Smart Community Monitoring System using Thingspeak IoT Plaform

Article *in* International Journal of Applied Engineering Research · September 2018

2 authors:

David Nettikadan
Jyothi Engineering College

Subodh Raj M S
Sahrdaya College of Engineering & Technology

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 17 (2018) pp. 13402-13408

Smart Community Monitoring System using Thingspeak IoT Plaform

David Nettikadan[1] and Subodh Raj M.S.[2]

[1] *Department of Electronics and Communication Engineering,*
Jyothi Engineering College, Kerala, India.

[2] *Department of Electronics and Communication Engineering,*
Sahrdaya College of Engineering and Technology, Kerala, India.

Abstract

The world especially homes are getting smarter on a daily basis, by the arrival of the Internet of Things (IoT). As man is a social being the smart homes must not be standing alone, they must collaborate each other to build a smart community. Smart community is a network of smart homes in a local geography. The smart community acts as an intermediate between smart homes and smart cities. Various papers propose smart community architecture where all homes in it are having the same functionality. But that is not the case in the real-world scenario. In a society, various homes will be needing various functionalities. Each home will have to monitor different parameters using different sensors and have to provide different responses and services accordingly. This paper presents a monitoring platform for a smart community, of which there are three custom-designed smart homes having different functionalities. The data is sent to ThingSpeak IoT platform via MQTT protocol which will be displayed in charts. The IoT platform also alerts the community manager and other homes in case of emergency using the ThingHTTP, React and TalkBack app in the ThingSpeak. Device control is based on time is implemented using TimeControl and based ib Tweet messages is implemented using TweetControl app. The monitoring platform was successfully implemented and was found functioning well.

Keywords: Smart Community, Smart Home Management System, Internet of Things (IoT), ThingSpeak, MQTT.

INTRODUCTION

Internet of Things (IoT) is a network of everyday objects often called as ‘things’, which are interconnected to each other. Internet of things is popularly used in making everyday appliances smarted, especially homes. These devices and record the surrounding conditions, or user activity and predict their future behavior. This helps the user to be prepared everything one step ahead according to one's preferences, comfort, and convenience. As the man being a social being, requires him to collaborate with each other. Similarly, the smart homes must not be standing individually, but collaborate each other building smart communities and societies. Dr. Mal Bryce pointed out that “our primary economic and social challenge for this first decade of the new millennium is to harness the new economy and create the new community…one that is an exciting place in which to live and work. The new economy is the Global Knowledge Economy and the new community is the Smart Community”. [1]

One of the simplest definitions on the Internet of Things is made by the RFID group as, "The worldwide network of interconnected objects uniquely addressable based on standard communication protocols." [2] There are three components for IoT systems- a) Hardware section which includes the sensors, actuators and the communication sections. b) Middleware is used to analyze the data received from the sensors by doing necessary computations after storing it. c) Presentation to visualize and interpret the data in an easy manner. IoT is used in smart homes, enterprise based applications, providing utilities, smart logistics and transportation and healthcare.

SMART COMMUNITIES

In simple words, a smart community is a “virtual environment composed of networked smart homes located in a local geographic region. It is formed upon the agreement of participating homeowners, with respect to local geographic, terrain, and zoning features.”[3] But in a broader sense, we can tell it as a group of a connected object over the ubiquitous network and the objects interact with each other to deliver smart services for all its members. The size of the smart community can vary. They are evolving over time, becoming smarter and smarter.

In technical language, a smart community can be defined as “a multihop network of smart homes that are interconnected through radio frequency following wireless communication standards such as WiFi (IEEE 802.11) and the third generation (3G) of mobile telephony.” [4] It is “a cyber-physical system, in which homes are virtually multifunction sensors with individual needs, continuously monitoring the community environment from various aspects; and, when necessary, automatic or human-controlled physical feedback is input to improve community safety, home security, healthcare quality, and emergency response abilities.”[4]

The smart homes are built by integrating three domains- the community domain, the home domain and the service domain. The community domain is the connected network of homes where the information from individual homes is disseminated. The home domain consists of a continuous real-time monitoring system to monitor the healthcare, security and environmental details. The service domain works as a call center which comprises of a communication device to communicate with trusted parties like medical team, fire or police department etc. The decisions are made on the basis of sensor data and the call is made to service provides or proper authorities through reliable communication channel [4].

Building a smart community is really sophisticated and requires many supporting techniques to overcome its technical challenges. One of the main challenges in building a smart community is that it has an integrating custom-designed smart home into a smart community. The requirements and features a homeowner wants from his smart home will be different from the other. Integrating the different smart home under one roof is a big challenge and is dealt with this project.

This paper organization is as follows: The related works done on smart communities is discussed in Section II. Section III put forwards the proposed system architecture and also explains about the MQTT communication protocol and ThingSpeak IoT platform. Section IV describes how the hardware is implemented and the apps in the ThingSpeak is configured in the smart community monitoring platform, it also evaluates the results provided by the implementation. The work is concluded in Section V.

RELATED WORKS

Various papers dealing with the smart community or MQTT implementations discussed here. The paper [4] is the first paper to introduce the concept of smart communities as an application of internet of things. It refers smart community to a network of cyber-physical systems (smart homes) as cooperating objects. It defines how the architecture of a smart community should be and explains the various methods to realize it. It discusses potential applications of smart community and the various challenges that are to be faced when building one smart community.

The project paper [3], develops a small smart community based on Internet of Things. It designs and develops a small smart community having two smart homes. These smart homes monitor various parameters like temperature, door status, and light intensity, and based on these details the fan/AC, tube light and security alarm is turned on and off. The data collected from the sensors are saved in a database system. A website is used to build an interconnection between the sensors and database. This web server can be accessed through the smartphone, computer or laptop via WiFi.

The study [5], builds a stand-alone, cheap and flexible Rasberry Pi-based home automation system to control multiple appliances which can be monitored online. Each homes are Arduino based smart homes with Nrf module to communicate with the server. The parameters measured by the sensors are sent to database via webpage and is is monitored frequently to avoid threats. As various IoT protocols are possible for communication, this paper compared various IoT communication protocols such as MQTT, HTTP, and CoAP and various observations are made.

The paper [6] proposes SeSAMe, a Software Defined Network (SDN), for smart homes alert management to be used in smart cities. In case of any incidents like fire, this system sends alerts to police, fire departments and other homes. It proposes a decentralized communication pattern, ie communication is done home to home, not home to the server, which makes the communication among homes very fast, but makes the service request from the server slower. This paper also defines different types of messages that are used to communicate between home to home or home to the controller. As the network is software defined, it is more flexible and manageable.

The paper [7], proposed a smart community where the community broker integrates the community services. This reduces the workload of community management staff. It provides better information services among the community and integrates the community with the surrounding environment in a superior way. The home end is an intranet where a fixed touch panel is integrated into the home controller. The home controller monitors various environmental, security and energy conditions using different sensors. Community server consists of the community computer server and the connected devices like video camera and other automation devices. This system uses MQTT for home control services and HTTP for delivering locations based integration.

The paper [8] proposed an architecture for smart cities which is cloud based. This architecture helps to obtain the real-time data gathered from the city by community service providers, citizens and city management. The sensor data is analyzed and future planning is done on the decisions made. It improves the quality of life for the inhabitants of the city. This has applications in sustainable communities, urban planning, transportation, public security, commerce and healthcare.

The paper [9] describes ThingSpeak API (Application Programming Interface) and web service for IoT. Practical examples of ThingSpeak interfacing is provided for the Arduino microcontroller. To communicate with the graphical interface, python script is used. Authors discuss the strengths and weakness of the platform along with its potential applications. The paper suggests Thingspeak for small hardware projects where dedicated communication server is not practical. But the system necessitates continuous connectivity over the internet.

The paper [10] presents a working model of a Thingspeak IoT platform. The project uses an Arduino Uno board and ESP8266 Wifi module to connect to the internet. The system uses a number of sensors and the data has been uploaded to ThingSpeak server and the data is visualized. The paper has not only received and visualized sensor data but also done a Matlab analysis of the data and plotted the data.

PROPOSED SYSTEM ARCHTECTURE

This paper develops a monitoring platform for IoT based smart community system for three custom designed smart comes. In this section, system architecture for the proposed system is described along with a detailed description of the IoT communication protocol used named MQTT

System Archtecture

The system architecture of the proposed system is shown in figure 1. The system consists of mainly four sections: three smart home automation system parts and one community server part. The smart homes are connected to WiFi and communicate to the server via MQTT protocol. The architecture of the proposed system is as follows. The Home automation system consists of multiple sensors to monitor the environment and security conditions. The value of sensor data is used to control devices or to trigger an alarm. The sensor data would be sent to ThingSpeak IoT platform using MQTT protocol. The data will be visualized in charts and can also trigger various activities like sending alert message via twitter or alert other homes based on various conditions. The user can monitor the security and environmental conditions of their respective homes via smartphone.

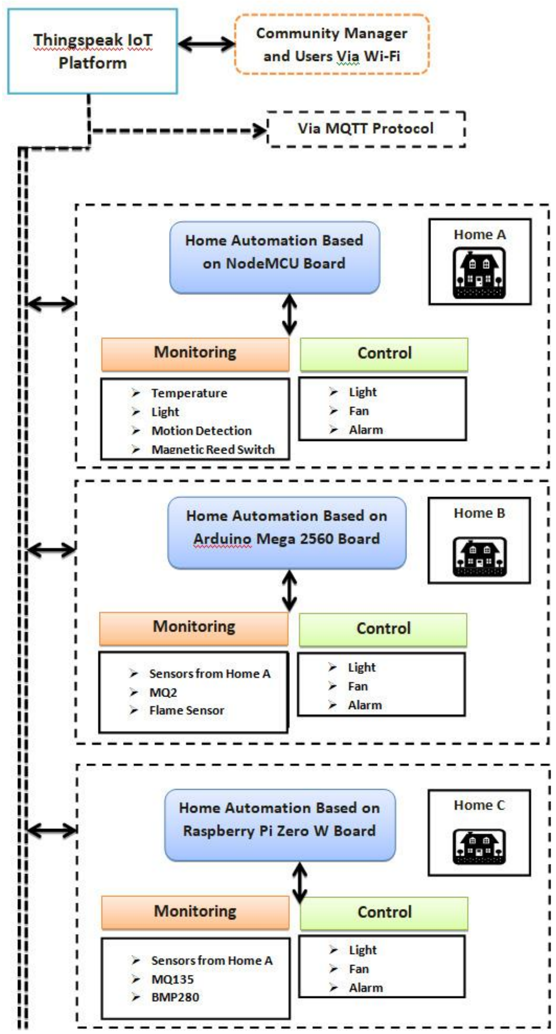

Figure 1: The Proposed Smart Community Architecture

MQTT (Massage Queuing Telemetry Transport) Protocol

Various protocols and methods have been developed for data communication in the application layer as the field of IoT develops. MQTT (Message Queue Telemetry Transport), CoAP (Constrained Application Protocol), XMPP (Extensible Messaging and Presence Protocol), AMQP (Advanced Message Queuing Protocol) etc are only a few to mention. MQTT was invented by Andy Stanford Clark and Arelen Nipper in 1999. The primary intention of this protocol is to reduce the power and bandwidth usage with a vision to be used in embedded applications. Because of these advantages it gains wide popularity. Another advantage is that the client side implementation is very simple as the complexity of the system resides in the broker side. It is also made royalty free few years back increasing its popularity.

It is a based on publish subscribe communication model. The component which produces the information publishes the information, so is called publisher. The party interested to receive the published information is called the subscriber. Broker ensure that the subscriber receive all the data that has been published without any loss. MQTT is topic based publish subscribe system. This means that the publisher publishes the information to a topic which can be subscribed by any subscribers. A subscriber can subscribe multiple topics.

The architecture of MQTT is described in figure 2. The publisher publishes the data along with the topic and is send to the broker, which acts as an intermediary between publisher and subscriber. The subscriber makes a request for a specific topic and as a response the broker sends the data along with the topic to the subscriber. MQTT is working on default TCP/IP port 1883. MQTT broker services are provided by various servers hive-mq, mosquitto, pahomqtt.

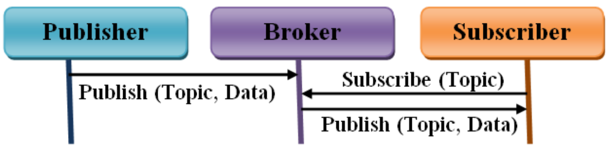

Figure 2: MQTT process

There are various advantages of MQTT. Compared to traditional client- server models, this provides more scalability. But it is not scalable to the range of millions of connections. It also provides decoupling in time, space and synchronization decoupling. That means that publisher and subscriber need not run simultaneously, need not know each other and the process is not halted during publish or receive process. It supports username and password for authentication making the communication more secure.

ThingSpeak IoT Platform

ThingSpeak is an IoT platform which allow one to collect, visualize, analyze live data and react according to it. It is open source application originally launched in 2010 by ioBridge. It helps one to build IoT systems without need of setting up extra servers. The data collection is done using REST API or MQTT. The data analysis and visualization is done using MATLAB analytics. There is also option to add various plugins that enable user to display google guage and other custom visualization and controls in private view.

Various actions can be done using apps provided by the platform. ThingSpeak provides apps that allow us for an easier integration with the web services, social networks and other APIs. It include ThingTweet app to tweet alerts and messages, TweetControl app which respond to tweet having some trigger words, TimeControl App to perform or schedule some specific actions, React app to do some actions when some conditions are met, Talkback app to send command to devices and ThingHTTP app to interface with various web services and APIs.

The main component of ThingSpeak is its channel which stores data send from various devices. Each channel can save up to eight fields along with device location, url etc. The channel can be made public which can be seen by other users or private which need the API key to view the data. The private channel can be shared for some specific users.

IMPLEMENTATION, RESULTS,AND DISCUSSIONS

Smart Home Automation Systems

Three Smart Home Automation Systems are made to monitor the environmental and security conditions of the respective homes. Each of the three homes is having different controllers and the functionalities of the homes vary. All the homes are having some basic sensors as common: Temperature and Light Sensor to monitor environmental conditions and Magnetic Reed Switch and PIR Sensor for monitor security conditions. There are many other sensors like gas sensor, flame sensor used to monitor various conditions of the homes. The hardware architecture of the Smart Home Automation System is shown in figure 3.

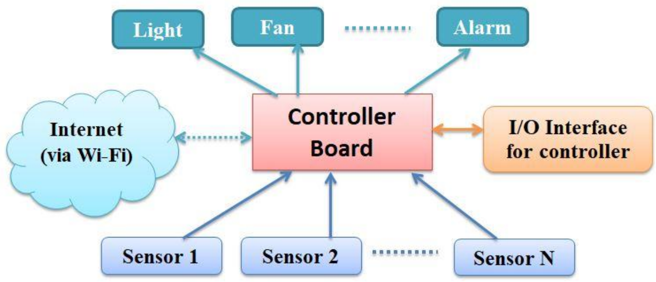

Figure 3: Architecture of Smart Homes Automation System

Various hardware modules used in smart home automation systems are mentioned in Table 1 and the details of sensors used are narrated in Table 2. Different I/O interfaces are used to see and change the system settings in various smart homes. Interfacing from a smart phone is done via Bluetooth module HC-05 in one smart home. In the second home input was given through 4*4 keypad and output through 16*2 LCD. In home based on Raspberry Pi 3, touch screen is used for interfacing.

Table 1: Hardware modules used in smart homes

Component	Description	Used in
NodeMCU	Programmable WiFi module, 10GPIO, 1ADC pin, PWM functionality, I2C and SPI communication.	Home 1 Controller
Bluetooth Module-HC05	Bluetooth Serial Port Protocol. Used as master or slave. +4dBm RF transmits power. UART interface. Integrated antenna	Home 1 I/O Interface
Arduino Mega2560	54 digital IO, 16 analog Input, 4 uart, 16MHz crystal oscillator, USB connection, power jack, ICSP header, reset button.	Home 2 Controller
WiFi Module-ESP8266	SOC with integrated TCP/IP protocol stack. Wi-Fi direct (P2P), soft-AP, +19.5dBm output power in 802.11b mode.	Home 2 to connect Controller to internet
16*2 LCD Display	16 character by 2 line Liquid crystal display. 5*8 dots with cursor, Built in controller	Home 2 output interface
4*4 Keypad	Matrix membrane keypad, 16 button keypad, ultra-thin design, 8 pin access.	Home 2 input interface
Rasbperry Pi 3	1GHz, Single-core CPU, 512MB RAM, HAT-compatible 40-pin header, 802.11n wireless LAN.	Home 3 Controller
7" Touch Screen Display	Screen Dimensions: 194mm x 110mm x 20mm, Screen Resolution 800 x 480 pixels, 10 finger capacitive touch.	Home 3 I/O interface

Table 2: Sensors used in home monitoring system

Component	Description
LM35-Temperature sensor	Semiconductor based sensor- calibrated in Celsius, with linear scale factor +10mv/ºC. Range: -55ºC to 150ºC.
LDR based Light Sensor	Detecting presence of light- high in presence of light, low in absence of light. Adjustable sensitivity using a potentiometer.
Reed Switch Sensor Module	Normally open reed, digital output, working voltage 3.3v-12v. Using wide voltage LM393 comparator.
PIR Sensor Module	Pyroelectric/ Passive Infrared Sensors. Detect human presence. Small, inexpensive, low-power module. Range 6 meter.
MQ2 Gas Sensor	Gas sensor module for gas leakage detection. For detecting H2, LPG, CH4, CO, alcohol, smoke or propane.
MQ135 Gas Sensor	Air pollution gas sensor- for air quality control equipment. Detect NH_3, NO_x, alcohol, Benzene, smoke, CO_2.
Flame Sensor	Detect flame or light source of wavelength within 760nm~1100nm. Based on YG1006 sensor. High photo sensitivity.
BMP280 Pressure and Temperature Sensor	Barometric pressure and temperature sensor. Low cost, high precision sensor. Communicate with I2C and SPI.

The sensor data is used to actuate different home appliances like bulb, fan, AC etc. For example the fan automatically turns on when the temperature exceeds 30 degree Celsius. The bulb turns on and off when the light intensity goes below or above certain thresholds etc. When the security mode is on the system produces alarm if the triggering condition is met. When the security mode is turned off the alarm will not be triggered but the sensor status will be send to the IoT monitoring platform. The user can decide which all sensor's data has to be send to the monitoring platform. If the user disables certain sensors the data won't be sent to the Thingspeak server.

ThingSpeak IoT Platform

Channel Visualization: The system uses ThingSpeak IoT platform for monitoring the data. The system uses three channels for monitoring the data, one for each homes. The channel one consists of five fields and channel two and three consists of six fields. Android app named ThingChart is used to visualize and monitor the data through a smart phone.

React App: React app Send a tweet or trigger a ThingHTTP request when the Channel meets a certain condition. In this project when the flame sensor of any house is triggered it triggers ThingTweet to send warning message to the community manager, the configuration of which is show in figure 5. The react app is also used to trigger the ThingHTTP app to warn other houses about the fire.

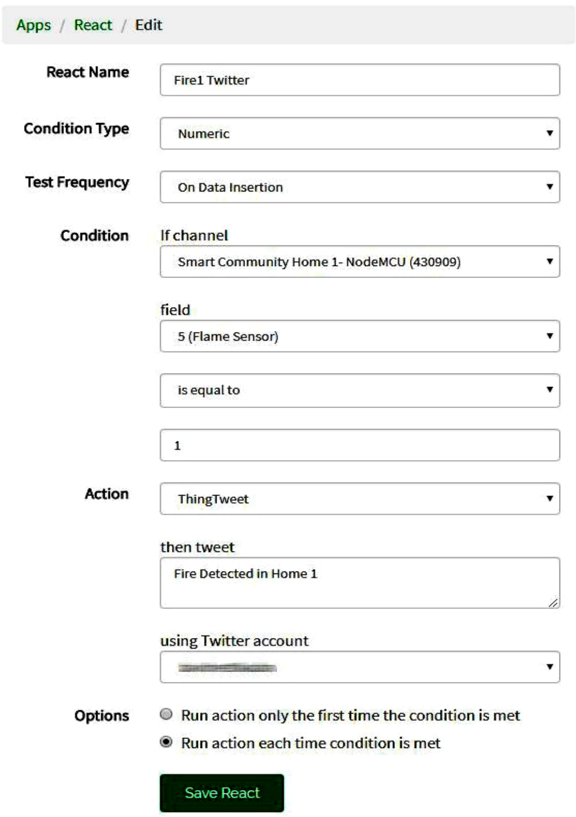

Figure 5: Configuration of React app to send a warning Twitter message to community manager when fire occurs at home 1

ThingHTTP App: ThingHTTP enables communication among devices, websites, and web services without having to implement the protocol on the device level. It is done using the GET, PUT, POST and DELETE methods of HTTP. In this project The ThingHTTP app is used to add a talkback command to each houses when fire is detected in the other houses. An example of ThingHTTP is shown in figure 6.

TalkBack App: TalkBack is used to queue up commands and then allow a device to act upon these queued commands. TalkBack API is used to add, get and execute a TalkBack command, which can be accessed by ThingHTTP app. In this project TalkBack app is used to add commands for device control based on time or twitter message and warning in case of fire. An example of Talkback configuration is shown in figure 7.

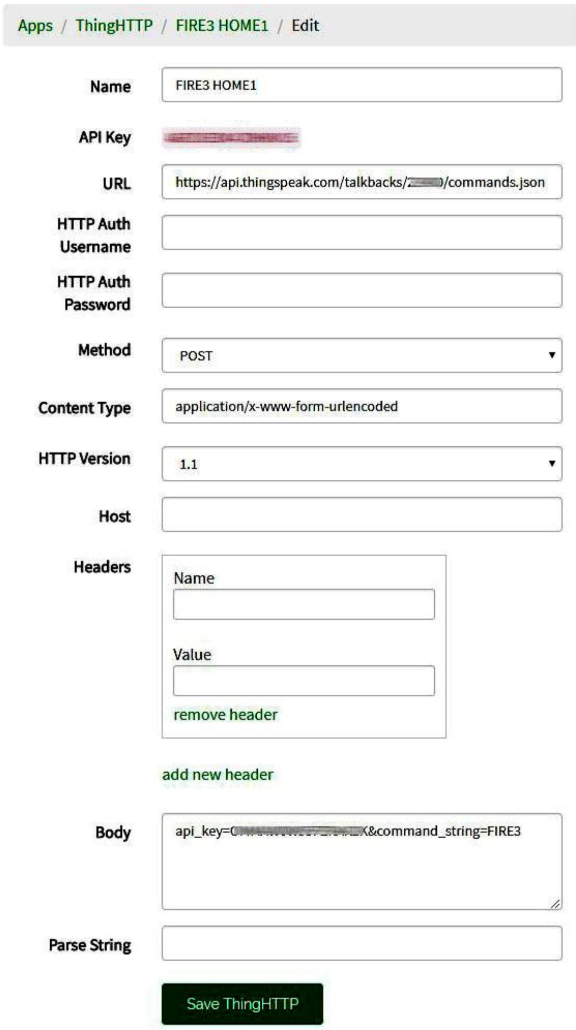

Figure 6: Configuration of ThingHTTP app to add a warning command to TalkBack app of home 1 when fire occurs at home 3.

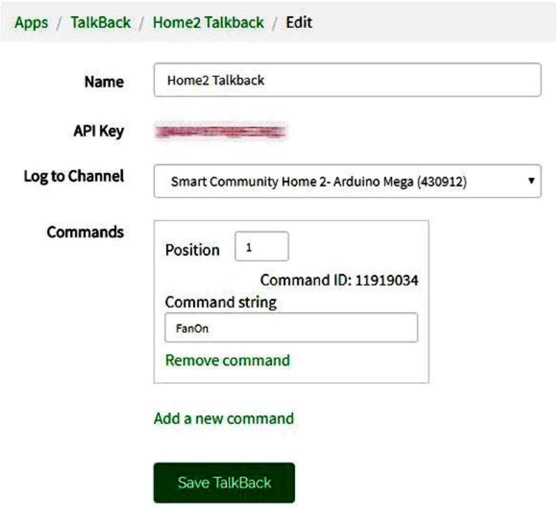

Figure 7: Configuration of TalkBack app to add a warning messages and device control commands to home 2.

TimeControl App: Through this app one can do a ThingTweet, ThingHTTP or a TalkBack at a specified time in the future. In this project the devices is turned on and off at preset times in home 2. An example of TimeControl configuration is shown in figure 8.

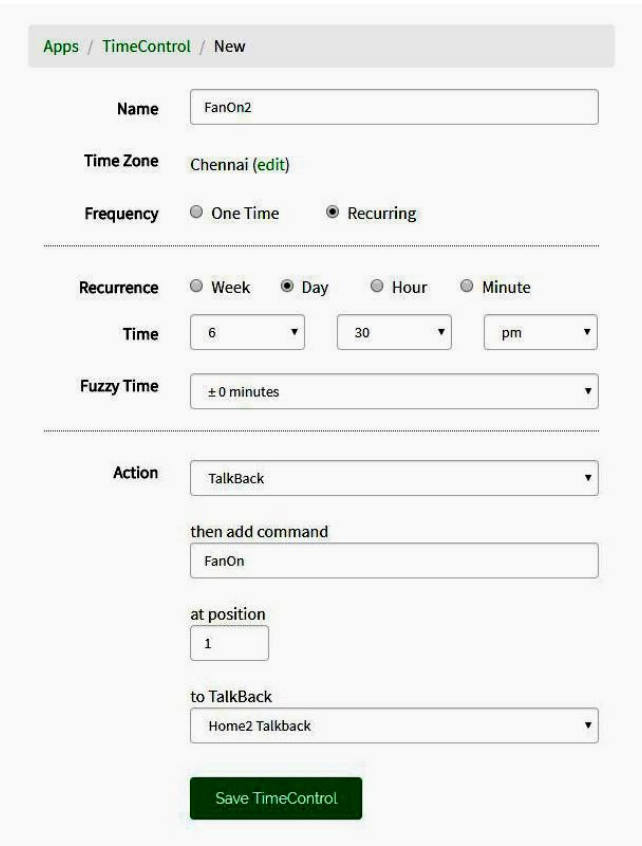

Figure 8: Configuration of TimeControl app to schedule the FanOn command to home 2.

TweetControl App: Using this, you can monitor your Twitter feeds for a specific key word and then process the request. Once the specific keyword is found in the twitter feed, you can then use ThingHTTP to connect to a different web service or execute a specific action. In this project home 3 uses TweetControl app to turn on fan when the trigger word #home3 #fanon is tweeted by the user. An example of TweetControl configuration is shown in figure 9.

Figure 9: Configuration of TweetControl app to respond to trigger word #SMCOMM #Fanon3 to turn on fan at home3.

If some cleaning or maintainance request is made by the users by dialing some predefined codes, the respective messages will be tweeted to the community manager. The community manager can send some talkback message to the smart homes which will be displayed in the output screens of the respective devices. The above two options enables the two way communications between the smart homes and the smart community.

RESULTS AND DISCUSSIONS

The system was able to operate in a good performance. The sensor data was monitored and based on the data the home devices were controlled. The light and fan were turned on and off according to the environmental conditions and the alarm was triggered when a security condition arises when the security feature was turned on.

The data form homes are sent to ThingSpeak server using MQTT protocol and was visualized in three channels. The channels receive sensor data from homes at an interval of 15 seconds, and are visualized as line graphs in the channel. figure 9 shows the visualization of sensor data in Home 2. The data visualization is also done in smart phone using android app ThinkChart, which is show in figure 10.

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 17 (2018) pp. 13402-13408

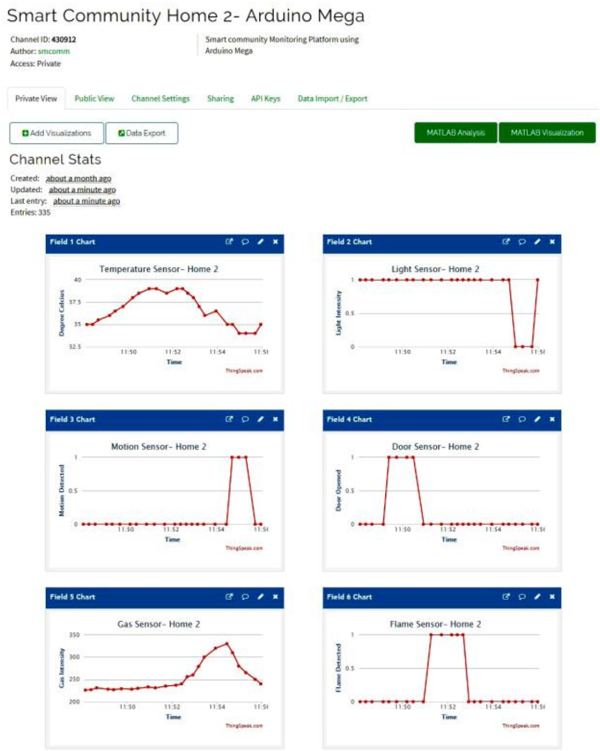

Figure 9: Sensor data of smart home 2 visualized in ThingSpeak.

Alert message was sent to the community manager and other homes when fire occurred at other houses. Devices were time controlled on preset schedule in home 2 and were able to be controlled by predefined twitter messages in home 3.

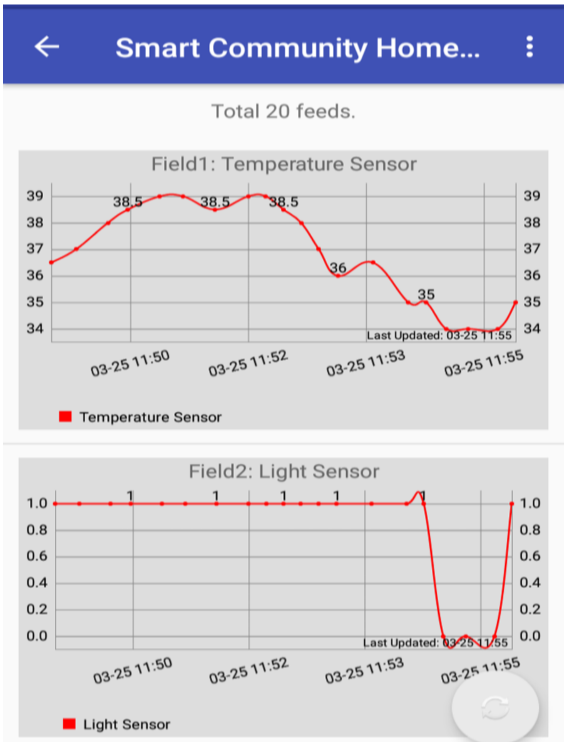

Figure 10: Sensor data of smart home 3 visualized in ThingChart android app.

CONCLUSION

Designing and implementation of IoT based smart community monitoring platform for custom designed smart homes is successfully completed. The system is able to monitor environmental and security conditions of the homes and switch on fan and light based on environmental conditions and alert the security alarm based on the security conditions. The community was able to receive data from smart homes to ThingSpeak platform via MQTT protocol, save the data in the database and visually display in the ThingSpeak webpage. The system functioned well as designed. The system was also able to be monitored remotely using ThingChart android app.

REFERENCES

[1] Lindskog, H., 2004, "Smart communities initiatives," Proceedings of the 3rd ISOneWorld Conference, Vol. 16, pp. 14–16.

[2] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M., 2013, "Internet of Things (IoT): A vision, architectural elements, and future directions," Future generation computer systems, 29(7), pp. 1645-1660.

[3] Y. Gunardi, A. Adriansyah, and T. Anindhito, 2015, "Small smart community: An application of internet of things," ARPN Journal of Engineering and Applied Sciences, 10(15), pp. 6341-6347.

[4] Li, X., Lu, R., Liang, X., Shen, X., Chen, J., & Lin, X., 2011, "Smart community: an internet of things application," IEEE Communications Magazine, 49(11), pp.68-75.

[5] Joshi, J., Rajapriya, V., Rahul, S. R., Kumar, P., Polepally, S., Samineni, R., & Tej, D. K., 2017, "Performance enhancement and IoT based monitoring for smart home," International Conference on Information Networking (ICOIN), pp. 468-473.

[6] Abhishek, R., Zhao, S., Tipper, D., & Medhi, D., 2017, "SeSAMe: Software defined smart home alert management system for smart communities," IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), pp. 1-6.

[7] Lee, Y. T., Hsiao, W. H., Huang, C. M., & Seng-cho, T. C., 2016, "An integrated cloud-based smart home management system with community hierarchy," IEEE Transactions on Consumer Electronics, 62(1), pp. 1-9.

[8] Jalali, R., El-Khatib, K., & McGregor, C., 2015, "Smart city architecture for community level services through the internet of things." 18th International Conference on Intelligence in Next Generation Networks (ICIN), pp. 108-113.

[9] Maureira, M. A. G., Oldenhof, D., & Teernstra, L., 2011, "ThingSpeak–an API and Web Service for the Internet of Things," Available online: http://mediatechnology.leiden.edu/images/uploads/docs/wt2014_thingspeak.pdf, Accessed on 25 March 2018.

[10] Pasha, S, 2016, "ThingSpeak based sensing and monitoring system for IoT with Matlab Analysis," International Journal of New Technology and Research (IJNTR), 2(6), pp. 19-23.