© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 14 Issue: 03 | April - June 2023
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

Special Edition
NCASIT 2023, 29th April 2023
Department of Computer Engineering,
St. Vincent Pallotti College of Engineering & Technology, Nagpur,

COMMUNICATING REACT NATIVE WITH NodeMCU

Aditya Arikar , Aditya Doye , Harsh Nipane , Om Lanjewar ,Khushboo Khobragade
UG students, Department of Computer Engineering
Dr . Abhishek Pathak
Professor , Department of Computer Engineering
St. Vincent Pallotti College of Engineering & Technology, Nagpur India

Abstract

As we can see a handful of gadgets such as PCs andsmartphones are internet-connected in the modern world. The Internet of Things (IoT) and the internethave completely taken over today's world. While it'strue that a lot of people use the internet to satisfy necessities like communication, information exchange, and leisure, not all requirements call for using the internet. Monitoring physical hardware ordevices are only a small part of what the IoT can do.A wide amount of businesses and applications couldbe transformed by this extensive and varied networkof interconnected devices. It can also refer to the interconnectivity of physical objects that have been implanted with electronics, sensors, software, and network connectivity. IOT makes it possible to control or sense objects remotely by using network infrastructure. As a result, accuracy, financial gains, and efficiency increase while human intervention decreases.

Keywords :

NodeMCU , ReactNative , WiFi , Bluetooth , MQTT , Node.js

1. INTRODUCTION [1,6]

The way we connect with the world around us has completely changed as a result of the Internet of Things (IoT) being incorporated into our daily lives.The ability to remotely control and monitor devicesis one of the fundamental features of IoT, which has encouraged the creation of several apps that make use of this technology. In this regard, two well-liked platforms for creating IoT systems and mobile applications are NodeMCU and React Native. Basedon the ESP8266 chip, the NodeMCU is a low-cost microcontroller with Wi-Fi functionality. The Arduino IDE may be used to program it, making it apopular option for enthusiasts and hobbyists. On theother hand, React Native is a framework used to create mobile applications for both iOS andAndroid. It offers a mechanism to create apps usingJavaScript, which allows for faster development and simpler maintenance. In this project, our goal is to establish Wi-Fi connection between a NodeMCU and a React Native app. The main goal is to create a system that enables mobile app-based relay control for a NodeMCU-connected relay. To do this, the Wi- Fi module is combined with the React Native app, enabling the app to issue commands to theNodeMCU. The graphical user interface of the application allows the user to toggle the relay after connection has been established. The project servesas a starting point for the creation of more advanced IoT systems and demonstrates the possibilities of employing Wi-Fi to establish communication between devices. It is cost-effective, easy to deploy, and can be modified to suit various applications, making it an attractive option for hobbyists and enthusiasts. Overall, the project demonstrates how the integration of two popular platforms can lead tothe development of innovative solutions for the IoT industry.

2. LITERATURE SURVEY

A well-liked framework for creating mobile applications that work on both iOS and Android devices is called React Native. On the other side, NodeMCU is a development board that connects tothe internet and manages numerous devices using the ESP8266 Wi-Fi module. Without referencing any previously published work, we will examine the many ways that React Native can be used to interface with NodeMCU in this literature review.

React Native and NodeMCU communicate with the use of a number of different technologies, including Bluetooth, Wi-Fi, and MQTT. This overview of the literature examines the advantages and disadvantages of the various methods of communication between React Native and NodeMCU.

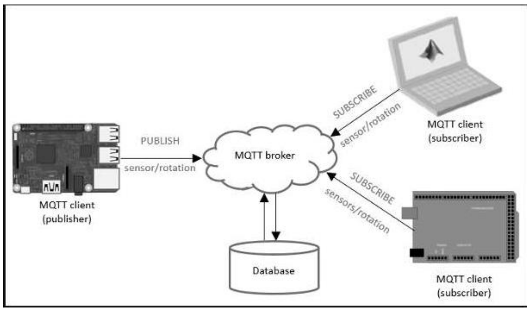
Bluetooth is a wireless technology that enables close-quarters communication between gadgets. It is frequently employed in IoT applications, especially those that make use of NodeMCU. The React Native Bluetooth Low Energy (BLE) library is used to facilitate Bluetooth connection in React Native. The Bluetooth functionality of NodeMCU can be difficult to set up and requires additional hardware.[4]

1. Another wireless technology that makes it possible for IoT devices to connect with one another is Wi-Fi. NodeMCU is a great option for IoT applications because it comes with built-in Wi-Fi connectivity. With Wi-Fi, React Native can communicate with NodeMCU via the HTTP and WebSocket protocols. While very simple to set up, this method needs a strong Wi-Fi connection. [3]

Fig.1 MQTT Publish and MQTT Subscribe blocks

2. A messaging protocol called MQTT (Message Queuing Telemetry Transport) makes it easier for IoT devices to communicate with one another. Due to its efficiency and minimal weight, it is perfect for IoT applications. MQTT works by having a publisher transmit messages to a broker, who then distributes them to a subscriber or subscribers. NodeMCU is a flexible option for IoT applications because it can function as both a publisher and a subscriber. With the MQTT.js package, React Native can connect with NodeMCU over MQTT. [2]

In addition to hardware modules, several software frameworks have been used to facilitate communication between React Native and NodeMCU. One such framework is the Node.js platform, which allows developers to create server-side applications using JavaScript. By using Node.json the server side and React Native on the client side, developers can establish a connection between the two platforms and exchange data



(Gupta et al., 2019). [5]

Fig.2 ESP8266 NodeMCU

React Native and NodeMCU can communicate in a variety of ways. The choice is based on the particular requirements of the project, as each method has benefits and drawbacks of its own. Wi-Fi is suitable for strong and dependable connections, whereasMQTT is portable and effective for short-rangecommunication. It's crucial to pick the righttechnology and build it according to best practices ifyou want React Native and NodeMCU to communicate effectively.

Comparison between wifi and Bluetoothtechnology

A React Native app can be connected to a NodeMCU using both WiFi and Bluetooth modules, but these connections have different ranges, data transfer rates, and battery consumption.

A WiFi module employs the 802.11 protocol and has a greater range than Bluetooth, typically up to 100 metres, as well as a high data transmission rate, typically ranging from 150 Mbps to 1 Gbps. Since WiFi modules use more power than Bluetooth modules do, they might not be the greatest option fordevices that run on batteries. For streaming data, such as video and music, they are an excellent option because they have a higher bandwidth and reduced latency.

In comparison to WiFi, Bluetooth protocol providesa lower data transfer rate, often between 1 and 3 Mbps, and a shorter range, typically up to 10 meters.Bluetooth modules are an acceptable option for battery-powered devices because they use less power than WiFi modules. Although they have a smaller bandwidth, they work well for sending littleamounts of data, such as sensor data.

Hence, WiFi module can be a better option if you need to transport a lot of data quickly and have a reliable power source. A Bluetooth module might bea better option, though, if you need to transfer only less amount of data or want to save power, while MQTT is suitable for lightweight messaging between multiple devices, regardless of the wirelesscommunication technology used.

3. DISCUSSION

Communicating React Native with NodeMCU can be accomplished through several methods. Using Wi-Fi or Bluetooth as a communication protocol between the two devices is one such method. You may accomplish this by deploying Node.js on the NodeMCU device to create a custom API, and the React Native application can communicate with it by sending HTTP requests. This enables the NodeMCU device to receive data from the React Native application and conduct actions based on the data As an alternative, you can communicate with the NodeMCU device over Bluetooth by using a third-party library like "react-native-ble-manager." The React Native application and the NodeMCU device can send and receive data using this library's straightforward interface for connecting to and communicating with Bluetooth devices. Security is a critical factor to take into account when communicating with NodeMCU, it is necessary to note. To prevent unauthorised access or data theft, you should make sure that your communication protocol is secure and that all data exchanged between the two devices is encrypted. There are various ways to communicate React Native with NodeMCU, but it can be a difficult effort overall. It'scrucial to carry out thorough study and choose the best approach that satisfies your unique requirements. Also, you should always check your work for piracy and make sure that any sources youreference for your research are properly credited.

4. CONCLUSION

Integrating React Native with NodeMCU is a challenging process that requires significant thoughtand research to ensure the most effective method is chosen. Giving due credit to any sources used during your study and describing the ideas and topics you are discussing in your own terms will help you avoidpiracy. You can use techniques like creating a custom API in Node.js or using a third-party librarylike "react-native-ble-manager" to interact over Bluetooth, "react-native-reborn-wifi" to interact over wifi and "react-native-mqtt" to interact using mqtt. In order to prevent unauthorised access or data theft, it is essential to make sure that your communication protocol is secure and that all data transported between the two devices is encrypted, regardless of the technique you use.

5. REFERENCES

1. A framework for building native apps usingreact. Available: https://reactnative.dev/
2. React Native-Native MQTT 0.1.8 client LibraryAvailable: https://www.npmjs.com/package/react-native-native-mqtt
3. React-Native-WiFi-Reborn 4.0.7 client libraryAvailable: https://www.npmjs.com/package/react-native-wifi-reborn
4. https://www.npmjs.com/package/react-native-ble-manager
5. R. Porkodi and V. Bhuvaneswari, “The Internet of Things (IoT) Applications andCommunication Enabling Technology Standards: An Overview,” in 2014 International Conference on Intelligent ComputingApplications, pp. 324 329, March 2014.
6. https://evothings.com/controlling-nodemcu-wifi-module-with-a-javascript-mobile-app/
7. International Journal of Research inEngineering,Science and Management Volume-2, Issue-2, February-2019. Available:https://www.ijresm.com/Vol.2_2019/Vol2_Iss 2_February19/IJRESM_V2_I2_127.pdf
8. https://medium.com/@AzilenTech/cross- platform-mobile-application-using-react- native-for-iot-devices-7e8da015b05
9. MQTT v3.1 protocol specification.http://public.dhe.ibm.com/software/dw/webser vices/ws-MQTT/MQTTv3r1.html
10. https://thesai.org/Downloads/Volume10No6/P aper_11- Internet_of_Things_IOT_Research_Challenges .pdf
11. Z. Alansari, N. B. Anuar, A. Kamsin, M. R.Belgaum, J. Alshaer, S. Soomro, and M. H. Miraz, “Internet of Things: Infrastructure, Architecture, Security and Privacy”, in 2018 International Conference on Com- puting, Electronics Communications Engineering(iCCECE), pp. 150 155, Aug 2018, DOI: 10.1109/iCCECOME.2018.8658516.
12. Oasis MQTT version 3.1.1. [Online]. Available:http://docs.oasis-open. – org/MQTT/MQTT/v3.1.1/os/mqtt-v3.1.1- os.html
13. D. B. Ansari, A.-U. Rehman, and R. Ali, - cols: A Brief “Internet of Things (IoT) Proto Exploration of MQTT and CoAP,” International Journal of Computer Applications, vol. 179, pp. 9 14, 03 2018.
14. Node mcu an open-source firmware based on esp8266 –wifi-soc. [Online]. Available: http://nodemcu.com/index en.html/
15. S. Rajguru, S. Kinhekar, and S. Pati, “Analysis of internet of things in a smart environment,” International Journal of Enhanced Research in Man-agement and Computer Applications,vol. 4, no. 4, pp. 40 43, 2015.

3. –