Personal Health and Wellbeing System Project Report

Table of Contents

[Project Specification and Motivation 2](#_Toc179921696)

[What is the goal of the project? 2](#_Toc179921697)

[What is the motivation? 2](#_Toc179921698)

[How does this project benefit in the real-world? 2](#_Toc179921699)

[Examples of applications utilising this project idea. 2](#_Toc179921700)

[How does this project improve or enhance the examples to drive solving a real-world problem? 3](#_Toc179921701)

[Hardware Design and Configuration 3](#_Toc179921702)

[What is your hardware architecture? 3](#_Toc179921703)

[Hardware used for the personal health and wellbeing system. 4](#_Toc179921704)

[Why do you design before using hardware? 4](#_Toc179921705)

[Hardware specification. 4](#_Toc179921706)

[Alternative hardware that could be used. 4](#_Toc179921707)

[How does the hardware used relate to the technologies introduced in the course? 5](#_Toc179921708)

[What experimentations have been done? What are the results? Have you done any testing? 5](#_Toc179921709)

[What challenges have you faced? How can these challenges be solved or mitigated? What could be done better? 7](#_Toc179921710)

[Software and Networking Design 8](#_Toc179921711)

[What is your software/networking architecture and how does this link to the hardware architecture? 8](#_Toc179921712)

[What is the software and networking used for the personal health and wellbeing system? 8](#_Toc179921713)

[Why do you design before using software and networking? Provide software and networking specification/design/implementation. 8](#_Toc179921714)

[Are there any alternative software and networking that could be used? 9](#_Toc179921715)

[How does the software and networking used relate to the technologies introduced in the course? 9](#_Toc179921716)

[What experimentations have been done? What are the results? Have you done functional testing or any test cases? 9](#_Toc179921717)

[What challenges have you faced? How can these challenges be solved or mitigated? What could be done better? 11](#_Toc179921718)

[Conclusion and Future Work 12](#_Toc179921719)

[Main reflection of the project 12](#_Toc179921720)

[Final comment on the project 12](#_Toc179921721)

[Suggestions for future improvement. 12](#_Toc179921722)

[References 12](#_Toc179921723)

**I acknowledge the use of Chat GPT in this assessment for proof reading my work to find typos and grammar inconsistencies.**

# Project Specification and Motivation

## What is the goal of the project?

The goal of this project is to develop a personal health monitoring system that tracks and displays vital health data, allowing users to view trends over time. It will include the following key features;

* + Measure and display real-time heart rate and blood oxygen levels.
  + Alert users with a flashing LED if levels exceed thresholds, otherwise display a steady green LED for normal readings.
  + Incorporate a push button to record and upload heart rate and blood oxygen data to the cloud.
  + Automatically save previous historical data to the cloud daily.

## What is the motivation?

The motivation behind this project is to explore how IoT devices can enhance healthcare by not only saving lives but also improving overall well-being and quality of life. Research shows that a key challenge in adopting IoT-enabled healthcare solutions is the significant impact of technology acceptance and innovation diffusion on users' willingness to adopt these products (Karahoca et al., 2018). Thus, this project aims to develop an effective healthcare monitoring system while also highlighting the importance of embracing technological innovation for its successful adoption.

## How does this project benefit in the real-world?

Many IoT projects offer significant real-world benefits, and this project is no exception. It automates the collection of critical health data, stores it daily, and alerts users when necessary. Research shows that IoT devices generate and exchange vast amounts of data, making daily life more convenient, aiding decision-making, and delivering valuable services (Kraijak et al., 2015). This project, in particular, improves the lives of individuals with heart rate or blood oxygen concerns by leveraging IoT technology to bridge the gap in health monitoring and management.

## Examples of applications utilising this project idea.

The ‘Smart Mirror’ project utilises non-contact technological innovations, universally accessed by simply looking into a mirror, while providing critical health insights to improve care. Main objectives of the project are to detect physiological markers using video input from the individual standing in front of the mirror, to then display the results in real-time (AbdElnapi et al., 2018).

Another example of an application that utilises this project idea is an IoT-based health monitoring system for emergency medical services, specifically for use in intensive care units. It uses an Intel Galileo 2nd generation development board to collect, integrate, and interoperate IoT data flexibility, allowing real-time health monitoring. This application helps to reduce health risks and costs by efficiently collecting, recording, analysing, and sharing large data quantities. It aims to reduce the need for patients to frequently visit doctors for basic health checks like blood pressure, heart rate, and temperature (Gupta et al., 2016).

An IoT-based adverse drug reaction system identifies each medicine by a unique barcode from the patient’s terminal. From here the system then checks drug compatibility by using a pharmaceutical intelligent information system, storing the patient’s allergy profile. Based on this allergy profile and other health data, the system then determines whether the medicine is safe for the patient (Pradhan et al., 2021).

## How does this project improve or enhance the examples to drive solving a real-world problem?

Our project enhances the previous existing examples by offering a much more user-friendly and flexible solution towards personal health monitoring. It automates real-time data collection, provides immediate alerts regarding health situations, and stores historical data, unlike the ‘Smart Mirror’ example which requires stationary use. Compared to IoT-based intensive care unit systems, the project brings similar benefits into a home-based environment. This is achieved through empowering users to monitor their health trends to reduce doctor visits, making more proactive health management more accessible.

# Hardware Design and Configuration

## What is your hardware architecture?

In the context of IoT, hardware architecture refers to the physical layout that ensures the seamless interconnection of devices, sensors, communication modules, and other hardware components. This architecture enables the collection, processing, transmission, and action on data in an organized manner (Evans, 2024). For example, in this system, the heart rate and blood oxygen level sensor generates the data that triggers responses from the rest of the system. Based on the readings, the LEDs respond accordingly (flashing red for elevated levels and displaying a stable green for normal levels). Additionally, a button allows the user to record and save the current sensor data, enabling on-demand data storage based on user input.

## Hardware used for the personal health and wellbeing system.

* 1x Raspberry Pi
* 1x Heart Rate/Blood Oxygen Level Sensor
* 1x Red LED
* 1x Multi-Colour LED (Green)
* 1x Push Button
* 1x Breadboard
* 2x Resistors
* 10x Jumper Cables

## Why do you design before using hardware?

Design is crucial before implementing the actual hardware, this is because unless a design is constructed, there is little to no knowledge of how the components will communicate to one another the best possible way. Expanding from this, designing before the use of the required hardware means there is a clear vision, optimal selection of components, minimised costs and wastage, scalability and flexibility, and a reduction of risks.

## Hardware specification.

* Raspberry Pi 3B (2015\*)
* XC3740 Jaycar Heartbeat Sensor
* LED (Red)
* LED (Multi Colour)
* Tactical Pushbutton
* Breadboard
* 330 OHM Resistors
* MF Jumpers

## Alternative hardware that could be used.

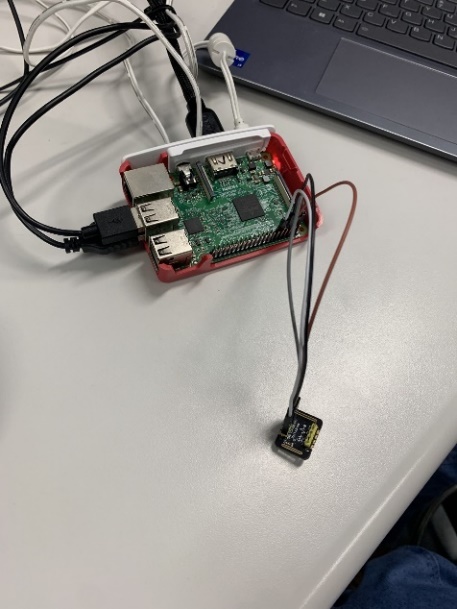
Alternative hardware options could be considered if the project were to be revised, including variants of the existing components. For instance, using a larger breadboard would provide more space for additional hardware, preventing overcrowding. A different heartbeat sensor, such as the KY-039 finger pulse sensor, could replace the XC3740 Jaycar sensor for more precise readings. Additionally, the pushbutton could be swapped for a switch, offering a more intuitive on/off control for starting and stopping data recording. Finally, instead of LEDs, a buzzer could be used to alert users—emitting a continuous sound when thresholds are breached and remaining silent when conditions are stable.

## How does the hardware used relate to the technologies introduced in the course?

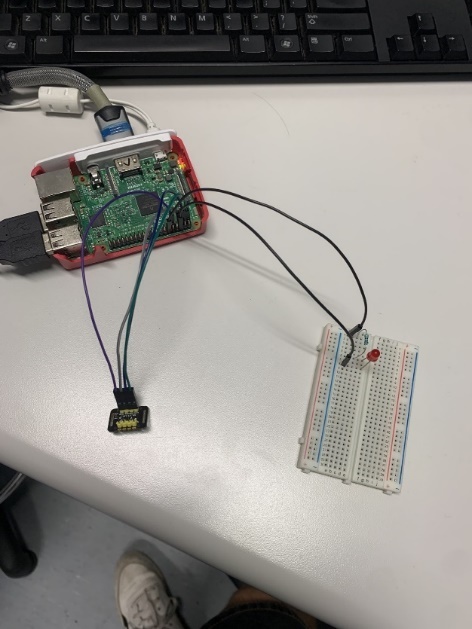
One of the most important technologies introduced in this course is the Raspberry Pi, a microprocessor often considered the heart of any IoT products. It processes raw data from sensors, extracting valuable insights. Every hardware component connects to the Raspberry Pi, as without this processing unit, the gathered data would have no value (1NCE, 2024). The Raspberry Pi's System on Chip (SoC) integrates key functions into a single chip, optimizing space and minimizing component usage.

In addition to the Raspberry Pi and its SoC, other technologies introduced in the course including sensors (e.g. heart rate or humidity sensors), directly communicate with the Raspberry Pi to collect and transmit recorded data. Communication modules (Wi-Fi and Bluetooth), allow the system to send processed data to cloud services or other physical devices with real-time monitoring and analysis. Furthermore, other hardware components like pushbuttons or LEDs serve simply as user interface elements, giving feedback based on interaction. Altogether, the hardware components, along with the Raspberry Pi, form the core of any IoT system through collecting, processing, and transmitting critical data, aligning with the technologies and principles covered throughout this course.

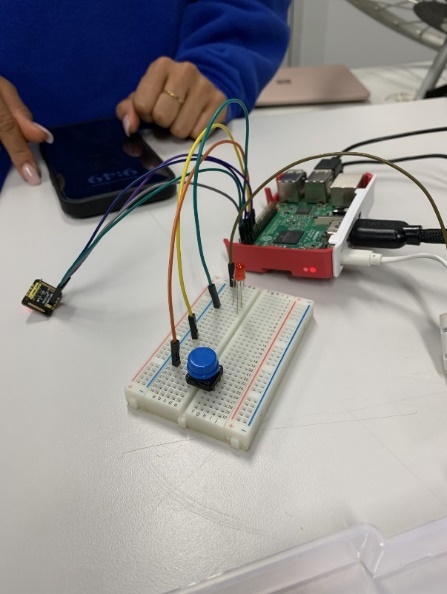
## What experimentations have been done? What are the results? Have you done any testing?

Several experiments were conducted on the hardware, with each experiment broken down into tasks aligned with the objectives of the IoT project. The first task was to ensure the heart rate and blood oxygen sensor displayed real-time metrics. This was tested by having different group members use the sensor to verify the accuracy and consistency of the readings. Below is the setup used to test this first requirement:

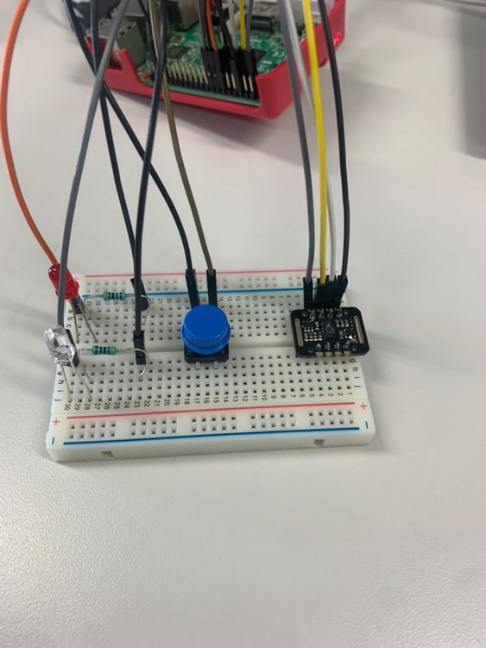
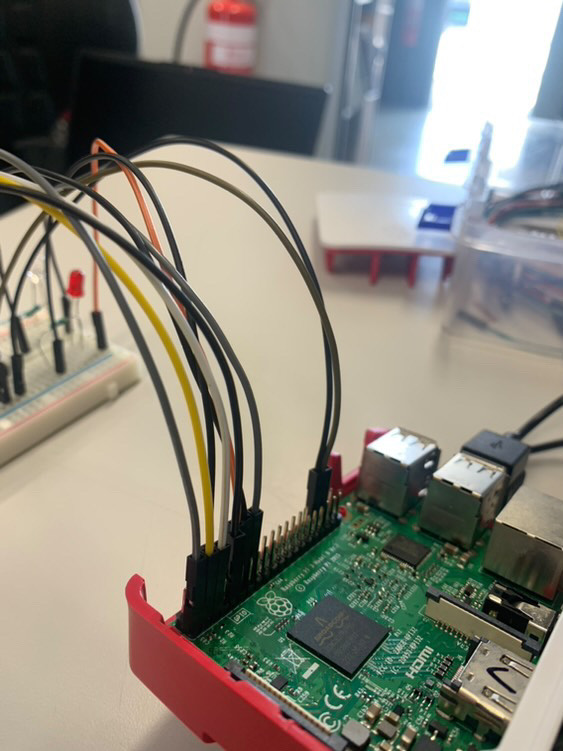
*Figure 1: Heart Rate and Blood Oxygen Sensor.*

The second task was to configure the hardware so that when a user's heart rate or blood oxygen level exceeded a specific threshold, a red LED would flash; otherwise, a green LED would remain steady. To test this, we established test thresholds and experimented with the hardware setup to ensure it functioned correctly. The hardware configuration used for this test is shown below:

*Figure 2: LED Hardware Setup.*

Finally, the last hardware experiment involved incorporating a push button to record heart rate and blood oxygen data, which would then be saved to the cloud. This was achieved by linking the push button to the sensor, ensuring that when pressed, the recorded data was saved as intended. The hardware configuration for this setup is shown below:

*Figure 3: Push Button Hardware Setup.*

Below are pictures showing every hardware component needed for the IoT projects interconnected as one system:

*Figures 4 and 5: Final Hardware Setup.*

## What challenges have you faced? How can these challenges be solved or mitigated? What could be done better?

The challenges faced during hardware development were not primarily with the individual setups (sensor, LED, and push button configurations), but rather with integrating all three components to communicate smoothly and efficiently. Difficulty arose from ensuring that the interaction between one component, such as the push button, didn’t interfere with the functioning of others, like the LEDs, especially when all elements were operating together in the same environment.

One approach to addressing these challenges is to develop each hardware component as independent modules, which can help minimize cross-interference and facilitate easier troubleshooting and testing. Additionally, implementing debouncing techniques for the push button can prevent false triggers that might interfere with the LED or sensor functions; this can be achieved by using a capacitor. Furthermore, incorporating proper shielding and signal isolation can help prevent electrical interference between components, ensuring smooth operation.

To enhance the hardware setup, the testing of each component could have been more thorough. This was necessary to ensure that each section was functioning correctly and in harmony with the others. Moving forward, I recognize the importance of dedicating more time during the design phase to critically evaluate and improve the functionality and ease of integration of all components.

# Software and Networking Design

## What is your software/networking architecture and how does this link to the hardware architecture?

In IoT, software architecture involves analysing system qualities and making decisions early in the development process, rather than during or after implementation (Carnegie Mellon University, 2024). Networking architecture focuses on how IT infrastructure is organized to securely and efficiently transfer data between devices and applications (Siddiqui, 2024). In our system, the software and network architecture ensures that data from the heart rate and blood oxygen sensor is processed by the LEDs, which either flash red or display a stable green light. The push button allows data to be saved to the cloud based on user input, with the system programmed to automatically save data daily if new data exists. The hardware architecture relies on these software and network inputs to perform tasks based on the received data.

## What is the software and networking used for the personal health and wellbeing system?

Software used for the IoT project include:

* Linux; is used as an open source operation system (OS) (Raspberry Pi Ltd, 2020).
* Raspbian; is a Unix like operating system integrated from the Debian Linux distribution (Raspberry Pi Ltd, 2024).
* Python; is the default program language.

Networking used for the IoT project include:

* Wi-Fi connectivity; for data transfers and connecting to the internet.
* Cloud database; for storing, processing, and retrieving data.
* Ngrok; is used for enabling remote access to the IoT device with the internet.

## Why do you design before using software and networking? Provide software and networking specification/design/implementation.

Specifying the software and networking is crucial to ensure that the system operates correctly, securely, and meets its intended objectives. For our personal health IoT project, the specification phase helped identify system requirements, prevent malfunctions or data inaccuracies, optimise resources, and ensure privacy and security measures are in place.

The design phase focused on ensuring efficient, secure, and reliable operation. This process clearly defined essential features like real-time monitoring, data storage, and alerts.

Finally, implementation involved processing sensor data, triggering alerts via LEDs, and recording data to the cloud with a push button. Networking implementation included establishing Wi-Fi connectivity and securely storing data in the cloud.

## Are there any alternative software and networking that could be used?

Alternative software solutions could be that instead of using Python on a Raspberry Pi, the use of an Arduino IDE could be used for programming either an ESP32 or ESP8266 microcontroller. Another part of the software that could be used instead is the use of a message queuing telemetry transport (MQTT) protocol, replacing use of any traditional HTTP-based APIs. Lastly, Node-Red could be implemented for simplifying integration with external services such as cloud storage or data visualisation tools.

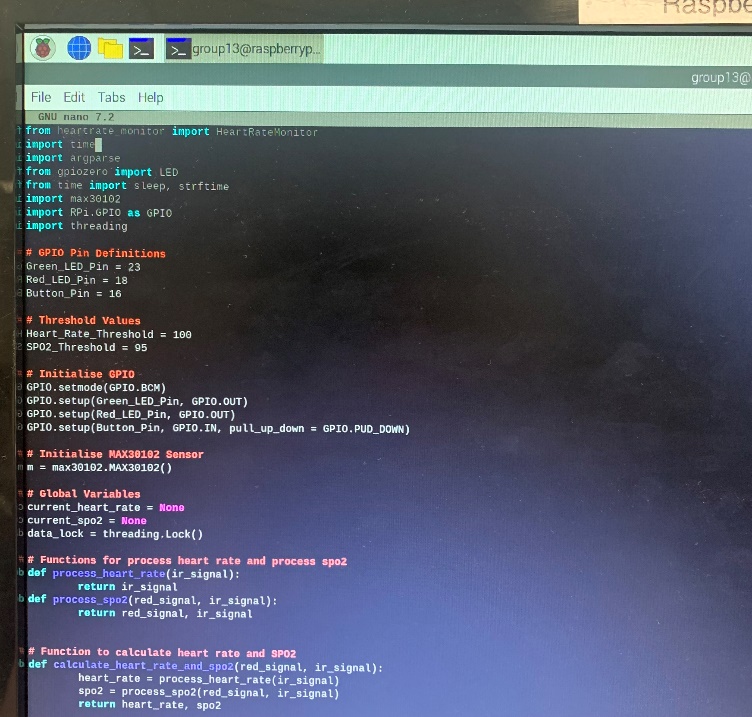
Whereas, for networking alternative options include using a long range wide area network (LoRaWAN) where Wi-Fi is not an option largely due to long-range. Another alternative networking option could be to use cellular networks such as 4G or 5G where once again Wi-Fi is not feasible. One more option may be to use Bluetooth low energy for short-range data transmission between a sensor and a nearby device.

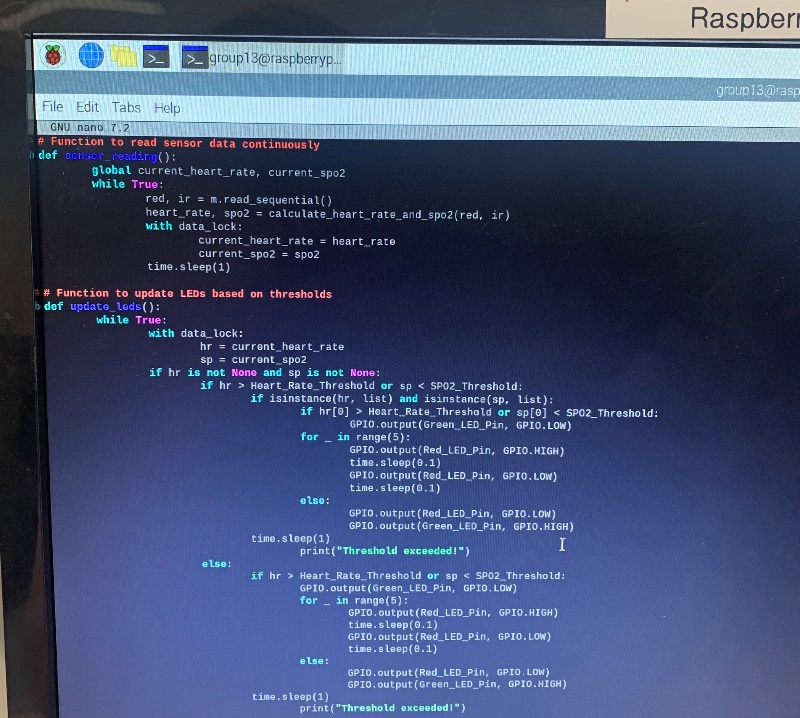
## How does the software and networking used relate to the technologies introduced in the course?

The software and networking used in this project closely align with the technologies introduced in the course. The Raspberry Pi serves as the central hub, handling sensor data processing and communication with the cloud, illustrating the role of microprocessors in IoT. The project also utilises web-based technologies, another key topic from the course, as IoT devices require internet connectivity to interact with other devices and cloud services. Additionally, cloud computing is directly linked to the software and networking setup, as it enables data storage and remote access via the internet, allowing real-time monitoring and analysis.

## What experimentations have been done? What are the results? Have you done functional testing or any test cases?

Experiments on the software and networking were conducted in sections, each focused on integrating hardware functionality and testing the results. At this stage of the project, not all basic feature requirements were fully achieved. However, with more time, the software and networking components could have been further refined to create a fully developed and functional system. Below is the code we developed for the hardware setup we created:

*Figure 6: First section of code used.*

**

*Figure 7: Second section of code used.*

*A computer screen with text on it

Description automatically generatedFigure 8: Third section of code used.*

## What challenges have you faced? How can these challenges be solved or mitigated? What could be done better?

The primary challenges we faced involved configuring the push button to send recorded data to the cloud and ensuring the code transitioned smoothly between sections without errors. It was particularly difficult to use the button solely for saving data, rather than triggering an action. Additionally, hardware-related issues often disrupted the code’s flow, preventing it from running as expected.

To mitigate these challenges, separating the push button's action and save functions into two distinct components or using an on/off switch could simplify the process. Optimising the code flow by using pre-existing code or APIs from similar IoT projects would also provide more efficient solutions. Testing each hardware component individually, followed by integration testing, would help isolate and address hardware-related issues affecting the software and networking.

To improve the process, more time should have been dedicated to thorough testing at each development stage. Additionally, designing a more robust system architecture with clearly defined roles and interactions for each component would have minimized integration challenges.

# Conclusion and Future Work

## Main reflection of the project

In summary, the hardware components of the project were successfully completed, meeting the necessary requirements for the personal health and wellbeing system. However, the software portion was only partially finished due to various minor errors and limitations, which restricted its full functionality. The networking aspect proved to be the most challenging, primarily because of time constraints and similar limitations encountered with the software.

## Final comment on the project

As a final comment on our IoT project, the biggest limitation that impacted our ability to fully complete the project was the recurring software and networking errors. These issues consumed a significant amount of time, ultimately preventing us from finishing the project as planned.

## Suggestions for future improvement.

Below are possible suggestions for future improvement of our personal health and wellbeing system IoT project:

* Restart the code to become more modular focused, separating each key function into a distinct module, making it easier to debug.
* Leverage the use of pre-built IoT APIs or libraries already designed for similar health monitoring to accelerate and improve development.
* Implement a better error detection and handling mechanism, eliminating the system from stalling due to minor issues.

# References

* 1NCE. (2024). IoT hardware and its place in the iot ecosystem. 1NCE. https://1nce.com/en-ap/resources/news-insights/blog/iot-hardware-and-ecosystem
* AbdElnapi, N. M. M., Omran, N. F., Ali, A. A., & Omara, F. A. (2018, February). A survey of internet of things technologies and projects for healthcare services. In 2018 International Conference on Innovative Trends in Computer Engineering (ITCE) (pp. 48-55). IEEE.
* Carnegie Mellon University Software Engineering Institute. (2024). Software architecture | software engineering institute. Www.sei.cmu.edu. https://www.sei.cmu.edu/our-work/software-architecture/
* Evans, H. (2024, March 12). IoT Hardware Development: Key Considerations Guide. Velvetech. https://www.velvetech.com/blog/iot-hardware-development/
* Gupta, P., Agrawal, D., Chhabra, J., & Dhir, P. K. (2016, March). IoT based smart healthcare kit. In 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT) (pp. 237-242). IEEE.
* Karahoca, A., Karahoca, D., & Aksöz, M. (2018). Examining intention to adopt to internet of things in healthcare technology products. Kybernetes, 47(4), 742–770. https://doi.org/10.1108/k-02-2017-0045
* Pradhan, B., Bhattacharyya, S., & Pal, K. (2021). IoT-Based applications in healthcare devices. Journal of Healthcare Engineering, 2021(1), 1–18. https://doi.org/10.1155/2021/6632599
* Raspberry Pi Ltd. (2020). Linux v. raspbian? - raspberry pi forums. Raspberrypi.com. https://forums.raspberrypi.com/viewtopic.php?t=291809
* Raspberry Pi Ltd. (2024). Raspberry pi OS – raspberry pi. Raspberry Pi. https://raspberrypi.com/software/
* Siddiqui, L. (2024, May 16). What is network architecture? Splunk. https://www.splunk.com/en\_us/blog/learn/network-architecture.html
* S. Kraijak and P. Tuwanut, (2015). A survey on IoT architectures, protocols, applications, security, privacy, real-world implementation and future trends. 11th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2015), Shanghai, pp. 1-6, doi: 10.1049/cp.2015.0714.