ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

HOME AUTOMATION SYSTEM USING ARDUINO UNO

Siddharth Gupta [*1], Shivaditya Jatar [2] & Ayush Mittal[3]
[*1,2&3] Computer Engineering, Shri Govindram Seksaria Institute of Technology and Science, Indore (M.P.), India

DOI: 10.5281/zenodo.1012557

ABSTRACT

Some people are always on the move from place to place due to business demands. They may have to spend a couple of days away from their home leaving all their household appliances without any kind of monitoring and control. Some devices are supposed to be left into the power sockets but there are many devices which need not be plugged in all the time. This may require an individual to manually attend to each of the devices independently from time to time. All such monitoring and control can be done without necessarily being around or inside the home. Therefore an Internet based home automation system to manage appliances remotely from anywhere, anytime is needed. Home automation has been implemented in a variety of ways, some of which include: Bluetooth, GSM, Wi-Fi, ZigBee. Though all of them are prevalent, but there are constraints regarding each of them. The significant constraints are related to range of operability, additional costs, delay in receiving commands and security. In order to overcome these constraints a system should provide a user friendly and easy way to give commands which can be executed immediately from anywhere in the world, given that there is an internet connection. This system should not be confined in terms of range. The proposed system is a distributed home automation system which consists of a microcontroller based hardware side and a website which forms the software side of the project. The system will be accessible from remote locations via web interface. The microcontroller will be interfaced with a GPRS module to bring it online and will have various relays and sensors to achieve the objective.

KEYWORDS: Arduino Uno, GSM, GPRS, Django framework

I. INTRODUCTION

Home automation or smart home is an extension of building automation and involves the control and automation of lighting, heating, ventilation, air conditioning (HVAC), appliances, and security. Modern systems generally consist of switches and sensors connected to a central hub sometimes called a "gateway" from which the system is controlled with a user interface that is interacted either with a wall-mounted terminal, mobile phone software, tablet computer or a web interface. Home Automation is a way of controlling electrical appliances from a distance. Commands are sent from one location and received and acted upon in another location. Home automation is a step towards what is referred to as the "Internet of Things," in which everything has an assigned IP address, and can be monitored and accessed remotely. IoT refers to the networked interconnection of everyday objects, which are often equipped with ubiquitous intelligence. IoT will increase the ubiquity of the Internet by integrating every object for interaction via embedded systems, which leads to a highly distributed network of devices communicating with human beings as well as other devices. Due to rapid advances in underlying technologies, IoT is opening tremendous opportunities for a large number of novel applications that can help to improve the quality of living of humans. Rapid advances have been made with regards to home automation by IoT.[1]

II. Tools and Technology

1. Arduino Uno Microcontroller

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. [3]

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

2. GPRS module

A GPRS module is a chip or circuit that will be used to establish communication between a mobile device or a computing machine and a GPRS system.

3. Sensors

There are a number of sensors which are required for the project. They are listed below:

PIR sensor: Used to detect presence in room, stands for passive infrared. Its range is upto 10 meters at an angle of +15 to -15 degrees.

Temperature Sensor (LM 35): To detect room temperature. Its range is from -550C to 1500C.

Smoke detector sensor: To detect smoke. It can detect LPG, i-butane, propane, methane, alcohol, hydrogen and smoke.

4. Relays

They are used to switch appliances on/off.

5. Buzzer

The buzzer is used to sound off the alarm and it works when LOW signal is provided.

6. Software Components

Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. [11]

Django Framework

Django is a free and open-source web framework, written in Python, which follows the model-view-template (MVT) architectural pattern. Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and "pluggability" of components and provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models. [9]

EAGLE

It is a scriptable electronic design automation application with schematic capture, printed circuit board layout, auto-router and computer-aided manufacturing features. EAGLE stands for Easily Applicable Graphical Layout Editor and is developed by CadSoft Computer GmbH. [10]

III. Organization of the System

The fundamental organization of the system, embodied in its components, their relationships to each other and to the environment, and the principles governing its design and evolution are described.

The system architecture of the home automation system consists of two main parts namely the home system and the client side user interface.

Hardware Side

The first component in the home system is the PCB to which all the sensors, devices and the GPRS module is attached. The GPRS module handles all the commands sent by the user and forwards them to the microcontroller. It is connected to the internet via a SIM.

The connection with the sensors can be wireless as well. It can be implemented by using RF transceiver. On a small scale, however, the devices and sensors are connected with the board through wires except one device.

Client Side User Interface

The client can access the home system by logging in a website which can be accessed on any internet browser. The client sends a request and gets a reply after the request is processed. The client can visit the website http://www.iotdemoo.herokuapp.com, choose appropriate mode and give various commands accordingly.

The main circuit consists of different sensors, devices, power supply, the Arduino UNO board and a SIM800 module.

A brief description of the components is given below. They can be broadly divided into two categories: Hardware side and Client side.

Hardware Side

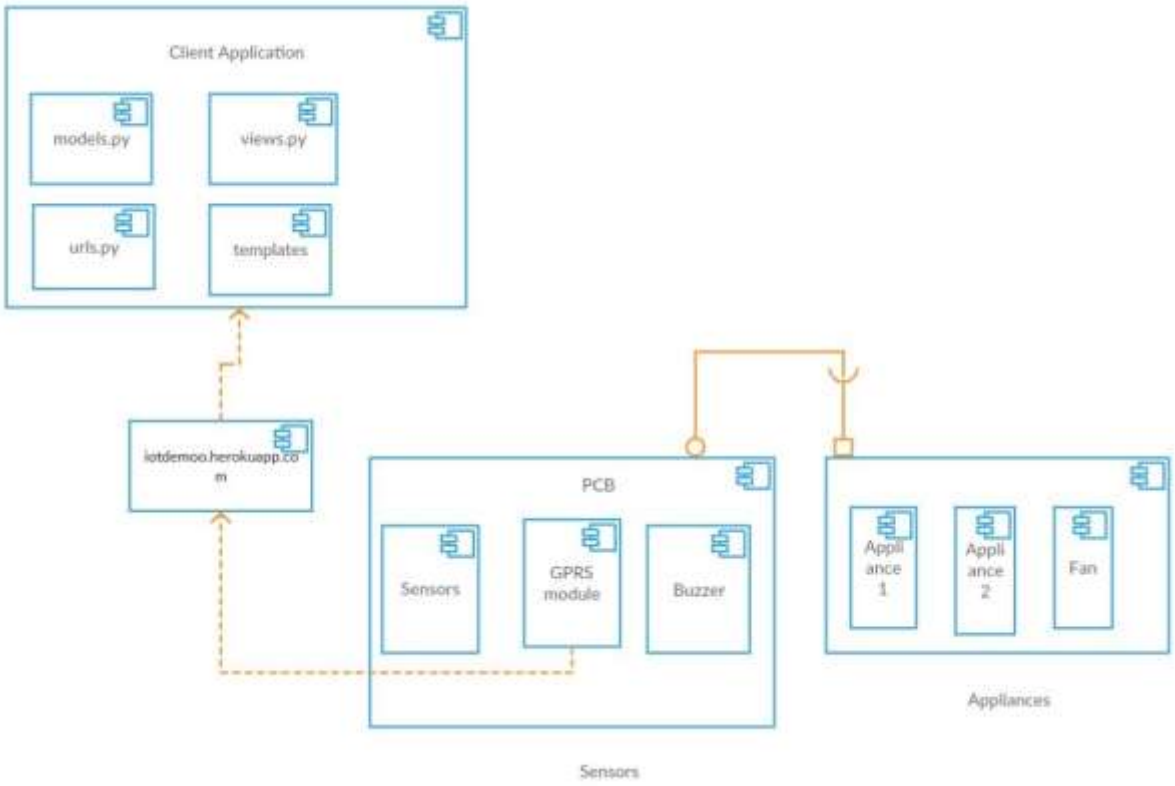
All the electrical appliances need power to operate. The modules either require 5V or 12V DC. To convert the 230V AC into DC, a power supply is used. There are two bulbs and a fan used in the prototype. They are connected through relays. The sensors include: PIR sensor, gas sensor and a temperature sensor. A buzzer is also present.

The microcontroller handles all the commands provided by the user and ensures that they are executed. It interacts to user through a website. The internet connection is established through the SIM800 module.

Client Side

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

It is implemented in Django framework. The different files which are used include: views.py, urls.py, models.py and templates.



IV. IMPLEMENTATION AND TESTING

1. Implementation

There are three main files in the code: models.py, urls.py, views.py. The design of the web page is in the templates folder which contains all the html files.

models.py

A model is the single, definitive source of information about the data. Each attribute of the model represents a data field. The attributes used in this project are status of the devices (d1, d2, d3) and the sensor readings (temp,gas,pir).

views.py

A view function, or *view* for short, is simply a Python function that takes a Web request and returns a Web response. This file checks the parameters of the model and sends a particular web page as a response. The functions in the file are:

- home_automation_status: This displays the status of the devices and the sensor readings on the web page.
- home_light_on(request) and home_light_off(request): These functions receive the request and change the device status on the website to red if off and green when on. fan_speed(request): This function checks the reques and displays the fan speed accordingly from a range of 1 to 5.
- home_next(request): This function sends the string with the status of all devices as a response to the hardware side code of the project.
- normal_mode(request) and security_mode(request): The mode is looked up in the request url and the corresponding page is displayed in response.

urls.py

This file is pure Python code and is a simple mapping between URL patterns (simple regular expressions) to Python functions (the views). Whenever the user requests a page from the site, this file is important in determining which python code to execute. Django runs through each URL pattern in the file, in order, and stops at the first on that matches the requested URL.

Templates

This folder contains the designs of all the web pages in the project which are coded in HTML and CSS. The included pages are:

- login.html: Displays the login page.
- invalid_login.html: Displays the page when login credentials are wrong.
- normal_mode.html: Displays the normal mode page.
- security_mode.html: Displays the security mode page.
- home_autoreload.html: Reloads the page in fixed interval of time so that the changes may be displayed, if any.

Hardware Side Code

All the functions used in the home automation hardware code are described in this part. First of all, the variables used for sensors, LCD, devices, buzzer and GPRS module are declared. Pin number for each accessory is then mentioned. The Arduino has two functions which are included in each code. They are: void setup( ) and void loop( ). The setup function is run only once. The loop function is run repeatedly. Other functions are added according to use.

void setup( )

The pins are set to output/input mode. The default off condition is written to the devices through digitalWrite. The GPRS module and connection to the website are initiated.

void loop( )

According to the data read from the url different commands are given to the devices. The data sent by the url consist of 6 bits. The first bit represents the status of device 1. The second bit represents the status of device 2. The third bit represents the status of fan. The fourth corresponds to the fan mode i.e, automatic or manual. Similarly, the fifth does to bulb mode (automatic/manual). The sixth bit tells about the normal or security mode.

Bit No.	Bit Value = 0	Bit Value = 1
1	Device 1 OFF	Device 1 ON
2	Device 2 OFF	Device 2 ON
3	Fan OFF	Fan ON
4	Automatic Mode Fan	Manual Mode Fan
5	Automatic Mode Bulb	Manual Mode Bulb
6	Normal Mode	Security mode

Description of String Sent by the URL

These bits are read from the string and accordingly the state of the appliances is changed. The data read from sensors is also updated in the appropriate variables.

void SubmitHttpRequest( )

This function uses various AT commands to establish connection to the home automation website. It first checks the signal strength of the sim attached and then initialises the SIM800 module in GPRS mode. The APN is set according to the service provider. The mode is then changed to http and a http session is initialised. Finally, the connection is established after the url is specified.

void send_data_url( )

The data gathered from the sensors is sent to the url using AT commands by this function.

void ShowSerialData( )

The replies or the returned values after executing the AT commands are displayed on the serial monitor using this function.

void ShowSerialData1( )

The string sent by the url is read and stored in an array and then displayed on the serial monitor.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

2. Testing

The exchange of commands and status between the website and the hardware is done via a string. The string is which is received from the website is: **"*d1 d2 speed auto_fan auto_light mode#"**.

Here the HTTPACTION indicates success or failure of connection to the web page.

2.1. Testing Scenarios for Normal Mode

All the strings in this mode have 0 in place of the mode character. Therefore all the following actions are performed in the normal mode.

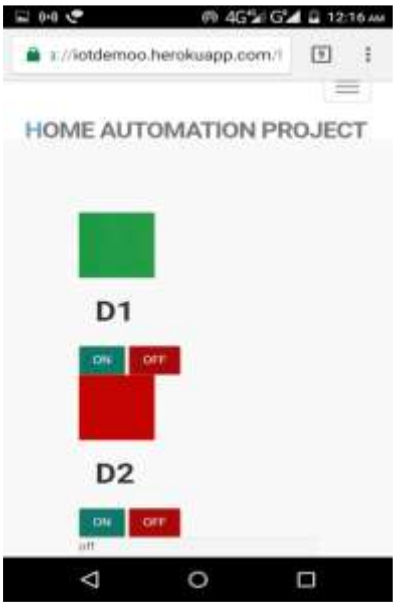
2.1.1 Switching device 1

The string received in this case has 1 at device 1's place; this indicates that the signal is received to switch on device 1. Therefore that action is performed.

```
+HTTPACTION: 0,200,21
AT+HTTPPARA="URL","http://iotdemoo.herokuapp.com/home_automatioAT+HTTPACTION=0
OK

+HTTPACTION: 0,200,9
*100000#
OK
*100000#
OK
normal modemanual fanlight manualDEVICE On
BULB OFF
fan Off
```

Test Device 1



Device 1 on

2.2 Testing Scenarios for Security Mode

All the strings in this mode have 1 in place of the mode character. Therefore all the following actions are performed in the security mode.

2.2.1 Intrusion Alert

Whenever the sensor detects intrusion, there is an alert sign on the website in the security mode.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

```
+HTTPACTION: 0,200,21
AT+HTTPPARA="URL","http://iotdemoo.herokuapp.com/home_automatioAT+HTTPACTION=0
OK

+HTTPACTION: 0,200,9
*000001#
OK
*000001#
OK
security modeAT+HTTPPARA="URL","http://iotdemoo.herokuapp.com/home_automatioAT+HTTPACTION=0
OK
```

Test Security Mode

Display Alert on Intrusion detection

V. CONCLUSION

In this project an approach for web based remote home automation has been presented. The features of the project are: Remotely controlling the electrical appliances at home, Intrusion detection, Gas leakage detection, Control speed of fan automatically according to the temperature, Switch lights on/off automatically by detecting the presence of a person in the room.

Following are the advantages of the proposed approach: The hardware part of this approach is cost effective and low maintenance, It overcomes the disadvantages of solutions using other communication technologies by using the internet, It is user friendly.

Following are the limitations of the proposed approach : The user has to be connected to the internet to interact with the system, The range of the sensors is limited to a single room, Communication cost between SIM module and internet is high, There is a delay of few seconds between the request and result.

VI. REFERENCES

1. Baris Yuksekkaya, A. Alper Kayalar, M. Bilgehan Tosun, M. Kaan Ozcan, and Ali Ziya Alkar "A GSM, Internet and Speech Controlled Wireless Interactive Home Automation System", 2006, IEEE Transactions on Consumer Electronics, Vol. 52(3), pp. 837 – 843.
2. Rozita Teymourzadeh,Salah Addin Ahmed,Kok Wai Chan and Mok Vee Hoong , "Smart GSM Based Home Automation System", 2013, IEEE Conference on Systems, Process & Control, Kuala Lumpur, Malaysia.
3. S.R.Bharanialankar, C.S.Manikanda Babu, "Intelligent HomeAppliance Status Intimation Control and System Using GSM", 2014, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 4(4), pp.554 - 556.

ISSN: 2277-9655
Impact Factor: 4.116
CODEN: IJESS7

4. Arnab Waheed Ahmad, Naeem Jan, Asaeed Iqbal, Chankil Lee, Ansan Korea, "Implementation of ZigBee- GSM based Home Security Monitoring and Remote Control System", 2011, IEEE Symposium on Circuits and Systems (MWSCAS), Seoul, pp. 1-4.
5. R.Piyare,M.Tazil, " Bluetooth Based Home Automation System Using Cell Phone", 2011 ,IEEE 15th International Symposium on Consumer Electronics, Singapore, pp. 192 - 195.
6. Home Automation System via Bluetooth Home Network", 2003, SICE Annual Conference, Fukui, Vol. 3, pp. 2824 - 2829.
7. H. Brooke Stauffer "Smart Enabling System for Home automation", 1991, IEEE Transactions on Consumer Electronics, Vol. 37(2), pp. 29-35.
8. Ersue, M.; Romascanu, D.; Schoenwaelder, J.; Sehgal, A. (4 July 2014). "Management of Networks with Constrained Devices: Use Cases". *IETF Internet Draft*.
9. "Django Framework" https://mail.google.com/mail/u/0/#inbox?compose=15f09d8d37d1b081
10. "EAGLE-PCB design made easy" https://www.autodesk.com/products/eagle/overview
11. "Arduino" https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwjz-PfBy_TQAhXCPo8KHciXB_8QFggZMAA&url=https%3A%2F%2Fwww.arduino.cc%2Fen%2FMain%2FArduinoBoardUno&usg=AFQjCNG2_XmI1B7CthSKYr1qYtl8q70AQ
12. "Ethernet Shield" https://www.arduino.cc/en/Guide/ArduinoEthernetShield

CITE AN ARTICLE

Gupta , S., Jatar, S., & Mittal, A. (2017). HOME AUTOMATION SYSTEM USING ARDUINO UNO. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY*, *6*(10), 399-405.