http://dx.doi.org/10.35596/1729-7648-2023-29-1-64-71

Original paper

UDC 339.138

SIMULATION OF IoT SMART HOME NETWORK WITH DECISION MAKING BASED ON MAJORDOMO PLATFORM

ULADZIMIR A. VISHNIAKOU, CHUYUE YU

Belarusian State University of informatics and Radioelectronics (Minsk, Republic of Belarus)

Submitted 27.01.2023

Abstract. The purpose of the article is to model a smart home with decision-making and the use of the MajorDoMo IoT platform instead of a cloud platform. The main features of the article are: modeling of the IoT network for controlling the control unit on the MajorDoMo platform, the use of the MajorDoMo IoT platform in the Windows system for making decisions when controlling household appliances (fan or air conditioner) based on the temperature and humidity values received from a smartphone modeling sensors, transmitting the results of controlled data to the user's home page. The structure of the MajorDoMo platform is given, its functions are described, and commands for decision support are considered. The use of product rules is proposed as a decision-making model. An example of such a description for data regulation by the household appliances is given. The user can log into the IoT network and view changes in temperature and humidity in real time, as well as check the regulation of the equipment. The possibilities of the PHP language for the implementation of the control process are presented, in which the household device begins to adjust the parameters when the environmental monitoring system detects that the corresponding data exceeds the set threshold

Keywords: IoT network, sensor simulation, smart home, decision-making, MajorDoMo platform, temperature, humidity.

Conflict of interests. The authors declare no conflict of interests.

For citation. Vishniakou U. A., ChuYue Yu (2023) Simulation of IoT Smart Home Network with Decision Making Based on MajorDoMo Platform. *Digital Transformation.* 29 (1), 64–71. http://dx.doi.org/10.35596/1729-7648-2023-29-1-64-71.

МОДЕЛИРОВАНИЕ СЕТИ IoT «УМНЫЙ ДОМ» С ПРИНЯТИЕМ РЕШЕНИЙ НА ОСНОВЕ ПЛАТФОРМЫ MAJORDOMO

В. А. ВИШНЯКОВ, ЧУЮЭ ЮЙ

Белорусский государственный университет информатики и радиоэлектроники (г. Минск, Республика Беларусь)

Поступила в редакцию 27.01.2023

Аннотация. Смоделирована сеть умного дома с принятием решений и использованием локальной IoT-платформы MajorDoMo вместо облачной платформы. Основными задачами являлись: моделирование сети IoT для управления на платформе MajorDoMo, использование платформы MajorDoMo IoT в системе Windows, описание принятия решений при управлении бытовой техникой (вентилятором или кондиционером) на основе значений температуры и влажности, полученных от датчиков (смоделированных средствами смартфона), передача результатов контролируемых данных на домашнюю страницу пользователя. Приведена структура платформы MajorDoMo, описаны ее функции и рассмотрены команды для поддержки принятия решений. В качестве модели принятия решений предлагается использовать продукционные правила. Приведен пример описания правил для регулирования данных бытовыми приборами. Пользователь может войти в сеть интернета вещей и просматривать изменения температуры и влажности в режиме реального времени, а также проверять регулировку оборудования. Представлены возможности языка PHP для реализации процесса управления, при котором бытовое устройство начинает корректировать параметры, когда система мониторинга окружающей среды обнаруживает, что соответствующие данные превышают установленный порог.

Ключевые слова: сеть интернета вещей, моделирование датчиков, умный дом, принятие решений, платформа MajorDoMo, температура, влажность.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Для цитирования. Вишняков, В. А. Моделирование сети IoT «Умный дом» с принятием решений на основе платформы MajorDoMo / В. А. Вишняков, ЧуЮэ Юй // Цифровая трансформация. 2023. Т. 29, № 1. С. 64–71. http://dx.doi.org/10.35596/1729-7648-2023-29-1-64-71.

Introduction

The smart home management system [1] uses the IoT platform as the basis for the implementation of the IoT network. It considers household appliances, home electronics, metering devices for resource consumption (water, electricity) as the main objects of management. It applies the following technologies: integrated wiring, network communication, security, automatic control, audio and video registration technology for effective control and management of residential premises. This network can receive different types of sensor signals from different sensor devices [2] and launch control commands or, possibly, extract control commands using human remote control or manually starting devices. If people are not indoors, they can also control all internal devices via the Internet at a distance [3]. In [4], the authors considered the process of modeling the IoT network. Alibaba IoT cloud platform has been selected as a cloud server for the IoT network project. Since it is not convenient to use cloud resources to manage small objects, consider using a local platform to manage a smart home. Taking the MajorDoMo platform as such an example, we will show how to implement the process of regulating the conditions in the house based on environmental data.

Smart home control system

The smart home management system has five main functions: flexible system composition, convenient management [5], numerous control functions, sharing of information resources [6], ease of debugging and installation. In order to integrate resources and provide holistic smart home services, such as multitasking, intelligence and platforming, a corresponding industry is developing [7]. Intelligent automatic control of home devices is the main feature of the smart home platform. An ideal smart home automatic control system should have three main functions. Firstly, it can automatically manage the home environment without disturbing the user, and free the user from the tedious work of monitoring home devices (maintenance automation). Secondly, it can accurately predict and correct the working condition of smart devices at home and avoid improper handling of them, i. e. ensure (accuracy of maintenance). Thirdly, it can fully study user habits, understand their needs and optimize user experience, i. e. create a service environment [8]. The decision-making model proposed in this article is intended for the first function, that is, "service automation".

MajorDoMo platform

We used the MajorDoMo IoT platform to manage a smart home, because it has a large number of users and an experienced forum and community where we can find information from the moment of its launch. The platform is implemented with open source and is constantly supported by a lot of specialists and users, it is becoming more versatile and able to help us flexibly solve various problems we have encountered, simulate a real environment so that we do not have to create it offline [9].

In MajorDoMo, methods and properties are distributed according to the class of the object, that is, the object that basically introduces the concept, and the collection of objects is divided into different classes, each with its own properties and methods of interaction. MajorDoMo also supports the introduction of other subclasses that will have their new properties that can be set by inheritance. The original hardware form of the MajorDoMo object is a controller that can communicate between the controlled device or sensor and the central system in which the MajorDoMo software package is located. Its structure includes primitives of classes, properties, methods and objects, which are introduced with the help of specialized additions. For a specific user, this means that he does not need to understand the nuances of device exchange protocols, their internal properties or methods in each case. Fig. 1 shows the general structure of the system when the MajorDoMo collects data.

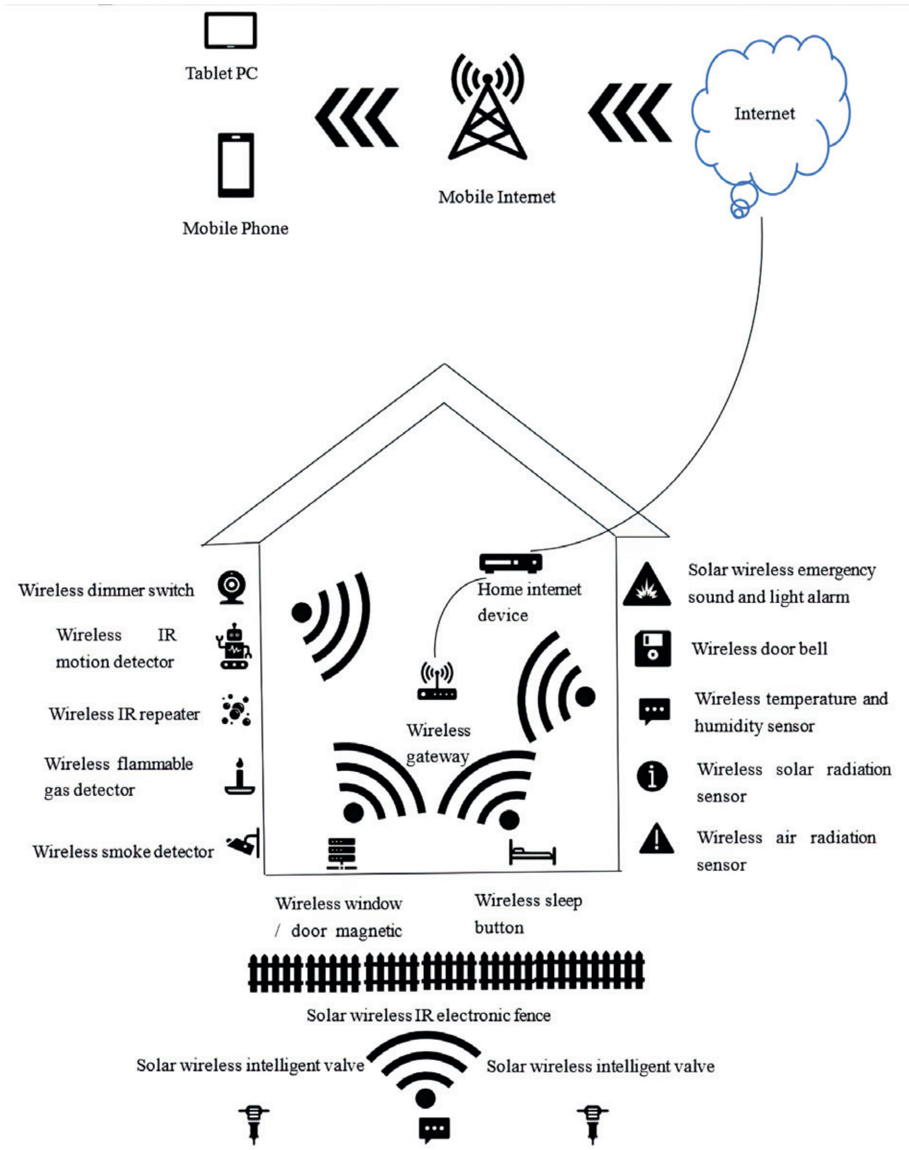

Fig. 1. The general structure of the system when MajorDoMo collects data

The main scripting language of MajorDoMo is the PHP programming language, which can be used to write scripts for the system's response to various events. In addition to the basic language structure, the environment has a library of native functions that can be used to automate any process. In addition to the built-in system functions, you can create your own custom functions.

Visual programming based on Google Blocky is integrated into this environment using a graphical representation, and when building algorithms, it is enough to combine them in the required order. MajorDoMo is designed with many functions, you can interactively set schedules, manage devices with or perform many other actions that will be performed thanks to the capabilities of the intelligent assistant. Fig. 2 shows some of the functions that majordomo can implement. Within the framework of the system, you can extract data from the network and use it in shell scripts.

One of the key features of MajorDoMo is that it supports many communication protocols and several switching protocols – MQTT, Z-Wave, Broadlink, etc. It also supports a large number of devices from different manufacturers and has a very wide range of adaptability. Communication between the components of the platform system is carried out via Wi-Fi, routers, Ethernet, etc., on the one hand, and interaction through transport protocols such as MQTT, etc.

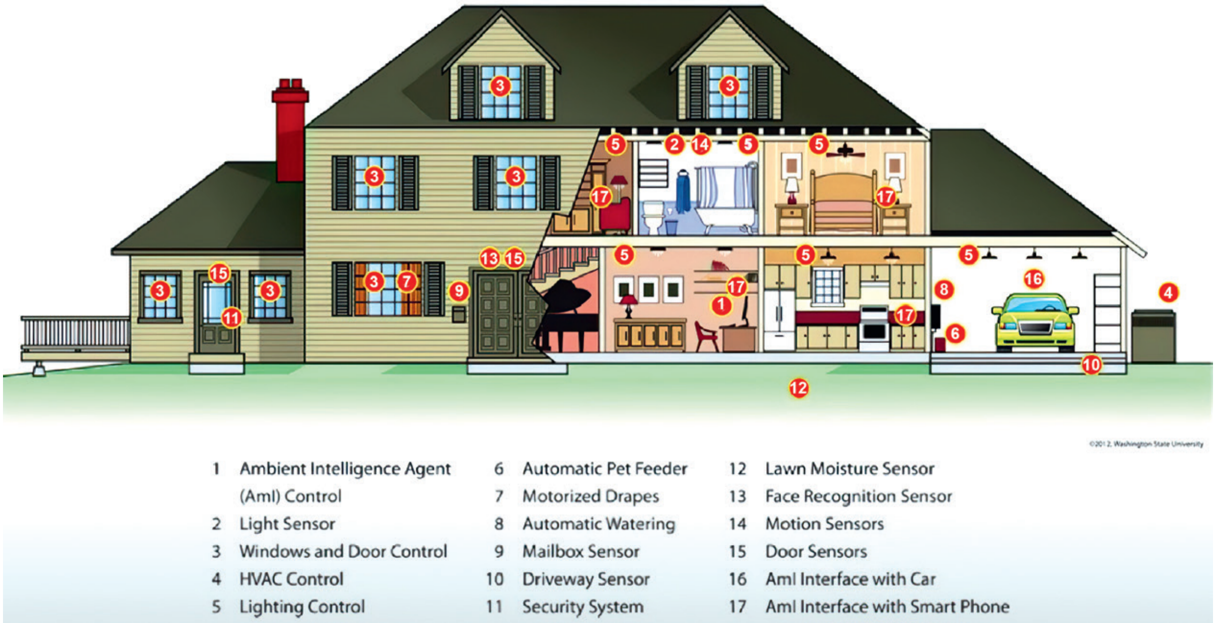

Fig. 2. Some of the features that MajorDoMo has

For most devices in a smart home, it is usually necessary not only to follow the instructions of the control part, but also to return their state. The platform uses a hybrid approach to task data management, i. e. the central server remains the largest data storage and collection point, as shown in Fig. 3.

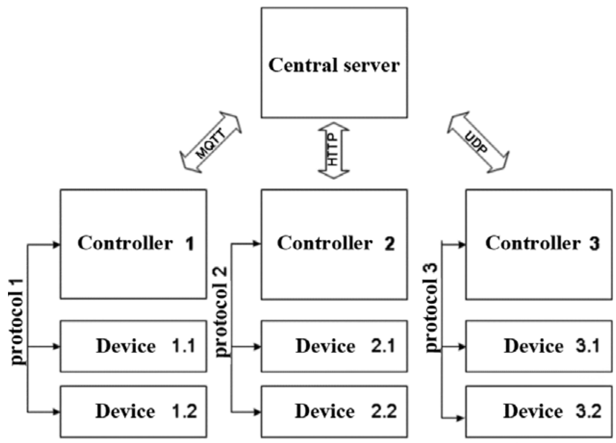

Fig. 3. Hybrid management of task data

In fact, the hardware components of the central computer itself are not important, the system is represented by a dedicated network shell operating in the server space of the home network, which can run under the operating system, Windows or Linux. Client devices can be represented by almost any device in production, from MQTT devices to Broadlink RF radio control. Being a full-fledged Internet of Things platform, MajorDoMo has four different components for implementing various interaction models, namely:

1. IoT devices collect various types of data (voice, video, images, or structured time series data such as vibration, heat, etc.) from the environment. Some IoT devices also interact with the environment rather than just collecting data.

2. Solutions for connecting to a global network that allow devices to transfer data to the cloud or receive commands from it.

3. Data processing software that performs analytics for devices. Based on the results of the analysis, they can make decisions based on the data.

4. Monitoring and management software that provides a user interface. The most common use of the user interface layer is remote device management and visualization of analytics results.

Real-time decision-making models in smart home control systems require the use of so-called peripheral computing on Internet of Things platforms, which is a technology that allows intelligent devices to send or receive data to and from the Internet of Things platform, as well as perform other operations.

As an example, here are some of the commands that can be executed on the platform in MajorDoMo, are:

DebMes ($errorMessage, $logLevel) – writes the message to the main log file. The *$logLevel* parameter is optional and is set to debug by default;

sayTo (Phrase, LEVEL, NAME) – can use the system terminal name as the NAME and if it is a terminal on MajorDroid, it will be sent a command to say it;

getHistoryMin ($varname, $start_time, $stop_time) – returns the minimum value for the period;

getHistoryMax ($varname, $start_time, $stop_time) – returns the maximum value for the period;

getHistoryCount ($varname, $start_time, $stop_time) – returns the number of values for the period;

getHistorySum ($varname, $start_time, $stop_time) – returns the sum of values for the period;

getHistoryAvg ($varname, $start_time, $stop_time) – returns the average of the period;

getHistory ($varname, $start_time, $stop_time) – returns an array of history values.

Real-time decision model of IoT smart home management

MajorDoMo is one of the smart home platforms, and the implementation of automatic control technology is based on the establishment of rules. Therefore, we set the rules for limiting the system in accordance with the target tasks and target objects. Assumed that the relationship between the object and the object was:

– when the humidity exceeds the threshold value, the fan turns on, otherwise it turns off. The initial value of the humidity threshold is 50 %;

– when the temperature exceeds the threshold value, the air conditioner turns on, otherwise it turns off. The initial value of the temperature threshold is 25 degrees;

– the user can remotely set the temperature and humidity threshold.

Let's choose a production model for decision-making: if A, then B [10]. Since there was no real humidity sensor and temperature sensor in the simulation to simulate the process of collecting data in real time, we defined the timer function so that the temperature value and humidity sensor were updated every minute. The value of the temperature sensor was “minutes of current time – hours of current time”, and the value of the humidity sensor was “minutes of current time + hours of current time”. The specific model in this case has the form:

TempSensor01.value = a
HumSensor01.value = b
Realtime.hour = h
Realtime.minute = m
Airconditioner1.status = {0,1}
Fan1.status = {0,1}
a = m – h
b = m + h
if a ≤ 25, then Airconditioner1.status = 0, else, Airconditioner1.status = 1.
if b ≤ 50, then Fan1.status = 0, else, Fan1.status = 1.

Model implementation based on the MajorDoMo platform

To implement the smart home control function, so that smart home devices will turn on or off depending on changes in temperature and humidity, first create objects. There were several categories of parent objects in the MajorDoMo objects column: computer, bedroom, weather, operation, room, device, system status, timer. Since we needed a sensor device object, so the SDevices object class was selected, a humidity sensor and a temperature sensor object were found in its subclasses, and an instance of a humidity sensor was created, which is named HumSensor01, as shown in Fig. 4.

Similarly, an instance of a temperature sensor was created, the value TempSensor01 was set for this sensor. Then an instance of the air conditioner and fan is created. Inter-object connections have also been created and temperature and humidity thresholds have been set.

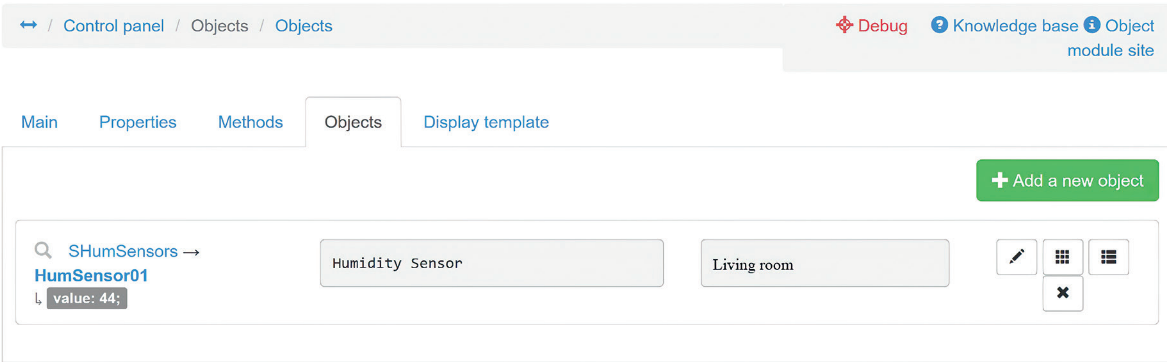

Fig. 4. Instantiated humidity sensors

After updating the system, you can see that the temperature and humidity on the MajorDoMo homepage have been gradually changing over time, with a frequency of once per minute, as shown in Fig. 5.

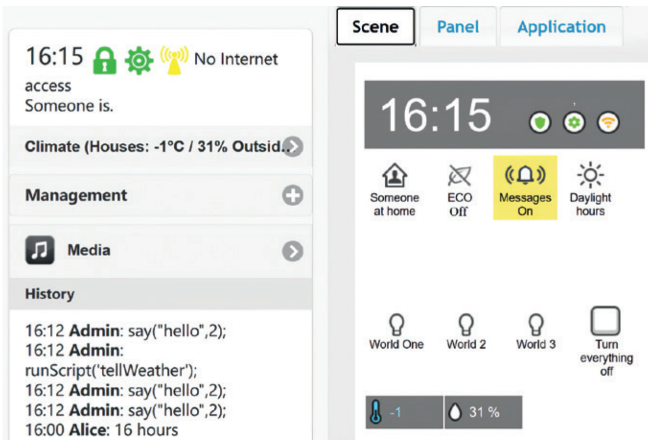

Fig. 5. The display of temperature and humidity data

Then the threshold values for the air conditioner and fan controllers are set during programming, these threshold values are recorded in the created StatusUpdated functions (air conditioner and fan, respectively) in order to achieve a status change depending on the temperature and humidity in the room. Then we set the temperature threshold for calling the air conditioner at 25 and similarly set the threshold for starting the fan when the humidity reaches 50 %, and selected "do not call the parent function" in the settings.

The test results are shown in Fig. 6, where the horizontal line reflects the change of temperature value, the dotted line reflects the change of humidity value, and round dots reflect the change of status of the fan and air conditioner, respectively.

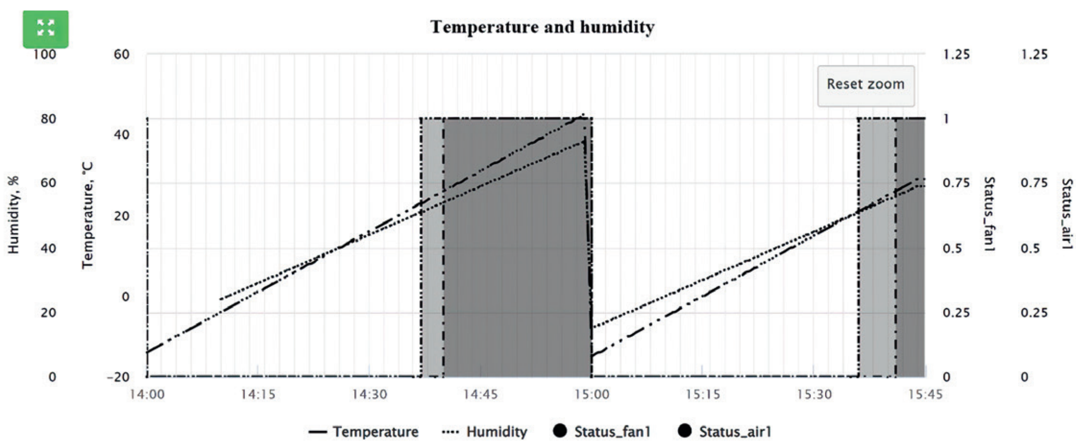

Fig. 6. The process of smart home data regulation

As shown in Fig. 7, for an air conditioner, it did not work when the temperature was below or equal to 25 degrees, and started working when the value exceeded the threshold value.

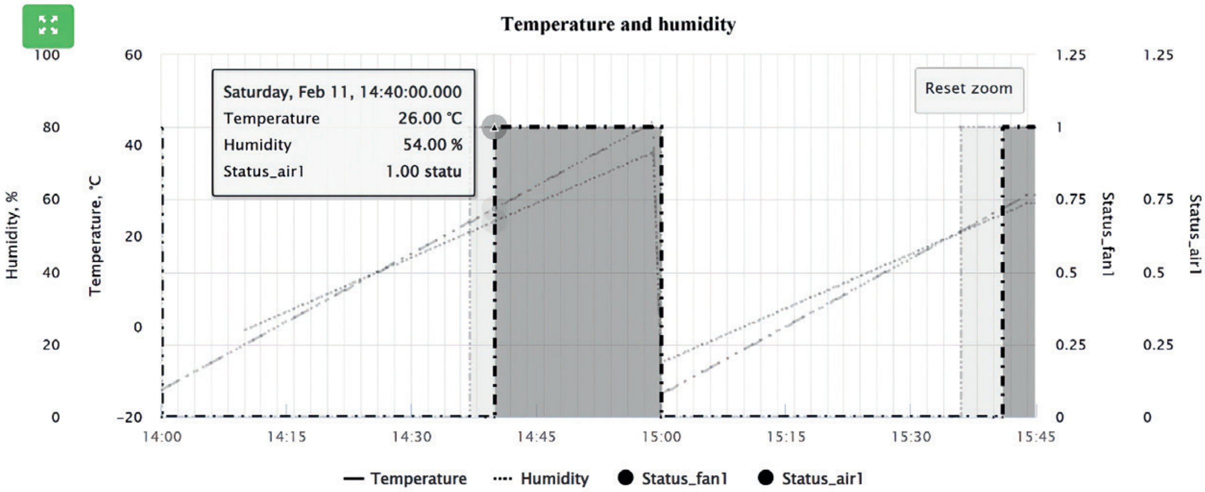

Fig. 7. Air conditioning management

As for the fan, it did not work when the humidity was below or equal to 50 degrees, and started working when its value exceeded the threshold value (Fig. 8).

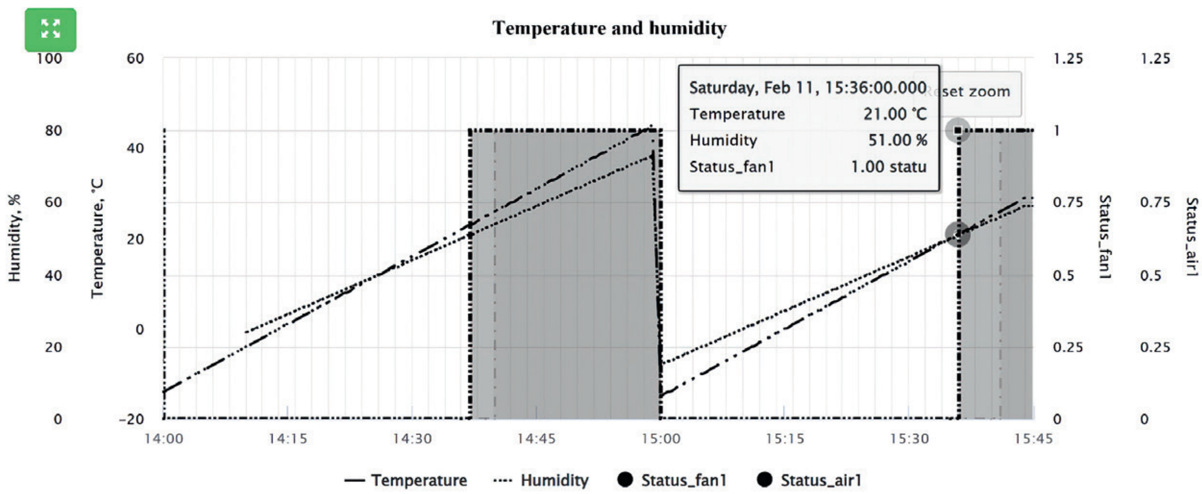

Fig. 8. Fan management

In addition, users can log into the Internet of Things platform and directly change the threshold value of temperature and humidity.

Conclusion

1. The simulation of an intelligent smart home management system based on the IoT local platform MajorDoMo is presented. Four different components are given for the implementation of four models of interaction with the external environment, including Internet, fog and cloud computing. The possibilities of using this platform for the implementation of an intelligent control system with the collection and processing of data for decision-making (using product rules) are shown. After forming a real-time decision-making model, the process of regulating the data of the home environment using the PHP scripting language is implemented.

2. Thresholds of maximum temperature and humidity are set, if the value of the corresponding sensor exceeds the threshold value, the device turns on, otherwise it is turned off. When the humidity was higher than the threshold value, the fan turned on, and when the temperature was higher than the threshold value, the air conditioner turned on. During the simulation, since there was no corresponding sensor, a timer was used to simulate real-time data collection.

References

1. Kadam R., Mahamuni P., Parikh Y. (2015) Smart Home System. *International Journal of Innovative Research in Advanced Engineering*. 2 (1), 81–86.
2. Zeng H. Y., Zheng X., Wei Y. (2023) Research on Smart Home Control System Design Based on Internet of Things Technology. *Electronic Production*. 31 (1), 116–120. https://d.wanfangdata.com.cn/periodical/dzzz202301030.
3. AmSuk O. (2018) Design and Implementation of Smart Home Remote Control Based on Internet of Things Service Platform. *Journal of the Korea Institute of Information and Communication Engineering*. 22 (12), 1563–1570. Available: https://doi.org/10.6109/jkiice.2018.22.12.1563.
4. Vishnyakou U. A., Yu C. Y. (2022) Modeling IoT Smart Home Network. *Doklady BGUIR*. 20 (6), 78–84. https://doi.org/10.35596/1729-7648-2022-20-6-78-84.
5. Jenal M., Omar A. N., Hisham M. A., Noh W. N., Razali Z. A. (2022) Smart Home Controlling System. *Journal of Electronic Voltage and Application.* 3 (1), 92–104.
6. Zhang L., Leung H., Chan K. C. C. (2008) Information Fusion Based Smart Home Control System and its Application. *IEEE Transactions on Consumer Electronics*. 54 (3), 1157–1165.
7. Papadopoulos N., Meliones A., Economou D., Karras I., Liverezas I. (2009) A Connected Home Platform and Development Framework for Smart Home Control Applications. *2009 7th IEEE International Conference on Industrial Informatics*. 402–409. Doi: 10.1109/INDIN.2009.5195838.
8. Mao B., Xu K., Jin Y. H., Wang X. L. (2017) DeepHome: a Control Model of Smart Home Based on DeepLearning. *Chinese Journal of Computers.* 40 (8), 1–15. https://www.cnki.net/kcms/detail/11.1826.tp.20170221.1009.002.html.
9. Gololobov V. (2019) *IoT as Intranet of Things with Raspberry Pi and MajorDoMo*. Moscow.
10. Russell S., Norvig P. (2020) Artificial Intelligence: a Modern Approach, 4th ed. *Pearson*. 1136.

Author's contribution

Vishniakou U. A. carried out the task setting for the article, proposed topics, suggested and provided information on the experimentally, selected IoT platform.

ChuYue Yu performed experiments, plotted diagrams, recorded experiments.

Information about the authors

Vishniakou U. A., Dr. of Sci. (Eng.), Professor at the Department of Infocommunication Technologies of the Belarusian State University of Informatics and Radioelectroctronics

ChuYue Yu, Postgraduate at the Department of Infocommunication Technologies of the Belarusian State University of Informatics and Radioelectroctronics

Address for correspondence

220013, Republic of Belarus,
Minsk, P. Brovki St., 6
Belarusian State University
of Informatics and Radioelectroctronics
Tel.: +375 44 486-71-82
E-mail: 1259017720@qq.com
ChuYue Yu