Modeling Wearable Technology Using Internet of Things For Effective Health Education and Monitoring

Nathanael Bowley
Dalhousie University
nt620579@dal.ca

Nupur
Dalhousie University
dal.ca

Kritika
Dalhousie University
@dal.ca

Mahsa
Dalhousie University
@dal.ca

Vishwa
Dalhousie University
@dal.ca

***Abstract*—**

***Index Terms*—Health Education, Health Monitoring, Internet of Things (IoT), Smart Wearable Devices, Smart Devices**

I. Introduction

Artificial Intelligence (AI) is fillout infomation . Generative Pre-training Transformers (GPT) are a new type of AI that is rapidly gaining popularity and importance in multiple fields of research. One such example is the growth of applicability and frequency of use of GPT technologies in healthcare settings [1]. Generative AI and the technology "ChatGPT" by OpenAI is used by [1] as a reference to create a user-orientated Taxonomy for classifying previous Generative AI usage in the medical literature in terms of application-oriented and user-oriented approaches. Although medical education has seen a great deal of attention in terms of "medical examinations", "student self-study", and composition of "medical questions", little has been pursued in the perspective of improving the health education from the patients perspective, nor has any work been shown in the literature for medical monitoring [1]. Given that ChatGPT and other generative AI can

II. Generative Artificial Intelligence in Health Care

III. Health Education

As medical apps change the way patients view their health and wellbeing, physicians must adapt and pivot with this consideration in mind. Patients and consumers are using fitness measurable technologies (like Fitbit), using medical apps on a regular basis, and having more control over their own health. Patients want more control over their health, and we have to work together to provide this in the best way. The best medical apps work to serve as a hub for the patient: managing everything pertinent to their recovery and maintenance of wellbeing. There are currently patient-oriented apps that help with: • Scheduling Appointments; • Notifying if the physician will be late; • Guiding different treatment plans; • Providing diagnostic historical data.

IV. Health Monitoring

V. Internet of Things

A. Limitations of IoT

A major limitation of consumer IoT devices is the lack of computational resources, as well as issues with data storage, transmission, analysis and sharing [2]. Due to these limitations the need for additional computational power is needed. Instead of processing on device a common approach is usage of Big Data, and cloud networking [3]. Due to these issues mentioned a method is needed to address this issue.

B. Monogramming

Monogramming is a technology for IoT communication invented by Dr. Ohno of Japan [4]. Monogramming is derived from Japanese words meaning "Thing", and "Programming", thus programming of things [4]. Monogramming uses a unique approach to connect IoT devices over the internet in a cheap and affordable way by host networks such as the open source website hivemq.com as one example [4].

VI. Model Design

The model design starts at the IoT device that mocks a wearable medical device. Due to financial constraints the most feasible materials to mock wearable medical bracelets that can track medical details such as heart rate, blood oxygenation levels, and body temperature would be customize made IoT devices provided. In Figure **??** a MOKA device can be seen which has switches, diodes, and regulator knobs for emulating a medical wearable tracking device.

Inside of it is installed the following code:

```
//
// Written By: Nathanael Bowley
// For: GenAI Hackathon at Dalhousie
    University
// 1/20/2024
//

long dummyNum = 100;
const int SWpins[3] = {A3, A4, A5};
const int POTpins[3] = {A0, A1, A2};

int LEDs[3] = {0,0,0};
const int LEDpins[3] = {11,10,9};
```

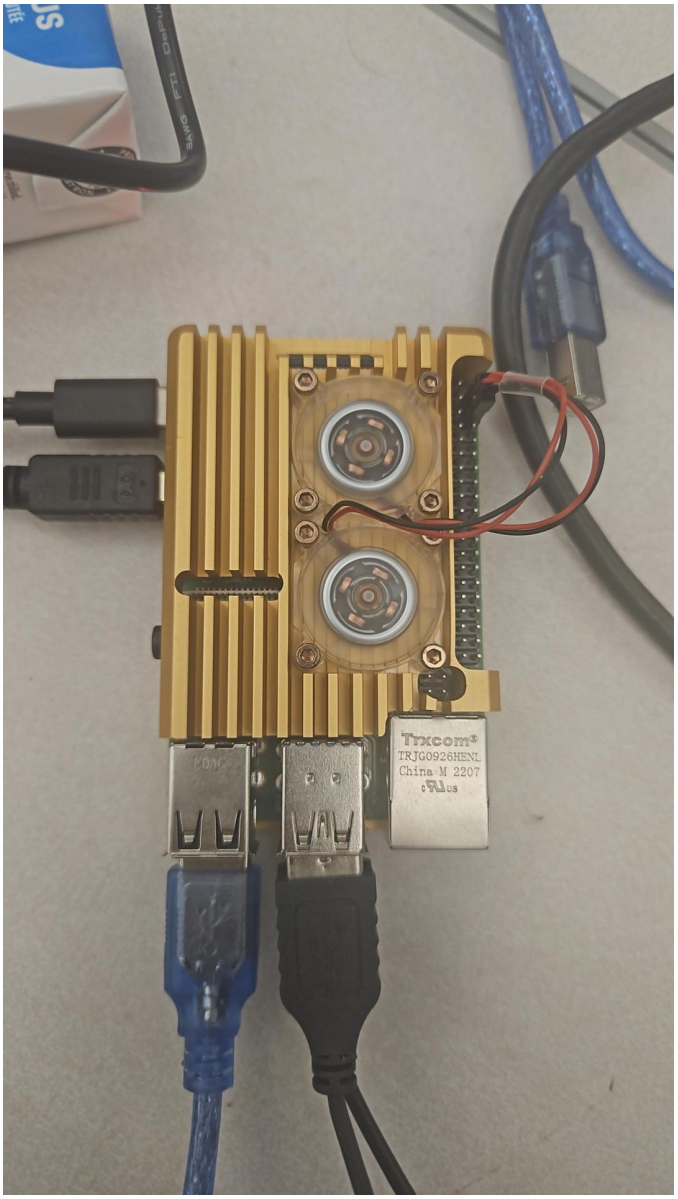

Fig. 1. Test

```
float maxSensorVal = 1023;
float maxBrightness = 255;
float ratioSensorVal;
int brightness;

void setup() {
  Serial.begin(57600);
  pinMode(SWpins[0], INPUT_PULLUP);
  pinMode(SWpins[1], INPUT_PULLUP);
  pinMode(SWpins[2], INPUT_PULLUP);

  pinMode(LEDpins[0], OUTPUT);
  pinMode(LEDpins[1], OUTPUT);
  pinMode(LEDpins[2], OUTPUT);
  randomSeed(micros());
}

void loop() {
  Serial.print("DataforPatient:1");
  Serial.print(",Time:");
      Serial.print(millis());
  Serial.print(",RequestHelp:");
      Serial.print(!digitalRead(SWpins[0]));
  Serial.print(",HeartRate:");
      Serial.print(analogRead(POTpins[0]));
  setBrightness(0);

  Serial.print(",AllGood:");
      Serial.print(digitalRead(SWpins[1]));
  Serial.print(",BloodPressure:");
      Serial.print(analogRead(POTpins[1]));
  setBrightness(1);

  Serial.print(",HungryandorWaterPlease:");
      Serial.print(!digitalRead(SWpins[2]));
  Serial.print(",BodyTemperature:");
      Serial.print(analogRead(POTpins[2]));
      Serial.print(";");
  setBrightness(2);

  Serial.println();
  delay(200);
}

void setBrightness(int switchNumber) {
  ratioSensorVal =
      analogRead(POTpins[switchNumber]) /
      maxSensorVal;
  brightness = floor(ratioSensorVal *
      maxBrightness);
  analogWrite(LEDpins[switchNumber],
      maxBrightness - brightness);
}
```

Attached to it as seen in Figure 2 is the Arduino Uno device running a Linux based operating system for Arduino devices, where the Monogramming commands are ran.

The following command is used to send commands from from the central hub to the hivemq networking service:
(cat -u | sudo cu -s 57600 -l /dev/ttyACM4) | mosquitto_pub -l -t mytechies/patients -h 0eee6950295f4d0bab0daafa4e48a781.s2.eu.hivemq.cloud -p 8883 -u mytechies -P Mytechies1
The full framework designed is shown in Figure 3, with python server acting as the central communication point between Web App, Ollama Generative AI software, and MQTT Cloud.

A. *Features*

Real time health monitoring and alerting. Offloading of commands for reducing computational load Usage of Generative AI to provide insights to health

B. *Challenges Faced*

We had some trouble connecting Monogramming with the api for the hivemq.

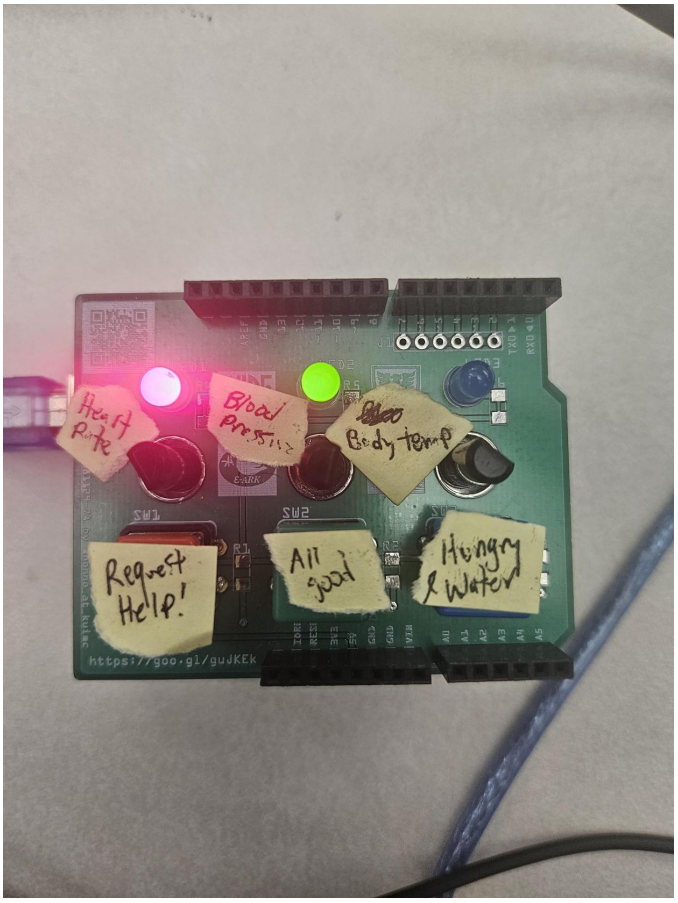

Fig. 2. Test

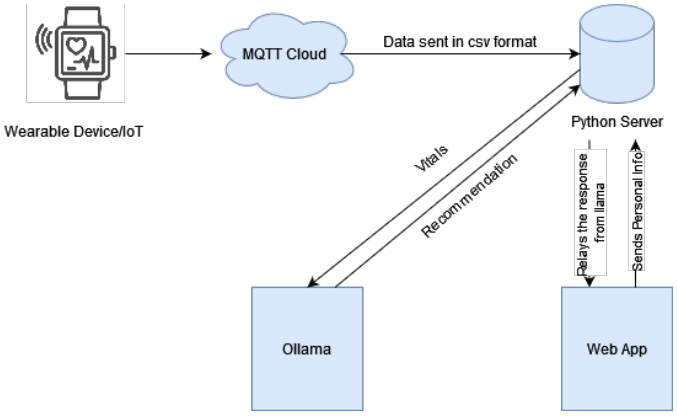

Fig. 3. Test

VII. Future Work

We hope to improve health monitoring of the software.

VIII. Conclusion

In conclusion we created a valuable wearable IoT device that communicates with Ollama by offloading via cloud to a more powerful device to provide Health education to patients.

References

[1] J. Li, A. Dada, B. Puladi, J. Kleesiek, and J. Egger, "Chatgpt in healthcare: A taxonomy and systematic review," *Computer Methods and Programs in Biomedicine*, p. 108013, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260724000087

[2] M. Wang, Y. Sun, H. Sun, and B. Zhang, "Security issues on industrial internet of things: Overview and challenges," *Computers*, vol. 12, no. 12, 2023. [Online]. Available: https://www.mdpi.com/2073-431X/12/12/256

[3] M. Serror, S. Hack, M. Henze, M. Schuba, and K. Wehrle, "Challenges and opportunities in securing the industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 2985–2996, 2021.

[4] H. Ohno, "Internet of things workshop with dr. hiroyuki ohno(monogramming workshop 2308)," 2023, this is a refined version of his work, prior versions of Monogramming came out in previous years in Japan from Dr. Ohno.