An IoT Based Smart Thermostat

Levent Özgür
International Computer Institute, Ege University
Bornova, 35100 Izmir, Turkey
e-mail: leventozgur@yandex.com

Moharram Challenger
International Computer Institute, Ege University
Bornova, 35100 Izmir, Turkey
e-mail: moharram.challenger@ege.edu.tr

Vahid Khalilpour Akram
International Computer Institute, Ege University
Bornova, 35100 Izmir, Turkey
e-mail: vahid.akram@ege.edu.tr

Orhan Dağdeviren
International Computer Institute, Ege University
Bornova, 35100 Izmir, Turkey
e-mail: orhan.dagdeviren@ege.edu.tr

Abstract—**The heating system and related thermostat is an important part of any home. As part of a solution for an intelligent home, the thermostat needs to behave in a way that controls the heating system to save the energy by providing automatic adjustments. In this paper, an IoT based smart thermostat is designed and implemented. The nodes in the system consist of DHT22 temperature sensors connected to Arduinos and communicating with each other through nRF24L01 modules. The sink node uses an ESP8266 to the Internet. The system can interact with authorized family members through a mobile application by sending push notifications. They also can adjust the home heating settings via this application. In this way, the system can intelligently turn the heating system on or off considering the availability of people at home. In addition, the system can adjust the rooms' temperature automatically and save the energy.**

Keywords-IoT; smart thermostat; arduino

I. INTRODUCTION

The number of devices connected to the Internet is increasing day by day. These devices increase the quality of life with Internet of Things (IoT) based applications. Each of these devices is expected to be connected to the internet and/or other devices under IoT. For example, in the scope of smart homes, these systems aim to enable the users to view the information of the home and control it from anywhere. According to Gartner research, 20 billion things (by the end of 2020) is expected to be connected to the internet of the device. In 2017, the number of connected devices used are 8.4 billion, [1].

The IoT has begun to be applied to many areas and one of these areas is smart homes and heating systems. The working principle of today's heating systems is almost the same. There are thermostats in each heating system which are used as a controlling tools that can keep the temperature value at a desired level. The change in temperature enables a sensitive part of the thermostat to act on a heating or cooling system by sending an electrical or pressure signal. The thermostat can be set to a specific temperature by the user. When the water is heated, it is moved to the room and heat of the transferred water increases the room temperature. The water in the system is stopped heating when the room temperature reached to the desired value. The hot water sent to the radiators cool down to under the desired value after some time and the boiler starts working again until the water temperature reaches the desired value and then stops again. In this operating principle, the temperature of the living areas depends on the temperature of the water circulating in the system and adjusted via the boiler thermostat. There are many studies about smart thermostats but most of them measure temperature from a single place.

Instead, we can place a thermostat in each room of the house and send the temperature value of each room to the server separately. According to the received temperature values, average temperature value is calculated, and unbalanced heating is prevented. The smart thermostat can also control sensors from web or mobile applications effectively. In addition, it is possible to create a schedule of energy consumption for the thermostats which optimize the energy consumption in smart homes.

This paper is organized as follows: In Section II, the related work is reported. In Section III the application domain has been explained. The system analysis and design are presented in Section IV and V, respectively. Section VI includes the use case implementation and Section VII covers its demonstrations. Finally, the conclusion has been drawn in Section VIII.

II. RELATED WORK

IoT technology is being more popular day by day. it started to be applied to many areas such as smart city and smart home. In the study of [2], authors describe an architectural model to facilitate tasks with IoT applications. The main components of the architecture are explained with the application in smart cities.

In the study of [1] the application of smart homes is argued along with the solutions for communication problems caused by different devices with different communication standards and technologies. Generally, ZigBee modules are used in IoT-Smart home systems because it provides a high bandwidth to use Wi-Fi.

The study in [3] discusses about collecting data from a room with sensors to provide indoor comfort and air quality. To collect data, a room was monitored by a wireless device

(SMD SHT 11), Internet connected sensor technology, and IoT gateway C4EBOX. Temperature and humidity was recorded in every 15 minutes using these sensors. The sensors were located based on some standards such as ISO 7726:1998 and ASHRAE 55:2010. Outdoor data is collected from the weather station of the Basque Government. When the indoor and outdoor temperature data is collected, it is analyzed to show the user's indoor comfort and air quality preferences.

In [4], an application of smart home is studied considering connectivity challenges. In this article, various standards such as RFID, Near-Field Communication, and Wi-Fi, are evaluated as the commonly used technologies. Also, in this article, the connectivity challenges (Interoperability, Self-Management and Maintainability, Signaling, Bandwidth, and Power consumption) are compared between the standards to show which ones can be used to handle these challenges.

In [5], a two-step framework is described for multiple hardware and software platforms and interoperate them effectively. It also proposes a 4-layer system architecture for large-scale applications of monitoring temperature. At layer 1, data is collected with traditional WSN. At layer 2, the WSN uses a micro controller to transmit the data. At layer 3, a cloud solution over a static server is recommended for data storage, handling data analytics and notifications. At layer 4, a mobile app is suggested to make the delivery more accessible. In addition, three significant design decisions for Layer 1 such as network topology selection, RF module selection, receiving RF module, and microprocessor selection are discussed.

In [6], two approaches for activity detection in smart home environment are presented. The activities include entrances to a room and exits from a room. They can be used in applications which monitor the wellbeing of people such as AAL (Ambient Assisted Living applications), HVAC (heating, ventilation and air conditioning), and regulating the temperature of the room, regulation of lighting systems, etc.

Finally, there are some other studies [7-9] which use a similar communication mechanism with our study but in other domains such as fire detection system.

III. Application Domain: Smart Home

Nowadays, most of the people prefer Combies for home heating but it does not provide a very good comfort as it is not controlled remotely, it provides unbalanced heating, and it increases energy consumption. Generally, thermostats stay in a room and Combi works according to the temperature of this room. Therefore, if the room is warmed up, the boiler will be turned off before the other rooms are warmed up which leads to unbalanced heating. Moreover, it can be forgotten to close the Combi when leaving the home. If the Combi is working when you are not at home, energy consumption will be increased. Furthermore, the Combi boilers are not controlled remotely so, there is no way to close it if you forget to turn it off when leaving the home; or if you want to turn it on to heat the home before you come home.

Smart homes are managed through sensors and assistive devices based on IoT applications. For this reason, it is expected that smart homes' temperature level can be managed in a more facilitated way. Most of the manufacturers produce an external smart thermostat for the existing Combi boilers. However, these thermostats only measure the temperature of the room where the temperature sensor is located. And then, all rooms' temperatures are set according to this information. So, temperature is unstable in the home.

In our system, several temperature sensors are distributed in each room (or different rooms) of the home and an average temperature is calculated. In this way, the smart thermostat can decide on the temperature of the Combi boiler. Each temperature node communicates with smart heat system with a nRF24L01 module and just root temperature node is connected to internet with a ESP8266 module. In this way, the average temperature of the home can be calculated and the smart thermostat can determine the temperature of the boiler according to the average temperature of the rooms. This reduces the total energy consumption of the system, and since the modules only communicate within themselves, the security is a bit more enhanced compared to other Internet based systems.

The user can easily change the temperature of the home with the mobile application. Moreover, the smart thermostat can automatically stop the Combi boiler when the user gets away from home, or it can restart when the user gets close to the home.

IV. System Analysis

An Android mobile application (as part of this system), is developed. The user can easily change the home temperature via the Android mobile app. With this application, the user's position is periodically checked so that the smart thermostat can automatically stop the boiler when the user is away from home, or can be restarted when the user goes close to home. When the boiler system is automatically turned on or off by the system, a push notification is sent to the user via Firebase.

ZigBee is usually used for communication in such systems. However, using ZigBee the total cost of the system will increases because more than one temperature sensors are used. According to comparison Table I, the nRF24L01 module has lower energy consumption and overall it costs less than ZigBee. Therefore, nRF24L01 are used in this study.

In this work, ESP8266 modules are used and web services are developed to communicate with mobile devices. The web services are developed on .Net Core with MVC framework. .NET Core is a module of .NET platform that provides an open platform and multi-platform support for developing web applications, web services, libraries and console applications. The reasons for using this platform are that it contains thousands of libraries suitable for every kind of solution, allowing applications to be developed more easily and quickly thanks to these libraries, support for multiple languages and multi-platform support so that applications can be run on many operating systems.

IIS is used to publish the developed web services on the Internet. In this work, MSSQL server is used to store and access the data. Figure 1 shows the system overview.

TABLE I. XBEE AND NRF24L01 COMPARISON

Model	XBee 802.15.4	nRF24L01
Transmitting consumption	45 mA	11.3 mA
Receiving consumption	50 mA	13.5 mA
Distance	90m	250m
RF data rate	250KBps	250KBps, 1MBps, 2MBps
Price	$29	$1,5

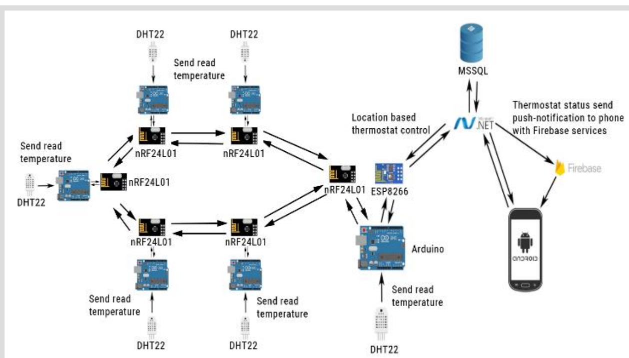

Figure 1. System overview

V. SYSTEM DESIGN

Smart thermostat proposed in this study has auto location option. If this option is close, the boiler works according to the commands entered by the user. But, if this option is open, the user's location is periodically checked by the system and the user's distance from the home is calculated. If the user is away from home, the Combi boiler is automatically shut down and then the user is notified via the mobile application. If the user approaches home, the system is started and as the first step the average temperature of home is calculated. According to this average value, if the temperature is lower than the user temperature, the Combi boiler is opened, the heating operation is started, and the user is notified via the mobile application. However, if the average value is higher than the desired temperature, the heating is stopped until the temperature falls. These operations are repeated continuously as long as the automatic position option is turned on, see Figure 2.

According to the system class diagrams, Figure 3, the user class has ID, name, E-mail, password attributes. E-mail and password are used to verify the users who want to login to the system. Each member is assigned a specific ID. User class has also two function members. These functions are used for authorization. The home class only holds the name of the home and the id assigned to the home. It also includes “OpenCombi”, “CloseCombi”, “startHeating" and “StopHeating” functions.

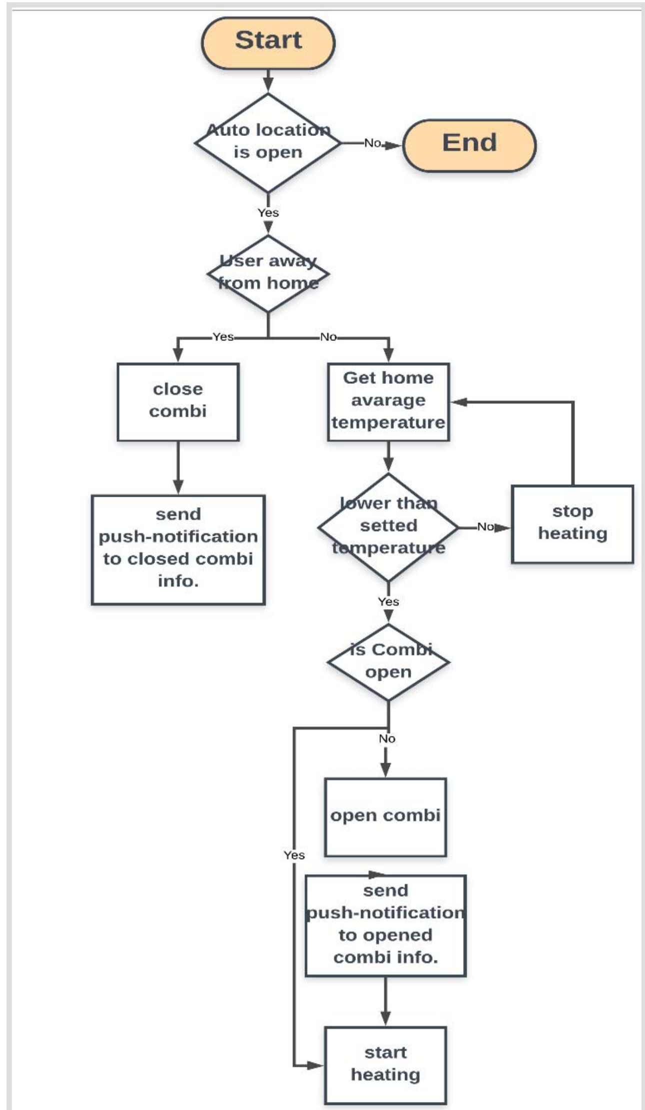

Figure 2. Smart Thermostat Activity Diagram

The user and home location information is kept in the Location class. The CalculateDistance function is executed when the user's location information is updated. If the user moves a certain distance away or comes close to home, the required function in the Home class is executed and the push notification is sent to the user via the sendNotification function. If the user approaches a certain distance from home, the average temperature of home is calculated by calling the CalculateAverageTemperature function in the Temperature class. Also in this class the desired temperature set by the user and the average home temperature information are stored. The average temperature is updated when the calculatedAverageTemperature function is run.

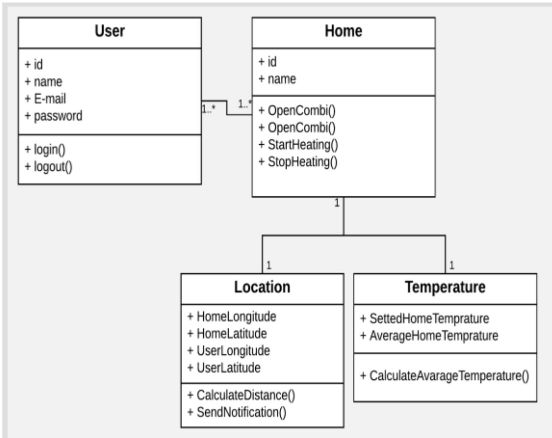

Figure 3. Smart Thermostat Structure.

VI. USE CASE IMPLEMENTATION

The smart thermostat system is developed in 4 different platforms. Arduinos is used as IoT device, Android devices are used as the user interfaces, and MSSQL database is used to store temperatures measured by Arduino devices. In addition, a web service has been developed on the .Net platform. This web service is used by the Android app to display the temperatures.

A. Source Node

Arduinos have id numbers and the id of root node must be 1. The source code is divided into 2 nodes. The nRF24L01 module defines the listener role of the root node. However, the other nodes have defined the sender role. Another feature of the root node is its ESP8266. So, only the root node has a ESP8266 and can connect to the Internet.

Nodes use nRF24L01 modules to send and receive temperature values. These devices communicate over channels called pipe. In the first step, the root node is connected to the Internet with given "ag_ismi" and "ag_sifresi", and then a GET request is sent to the given IP address including “yollanacakkomut”.

B. Mobile App

The Android application periodically receives the user's location information and sends it to the server. This feature is provided with “GoogleService.class” in Libraries folder. This service continues to get the user's location information when the Smart Thermostat Android Application is open.

With the location information, the distance from the user's home is calculated on the server side and the thermostat's status can be changed according to this distance.

When the thermostat changes state (when it is turned on or off) the user is informed by push-notification. For this feature, Google's Firebase Notification API is used.

C. Database

The database schema has 5 tables. In home table we have a private key called HomeID to distinguish homes from each other. This table has 6 columns. “Ev_sicaklik” column is the temperature level at which the home should be, and it is defined by user. “is_running” column indicates whether the Combi is working or not."Ortalama_sicaklik” column represents the average temperature of home and it is calculated using the received data from Arduino sensors. The hassasiyet (sensivity) column is defined to tolerate the small changes in the temperature of home. In this way, the Combi will not be constantly opened and closed due to minor changes. For example, if “Ev_Sicaklik” is 25.0° and “Hassasiyet” is 2.0°**,** the Combi will be only opened when the average temperature is less than 23.0° and it will be closed only when the average temperature is greater than 23.0°**.** Longitude and latitude is used to calculate the user’s distance from home to open or closed the Combi. The tables of the database are shown in Figure 4.

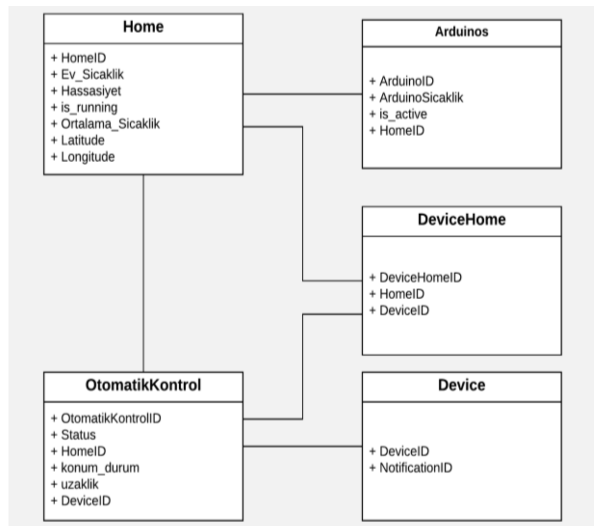

Figure 4. Database Class Diagram.

In Arduino table, we defined a private key called “ArduinoID” to distinguish Arduinos from each other and defined a column called “HomeID” to know which Arduino belongs to which home. “Arduino_Sicaklik” column holds the temperature data which received from the sensors. “is_active” column holds whether this Arduino’s temperature will be taken into consideration or not when calculating the average temperature of the home.

In OtomatikKontrol table, we defined a private key called OtomatikKontrolID to distinguish information from each other and we defined a column called “HomeID" to know which information belongs to which home. “konum_durum” column keeps the information of whether the user is at home or not. We defined “Status” column to see if automatic control is active. “uzaklik” column holds the radius value around the home. The thermostat is turned off when the user goes out of this radius or the thermostat is opened when the user enters to this radius.

In Device table, we defined a private key called DeviceID to distinguish client devices from each other. “NotificationID” column keeps the notification IDs to be able to send notification to the devices. We created DeviceHome table to know which device can control which home. Using this structure, a home can be controlled by multiple devices, and a device can control multiple homes.

D. Server Side

In this work, 4 RESTFUL services are developed: one of them is for Arduino and the others are for Mobile applications. When the user opens the mobile application, it requests the device to register the service with its device Id and notification ID. The reason for sending notification ID each time, is that the notification IDs change at a regular base. As a solution, the server sends the mobile app along with the home information (average temperature, desired temperature, Arduinos which belong to this home, etc.). Mobile client can change the information such as active status of Arduinos and desired temperature level using the update home service. The mobile application sends the request to update location service periodically. The user distance from the home is calculated and if the user is far away from specified distance the Combi is closed and automatic control turned on. Thus, as the user is not at home, the Combi will not be opened even if the home average temperature is less then desired home temperature. We developed a web service for Arduino by which the root Arduino transmit the temperature data which is received from all Arduinos. The service calculates the average temperature and if it is less than the desired temperature and automatic control is not active, the Combi will be opened, and if the average temperature is greater than the desired temperature and the Combi is running, the Combi will be closed.

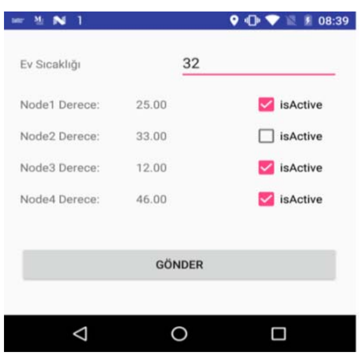

Figure 5. Android Remote Control Screen.

With the mobile application, we can change the average temperature of the home as shown in Figure 5. Moreover, you can observe the temperature of the rooms remotely. You can also change the states (as active or inactive) for the room modules.

VII. CONCLUSION

In this paper, legacy electrical systems are transformed into smart systems. Smart thermostat is also one of these systems. The benefits of the smart thermostat system are:

- The temperature of each room e is measured and the optimum thermostat grade is calculated by means of an algorithm. So, the heat is balanced all over the home.
- Remote management with mobile devices provides the capability that if the user forgets to close the Combi when leaving the home, it can be turned off automatically and unnecessary energy consumption is avoided.
- It makes the user's life easier and provides comfort. For example, the user can open the Combi remotely to warm up the home before going home.
- With the automatic mode, the user does not need to do anything. For example, when you move away from the home, the Combi boiler is automatically closed and then when the user approaches to the home, the Combi boiler is automatically opened. This will increase energy efficiency while providing comfort to the user.

This smart system can contribute to the efficient consumption of energy resources. It can also contribute to the user's life through preventing environmental threats and providing a more comfort for the user.

ACKNOWLEDGMENT

This study is partially supported by the Scientific Research Project No 17-UBE-002 at EGE University, Izmir-Turkey.

REFERENCES

[1] S. Pradeep, T. Kousalya, K. M. Aarsha Suresh, J. Edwin, "IoT and Its Connectivity Challenges in Smart Home", International Research Journal of Engineering and Technology, 2016, pp.1040 - 1043.

[2] I. Khajenasiria, A. Estebsari, M. Verhelst, G. Gielen, "A review on Internet of Things solutions for intelligent energy control in buildings for smart city applications", 8th International Conference on Sustainability in Energy and Buildings, 2017, pp. 770 – 779.

[3] O. Irulegi, A. Serra, R. Hernandez,"Data on records of indoor temperature and relative humidity in a University building", Elsevier, 2017 , pp. 248-252.

[4] S.Sujin Issac Samuel, "A Review of Connectivity Challenges in IoT-Smart Home", 2016 3rd MEC International Conference on Big Data and Smart City, 2016, 4p.

[5] W. Li,S. Kara, "Methodology for Monitoring Manufacturing Environment by Using Wireless Sensor Networks (WSN) and the Internet of Things (IoT)" ,The 24th CIRP Conference on Life Cycle Engineering , 2017 , pp. 323-328.

[6] P. Skocir, P. Krivic, M. Tomeljak, M. Kusek, G. Jezic, "Activity Detection in Smart Home Environment" ,20th International Conference on Knowledge Based and Intelligent Information and Engineering Systems ,2016, pp. 672-681.

[7] Karaduman B., Challenger M., Eslampanah R., "ContikiOS based Library Fire Detection System", 5th International Conference on Electrical and Electronics Engineering (ICEEE 2018), Istanbul, Turkey, May 3-5, 2018. (Accepted)

[8] B. Karaduman, T. Asici, M. Challenger, R. Eslampanah, "A Cloud and Contiki based Fire Detection System using Multi-Hop Wireless Sensor Networks", International Conference on Engineering & MIS (ICEMIS 2018), Istanbul-Turkey, 19-21 June, 2018 (Accepted)

[9] S. Arslan, M. Challenger, O. Dagdeviren, "Wireless Sensor Network based Fire Detection System for Libraries", 2nd International Conference on Computer Science and Engineering (UBMK'17), Antalya-Turkey, 05-08 October, 2017, DOI: 10.1109/UBMK.2017.8093388 (In Turkish).