International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.02, April 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

Design and Implementation of a Wifi Based Home Automation System

Assistant Prof. Mr. Damodar kumawat[1], Priyanshi gautam Rehana parveen[2] ,Ritu sharma[3] ,Vikram mandawat[4]

(Gurukul Institute of Engineering Technology, kota)

(Department of electrical engineering 4th year GIET kota)

ABSTRACT

In this paper we have worked on a home automation system using ESP8266 WiFi module and Arduino Uno. Using this we will be able to control lights, electric fan and other home appliances through a web browser using your PC or mobile. These AC mains appliances will be connected to relays which are controlled by the Arduino. ESP8266 and Arduino together acts as a Web Server and we will send control commands through a Web Browser like Google Chrome or Mozilla Firefox. ESP8266 is the one of the most popular and low cost wifi module available in the market today. You can ready more about it here, ESP8266 – WiFi SoC.

Keywords :Home automation, Wireless LAN, WiFi ,MicroControllers

I. INTRODUCTION

A. Overview- Now a days home and building automation systems are used more and more. On the one hand, they provide increased comfort especially when employed in a private home. On the other hand, automation systems installed in commercial buildings do not only increase comfort, but also allow centralized control of heating, ventilation, air condition and lighting. Hence, they contribute to an overall cost reduction and also to energy saving which is certainly a main issue today. Existing, well-established systems are based on wired communication. Examples include BA C net, Lon Works and KNX [1].

Employing a traditional wired automation system does not pose a problem as long as the system is planned before and installed during the physical construction of the building. If, however, already existing buildings should be augmented with automation systems, this requires much effort and mush cost since cabling is necessary. Obviously, wireless systems [1] can come to help here. In the past few years, wireless technologies reached their breakthrough. Wireless based systems, used every day and everywhere, range from wireless home networks and mobile phone to garage door openers. As of today, little comparative research of wireless automation standards has been done, although such knowledge would provide valuable information to everyone looking for the most suitable system for given requirements.

B. Features and benefits of home automation system [2]-

In recent years, wireless systems like WLAN have become more and more common in home networking. Also in home and building automation systems, the use of wireless technologies gives several advantages that could not be achieved using a wired network only.

1. Reduced installation costs: First and foremost, installationcosts are significantly reduced since no cabling is necessary. Wired solutions require cabling, where material as well as the professional laying of cables (e.g.into walls) is expensive.

2. Easy deployment, installation, and coverage: Wireless nodes can be mounted almost anywhere. In adjacent or remote places, where cabling may not be feasible at all, e.g., a garden house or the patio, connection to the home network is accomplished instantly by simply mounting nodes in the area. Hence, wireless technology also helps to enlarge the covered area.

3. System scalability and easy extension: Deploying a wireless network is especially advantageous when, due to new or changed requirements, extension of the network is necessary. In contrast to wired installations, additional nodes do not require additional cabling which makes extension rather trivial. This makes wireless installations a seminal investment.

4. Aesthetical benefits: As mentioned before, placement of wireless nodes is easy. Apart from covering a larger area, this attribute helps to full aesthetical requirements as well. Examples include representative buildings with all-glass architecture and historical buildings where design or conservatory reasons do not allow laying of cables.

5. Integration of mobile devices: With wireless networks, associating mobile devices such as PDAs and Smartphones with the automation system becomes possible everywhere and at any time, as a device's exact physical location is no longer crucial for a connection (as long as the device is in reach of the network). Typical examples include an engineer who connects to the network, performs a particular management task, and disconnects after having finished the task; or control of blinds using a remote control. For all these reasons, wireless technology is not only an attractive choice in renovation and refurbishment, but also for new installations.

II. SYSTEM ANALYSI

A. Problem definition

Home automation systems face four main challenges [3], these are high cost of ownership, inflexibility, poor manageability, and difficulty achieving security. The main objectives of that research is to design and to implement a cheap and open source home automation system that is capable of controlling and automating most of the house appliance through an easy manageable web interface to run Design and Implementation of a WiFi Based Home and maintain the home automation system. The proposed system has a great flexibility by using WiFi technology to interconnect its distributed modules to home automation server. That will decrease deployment cost and will increase the ability of upgrading, and system reconfiguration. System will make use of secure wireless LAN connections between distributed hardware modules and server, and secure communication protocols between users and server.

B. Proposed system feature

The proposed system is a distributed home automation system, consists of server, hardware interface modules. Server controls hardware one interface module, and can be easily configured to handle more hardware interface module. The hardware interface module in turn controls its alarms and actuators. Server is a normal PC, with built in WiFi card, acts as web server. The webserver software is developed using asp.net technology, so web server should support asp application and.net frame work 4.0, like IIS7.0 for windows OS. System can be accessed from the web browser of any local PC in the same LAN using server IP, or remotely from any PC or mobile handheld device connected to the internet with appropriate web browser supports asp.net technology through server real IP (internet IP). WiFi technology is selected to be the network infrastructure that connects server and hardware interface modules. WiFi is chosen to improve system security (by using secure WiFi connection), and to increase system mobility and scalability. Even if, user intends to add new hardware interface modules out of the coverage of central access point, repeaters or managed wireless LAN will perfectly solve that problem. The main functions of the server is to manage, control, and monitor distrusted system components, that enables hardware interface modules to execute their assigned tasks (through actuators), and to report server with triggered events (from sensors). In setup mode, user can add and remove hardware interface modules, and can create basic macros involving simple triggers and to customize the macros to perform complex series of events. Macros can be activated manually or as a reaction for certain trigger like motion sensors and surveillance cameras. User can also program macros to activate at random; this feature allows your system to turn the lights on and off at random or semi-random intervals.In running mode, if hardware interface modules report server with received events and execute their pre-programmed macros.

Hardware interface modules are directly connected to sensors and actuator through direct wires connections.

Hardware interface modules has the capabilities to control energy management systems like lighting, thermostats and HVAC (heating, ventilation, and cooling) systems, and security systems (door locks, cameras, motion detectors, fire alarms…).

C. System requirements

The following list gives an overview of the most important requirements of the proposed system

1. User friendly interface: User can easily manage system locally or remotely home automation system, through easy web based interface.

2. Security and authentication: Only authorized user can login to the system (locally, or remotely) in order to manage, control, & monitor. If system detects intruders it should immediately alert the system owner and lock login capability for a while.

3. Low cost per node / High node count: Thinking of building automation, hundreds of nodes may be needed to provide automation. However, the market requires competitive performance (compared to wired

networks) to be delivered at this low system cost. Additionally, also protocols need to scale to high node count e.g., ensuring message delivery

4. Large area coverage: Another challenge lies in the fact that devices of a building automation system are dispersed over large areas. Since transceivers must not consume so much power, they cannot be built with a transmission range sufficient for sensors to reach associated controllers or actuators directly. Also, they may rely on an infrastructure of access points and a wired backbone network (or particularly sensitive receivers).

5. System Scalability: Scalability is the ability of a system, network, or process, to handle growing amount of work in a capable manner or its ability to be enlarged to accommodate that growth. For example, system upgrade/downgrade by adding/removing hardware interface module should be easy and systematic task.

III. SYSTEM DESIGN AND IMPLEMENTATION

Proposed Home Automation System layout As mentioned the proposed home automation system consists of three main modules, the server, the hardware interface module, and the software package. Secure WiFi technology is used by server, and hardware interface module to communicate with each other. User may use the same technology to login to the server web based application. if server is connected to the internet, so remote users can access server web based application through the internet using compatible web browser

Temperature and humidity

Motion detection

Fire and smoke detection

Door status

Light level

Video monitoring

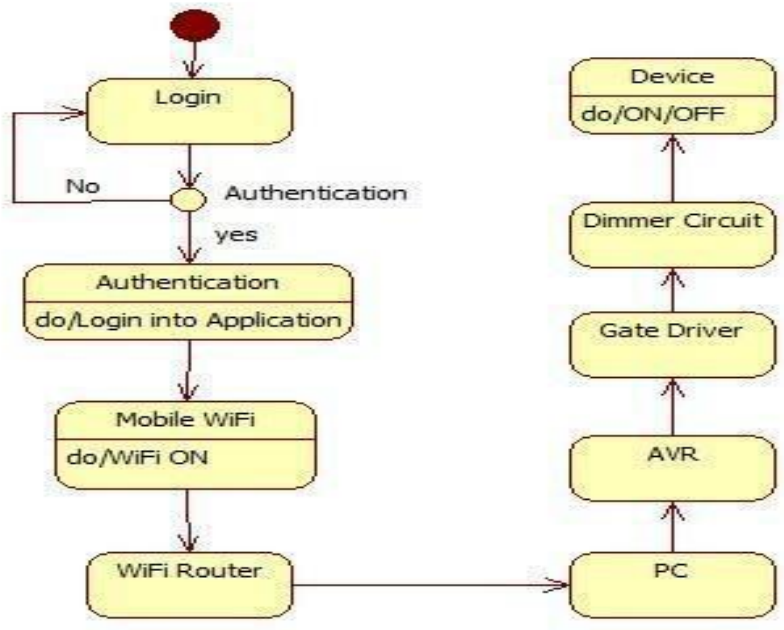


Fig. 1 The proposed home automation system layout

The proposed home automation system can control the
following appliance;
Serine
Lights on/off/dim
HVAC on/off
Door lock
Window shutdown
On/off different appliance

C. User Classes and Characteristics

The proposed home automation system is designed as a tool for the casual user. A casual user; shall be defined as one possessing general knowledge of the Microsoft Windows operating system and general knowledge of using the Internet by employing a standard browser such as Microsoft Internet Explorer General user; who will have the most use of the system functionality. Administrator; who will control the access and permissions policy of the system, and can add and delete user accounts, anything that a general user can perform, the administrator can also perform

D. Design and Implementation Constraints

The Proposed home automation system is implemented using ASP, HTML and CSS. The server application is implemented in ASP.Net, and the embedded hardware interface application shall be implemented using C Processing Language.

E. Assumptions and Dependencies

The component of the system will always be connected
Each User must have a User ID and password
There is only one Administrator.
Server must always run under windows system
There should be Internet connection available.
Proper browsers should be installed
Proper Hardware Components are available
User is capable of using a computer

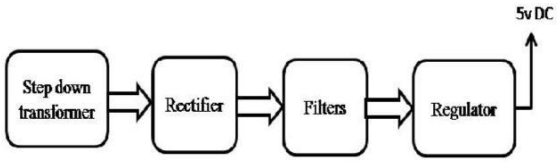


Fig- Regulated power supply

F. Software design concept

Software of the proposed home automation system is divided to server application software, and Microcontroller (Arduino) firmware.

The server application software package for the proposed home automation system, is a web based application built

using asp.net, Microsoft Visual Studio 2010. Server application software runs on windows OS, requires IIS web server, and —.Net version 4.0 being installed. The server application software can be accessed from internal network or from internet if the server has real IP on the internet using any internet navigator supports asp.net technology. Server application software is responsible of setup, configuration, maintain the whole home automation system. Server use database to keep log of home automation system components, we choose to use XML files to save system log.

The Arduino software, built using C language, using IDE comes with the microcontroller itself. Arduino software is responsible for collecting events from connected sensors, then apply action to actuators and pre-programed in the server.

Another job is to report the and record the history in the server DB. Figure(2) shows the architecture of the proposed homeautomation system. The following figure (3) shows classes diagram of proposed system, which consists of five main classes.

G.Classes description

1) dataSourceLog:

Create a log text file in a specific path. This log file has a new log entry, ready to be filled with data. A log entry has the date of the day and a log number.

writeToLogEntry: Function to write the data into the log file; using a stream writer to convert it and save it into a TEXT file.

readLog: Read from the file line by line, the log entry with the data

readNewLog: This function reads the new log entry as just been saved.

2) datasourceXML:

responsible for all operations regarding the main XML files,

User, Sensors and Automation

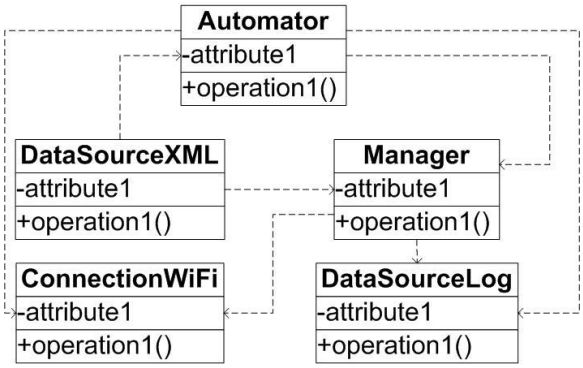


Fig. 3 proosed system class diagram

get User Hex: User Hex is a security issue. Checking progress will not only be on the username and password, but also on a random generated number for each user. This number changes every time the user logs into the system

checkUser: Check if the entered username and password is the exact match with the saved XML file or not

getSensorStatus: Get a sensor last status from the file automationTime: Sets the time that the automation is supposed to be activated, in hours, minutes, and seconds. The following are of the same functionality; which is to change the data of a specific attribute in the XML file given from the function name:

automationMotion, automationDoor, automationTemp, automationAppliance, automationDuration, automationSecurity

3. Automator: Responsible for all automation processes, including the monitoring of the environment for conditions that matches the automation rules configured by the user statusInitializer: set all the sensors and actuators back to its initial values. Sensors will be set to 0, and the actuators will be set to the last update in the XML data files.

getDataFromXML: Get data from XML function Process: check every one of the sensors and appliance either it contains the key word from the XML file dictionary or not. If it contains the key word it sends the data to check dependency function.

checkDependency: check if the automation of a specific sensor in an automation is dependent on any other sensor, appliance, or a specific time to be activated.

watcher: Watcher function acts like a guard; it keeps watching the timer and all other sensors and actuators last updated statuses from the XML file. When all conditions of an automation becomes true. Automation executed.

processSensor: checks sensors to see the last updates, in case of any dependent automation on one of the sensors.

activateAppliance: Activates any appliance that should be activated according to the running automation.

4) Manager:

Responsible for maintaining the whole system in general and initializing all that is needed for further operation when the system is first started

Processor: initializes and load all needed data into the system and establish communication with the hardware module

add Automat or Object: load an automation entry into the system to be watched

delete Automation: deletes an automation from the system and XML file and also stops the system from watching it anymore get New Automation: get all the automation that are assigned to run today go Manual and go Automation: toggle the system from manual control to automatic control, needed if the user wishes to stop any automation and have full control add Automat or Object: load an automation entry into the system to be watched delete Automation: deletes an automation from the system and XML file and also stops the system from watching it anymore ge New Automation: get all the automation that are assigned to run today

go Manual and go Automation: toggle the system from manual control to automatic control, needed if the user wishes to stop any automation and have full control.

connection Wifi: Responsible for all communication to the hardware module and the data transmission in between

wifiIntializer: initialize the connection to a specified socket

writeToSocket: sends data to the specified socket in the form of a string

readFromSocket: read data from socket after sending it a variable to flag the type of data that is needed to be received The following are of the same functionality, which is to get the latest status of the specified sensor given in the function name: getDoor, getMotion, getTemperature The following are of the same functionality, which is to toggle the specified appliance given in the function name to a World Academy of Science, Engineering and Technology 68 2012 2180

new state: switchLights, switchAC, switchAlarm

H.Data Flow

Login interface Security Module Data Source

Logger

Log Interface

To start using the system; the user has to use the login interface to log in into the system. Data passes a security module to be transferred to a 128-bit hex key and checked for availability in the data source represented in XML files. The the system writes this action into the logger which in turn sends it to the Log interface.

Status Interface

Communication Module

Hardware Interface

Logger

Log Interface.

Acquiring the status of an actuator or a sensor using the Status interface, is done by receiving these data directly from The communication module, which in turn gets the data required through the Hardware interface represented into the microcontroller. While this process is done and the status is checked repeatedly, any changes will be written in the Logger and appears in the Log interface.

Control Interface Communication Module Hardware Interface Logger Log Interface. Changing the status of an actuator through the Control interface, is done by sending data directly to the Communication Module, which in turn sends it to Hardware interface represented in the microcontroller to apply the required changes. During the process of sending data and changing the actuator, changes are being written into the Logger and appear in the Log interface.

Control Interface Data Source.

After changing a status of an actuator and writing it into the log using the Control interface, these changes are being written in the Data Source represented in XML files for later checks. Automator Data Source. Setting up a new Automation for the system to do is done by the Automation interface, which just saves it into the Data Source XML files; waiting to be activated.

Status, Control, Data Source Automator

Communication Module Hardware Interface Logger Log Interface.

The Automator is responsible for executing the automations that have been already saved by the users. In order to do that, it gathers data from the Data Source XML files, Status, and Control interfaces; compares these data with the entered ones in the XML files. Then it sends the appropriate changes required to the Communication Module in order to apply it in the Hardware Interface. During this process changes are being written in the Logger and appear in the Log Interface.

I. Hardware design

The second part of the proposed home automation system design was the choosing of a suitable micro-controller. The requirements for the micro-controller are; a RS232 port, a fair amount of output Digital I/O, and a reasonable speed

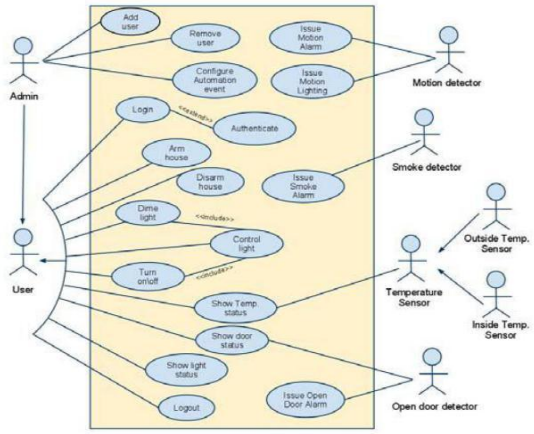
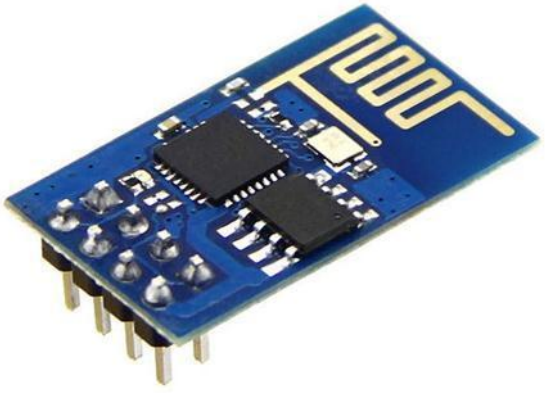


Fig. 4 Proposed system use cases

Also a reasonable amount of EEPROM was needed to enables the system to store device status, and username and passwords related to login, so that data is not lost in the case of power outages

ESP-01 E SP8266 Module-



ESP-01 is the one of the most popular ESP8266 module available in the market. ESP8266 is a self contained SoC with integrated TCP/IP stack which helps any microcontroller having UART to access a wifi network. It can act as both WiFi access point as well as a WiFi client. It is pre-programmed with AT commands, so we can easily access and configure it using a microcontroller. Arduino TTL input pins will detect 3.3V as logic high, so we can directly connect 3.3V output of ESP8266 Tx to Arduino Rx pin. Circuit First we can connect ESP8266 with the Arduino Uno. The ESP8266 runs on 3.3V, it may damage if you connect it directly to 5V from Arduino. The pin out of the ESP-01 ESP8266 moduleConnect the VCC and CH_PD of the ESP8266 to the 3.3V output pin of Arduino. CH_PD is Chip Power Down pin, which is active low. So we will give 3.3V to it, which will enable the chip. Then connect the TXD pin of the ESP8266 with the digital pin 2 of the Arduino. Then make a voltage divider to make 3.3V for the RXD of the ESP8266 which is connected to the pin 3 of Arduino. Here we are using software UART through digital pins 2 & 3 of Arduino. Lastly, connect the ground of the ESP8266 with the ground of the Arduino [4].

Now we can connect relays to Arduino. Connect three relays to pins 11, 12 and 13 of the Arduino. Also connect 5V and ground from the Arduino to power the relay. Note that here I am using relay modules which having built in transistor driver. So don't forget to add driver when you are using bare relays. We can connect AC devices to the output terminals of those relays. First connect one wire (Phase) of the AC source with the common terminal (COM) of all ESP8266 runs on 3.3V and its input pins are not 5V tolerant. So we need to reduce the 5V output of the Adriano Tx pin to 3.3V by using voltage dividing resistors to connect to Rx pin of ESP8266 module. relays and the second wire (Neutral) of AC source to one terminal of AC devices. Then connect the other terminal of AC devices to the NO (Normally Open) terminal of relays. is shown below. Program

Arduino Sketch-

```
#include <SoftwareSerial.h> //Including the software serial library
#define DEBUG true
SoftwareSerial esp8266(2,3); // This will make the Arduino pin 2 as the RX pin and Arduino pin 3 as the TX. Software UART
/* So you have to connect the TX of the esp8266 to the pin 2 of the Arduino and the TX of the esp8266 to the pin 3 of the Arduino. This means that you need to connect the TX line from the esp to the Arduino's pin 2 */
void setup()
{
Serial.begin(9600);          // Setting the baudrate to 9600
esp8266.begin(9600); // Set it according to your asp's baudrate. Different asp's have d pin Mode(11,OUTPUT);
// Setting the pin 11 as the output pin.
digital Write(11,LOW); // Making it low.
pin Mode(12,OUTPUT);   // Setting the pin 12 as the output pin..

digital   Write(12,LOW); //        Making  pin       12       low.      pin

Mode(13,OUTPUT);       // Setting the pin 13 as the output pin.
```

```
digital Write(13,LOW); // Making pin 13 low.
SendData("AT+RST\r\n",2000,DEBUG);     //This command
will reset module to default
Send Data("AT+CWMODE=2\r\n",1000,DEBUG);    // This
will configure the mode as access point
sendData("AT+CIFSR\r\n",1000,DEBUG);  // This will get
ip address and will show it
sendData("AT+CIPMUX=1\r\n",1000,DEBUG);       // This will
configure the ESP8266 for multiple connections
sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // This will set the server on port 80
}
void loop(){
if(esp8266.available()) // Checking that whether the esp8266 is sending a message or not (Software UART Data)
{
if(esp8266.find("+IPD,"))
{           delay(1000);       // Waiting for 1 sec
int connectionId = esp8266.read()-48; // Subtracting 48 from the character to get the number.
esp8266.find("pin=");       // Advancing the cursor to the
"pin="
int pinNumber = (esp8266.read()-48)*10; // Getting the first number which is pin 13
pinNumber += (esp8266.read()-48); // This will get the second number. For example, if the pin number is 13 then the 2nd number will be 3 and then add it to the first number
digital Write(pinNumber, !digitalRead(pinNumber)); // This will toggle the pin
The following commands will close the connection String closeCommand = "AT+CIPCLOSE="; closeCommand+=connectionId; closeCommand+="\r\n";
sendData(closeCommand,1000,DEBUG);     //  Sending the
data to the ESP8266 to close the command
}
}
}
String sendData(String command, const int timeout, boolean debug) // Function to send the data to the esp8266
{
String response = "";  esp8266.print(command);         // Send the
command to the ESP8266

long int time = millis(); while( (time+timeout) > millis()) // ESP8266 will wait for some time for the data to receive
```

```
{
while(esp8266.available())// Checking whether ESP8266 has
received the data or not
{
char c = esp8266.read();    // Read the next character.

response+=c;       // Storing the response from the
ESP8266
}

}

if(debug)

{
Serial.print(response);       // Printing the response of the

ESP8266 on the serial monitor.
}
return response;
}
```

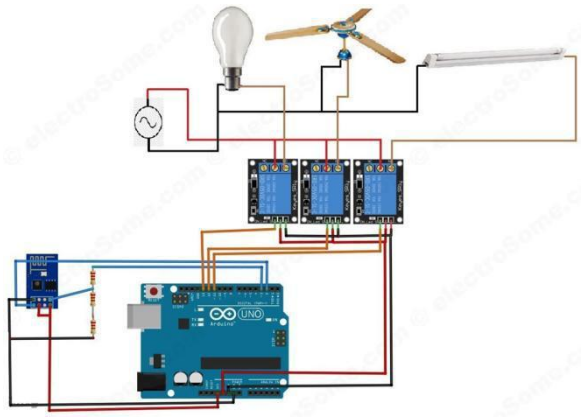
Explanation-The code is very long but easy to understand as it is well commented. First we will initialize the software uart with digital pins 2 & 3 of Arduino for the communication with ESP8266. After that we will initialize pins to which we will connect relays as output pins. Then we will configure ESP8266 in access point mode. Arduino + ESP8266 is programmed as a web server such that we can control those relays through a web browser.

HTML Code

```
<html>

<head>

<title>Home Automation System</title> <!-- This will be the page title -->

</head>

<body> <!-- All the data in it will be shown on the page -->
```

```
<button id="11" class="led">Toggle Pin 11</button> <!-- This will create the button for pin 11 -->

<button id="12" class="led">Toggle Pin 12</button> <!-- This will create the button for pin 12 -->

<button id="13" class="led">Toggle Pin 13</button> <!-- This will create the button for pin 13 -->

<script src="jquery.min.js"></script> <!-- This will read the script from the jquery -->

<script type="text/javascript">

$(document).ready(function(){

$(".led").click(function(){

var p = $(this).attr('id'); // Getting the id value that which pin to toggle, pin 13 or pin 13 or pin 11

Sending the request to the ip address of the server with the pin number to toggle the pin

$.get("http://192.168.4.1:80/", {pin:p}); // Sending the get request

});

});

</body>

);

</script>
```

J.Hardwarelayout



Hardware consists of four different PCBs, the Arduino PCB (ready-made), WiFi shield PCB, 3 input alarms PCB, and 3 output actuators PCB.

IV. IMPLEMENTED PROTOTYPE SNAP SHOOTS

Figures (8) and (9) show snaps shots from proposed system user interface. For more information regarding the functionality and features of the proposed Home Automation System, check the prototype published on line in the following

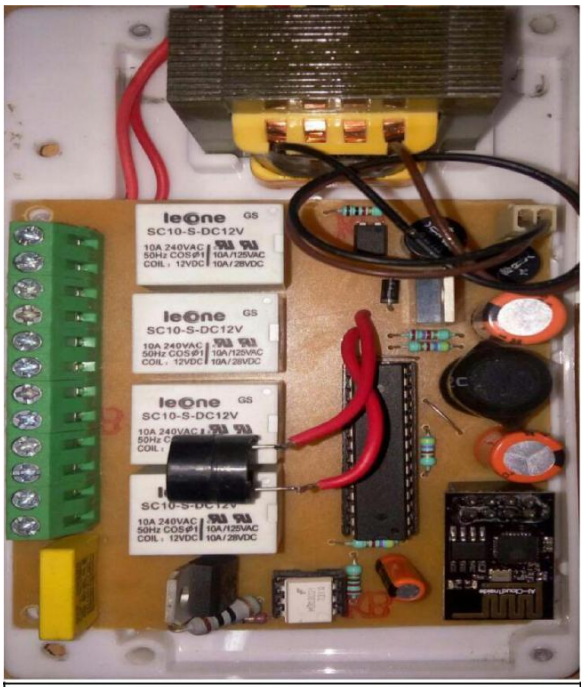
V. FUTUREWORK

The following point presents the suggested future work for implemented prototype: Implements more hardware interface modules, and modify server application software to handle them.

Modify hardware interface module to be able to communicate with sensors and actuators that use wireless technologies like X10, Zigbee, etc. By doing this system will increase system mobility, configurable, and scalability.

More intelligent should be added to hardware modules to make them capable to take decision according to triggered alarms. Without referring to server for each event and action.

That will increase the response time of the system. While hardware interface module reports server with events and actions on pre-programmed intervals .Replace the WiFly WiFi module with more reliable an stable WiFi module, to increase system reliability



VI.CONCLUSION

This paper proposes a low cost, secure, ubiquitously accessible, auto-configurable, remotely controlled solution. The approach discussed in the paper is novel and has achieved the target to control home appliances remotely using the WiFi technology to connects system parts, satisfying user needs and requirements. WiFi technology capable solution has proved to be controlled remotely, provide home security and is cost-effective as compared to the previously existing systems. Hence we can conclude that the required goals and objectives of home automation system have been achieved. The system design and architecture were discussed, and prototype presents the basic level of home appliance control and remote monitoring has been implemented Finally, the proposed system is better from the scalability and flexibility point of view than the commercially available home automation systems.

ACKNOWLEDGEMENT

This paper is based on B. tech graduation project accomplished At Gurukul Institute of engineering Technology, project was supervised by the 1st author, and team is represented by 2nd author.

RESULT

The robust and scalable architecture involves a Web socket and JSON based communication protocol that saves a great deal of bandwidth with terse messages exchanged and exhibits a very low latency needed for a real-time home automation operations saving time, money and space. Although, the actual link characteristics depend on the available Internet connection strength, even at relatively poor connection situations, the system performance is guaranteed. While, in the event of total Internet interruption or offline use, the embedded mini web and Web socket servers running on the CPE suffice for continual operations

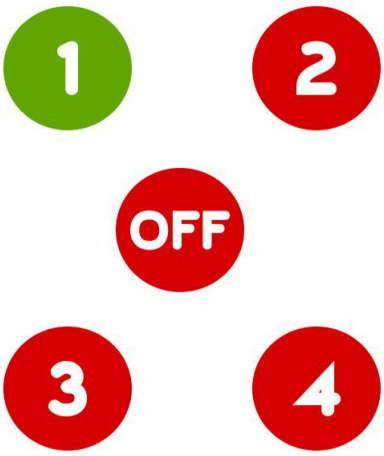
REFERENCES

1. A.J. Bernheim Brush, Bongshin Lee, Ratul Mahajan, Sharad Agarwal, Stefan Saroiu, and Colin Dixon, "Home Automation in the Wild: Challenges and Opportunities", CHI 2011, May 7–12, 2011, Vancouver, BC, Canada

2. Christian Reinisch ,―Wireless Communication in Home and Building Automation‖ , Master thesis, Viennia univeristy of technlogy, Feb 2007.

3. [3] E. Ahmed and A. H. Karim, ―Design and Implementation of a Wi-Fi Based Home Automation System,‖ World Academy of Science,Engineering and Technology, vol. 6, pp. 1856-1862, 2015.

[]http://wiki.smarthome.com/index.php?title=Home_Automatin



Designed By 5 Tech Solutions