Volume 10, Issue 6
December 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Development of IOT Application with Cloud and Raspberry Pi using Node-Red Programming

[1] **Syed Owais Ahmad**, [2] **Sandeep Kumar Jain**, [3]**Neeraj Sharma**

[1]Department of Electronics & Communication Engineering, Vivekananda Global University, Jaipur

[2,3]Department of Electronics & Communication Engineering Vivekananda Global University, Jaipur

E-mail: [1]syedowais@yahoo.co.in, [2]sandeep_jain@vgu.ac.in.

ABSTRACT

With increase in the advancement of the IOT based systems, the concept of IOT has found applications in diverse fields ranging from healthcare, Smart mobility and much more. With constant communication between the IOT based systems, a large number of data is transferred between the systems which is processed somewhere for deducing valid and logical conclusions of the communication. The development of cutting-edge tangible frameworks and shrewd apparatus has brought about a gigantic variety of gadgets creating a wide range of structured and unstructured data. Gathering and understanding that information, joining it with different wellsprings of data types and effectively utilizing it tends to be accomplished by utilizing availability and systems administration between the cloud elements. The cloud right now possesses analytical and cognitive services which allow development and deployment of solutions in a faster and effective rate than ever before. This paper acquaints with applications that improve availability and investigation as a feature of an incorporated IOT stage, utilizing Nod-Red, an open-source visual application advancement climate, on both the gadget and the cloud. This is the means by which we make an essential IOT arrangement by utilizing prebuilt blocks of code that improve and make fast the development process. The use of Nod-Red programming in a cloud based raspberry pie is very low thus the aim of observations would be demonstrated using lower-level programming. The use of IOT application and cloud systems is very useful way of harnessing technology in a positive way.

Keywords: *Data processing, Internet of things, Node-RED, Cloud computing, Smart Machinery*.

1. INTRODUCTION

In last few decades, there has been a sharp increase in the number of elderly peoples around the globe due to the increase in the life expectancy and less deaths due to chronic diseases of human beings. A recent report published by United Nations expected 2 billion people (22% of world population) as elderly people. Also, various medical research surveys and research found that 80% of the elderly people mostly older than 65 years suffer from at-least one chronic disease which makes them difficult and problematic in taking care of themselves [1]. Giving a fair personal satisfaction for older individuals has turned into a genuine social test right now. The quick multiplication of data and correspondence advances is empowering creative medical care arrangements and devices that show guarantee in tending to the previously mentioned challenge. Internet of Things (IOT) as a service has arisen as one of the most impressive correspondence standards of the 21st century. Because of their communication and processing capabilities like data reception and processing (including microcontrollers, automation circuits, transceivers for digital communication, and appropriate protocol stacks). that allow them to interface with other things like smart phones and gadgets of daily use, all items in our daily lives become part of the Internet in the IOT environment. The systematic model is given in fig.1.

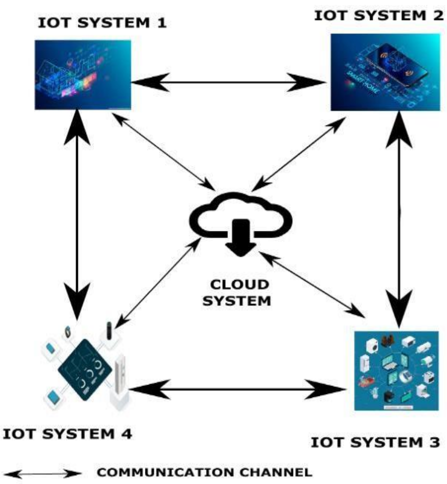

Figure 1: Interconnection between cloud system and IoT system.

IOT broadens the Internet's scope and makes it more widespread. The consistent and basic communications among different kinds of gadgets, including as cars, PCs, computerized watches, clinical sensors, observing cameras, scanners, home machines like fountains, climate control systems, shrewd TVs, etc, have brought about the introduction of numerous applications, including savvy city, home

Volume 10, Issue 6

ITEE Journal

Information Technology & Electrical Engineering

ISSN: - 2306-708X

Computerization, brilliant framework, traffic the board, etc. in the IOT climate. IOT in the field of healthcare involves a diversity of low-cost sensors (wearable, implanted, and environmental) that provides a facility to elderly people to receive medical treatment wherever they are, at any time. They not only provide medical staff with convenience and a simple method to operate gadgets, but they also significantly increase the quality and enjoyment of life for senior individuals.

IOT devices, in addition to standard processing, storage, and communication hardware, must include a range of sensors and actuators to permit interaction between the physical and cyber worlds [2]. Already, a slew of consumer devices, including smart phones, tablets, and wearable's (all of which are now on the market), contain a variety of sensors. Among them are an accelerometer, gyroscope, proximity sensor, compass, barometer, microphone, and camera. Also included are actuators such as a display, vibrator, and speaker. Recently, attempts to develop improved smart phone hardware (mainly to fulfill the different needs of consumers) have demonstrated the ability of smart phone platforms to support IOT applications. IOT devices, on the other hand, aim to serve applications and services that have a global footprint, in contrast to consumer electronics design requirements. And, in order to achieve so a significant number of IOT devices and peripherals must be deployed and operated with very little or minimum human intervention. Basically, most IOT devices are expected to operate and perform for extended periods of time on electric charge in the form of batteries or with limited power supplies. As a result, energy efficiency and energy storage become a critical and important design consideration. The device's operational efficiency is jeopardized by the device's later design priority consideration. Considering the applications of the IOT based device, it can be used in a certain environment where it needs to make some decisions and behave as a smart device. To do so we need to enhance analytical abilities of the IOT based device. Understanding the present scenario of the cloud computing, where an entire virtual machine can be designed in a cloud. The method opens the gates for the IOT based systems to design the cloud-based applications where IOT works as virtual computers. This paper makes an attempt to look into the opportunity and design a model for the same. The remainder of the paper is arranged as follows: Section II is dedicated to the implementation of Node-Red within cloud applications. Section III describes the application of Node-Red on Raspberry Pi. Section IV holds the description of end-to-end implementation of the entire system of programming which is followed by the conclusion of the paper in Section V.

2. RELATED WORK

A. Implementation of Node-Red Application (Within Cloud)

Node-RED is a programming language that allows you to link physical devices, APIs, and web services in unique ways. To accomplish this goal, we propose our Flight Data Analyzer framework, which has 4 main objectives: (1) identify flight-related categories (or clusters) being tweeted/re-tweeted, (2) gather and assess partial and exact information, (3) ascertain possible correlation of Twitter data with weather condition Node-RED features a browser-based flow editor for connecting flows using the palette's various nodes. With a single click, the flows may be pushed to the runtime. JavaScript functions may be created within a rich text editor. You may store and reuse valuable functions, templates, or flows using the built-in library. It started as an open-source task and programming at IBM in late 2013 to address their issue to rapidly associate equipment and gadgets to web administrations and other programming - as a kind of IoT stick - however has in short order developed into a broadly useful IoT programming apparatus [3]. Critically, Node-RED has in short order grown an enormous and developing client base, as well as a functioning engineer local area that is continually adding new hubs that permit software engineers to reuse Node-RED code for an assortment of assignments and projects.

B. Node RED Application to generate random numbers (within cloud).

In this application, we have created a Node-Red flow to generate a random number between 0 and 10 inclusive. The random number generator should start without a user having to initiate the flow and a new number should be generated every 5 seconds. My application flow will output the following to the debug console:

• If the random number is less than 5 then output to the debug console:

Number <the generated random number> is a low number

• If the random number is 5 or higher then output to the debug console:

Number <the generated number> is a high number

Where <the generated number> should be replaced with the random number generated

Number 6 is a high number

Number 2 is a low number

The process flow of the implementation is shown in figure 1 and the console results are shown in figure 2

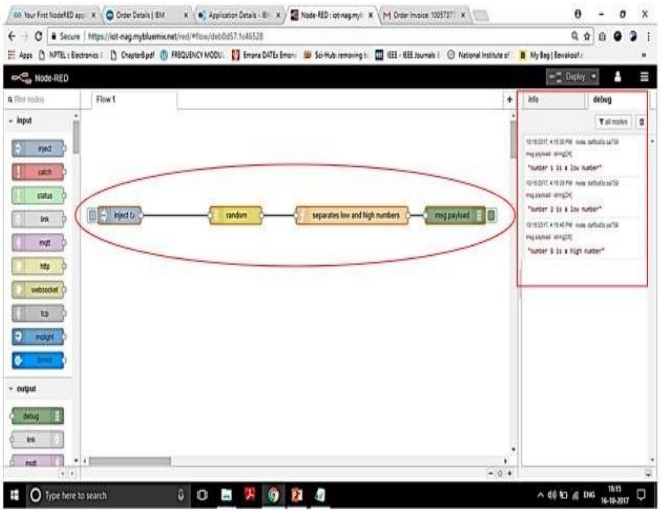

Figure 2: Overview of Node-RED flow and debug console for generating random numbers

C. Node-Red Application to serve http GET request (within cloud)

In this section, we have created an application using Node-Red that will accept an http GET request to http:// <my-node-red-instance>/time that returns a JSON document containing the current time on the Node-Red server.

Example: <my-node-red-instance>= iot-nag.mybluemix.net

An http GET request to http://<my-node-red-instance>/random that returns a JSON document containing a generated random number.

An http GET request to http:// <my-node-red-instance>/page that return an http page that shows:

When the last time was requested.

When the last random number was requested.

The value of the last random number generated.

It is important that the time and random number reported by the /page request match exactly the values last returned from /time and /random requests. So, I have used facilities available in Node-RED to hold these values.

Node-RED code to serve http GET request:

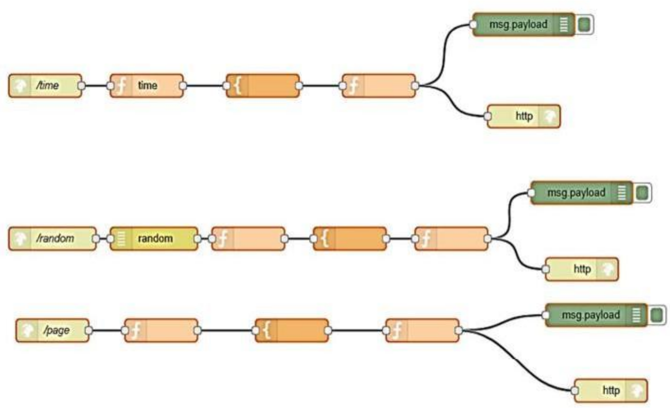

Figure 3: Node-RED flows to serve http GET requests.

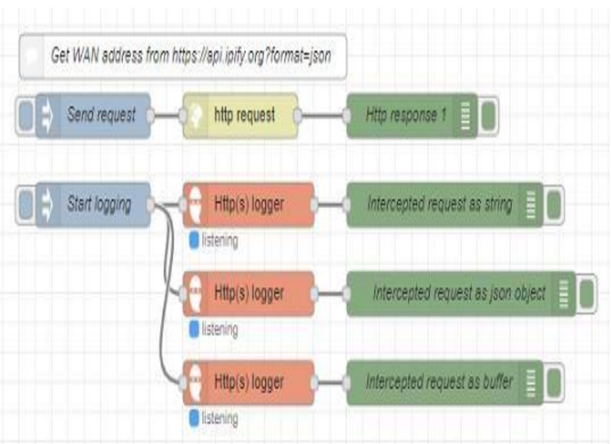

Figure 4: http response for different http GET requests

The process flow of the http GET request scheme is shown in figure 3 and the console results of the process are shown in figure 4.

2.1 APPLICATION OF NODE-RED ON RASPBERRY PI

Node-Red has this capability of holding the premises of scribing hardware, API and cloud services together. Hence, we wrote the hardware device raspberry pi to implement Quick-Start process on the hardware.

2.1.1. Implemented Quick-Start flow on Raspberry Pi using Node- red.

In this section, we have published the CPU temperature from my raspberry pi to the Watson IOT Platform Quick-Start service. I have used Node-RED as the runtime to create a flow to get the CPU temperature using the vcgencmd and publish the temperature in the appropriate format to the Quick-Start service [4]. The implementation model can be seen below.

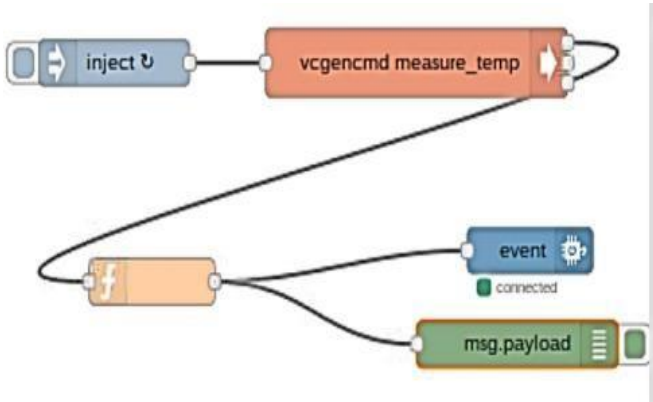

Figure 5: Node-RED flow to generate Quick-Start flow on Raspberry pi.

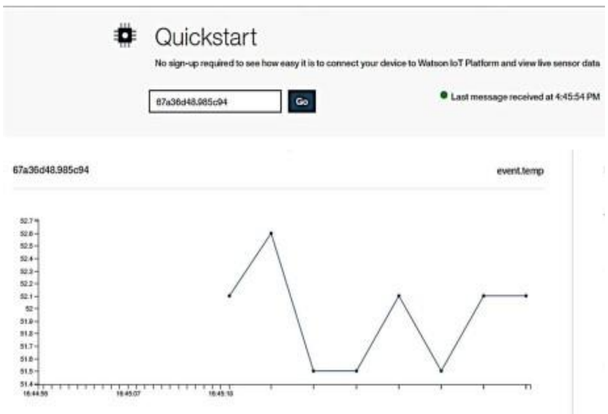

Figure 6: Quick-Start flow showing Raspberry pi processor temperature Vs time graph

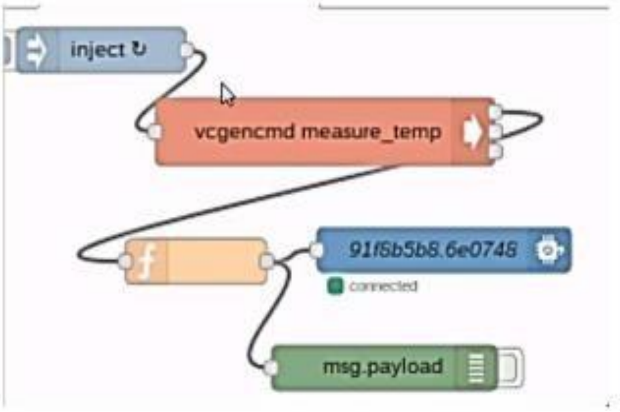

Fig.7. Node-Red flow

22 IMPLEMENTED END TO END SCENARIO

In this section, we have created a Node-RED application on both the Raspberry Pi and on Bluemix (Bluemix is the IBM open cloud stage that gives versatile and web engineers admittance to IBM programming for mix security.) to create an end-to-end scenario [5]. We have sent temperature data from the Raspberry Pi and Sense HAT (The Sense HAT is an extra board for Raspberry Pi, made particularly for the Astro-Pi mission – it dispatched to the International Space Station in December 2015 – and is currently accessible to purchase) to a cloud-based application where it is analyzed [6]. The cloud application will send a command back to the device if the temperature rises above, or falls below a preset limit. The Raspberry Pi will display the current temperature status, based on the latest cloud application command and also the current temperature from the Sense HAT temperature sensor [7]. This process is implemented in different parts and each part is explained below with related figures.

Part 1 - Register your Raspberry Pi as a gateway.

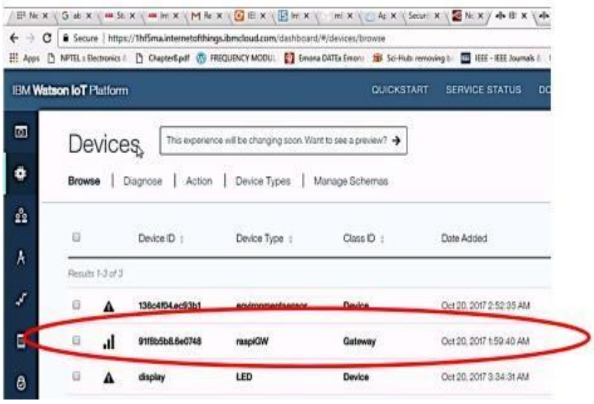

Figure 8: Showing Raspberry pi registered as Gateway in IBM Wastson IoT platform

- Part 2 - Publish temperature data.

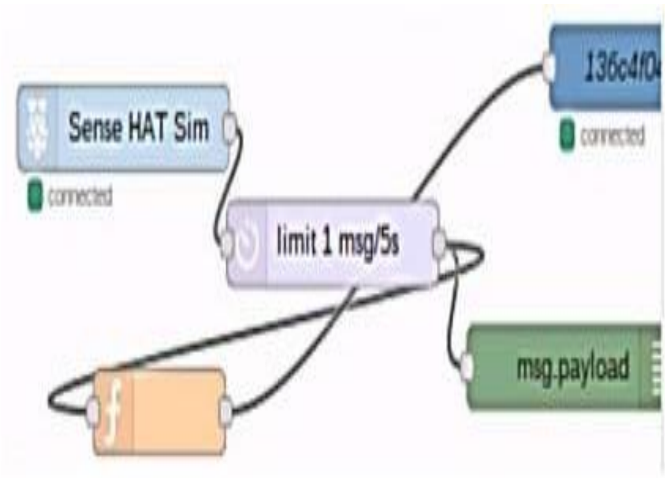

Figure 9: Node RED flow to send Sense HAT simulator temperature data from the raspberry pi to a cloud-based application.

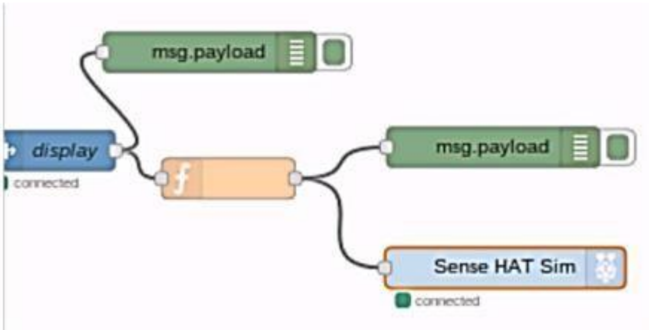

Figure 10: Node RED flow to display current temperature status on LED display.

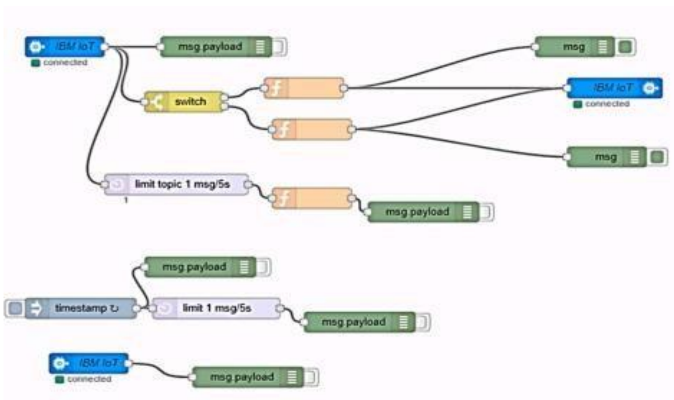

Figure 11: Showing Node RED flows in IBM cloud to receive CPU temperature, Sense HAT simulator temperature and to send display commands to LED display attached to Raspberry

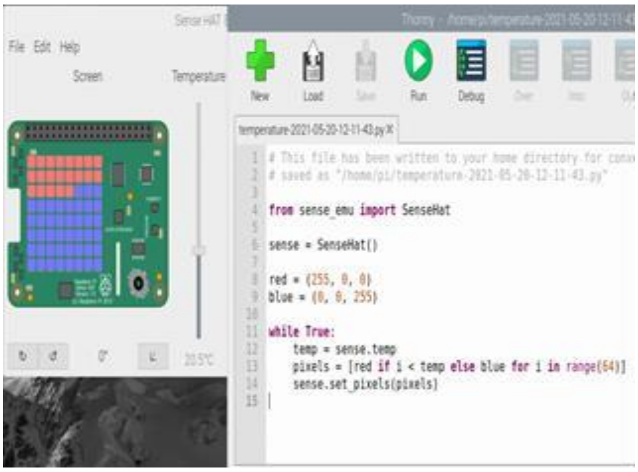

Figure 12: Showing received Sense HAT simulator temperature and humidity data in debug column.

- Part 4 - Update the Sense HAT LED panel.

The following Figure 13 shows the expected results on the LED panel. The design and implementation is being adopted in the form of LED panel that shows results accordingly [8].

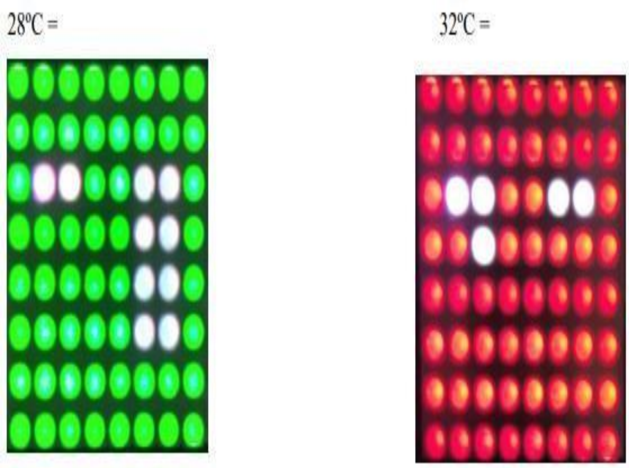

Figure 13: LED panel displaying the temperature data as per display commands received from cloud application.

3. CONCLUSION

In this work we have attempted to make IOT not only a communicating system but also an analytical entity. Also tried to show how IOT has a capability of taking rational and feasible decisions based on the analysis of the transferred data to IOT based systems. With growth in high end computing the need for independent decision making of the IOT systems has also increased and hence comes the need for intellectual enhancement of the IOT based systems. This work is solely dedicated to the intellectual enhancement and decision-making ability of the IOT based systems.

REFERENCES

[1] D. He, and S. Zeadally, “An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography,” IEEE Internet Things J., vol. 2, no. 1, pp. 72-83, 2015.

[2] P. Kolios, C. Panayiotou, G. Ellinas, and M.Polycarpou, “Data-Driven event triggering for IoT applications,” IEEE Internet Things J., 2016.

[3] U. Satija, B. .Ramkumar, and M. Sabarimalai Manikandan, “Real-Time Signal Quality-Aware ECG Telemetry System for IoT-Based Health Care Monitoring”, IEEE Internet of Things Journal, 2017. K. Elissa, “Title of paper if known,” unpublished.

[4] L.Catarinucci, et al., “An IoT-aware architecture for smart healthcare systems,” IEEE Internet Things J., vol. 2, no. 6, pp. 515-526, 2015.

[5] X. Li, Q. Wang, X. Lan, X. Chen, N. Zhang and D. Chen, "Enhancing Cloud-Based IoT Security Through Trustworthy Cloud Service: An Integration of Security and Reputation Approach," in IEEE Access, vol. 7, pp. 9368-9383, 2019, doi: 10.1109/ACCESS.2018.2890432.

[6] M. Shen, B. Ma, L. Zhu, X. Du and K. Xu, "Secure Phrase Search for Intelligent Processing of Encrypted Data in Cloud-Based IoT," in IEEE Internet of Things Journal, vol. 6, no. 2, pp. 1998-2008, April 2019, doi: 10.1109/JIOT.2018.2871607.

[7] S. Misra, M. Satyadeep and K. Manas, “Social sensing-based duty cycle management for monitoring rare events in wireless sensor networks”, IET Wireless Sensor Systems,2014, doi: 10.1049/iet-wss.2013.0125.

[8] S. Bera, S. Misra, S.K. Roy, and Mohammad S. Obaidat, “SoftWSN: Software-Defined WSN Management System for IoT Applications”, IEEE Systems Journal, 2016, Volume: PP, issue:-99, Pages: 1.

[9] S. Singh, I. Chana and R. Buyya, "STAR: SLA-aware Autonomic Management of Cloud Resources," in IEEE Transactions on Cloud Computing, vol. 8, no. 4, pp. 1040-1053, 1 Oct.-Dec. 2020, doi: 10.1109/TCC.2017.26487

[10] S. Girs, S. Sentilles, S. A. Asadollah, M. Ashjaei and S. Mubeen, "A Systematic Literature Study on Definition and Modeling of Service-Level Agreements for Cloud Services in IoT," in IEEE Access, vol. 8, pp. 134498-134513, 2020, doi: 10.1109/ACCESS.2020.3011483

AUTHORS PROFILE

Mr. Syed Owais received his bachelor's degree in Electronics & Communication Engineering from Jodhpur National University, Rajasthan. After completing B. Tech, he served as system engineer in Matrix Comsec Pvt. Ltd., Gujarat. He is a research student in Vivekananda Global University, Jaipur.

Mr. Sandeep Kumar Jain received his bachelor's degree from University of Rajasthan, Jaipur in 2007. He is working as associate professor in Vivekananda Global University, Jaipur. His areas of research are SPR based fiber optics biosensors and VLSI design.

Mr. Neeraj Sharma received his bachelor's degree from University of Rajasthan, Jaipur. He has done his Master's in Digital Communication from IET-DAVV, Indore. He is currently working as Assistant Professor in Vivekananda Global University, Jaipur.