Internet of Things Cloud Framework for Smart Homes

RaafatAburukba, A. R. Al-Ali, Mohammed Rashid, and Rizwan Hassan

***Abstract*—Smart appliances and renewable energy resources are becoming an integral part of smart homes. Nowadays, home appliances are communicating with each other with home gateways, using existing short-range home area network communication protocols such as ZigBee, Bluetooth, RFID, and WiFi. A Gateway allows homeowners and utilities to communicate remotely with the appliances via long-range communication networks such as GPRS, WiMax, LTE, and power liner carrier. This paper utilizes the Internet of Things (IoT) concepts to monitor and control home appliances. Moreover, this paper proposes a framework that enables the integration and the coordination of Human-to-Appliance, Utility-to-Appliance, and Appliance-to-Appliance. Utilizing the concepts of Internet of Things leads to one standard communication protocols, TCP/IPV6, which overcomes the many diverse home area networks and neighborhood area networks protocols. This work proposes a cloud based framework that enables the IoTs integration and supports the coordination between devices, as well as with device-human interaction. A prototype is designed, implemented, and tested to validate the proposed solution.**

***Index Terms*—Internet of things (IoT), Internet of things (IoT) cloud framework, smart homes, smart appliances.**

1. Introduction

As the smart grid evolves, smart home environment is becoming a focus area for building a smart monitoring and control home appliances system as well as managing the integration of renewable and storage energy devices in residential areas.

Many home appliances monitor and control systems use short and long range communication protocols. Smart home appliances monitor and control systems utilizing ZigBee, Bluetooth,Radio-Frequency Identification (RFID), and WiFi communication protocols were reported in [1]-[4]. These systems use more than one short range home area networks communication protocols within the same house. Also, some smart home energy management system including renewable and storage energy devices were developed based on ZigBee and WiFi[5]-[7]. These systems are based on short range communication protocols as well.

On the other hand, long-range communication protocols are utilized to extend the monitor and control range further. For example,Worldwide Interoperability for microwave access (WiMax)[8], power line carrier (PLC)[5],[9], and general packet radio service (GPRS)[10],[11] are used to upload appliances and devices status, and download control commands from a faraway utility service or remote homeowner mobile handset. These systems used different communication protocols for long distance monitor and control.This hybrid and multiple communication protocols use more than one standard. Such multiple standards architecture introduces challenges and issues related to quality of service[12], cross technologies interference[13], frequency licenses[14], scalability[15], and cost[16]. Fig. 1 depicts the exiting smart home monitoring and control system with different communication protocols.

To overcome some of the above mentioned challenges and limitation, a one communication protocol for home and neighborhood area networks would be the optimum solution. Such protocol should be able to handle short and long range access to home appliances and renewable energy devices.

Nowadays, a communication protocol for home automation and energy management is becoming possible by utilizing the Internet of things (IoT) concept. Every home appliance, renewable and storage energy devices can be considered as objects (things).Also, homeowners' mobile handsets, as well as utility application servers can be considered as objects (things).

Each object can be assigned a unique IP address with the capability of receiving commands and transferring data over the internet. With such capabilities those objects become IoT. Once an object becomes an IoT appliance or device, it can exchange data and commands with other connected devices (things) such as smart meter, utility services and other devices i.e. machine to machine or

Manuscript received June 26, 2016; revised July 26, 2016. This work was supported in part by the Department of Computer Science and Engineering at the American University of Sharjah, UAE.

RaafatAburukba, A. R. Al-Ali (corresponding author), M. Rashid, and R. Hassan are with the Department of Computer Science and Engineering, American University of Sharjah, Sharjah, UAE (email: raburukba@aus.edu; aali@aus.edu; b00042500@aus.edu; b00042130@aus.edu).

Color versions of one or more of the figures in this paper are available online at http://www.journal.uestc.edu.cn.

Digital Object Identifier: 10.11989/JEST.1674-862X.606261

human to machine or utility server to machine. This architecture will eliminate the need for multiple short and long range communication protocols by utilizing the IPv6 over low power wireless personal area networks (6LowPAN) communication protocol[15]-[17]. The 6LowPAN is a limited edition of IPV6. It can have up to 65-75 bytes of payload, which is more than enough to handle smart home appliances and renewable and storage energy device operation[15],[17]. Utilizing the IPV6 increases the number of unique addresses from 2^{32} to 2^{128}[17] i.e. eliminating the scalability issue from the addressing point of view.

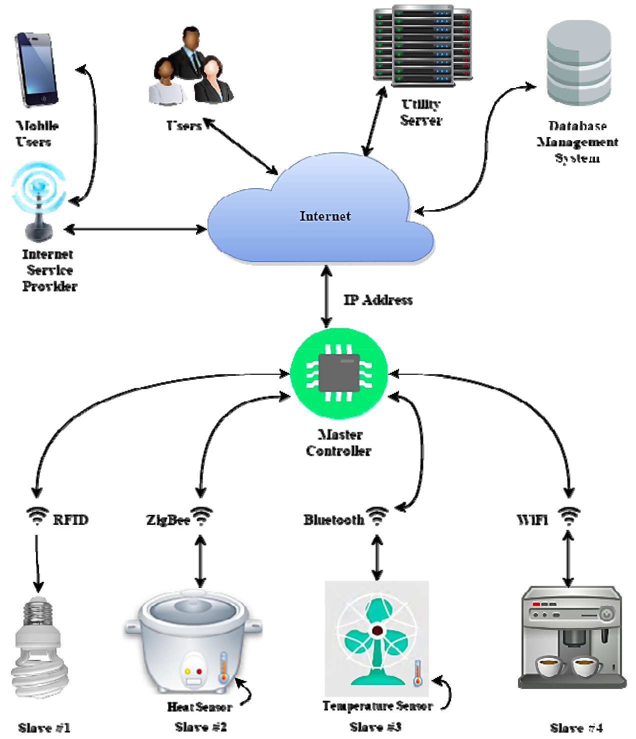


Fig. 1. Existing smart home systems with different communication protocols.

The smart home is distinguished by three main characteristics:

- The ability to sense and monitor changing conditions.
- Appliances and devices have the ability to communicate and interact with people, systems, and other objects.
- The ability to make quantitative decisions based on defined knowledge.

Some proposed frameworks in the literature aimed in enabling the coordination between devices in a smart home environment. The proposed framework in [18] enables smart home systems execution and coordination between different elements in the smart home. The approach uses the Simple Object Access Protocol (SOAP) technology that provides platform independent interoperation among heterogeneous systems. Moreover, another proposed rule-based framework in [19] to manage systems within the smart home environment. It is based on event-condition-action (ECA) rule mechanism with SOAP technology that provides interoperability among those systems. The proposed framework in the literature tackle the heterogeneity issue for IoT devices within a smart home environment, however, it is limited to a single home instant, no automated decision based on home owner preferences and criterions, and the manual configuration with tight coupling of specific devices in a smart home environment. Recall that each appliance in a smart home environment may have different functionalities and interfaces. Hence manual configuration within a framework is not feasible.

The objective of this work is to create a smart home cloud framework capable of handling the aforementioned characteristics of a smart home environment and enables the seamless integration of IoT devices/appliances within a smart home environment.

Deploying the proposed smart home framework to the cloudovercome the interoperability challenges of having diversestandards, operations, and interfaces in home devices. This is done by adopting a common interoperability framework for home devices and web-based services. Hence, devices communicate with the services in the cloud using web-based standards, regardless of the access networking technology or other standards used, and different services can interoperate in the cloud using web services standards. Fig. 2 shows a high-level view of the proposed system architecture using an IP communication protocol; each home appliance has a unique IP address.

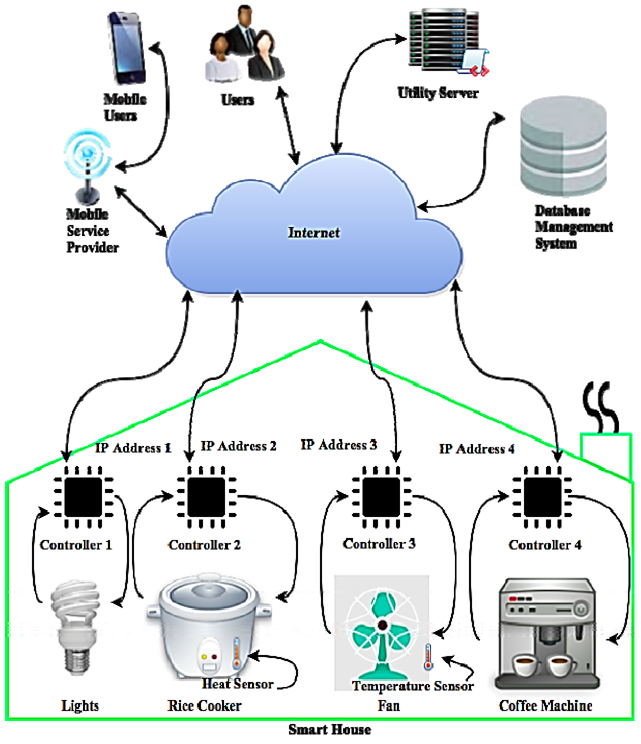


Fig. 2. High-level view of the proposed system architecture using an IP communication protocol; each home appliance has unique IP address.

2. Environment Analysis and Requirements

In a smart home environment, a home with m appliances can be denoted by (1).

$$A=\{A_1,A_2,\cdots,A_m\}. \tag{1}$$

An appliance is a physical device that is capable of executing tasks or requests to receive command and exchange data. For example, in a smart home environment an appliance can be a television, an air condition, and a washing machine. Each appliance can be defined by a set of capabilities in (2).

$$A_i=\left\{c_i^1,c_i^2,\cdots,c_i^{a_j}\right\} \tag{2}$$

where a_j is the number of capabilities that describes A_i. For example, capabilities for a washing machine can be rinse, normal wash, hand wash, water temperature, and spin speed. Each appliance in the smart home environment is placed in a specific home zone. A home zone defines specific areas inside the home. For example, master bedroom can be a zone. A home zone can be presented as

$$Z=\{Z_1,Z_2,\cdots,Z_q\}. \tag{3}$$

Let $A_{i,q}$ denote appliance i in zone q. Given a set of n tasks for a smart home environment, denoted in (4):

$$T=\{T_1,T_2,\cdots,T_n\}. \tag{4}$$

Task $T_j\in T \quad (j=1,\ldots,n)$ is formulated as a set of operations in (5):

$$T_j=\left\{o_j^1,o_j^2,\cdots,o_j^{t_\partial}\right\} \tag{5}$$

t_∂ is the number of operations belonging to T_j. For example, task 1 can be defined as a request to the washer machine, with the operations normal wash and spin speed 800 rpm.

The objective of this work is to create a framework that enables the integration, monitoring, and both automatic and manual control of appliances in a smart home environment.

The main characteristic of a smart home environment is ability to take decision. The decision is based on the allocation of tasks or requests to the capable appliances/devices at a specific time. A task or a request may be processed by a single appliance A_i or a group of appliances in A, in which the appliances are capable to process the task T_j. For example, a request in a smart home environment can be cooling a specific area in the home to 19 °C by 5 pm, where the home has three A/C units. The system can allocate the request to the different A/C units at home to achieve this request.

The functional requirements to enable the smart home framework are as follows:

An IoT Appliance/Device Requirements:

- An IoT shall register its capabilities.
- An IoT shall request events. An event request is a task to be executed by a capable IoT at specific time.
- An IoTshall accept events (tasks).
- An IoT shall execute requested events (tasks).
- Each IoT must have a unique address.

Controller Requirements:

Each IoT has off-the-shelf credit card-sized single-board computer with the following built-in resources[8],[20]:

- System on chip CPU.
- SDRAM.
- GPIO (general proposes digital input/output ports)
- BaseT Ethernet socket.
- Two USB connector and display serial interface analog to digital converter (ADC) adapter board.

The above communications, digital and analog input/outputs ports are utilized in the proposed system as follows:

- The digital inputs are used to enable homeowners to operate the appliance locally.
- The digital outputs are utilized to drive the home appliances through solid states relays to overcome the voltage and current limitation that are provided by the board ports[16].
- The USB port is used to connect the controller to a PC to upload and download developed programs.
- The ADC channels to read the analog metrics that home appliances may have such as temperature, humidity, weight, and pressure.
- 10/100 BaseT Ethernet port to connect the system to the Internet which makes it an IoT thing that can be accessed through the World Wide Web.

Framework Requirements:

- The framework provides a unique identification to homes connected to it.
- The framework shall provide a unique identification to the smart home appliances and devices.
- The framework associates appliances to a home.
- The framework manages appliances within a home.
- The framework manages users and roles of users within a home.
- The framework allows users to define roles and accessibility.
- The framework allows appliances to register with its capabilities.

- The framework allows appliances to register its web service definition.
- The framework enables users to define their tasks and operations.
- The framework enables manual task execution.
- The framework enables an automated task execution based on the user defined events.
- The framework should provide a monitoring service for specific defined events by IoT.
- The framework allows users to define rules and objectives for the smart home to govern the behavior of the automated decision making.
- The framework enables the smartness of the home by utilizing the defined home rules and objectives, and takes automated decision.

3. Proposed Architecture

This section proposes a smart home framework service architecture that is deployed as a Software as a Service (SaaS) and uses the cloud Platform as a Service (PaaS). The proposed framework serves as a general architecture for any IoT within a smart home environment and uses web services to communicate with the smart home framework. The proposed framework provides a solution to the aforementioned functional requirements in the previous section. The proposed smart home framework as shown in Fig. 3 contains two main layers: 1) Data Access and Integration Layer (DAI), 2) Decision Making Layer. Those two layers are designed to maintain loosely coupled components within the framework. Given the nature of the environment being distributed, security becomes a challenge. Security in the proposed architecture in Fig. 3 must be presented across all layers. This means that the things, smart home area network framework, the PaaS, and the IaaS, must adhere to the security objectives and governed by the security mechanisms. The topic of security is beyond the scope of this paper, and studied in the literature such as in [21] and [22].

3.1 Integration Layer

This layer aims on providing data integration for the IoTs within the smart home environment. It also provides data accessibility to the different services within the smart home framework. This layer contains the following services:

Registration Service:

This service enables devices to register with the proposed smart home framework service. The registration service has the following modules:

- Device registration: The main functionality of this module is to handle registration requests from devices. It takes in the device name, identification, capabilities, and service definition for each capability, and assigns a global unique identification to the registered device.
- Schema registration: Each device can produce or consume data based on specific data representation. This module enables a device to register its schema in the smart home framework service.
- Event registration: Based on the device operation, it can produce specific events. Event registration enables a device to register the events that it can produce. Also, it allows users or other devices to register to consumer specific events from a device.
- Register data store: Database repository is registered for the device.
- Unregister: This module enables a device, schemas, data stores, and events to be unregistered from the smart home framework service.

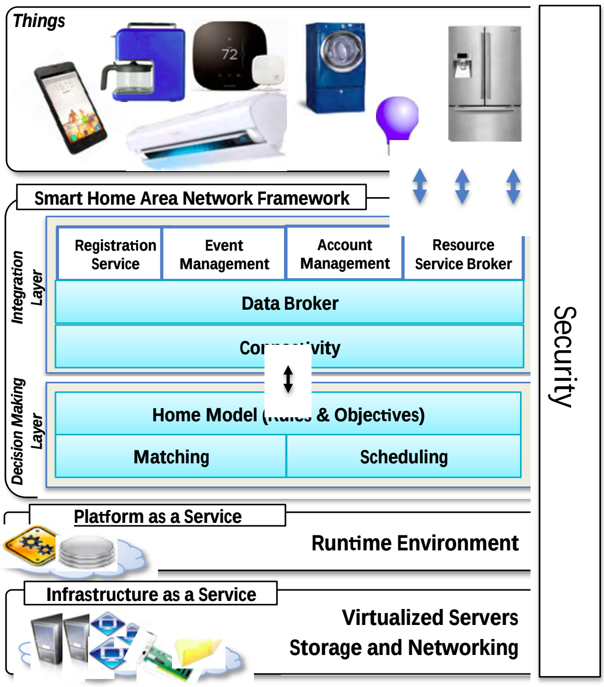


Fig. 3. Proposed smart home framework architecture.

Account Management:

This service deals with the creation of user accounts. It delivers the following main functionalities:

- Handles the operations to create, update, and delete user accounts.
- Verifies and validates the registered devices belong to the user.
- Links devices to user: associates the created users with the registered IoT appliances/devices. Users can be Smart home owner or verified user that is given access to devices and appliances within a smart home environment.

Event Management Service:

This service manages the distribution of events produced and consumed. This service has the following modules:

- Event listing: Provides a listing of events currently available for subscription.
- Retrieve event definition: Provides the details required to register for an event.
- Event trigger: Deals with the set of actions to be executed when an event is triggered.
- Event monitor: Monitors the occurrence of an event on a device.
- Event coordinator: Deals with the coordination between the event producer and consumer when an event is triggered to execute the defined set of actions.

IoT Resource Service Broker:

The resource service broker enables the capability based integration of IoTs with specific capability requests. The resource service broker has the following modules:

- IoT discovery: This module discovers specific IoT service based on its identifications and capabilities.
- Retrieve service definition: Gets the service definition provided by an IoT.
- Associate event to service: Links specific event to a defined service.
- Associate service to service: Links a specific defined IoT service to another IoT service. This enables specific dependencies of requests to be executed within a single device or multiple devices.
- Invoke service: Determines the address to connect to the service. This component connects and invokes the service on the IoT device. This also provides an orchestration between services if services are associated with the dependency relationship.

IoT Data Broker:

IoT data broker service enables devices in the IoT to persist its data. The following components enable persisting the IoT device data within the cloud infrastructure, as well as with data accessibility, extraction, loading, and transformation.

- Data access: This module enables data accessibility for an IoT device. It provides the insert, delete, and update operations to devices and other components within the smart home framework.
- Data extraction: Determines the source and extracts the data for the IoT device.
- Data loading: Sends the data to the appropriate destination.
- Data transformation: Transforms the data format from the stored data source to the data consumer destination format.
- Data monitoring: Monitors specific data for changes. This component is consumed by the event to monitor specific data changes related to an IoT device.

3.2 Decision Making Layer

This layer contains the services that allow home owners to model and manage their home preferences, rules, and objectives that govern the decision making behavior within the system. This layer contains the following services:

Home Model

The home model enables the definition and management of rules and objectives within homes. The Home Model service contains three main modules:

- Meta-model: this includes the structure to define the smart home rules and objectives.
- Model instance: uses the meta-model to enable the user to define rules and objectives for the smart home;
- Model management system: enables the user to manage (add, update, delete) rules and objectives.

Objectives can be modeled as an optimization model[23] to represent a quantitative model to the smart home. The optimization model contains two main elements: objectives and constraints. Objectives are decision criteria which can be modelled to maximize or/and minimize some function. Constraints are set of conditions which must be satisfied. Constraints may be operation-based, task-based, appliance-based, location-based or a combination.

Rules on the other hand, describe a rule as a statement that defines or constraints some aspect of the home. A home rule is intended to assert home structure or to control or influence the smart home behavior. A rule engine evaluates and executes rules, which can be expressed as if-then statements. The power of home rules lies in their ability both to separate knowledge from its implementation logic and to be changed without changing source code. A rule is composed of two parts, a condition and an action. When the condition is met, the action is executed[19],[23].

The concept of home rules captures precise logic in home appliances, tasks/requests, and time to govern the behavior of the decision making within the smart home. Six home rule types are proposed:

- Location rules are related to the home location zones such as living room and bed room zones. Each appliance and task has home rules associated with it. Assigning rules to both tasks and appliances will govern the matching and scheduling components to find the right location for the task.
- Appliance rules are related to the home appliance within a home and its capability. Each appliance is associated with appliances' rules. This association constrains the feasibility space with any dependency relationship among appliances.
- Tasks rules are related to an operation or sequence of operations to be executed in appliances or devices. Each task is associated with tasks' rules. This association constrainsthe feasibility space for the defined tasks sequence or order.
- Time rules are related to time such as morning time. Time intervals and tasks are associated with time rules. This association constraint the feasibility of the solution to a defined time constrains.

- Location-appliance rule associates appliances to specific defined location rule. This association allows the definition of an appliance location within a smart home environment.
- Task-time rule associates a defined task rule to a time rule. This association constrains the feasibility of the solution space with respect to the tasks' execution to a specific time.

Matching

This service contains a matchmaking algorithm that utilizes the defined home rules modeled within the Home Model service and finds the feasible scheduling space to the tasks/requests within the smart home.

Scheduling

This service contains an optimization algorithm that can be based on either exact or heuristic techniques[23]. The scheduling service utilizes the output from the matching of rules which contain the possible feasible space and finds the optimal solution (if exact optimization algorithm used) based on the defined optimization model defined within the Home Model service. The outcome of this schedule which contains the allocation of the request to an IoT at specific time is sent to the Invoke service within the IoT service broker in the integration layer.

4. Implementation and Testing

For the purpose of validation, a prototype was developed to mimic the proposed framework and deployed on a cloud platform provider. Fig. 4 shows the prototype setup that is implemented for testing and validation purpose. A light, fan, rice cooker, and coffee maker were each interfaced with its own microcontroller with a unique IP address. Since the microcontroller cannot supply the required power for the appliances, signal conditioning circuits in the input and driving circuits in the output were designed to satisfy the power requirements. The system can be operated manually by the home owner and can be accessed remotely via the IP address for each device. As described in the previous sections, each device status is read by its own microcontroller via its digital and analog inputs, process the data, enable or disable devices, based on the user requests whither it is local or remote requests. Locally, users can monitor and operate their home appliances by using the normal standard switches. Remotely, authorized users and utilities can access each individual appliance via its own IP address to monitor and control the appliances accordingly.

The microcontroller is configured to execute a script on boot up to handle operations of the device such as turning ON/OFF the lights, controlling intensity and choosing between remote and local control. The script is divided into two parts, namely, local control and remote + local control part. The local control runs only when there are some error in connecting to the cloud or if the Internet connection is not available.

The script reads data from the device configuration file in the microcontroller and interfaces with the smart home framework discovery service. The microcontroller initiates a device record checkup via registration service provided by the smart home framework. If no record exists for the device, the service creates a record using the data sent by the device such as manufacturer name, product ID, and user-pairing code. The user-pairing code is used to pair the device with the user. It is used as a one-time device-to-user pairing mechanism. Once the record for the device is created, the device and the smart home framework in the cloud exchange information. For instance, a light bulb exchanges information such as bulb status (ON/OFF) and intensity of the light. The microcontroller starts checking if any device operation change was committed by user and reflects the change via services provided by the smart home framework. Additionally, the microcontroller can be configured to operate automatically based on time or other parameters such as the room temperature to operate the air conditioner appliance accordingly or turning the lights ON/OFF. If the device has no Internet connectivity, the microcontroller switches to local mode operation with switch inputs and checks for Internet connectivity every one minute.

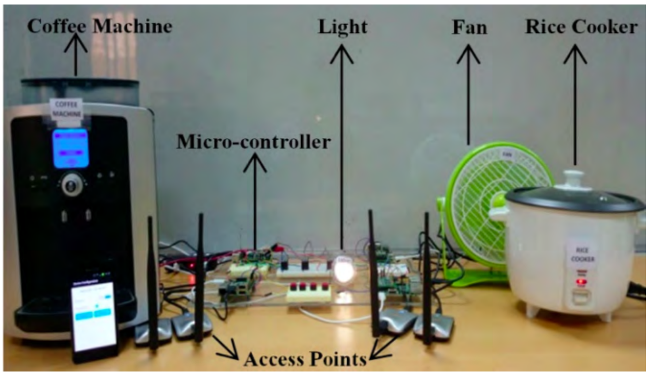
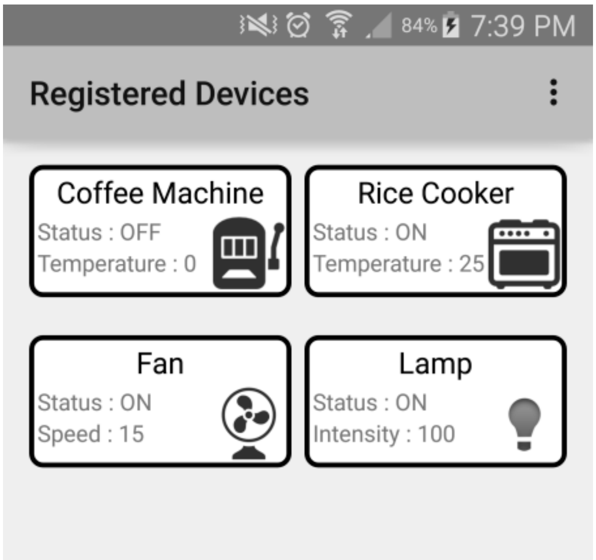


Fig. 4. High-level view of the implementation architecture.

Any microcontroller connects to the cloud smart home framework services via wireless or wired Ethernet technology. The appliances can be monitored and controlled via any platform by calling services offered by the smart home framework. For demonstration of using the smart home framework services, a mobile application is implemented. In the application, the user has to register and then he/she must input his newly created account username and password to sign in. Once logged in, the user management service authenticates the user through pulling the user credentials through the data access and integration components. Once the authentication is successful, the smart home framework sends success message to the mobile application along with the information of all devices associated with the user. The user can interact with mobile application to remotely control the operation of devices through the smart home cloud framework service. The user can also pair devices under his/her name using the pairing code of the device. Once pairing is successful, users can access appliances registered under his/her name. As shown in Fig. 5, the application lists all the appliances connected along with their status.



Registered Devices

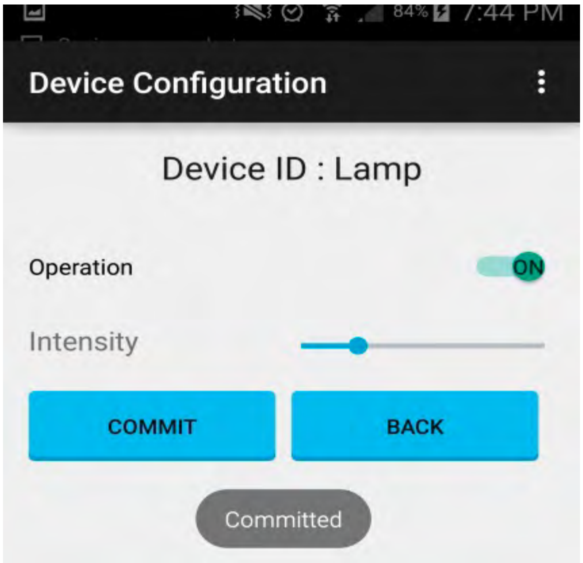
Coffee Machine
Status : OFF
Temperature : 0

Rice Cooker
Status : ON
Temperature : 25

Fan
Status : ON
Speed : 15

Lamp
Status : ON
Intensity : 100

Fig. 5. Device registered.



Device Configuration

Device ID : Lamp

Operation ON

Intensity

Fig. 6. Lamp configuration.

When the user taps on the desired appliance to change its settings, it opens a new window that allows the user to control the settings for that particular appliance. After the user is done with the settings, then he/she can commit the changes to the smart home framework via a tap on the onscreen button. Once the user commit a change, the application sends the data to the IoT cloud framework, which processes the information and reflects the change in the database and sends the commands to the device. For example, after the user selects to control the light intensity from the registered device window as shown in Fig. 5, the application executes the request accordingly as shown in Fig. 6 and allows the user to switch the light ON/OFF or to dim it.

5. Conclusions

In this work, an Internet-based smart home monitoring and control system was presented. Each home appliance was assigned a unique address that is used to upload device status and download command from users. This allows the IoTs to be represented as a set of objects that are uniquely identifiable as part of the Internet. When these objects connect with one another over the Internet, they provide many types of services and produce massive amount of data and information. For this reason, a smart home cloud framework service was proposed and developed using a cloud Platform as a Service provider. The testing result showed the integration of the IoT and the smart home appliances by using the smart home framework. The major contribution of this work is as follows:

- A plug-and-play feature for IoT in smart homes to enable the coordination between appliances, devices, system, and human.
- Utilizing the cloud computing paradigm to overcome the interoperability challenges of having diverse standards in home devices. This is done by adopting a common web service standard.
- The smart home framework as a service is scalable. Instances of smart homes can be integrated and can be monitored and controlled using the same framework services.
- The decision making layer that enables the home to be "smart" by making decision on behalf of home owners based on defined rules and objectives.

The system concept can be implemented to serve in a smart micro grid that supplies residential neighborhood housing.

Security is a major challenge given the nature of the domain being distributed. Future studies will be done on security for the proposed architecture. Also, an expansion will be studied for the home model service in the decision layer to include the design of the meta-model to enable users to model instances of smart home. Moreover, optimization models and algorithms will be explored and evaluated as part of the future expansion of the proposed smart home framework.

Acknowledgment

The authors thank the Computer Science and Engineering Department, the American University of Sharjah, UAE, for providing the support and resources.

References

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

[2] S. Kuma, and S. Lee, "Android based smart home system with control via Bluetooth and internet connectivity," in *Proc. of the 18th IEEE Intl. Symposium on Consumer Electronics*, 2014, pp. 1-2.

[3] N. K. Suryadevara, S. C. Mukhopadhyay, S. D. T. Kelly, and S. P. S. Gill, "WSN-based smart sensors and actuator for power management in intelligent buildings," *IEEE/ASME Trans. On Mechatronics*, vol. 20, no. 2, pp. 564-571, April 2015.

[4] N. Komninos, E. Philippou, and A. Pitsillides, "Survey in smart grid and smart home security: issues, challenges and countermeasures," *IEEE Communication Surveys & Tutorials*, vol. 16, no. 4, pp. 1933-1954, 2014.

[5] J. Han, C. Choi, W. Park, I. Lee, and S. Kim, "Smart home energy management system including renewable energy based on ZigBee and PLC," *IEEE Trans. on Consumer Electronics*, vol. 60, no. 2, pp. 198-202, May 2014.

[6] D. Han and J. Lim, "Smart home energy management system using IEEE 802.15.4 and ZigBee," *IEEE Trans. on Consumer Electronics*, vol. 56, no. 3, pp. 1403-1409, Aug. 2010.

[7] K. Gill, S. Yang, F. Yao, and X. Lu, "A ZigBee-based home automation system," *IEEE Trans. on Consumer Electronics*, vol. 55, no. 2, pp. 422-430, May 2009.

[8] M. Kuzlu, M. Pipattanasomporn, and S. Rahman, "Communication network requirements for major smart grid applications in HAN, NAN and WAN," *Computer Networks*, vol. 67, pp. 74-88, Jul. 2014.

[9] J. Han, C. Choi, W. Park, I. Lee, and S. Kim, "PLC-based photovoltaic system management for smart home energy management system," *IEEE Trans. on Consumer Electronics*, vol. 60, no. 2, pp. 184-189, May 2014.

[10] A. R. Al-Ali, I. A. Zualkernan, A. Lasfer, A. Chreide, and H. Abu Ouda, "GPRS-based distributed home-monitoring using internet-based geographical information system," *IEEE Trans. on Consumer Electronics*, vol. 57, no. 4, pp. 1688-1694, Nov. 2011.

[11] H. Jing and W. Zhang, "Design of remote intelligent home system based on ZigBee and GPRS technology," in *Proc. of the 2nd Intl. Conf. on Consumer Electronics, Communications and Networks*, 2012, pp. 264-267.

[12] P. Yang , Y. Yan, X. Li, Y. Zhang, Y. Tao, and L. You, "Taming cross-technology interference for WiFi and ZigBee coexistence networks mobile computing," *IEEE Trans. on Mobile Computing*, vol. PP, no. 99, pp. 1-14, Jun. 2015.

[13] E. Doulatyari, P. A. Kohan, and A. Esmailpour, "Quality of service support for a WiFi-WiMAX network in a test-bed environment," in *Proc. of 2012 Intl. Conf. on Innovations in Information Technology*, pp. 316-321, 2012.

[14] J. Vella, and S. Zammit, "A survey of multicasting over wireless access networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 718-753, 2013.

[15] T. Gonnot and J.Saniie, "User defined interactions between devices on a 6LoWPAN network for home automation," in *Proc. of IEEE Intl. of Technology Management Conf.*, 2014, pp. 1-4.

[16] A. Mahmood, N. Javaid, and S. Razzaq, "A review of wireless communications for smart grid," *Renewable and Sustainable Energy Reviews*, vol. 41, pp. 248-260, Jan. 2015.

[17] Z.-C. Huang and F. Yuan, "Implementation of 6LoWPAN and its application in smart lighting," *Journal of Computer and Communications*, vol. 3, no. 3, pp. 80-85, 2015.

[18] T. Perumal, A. R. Ramli, and C. Leong, "Interoperability framework for smart home systems," *IEEE Trans. on Consumer Electronics*, vol. 57, no. 4, pp. 1607-1611, Nov. 2011.

[19] C. Leong, A. R. Ramli, and T. Perumal, "A rule-based framework for heterogeneous subsystems management in smart home environment," *IEEE Trans. on Consumer Electronics*, vol. 55, no. 3, pp. 1208-1213, Aug. 2009.

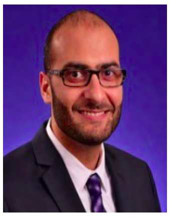
[20] C. Bysani, T. S. R. K. Prasad, and S. Chundi, "Raspberry Pi for Commercial Applications," *Intl. Journal of Computers and Technology*, vol. 11, no. 2. pp. 2250-2255, Oct. 2013.

[21] M. Ali, S. U. Khan, and A. V. Vasilakos, "Security in cloud computing: Opportunities and challenges," *Information Sciences*, vol. 305, pp. 357-383, Jun. 2015.

[22] V. Chang, Y. Kuo, and M. Ramachandran, "Cloud computing adoption framework: A security framework for business clouds," *Future Generation Computer Systems*, vol. 57, pp. 24-41, Apr. 2016.

[23] S. M. Huynh, D. Parry, A. C. M. Fong, and J. Tang, "Novel RFID and Ontology based Home Localization System for Misplaced Object," *IEEE Trans. on Consumer Electronics*, vol. 60, no. 3, pp. 402-410, Aug. 2014.

[24] R. L. Rardin, Optimization in Operations Research, 1st ed. *Prentice Hall*, 1997, pp. 25-287.



RaafatAburukba received his bachelor degree in computer science with software engineering, master and Ph.D. degrees in software engineering, all from Western University, London Ontario, Canada. He is currently an assistant professor at the American University of Sharjah (AUS), UAE. Prior to joining AUS, he was with the Pennsylvania State University, USA, and Erie and Western University, Canada. He was also a senior software engineer and a research assistant at EK3 Technologies in London Ontario, Canada. His research interest include various topics in cloud computing, cooperation and coordination in distributed systems, economic-based models and approaches for decentralized scheduling, and applications to cloud computing, IoT, and smart spaces.

A. R. Al-Ali received his Ph.D. degree in electrical engineering and a minor in computer science from Vanderbilt University, Nashville, TN, USA in 1990; the master degree from Polytechnic Institute of New York, USA in 1986, and the B.Sc. in electrical engineering from Aleppo University, Syria in 1979. From 1991 to 2000, he worked as an associate /assistant professor with the King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia. Since 2000 and till now, he has been a professor in computer science and engineering with the AUS, UAE. His research interest is on embedded systems applications in smart grid and home automations. He published more than 100 journal and conference papers.

Mohammed Rashid is a research assistant at the Computer Science and Engineering Department, AUS. His research interest is on embedded systems applications, Internet of things applications in home automation and smart energy, and mobile Apps. development.

Rizwan Hassan is a research assistant at the Computer Science and Engineering Department, AUS. His research interest is on embedded systems applications in industry.