Mobile Phototherapy Suit

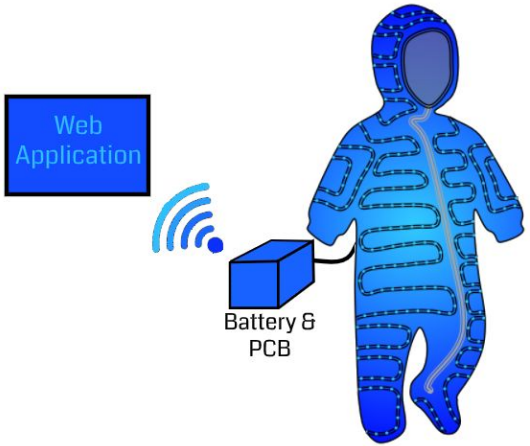
ECE 445 Fall 2020
Abhay Patel, Satwik Pachigolla, Dhivan Patel
Team 19
Health Maker Project Lead: Yusef Shari'ati

Introduction

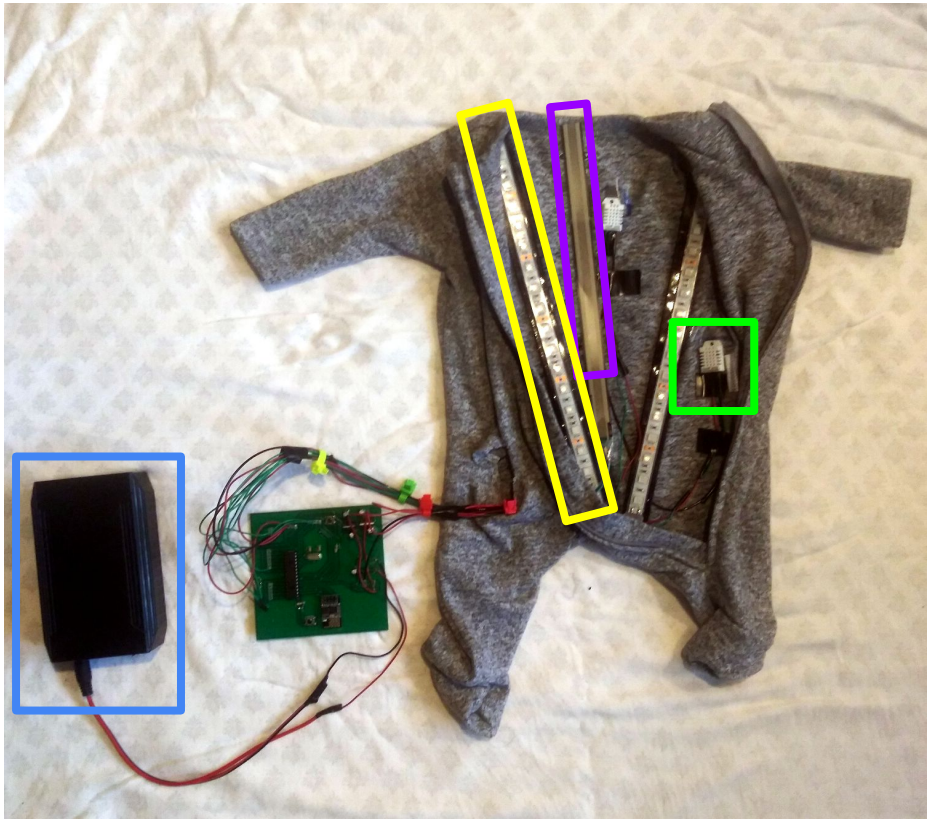
- Neonatal Hyperbilirubinemia is a condition where newborns have heightened bilirubin levels
- Bilirubin is the substance produced from the breakdown of red blood cells
- Clinical jaundice is one of the main issues arising from such a condition
- Current treatment is blue light phototherapy

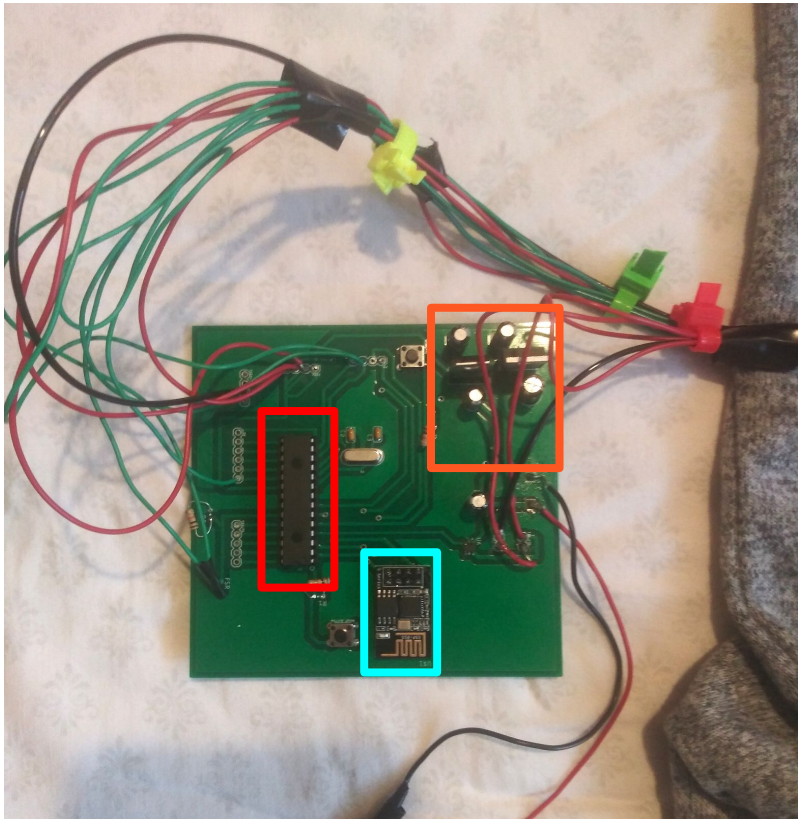
Objective

- Current treatment extends hospital stays, and separates parents from their child for extended periods of time
- To solve this we make it portable and let the family take the device home
- Frees up hospital beds
- Reduces the separation anxiety between parent and child

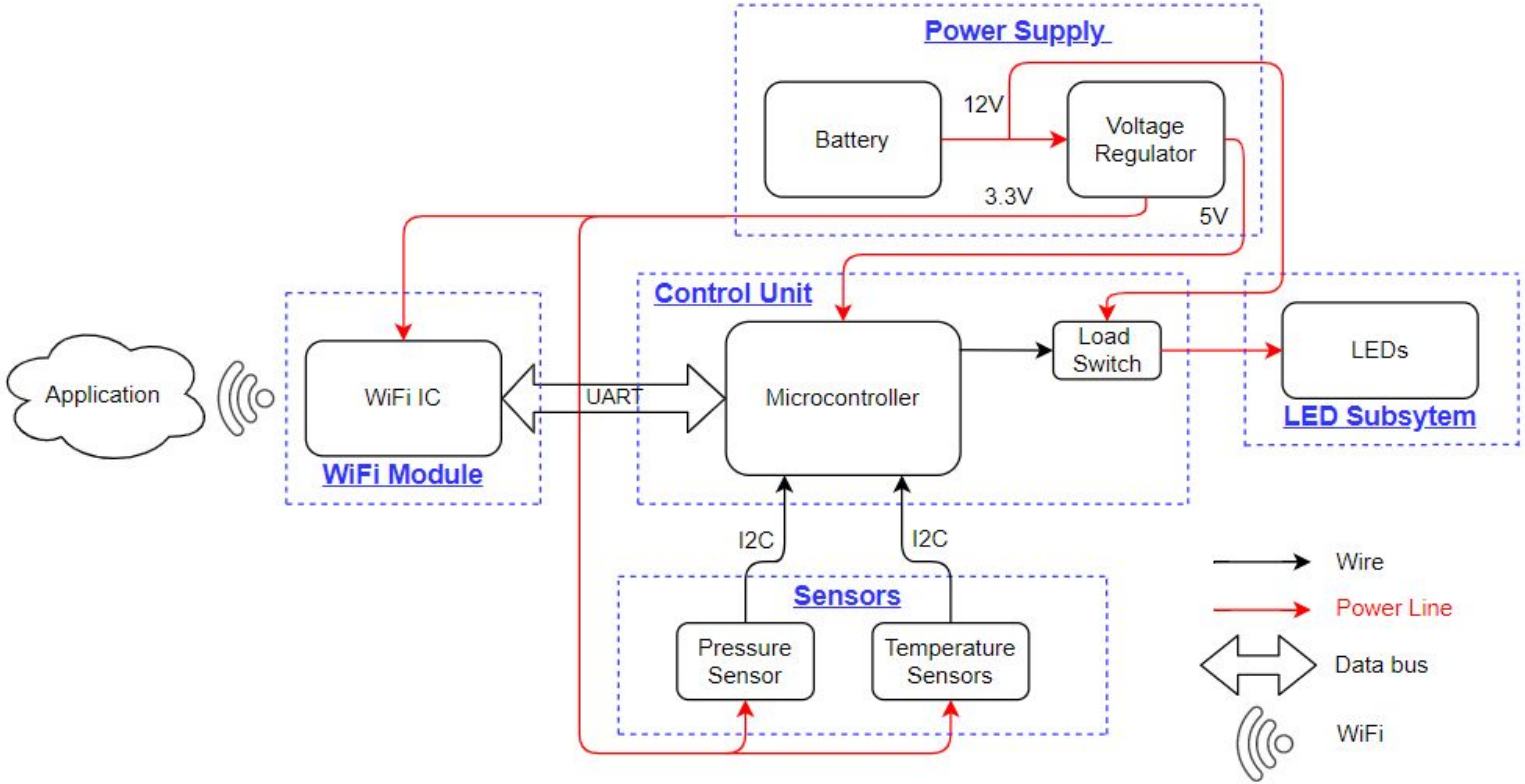


Physical Design and PCB





Block Diagram



Power Supply - Battery

- Battery powers suit for 5-7 hours
- Battery used: 12V TalentCell Rechargeable Battery
- Fully charged battery ran suit for 5 hours
- 12V DC connection to PCB

Power Supply - Voltage Regulation

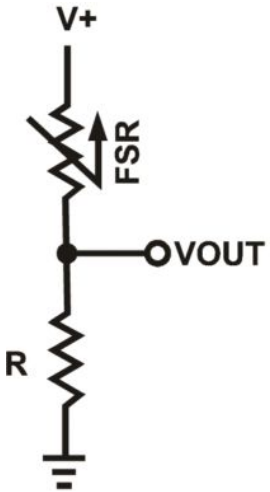
- One regulator must step down voltage from 12V to 5V
- Another must step down from 12V to 3.3V
- The outputs should be within 5%
- LDO regulators

Output voltage measured on the regulators, and the calculated percent

Regulator	Output Voltage(V)	Difference (%)
3.3 V LDO	3.291	0.002727
5 V LDO	4.999	0.0002

Sensors - FSR

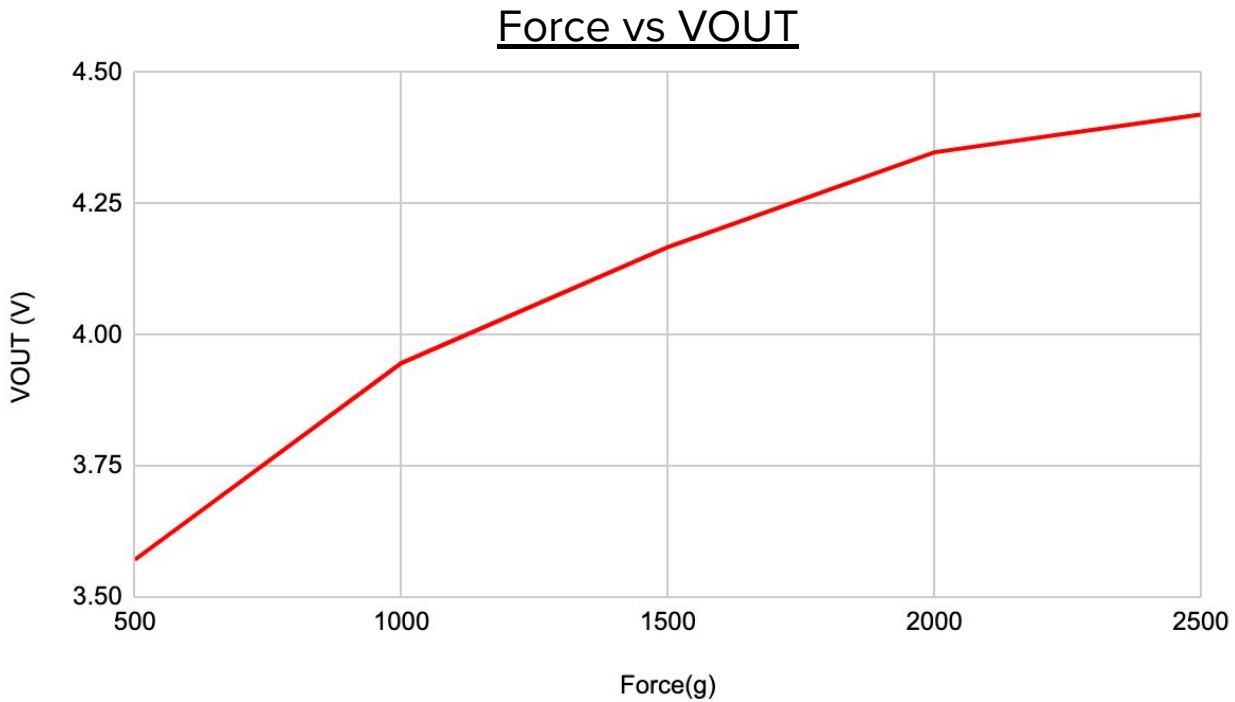
- Place 34N +/- 3 force on the sensor
- The sensor should be able to detect the force in terms of its resistance value



VOUT readings from FSR

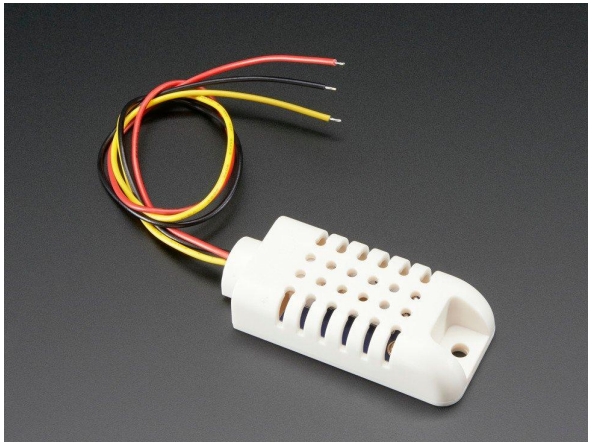
Force(g)	VOUT (V)	Analog Reading
500	3.571	730
1000	3.946	807
1500	4.167	852
2000	4.348	889
2500	4.42	904

Sensors - FSR


Force vs VOUT
4.50
4.25
4.00
3.75
3.50
VOUT (V)
500
1000
1500
2000
2500
Force(g)

Sensors - Temperature Sensors

- Measure between 15°C and 45°C for temperature readings
- Adafruit DHT22's
- Chosen for increased accuracy because it has a 2-5% threshold



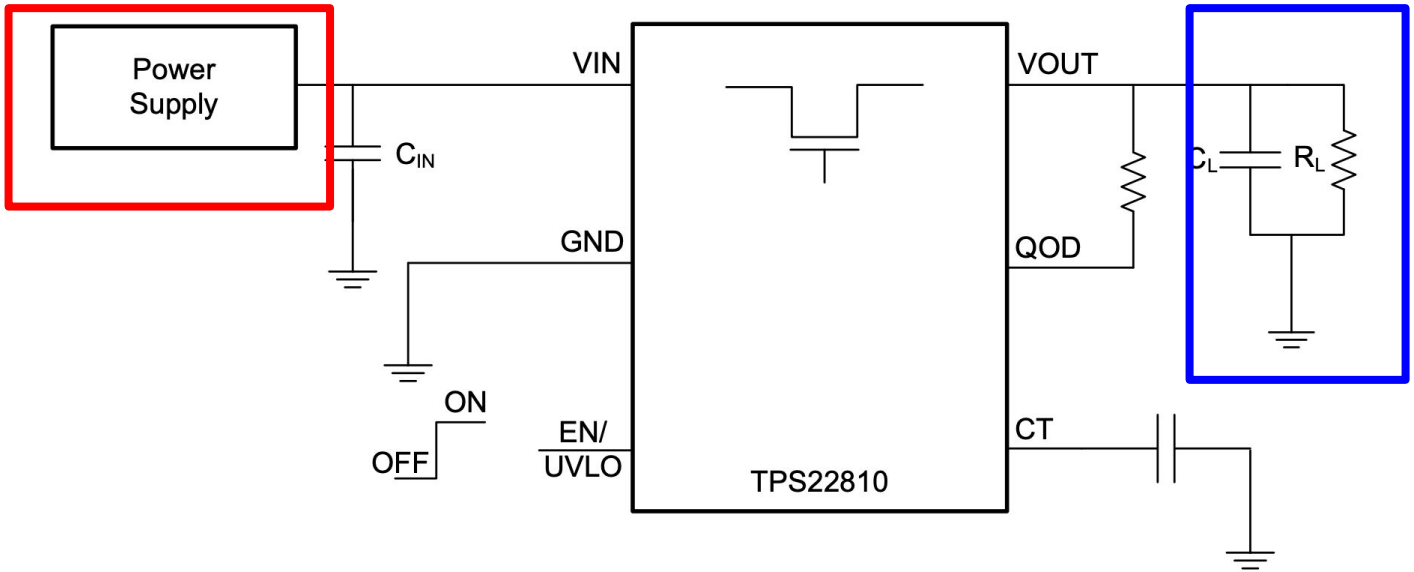
Sensors - Temperature Sensors

DHT temperature compared to Thermometer Reading

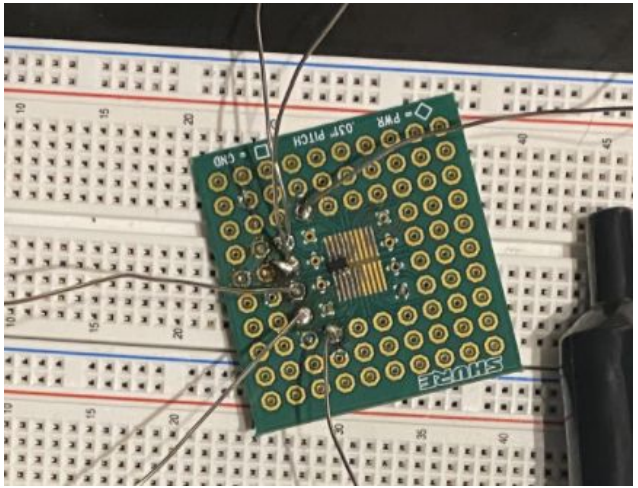
DHT Temperature Reading (°C)	Actual Temperature With Thermometer (°C)
15.10	15.0
24.00	24.2
25.70	25.7
32.33	32.3
35.72	35.7

Control Unit - Load Switch

- Switch that must control a 12V input for the LEDs
- Must produce a 12V output or no output with a control signal provided by the MCU within 5 seconds



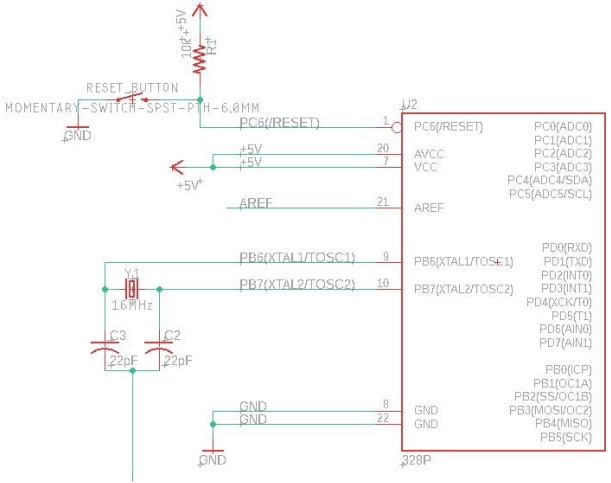
Control Unit - Load Switch

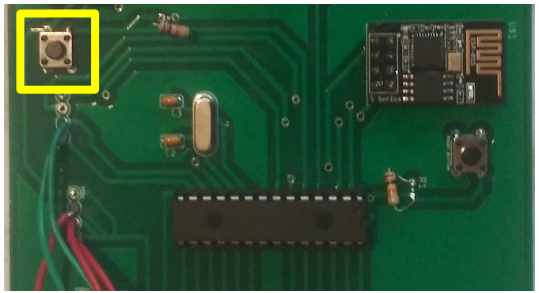


Input Voltage (V)	Control Signal	Output Voltage (V)
12	High	11.99
12	Low	0.541

Control Unit - Microcontroller

- Using ATmega328P (6 Analog Input Pins and 14 Digital I/O Pins)
- Receives data from FSR and temperature sensors through I2C
- Controls the load switch
- User button control for LEDs
- Sends temperature and usage data to WiFi module every 2 minutes





Control Unit - Microcontroller

- No real multithreading on ATmega328
  - Use time to have multitasking in the loop
- Only turn on LEDs only when FSR detects presence and button is pressed
- Temperature sensors will determine global max temperature every 5 seconds
  - If maximum temperature is greater than set threshold, turn off LEDs by sending signal to load switch

```
void loop() {
  // get time elapsed
  unsigned long currentTime = millis();

  if(fsrReading >= 850){
    digitalWrite(LED_DEBUG, HIGH);
    if (digitalRead(button) == true){
      status = !status;
      digitalWrite(controlPin, status);
    } while(digitalRead(button) == true);
    delay(50);
  }
  else{
    status = false;
    digitalWrite(controlPin, status);
    digitalWrite(LED_DEBUG, LOW);
  }

  if (currentTime - previousTime_fsr >= fsrTime){
    fsrReading = analogRead(fsrAnalogPin);
  }
```

LED Subsystem

- Blue light LED strips are responsible for delivering light between 465nm and 470nm to the infant
- User can toggle the LEDs on and off with a button
- Emergency shutoff is triggered when temperature crosses a threshold or no presence is detected within the suit

LED Subsystem

Waist = 6.8” = W_t

Shoulders = 7.3” = S

Height = 10.5” = H

I_p = 4.4 W/ft = 14.4 W/m (LED Input Power Consumption)

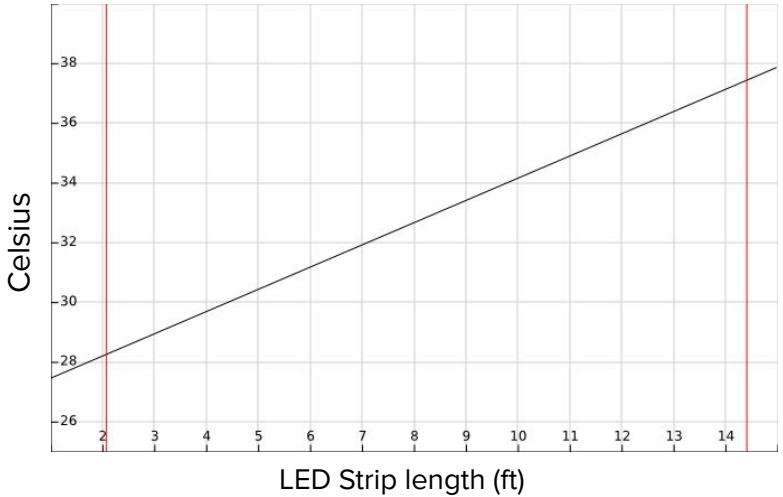
LED Efficiency = 21.6%

$S_A = (W_t + S) * H * 6.4516 cm^2/inches^2 = 955.15938\ cm^2$

Lower-bound Intensity: 0.002 W/cm^2

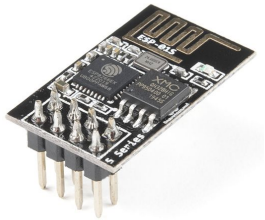
Upper-bound Intensity: 30 $uW/cm^2/nm$, @ 467.5 nm = 0.014025 W/cm^2

Equilibrium Temperature vs Strip Length



WiFi Module

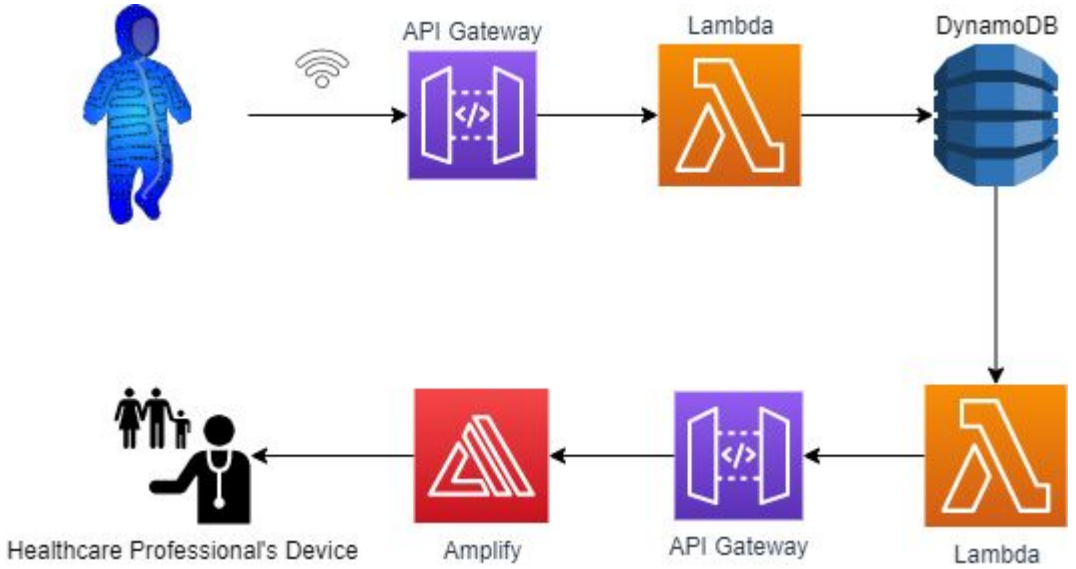
- Using ESP-01 (ESP8266) WiFi Module
- Receives data from MCU through UART every 2 mins
- Data communicated using AT Commands
- Sends data to AWS Invocation URL using TCP/IP connection



```
if(currentTime - previousTime_wifi >= wifiTime){
  //TCP connection
  if(status == true){
    String cmd = "AT+CIPSTART=\"TCP\",\"";

    //EDIT THIS FOR FULL FUNCTIONALITY
    //IP ADDRESS OF FLASK FORWARDER DEVICE
    cmd += "192.168.43.24";

    //PORT NUMBER
    cmd += "\",5000";
    ser.println(cmd);
    if(ser.find("ERROR")){
      Serial.println("AT+CIPSTART error");
      return;
    }

    unsigned long totalTime = (currentTime - previousTime_wifi)/1000;

    //prepare GET string
    String getStr = "GET /upload?";

    getStr +="&SuitID=";
    getStr += String(SuitID);
    getStr +="&Temp=";
    getStr += String(global_max_temp);
    getStr +="&Time=";
    getStr += String(totalTime);
    getStr += " HTTP/1.1";
    getStr += "\r\n\r\n";

    previousTime_wifi = currentTime;
```

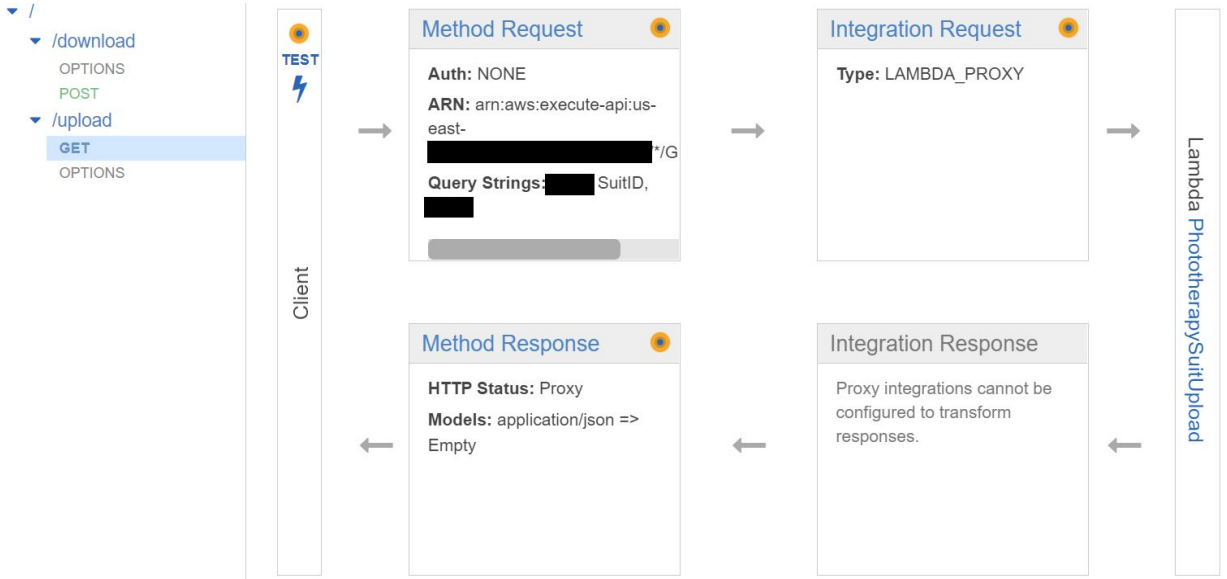
Software - IoT with AWS

- IoT using AWS serverless products
- Suit communicates using a WiFi connection
- Each suit has its own ID
- AWS serverless functionality is cost effective and accessible
- Code can be ported over to provider's servers



API Gateway

- Exposes the API for data uploads and downloads
- Adds another layer of authentication
- Validates requests format without spinning up computational resources



Lambda

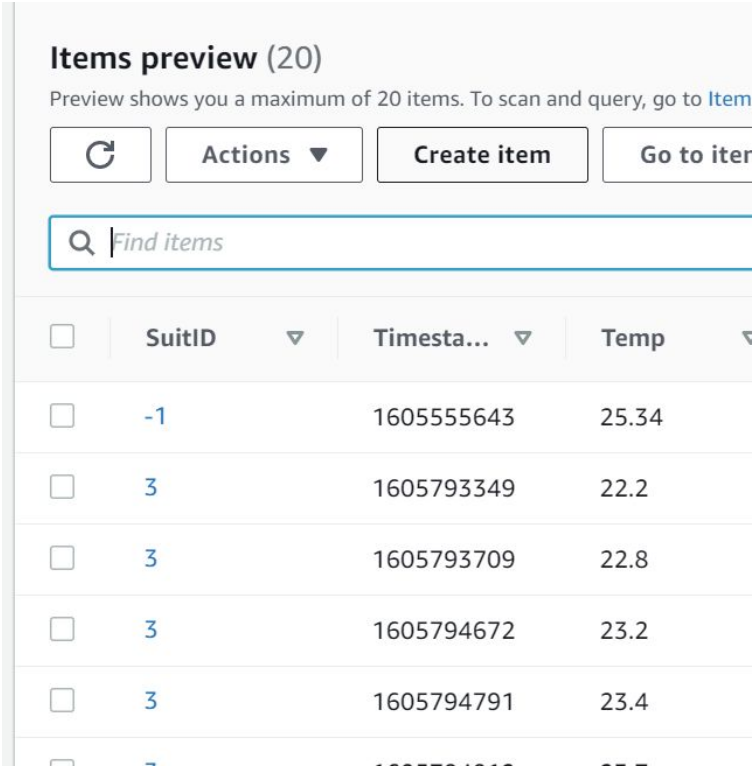
- Serverless Computation Hosting
- Used Python3 for both functions
- Performs minimal cost data transformation computation
- Interacts with DynamoDB database

```
19    try:
20        suitString = event['SuitID']
21        suitID = Decimal(suitString)
22    except:
23        return malformedInput('suitID')
24    response = table.query(
25        ProjectionExpression='#t1,#t2,#t3',
26        KeyConditionExpression='SuitID = :suitID AND #t1 BETWEEN :timestamp1 AND :timestamp2',
27        ExpressionAttributeNames={'#t1': 'Timestamp', '#t2':'Temp', '#t3':'Time'},
28        ExpressionAttributeValues={
29            ':suitID': suitID,
30            ':timestamp1': startNum,
31            ':timestamp2': endNum
32        }
33    )
34    data = []
35    for item in response['Items']:
36        temp = dict(item)
37        for key in temp:
38            if key == 'Temp':
39                temp[key] = float(temp[key])
40            else:
41                temp[key] = int(temp[key])
42        data.append(temp)
43    return ({
44        'statusCode': 200,
45        'body': json.dumps(data)
46    })
```

(11 Bytes) 24

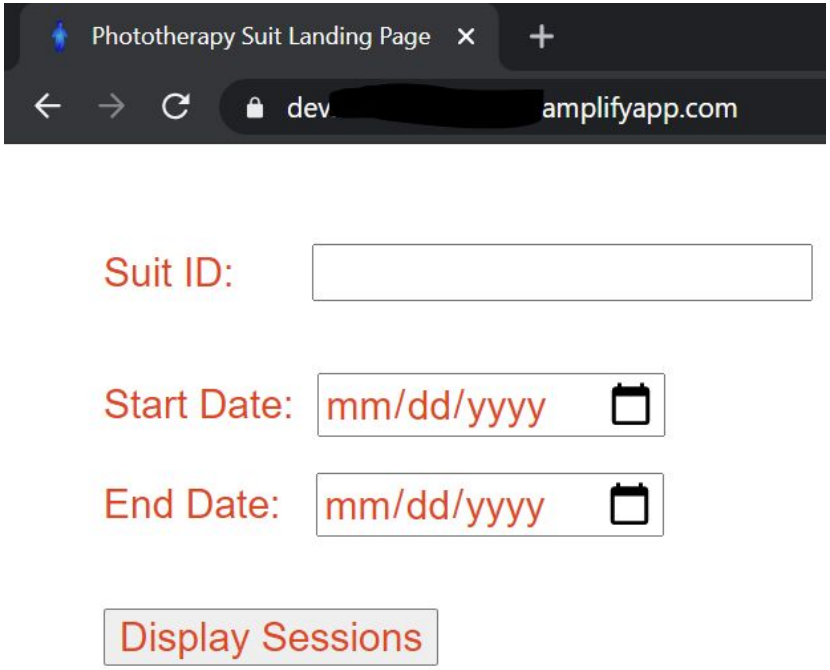
DynamoDB

- NoSQL database
- Stores data uploaded from the suit
- The data is available to be retrieved from the website hosted on Amplify
- Can be queried efficiently for large datasets
- Joins the upload and download/retrieval datapaths



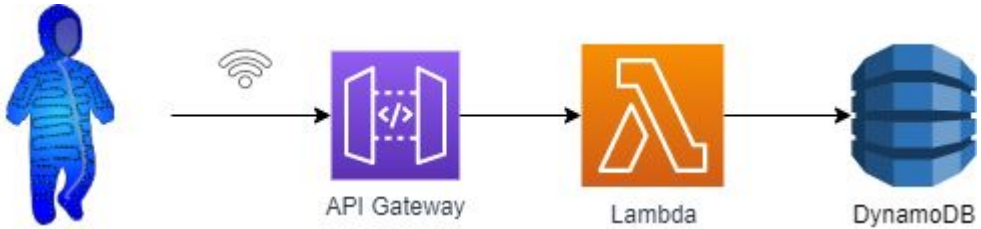
Amplify

- Hosts the website for healthcare professionals to access data
- Written using only native HTML5 and native JavaScript for easy portability
- The data is displayed in an aggregated manner
- Implements another level of authentication
- Can be accessed from any common device



Uploading Data

- Data sent from the ESP8266 chip on the suit arrives at API Gateway
- API Gateway routes the request with the data to Lambda
- Lambda processes the data and uploads it to DynamoDB
- DynamoDB holds the data which is sorted based on the Suit ID and timestamp of the upload.



Retrieving Data and Displaying Results

- Data can be queried based on Suit ID and a given date range for healthcare providers to determine if sufficient treatment is being provided to the patient

Suit ID: 5

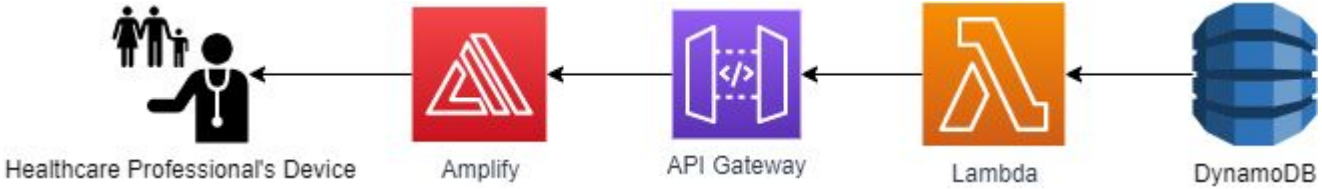
Start Date: 11/02/2020

End Date: 11/26/2020

Display Sessions

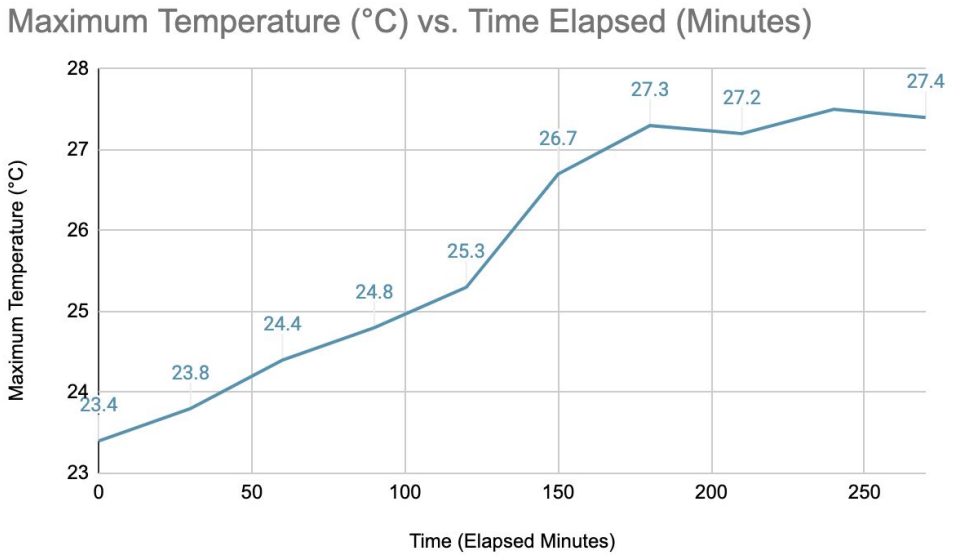
Date & Time	Treatment Time (HH:MM:SS)	Maximum Temperature (°C)	Average Temperature (°C)	Minimum Temperature (°C)
11/16/2020, 2:00:00 PM - 2:59:59 PM	00:12:03	23.4	19.0	0
11/16/2020, 3:00:00 PM - 3:59:59 PM	00:10:00	24.4	23.7	22.1
11/16/2020, 4:00:00 PM - 4:59:59 PM	00:08:00	25.3	24.6	23.9
11/18/2020, 2:00:00 PM - 2:59:59 PM	00:10:00	24.4	23.4	22.3
11/18/2020, 4:00:00 PM - 4:59:59 PM	00:18:00	27.3	25.6	23.8
11/18/2020, 5:00:00 PM - 5:59:59 PM	00:08:00	28.1	27.8	27.5
11/18/2020, 9:00:00 PM - 9:59:59 PM	00:30:42	25.8	24.9	22.9
11/18/2020, 10:00:00 PM - 10:59:59 PM	00:56:00	28.2	27.1	25.7

Total Treatment Time (HH:MM:SS): 02:32:45



Challenges

- Temperature Equilibrium
  - The temperature in the suit was quickly reaching hazardous levels
    - Calculated optimal length of LED strip

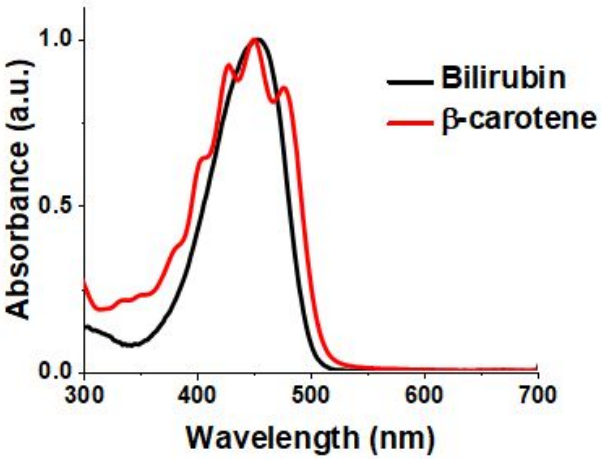


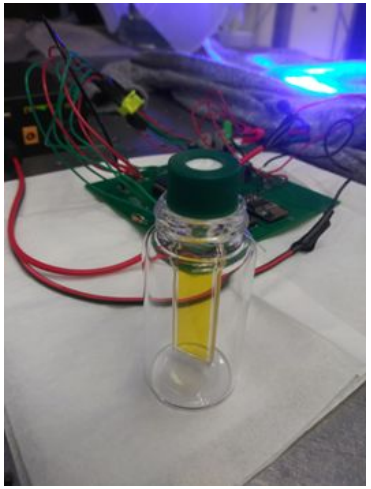
Challenges

- Mounting electronics in a cloth suit, especially connecting and testing the FSR
- Communication from the MCU / ESP8266 to AWS
  - AWS API Gateway configuration was blocking requests from the ESP8266
    - Changed the configuration
  - The serial buffer on the ATmega328 was too small to hold the entire string for the API Gateway URL
    - Used a shortened URL and forwarded the request from there

Results - Approach

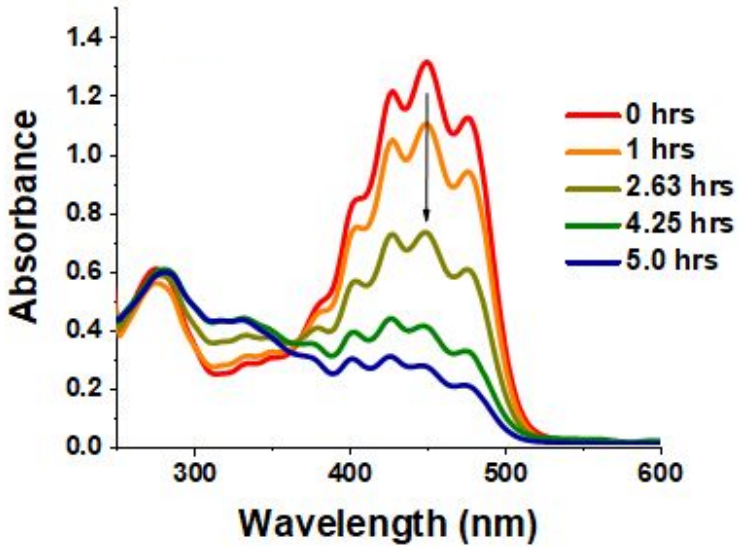
- Chemical analysis measuring the effect of the suit on Bilirubin levels.
- Model Bilirubin using β-Carotene which has a nearly identical absorption spectrum



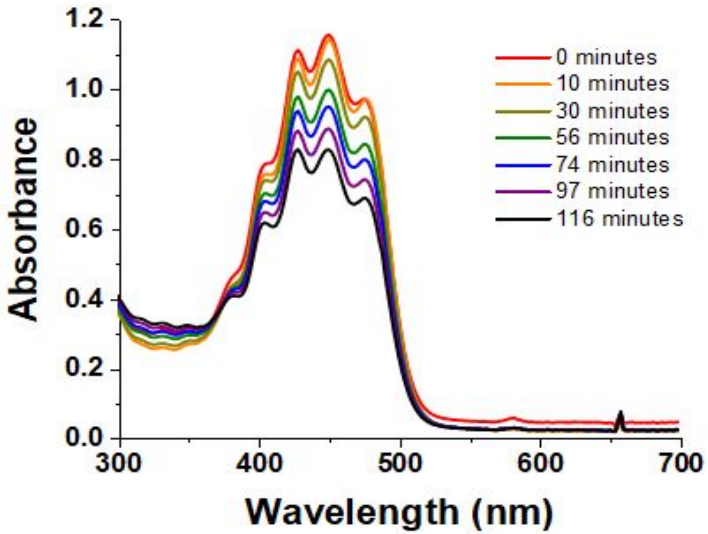


Results - Photodegradation Spectrum

Reference Plot of Photodegradation

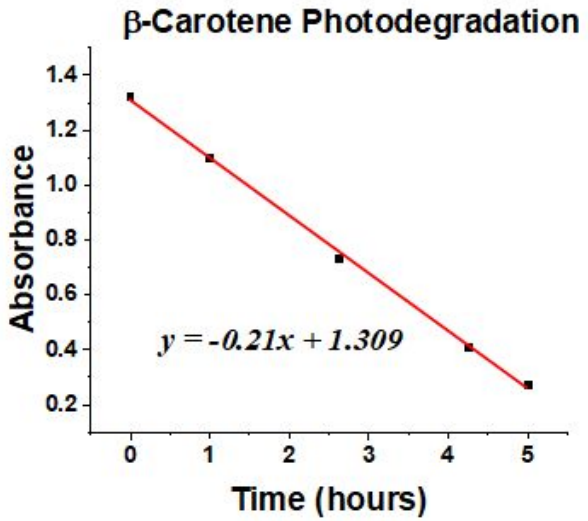


Suit Plot of Photodegradation

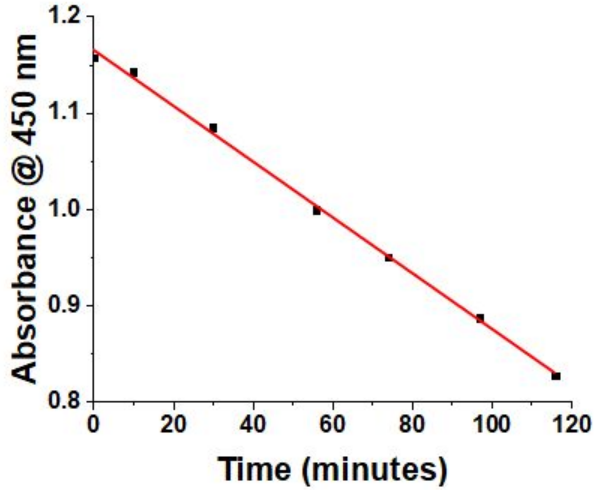


Results - Peak Absorbance

Reference Plot of Peak Absorbance

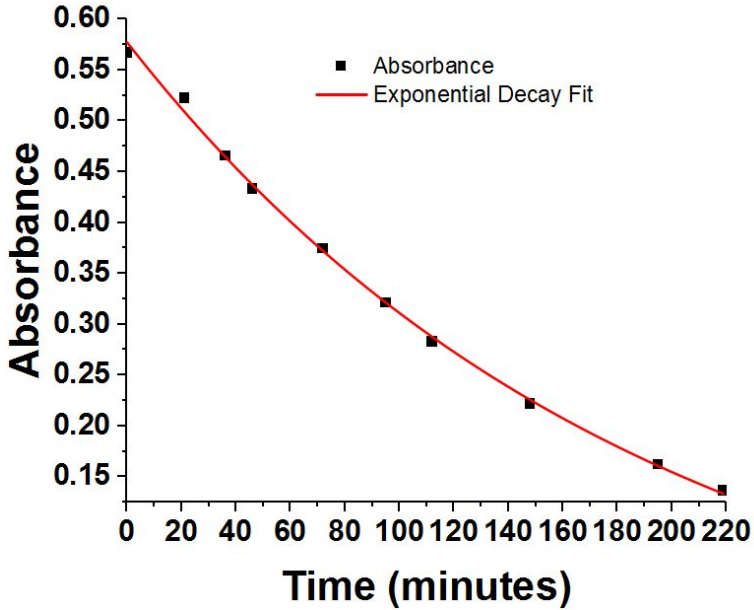


Suit Plot of Peak Absorbance



Results - Abnormality

- Started at lower absorbance and ran a longer trial to get more sensitive results
- Somewhat of an exponential trend seen rather than the expected linear trend
- This is most likely due to small degradations in the output voltage of the battery as it discharges



Ethics

- Major concern is the testing phase
- Test on infants as late as possible by using chemical models and simulations first
- Privacy concerns with gathering data from the suit
- Collect only data that is needed: temperature, and time in use

“...minimize negative consequences of computing, including threats to health, safety” - ACM Code of Ethics

Future Work

Areas of Improvement

- Automatic intensity variation using a set maximum intensity and temperature measurements
- Use of more suitable illumination technology (compact fiber optics, lower intensity LED strips or sheets, higher efficiency lighting)

Future Iteration Features (Textile Engineering)

- Integrate electronics into the outer shell of the suit in e-textiles
- Fabricate an inner shell to be replaced between uses that integrates illumination or blocks heat from external light sources

Expand IoT integration to allow for bluetooth communication in addition to WiFi

Acknowledgements

Health Maker Project Lead: Yusef Shari'ati

TA: William Zhang

Professor: Jing Jiang

Questions?