected to pin 19 and pin 18 accordingly. This arrangement is shown in Figure 7:

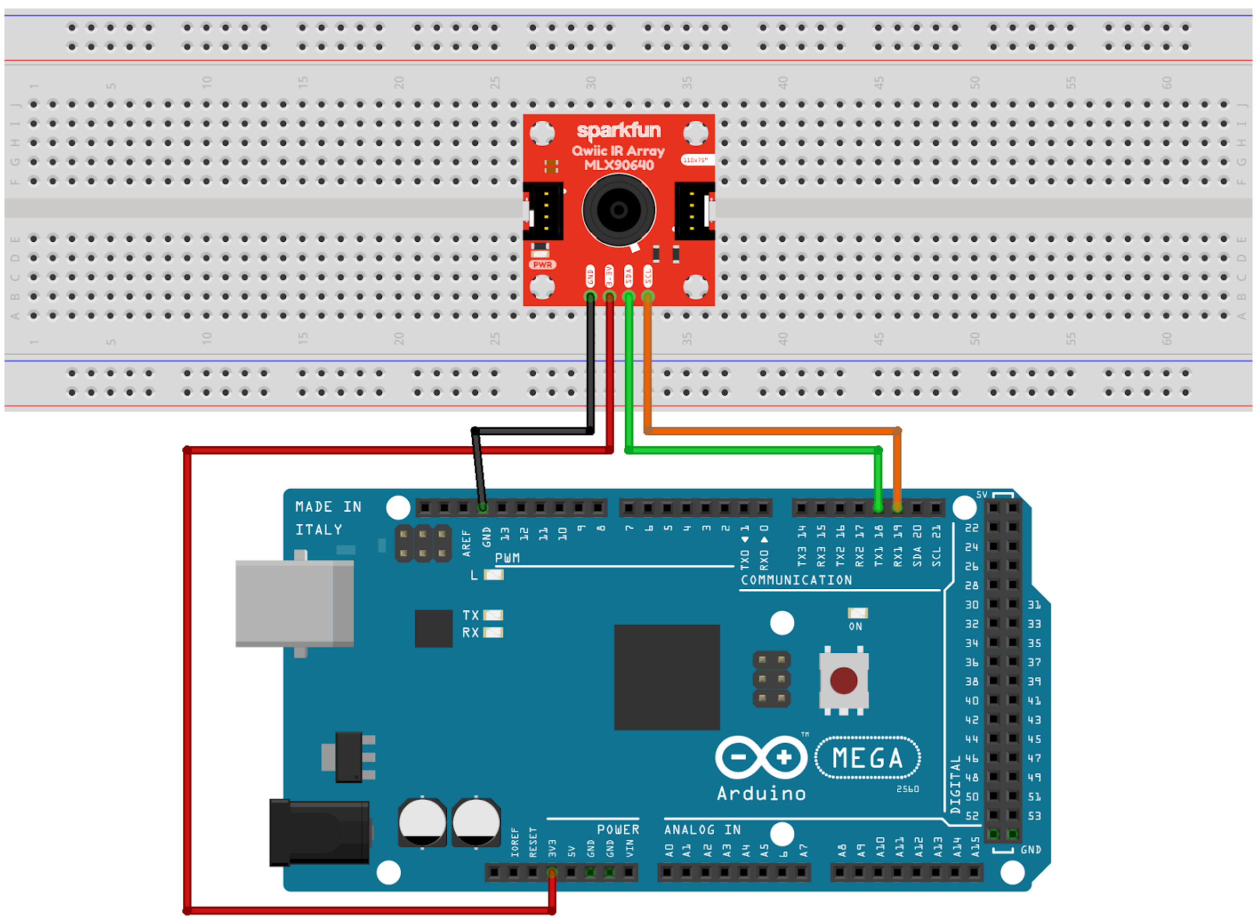

Figure 7.

3.2 Software Design

3.2.1 Object Recognition & Tracking

As one of the main features of the device, recognizing people in the scene is going to be what we focus on the most when building the software of PoC. After connecting the Thermal Sensor module (MLX90640) to the arduino board (detailed in hardware design), we can use the driver and examples provided by SparkFun.

This Thermal Sensor module, like most other cheap Thermal Sensor modules, operate by sensing the heat difference between the pixels and objects in front (within the field of view). In order to get the actual temperature of the object in view, we have to add the sensor reading on top of the ambient temperature of the board. We can get both of those information from the SparkFun driver.

The provided examples also do a few important setups for us. First is to confirm if we have the camera wired correctly. Second is to get the factory calibration data off the camera. This step is only needed to be done during the setup of the arduino board. Each MLX90640 module is factory and the bias of each pixel recorded. The example extract this information from the sensor at setup, then adjust the data coming from the sensor accordingly. This ensures we are getting the right reading during the main program.

By default, the camera will take four 32 pixel by 24 pixel snapshot every second The bottleneck for the frame rate is the connection that is used between the sensor and the board, the sensor does have the ability to output 64 frames per second. The output data stream are formatted in left to right, top to bottom arrangement. A visual of the image can be obtained by utilizing an open source project called Processing[7]. This is shown in Figure 8:.

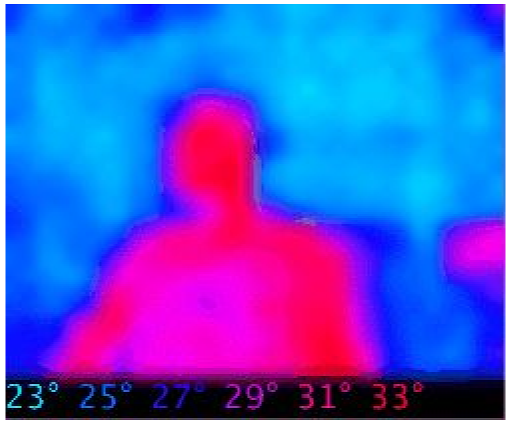

Figure 8.

This image shows that a person sitting in front of the image sensor is easily visible, but can not be used to identify the individual, which was one of the main concerns.

After getting the image, we now need to identify the objects. Since the FANTOM can not adjust tilt, the algorithm is going to be mostly focused on finding the object on the horizontal plane. Since the sensor have such limited resolution and arduino being a relatively low powered processor, it is not really feasible to run machine learning algorithm for object recognition. The prototype is going to mainly recognize its target though temperature.

The first thing the program going to do is to flatten the heat map from the camera to a one dimensional array by counting how many pixels in each column of pixels have values around the human skin surface temperature (28 - 34 °C). This range and its corresponding heat signature is shown in Figure 9 below;

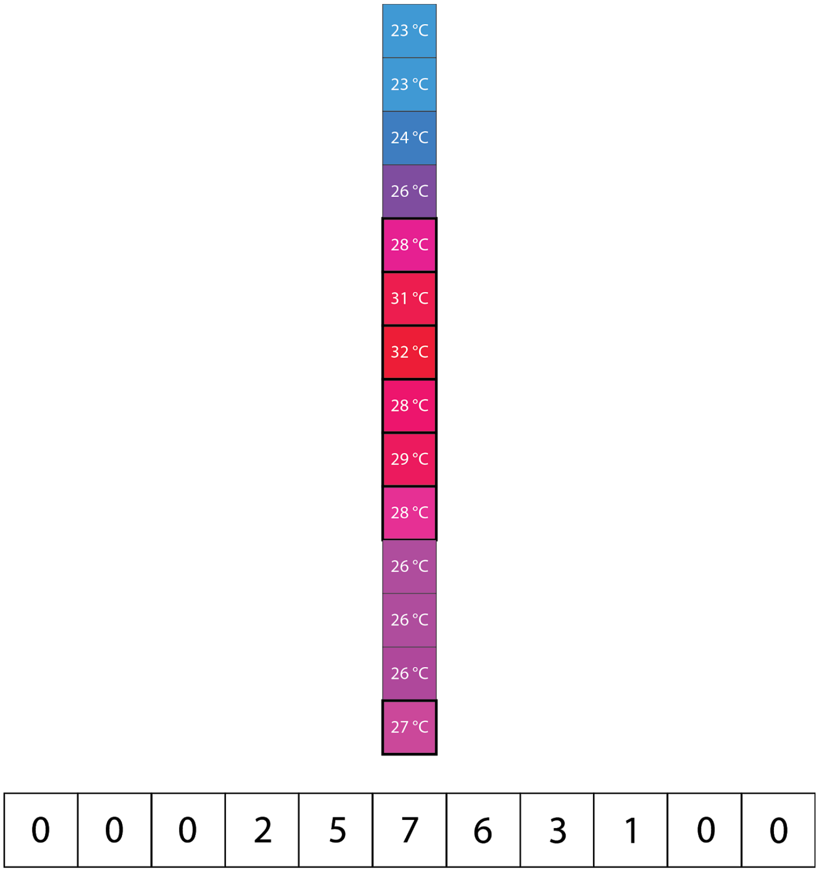

Figure 9.

After we count the number of pixels within range in each column, we should get an array like the image shown above. In this case where there is a local maximum that is significantly higher than the average (when there are only one person within the frame). Depending on if that local average is on the left or the right, the arduino will accordingly center itself on the local maximum.

If there is no local maximum value, but the total amount of pixel within the range is greater than a threshold (The ambient environment is around the temperature range), the fan will switch to manual control. Fan speed, direction and rotation will be depend on the buttons and mobile application (detailed in the UI document).

The device will go into scan mode when there are no local maximum values and the total amount of pixel within the temperature range is below the threshold for 5 seconds. The time delay is designed so the fan motor does not switch between on and off state within a short period of time, causing unnecessary stress to the motor. After 5 seconds, the device will scan the room twice. If a new heat source is found during the scan, the device will set that as the new target, otherwise, the device will return to the front position and go into sleep mode.

In sleep mode, the fan motor and the rotation motor will be powered off. The device will still take snapshot in the direction the IR sensor is pointed but at a reduced frequency to save power. When the sensor picks up a new target within the frame, the processor will wake up the motors and resume normal function.

A multi-target system is also under development. The current design is for the device to oscillate between the targets, but the specifics of this feature have yet to be finalized. The main issue is discovery and identification. We will update this in a future iteration.

3.2.2 Motor Control

There are two motors that need to be controlled by the processing unit, the stepper motor which controls the oscillation of the fan and the fan motor which controls the speed of the blades.

The stepper motor is responsible for carrying out the tracking feature. When the target is discovered by the sensor, depending on which side of the frame it is, the processor will issue turn commands to the stepper motor. Originally we have the idea of depending on how far away the target is from the center of the frame (measured in pixels), we will have a lookup table which turns the stepper motor the corresponding amount. After some testing we found that this method performed poorly when the target is moving, and result in a lot of unnecessary movements. We then decided to issue the move command as long as the target is within the center of the frame. We then widen the tolerance on center of the fame to be 5 pixel wide. This change is to combat jitter and noise from the IR sensor during operation.

Another concern when designing the operation of the stepper motor is that when turning the fan, there might be something blocking the way. This is often the case when using rotation on a normal fan. It will try turning even though there is something in the way. This usually cause the fan to make an unpleasant clicking sound. The advantage is using the stepper motor we chose is that the motor can be set up to send back the current angle as part of a feedback loop. This allows us to monitor when the motor get stuck as the Arduino will detect when the stepper motor reports an unchanging current position. In this case the processor will stop issuing the turning command. In a later prototype version, the device will notify when this happens though either an LED indicator or the mobile application. It would also have the ability to oscillate between a given angle (which can be set in the mobile application specified in the user interface document).

The fan motor controls the speed of the blades which translate to the amount of air getting pushed through the fan. There are two main things that would dictate the fan speed, the slider on the front of the fan (equivalent to the slider in the mobile application specified in the user interface document), and the distance between the fan and the target.

The slider on the fan does directly controls the duty cycle of the Arduino's PWM output. This design allows us to change the fan speed through other means. When the slider is moved, the signal will be processed by the processor unit, which will then change the PWM duty cycle going into the fan motor if the change is within the operating limits of the fan motor.

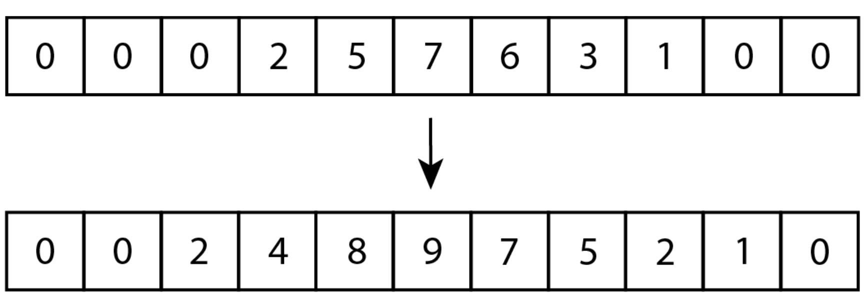

Figure 10.

When the device is tracking a target, the distance between the target and the fan is calculated based on how many pixels are within the human temperature range. When the amount increases (like shown in figure above), the fan will slow down in order to provide a consistent experience for the user. This feature can be turned off from either the switch on the fan or through the mobile application.

4. Electrical System Design

The Electrical section of our system consists of discussing the specification of the power supply we chose for our project and that how the power supply distributes power to each of the components in the system and how all of those components are connected to one another and to the fan as a whole. A list of design requirements for the Electrical section is mentioned in Table 2.

Design ID	**Design Requirement**
Design Req 4.1	The power supply must be able to provide adequate power to all the components that are connected to it inside the system, and voltage that is supplied to each component must remain stable.
Design Req 4.2	The voltage divider must be able to work properly and be able to reduce the power supply voltage of 12V DC to 5V DC for Arduino Mega and the Drive Controller board.
Design Req 4.3	Arduino Mega must successfully provide signal to both the Drive Controller board and the PWM to control the motors.
Design Req 4.4	Drive Controller board must be able to make the Stepper motor to rotate and function through its connection to the Arduino Mega and the Power supply.
Design Req 4.5	The stepper Motor must be able to maintain the fan’s weight and be able to rotate the fan 360 degrees smoothly.
Design Req 4.6	The PWM must be able to make the DC motor to rotate the fan’s blades through its connection to the Arduino and the power supply.
Design Req 4.7	The system power must turn on when plug-in to the wall.

Table 2.

4.1 Power Supply

Our team is using a power supply to provide power to the device. The power supply that we are using for this purpose can provide upto 2A current with 12V DC output voltage and 24W power to the device. Furthermore, it can convert AC to DC by using an internal rectifier, and reduce the 120V voltage input to 12V by using a transformer. Previously, the goal was to purchase a full bridge rectifier and transformer and integrate it into a circuit; however, in the end it was decided to use just a power supply instead as it encompasses both the rectifier and a transformer inside, and will provide our system with the power that is needed to operate the fan and its various components.

4.2 Power Distribution

The power supply needs to provide power to various components in the system. These components are the DC motor powering the fan's blades, the Stepper motor that is rotating the fan's main body, and the Arduino Mega which is sending signals to both Drive Controller board and the PWM.

The Power supply provides 12V DC voltage, which will be delivered to the drive controller board that is controlling the stepper motor for rotating the fan's body. However, for the Arduino Mega and the PWM (MOSFET) we utilize a voltage divider to reduce the voltage from the power supply down to 5V DC. The voltage divider will be placed in between the power supply and the Arduino Mega and the PWM. This enables us to control the DC motor from the Arduino Mega and the PWM to rotate the fan's blades at various intensities depending on the power delivered; which could be as high as 5W.

Arduino Mega itself, will be supplied by 5V DC after the voltage divider which in turn send signals and controls both the PWM and the Drive Controller board with the conjunction of the power supply. In this case the power supply directly provides the Drive Controller board with 12V DC and also provides 5V DC to the PWM after going through the voltage divider. Figure 11 below illustrates how the components are connected to the power supply and to the system as a whole.

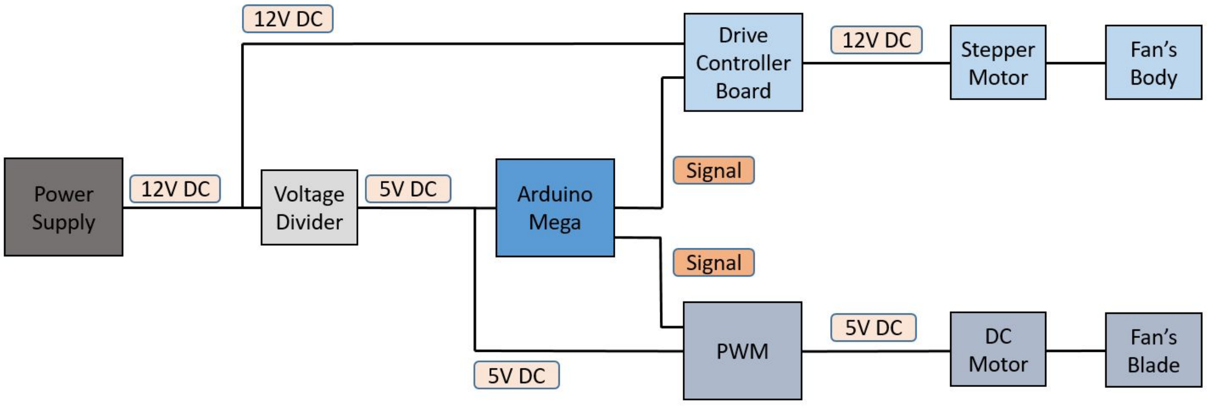
12V DC
Drive Controller Board
12V DC
Stepper Motor
Fan's Body
Power Supply
12V DC
Voltage Divider
5V DC
Arduino Mega
Signal
Signal
PWM
5V DC
DC Motor
Fan's Blade
5V DC

Figure 11.

5.Mechanical and Structural System Design

The majority of the mechanics used in the device involve motors. One motor is needed to rotate the fan blades with variable intensity, and another is needed for the oscillation and seeking features of the device. In both cases, DC motors were chosen for their ease of use when interfacing with microprocessors.

Design ID	**Design Requirement**
Design Req 5.1	The blade motor must be DC (Changed from Requirements Specification Document).
Design Req 5.2	The blade motor must be brushless.
Design Req 5.3	The blade motor must be able to spin the blades at a speed that matches or is higher than a regular desk fan of comparable size.
Design Req 5.4	The stepper motor must be able to support the weight of the fan structure (which will house the fan motor, blades and grating).
Design Req 5.5	The stepper motor must be able to rotate the fan structure in either direction.
Design Req 5.6	The stepper motor must be able to rotate the fan structure in 360°.
Design Req 5.7	The base mounting structure must be able to support the weight of the fan structure and the stepper motor.
Design Req 5.8	The base mounting structure must have adequate space to house the stepper motor.

Design Req 5.9	The base mounting structure must be stable enough to support the fan structure during oscillation.
Design Req 5.10	The base mounting structure must be heavy enough to resist changes in momentum (i.e. when the fan turns from one direction to another).
Design Req 5.11	The base mounting structure must house an on/off switch.

Table 3.

5.1 Mechanical Design

5.1.1 Fan Blades with DC Motor

As mentioned previously, the motor that will be used for the Proof-of-Concept will be a 5V, 5W DC motor. It will be a brushless motor to reduce the generation of heat and improve efficiency.

This motor was chosen due to the speed requirements that have been stipulated prior. A regular 14 inch desk fan spins at 1500 RPM on average. Given the size of our fan (4 inches), we opted to go with this particular motor due to the fact that it spins at ~2400 RPM at its maximum speed. This was calculated with the following equation:

Rated Maximum Torque = 2 Ncm

Rated Maximum Power = 5 W

$$Rotational\ Speed = \frac{60 \cdot Power}{2\pi \cdot Torque} = \frac{60 \cdot 5}{2\pi \cdot 0.02} = 2387.32\ RPM$$

This shows that the motor has an unloaded maximum rotational speed of 2387 RPM. Given that the fan blades weigh approx. 45 grams, we can consider their weight to be negligible when considering the moment of inertia during rotation at maximal speeds.

Figure 12 shows the Torque/Speed graph as was obtained experimentally. The graph was averaged to give a smoother curve:

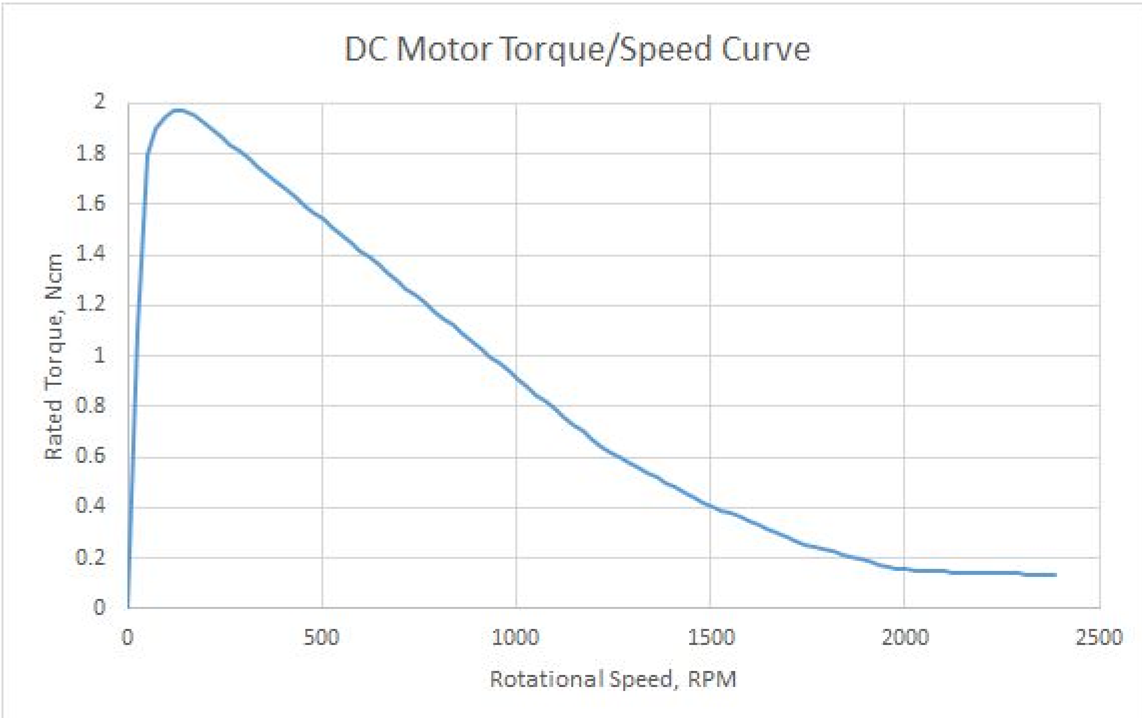

Figure 12.

5.1.2 Oscillation Unit (Stepper Motor)

The stepper motor will be mounted on a custom built base. This motor will be responsible for a) oscillating the fan when set to non-tracking mode, but also not set to point in a fixed direction, b) controlling the direction of the fan when set to tracking mode and c) in sweep mode while searching for human targets.

The motor is specified to have a torque of 26 Ncm, and is bipolar. However, given that our fan and motor have a weight of 160g, and that we intend the maximum turn speed to be 30 RPM, we will be restricting the torque to 0.31 Ncm, according to the following calculations:

$$T = \frac{(WK^2)\Delta N}{308t}$$

and

$$WK^2 = \frac{1}{2}WR^2$$

Where

$T = Required\ Torque,\ Ncm$

$WK^2 = Mass\ Moment\ of\ Inertia\ of\ load\ to\ be\ accelerated,\ Ncm^2$

$\Delta N = Time\ to\ accelerate\ the\ load,\ seconds$

$W = Weight\ of\ object,\ kg$

$R = Radius\ of\ cylinder,\ cm$

Firstly, we will calculate the Mass Moment of Inertia of the load to be accelerated:

$$WK^2 = \frac{1}{2}WR^2$$

$$= \frac{1}{2}(0.160)(6.35)^2$$

$$= 3.2258\ Ncm^2$$

Now, we will use this result to calculate the required torque:

$$T = \frac{(WK^2)\Delta N}{308t}$$

$$= \frac{(3.2258)(30)}{308(1)}$$

$$= 0.31\ Ncm$$

As can be seen from this result, the stepper motor can produce much more torque than is needed for the PoC. This is because we intend to reuse the stepper motor for our final prototype as well, which will feature a much larger, and thus heavier fan. It also gives us the opportunity to learn how to use the stepper motor for the PoC, so that we have no difficulty in using it for the final prototype.

5.2 Structural Design

5.2.1 Fan Mounting Structure

The fan mount for our PoC will be a scale model of the mount that we will use for our final prototype. This is due to the fact that the DC motor is small and lightweight compared to our idea for the final prototype. However, it needs to be large enough to house the stepper motor, and heavy enough to not move due to shifts in momentum of the stepper motor. The design of the base is shown in Figure 13 below:

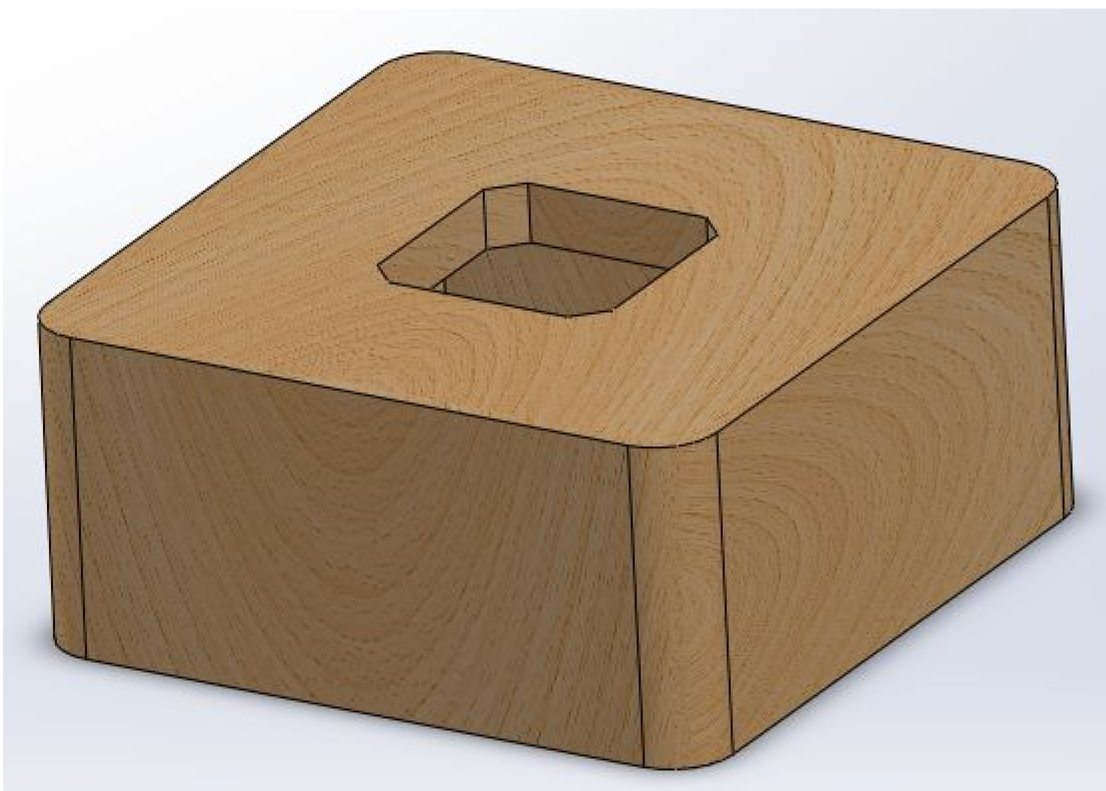

Figure 13.

Figure 14.

Figure 15.

As seen from the diagrams above, the mounting for the PoC will be made in wood. It will feature a 42.00 mm x 42.00 mm slot to house the stepper motor, and the mounting itself will 120.00 mm x 120.00 mm x 54.00 mm.

The weight needed to resist the impulse forces created by a change in direction of the stepper motor will be determined experimentally after construction of all the sub-components of the fan has been completed. We will achieve this by adding weight in the form of measuring weights until the fan structure does not react when the stepper motor changes direction.

6. Conclusion

We at FANtastique Inc. believe that FANTOM will be a groundbreaking product when it comes to fan and air conditioning systems in the market. With its affordable and low cost compared to other cooling systems in today's market and its energy efficient and environmentally friendly focus, FANTOM can be a great choice for many customers that are looking for new cooling systems for their homes or their offices; even more today as the awareness for global warming and the presence of carbon footprints in the industry is rising.
In addition, our goal is to deliver a product that will provide a convenient cooling system for the users, and it was this objective that motivated us to make FANTOM fully automated and

customizable for the users, such as being able to target the user and blow wind in the direction of the user automatically without any user input or turning off automatically when no user is present in the room to preserve energy.

The roadmap for the design of the FANTOM includes the completion of the PoC of the device by the end of this term, that includes making the fan to function and be able to rotate 360 degrees, and be able to track and respond when detecting individuals within its perimeter using the infrared sensor, and the ability to scan the environment when no target is present and subsequently turn off if no new target has been detected by the sensor. For the final product, in addition to having a fully automated fan device, we are aiming to deliver an app that accompanies the fan, in which the user can turn the fan ON/OFF, or be able to control the intensity, or even be able to customize some of the features of the fan to its preference through the app.

We at FANtastique Inc. hope that with this roadmap in mind we will be able to deliver a fan device that not only be able to revolutionize today's cooling system in terms of the convenience that presents for the user, but also is affordable and envion renmetaly conscious as well for today's industry standards.

7. References

[1] "Specifications | Bluetooth Technology Website", Bluetooth *Technology Website*, 2019. [Online]. Available: https://www.bluetooth.com/specifications/. [Accessed: 09- Jun- 2019]

[2] "CAN/CSA-C22.2 NO. 0-10 (R2015). General requirements - Canadian electrical code, part II", CSA Website, 2015. [Online]. Available: https://store.csagroup.org/ccrz__ProductDetails?viewState=DetailView&cartID=&sku=CAN/CSA-C22.2%20NO.%200-10 [Accessed: 09- Jun- 2019]

[3] "IEC 61558-1:2017 Safety of transformers, reactors, power supply units and combinations thereof - Part 1: General requirements and tests", IEC Website, 2017. [Online]. Available: https://webstore.iec.ch/publication/26261 [Accessed: 09- Jun- 2019]

[4] ISO/TR 12100-1:1992, Safety of machinery — Basic concepts, general principles for design — Part 1: Basic terminology, methodology.

[5] ISO 14120: 2015, Safety of machinery — General requirements for the design and construction of guards (fixed, movable).

[6] ISO 13852:2008, Safety of machinery — Safety distances to prevent danger zones being reached by the upper limbs.

[7] SparkFun Electronics ® (2018). *IR Camera Display*. [image] Available at: https://cdn.sparkfun.com//assets/parts/1/3/0/1/4/SparkFun_MLX90640_Thermal_Imaging_Camera-Demo.gif [Accessed 6 Jul. 2019].

8. Appendix A: Test Plan

8.1 Introduction

In order to ensure that the device meets the adequate standards for reliability and operation, it will be undergoing rigorous testing in various aspects. The test will be subjected to three main systems; the electronics, software and mechanical. The following sections will detail these respective systems as well as the system as a whole.

8.2 Test: User

The following tests will be conducted on the user interface and will cover the functionality and subsystems of the device as a whole.

Date:	**Test Name:** Active/Standby Transition
Test Description: Press the power button to turn the device on or off	
Expected Outcome: The device should power on with one press of the button, and then turn off with the second press.	
Actual Outcome:	

Table 4.

Date:	**Test Name:** Fan Speed Cycling
Test Description: Move the fan speed slider to various positions	
Expected Outcome: The speed of the fan's blades should either increase or decrease in accordance to the position of the slider.	
Actual Outcome:	

Table 5.

For the PoC phase, the device will only have the power button serving as the physical user interface component. The other various functions, such as manually setting the fan speed, will be done through the software and its respective testing procedures will be covered there. The prototype will incorporate a physical slider onto the body of the device to provide a secondary method of controller the fan speed.

8.3 Test: Electronics

The electronics testing will cover the power distribution throughout the system as well as several safety aspects to ensure safe operation.

Date:	**Test Name:** Regulated Output Efficiency
Test Description: Measure the voltage outputs at various stages and compare to the expected calculations. Monitor variations in both measurements.	
Expected Outcome: 12 volts DC should be present at the stepper motor, and 5 volts DC at the microcontroller and DC motor.	
Actual Outcome:	

Table 6.

Date:	**Test Name:** Current Draw Measurements
Test Description: Measure the current draw of various components of the system	
Expected Outcome: The entire system should not draw more than one amp of current.	
Actual Outcome:	

Table 7.

8.4 Test: Software

A majority of the functionality of the device is dependant on the software used to implement these features. This will be controlled through the companion application that will be part of the overall experience.

Date:	**Test Name:** Active/Standby Transition Within the App
Test Description: Toggle power to the fan through the software console	
Expected Outcome: The device should power on when an "on" command is issued through the software console, and vice versa	
Actual Outcome:	

Table 8.

Date:	**Test Name:** Fan Speed Cycling Within the App
Test Description: Control the fan speed through the software console	
Expected Outcome: The fan speed should increase or decrease when the appropriate command is issued through the software console	
Actual Outcome:	

Table 9.

The fan speed will be controlled through a debugging console for the PoC phase. In the Prototype phase, a physical slider and a companion app will be used to control its speed.

Date:	**Test Name:** Tracking of The Subject
Test Description: Power on the device. Walk within the vicinity of the device.	
Expected Outcome: The device recognizes that a person is within its field of view and starts tracking their position.	
Actual Outcome:	

Table 10.

Date:	**Test Name:** Tracking of The Subject from Standby
Test Description: Power on the device and allow it to go into Standby mode. Walk within the vicinity of the device.	
Expected Outcome: The fan should recognize that a person is within its field of view automatically wake up and start tracking their position.	
Actual Outcome:	

Table 11.

Date:	**Test Name:** Transition to Standby with no Human Presence.
Test Description: The user will leave the room or the vicinity of the device	
Expected Outcome: The device should perform two 360 degree sweeps to check that there is nobody present in its vicinity. Once done, it will transition to standby mode.	
Actual Outcome:	

Table 12.

8.5 Test: Mechanical

The test plan for the mechanical aspect of the device will test for the functionality of the various moving components of the fan as well as the overall structure and its integrity when subjected to different environmental conditions.

Date:	**Test Name:** Stepper Motor Functionality
Test Description: Making stepper motor rotate the fan 360° degrees for multiple runs	
Expected Outcome: The fan's main body should rotate smoothly for every 360° run both clockwise and counterclockwise	
Actual Outcome:	

Table 13.

Date:	**Test Name:** Fan Movement Inside the Mounting Structure
Test Description: After the fan is attached to the stepper motor inside the mount, make the stepper motor to rotate the fan multiple times.	
Expected Outcome: The mount should not slow or block the movement of the fan inside the mount, and the fan should rotate smoothly	
Actual Outcome:	

Table 14.

Date:	**Test Name:** Stepper Motor for Mounting Structure
Test Description: Try to fit the stepper motor inside the mounting structure, and place the fan on top and rotate the fan multiple times	
Expected Outcome: The mounting structure will fit the stepper motor perfectly inside itself and will maintain the weight of the fan on top, and as the fan is rotating, the mounting structure will maintain its position and it will not move from the motor and the fan's momentum due to the change of direction	
Actual Outcome:	

Table 15.

Date:	**Test Name:** Stepper motor and DC motor
Test Description: Make both the Stepper motor and the DC motor to run for a long period of time	
Expected Outcome: The performance should remain stable both in terms of fan blades intensity and the rotation of fan's main body	
Actual Outcome:	

Table 16.

Date:	**Test Name:** Fan's Blades Intensity
Test Description: Decrease or increase the level of intensity of fan's blade and observe if the intensity is corresponding to what has been set	
Expected Outcome: The fan should reduce or increase its intensity accordingly	
Actual Outcome:	

Table 17.

Date:	**Test Name:** An Object Blocking the Movement of the Fan
Test Description: Putting an Object on the path of the fan movement and preventing the fan from rotating toward the direction that is expected	
Expected Outcome: Through the feedback system from Arduino Mega, the fan will try to resist the object on its path for the duration of 5 seconds, after that short window of time if the angle from fan rotation remain constant then the fan will stop rotating	
Actual Outcome:	

Table 18.

Date:	**Test Name:** The Fan Falling down due to a collision
Test Description: An object will collide with the fan, and causes the fan to either start wobbling or fall down to the ground depending on the intensity of the collision	
Expected Outcome: If the fan simply wobbles, it continues operation as expected. If it falls over, the obstruction algorithm should turn it off.	
Actual Outcome:	

Table 19.

9. Appendix B: User Interface

9.1 Introduction

The user interface of the FANTOM there are two main components. The user interface on the FANTOM will mainly be buttons. They only control the most basic function of the device, including turning on/off, increase/decrease fan speed. The second part of the user interface is going to be the mobile application which is part of the prototype. The application is going to be connected to the FANTOM through bluetooth. It will not only show the current status of the FANTOM, but also let the user control the fan speed, rotation and switching between different modes.

9.2 User Analysis

The main target consumer of the FANTOM is the higher end of the cooling market, who are willing to spend a bit more money upfront to get the benefit of automation. This also include high end office spaces and shops fronts. These consumers are not particularly budget sensitive but they expect things to work intuitively and reliably. This is why having a good user interface that provides the user with speedy control and feedback is going to be very important. The UI design of this app will assume that the user has experience with using smartphone apps and is generally familiar with the iconography used in ventilation and airflow systems.

The device will require AC power for operation. The user will have to plug the device into a standard power outlet to deliver power. As soon as the device is powered, it starts operating by performing a certain amount of rotations to look for users. Once plugged in, the device will be operating while there are users in the room. If the room is empty, the device will go into sleep mode to save power. There will be a physical switch for the user to turn the FANTOM on/off manually. The Phone application can also be used to turn the device on or off.

9.3 Graphical Presentation

The physical representation of the companion application is shown below in Figure 16 running on an iOS device.

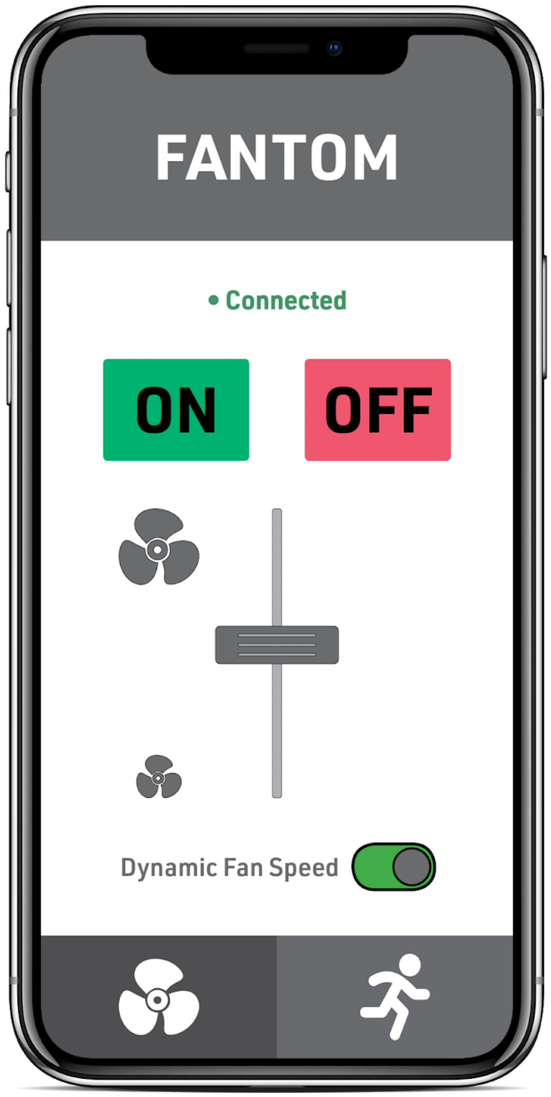

Figure 16.

Figure 17.

Figure 17. Connection Indicator: The connected icon will appear when the phone is connected to the device via a Bluetooth connection. When the connection is terminated, the icon will turn red, and read "Disconnected". Additionally, the rest of the icons that are detailed below will be greyed out, rendering the operation of the device through the phone unavailable.

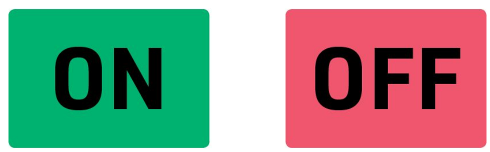

Figure 18.

Figure 18. On/Off Buttons: The user can turn the fan on or off using these buttons.

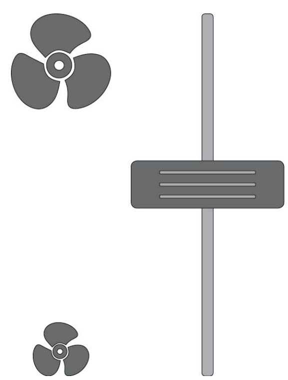

Figure 19.

Figure 19. Fan Intensity Slider: The user will be able to control how fast the fan spins. If Dynamic Fan Speed is off, the fan will stay at the selected speed regardless of the distance of the user from the fan. If Dynamic Fan Speed is on, the fan will reduce its intensity as the user moves closer to the device, but will increase as the user moves away from the screen until it reaches the speed limit set by the user.

Figure 20.

Figure 20. Dynamic Fan Speed Button: Revert to automatic control of the airflow intensity. The speed will be dependent on the distance that the person is located with respect to the sensor.

Figure 21.

Figure 21. The fan icon will correspond to the application page with manual control of the device operation and fan speed. The person icon will correspond to the application page in which to select which mode the device will operate in. There are two modes of operation; tracking and oscillating mode as shown in Figure X below.

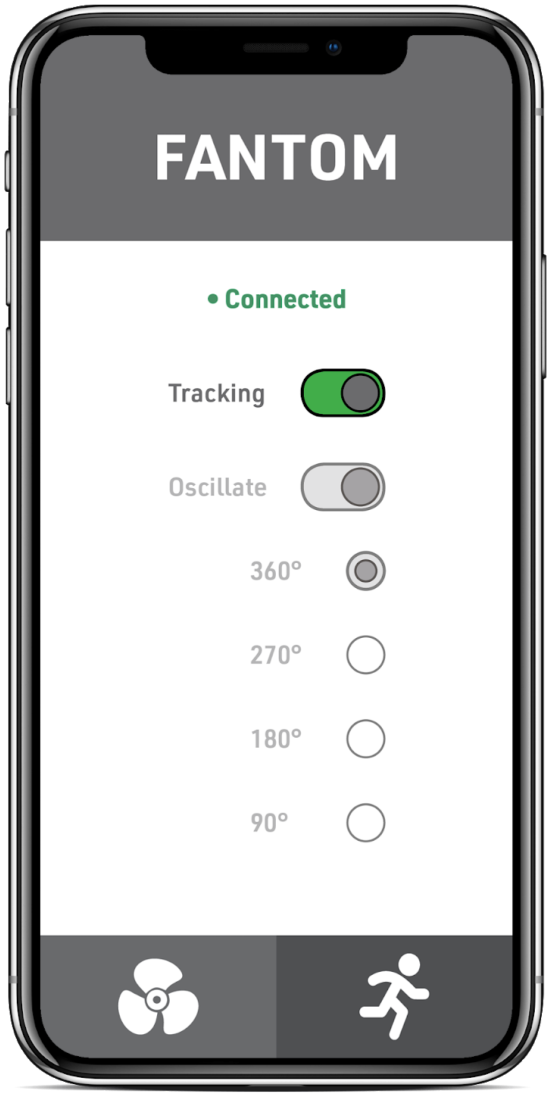

Figure 22.

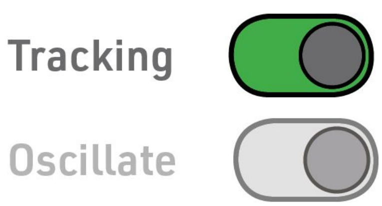

Figure 23.

Figure 23. Tracking Toggle Button: This button allows the user to select whether or not they want the fan to track their location and keep the airflow directed at them. The Oscillate option is not available while Tracking is turned on.

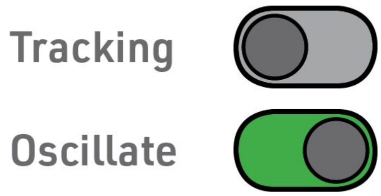

Figure 24.

Figure 24. Oscillate Toggle Button: This button allows the user to select whether or not the fan will oscillate in a given range that will be chosen by the user. The options for the default ranges are shown below in Figure X.

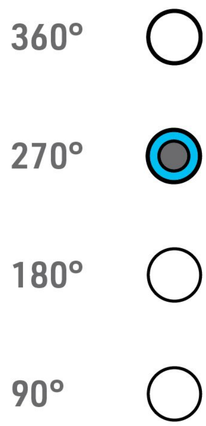

Figure 25.

Figure 25. Oscillation Range Options: These options will allow the user to set the range of oscillation of the fan when Oscillate mode is turned on. All of these options are with respect to the front of the fan.

9.3 Technical Analysis

The FANTOM had two main features that directly interact with the end user. The on board control and the mobile application. In order to minimize the learning curve for the consumers coming from the traditional fan market, the on board control will be kept relatively simple, and the mobile application interface is going to as closely mimic the layout as the on board control as possible.

The first step of building a user interface is visibility/discoverability. This will be achieved by having all of FANTOM onboard control in one place on the base of the machine. There are going to be a few buttons, slider, and knob. All the buttons should be spring loaded so when not under use or light touching of the button does not register, but a light press with one finger on the button should even when operated by a child (affordance of being pressed), an elder or people with repetitive strain injury. Fast repeated pressing should not result in noticeable hand strain. The button should bounce-back to its original position immediately after it is being released. There should be audio feedback when pressing and releasing the button. The slider should have enough resistance so accidental touching does not result in a change in position. When pressure is applied, the slider should glide smoothly within the track with no side to side movement. This should be easily achievable using one hand even for people with limited hand/wrist strength (affordance of being adjected). When the slider reaches either end of the track, the handle should stop immediately with no jitter. Pushing the slider back from either end of the track should not require extra force to initiate. The knob should only be able to stop at the given presets. When set to a given preset, the knob should stay in place with no wobble. It should be spring loaded so light touching would not rotate the handle, but would also be easily adjustable with one hand even by people with limited hand/wrist strength (affordance of being turned).

There are also a few LED indicators on the FANTOM control board. They should be bright enough to be clearly visible during daylight, but also dim enough so they are not harsh to look at during nighttime.

The ON and OFF buttons will be labeled with large text and colour coded (green for the on button and red for the off button). These colour are chosen because of their traditional association with turning on or turning off a machinery in North America. When the user pushes ON, the power indicator (a white LED with the word power write on top) located next to the two button will turn on. The OFF button will turn off the power indicator. This feedback should appear instantaneously to the user. Pressing the ON button when powered or pressing the OFF button when not powered will not result in any response. Alternatively, pressing the ON or OFF bottoms will result in the same behaviour.

The fan speed slider will be position under the ON and OFF button (closer to the using when facing the FANTOM), On the top of the slide there will be a large fan icon, and on the button there will be a smaller but still clearly visible fan icon. The size of the fan positively reflects the amount of wind generated by the fan. When a user positioned in front of the machine adjusting the fan speed, pushing away from themselves (toward the bigger fan icon) will result in a faster fan speed. When pushing closer to themselves (toward the smaller fan icon) will result in a slower fan speed. This orientation is also chosen based on the widely agreed upon conceptual model. The user should feel the change in wind speed as they are adjusting the slider.

The dynamic fan speed button is located under the fan speed slider (closer to the user when facing the front of FANTOM). A white LED indicator labeled with the same name indicate current status. Odd number of button presses turn the feature on along with the indicator, and even number of button presses turn the feature and the indicator off.

The tracking button and indicator (white LED) is placed on the right of the fan speed slider, it functions largely the same as the dynamic fan speed button. When tracking is turned on, any adjustment to the oscillation knob will turn the tracking feature and indicator off (in order to mimic the behavior in the mobile application where the oscillate option is greyed out when tracking is turned on).

The oscillation knob (shown in figure below) is located on the right of the fan speed slider. It serves the same function as the oscillate selector in the mobile application. The location of each angle are mapped to how far away from the center axis the fan is going to turn (±45° when the oscillation is set to 90°, etc.). Any manual adjustment to the knob will turn off the tracking feature (and the tracking indicator).

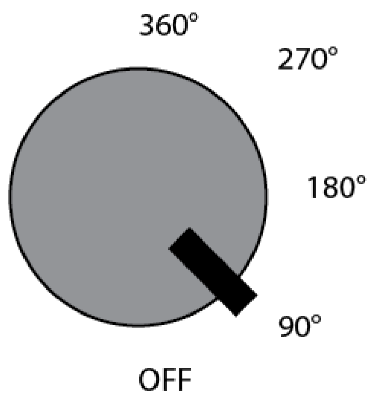

Figure 26.

9.5 Engineering Standards

Standard	**Description**
Bluetooth Specification Version 4.2 [1]	Bluetooth Energy Use Standard
IEEE 1012-2012 [2]	System and Software Verification and Validation
ISO 20282 [3]	Ease of Operation of Everyday Products

Table 20.

9.6 Analytical Usability Testing

We carried out our analytical usability testing on the UI by analyzing several devices from various industries that have a remote control feature through smartphone apps. We identified features from these apps that would work with our device and then made several iterations on the initial design. The following list details the usability features that were decided on for the final iteration:

Connection Indicator: Will clearly display whether or not a Bluetooth Connection with the fan is established; Connected when true, Disconnected when false.
On/Off Buttons: Will clearly highlight the state that the fan is currently in; On will be slightly greyed out when the fan is off, and vice versa.
Fan Intensity Slider: Will correspond smoothly to the current fan speed, i.e. the fan will increment its rotation speed in a continuous manner
Dynamic Fan Speed Toggler: Will be greyed out when off, and will be highlighted green when on.
Fan/Tracking Options Tab: Will quickly transition between the two available tabs. It also uses symbols that will be easy to interpret by the user.
Tracking/Oscillation Togglers: Will be greyed out when off, and will be highlighted green when on.
Oscillation Range Radio Buttons: Chosen option will be clearly highlighted, all other options will be greyed out.
All the physical UI that will be mounted onto the fan itself is still being discussed at this stage.

9.7 Empirical Usability Testing

Future iterations of the device and the application will depend heavily on user feedback. All feedback that concerns the UI will be considered and implemented if it is decided that the benefit of doing so would improve the UI significantly. In addition, all feedback concerning the fan's features itself and will be considered for future versions of this. This may result in changes being made to the UI and will be made accordingly. The current design relies on the user's familiarity with smartphone applications in the case of the phone app, and the intuitivity of the physical UI that is present on the fan itself.

9.8 Conclusion

At the PoC stage, the UI of our device will be limited to a physical On/Off switch. This early prototype will be set up to showcase the features of our fan in a minimal fashion. All these features will be shown through changes in code and live demonstrations.

Our final prototype will have a fully features smartphone app, as well as a physical UI present on the device itself. As stated, this UI will allow the user to control all possible features of the fan without leading to errors due to the many interlocks and failsafes put in place. The intuitive design of both application and physical UIs will allow anyone with experience in using modern devices to control the fan to their liking.

9.9 References

[1] "Specifications | Bluetooth Technology Website", Bluetooth *Technology Website*, 2019. [Online]. Available: https://www.bluetooth.com/specifications/. [Accessed: 09- Jun- 2019]
[2] IEEE Standard for System and Software Verification and Validation - Redline," in *IEEE Std 1012-2012 (Revision of IEEE Std 1012-2004) - Redline* , vol., no., pp.1-223, 25 May 2012
keywords: {IEEE standards;Software engineering;Formal verification;Software testing;Computer security;Software resuability;Product life cycle management;Enviornmental factors;Hardware;environmental verification and validation (V&V) factors;hardware V&V;IEEE 1012;integrity level;independent V&V (IV&V);risk/hazard/security analyses;software life cycle;software V&V;system life cycle;system V&V;V&V;V&V measures;V&V of reuse software},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6251988&isnumber=6251987
[3] ISO 20282-1:2006, Ease of operation of everyday products -- Part 1: Design requirements for context of use and user characteristics