IoT-Fog-Cloud based architecture for Smart City: Prototype of a Smart Building

Joy Dutta and Sarbani Roy
Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India
joy.dutta.in@ieee.org, sarbani.roy@cse.jdvu.ac.in

***Abstract* — Here, we present a prototype of a smart building using newly surfacing technologies like IoT (Internet of Things), fog and cloud for the smart city. The demand for everything smart is increasing daily, but the main stumbling block is its high price. So, our aim is to improve the standard of living in home and in office with newly improved working facilities where the whole system will be automatic, efficient and will be under the control of the user via his/her smartphone or computer but the cost will stay within the budget of a common man. All these are done by the incorporation of IoT, fog and cloud. The assimilation is done using open source hardwares and softwares to reduce the cost dramatically than the other existing solutions and implement it in an impressive and ingenious way without compromising QoS (Quality of Service) of any of the functionalities provided by other existing solutions.**

Keywords— ***IoT; Fog; Cloud; Smart Building; Automation; Intelligent Control; Energy Efficient***

I. INTRODUCTION

The word *smart* is a buzzword nowadays and used to denote a feather to our standard of living. In this busy world, automation is needed to simplify our daily life. Today's people want to monitor and control appliances effortlessly. A smart building is a building that is equipped with different electrical and electronic devices that can be monitored and controlled by smart phones or PC. These buildings are the basic blocks of our society as it is used for housing complexes, office purposes, shopping malls or shops and so on. So, to make them smart is a great step towards smart city. Smart buildings also minimizes the domestic energy waste by controlling the high energy demanded building appliances intelligently.

With the growing popularity, IoT provides great opportunities for making smart building application. To address the varied needs of the users considering technological and social changes, the system must be designed as being adaptable and responsive. However, the main challenge here is to make the cost of the system affordable without compromising with its quality of service (QoS). IoT consists of physical objects, sensing devices and communication infrastructure, but lacks computation and storage infrastructure. Thus, to bridge this gap, IoT is integrated with the cloud. To propose a smart building solution, we need to handle a large set of heterogeneous data coming from various sensors, in different speed and with a different meaning. As the numbers of sensors increases, the rate of data generation along with variety and volume is also increasing. As a result, the smart building system is generating actually big data when you are taking into account all these sorts of data. So, a smart building solution should react to real and physical world events in real time by running processes in cloud and fog that basically triggers appropriate actions and services with or without human intervention.

Nowadays, smart phones and tablets are increasingly becoming an essential part of human life as the most effective and convenient communication tools not bounded by time and space. In the proposed smart building system, users can easily monitor and control the building appliances through their smart phones both locally and remotely. Fig. 1 depicts the envisioned remotely controlled smart building using smartphone or computer [7].

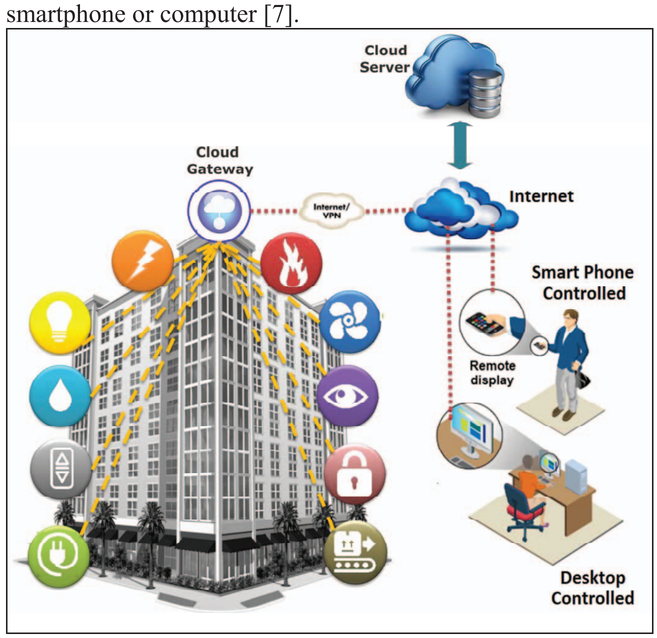

Fig 1. Facilities provided for Smart Building

Some relevant smart building related solutions based on sensors have been described in [1-4]. In paper [1] Smart homes based on WSN protocols are used to provide an assisted living environment to humans, which provides a safe environment for the well-being of its inhabitants. Paper [2] presents a smart home management system in which a community broker role is used for integrating community services, thereby reducing the workload of community management staff, providing electronic information services, and deepening the community's integration with the surrounding environment. Furthermore, to achieve multiple in home displays, standard interface devices can be employed to separate the logic and user interfaces. In [3] authors claimed that energy is saved using wireless sensor network (WSN) and power line communications (PLCs) by transferring control messages for home appliances using PLCs directly rather than via WSNs. Whereas in [4], the authors proposed a solution for full home control from a remote location using GSM modem. This smart GSM based home automation system can control home appliances from a remote location via SMS text messages.

In contrast to [1-4], our proposed solution provides not only an intelligent, green and connected building, but also a building which gives users a real-time control over all the appliances present there. It uses the cloud [10] service to control all the devices along with real time notification to the user directly. With all these facilities it is providing a smart building solution. Moreover, the price of our smart building system (for a single 2BHK unit will be approximately 7500 INR- without the router). This solution is very inexpensive as well as flexible. Moreover, due to the presence of the fog enabled router in the architecture, better QoS (Quality of Service) can be achieved, e.g., decreasing the effective latency time in communications.

This paper is organized as follows. In Section II, we present the design of the proposed architecture. Prototype functionalities are described in section III and Section IV discusses the implementation of the system. Then it is evaluated in Section V. Finally, we conclude our work in section VI.

II. IoT-Fog-Cloud based ARCHITECTURE

Things of any standard building that we want to consider as things of IoT in the smart building application are actually not pluggable as it is. To make them IoT things, they have to give the power of sensor, processor (low-power), and a way of communication (usually wireless). In smart building application, these things may connect directly to the smart phones or nearby Internet gateway devices .

The prototype is built using Arduino Uno, Sensors, Bluetooth Module, WIFI module, fog gateway server, the cloud platform (for online data storage and analysis) to control building appliances from anywhere in this world. Things which are part of the smart building are controlled using only a smart phone or any mobile devices (e.g., tablet, laptop etc.) or even using a PC also. Our envisioned smart building consists of smart office, living room, corridor, kitchen, lift, garage, entrance in the building, security, pumping system, which are not only just automatic, but also has the inbuilt intelligence to respond back depending on environmental conditions. The system components can work on their own with their inbuilt intelligence.

However, the idea is to have the option to overwrite the system's control from the smart phone or computer. So that the user can monitor and control their room remotely. Another facility is that, if some appliances fail, then it can directly send a mail to the registered user or can even inform him/her by calling about the system status. This prototype is able to automate our building, curtail unnecessary energy consumption, reduces human effort for its monitoring and control, thus helps us to be a part of smart city.

A. Layer Descriptions

The proposed smart building system is designed as a three tier architecture as shown in Fig. 2. Description of each layer is given below.

1) IoT Layer

In tier 1, sensing and communication capabilities are incorporated within the physical things of building to get them ready for IoT. Fig. 3 shows the basic circuit diagram of this process.

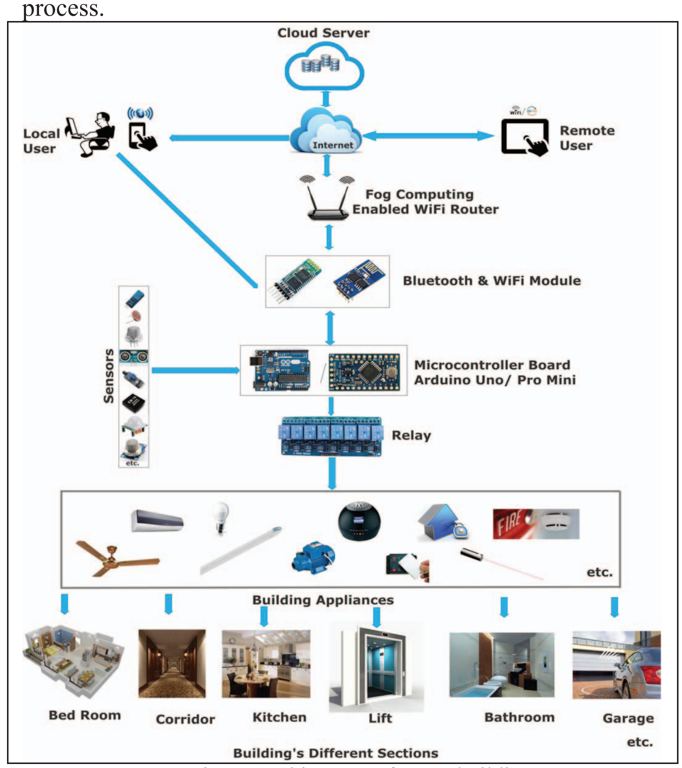

Fig. 2. Architecture of smart building

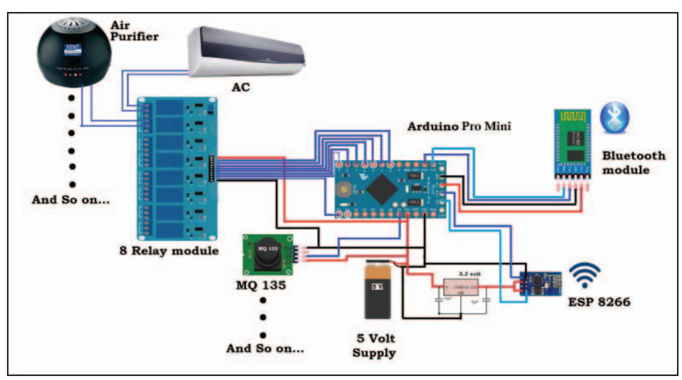

Fig. 3. Things to smart things -basic circuit diagram

A microcontroller is used for controlling things equipped with sensors and communication units. Now, these things can sense the environment and also can communicate with other devices using Bluetooth or WiFi. In smart building application, multiple such things can be present in close proximity. Thus to manage a group of high voltage electrical devices by Arduino, a relay module is used.

2) Fog Layer

In tier 2, the fog gateway server is maintained, which can take some local decision based on sensor's data. This layer is responsible for collecting, filtering and preprocessing the data. It also works as an internet gateway which actually connects IoT with internet using VPN.

3) Cloud Layer

In tier 3, data from the fog layer is forwarded to cloud for storage and computation. Here, a database is maintained for cloud server, which is used for controlling the IoT from smart phones or computer.

B. Hardware

As discussed, different hardware modules are used to make simple things IoT enabled. In this work, we have used Arduino [5, 8-9] which is an open-source prototyping platform based on easy-to-use hardware and software. Table 1 and 2 describe different hardware modules and sensors used in the proposed smart building application.

Table 1. Description of hardware modules

Component	Description	Placed in
Arduino Uno / Pro Mini	A microcontroller board based on the ATmega328. It has 14 digital input/output pins.	Each Switch Board
Relay	An electrically operated switch that allows to turn on or off a circuit using voltage and/or current much higher than the Arduino could handle.	Each Switch Board
WIFI module-ESP8266	A WiFi Module is a self contained SOC with integrated TCP/IP protocol stack.	Each Switch Board
Bluetooth module-HC 05	It is a class-2 Bluetooth module with serial port profile, which can configure as either master or slave.	Each Switch Board
Cisco ISR 4000 series Router	Router to act as a fog gateway	One for the whole building

C. Software

The Arduino IDE is a cross-platform Java application that serves as a code editor and compiler and is also capable of transferring firmware serially to the board. We used this platform to program the ARDUINO's ATMEGA328 to control the building appliances. Using the android application user can view the current status (e.g., switch on/off) of the building devices and can also control these devices both locally or remotely. Android studio is used to build the application. In cloud server, we have tested our prototype with both the MySQL and NoSQL (MongoDB) databases. The reason behind moving to MongoDB was our requirement of handling all types of data (heterogeneous) like audio, video, etc. We have used OPENSHIFT for this purpose as it provides us PAAS which is very useful for this kind of applications [6].

Table 2. Description of sensors used in the smart building

Sensor	Description	Purpose
EM 18 / RC 522-AN / RFID	RFID reader module, to read 125KHz tags.	Entry, Security
LASER	A small device with a laser diode emitting a very narrow coherent low-powered laser beam of visible light.	Entry, Security
LDR	A photo resistor device whose resistivity is a function of the incident electromagnetic radiation. Hence, they are light sensitive.	Entry, Living room, Hall, Corridor, Lift
DHT 11	Basic and ultra low-cost digital temperature and humidity sensor.	Living Room, Hall
HC SR 501	A PIR sensor detects motion. It is used to detect whether a human has moved in or out of the sensor range	Security, Living room, Hall, Corridor, Lift
HC SR 04	An ultrasonic sensor uses sonar to determine distance to an object (from 2cm to 400 cm or 1" to 13 feet.).	Security, Garage, Pumping System, Corridor, Lift
MQ2/MQ6/ MQ7/MQ9	Gas sensor module, useful for gas leakage detection (in building and industry). It is suitable for detecting H2, LPG, CH4, CO, alcohol, smoke or propane.	Living room, Hall, Fire Safety
MQ135	Air pollution gas sensor is used in air quality control equipment for buildings/offices These are suitable for detecting of NH_3, NO_x, alcohol, Benzene, smoke, CO_2, etc.	Living room, Hall, Fire Safety
Flame Sensor	It can detect flame or wavelength of light source within 760nm~1100nm.	Fire Safety

III. PROTOTYPE FUNCTIONALITIES

This section presents the description of the prototype build of the smart building application. All hardware modules described in Table 1 are essential for all these scenarios. However, specialized sensor(s) will be required for a particular scenario. Sensors used in the following section are described in Table 2.

Entry - Every member will have a specific RFID card and whenever he/she will enter into the room, identification will be verified using EM 18/ RC522 module. For a verified user door will be unlocked automatically. In case of un-authenticated user, an alarm will be generated inside the house, such that people get notified about the fact. Along with this, sensors like LASER and LDR are used to make buildings safer. If someone crosses the laser, an automatic security alarm will start. To make the security system more strong, PIR motion sensor and ultrasonic ranging modules are used to detect any intruder.

Light, Fan and AC - Controlling light or fan or AC is very simple in nature, only a relay and a Bluetooth/ WiFi module are required to remotely switch on/off. To monitor the current room temperature, DHT 11 sensor is used. However, light controlling does not mean only on/off the light switch using smart phone but also with some intelligence. In addition to on/off the light using smart phone both locally or remotely, we have used a LDR sensor to measure the light intensity of the surroundings. When the surroundings are dark, IE, if it is lower than some predefined threshold, then it will be automatically on. It is mainly useful when implemented for the lights that are outside our house. Moreover, if there is any human movement detected using HC SR 501 sensor, it can switch on the light.

Garage - Three functionalities are given here: i) opening garage door ii) controlling light and iii) parking indicator. In case of door and light, we are using a relay and the WiFi/Bluetooth module according to the need. To handle the case of parking indicator, there are 2 parts. One is a rear parking sensor HC SR 04 and another one is a metal detector module. Rear parking sensor informs the driver by sensing car's distance from the back wall. Here the function of metal detector is to note if there is any car is in the garage or not, which is placed at the middle of the garage. So the remote user can easily check the availability of the parking in the garage. This approach can be extended to manage parking lots at the office or a shopping mall.

Water and Pumping System - The idea is to measure the distance of the water level's upper layer from the sensor attached point. Here, sensor HC SR 04 is used to measure the distance. The logic is that when the water level is below some pre specified value, sensor HC SR 04 will automatically trigger the pump switch and water will be filled in the tank. After reaching the maximum level (predefined), the switch will be off automatically. To stop wastage of water, sensor HC SR 04 is used to detect an entity in front of the basin and will automatically control water flow.

Air Purifier and Fire Safety - Air purifier is nowadays one of the essential commodity in our daily life. Air quality sensor MQ 135 is used to intelligently control the air purifier. Moreover, gas sensors like MQ 2, MQ 6, MQ7 and MQ 9 are used to sense the LPG and air quality. Exhaust can also connect to IoT. If the monitored LPG level is above some predefined threshold, the exhaust will be automatically on, which will remove the LPG from the kitchen and also start an alarm. These sensors are also used in different parts of a building to make it fire safe.

Corridor and Lift – In the case of the corridor and lift, the concept is that it can sense the human presence with the help of HC SR 04 and PIR sensor. Along with this, in corridor lights can be controlled with the help of LDR to enable the green building concept intelligently. Whereas, fan and light are unavoidable in the lift. Since, it is used all day long by the inhabitants of the building and also not in a fixed fashion. So, to reduce electricity wastage in the lift, we detect human presence in the lift and if it detects no person for a certain duration , then it automatically turns off the light and fan of the lift which saves energy.

IV. IMPLEMENTATION

Basic workflow of the proposed system is described here with the help of an interaction diagram as shown in Fig. 4. User at building can control the building devices through smart phone using a Bluetooth connection, while remote user will do this using WiFi or other internet connections

User at building can control the building devices through smart phone (i.e., `changeIoTStatus()` as shown in Fig. 4) using Bluetooth connection, while remote user will do this using WiFi or other internet connections. We maintained our server in the cloud to store and update the status of the devices that we are using in the smart building application. The state of the switch is reflected in the application using an event trigger mechanism implemented on the server side. The status of each device is stored in the cloud database as well as in the fog gateway and updated whenever there is any change (i.e., `changeFogStatus()` and `changeCloudStatus()` as shown in Fig. 4). For example, if the user changes the state of any device locally through the smart phone application, the information is pushed into the fog and then into the cloud with the single button click event. This change will also reflect on the Android application and PC interface (i.e., `updateUserInterface()` as shown in Fig. 4). Whenever there is a request to change the status of any IoT device remotely, that request is sent to the cloud server, the server forwards this request to the fog server which generates IoT status change request (i.e., `changeIoTStatus()as shown in Fig. 4`) for the respective IoT enabled devices. Similarly, when the status of the IoT enabled device is changed in reality, then it's changed status is forwarded to the fog, as well as in the cloud and also updated in the database and after that this change is reflected in the Android application.

For example, if the user changes the state of any device locally through the smart phone application, the information is pushed into the fog and then into the cloud with the single button click event. Whenever there is a request to change the status of any IoT device remotely, that request is sent to the cloud server, the server forwards this request to the fog server, which generates IoT status change request for the respective IoT enabled devices. Similarly, when the status of the IoT enabled device is changed in reality, then it's changed status is forwarded to the fog, as well as in the cloud and also updated in the database and after that this change is reflected in the Android application.

Here electrical device identification is a crucial part as in a building, there are multiple floors, in each floor there are multiple rooms and there are multiple electrical appliances. So, to solve this issue, we have used a single Arduino board for each room and given each board a unique ID. Now, To identify an electrical device uniquely we generate a `Thing_ID` which is composed of two parts, i.e, `Board_ID` and `Pin`. Here,the pin is the board pin of Arduino at which the appliance is attached via Relay. So, as a result, the user request is generated in tuple <Thing_ID, Request>.

V. EVALUATION

The proposed smart building system is evaluated from both locally and remotely. All the functionalities stated in this paper are tested using the prototype model implemented in our lab. The manual overwrite functionality has been exhaustively checked for different sensing conditions.

Both the MySQL and NoSQL database (MongoDB) are maintained in the cloud using OPENSHIFT platform. We have used PHP to connect and handle the server database. Snapshots of MySQL and MongoDB for smart building control are shown in Fig. 5 and Fig. 6 respectively.

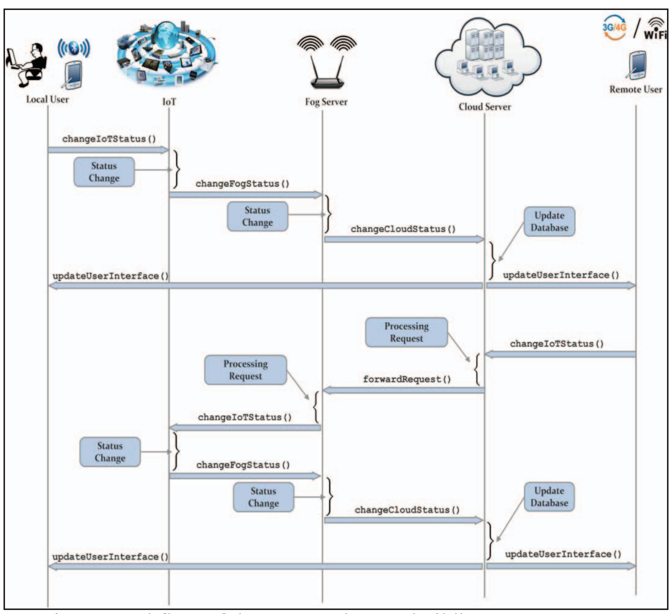

Fig. 4. Workflow of the proposed smart building system

In the proposed architecture the presence of fog has a very significant role. We have tested the prototype for a round trip time delay for communication purpose in both the cases of controlling building with and without fog. Actually, we found that there are some scenarios where we can avoid pushing data to the cloud and decrease the system's overall response time. Fog Server can reduce the internet traffic by handling many aspects locally using it's computation and storage power. Thus, it makes the system more agile and responsive (real time).

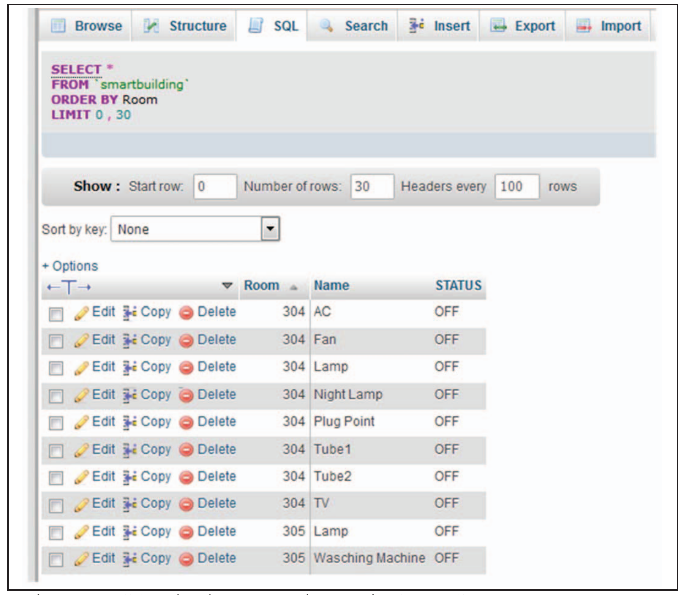

Fig. 5: MySQL database snapshot, as in OPENSHIFT

Fig. 6: MongoDB database snapshot, as in OPENSHIFT

However, for enhanced analytics and long term data storage, cloud is essential. Thus, the proposed IoT-fog-cloud is the ideal architecture as locally the user can control the appliances in real time (10ms approx, using IoT-fog in our lab setup) where as remotely the response time is slightly higher (1230 ms approx, using IoT-fog-cloud in our lab setup). These values are averages calculated on 20 different instances.

We have also monitored that because of building's inbuilt intelligence, it also saves electricity as it can detect human presence and stops unnecessary wastage of power and thus it confirms a green building. We have also considered user's security perspective, i.e., if the mobile is even lost, he will not lose control of his building or his room, the user need to install the app and need to change the password which will be verified with an Email. After confirming that, the user can change the apps password which is maintained in the server. Thus, it is safe to use also.

Fig. 7 shows the light control with the help of LDR and smartphone using a WiFi module that connects to the lab's WiFi. The same concept is used to connect the other devices of the building. Here, we notice that Arduino MKR 1000 is one of the best choice for implementing this sort of prototype where we need not to integrate the ESP8266 module to have the WIFI functionality with the microcontroller as the WIFI is inherently there in MKR 1000.

VI. CONCLUSION

The main purpose of this work is to develop a simple and low cost, smart green building using IoT, fog and cloud technologies. The detail design and prototype description of the proposed system are discussed here. This system work perfectly for both MySQL and NoSQL databases, but choice of the database depends on our need.

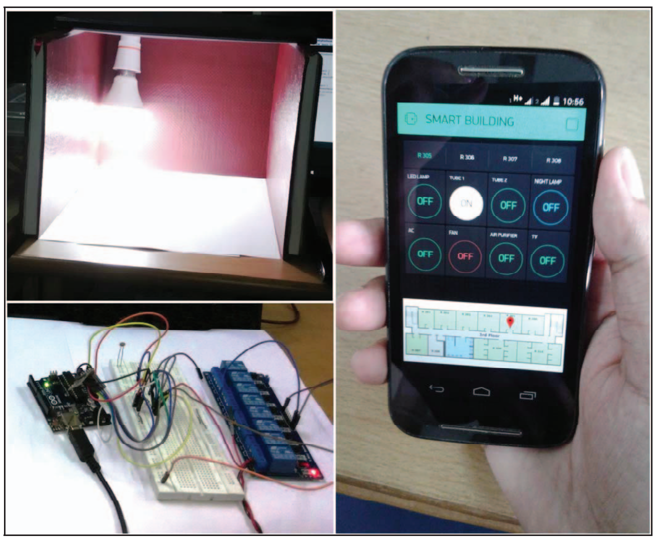

Fig. 7: Snapshot of light control in smart building

If we want to analyze the audio and video data (heterogeneous in nature) collected in the building along with the images taken in the CCTV, then we have to shift to NoSQL database like MongoDB as used in this paper, otherwise, we can use MySQL.So, basically when the data types are heterogeneous, to handle that various sensor generated data, we need MongoDB as it is capable of handling not only heterogeneous data but also the big data. A small prototype of the proposed architecture has been implemented in a lab environment to demonstrate the effectiveness.

VII. ACKNOWLEDGMENT

Research of the first author is funded by "Visvesvaraya PhD Scheme, Ministry of Communication & IT, Government of India".

REFERENCES

[1] H. Ghayvat, J. Liu, S. C. Mukhopadhyay and X. Gui, "Wellness Sensor Networks: A Proposal and Implementation for Smart Home for Assisted Living," in *IEEE Sensors Journal*, vol. 15, no. 12, pp. 7341-7348, Dec. 2015.

[2] Y. T. Lee, W. H. Hsiao, C. M. Huang and S. C. T. Chou, "An integrated cloud-based smart home management system with community hierarchy," in *IEEE Transactions on Consumer Electronics*, vol. 62, no. 1, pp. 1-9, February 2016.

[3] M. Li, and H. J. Lin, "Design and implementation of smart home control systems based on wireless sensor networks and power line communications," *Industrial Electronics, IEEE Trans. on*, vol. 62, no. 7, pp. 4430-4442, Jul. 2015.

[4] R. Teymourzadeh , S. A. A. Ahmed, K.W. Chan, and M.V. Hoong. "Smart GSM Based Home Automation System," in Systems, Process & Control (ICSPC2013), pp. 306-309, Dec. 2013.

[5] Arduino: Open-source electronic prototyping platform enabling users to create interactive electronic objects., URL: https://www.arduino.cc/

[6] Openshift: Red Hat's Platform-as-a-Service (PaaS) URL: https://www.openshift.com/

[7] IBM Bluemix, URL: https://developer.ibm.com/bluemix/2015/05/07/sogeti-high-tech-and-ibm/

[8] J. Dutta, F. Gazi, S. Roy, C. Chowdhury, "AirSense: Opportunistic Crowd-Sensing based Air Quality monitoring System for Smart City," in the proceedings of *the IEEE SENSORS 2016*, pp. 976-978, Oct. 2016.

[9] J. Dutta, C. Chowdhury, S. Roy, A.I. Middiya, F. Gazi, "Towards Smart City: Sensing Air Quality in City based on Opportunistic Crowd-sensing," International Conference on Distributed Computing and Networking (*ICDCN*) 2017, ACM, DOI: http://dx.doi.org/10.1145/3007748.3018286.

[10] B. K. Ray, S. Khatua and S. Roy, "Negotiation based service brokering using game theory," Applications and Innovations in Mobile Computing (AIMoC), 2014, Kolkata, 2014, pp. 1-8.