

# Design and Implementation of a Smart House System with Voice Control



Nataliia Konkina

**Abstract:** *Smart House is an intelligent management system that integrates all equipment into a single complex. It solves various tasks in the field of security, life support, entertainment, and communication. This paper presents a complete design and implementation of a Smart House system with voice control, describes the hardware and software parts as well as the interaction between them. Voice control performed with simple instructions using Microsoft Speech Platform. Recognized commands will be encrypted on the software side and then will be sent via Bluetooth HC-06 module to the hardware side for execution. Among the developed features for the created prototype are lighting control, home temperature control, sleep mode control, the possibility of setting an alarm clock, security mode and gas leakage check. In case of problems, a user will receive a notification via email and/or SMS. Finally, this paper presents the results of experiments for voice control, which shows that voice control in Smart House is the next step in improving this intelligent system, is the next step in improving human-machine interaction and it provides great help for people with special needs and disabilities.*

**Keywords :** Arduino, Smart Home, Smart House, voice control.

## I. INTRODUCTION

Creation of automated control systems for engineering equipment of buildings and structures is the newest direction in the field of industrial automation. It is named Smart House and is defined as a comprehensive set of hardware and software to build an integrated automation system for engineering subsystems [1]. Smart House is comfort and safety, environmental friendliness and energy saving. In addition, automatic systems help people with disabilities to save on the services of a carer or nurse. That is why this system is becoming increasingly popular all around the world. The Smart House system is a complex of subsystems that are integrated into a single one. Such subsystems include heating, water supply, air conditioning, lighting, access, security and safety subsystems, audio-video equipment (multi-room) and a number of others. In today's world, smart houses are controlled with the help of mobile devices and personal computers. Communication between the graphical

Revised Manuscript Received on February 05, 2020.

\* Correspondence Author

**Nataliia Konkina\***, Department of Automation of Power Processes and Systems Engineering, National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv 03056, Ukraine. Email: natalia.konkina.13@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](#) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

interface and the hardware is carried out via a home Internet network [2-7].

The article describes the process of creating a microprocessor management system of the building and user interface (UI) that will enable to manage and track the operation of its objects (lighting, power supply, air conditioning) in automatic mode. It will also demonstrate stages of creating a working prototype for the Smart House system (from the concept to a ready solution). To support people with disabilities, each command in the system is voiced and user can control the house with the help of his/her voice.

## II. DEFINING THE FUNCTIONAL REQUIREMENTS FOR A SMART HOUSE PROTOTYPE

It is impossible to start developing a prototype without clear objectives, so it is necessary to decide which functionality the Smart House will have. For this goal, the author decided to create a UML use case diagram describing the given tasks (Figure 1).

After that it has been decided which sensors should be used, the author established hardware architecture, selected appropriate microcontrollers, considered the design of user interface, selected the most appropriate technologies and tools for software implementation and chose the type of communication between the hardware and the software parts.

### A. Determination of prototype functionality

Smart House is a complex of standards, combined with various devices into a single building management system [8-11]. Each of the subsystems performs its specific task.

The existing varieties of subsystems are: microclimate system (heating, air conditioning, humidification) [12], security system (security, fire, access system, gas leakage control, video surveillance) [13], power supply system (standby systems, overload control, lighting system) [14], communication system (phone, local area network, SMS alert) [15-16], remote control system [17], entertainment system (multi-room) [18] and other.

The decision was made to implement as many subsystems as possible, provide voice control for them and look at the final results.

The following use case diagram describes the main features of two actors - the building and its owner. The precedents of each subsystem (microclimate, security, power supply, communication and entertainment) are highlighted into separate rectangular groups.



# Design and implementation of a Smart House system with voice control

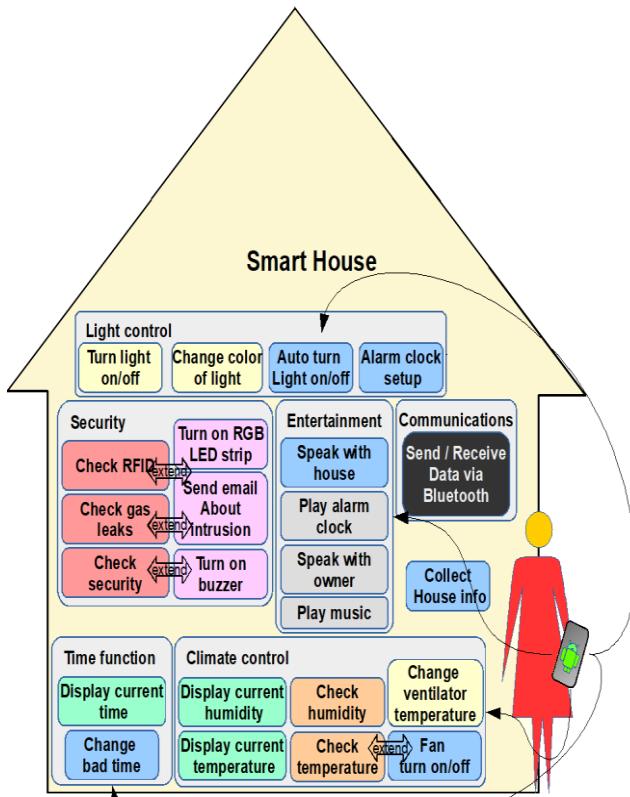


Fig. 1. UML use case diagram

As you can see, the author decided to implement the next set of systems: security, timing, lighting, climate control, communication and entertainment. The main functionality of each subsystem described in detail below.

**- Security system.** This subsystem gives protection to the house when there is no owner is present at the house. The security mode should be easily enabled/disabled through the graphical interface. Also, it should be switched off with an RFID tag (with this approach we can allow only authorized users to enter the house), so the ID of each card should be saved. Only users who have cards with appropriate identification codes will be allowed to open the house and deactivate the alarm system. In 'security mode', if at least one of the sensors is triggered, the microcontroller must send a signal to a unit/device to turn on the LED strip as well as transmit code term/messages to the graphical interface, which will signal about the intrusion. The graphical interface will receive the messages and generate notifications that will be sent to the user.

Also, a gas leak and smoke detection monitoring test should be created in the prototype(the working principle should be similar to a security system, with the small exception - it will be impossible for the owner to disable gas checking and smoke leakage).

**- The climate control system** should work as follows: the user will enter the desired air temperature into a graphical interface, which will generate a code term and send it to the microcontroller. At this time the microcontroller will be waiting for messages. If the microcontroller receives a message that corresponds to the temperature change, the microcontroller will remember the new value of the desired air temperature. If the air temperature becomes equal to or higher than the normal value, the microcontroller must signal to turn on the fan.

**- Light control system** (power supply system). It should be possible to control the light by pressing just one key on the UI. The standard light control chain should be very simple - a graphical interface - Internet/Bluetooth - a microcontroller - a LED lamp.

Automatic light control should work with an Ultrasonic distance sensor and should be fully performed by a microcontroller.

**- Time function.** The time module has a backup power source - a regular watch battery (e.g. CR2032). If electricity supply of the house is disrupted, the module keeps counting current date and time due to the backup battery.

**- Communication system.** This system should provide communication (signals, messages, datagrams) between the microcontroller, all the sensors connected to it and a graphical interface.

**- Entertainment system.** The system provides the ability to control the colour of lighting in the living room. To do this, the user will need to attach a coloured card to the sensor and the room lighting will have to change to the colour of the card. Changing the colour of the lighting should also be available from the graphical interface.

Moreover, for the convenience of a user, it is necessary to provide voice control. The house should be able to respond to code phrases. Even more, the house must support a conversation with the owner, manage music playback – enable/disable music by voice commands. It is necessary to provide an opportunity to set an alarm clock, to control 'sleep time' mode.

## III. SMART HOUSE HARDWARE DESIGN

### A. Required sensors

To build the prototype, it has been decided to use the following sensors (see Table 1).

Table- I: Sensors, used in the "Smart House" prototype

Subsystem	Sensors, devices
Climate	DHT11 Temperature & Humidity Sensor, 30x30x10 fan 5V
Lightning	4 LED lamp, RGB LED strip, HC-SR04 ultrasonic distance sensor
Clock	LCD1602 Module, 4-digit LED module, DS3231 Real-time Clock Module
Security	RC522 RFID Reader Module, HMB-12 active buzzer, MQ2 analog gas sensor
Connection	HC-06 Bluetooth Transceiver Module
Entertainment	RGB LED TCS230 color sensor
Voice control	Microphone, Microsoft Speech Platform

### B. Selecting a central controller

The author has done a comparison of Raspberry Pi, Arduino, Intel Galileo, UDOO NEO and Beagle bone models as shown in Table 2.



Table- II: . Raspberry Pi, Arduino, Intel Galileo, UDOO NEO and Beagle bone comparison [19].

Parameter	Raspberry Pi	Arduino	Intel Galileo	UDOO NEO	Beagle bone
Processor	Broadcom BCM2837 64bit ARMv7 Quad Core 1.3 Ghz	ATMEGA8, ATMEGA 1280	Intel Quark X1000 – 400 MH z single core	Free scale i.MX 6SoloX 1GHz ARM Cortex-A9 with Cortex-M4	AM335 9 GHz ARM Cortex-A8
RAM	1GB	16-32 KB	512 Kb on-chip SRAM 256Mb DRAM	512MB or 1GB(full)	512MB DDR3 RAM
POWER[W]	10	5	15	10	15
COST[\$]	40	30	70	65	55

For the price, community support in the Internet, it has been decided to choose between Raspberry Pi and Arduino platforms.

The Arduino platform is very convenient for prototyping. Open architecture allows developers to customize Arduino for any purpose. The Arduino platform is especially convenient for developers to work with because it gives an almost instantaneous response to programmed commands. Raspberry PI does not have such development tools as

Arduino. So, it has been decided to use an Arduino for the development.

As there are a number of devices that we had to connect, it has been decided to use the Arduino Mega platform (with Atmega 2560 microcontroller).

### C. Connecting devices to the Arduino platform

Fig. 2 shows a diagram of the connected devices to the ATMEGA 2560 microcontroller.

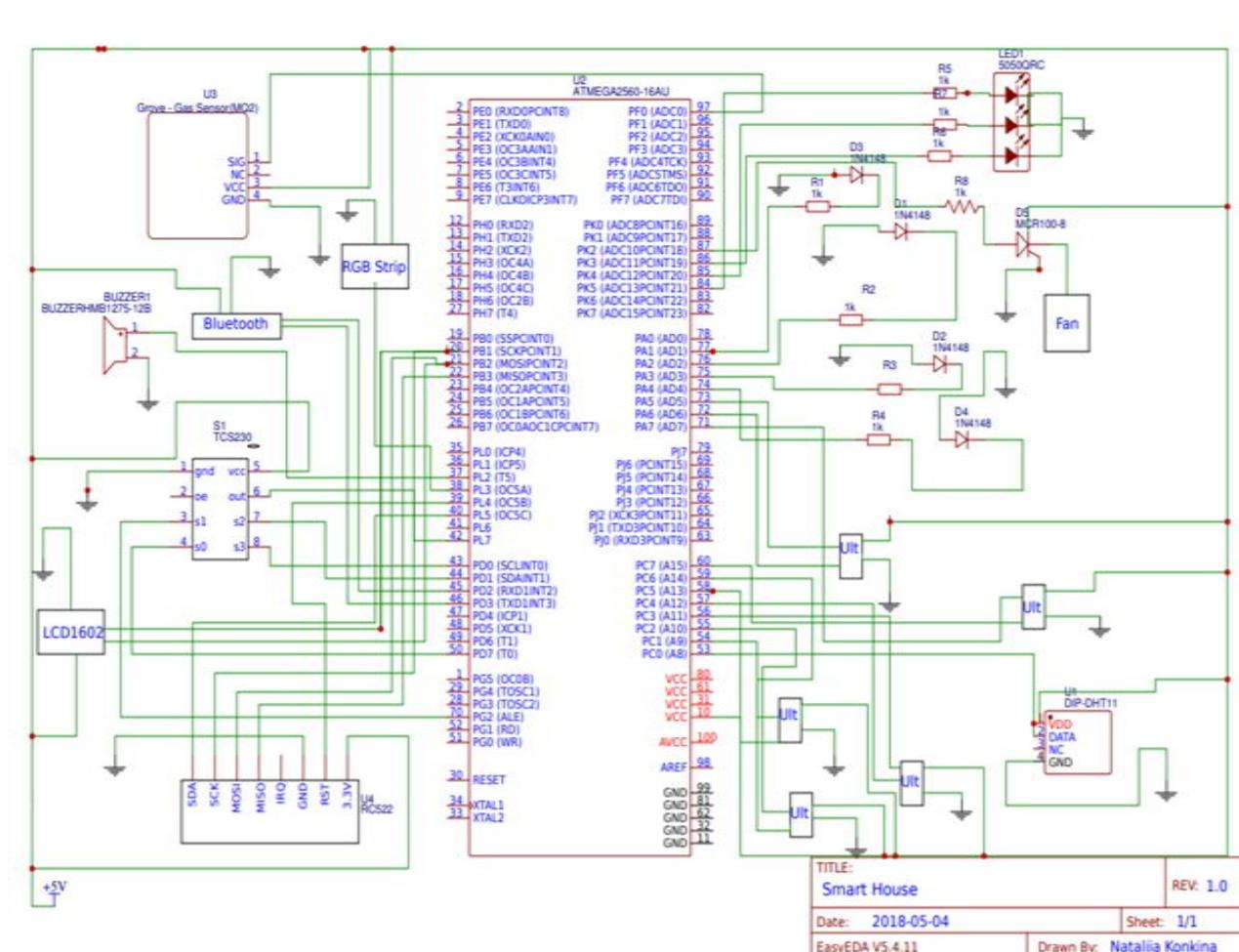


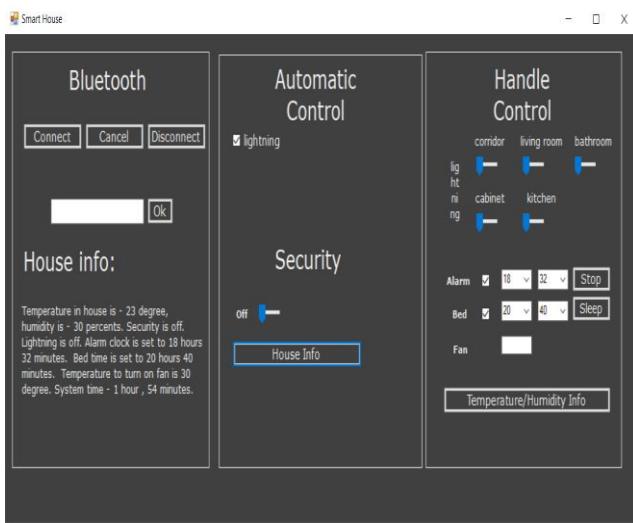
Fig. 2. Diagram of the connected devices to the ATMEGA 2560 MC

## IV. SMART HOUSE SOFTWARE DESIGN

A graphical user interface allows you to monitor the condition of the house, as well as manage devices connected to the Smart House system. The standard purpose of the graphical interface is to control the light, turn on/off security mode, display humidity and temperature information, obtain current information about the condition of the sensors, setup the bad time mode[21, 22].

The following tools were used for prototyping: C#, WinForms, Microsoft Speech Platform, a regular text file was used as a database. However, in large Smart House systems, professional RDBMS can be used, e.g. SQLight, PostgreSQL, MySQL.

Figure 3 shows the graphical interface, which was the result of the development.



**Fig. 3. Graphical interface**

To use the system comfortably, as well as to support people with disabilities, each command voiced by the house. To start recognition, a user has to say "Start recognition". After this phase, the computer will allow the user to talk to it and listen to music. The command "Stop recognition" will stop the voice control.

However, speech recognition works all the time even if the functions mentioned above are turned off. For example, in emergency cases, some phrases will be recognized anyway - phrases that signal that the user needs help or special services from outside the house. For example, the phrase "House, I need medical help!" will be recognized anyway, in this case, a special message will be sent to the medical centre. The phrase "House, I'm in mortal peril!" is a message to the security service, etc.

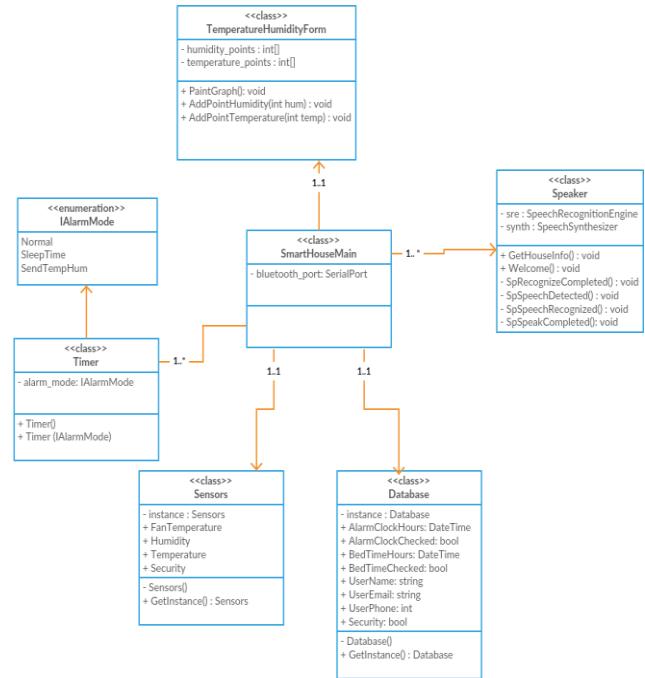
If the security mode is activated, in the case if at least one indicator from the Ultrasonic sensors changes, the light in the house will turn on, the Piezo Emitter will start working, the LED strip will turn red, the owner of the house will receive an e-mail about the intervention. The message will also be sent to the police.

Figure 4 shows a class diagram that displays the general architecture of the graphical interface.

Tools used in interface development may vary, but the overall architecture will remain the same on all platforms.

In developing the GUI the author used the C# programming language as well as WinForms technology, but

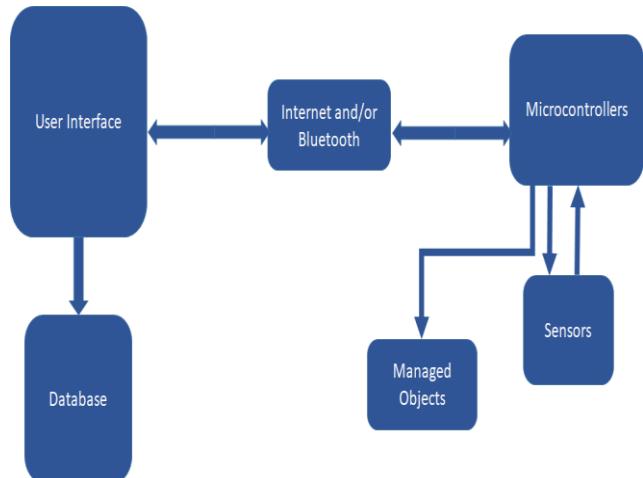
the choice may vary depending on the developers' preferences.



**Fig. 4. UML class diagram**

## V. HARDWARE AND SOFTWARE INTERACTION

To implement the Smart House system the following interaction between the hardware and software can be offered (see Figure 5).



**Fig. 5. The interaction between software and hardware parts**

Where the **User Interface** is a graphical interface that controls and monitors the House.

**Database** - a database that will store user information as well as statistical data such as air temperature, humidity, bedtime, alarm clock and any other useful information that will be implemented into the system.

Depending on the complexity of the system, tiny database as SQLite or large RDBMS such as MongoDB, MySQL, etc can be used, as well as a simple text document in which the necessary information will be stored can act as a database.

**Internet/Bluetooth.** There are a large number of solutions exist to transfer information from the software to the hardware. It's possible to do with Wi-Fi, Bluetooth, ZigBee and other standards.

To implement the prototype, it has been decided to use the Bluetooth HC-06 module. The operating voltage of this Bluetooth module is 3.3V, but its inputs are tolerant to 5V, so it is compatible with all Arduino boards. Bluetooth module HC-06 can act only in slave mode which means that it cannot independently connect to other Bluetooth devices.

**Microcontrollers** - depending on the chosen architecture (distributed or centralized), the number of microcontrollers will vary. For home automation, you can use microcontrollers of any popular family: 8051, PIC, AVR, ARM etc.

**Sensors** - sensors that allow you to get the current state of the environment inside or outside the house. The most useful examples of sensors are humidity and temperature sensor, PIR motion sensors, ultrasonic distant sensors, timer, gas and smoke detectors, etc.

**Managed Objects** - Thanks to the advances in science and technology, almost any modern house equipment can be controlled or can act independently. These are electric lamps, surveillance cameras, TVs, boilers, electric stoves, other kitchenware, etc. Each object can be controlled manually as well as with the graphical user interface.

In addition, the Smart House should make its own decision about switching the objects on/off (for example, switching off the light if it's bedtime, notify the security service about the intrusion, send SMS and email notifications to the owner in case of problems, etc.).

## VI. SPEECH RECOGNITION

The platform from Microsoft was chosen for voice control because it provides a large number of features that work perfectly well with the .NET framework. Thanks to the platform from Microsoft, the implementation of voice control was extremely easy:

```
switch (voice_command)
{
case 0: call_voice_command_0; break;
case 1: call_voice_command_1; break;
case 2: call_voice_command_2; break;
...
case n: call_voice_command_n; break;
}
```

The proposed algorithm of voice control is quite simple:  
 1. User should say a voice command (for example, "turn on light in room 1").  
 2. The resulting voice command will be recognized by Microsoft Speech Engine and will be converted into a string.  
 3. Each recognized voice command has a corresponding numerical value. If the command is identified, the corresponding digital value will be transmitted to the hardware side using Bluetooth HC-06 module.  
 4. On each cycle microcontroller waits for the voice command. If microcontroller receives the known digital value, this command will be sent to the corresponding sensor.

If we want to get information from the sensors (for example we want to receive current temperature), the 5 and 6 steps should be added:

5. Send a response line in the form of "accepted command - result". (For example, "7 - 24" means "current temperature is 24 degree", where 7 is the digital representation of "current temperature" command).

6. The GUI will get this value and decode it. After that, the GUI will be updated, and the result will be pronounced using the same Microsoft Speech Engine.

Voice control is used to turn on/off the lighting in each room, turn on/off sleep mode, alarm clock, security mode, provide temperature and humidity control in the house, turn on/off music etc. In total it has been created 25 phrases for recognition.

## VII. RESEARCH ANALYSIS AND RESULTS

This section was designed to evaluate the voice control for the intelligent building prototype.

This experiment was created to demonstrate how voice control works in Smart House system. In this experiment, the author collected the results from 10 participants (men, women of different ages and 1 kid). The participants of the experiment were asked to control the created prototype with the help of the voice. Each participant said a minimum of 5 commands to control the prototype. Phrases that were used for recognition - "turn on the music", "turn on/off light in room 1/room 2/room 3/room 4/room 5", "set the alarm clock", "turn on/off security mode", "change lighting colour", "change temperature", "turn on/off fan", "say current time", "set time to sleep", "say current humidity". Phrases were recognized with an accuracy of 81%. Problems with the recognition of phrases occurred when a phrase was said too quickly or not clearly.

The latency with and without voice commands is shown in the table 3.

**Table- III: . Speech recognition latency comparison with and without the voice control.**

Latency Without Speech Engine(s)	Latency With Speech Engine (s)
0.1 +/- 0.05	3.1 +/- 2.3

As we can see from table 3, the delay can occur with voice control. The delay occurs in two "places" - in the process of transmitting a message from the interface via Bluetooth to the Arduino and then to the sensors. And the second place is the delay in voice recognition.

However, the use of voice recognition is a major step towards improving the Smart House system because it gives the support for the people with disability, it gives an ability to control the system without using additional devices, thus, it is a more natural approach for humans to communicate with the intellectual system.

The experiments have shown that voice control of the house when the washing machine, kettle, coffee maker and microwave oven were turned on, the recognition performed extremely well, which is a good sign that this approach is suitable for Smart Home control.



## Design and implementation of a Smart House system with voice control

To implements the Smart House system, the following technologies have been used.: Arduino Mega with ATMega2560 microcontroller and Arduino Uno with ATMega 328P microcontroller, C/C++ languages have been used for write programs for microcontrollers, DHT11 Temperature & Humidity Sensor, 4-digit LED module, DS3231 Clock Module, RC522 RFID module, RGB strip, active buzzer, 30x30x10 fan, Bluetooth HC06, 5 HC-SP04 ultrasonic sensors, analog gas sensor, 4 LED, colour sensor, RGB LED as sensors. UI was written on WinForms technology with the help of C# language. Microsoft Speech Platform was used for voice control.

Figure 6 shows the prototype, which was the result of the development.



**Fig. 6. The Smart House prototype**

## VIII. CONCLUSIONS

As a result of this research, the working prototype of the Smart House system has been created in which the main features of the following subsystems have been implemented: security system, climate control system, power supply system, communication system, entertainment system, graphic interface. In addition, the voice control was provided to control the intelligent system. The strategy for voice control was briefly presented in this article and voice control was evaluated by the sets of experiments. Experiments have shown that voice control is an important step in the further development of such systems, as well as a great help for people with special needs and disabilities.

## ACKNOWLEDGMENT

The author would like to show sincere gratitude to Zabobonin Sergiy for his knowledge and tips on writing this manuscript.

Also a great thanks to all those who took part in the experiments for your support and enthusiasm.

## REFERENCES

1. Baigozin D.V. , Pervukhin D.N., Zakharova G.B. "The development of principles of intellectual management of engineering equipment in the «Smart house» system." In Bulletin of the Tomsk Polytechnic University. 2008, v. 313, № 5. pp 168–172
2. Mr.Rohit Kadam, Mr.Pranav Mahamuni, Mr.Yash Parikh "Smart Home System" in International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163 Volume 2 Issue 1 (January 2015)
3. FIRDOUS KAUSAR, EISA AL EISA, IMAM BAKHSH "INTELLIGENT HOME MONITORING USING RSSI IN WIRELESS SENSOR NETWORKS" in International Journal of Computer Networks & Communications (IJCNC) Vol.4, No.6, November 2012
4. Y. Rahal, H. Pigo t, and P. Mabilleau, "Location Estimation in a Smart Home: System Implementation and Evaluation Using Experimental Data", in International Journal of Telemedicine and Applications, Vol. 2008.
5. Shiu Kumar, "UBIQUITOUS SMART HOME SYSTEM USING ANDROID APPLICATION", in International Journal of Computer Networks & Communications (IJCNC) Vol.6, No.1, January 2014
6. R. Piyare and M. Tazil, "Bluetooth Based Home Automation System Using Cell phone," in IEEE 15th International Symposium on Consumer Electronics, Singapore 2011, pp. 192 - 195.
7. D. Javale, M. Mohsin, S. Nandanwar, and M. Shingate, "Home Automation and Security System Using Android ADK," in International Journal of Electronics Communication and Computer Technology" (IJECCCT), vol. 3, pp. 382-385, March 2013.
8. P. Hamernik, P. Tanuska, Member, IACSIT, and D. Mudroncik, "Classification of Functions in Smart Home" in International Journal of Information and Education Technology, Vol. 2, No. 2, April 2012
9. Basil Hamed, "Design & Implementation of Smart House Control Using LabVIEW" in International Journal of Soft Computing and Engineering (JSCE) ISSN: 2231-2307, Volume-1, Issue-6, January 2012
10. Alam M.R., Reaz M.B.I., Ali M.A.M. A review of smart homes-Past, present, and future. IEEE Trans. Syst. Man Cybern. Part C. 2012;42:1190–1203. doi: 10.1109/TSMCC.2012.2189204.
11. Mingfu Li, Hung-Ju Lin, "Design and Implementation of Smart Home Control Systems Based on Wireless Sensor Networks and Power Line Communications" in IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, July 2015
12. Aigerim Bakatkaliyevna Altayeva, Batyrkhan Sultanovich Omarov, and Young Im Cho, "Intelligent Microclimate Control System Based on IoT" in International Journal of Fuzzy Logic and Intelligent Systems Vol. 16, No. 4, December 2016, pp. 254-261 <http://dx.doi.org/10.5391/IJFIS.2016.16.4.254>
13. Md. Wahidur Rahman, Md. Harun-Ar-Rashid, Rahabul Islam, Dr. Mohammad Motiur Rahman, "Embodyment of IOT based Smart Home Security System" in International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887 Volume 6 Issue IX, Sep 2018 Available at [www.ijraset.com](http://www.ijraset.com)
14. Bilal Mubdir, Asaad Al-Hindawi, "Design of Smart Home Energy Management System for Saving Energy", in European Scientific Journal November 2016 edition vol.12, No.33 ISSN: 1857 – 7881 (Print) e - ISSN 1857-7431
15. Tiago D. P. Mendes, Radu Godina, Eduardo M. G. Rodrigues, João C. O. Matias and João P. S. Catalão, "Smart Home Communication Technologies and Applications: Wireless Protocol Assessment for Home Area Network Resources" in Article in Energies · July 2015
16. M. Kuzlu, M. Pipattanasomporn , S. Rahman, "Review of communication technologies for smart homes/building applications" in 2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)
17. Jiwei Yuan, Jiakang Liu, Heng Luo, Qiuhua Guo, Wenxiu Wu, "Design and implementation of smart home remote control system based on SNS open platform" in 5th IET International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN 2013)
18. Meensika Sripan, Xuanxia Lin, Ponchan Petchlorlean and Mahasak Ketcham, "Research and Thinking of Smart Home Technology" in International Conference on Systems and Electronic Engineering (ICSEE'2012) December 18-19, 2012 Phuket (Thailand), pp 61-63
19. Tony DiCola. Embedded Linux Board Comparison
20. Mauro C. Balasubramaniyan\* and D. Manivannan, IoT Enabled Air Quality Monitoring System (AQMS) using Raspberry Pi, Indian Journal of Science and Technology, Vol 9(39), DOI:10.17485/ijst/2016/v9i39/90414, October 2016.
21. A. ElShafee and K. A. Hamed, "Design and Implementation of a WiFi Based Home Automation System," in World Academy of Science, Engineering and Technology, vol. 68, pp. 2177-2180, 2012.

22. R. D. Caytiles and B. Park, "Mobile IP-Based Architecture for Smart Homes," International Journal of Smart Home, vol. 6, pp. 29-36, 2012
23. Davies E. I. & Anireh, V.I.E. "Design and Implementation of Smart Home System Using Internet of Things", vol. 7. No. 1, March 2019, pp 33-42

### AUTHORS PROFILE



**Nataliia Konkina** Pursuing Master Degree at National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". Her interested areas are IOT, artificial intelligence, 3D computer graphics and modeling, microcontrollers, operating systems and algorithms.