Design and Implementation of Automation Appliances Control Based on MVC Model Using Distributed MQTT Broker in CoT Networks

[1]Faisal Mehmood, Shabir Ahmad, DoHyeun Kim

Abstract: Background/Objectives: Nowadays, researchers and engineers aim to provide smart home services by integrating Internet of Things with cloud services. This paper aims to provide design and implementation of smart home automation system. With this system, users can completely control their home appliances from remote locations. Methods/Statistical analysis: The web application is based on MVC (Model View Controller) model and deployed on AWS (Amazon Web Service) cloud. MQTT (Message Queuing Telemetry Transport) protocol is used for IoT device communication. MQTT Broker is configured on AWS cloud for publishing and subscribing messages from device to device. Due to load on MQTT Broker we have proposed distributed broker in this system to support high number of publishers and subscribers. This system is comprised of one main centralized MQTT Broker and many distributed MQTT Brokers. We have used MQTT connectivity protocol for M2M (Machine to Machine) communication. IoT device is connected with ESP8266, a Wi-Fi based development board. Findings: The design and implementation of Automation Appliances Control service based on MVC model using distributed MQTT broker in CoT networks has been carried out. There is one centralized MQTT broker that is responsible for receiving messages from client application. Distributed MQTT broker are responsible for subscribing and publishing messages from embedded devices to MQTT broker and vice versa. Each smart home consists of many home appliances so there is a load on the MQTT broker. Distributed MQTT broker is useful to control large number of smart homes. Each IoT device is subscribed to a topic and perform action accordingly. After testing the proposed application, we come to conclusion that distributed MQTT broker has reduced the load of publish/subscribe messages and increased the performance of monitoring and controlling smart devices. Improvements/Applications: The concept of Smart Home is becoming a trend with more appliances and sensors getting connected into a single house infrastructure. The aim is to make our living more convenient, secure and carefree by embracing hardware and software advancements. In future we can make improvements in voice control, intelligent control, burglary prevention, and remote health monitoring.

Keywords: Internet of Things (IoT), Cloud of Things (CoT), Smart Homes, Message Queuing Telemetry Transport (MQTT), Amazon Web Services (EC2), ESP8266 Arduino

I. INTRODUCTION

Nowadays, Internet of things (IoT) is on the rise in the IT industry. The researchers in this area are trying to build IoT systems so that it can meet the desired needs of the IT industry. These systems will serve in different fields of the IT industry e.g. telecommunication, medical, banking sector etc. IoT is a network of different devices that are connected with each other via internet connection to exchange useful data [1]. Normally, these devices get useful data from other devices using state-of-the-art communication protocols and communicate with each other. A lot of research has been carried out in the field of internet of things and smart homes. Smart home is a system that can control home appliances remotely using IoT technology. Smart homes can monitor and control IoT devices remotely via internet connection.

Cloud computing services are also gaining the attention of IT industry. There are many companies that provide cloud computing services. The most popular are Microsoft, Google, and Amazon etc. In our proposed system we have used Amazon Web Services. Amazon. In our proposed system we have used Amazon Elastic Compute Cloud (Amazon EC2) web service. Amazon EC2 provides easy computation for developers on the cloud and it secure and reliable [2]. We have developed web-based application named as ‘Home Appliance Control System’ and deployed on Amazon cloud. We have configured Node.js on the cloud and it is running as cloud-based server. The web application is user friendly and end users can easily create virtual design of home and add rooms on the web. They can add devices in each room. End users will be able to monitor and control those devices remotely.

There are many communication protocols for data transportation [3]. The most popular ones are CoAP (Constrained Application Protocol), and MQTT (Message Queuing Telemetry Protocol) etc. In our proposed system, we have used MQTT protocol for IoT connectivity because of its lightweight feature. There are many types of MQTT like hive-mqtt, paho-mqtt, and mosquito etc. In our system we have used MQTT-Mosquitto. It is a protocol used in IoT applications and machine-to-machine (M2M). It is designed to be lightweight publish and subscribe message protocol. It is used in mobile applications because it uses less data for publishing and subscribing messages. There are two types of agents in MQTT connectivity; the client and the broker. Client is responsible for sending and receiving messages from device to broker and vice versa. The broker is responsible for connecting different IoT devices. In this research paper, we have proposed the concept and implementation of distributed MQTT broker to overcome the load. The broker is configured on cloud. The default port for MQTT

Revised Manuscript Received on January 03, 2019.

Faisal Mehmood, Computer Engineering Department, Jeju National University, Republic of Korea,

Shabir Ahmad, Computer Engineering Department, Jeju National University, Republic of Korea,

DoHyeun Kim, Corresponding author, Professor, Computer Engineering Department, Jeju National University, Republic of Korea,

*Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication*

protocol is 1883.

In our proposed system, we have used WeMos D1 ESP8266 Wi-Fi development board. ESP8266 is a low-cost development board used for rapid prototyping [4]. IoT devices are connected with the ESP8266 board. These devices can be monitored and controlled via web application using MQTT protocol. We have used MySQL database to store the data of IoT devices. This data is used to monitor and control the IoT devices. The IoT devices can be turned ON or OFF using our Home Appliances Control System. Users can visualize the amount of power consumed by IoT devices on daily or monthly basis which will help to reduce power consumption and overcome electricity bills. Users can schedule IoT devices to turn ON or OFF. Users can schedule a date for IoT device to get serviced. Users will receive notification alerts on the scheduled date which will be helpful in the maintenance of IoT devices.

The rest of this paper is organized as follows. In Section 2, related work is described briefly. Existing research work in relevant field is highlighted. Different methods and techniques are explained in detail. In section 3, each component of system is explained in detail. The architecture of the system and its database is explained briefly. In section 4, proposed architecture is explained. The working of the system is described briefly. In section 5, hardware and software services are explained. The development model of the system is described in detail. Section 6 is experimental environment. In this section we explain the configuration of the hardware devices. And we explain the results achieved in the testing phase. In last section 7, we briefly described the conclusion.

II. RELATED WORK

Internet of Things is an IP based connectivity between devices, systems, and services. IoT goes beyond traditional human-to-machine (H2M), and machine-to-machine (M2M) interactions.

Nowadays Internet of Things is attracting IT industries with advance development of smart home services. It has attracted attention of consumers and enterprise electronics. Engineers have developed low-cost embedded devices that can communicate with each other via protocols. Normally a lot of work is done on the implementation of server-side applications and sensor actuators work is done at the client side. This mode of communication requires robust networking infrastructure. To reduce the risk of failure, researchers has proposed an offline online strategy for IoT applications [5]. In this methodology MQTT protocol is used for asynchronous communication.

MQTT is a machine-to-machine connection-oriented protocol [6]. It is normally used to communicate two or more devices. It is specifically designed as lightweight protocol which works above the http protocol layer. It is used to publish and subscribe messages. MQTT protocol is used where small code is required or where the network bandwidth is expensive. It is used in remote locations e.g. sensors communicating to a broker through satellite. It is extremely useful for mobile applications because MQTT protocol is very light weight, it consumes less power as compared to other protocols, minimum data packets are transferred, and it distributes information to one or more receivers in efficient manner.

A variety of efforts have been made in integrating applications and software with large number of physical objects and devices in a huge IoT network. Some of the example of these applications are e-banking, electronic health record (EHR), intelligent transportation systems. These applications require real-time control, large scale database for data storage, data analytic mechanism, and powerful processing capabilities to operate in large environment [7,17,18]. The solution to this requirement is promising technology known as Cloud of Things (CoT). These applications can operate based on CoT networks. CoT can provide a platform to integrate IoT with cloud services [9].

Home Automation [10,11] refers to remotely monitoring the conditions of home and performing the required actuation. Through home automation we can control our electronic devices such as lamp, fan, television etc.

Smart home services are on the rise which mainly include monitoring and controlling home appliances via the internet [12]. Security is the major concern in Smart Homes. Researchers aim to provide secure smart home automation system. In this paper “MQTT Based Secured Home Automation System" [13], the author has designed and implemented secure smart home automation system by using Raspberry pi, sensors. MQTT protocol is used for transportation of sensor data. The system is comprised of ACL (access control list) to provide encryption. User can monitor device data on a webpage. Smart home services also include object detection based on deep learning algorithms using OpenCV library [14,15].

III. SYSTEM MODEL

The proposed system is based on MVC model. MVC is an architectural pattern commonly used for developing user interfaces that divides an application into three interconnected parts. This is done to separate internal representations of information from the way’s information is presented to and accepted from the user. The MVC design pattern decouples these major components allowing for efficient code reuse and parallel development.

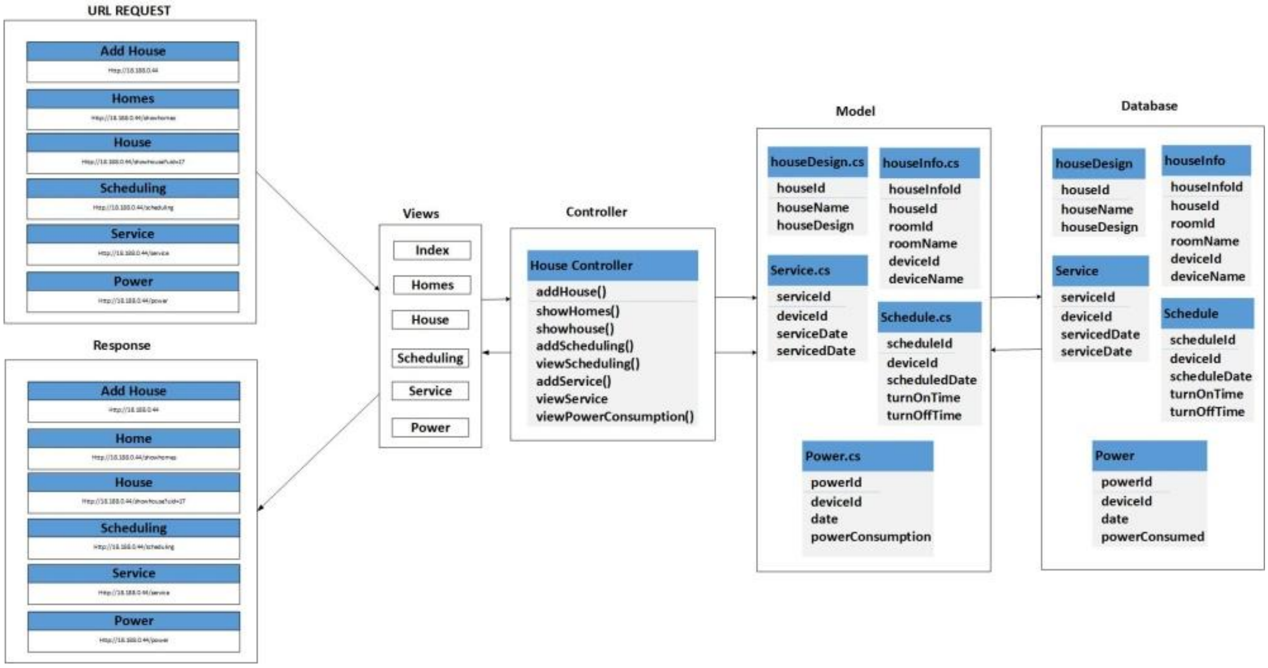

Figure 1. MVC Architecture of Home Appliances Control System

In Figure 1, MVC-based Home Appliances Control System is shown. The proposed system is very user friendly. We have separated the system into three different modules i.e. Views, controller and model. The design is influenced from [16] which also emphasized on the use of lightweight MVC frameworks. In the proposed system user can create virtual design of house and add rooms in the house to control IoT devices. The End users can schedule IoT devices to turn ON or turn OFF. Users can schedule a service date for IoT devices.

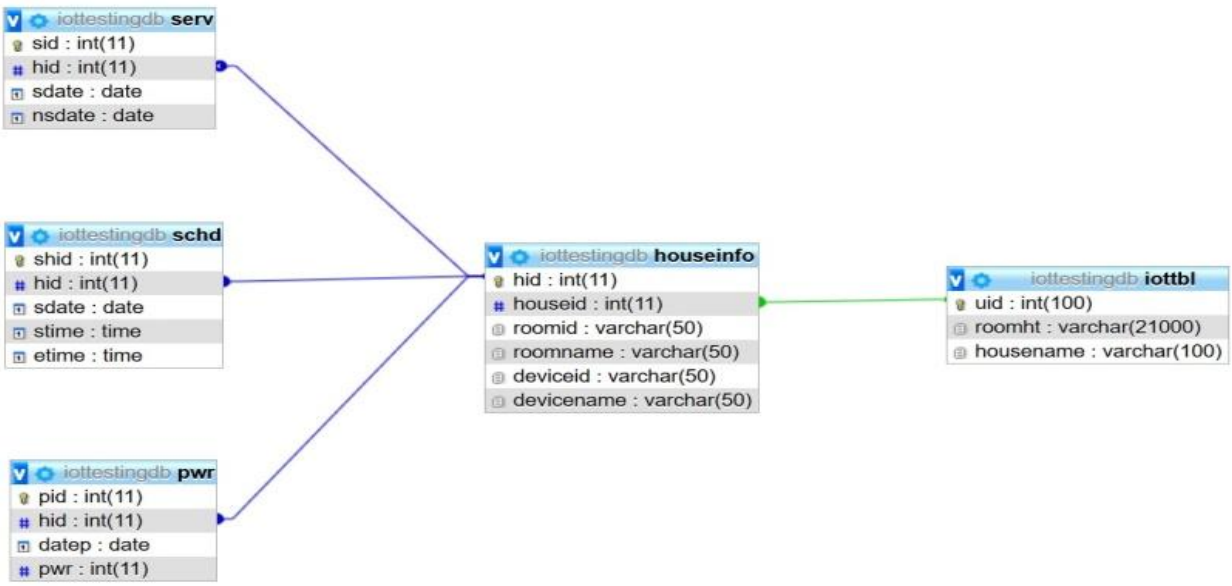

Figure 2. Database Schema of Home Appliances Control System

In figure 2, Entity relation diagram of the proposed system is portrayed. In the proposed system we have used open source database i.e. MySQL to keep the data of smart homes. There are 5 database tables in the proposed system. In database we have normalized data. The database tables are well structured. There is primary and foreign key relationship between different tables. The data types are selected according to the need of data fields.

IV. PROPOSED ARCHITECTURE

Figure 3 shows the design of the system. In this figure Application layer is the layer in which user interacts with the web server via PC, mobile. Our web server is hosted on AWS Cloud. We have created Ubuntu instance on AWS cloud. Our web application is MVC-based. On AWS Ubuntu instance, we have configured MQTT Broker. In figure you can see there is a centralized MQTT Broker and a distributed MQTT Broker due to load on the MQTT broker. User interacts with the web server via web sockets. The communication between MQTT Broker and client is via MQTT protocol.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

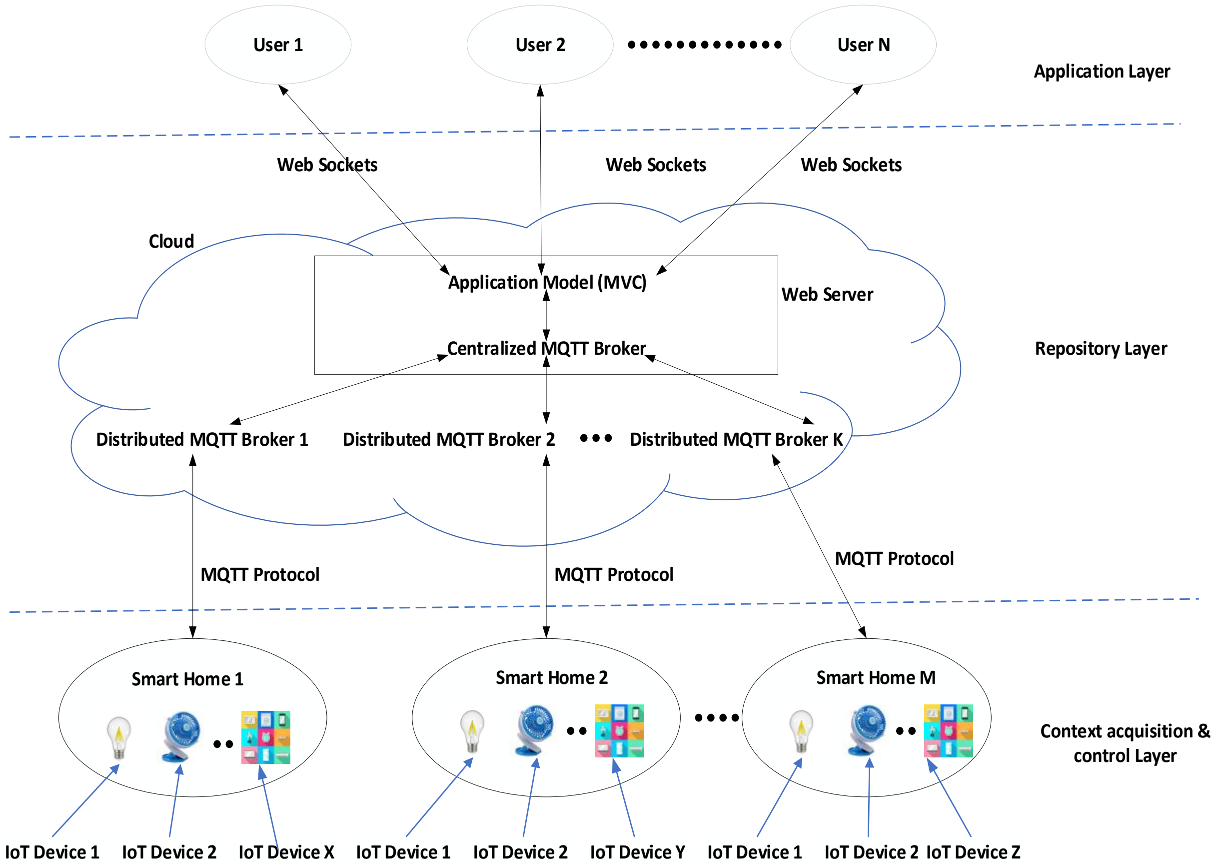

Figure 3. General Proposed Architecture

Figure 4 represents the sequence diagram of the Home Appliances Control System. There are three layers; Application layer, Repository layer, and Context acquisition & control layer. In the application layer user requests web application which is hosted on AWS cloud. After validation, user performs some action e.g. User wants to Turn ON the light. The message is then sent to Web Server which is hosted on AWS Cloud via web sockets. On AWS Cloud we configured MQTT Broker. The message is received by MQTT broker. The MQTT Broker then publishes the message to client. The client then successfully subscribes the message and the light is switched ON.

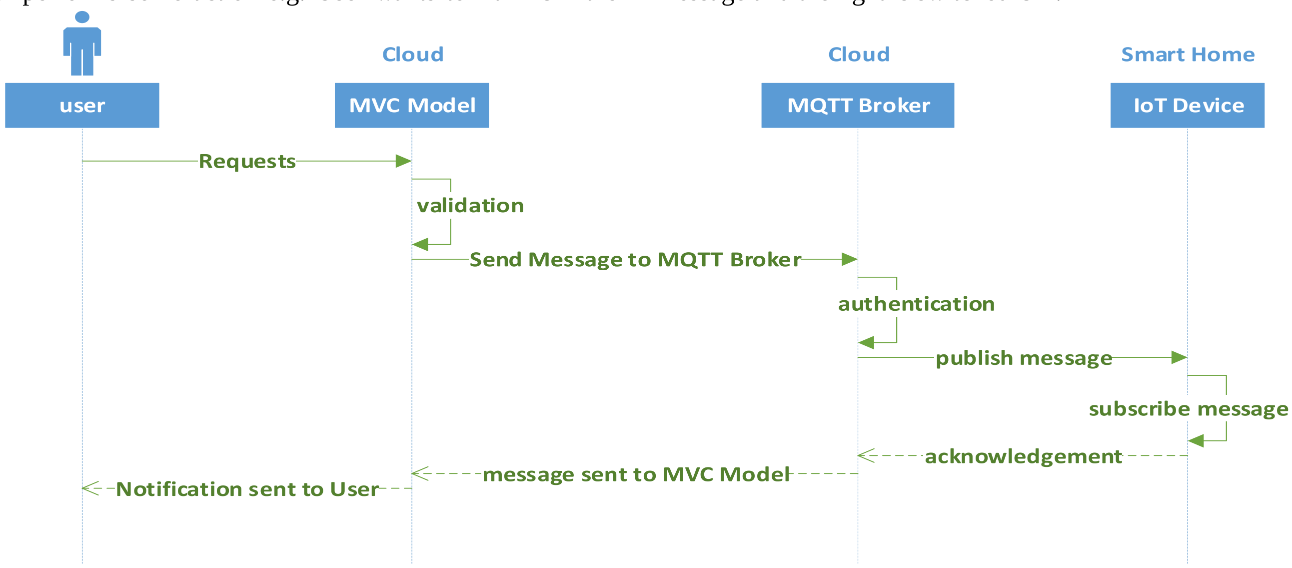

Figure 4. Sequence Diagram of Home Appliances Control System

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

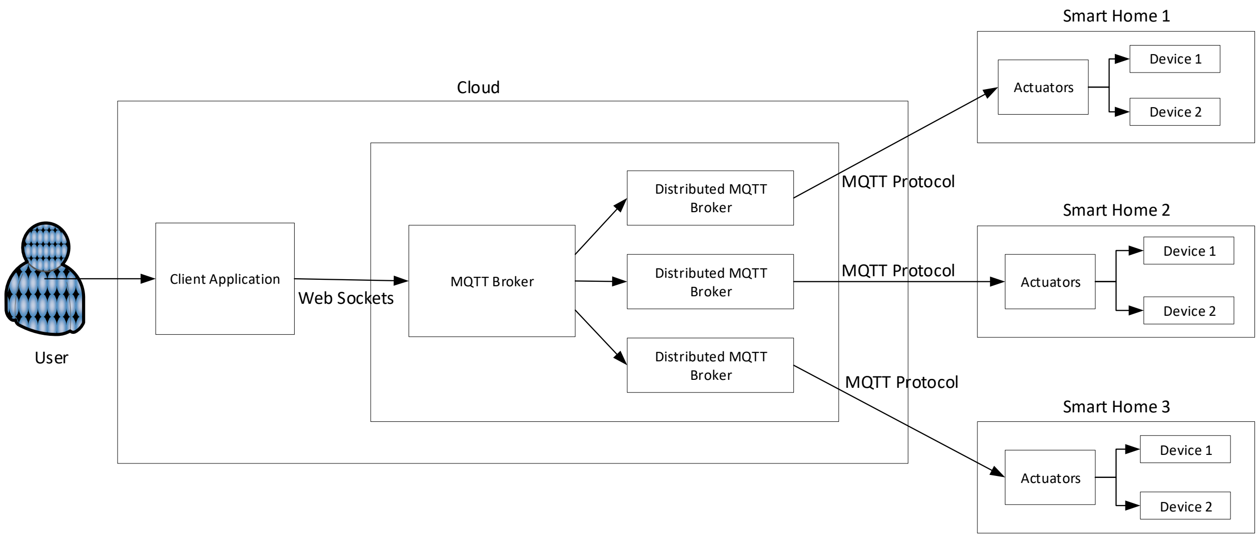

Figure 5. Detailed System Flow Diagram

Figure 5 shows the detailed system flow diagram. User can access the client application remotely from anywhere. The client application is deployed on Amazon Cloud. MQTT broker is also installed and configured on ubuntu running on Amazon EC2 service. Client application and MQTT broker communicate via web sockets. In proposed architecture, there is concept of distributed MQTT broker to overcome the load on the broker. Each broker communicates with smart home via MQTT protocol.

V. IMPLEMENTATION

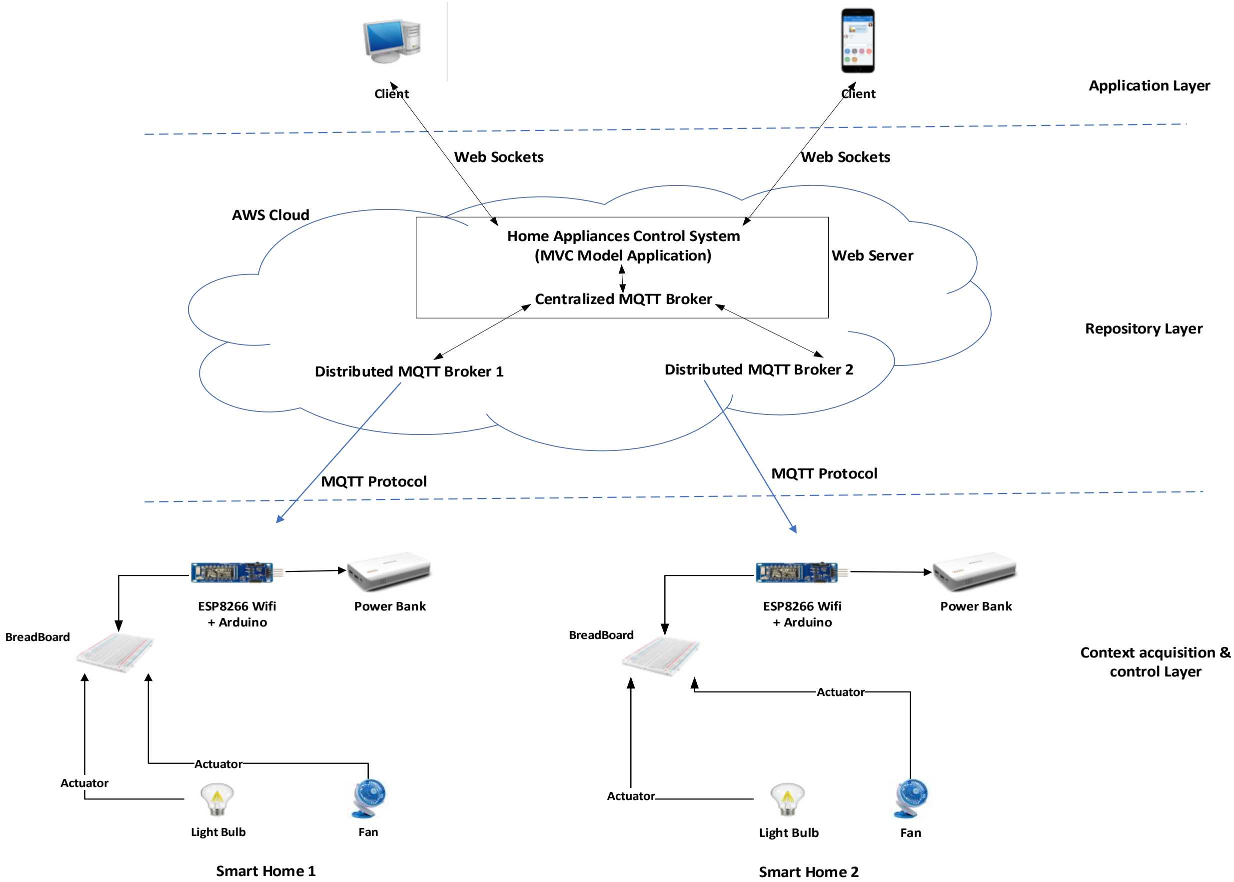

Figure 6. CoT Configuration for Developing Appliances Control in Smart Home

Figure 6 represents the development model of Home Appliances Control System. It can be seen that the client can communicate with the web server via web sockets. MQTT Broker is configured on Web Server AWS Cloud. MQTT Broker communicates with the client machine. We have used ESP8266 Wi-Fi development board in our prototype. In figure 6, it can be seen that light bulb is connected with ESP8266 Wi-Fi module via breadboard. ESP8266 Wi-Fi module is connected with Power Bank.

Table 1. summarizes the Initial set of features used in the experimentation. ESP8266 Arduino Wi-Fi is used to communicate between client application and IoT devices. Different software tools are used to develop client application like Web Storm is used for JavaScript development. Sublime Editor is used for

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

designing web application interface. FileZila is used to upload the client application to Amazon Cloud. PuTTY is used to access ubuntu installed on the amazon cloud.

Table 1: Initial Set of features used for the experimentation

	Component	Description
ESP8266 Arduino Wi-Fi	Model	Arduino ESP8266 Wi-Fi
	CPU	32 bit RISC microprocessor at 80 MHz
	GPIO	16 pins
	Network	IEEE 802.11 b/g/n Wi-Fi
Software Tools	WebStorm	WebStorm 2018.1.3 for Javascript development
	Sublime Editor	Sublime Text 3 for web application design
	File Zila	FileZila 3.36 to upload files on Amazon Cloud
	PuTTY	PuTTY to access ubuntu server.
Programming Languages	Node.js	Node.js 4.8.1
	Angular JS	AngularJS 1.1
	HTML	Html is used for designing web application interface
	CSS	CSS is used for styling the web interface
	Nodemailer	Nodemailer v3 for email notification service.
	Nexmo	Nexmo SMS API for SMS notification service
Cloud Services	Amazon Web Service	Elastic Compute Cloud (EC2)
	Operating System	Ubuntu Server 16.04 LTS (HVM)
	CPU	1 vCPUs 2.5 GHz Intel Xeon Family
	Memory	1 GB

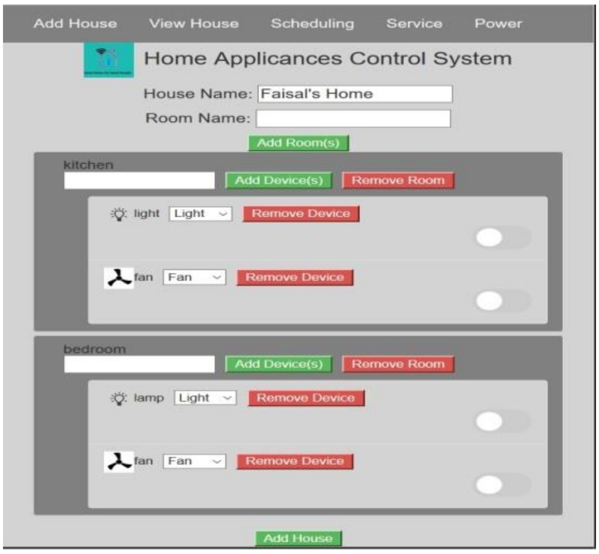

Figure 7. Creating Virtual House

Figure 7 is the interface of the web application which is hosted on AWS Cloud running on Node JS server. In this figure users have the ability to create virtual house. Similarly, they can add multiple rooms to their virtual house and accordingly can add multiple virtual devices in each room. Apart from this, they can also create a virtual design of their houses. In figure 7, a user has added two rooms; kitchen and bedroom. In each room, the user has added light and fan.

Figure 8. Monitoring and Controlling Smart Devices

In figure 8 an interface is shown to allow users to control their home appliances via the Internet. Users can turn different devices ON or OFF using the web application. Users can also control devices of each room and can turn ON the lamp and Fan in Bedroom. Similarly, they can also turn ON light and Fan in kitchen and can add many devices in each room.

VI. EXPERIMENT AND RESULTS

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

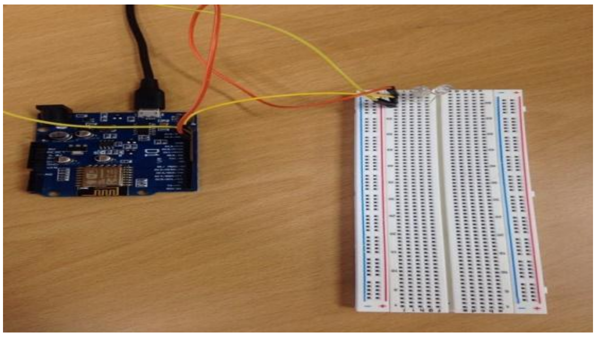

Figure 9. Experimental Environment of Home Appliances Control System

The ESP8266 Wi-Fi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to a Wi-Fi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, it can be simply hooked up to Arduino device and get about as much Wi-Fi-ability as a Wi-Fi Shield offers. The ESP8266 module is an extremely cost-effective board with a huge, and ever growing, community.

This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime.

In figure 10, we can see the results in the console. When a user Turns device ON, we can see the commands in console. Alternatively, when a user turns OFF the device, we can see the result in console.

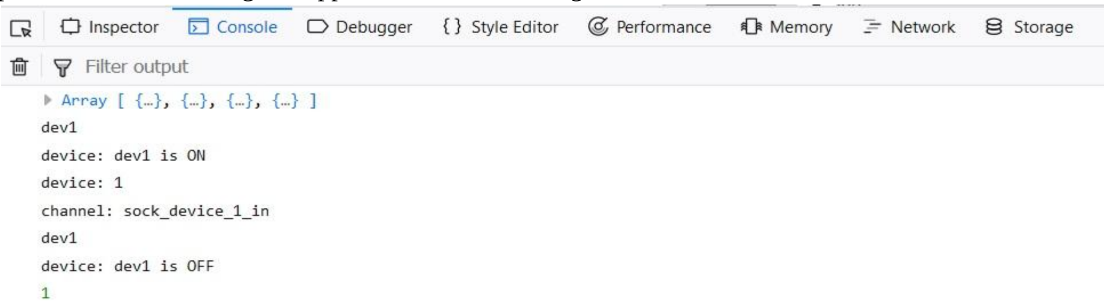

Figure 10. Activation and Deactivation of IoT Device

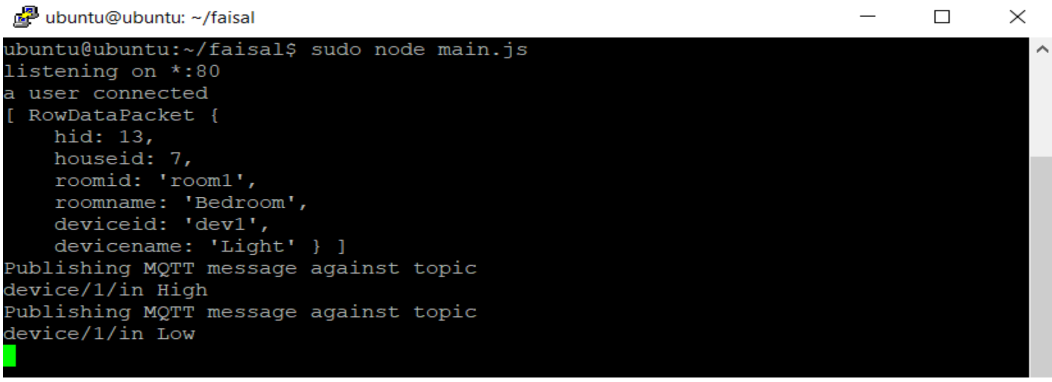

Figure 11. Message publish from MQTT broker to IoT Device

In figure 11, we can see in the console application that MQTT broker is publishing the message to the IoT device. When the message is ‘High’, it means to turn on the device and when the message is ‘Low’ it means to turn Off the device.

VII. CONCLUSION

In this paper, we have successfully designed and implemented a Home Appliances control system that can monitor and control home appliances remotely using MQTT protocol. Due to the load on MQTT broker, we have proposed distributed MQTT broker to overcome the load. We came to conclusion that by implementing distributed broker, the performance of publish/subscribe message is increased. Moreover, the web interfaces of the application is user-friendly and end-users can create virtual design of their home, add rooms, and devices. They can also monitor and control their devices remotely.

ACKNOWLEDGMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government (MSIT) (No.2018-0-01456, AutoMaTa: Autonomous Management framework based on artificial intelligent Technology for adaptive and disposable IoT)), and this research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2018R1D1A1A09082919). Any correspondence related to this paper should be addressed to DoHyeun Kim; kimdh@jejunu.ac.kr.

REFERENCES

1. Xia F, Yang LT, Wang L, Vinel A. Internet of things. International Journal of Communication Systems. 2012 Sep;25(9):1101-2.
2. Juve G, Deelman E, Vahi K, Mehta G, Berriman B, Berman BP, Maechling P. Scientific workflow applications on Amazon EC2. In2009 5th IEEE international conference on e-science workshops 2009 Dec 9 (pp. 59-66). IEEE.
3. Fysarakis K, Askoxylakis I, Soultatos O, Papaefstathiou I, Manifavas C, Katos V. Which IoT protocol? Comparing standardized approaches over a common M2M application. InGlobal Communications Conference (GLOBECOM), 2016 IEEE 2016 Dec 4 (pp. 1-7). IEEE.
4. Kodali RK, Soratkal S. MQTT based home automation system using ESP8266. InHumanitarian Technology Conference (R10-HTC), 2016 IEEE Region 10 2016 Dec 21 (pp. 1-5). IEEE.
5. Sahadevan A, Mathew D, Mookathana J, Jose BA. An Offline Online Strategy for IoT Using MQTT. In2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud) 2017 Jun 1 (pp. 369-373). IEEE.
6. Hunkeler U, Truong HL, Stanford-Clark A. MQTT-S—A publish/subscribe protocol for Wireless Sensor Networks. InCommunication systems software and middleware and workshops, 2008. comsware 2008. 3rd international conference on 2008 Jan 6 (pp. 791-798). IEEE.
7. Ahmad S, Malik S, Ullah I, Fayaz M, Park DH, Kim K, Kim D. An Adaptive Approach Based on Resource-Awareness Towards Power-Efficient Real-Time Periodic Task Modeling on Embedded IoT Devices. Processes. 2018 Jul;6(7):90.
8. Al-Jaroodi J, Mohamed N, Jawhar I, Mahmoud S. CoTWare: A Cloud of Things Middleware. In2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW) 2017 Jun 1 (pp. 214-219). IEEE.
9. Abdelwahab S, Hamdaoui B, Guizani M, Znati T. Cloud of things for sensing-as-a-service: Architecture, algorithms, and use case. IEEE Internet of Things Journal. 2016 Dec;3(6):1099-112.
10. Nunes RJ, Delgado JC. An Internet application for home automation. InElectrotechnical Conference, 2000. MELECON 2000. 10th Mediterranean 2000 (Vol. 1, pp. 298-301). IEEE.
11. Prabaharan J, Swamy A, Sharma A, Bharath KN, Mundra PR, Mohammed KJ. Wireless home automation and security system using MQTT protocol. InRecent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on 2017 May 19 (pp. 2043-2045). IEEE.
12. Tejesh BS, Neeraja S. Implementation of an Efficient Smart Home System using MQTT.
13. Upadhyay Y, Borole A, Dileepan D. MQTT based secured home automation system. In2016 Symposium on Colossal Data Analysis and Networking (CDAN) 2016 Mar 18 (pp. 1-4). IEEE.
14. Druzhkov PN, Erukhimov VL, Zolotykh NY, Kozinov EA, Kustikova VD, Meerov IB, Polovinkin AN. New object detection features in the OpenCV library. Pattern Recognition and Image Analysis. 2011 Sep 1;21(3):384.
15. LeCun Y, Bengio Y, Hinton G. Deep learning. nature 521 (7553): 436. Google Scholar. 2015.
16. Ahmad S, Hang L, Kim DH. Design and Implementation of Cloud-Centric Configuration Repository for DIY IoT Applications. Sensors. 2018 Feb 6;18(2):474.
17. Ahmad S, Malik S, Kim DH. Comparative Analysis of Simulation Tools with Visualization based on Real-time Task Scheduling Algorithms for IoT Embedded Applications. International Journal of Grid and Distributed Computing. 2018 Feb 1;11(2):1-0.
18. Lin YC, Hung MH, Huang HC, Chen CC, Yang HC, Hsieh YS, Cheng FT. Development of Advanced Manufacturing Cloud of Things (AMCoT)—A Smart Manufacturing Platform. IEEE Robotics and Automation Letters. 2017 Jul;2(3):1809-16.