International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958 (Online), Volume-9 Issue-4, April 2020

IoT based Smart Automation System with Analysis for Power Optimization

Buvaneswari S, Abraar Mohammed Malik S, Albrin R, Abubacker Siddique N

Abstract: ***The increasing number of electrical devices and appliances has led to several problems such as over usage of power and wastage of power. These problems can be solved if the devices and appliances are monitored by analyzing their usage and controlling them on a regular basis. This project solves the emerging power crises by collecting data from appliances through IoT enabled microcontrollers. A cloud server is used to store collected data for analysis. The microcontroller is connected to the cloud server via Internet. A dashboard is presented to the user where the state of all the appliances are visually represented using graphs and charts. This visualization can be used to analyze the power usage of every appliance which informs the user of any wastage of power and other information. The dashboard also comes with controls that enables the user to control the devices remotely with the click of a button.***

Keywords: Energy, Internet of Things, Power, Smart Home

I. INTRODUCTION

In this age of internet, plethora of devices are interconnected and are communicating with each other. IoT devices enable machine to machine communication over the internet. Home Automation is the part of the larger Internet of Things (IoT) community. It enables various devices and appliances to be monitored and controlled over the internet. In Internet of Things (IoT) each machine is given a unique identifier (UID) to be identified uniquely over the network. IoT devices can be used to monitor and control mechanical, electrical and electronic systems used in buildings. Long-term benefits could include energy savings by automatically ensuring lights and electronics are turned off. A smart home or automated home could be based on a platform or a hub that controls smart devices and appliances. The integration of the Internet with building energy management systems helps to create energy efficient and IOT-driven "smart buildings". These devices allow for remote control by users, or central management via a cloud- based interface, and enable functions like scheduling. Internet-connected devices, electric utilities not only collect data from end-users, but also manage distribution automation devices like transformers.

Revised Manuscript Received on April 27, 2020.
* Correspondence Author
Buvaneswari S*, Assistant Professor, Easwari Engineering college, Chennai, Tamil Nadu.
Abraar Mohammed Malik S, Student, Computer Science and Engineering, Easwari Engineering.
Albrin R, Student, Computer Science and Engineering, Easwari Engineering .
Abubacker Siddique N, Student, Computer Science and Engineering, Easwari Engineering .

IoT devices can not only be used to monitor and control the devices, but also be used to collect the data from these devices for power analysis and optimization.

II. LITERATURE SURVEY

A. Iot Based Smart Home Using Adafruit And IFTTT

This is a possible solution for cost effective fabrication of smart home device which is like a box that could be connected anywhere. This System consist of an IoT enabled microcontroller which is connected to various sensors to collect and send data to the Adafruit server which reflects the current state of each appliance and can be controlled using IFTTT. Though, this IoT system is portable and can be controlled over the Internet ,the server which holds the current state (Adafruit) and the control Interface (IFTTT) are two separate entities due to which many incompatibility issues may occur and the IoT Hardware operates on Battery which is not always reliable and the power source to the microcontroller must be an more reliable source like direct AC power [1].

B. Smart Energy Efficient Home Automation System Using IoT

This is a solution for controlling the home appliances from anywhere in the world. This system consists of Internet connectivity module attached to the main supply unit which could be accessed through the internet. This system is very useful for controlling the appliance from elsewhere. Though, this IoT system can be controlled over the internet, it requires a static IP to be accessed, which may not be feasible in household Internet environment. Additionally, implementing static IP becomes exponentially expensive as the number of devices increase. IoT devices that work in a dynamic IP setup is preferred [2].

C. Nodemcu An IoT Enabled Microcontroller Which Is Controlled Using An Android App

This is a solution for automating homes using IoT enabled NodeMcu which could be controlled using an Android App. This System consist of a Microcontroller to which various light switches, temperature and humidity sensors, gas sensors, motion detection sensors and alarms are connected and controlled using the Android app. This system also works even if the home Internet connection is down using 3G/4G connectivity. Though, this IoT System works under adverse condition , it lacks in the department of power optimization and Scalability of device addition [3].

D. Reconfigurable Real-Time Middleware And Wukong

A Merkle hash tree, is a hash tree that is used in cryptography. The leaf nodes of the tree consist of the hash value of the data blocks. All the non-leaf nodes consist of the hash value of the hash contained in their child nodes. These hash trees are used for verifying the data, that is stored in data structures and also to verify data that are transferred between multiple computers. The hash tree is very useful in a peer-to-peer network, where it is used to verify the integrity of the transmitted block and also to ensure that the sending peer is not transmitting a fake block of data [4].

E. Decarbonization Of Electric Power And Energy System

This presents an important role of IoT in transforming Electric Power and Energy Systems (EPES). IoT has a significant impact on EPESs and offers several opportunities for growth and development. The authors discuss various areas of interest such as the role of IoT in Intelligent Electric Power Networks, the impact of IoT on Electric Power and Energy Systems, and the constraints for the deployment of IoT in Electric Power and Energy Systems. Deploying IoT in EPESs has several advantages, but some challenges exist such as: sensing, connectivity, power management, big data, computation, complexity, and security. The authors provide recommended solutions to overcome these challenges. Digitizing the electric power ecosystem using IoT helps to better account for DER integration, reduce energy wastage, generate savings, and improve the efficiency of the electric power networks [5].

F. Lora Mesh Network

The development of smart buildings is impeded by the high installation/maintenance cost and the difficulty of large-scale evaluation. The authors propose the use of LoRa mesh network to cover the whole building using one gateway only. This system is built on top of existing facilities in the building, so that the deployment and maintenance cost is reduced. Evaluation results show that the systems are stable and robust to environmental changes [6].

G. Iot, Brute Force Search Algorithm

This algorithm provides the design and implementation of a smart home control system where load shedding and reduced energy consumption is encouraged. The authors propose the use of a central controller that incorporates Brute Force search algorithm and gives On/Off commands to the smart plugs that are connected to the electrical appliances, based on the Time of Use (TOU) tariffs and the load shedding commands produced by the power utility. The proposed system aims at reducing the consumption and cost by the implementation of Conservation Voltage Reduction (CVR). This implementation in the system leads to reduced cost and voltage which makes the system efficient [7].

III. SYSTEM ARCHITECTURE

The end user interacts with the dashboard using an input device like mouse or touch. The click or touch interaction triggers a HTTP Request to the server API which contains the user queries in the request

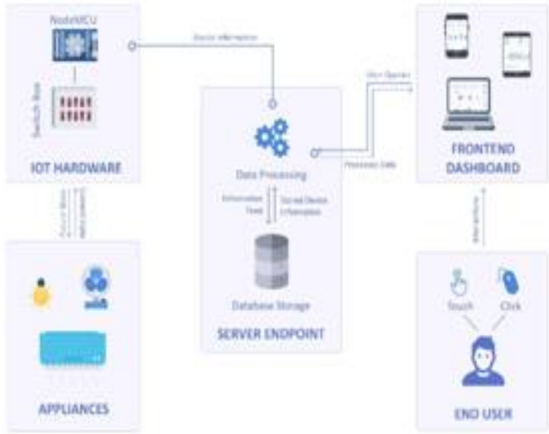

Fig. 1: Overall Structural Architecture

body. The request is sent in JSON format. The server receives the payload from the request sent by the dashboard and determines the type of function to be performed. The data from the dashboard is logged into the database. The microcontroller requests the state of the appliances at regular intervals and updates the current state of the devices using the received data, if there is a change. The server analyses the data stored in the database and sends the processed data to the frontend for visualization in the dashboard.

IV. FUNCTIONAL ARCHITECTURE

The proposed system consists of three main phases. They are Frontend, Cloud API and Dashboard display. In the first phase the user is required to click button which triggers Ajax to sends a CRUD command with the device id and the state that are used for creation of a device or to modify the state of the existing device state. The Front end also Request the device logging information from the cloud API which is then parsed and displayed in dashboard for power optimization.

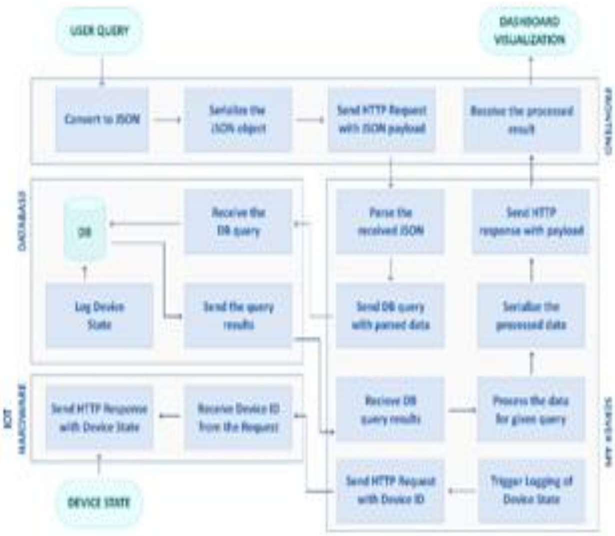

Fig. 2: Functional Architecture: Analysis and Visualization

Retrieval Number: D7562049420/2020©BEIESP
DOI: 10.35940/ijeat.D7562.049420
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958 (Online), Volume-9 Issue-4, April 2020

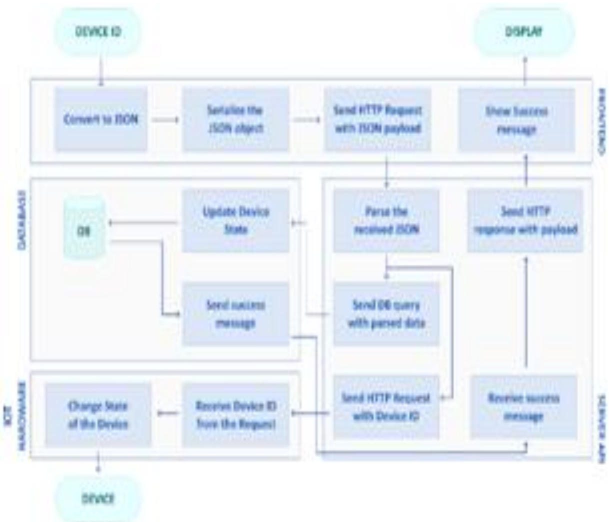

Fig. 3: Functional Architecture: Device Control

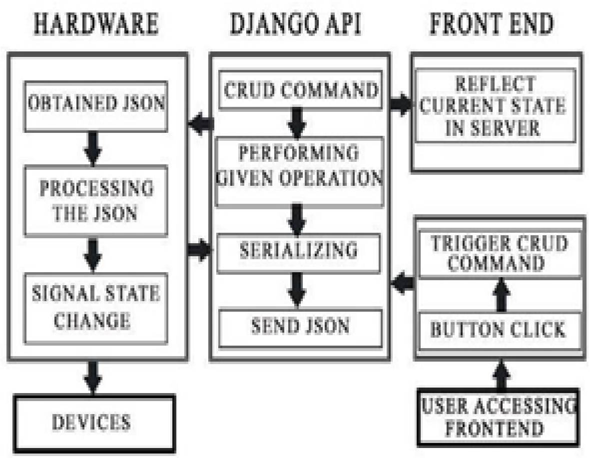

Fig. 4: Detailed Functional Architecture

In the second phase the cloud API is developed using Django REST API and hosted in Amazon Web Services. The cloud API receives the CRUD command from Frontend or Hardware and analysis the type of command and trigger the function associated with each type of command. The obtained data is serialized and will be sent as JSON.

In the third phase the hardware connects with the internet using SSID and password. After connecting to the Internet, the microcontroller constantly hit the server in a loop to obtain the device_id and state of devices. Then the Hardware triggers the serial Pin to turn relay ON/OFF which in turn turns the device ON/OFF.

V. IMPLEMENTATION OF THE PROPOSED SYSTEM

A. Dashboard For Power Analysis And Control

The dashboard is a frontend web application developed using HTML, CSS, and JavaScript. It is hosted on a web server online and can be accessed from anywhere in the world using a URL. It's developed using a JavaScript library, called React, that facilitates reactive programming and state management.

Fig. 5: Frontend Dashboard Homepage

The application works based on RESTful API commands sent to a server as HTTP requests and receives the response as JSON from the server. The JSON converted toJavaScript objects and saved in the application's state. The React library then detects that the state is changed and renders the changes to the Document Object Model (DOM). Whenever the state changes, the DOM is re-rendered, effectively managing the rendering process from a single source of truth – the state of the application. The graphs and charts are developed using a library called D3. The library takes the data provided as objects and convert into visual graphs using Scalable Vector Graphics.

B. Django REST API

The cloud API is developed using Django REST API and hosted in Amazon Web Services. The cloud API receives the CRUD command from Frontend or Hardware and analysis the type of command such as GET, PUT, POST, DELETE and trigger the function associated with each type of CRUD command. The GET command is used to get the device details such as device_name, device_id, device_type, state and serialize the obtained data and return the data as JSON for the request.

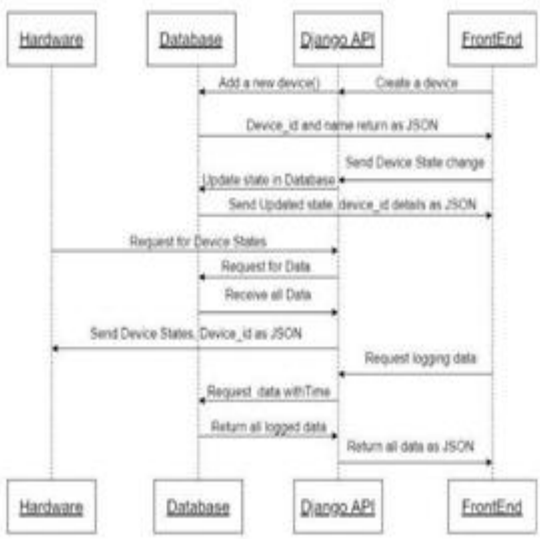

Fig. 8: Sequence Diagram for Django Cloud API

Retrieval Number: D7562049420/2020©BEIESP
DOI: 10.35940/ijeat.D7562.049420
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

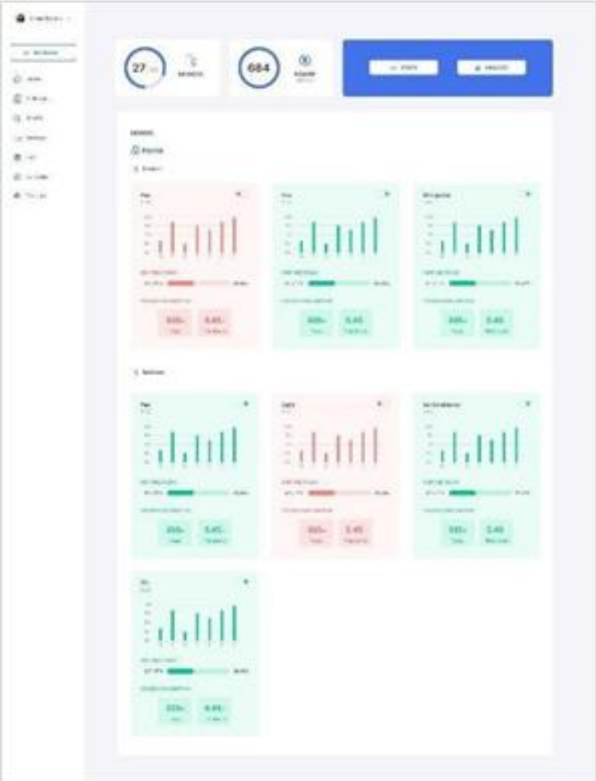

Fig. 6: Device List with weekly Analysis

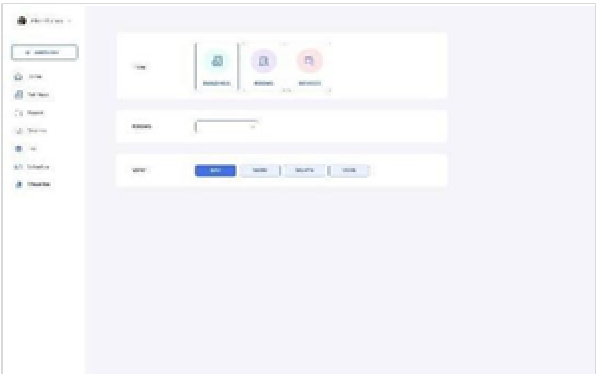

Fig. 7: Device Visualization for Analysis

The POST command is used to create device details entry with data such as device_name, device_id, device_type, state and serialize the obtained data and update the database table, return the acknowledgement for creating a table entry data as JSON for the request. The PUT command is used to modify the device details such as device_name, device_id, device_type, state and serialize the updated data and return the data as JSON for the request. The DELETE command is used to delete the device details such as device_name, device_id, device_type, state and return the data as blank for the request.

```
class devicesListDetailedView(APIView):
def get_object(self, id):
try:
return Devices.objects.get(id=id) except
Devices.DoesNotExist as e:
return Response({"error": "Given question object not
found."}, status=404)
def get(self, request, id=None): instance =
self.get_object(id) serializer = devicesSerializer(instance)
return Response(serializer.data)
```

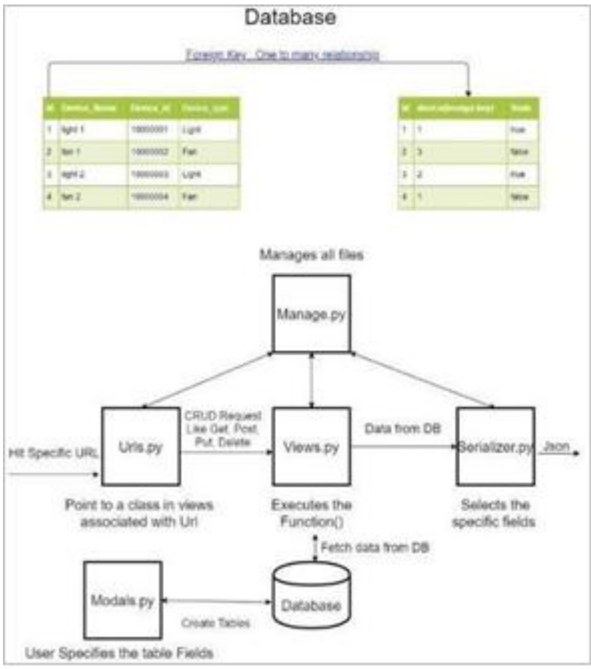

Fig. 9: Detailed Working of Django REST API

V.B.1. Crud Command Functions

It receives the HTTP request and provide the data as JSON. In the below code, based on the type of request get(), put(), post(), delete() functions are executed and the response data is sent as JSON.

```
class devicesList(APIView):
def get(self, request):
devices1 = Devices.objects.all()
        serializer = devicesSerializer(devices1, many=True)
        return Response(serializer.data)
    def post(self, request):
        data = request.data
        serializer = devicesSerializer(data=data)
        if serializer.is_valid():
          serializer.save()
          return Response(serializer.data, status=201)
        return Response(serializer.errors, status=400)
    def put(self, request, id=None):
        data = request.data
        instance = self.get_object(id)
        serializer = devicesSerializer(instance,
        data=data) if serializer.is_valid():
          serializer.save()
          return Response(serializer.data, status=200)
        return Response(serializer.errors, status=400)
    def delete(self, request, id=None):
        instance = self.get_object(id)
        instance.delete()
        return HttpResponse(status=204)
```

V.B.2. Response From The Server

The Django REST API Server sends the response to the client with the payload in JSON format. Each building, room, and device is represented using a unique id and the data is sent hierarchically for easier manipulation.

Retrieval Number: D7562049420/2020©BEIESP
DOI: 10.35940/ijeat.D7562.049420
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

the device_id and state of devices. If the current state of device is different from the obtained state from server then the Hardware triggers the serial Pin to turn relay ON/OFF which in turn turns the device ON/OFF.

```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[
    {
        "id": 2,
        "rooms": [
            {
                "id": 1,
                "devices": [
                    {
                        "id": 1,
                        "device_type": "light",
                        "device_name": "Light1",
                        "device_id": 11000001,
                        "room": 1
                    },
                    {
                        "id": 2,
                        "device_type": "socket",
                        "device_name": "Socket",
                        "device_id": 11000002,
                        "room": 1
                    }
                ],
                "room_name": "Room1",
                "room_id": 11000000,
                "building": 2
            },
            {
```

Fig. 10: HTTP Response in JSON format from the server

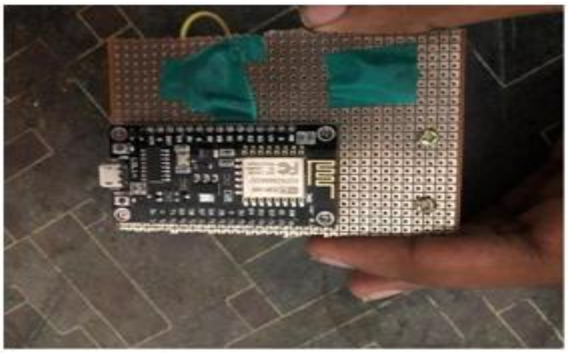

Fig. 11: Hardware Module: NodeMcu Microcontroller

C. IoT Hardware Fabrication

The hardware connects with the internet using SSID and password. After connecting to the Internet, the microcontroller constantly hit the server in a loop to obtain

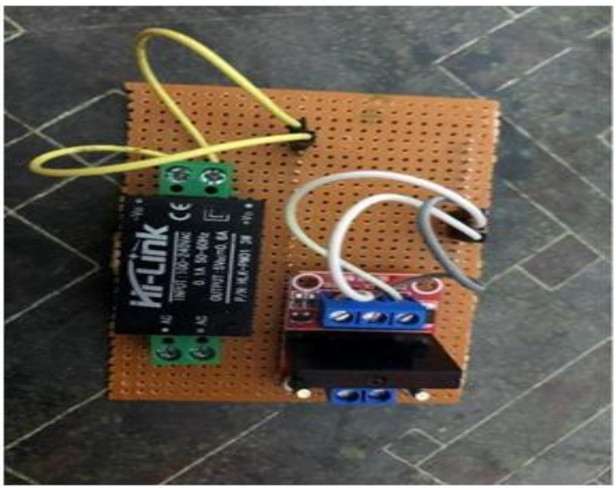

Fig. 12: Hardware Module: Transformer and Relay

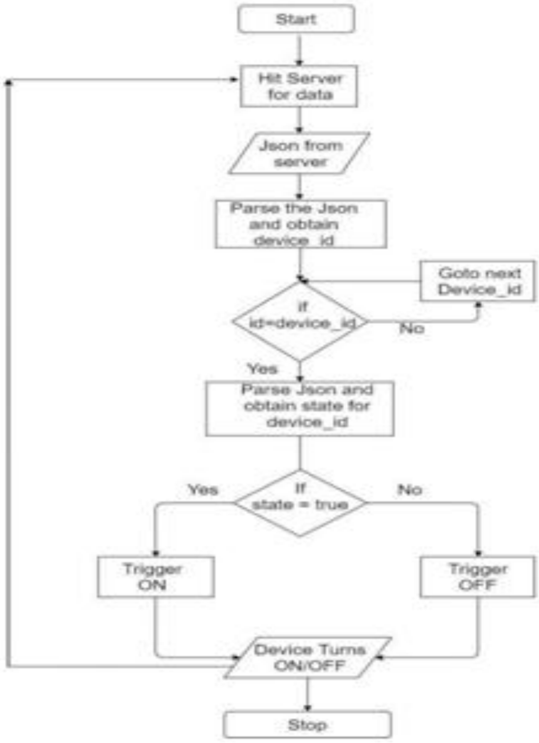

Fig. 13: Flowchart of the Hardware Module

VI. CONCLUSION AND FUTURE WORK

The main potential of IoT is not being used to its fullest and the focus has been on controlling the appliances using the internet and various other alternatives such as 4G, LoRa mesh, etc. Many systems have been developed for controlling the appliances using IoT but most of them lack in the department of power optimization. Thus, the data collected from each IoT devices can be visualized using Graphs and Bar charts for future power optimization. In the future, the IoT devices can also be voice automated for easy control of devices remotely and various technologies such as 4G, LoRa mesh can be used as backup.

REFERENCES

1. W. A. Jabbar et al., "Design and Fabrication of Smart Home With Internet of Things Enabled Automation System," in IEEE Access, vol. 7, pp. 144059-144074, 2019.
2. S. K. Vishwakarma, P. Upadhyaya, B. Kumari and A. K. Mishra, "Smart Energy Efficient Home Automation System Using IoT," 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Ghaziabad, India, 2019, pp. 1-4.
3. L. Liao et al., "Design and Validation of a Multifunctional Android-Based Smart Home Control and Monitoring System," in IEEE Access, vol. 7, pp. 163313-163322, 2019.
4. C.Shih, J. Chou, N. Reijers and T. Kuo, "Designing CPS/IoT applications for smart buildings and cities," in IET Cyber- Physical Systems: Theory & Applications, vol. 1, no. 1, pp. 3-12, 12 2016.
5. Bedi, G., Venayagamoorthy, G. K., Singh, R., Brooks, R. R., & Wang, K.-C. (2018). Review of Internet of Things (IoT) in Electric Power and Energy Systems. IEEE Internet of Things Journal, 5(2), 847ñ870. doi:10.1109/jiot.2018.2802704
6. Xu, W., Zhang, J., Kim, J. Y., Huang, W., Kanhere, S., Jha, S., & Hu, W. (2019). The Design, Implementation, and Deployment of a Smart Lighting System for Smart Buildings. IEEE,Internet of Things Journal, 1ñ1.

7. Parsa, T. A. Najafabadi and F. R. Salmasi, "A Hierarchical Smart Home Control System for Improving Load Shedding and Energy Consumption: Design and Implementation," in IEEE Sensors Journal, vol. 19, no. 9, pp. 3383- 3390, 1 May1, 2019.

AUTHORS PROFILE

Buvaneswari S, is working as Assistant Professor in Easwari Engineering college, Chennai, Tamil nadu having 15 + years of experience in academics and research. She Completed M.E. Degree from Anna University and interested in innovative projects and keen on research work in the area of wireless sensor networks, IoT, algorithms etc.

Abraar Mohammed Malik S, Final Year Student, Computer Science and Engineering, Easwari Engineering College. Currently pursuing under graduation in B.E. CSE. He has completed an internship in App Development. He has attended workshops on IoT and completed courses in Web Development, Java, Data Structures and Algorithms, Machine Learning.

Albrin R, Final Year Student, Computer Science and Engineering, Easwari Engineering College. Currently pursuing under graduation in, B.E CSE. He has published a paper in a renowned international journal, IJSTR. He has completed an internship in Information and Cyber Security at Live Wire. He has attended workshops on IoT, Arduino, Microsoft Azure Cloud Computing and IET Intelligent Transportation System.

Abubacker Siddique N, Final Year Student, Computer Science and Engineering, Easwari Engineering College. Currently pursuing under graduation in B.E. CSE. He has done courses on Web Development, using Java, and JavaScript. He has attended workshops on Cyber Security, and Web Development. He has worked on several projects related to Application Development and Web Development.

Retrieval Number: D7562049420/2020©BEIESP
DOI: 10.35940/ijeat.D7562.049420
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication