

Remote Control Android-based Applications for a Home Automation implemented with Arduino Mega 2560 and ESP32

Robert Susany¹, Raul Rotar²

¹“Politehnica” University of Timisoara, Advanced Computing Systems and Architectures (ACSA) Laboratory, Computers and Information Technology Department, 2 V. Parvan Blvd, 300223 Timisoara, Romania
E-Mail: robert.susany@student.upt.ro

²“Politehnica” University of Timisoara, Advanced Computing Systems and Architectures (ACSA) Laboratory, Computers and Information Technology Department, 2 V. Parvan Blvd, 300223 Timisoara, Romania
E-Mail: raul.rotar@student.upt.ro

Abstract. As technology becomes cheaper and more reliable, people are taking advantage of it to improve their standard of living. This paper describes how we can integrate some of the available technology into our daily lives to create a safe and comfortable living environment. The provision of high-speed Internet makes it easy to access and monitor various types of devices in real-time from a desktop, a tablet or any device that has the capability to do so. With the help of this framework, all smart home sensors and devices can be tailored to the user's personal needs, and some of the parameters can even be observed and controlled using a mobile phone. There are a range of home automation systems on the market, but they are either generic or, if customizable, costly. The goal of this project was to develop secure and affordable home applications and to examine how some of the applications can be operated from very large distances using Wi-Fi and smartphone technology.

Keywords. Arduino Mega 2560, ESP32, Sensors, Wireless Communication, Remote Control

1. Introduction

The Internet of Things (IoT) is becoming more and more popular today, with the main advantage that it can gather a huge amount of data and use it in different ways depending on the needs of a particular user. IoT is an extension of the internet, mainly used to control or monitor remote devices and other everyday objects [1]. Home Automation is an area where the Internet of Things is commonly used to control different appliances and to monitor certain devices, to send all that data to the user, with the help of the Internet and other Wi-Fi applications [2].

Nowadays home automation systems are mostly used in office buildings to reduce the amount of money spent on utilities and to provide greater comfort for employees, which will help them to perform better at work, giving better results. In developed countries, many homes have some kind of home automation depending on the budget each household has and is willing to invest in their comfort. In the near future, more and more people will see the benefits of new advances in different

technologies that are becoming more environmentally friendly and will offer them an economic incentive to start implementing them.

This paper proposes the implementation of various sensors and devices connected to the Arduino Mega 2560 [3] development board to monitor the house and the use of the ESP 32 development board [4], to monitor and control other devices from a distance using Wi-Fi and an application installed on a smartphone, called Blynk [5]. The proposed automation system represents a low-cost implementation [6] which can be used locally with different Android-based smartphones via Wi-Fi or can be operated remotely via different IoT platforms[7], giving users the power to manage all devices with a minimum of effort.

The paper is organized as follows. Section II presents the related work regarding Home Automation applications. Section III details the hardware components of the proposed home automation system, which is implemented using the Arduino Mega 2560 and ESP 32 development boards. Further details of the developed software applications are provided in Section IV of this paper. Finally, Section V presents conclusions and future work.

2. Related Work

In this segment of the paper, we will compare other home automation projects in order to have a better understanding of what people have done so far in this field.

In paper [8], the authors use a solar panel that rotates with the sun using a servomotor to power the Arduino board, a Bluetooth module for sending commands to a relay that switches on or off a light bulb, and these commands are sent from a cell phone. While this project aims to use energy from the sun to control the circuit board, an external battery is also required to store the energy obtained from the photovoltaic panel when more than is necessary is produced and to store energy during the periods when the sun is unable to provide the sufficient amount of energy to power the circuits, such as during the night time.

Voice activation is used in a variety of things, such as smartphones, cars and home automation, to give certain commands that control the devices according to the needs of users. In [9] a voice control system is used for home automation. In this system, an Arduino Uno board is used to connect two sensors to track the environment, a DHT 11 digital temperature and humidity sensor and a Light Dependence (LDR) sensor. A few LEDs are used to mimic various actuators. Another board is used to connect Arduino Uno to the internet, the board is called the Wiznet W5100 Ethernet Board, which is an Ethernet shield that goes to the top of the Arduino Uno.

For device voice control, BitVoicer is used as a voice control program that is compliant with Arduino. Throughway of BitVoicer, such commands can be sent to the Arduino board and the functionality of the device can be managed. The system uses the Ethernet Shield to store data sent by the sensors in the database and to send commands to different actuators depending on the data received when the system is being used in automatic mode. In manual mode, the actuators (light, fan, etc.) can be controlled using voice commands sent to the computer by the user and, with the help of BitVoicer, these commands are sent to Arduino to control the actuators. This home automation system relies heavily on voice commands, which is a very easy way to control different objects, the trouble is when you want to operate the device without using your voice, since different health problems may arise, you can't send instructions to the machine unless you can open the computer and make the necessary changes to the home automation system software.

With the current technological advances and the constant need to make everything around us more efficient, especially when it comes to the use of different kinds of important resources, such as water or energy, the authors in [10] are proposing smart home automation and a metering system using the Internet of Things. In this paper, researchers have built a website through which that they can measure energy consumption and issue bills, and they can control different devices, such as a light or fan, to turn them on or off remotely using a computer, a tablet or a smartphone. The hardware used for their

project is the Arduino Pro Mini, to control the equipment, the Wi-Fi module (ESP8266 Wi-Fi chip), to connect the microcontroller to the internet, a few relays and an LCD will display the device state (ON / OFF), the date, time and temperature in real-time. All data displayed on the screen will also be available on the website so that the customer can equate the data from the database with the data from the devices and ensure that there are no inconsistencies and that no changes have happened.

This application shows the many benefits that a smart home, combining different technologies, can offer to the customer, such as improved convenience, remote access and control of different devices inside the home, and smart meter reading and billing capability. In the future, all the data collected from this program can be used to better optimize the use of the most necessary and expensive services and to help users make substantial changes for their budget that they can use for different purposes.

3. Hardware components of the Home Automation System

The hardware implementation of the Home Automation system is constructed around two development boards, namely the Arduino Mega 2560 microcontroller and the ESP 32. In this section, we will detail the hardware components that are connected to each of the listed devices.

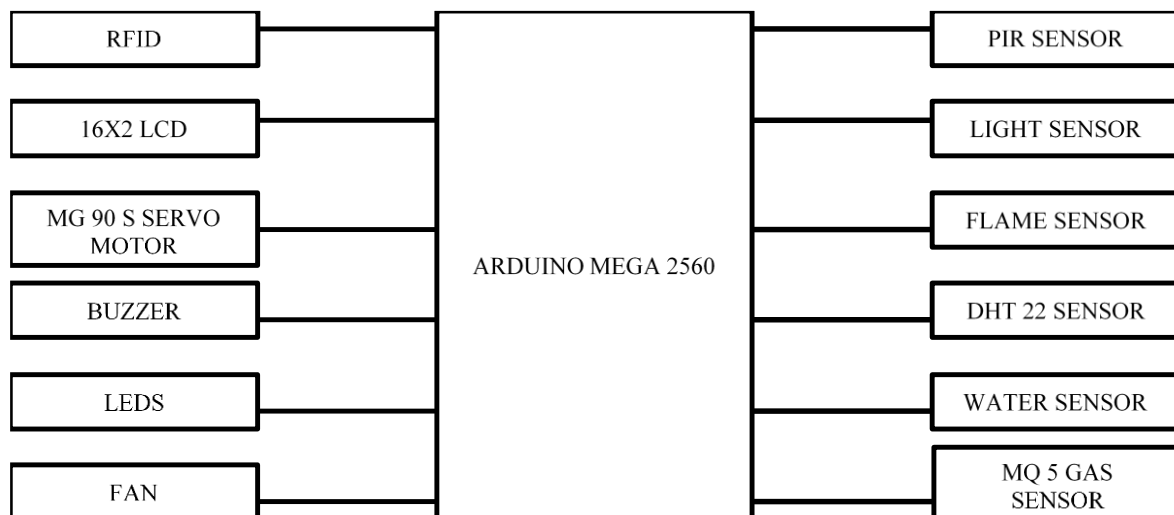


Figure 1. Block diagram of the home automation system using Arduino Mega 2560.

3.1. Arduino Mega 2560

According to the Block diagram which is depicted in Fig. 1, a set of sensors, RFID, LEDs, display and a fan are linked to the Arduino Mega 2560 mainboard. Each of the devices will be briefly introduced in the following lines:

- **Arduino Mega 2560:** Represents the microcontroller used to control, receive and send data to all the components used in Fig.1.
- **RFID:** The radio-frequency identification system is used in this system to verify the validity of the access cards used by the user.
- **16X2 LCD:** The liquid crystal display is used to display a welcoming message and to show the identification number of each tag.
- **MG 90 S SERVO MOTOR:** This servomotor is used to automatically open and close the door if a valid tag with access rights is presented to the RFID.
- **BUZZER:** This buzzer is used to alert the people in the house if there is a fire, a flood, or a gas leakage.

- LEDs: The installed LEDs are used to illuminate the house.
- FAN: This fan is used to reduce the temperature inside the house.
- MQ5 GAS SENSOR: The gas sensor is used to detect any gas leakage.
- WATER SENSOR: This water sensor is used to detect if there is a flood in the bathroom.
- DHT 22 SENSOR: This sensor is a temperature and humidity sensor used to monitor the temperature and humidity inside the house.
- FLAME SENSOR: This flame sensor is used to detect a fire inside the house.
- LIGHT SENSOR: The light sensor is used to detect the level of light inside the house.
- PIR SENSOR: This sensor is used to detect movement inside the house.

3.2. ESP 32 Development Board

The second set of equipment is placed around the ESP 32 Development Board and consists of a sensor, a servomotor, and a fan, as can be seen in Fig. 2. Further details of the used components will be listed as follows:

- ESP 32: This is the microprocessor used to send and receive data to and from devices.
- DHT 22 SENSOR: This sensor is a temperature and humidity sensor used to monitor the temperature and humidity inside the house.
- FAN: This fan will lower the temperature inside the house to a programmed value.
- MG 90 S SERVOMOTOR: This servomotor is used to open and close the garage door.

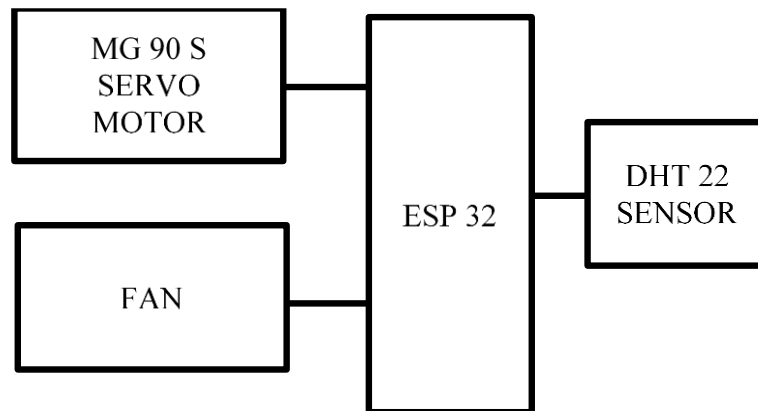


Figure 2. Block diagram of the home automation system using ESP 32.

The Wi-Fi network consists of two-way communications as shown in Fig.3. The first choice is to control the servo motor and the fan with the assistance of the phone using the Blynk app, and the second option is that the data sent to the server via the ESP 32 from the DHT 22 sensor enables the user to gain knowledge of the temperature and humidity inside the house and also the status of the garage door, that is mirrored on the phone in real-time.

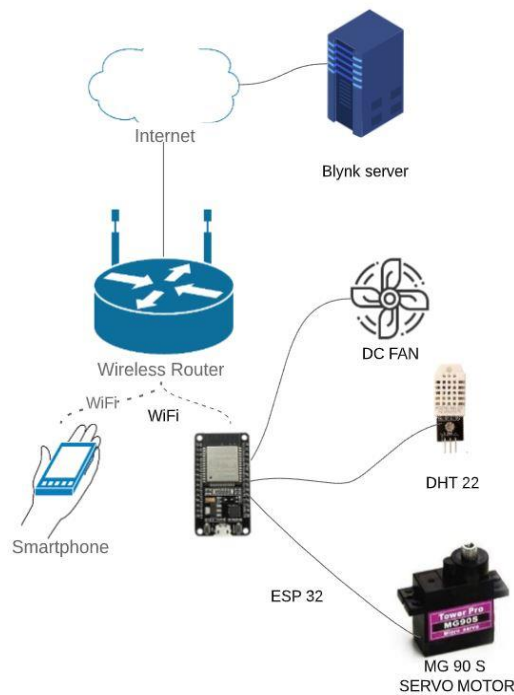


Figure 3. Network Architecture for ESP 32.

4. Description of the Software Applications

The developed software applications are linked to the previously enumerated hardware components (sensor values). In Fig. 4, the software diagram used for Arduino Mega and all devices connected to it is shown to have a better understanding of how everything is incorporated into this system.

4.1. Application for gas detection in a room

If there is a gas leak inside the room where the MQ5 sensor has been installed and a certain gas concentration limit introduced by me in the Arduino program has been exceeded, the buzzer will ring, alerting all the people in the house that there is a gas leak and that they should stop the gas as soon as possible, to avoid further problems. The alarm will ring until the gas concentration limit is lower than the limit entered by me in the program.

4.2. Application for detecting a flood in the bathroom

If there is a flood in the bathroom, the water sensor installed on the floor will detect that, due to the difference in voltage caused by the water falling on the sensor, if the limit inserted by me in the program is exceeded, the buzzer will ring, alerting the people in the house that there is a flood and that they should act as soon as possible to close the water. The alarm stops only if the sensor is dry and there is no water on the surface of the sensor.

4.3. Application for maintaining a constant room temperature

The DHT 22 sensor monitors the temperature and humidity inside the house, I set the maximum temperature limit within the Arduino program, and the maximum temperature limit is 24 degrees Celsius. When this limit is exceeded, the ventilator installed inside the house will turn on and run until the temperature inside the house is below 24 degrees Celsius, maintaining constant room temperature.

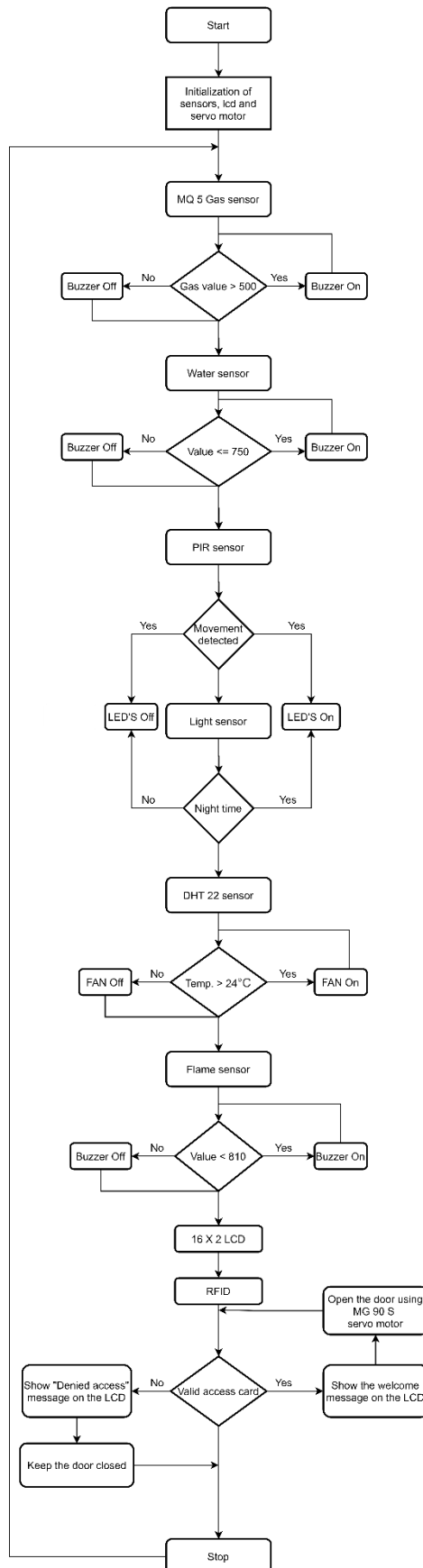


Figure 4. Flowchart of mobile-based software applications.

4.4. Application for fire detection in a room

If there is a fire inside the house, the flame sensor installed will detect this by measuring the difference in infrared light inside the room. After testing the sensor and inserting a detection limit in the Arduino program, if that limit changes, the buzzer rings alerting the people in the house that there is a fire and they should act as soon as possible to stop the fire. The buzzer rings until the fire is extinguished and the values of the sensor are within the programmed limits.

4.5. Application for turning on the light automatically

If a person enters the house at night and all the lights are off, the PIR sensor will detect the movement, the light sensor will detect if it is day or night. If it is night and movement is detected inside the house all the LEDs installed inside the house will turn on automatically, illuminating the interior of the house. The lights would not turn on automatically during the day, only during the night.

4.6. Application for Home Access using an RFID module

When a person approaches a valid RFID card to the card reader at a maximum distance of two centimeters, it will check to see if the access card is valid, and if it is then the code of that card and a welcoming message will appear on the LCD screen mounted above the card reader, then the entrance door will open automatically allowing the person to enter the house, and after a few seconds, it will close. If a person approaches a compatible access card that does not have access rights, the code of the access card and a message will appear on the screen and the door will not open.

4.7. Application for displaying the temperature and humidity on mobile phone

To use this application we need to connect the ESP 32 development board and our smartphone to the internet. After that, the DHT 22 sensor connected to ESP 32 will start sending temperature and humidity to the board. We need to have installed Blynk application [5] on our mobile phone after we have that installed and we have made all the required settings, we will receive the temperature and humidity to our phone in real-time as long as the smartphone and the ESP 32 are connected to the internet. The data displayed on the mobile phone will be updated every two seconds.

4.8. Application for garage door control using a mobile phone

To use this application we need to have a Blynk application installed on our smartphone. After the servomotor is connected to ESP 32, the smartphone, the development board are connected to the internet, and all the settings have been made, we have created a virtual slider button in the Blynk app, to open and close the garage door. When we use the application, if we slide that button to the right the garage door will open, and if we slide that button to the left the garage door will close, and we can see in real-time the degree of opening of the garage door. This application is very useful if we want to enter or exit the garage without having to get out of the car.

4.9. Application for fan control using a mobile phone

In order to use this application, we have downloaded the Blynk application on my mobile phone and we have made all the settings needed in the application to create a virtual button to turn on and off the fan. After the fan is connected to ESP 32, we have connected the development board and the phone to the internet. If we press the virtual button, we can turn on and off the fan, and by changing the color of that button, we can visualize in real-time if the fan is on or off. This application combined with the application for displaying the temperature and humidity on a mobile phone is very useful to maintain a certain temperature in a certain place and to monitor the temperature from a distance.

5. Conclusions and Future Work

Developing this type of smart home we have been able to demonstrate that it is possible to build such a device, using cheap sensors and developing boards such as Arduino Mega 2560 and ESP 32 and that they can be combined to provide the best solution for every home. All applications presented are functional and operate as planned, with the power provided to the sensors and the development boards coming from the power bank. We were also able to show how we can build a system that covers many areas in a smart home such as constant monitoring, security, and control. We hope that more and more people will be able to take advantage of the new technologies available and to use them to improve their standard of living. For future improvements, we have planned to do the following changes:

- To receive a fire alarm notification on the smartphone
- To add a security notification on the smartphone using Blynk application and a magnetic switch
- To change the screen with an ePaper display in order to reduce power consumption
- To charge our power bank during off-peak hours using a timer delay.

References

- [1] https://en.wikipedia.org/wiki/Internet_of_things
- [2] P. Siva Nagendra Reddy; K. Tharun Kumar Reddy; P. Ajay Kumar Reddy; G. N. Kodanda Ramaiah; S. Nanda Kishor, "An IoT based home automation using android application", *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPE5)*, 2016
- [3] <https://www.arduino.cc/>
- [4] <https://www.espressif.com/en/products/hardware/esp32/overview>
- [5] <https://blynk.io/>
- [6] S. L. S. Sri Harsha; S Chakrapani Reddy; S Prince Mary, "Enhanced home automation system using Internet of Things", *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2017.
- [7] Waheb A. Jabbar; Mohammed Hayyan Alsibai; Nur Syaira S. Amran; Samiah K. Mahayadin, "Design and Implementation of IoT-Based Automation System for Smart Home", *2018 International Symposium on Networks, Computers and Communications (ISNCC)*, 2018.
- [8] Apurva Singh; Harsh Mehta; Anuj Nawal; O.V. Gnana Swathika, "Arduino Based Home Automation Control Powered by Photovoltaic Cells", *2018 Second International Conference on Computing Methodologies and Communication (ICCMC)*, 2018.
- [9] Ishan Krishna; K. Lavanya, "Intelligent Home Automation System using BitVoicer", *2017 11th International Conference on Intelligent Systems and Control (ISCO)*, 2017.
- [10] Sadi Mahmud; Safayet Ahmed; Kawshik Shikder, "A Smart Home Automation and Metering System using Internet of Things (IoT)", *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, 2019.