Deliverable G - Prototype II and Customer Feedback

Nada El Rayes - 300143185
Amanda Beraldo Brandao de Souza - 300211045
DongYu Wang - 300114760
Aaron MacNeil - 300199522
Shayleen Ghanaat - 300198724

GNG 1103

University of Ottawa

March 14th, 2021

Table of contents

Introduction

JAMZ Automated Delivery is a drone delivery service focused mainly on the shipment of food from restaurants to the client. Their drones are ready for use, however, some essential features, like a climate sensor inside the package, are not functional yet. The main goal of this project is to develop a reliable solution that generates information about the content of the package during delivery. The device should provide valid and consistent data on the temperature and humidity of the food and send a warning to the drone's microcontroller (Raspberry Pi) when the conditions inside the package are not ideal. In this document, group D8 presents the improvements done based on client feedback, a detailed description of the second prototype, the testing process and the analysis of the results obtained.

Feedback from client

Table 1. Feedback from the client meeting

Concept Title	Description	Feedback
Basic principles observed and reported	knowledge generated underpinning hardware project concepts and presentation.	The opening and ending of the presentation has been praised. Some have suggested that the group should have used more visulas. It was pointed out that the presentation has been interrupted by an individual who has forgotten to mute the mice, it has confused some of the audience. It was mentioned that the group was dressed semi-formal. There were some pusing while presenting. The group should work on clarity of their voices. The client has prized the group for including all the information they needed in the presentation. The transition between the slides need to be improved
Technology concept and/or	Invention begins, practical	Some commented that the group

application formulated.	application that is identified	has significant ideas and great understanding of needs for the project. It was noted that the project had made excellent progress. The AutoCad has been highlighted. It has been noted that the Cad model has shown a lot of insight of how the group model would work. Also it has pointed out the Cad model has been very detailed and shown in 3 views.
System/sub-system model or prototype demonstration	A high fidelity system/component prototype that adequately addresses all critical scaling issues is built	The technical schematic of housing and fan has gotten attention.

Project updates

The sensor that we ordered (Sensirion AG SHT31-DIHIH8120-021-001S-P2.5KS) arrived this week and we started to test it with the Arduino Uno. When performing the tests, it was possible to notice that the sensor was not ideal for our project since the small size makes it hard to solder it with a regular soldering iron. To overcome this challenge the group decided to use the sensor DHT22 that is less precise but is expected to be equally reliable and consistent in providing data. Since the sensors DHT11 and DHT22 are very similar, we could use the code previously written for DHT11 to develop a code for DHT22 without many changes.

From the data obtained on the tests for the housing, it was possible to conclude that it is efficient in being waterproof and the components do not overheat inside it when being used for long periods. Therefore, we will use the housing LeMotech ABS IP65 Junction Box in other prototypes and in the final product. We decided to add to the housing a cooling fan to assist in controlling the temperature of the components when the drone is working exposed to the sun.

Prototype II

Description

Our second prototype called "SAAND II" is a physical comprehensive prototype composed of two sensors DHT22, one microcontroller Arduino Uno, a housing LeMotech ABS IP65 Junction Box and a cooling fan San Ace 60 103P0605H701.

Objectives

The objectives of this prototype are to develop and test a code that receives the measurements of temperature and humidity from the two sensors, analyzes this data and sends a warning to another microcontroller indicating that the temperature and/or humidity are not in the ideal range and to ensure that the cooling fan works properly.

Analysis of critical components

Critical components for the sensor

- 2×sensors DHT22

Critical components for the microcontroller

- Arduino Uno

Critical components for the housing

- Housing LeMotech ABS
- Cooling fan San Ace 60 103P0605H701.

Analysis of system integration

The sensors will stay outside the housing to properly produce data of humidity and temperature, so it should be connected to the Arduino Uno through wires. A waterproof tube will connect the wiring from the microcontroller Arduino Uno that is inside the waterproof box to the sensor. The data received will be interpreted using a code, and a warning will be sent to the drone's microcontroller if the temperature is below or above the accepted range and/or the humidity is above a certain limit value. The fan will be on to control the temperature of the components inside the housing.

Feasibility analysis

The costs for the components are at an affordable price, and the components can be found in various different locations. Both of these factors are important when purchasing components for the temperature and humidity sensor. For example, The Arduino Uno and the cooling fan are important parts of the sensor and can easily be found.

Component	Price
Housing (junction Box):	$14
Sensor (DHT22)	$13.27
Microcontroller (Arduino Uno)	$26.95
Cooling Fan	$14.32
Total Cost	$68.54

The total project cost is $68.54, however it cost the group $50.84 because the group already has the microcontroller.
The initial budget that was assigned to the group is $60. The group has managed to provide all the need for the temperature and humidity sensor project, and even save money.

Risk mitigation

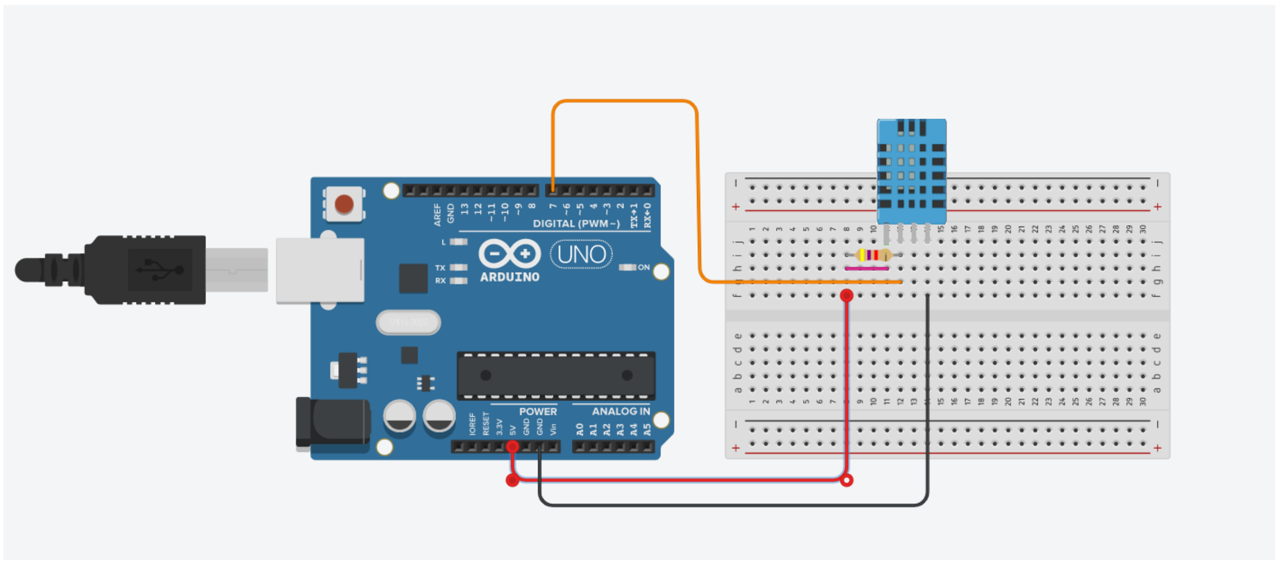
Problem	Action expected
Difficulties with the code	Asking insight from the professor and TA's Watching Tutorial
Fan breaks/fails on the test	Try a 5V fan Choosing alternative way to replace the fan
Sensor breaks/fails on the test	Try the DHT11 sensor
Housing	Try alternative Housing

Stopping criteria for tests

- No substantial change (± 0.1 °C for temperature and ±15 points for humidity) on the readings from the sensors after one hour of testing
- For the second test the stopping criteria is the fan working without interruption for an hour.

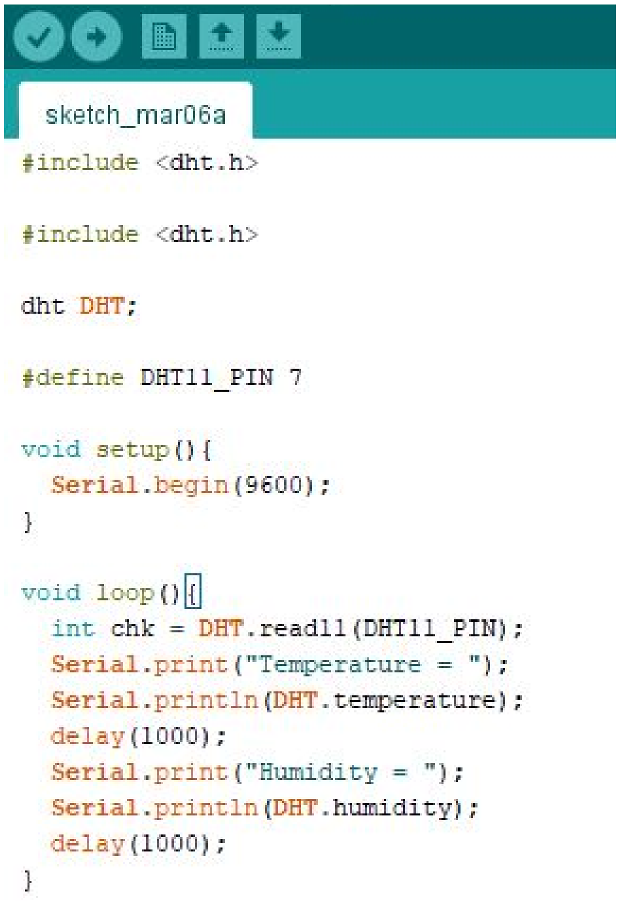
Tests

Tests were performed to evaluate the parameters of the prototype previously described. In this section, we describe in detail how the tests were held and what are the results obtained.



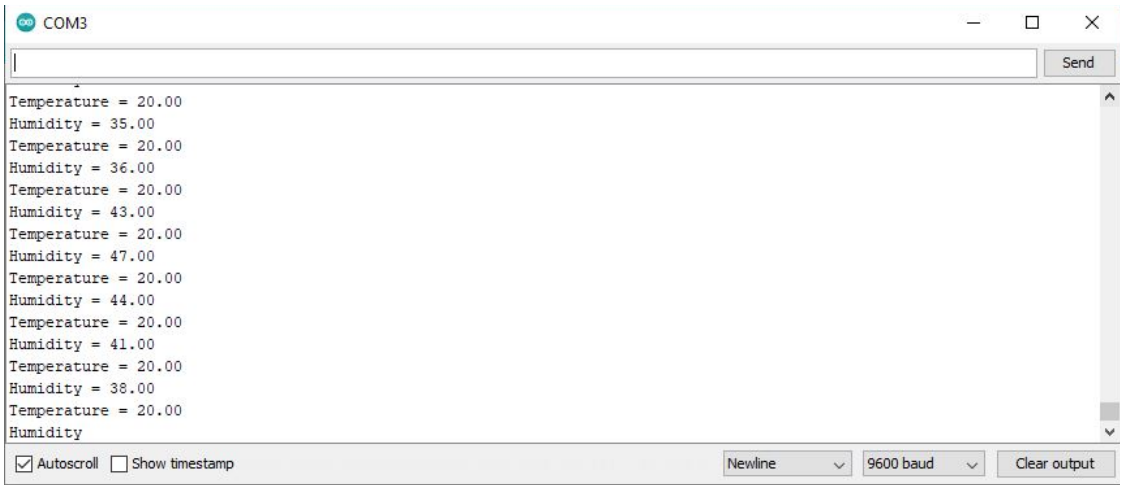
- **Code for the sensor**

Figure 1.1 - Simulation of the circuit using an Arduino Uno, a breadboard, a resistor and the sensor DHT11 used for the code testing on the first prototype.



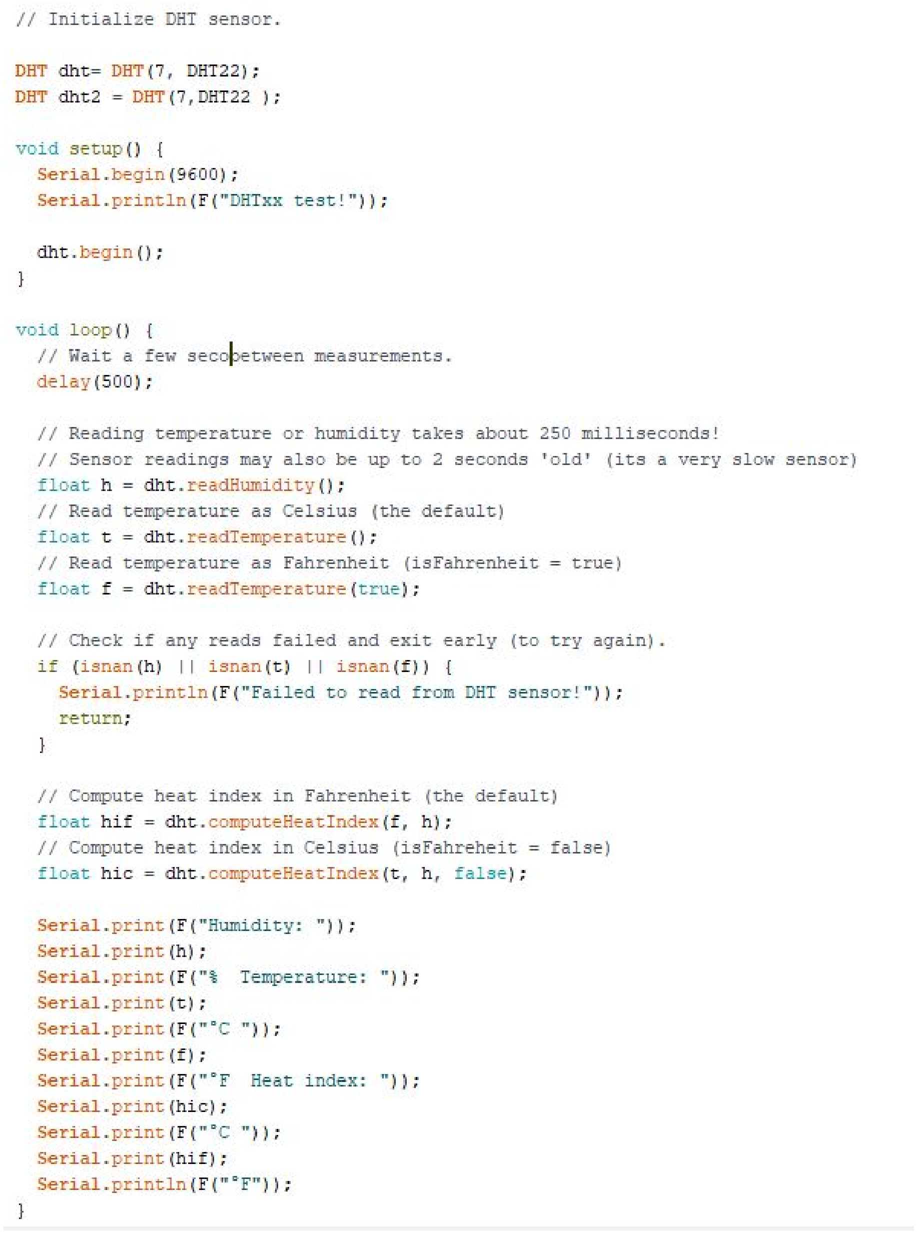
```
#include <dht.h>

#include <dht.h>

dht DHT;

#define DHT11_PIN 7

void setup(){
  Serial.begin(9600);
}

void loop(){
  int chk = DHT.read11(DHT11_PIN);
  Serial.print("Temperature = ");
  Serial.println(DHT.temperature);
  delay(1000);
  Serial.print("Humidity = ");
  Serial.println(DHT.humidity);
  delay(1000);
}
```

Figure 1.2 - Code used for the sensor DHT11 on the first prototype



```
Temperature = 20.00
Humidity = 35.00
Temperature = 20.00
Humidity = 36.00
Temperature = 20.00
Humidity = 43.00
Temperature = 20.00
Humidity = 47.00
Temperature = 20.00
Humidity = 44.00
Temperature = 20.00
Humidity = 41.00
Temperature = 20.00
Humidity = 38.00
Temperature = 20.00
Humidity
```

Figure 1.3 Figure 1.3 - Data received from the sensor DHT11 on the first prototype



```
// Initialize DHT sensor.

DHT dht= DHT(7, DHT22);
DHT dht2 = DHT(7,DHT22 );

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}

void loop() {
  // Wait a few secobetween measurements.
  delay(500);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print(F("Humidity: "));
  Serial.print(h);
  Serial.print(F("%  Temperature: "));
  Serial.print(t);
  Serial.print(F("°C "));
  Serial.print(f);
  Serial.print(F("°F  Heat index: "));
  Serial.print(hic);
  Serial.print(F("°C "));
  Serial.print(hif);
  Serial.println(F("°F"));
}
```

Figure 1.4 - First part of the code used for the sensor DHT22 on the second prototype

```
#include <DHT_U.h>



#include <dht.h>

#include <Adafruit_Sensor.h>

// Example testing sketch for various DHT humidity/temperature sensors
// Written by ladyada, public domain

// REQUIRES the following Arduino libraries:
// - DHT Sensor Library: https://github.com/adafruit/DHT-sensor-library
// - Adafruit Unified Sensor Lib: https://github.com/adafruit/Adafruit_Sensor

#include "DHT.h"

#define DHTPIN 7  // Digital pin connected to the DHT sensor

#define DHTPIN 8    // Digital pin connected to the DHT sensor



#define DHTTYPE DHT22   // DHT 22  (AM2302), AM2321




// Initialize DHT sensor.

DHT dht= DHT(7, DHT22);
DHT dht2 = DHT(7,DHT22 );

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}

void loop() {
  // Wait a few secobetween measurements.
  delay(500);
```

Figure 1.5 - Second part of the code used for the sensor DHT 22 on the second prototype

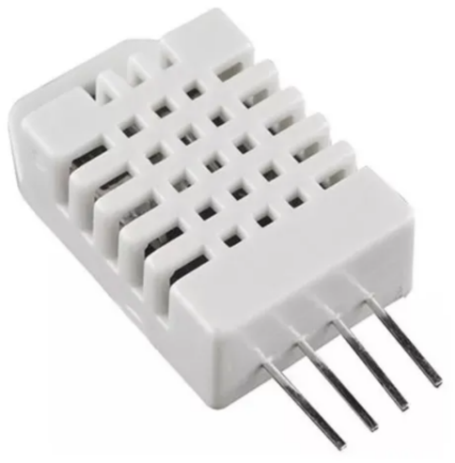


Figure 1.6 - Sensor DHT22

Testing method

Analytical Prototype Testing

Estimate Duration Time

1 hour

Description of Prototype

Code that receives the information from the sensor and prints it on the serial monitor.

Description of Results to be Recorded and how these results will be used

The data printed on the serial monitor shows the temperature and the humidity readings obtained from the sensor DHT11 on the first prototype. These results were used to develop the code for a similar sensor, the DHT22, that will be effectively tested on the next prototype.

- **Cooling fan**

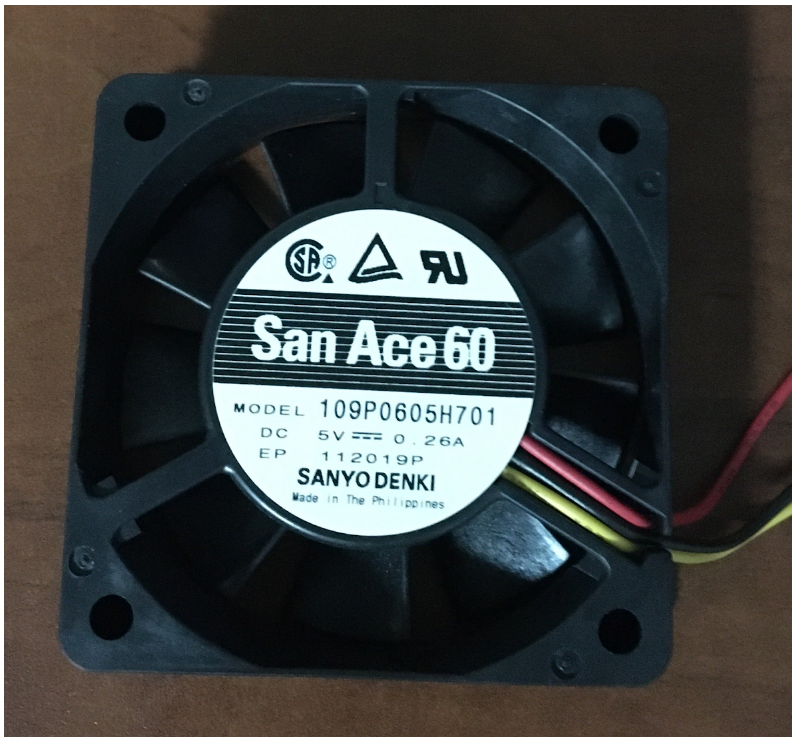


Figure 2.1 - Cooling fan San Ace 60 103P0605H701.

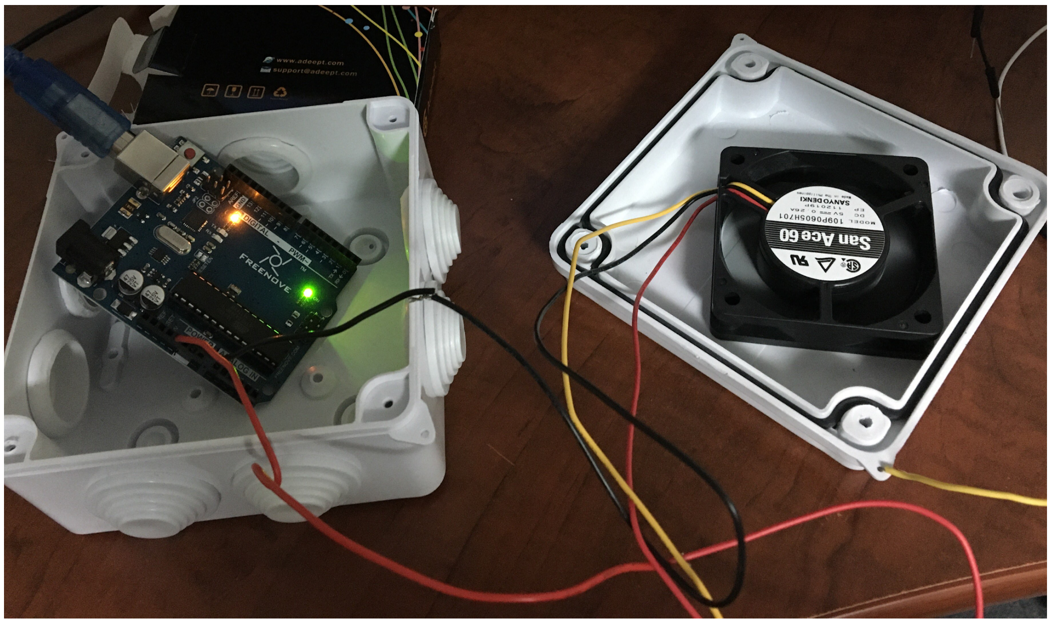


Figure 2.2 - Arduino Uno inside the housing linked with the cooling fan

Testing method

Physical Prototype Testing

Estimate Duration Time

1 hour

Description of Prototype

Cooling fan connected to an Arduino Uno

Description of Results to be Recorded and how these results will be used

The Arduino effectively powers the cooling fan for the one hour period analyzed. This information will be used in the implementation of the fan on the final assembly. With the test we can see that the fan has extremely low power consumption and, therefore, can be used in the product's final assembly.

Conclusion

The results obtained from the tests helped the team to better understand the behaviour of the code and to define specific points to work on the next prototype. Because of the unexpected change in the sensor, we have to wait to receive the sensor DHT22 that we will be using for our third prototype. We expect to be able to use this new sensor for the tests until the end of the week. The tests for the cooling fan confirmed that it has low power consumption and, therefore, will be used on the final product.

Until now, we have three components that have proven to be adequate for use on the final product. These components are: the Housing LeMotech ABS, the Arduino Uno, and the fan San Ace 60 103P0605H701. On the next prototype we will test and decide on the sensor that will be used on the final project.

Project Plan Update

For Project Deliverable G, tasks related to the physical aspect of testing, such as testing the code with the sensors, were assigned to Shayleen and Aaron, since they had access to the physical components. A meeting was held to perform the tests so that all team members could see the process and results. The other members of the group, Amanda, Nada and DongYu worked in documenting the steps of the tests and their results, as well as completing the document of the deliverable with analysis and feedback from the client.

For the next two weeks, the team will be working on the third and final prototype. The main objective of this prototype is to refine both the code developed on the second prototype and the mechanism for the activation of the fan. Aaron and Shayleen will be working on the physical and code testing for this prototype as well, and Nada, DongYu and Amanda are assigned to tasks related to the documentation of the tests, analysis and feedback from the client.

Our project plan is available on Wrike at the following link: https://www.wrike.com/frontend/ganttchart/index.html?snapshotId=t2yAhqcK65CTnm6dge5LTQO0i5SI6EoJ%7CIE2DGNBVHA3TCLSTGE3A