

# **SECURED, EFFICIENT SURVEILLANCE AND MONITORING APPLIANCES USING WEB**

## **A Mini Project Report**

*Submitted by*

<b>A.TEJASWI</b>	<b>158W1A04N5</b>
<b>K.VIJAY SAMBA SIVA RAO</b>	<b>158W1A04K3</b>
<b>S.SRAVANI</b>	<b>158W1A04N2</b>
<b>P.RAVI TEJA</b>	<b>158W1A04M1</b>

*In partial fulfilment of the requirements*

*for award of the degree of*

**BACHELOR OF TECHNOLOGY**

with specialization in

**ELECTRONICS AND COMMUNICATION ENGINEERING**

Under the esteemed guidance of

**Dr. A.ANITHA**

**Sr. Assistant Professor**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**VELAGAPUDI RAMAKRISHNA SIDDHARTHA ENGINEERING COLLEGE**

**VIJAYAWADA**

**NOVEMBER 2018**

**VELAGAPUDI RAMAKRISHNA SIDDHARTHA ENGINEERING  
COLLEGE**

**VIJAYAWADA – 520007**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION  
ENGINEERING**



**CERTIFICATE**

This is to certify that the Mini project entitled “**SECURED, EFFICIENT SURVEILLANCE AND MONITORING APPLIANCES USING WEB**” that is being submitted by **A.TEJASWI (158W1A04N5) , K.VIJAY SAMBA SIVA RAO (158W1A04K3), S.SRAVANI (158W1A04N2), P.RAVI TEJA (158W1A04M1)** of B.Tech, VI Semester, Electronics and Communication Engineering in partial fulfilment of requirements for the award of the degree of Bachelor of Technology in the branch of Electronics and Communication Engineering to the Jawaharlal Nehru Technological University, Kakinada is a record of bonafide work carried out under my guidance and supervision. The results embodied in this report have not been submitted to any other university or institute for the award of any degree.

**TERM PAPER GUIDE**

**(Dr. A.ANITHA)**

DATE:

**HEAD OF THE DEPARTMENT**

**(Dr. K. SRI RAMA KRISHNA)**

DATE:

## ACKNOWLEDGEMENT

We would like to articulate our profound gratitude and indebtedness to our guide **Dr.A.Anitha, Sr. Assistant Professor** who has always been a constant motivation and guiding factor throughout the Mini project time in and out as well. It has been a great pleasure for us to get an opportunity to work under her guidance and complete the Mini project successfully.

We wish to extend our sincere thanks to **Dr. K. Sri Rama Krishna**, Professor and Head of the Electronics and Communication Engineering Department, for his constant encouragement throughout the work.

We sincerely thank our Principal **Dr. A. V. Ratna Prasad**, for his encouragement during the course of Mini project.

We express our heartfelt gratitude to our Mini project Co-ordinator who helped us in all aspects.

We thank one and all who have rendered help to us directly or indirectly in the completion of work.

<b>A.TEJASWI</b>	<b>(158W1A04N5)</b>
<b>K.VIJAY SAMBA SIVA RAO</b>	<b>(158W1A04K3)</b>
<b>S.SRAVANI</b>	<b>(158W1A04N2)</b>
<b>P.RAVI TEJA</b>	<b>(158W1A04M1)</b>

## **DECLARATION**

We hereby declare that the work is being presented in this Mini Project entitled **“SECURED, EFFICIENT SURVEILLANCE AND MONITORING APPLIANCES USING WEB”**, submitted towards the partial fulfilment of requirements for the award of the degree of **Bachelor of Technology in Electronics and Communication Engineering** in V.R.Siddhartha Engineering College, Vijayawada is an authentic record of our work carried out under the supervision of **Dr. A.Anitha, Sr. Assistant Professor** in ECE Department, in V. R. Siddhartha Engineering College, Vijayawada.

The matter embodied in this dissertation report has not been submitted by us for the award of any other degree. Furthermore, the technical details furnished in various chapters of this report are purely relevant to the above Mini Project.

<b>A.TEJASWI</b>	<b>(158W1A04N5)</b>
<b>K.VIJAY SAMBA SIVA RAO</b>	<b>(158W1A04K3)</b>
<b>S.SRAVANI</b>	<b>(158W1A04N2)</b>
<b>P.RAVI TEJA</b>	<b>(158W1A04M1)</b>

## **TABLE OF CONTENTS**

<b>Contents</b>	<b>Page No:</b>
<b>ABSTRACT</b>	<b>i</b>
<b>LIST OF FIGURES</b>	<b>ii</b>
<b>LIST OF TABLES</b>	<b>iii</b>
<b>CHAPTER-1</b>	
<b>INTRODUCTION</b>	<b>1-8</b>
1.1 AUTOMATION	1
1.1.1 Industrial Automation	2
1.1.2 Home Automation	4
1.2 WHY AUTOMATION	6
1.3 PROJECT OBJECTIVES	8
<b>CHAPTER-2</b>	
<b>LITERATURE SURVEY</b>	<b>9-13</b>
<b>CHAPTER-3</b>	
<b>SYSTEM ANALYSIS AND IMPLEMENTATION</b>	<b>14-16</b>
3.1 PROBLEM IDENTIFIED	14
3.2 SYSTEM REQUIREMENTS	14
3.3 PROPOSED SYSTEM	15
3.4 SYSTEM DETAILS	15
3.4.1 Block diagram	16
3.4.2 Circuit diagram	16
<b>CHAPTER-4</b>	
<b>HARDWARE REQUIREMENTS</b>	<b>17-21</b>

4.1 NODE MCU	17
4.1.1 Working principle of NodeMCU	17
4.1.2 Connection Setup	18
4.1.3 Communication with NodeMCU	18
4.2 TEMPARETURE AND HUMIDITY SENSOR (DHT 11)	18
4.3 FLOWCHART AND CODE	20
4.3.1 Flow Chart	20
4.3.2 Algorithm of NodeMCU	21
<b>CHAPTER-5</b>	
<b>SOFTWARE REQUIREMENTS</b>	<b>22-24</b>
5.1 XAMPP PACK	22
5.2 ARDUINO IDE	23
<b>CHAPTER-6</b>	
<b>SETTING UP PROGRAM</b>	<b>25-30</b>
6.1 STEPWISE PROCEDURE	25
6.2 OVERALL WORKING	30
<b>CHAPTER-7</b>	
<b>RESULTS AND CONCLUSION</b>	<b>31-34</b>
7.1 RESULTS AND DISCUSSION	31
7.2 LIMITATIONS	33
7.3 CONCLUSION	33
7.4 FUTURE WORK	34
<b>REFERENCES</b>	<b>35</b>
<b>APPENDIX</b>	<b>36-59</b>

# **CHAPTER-1**

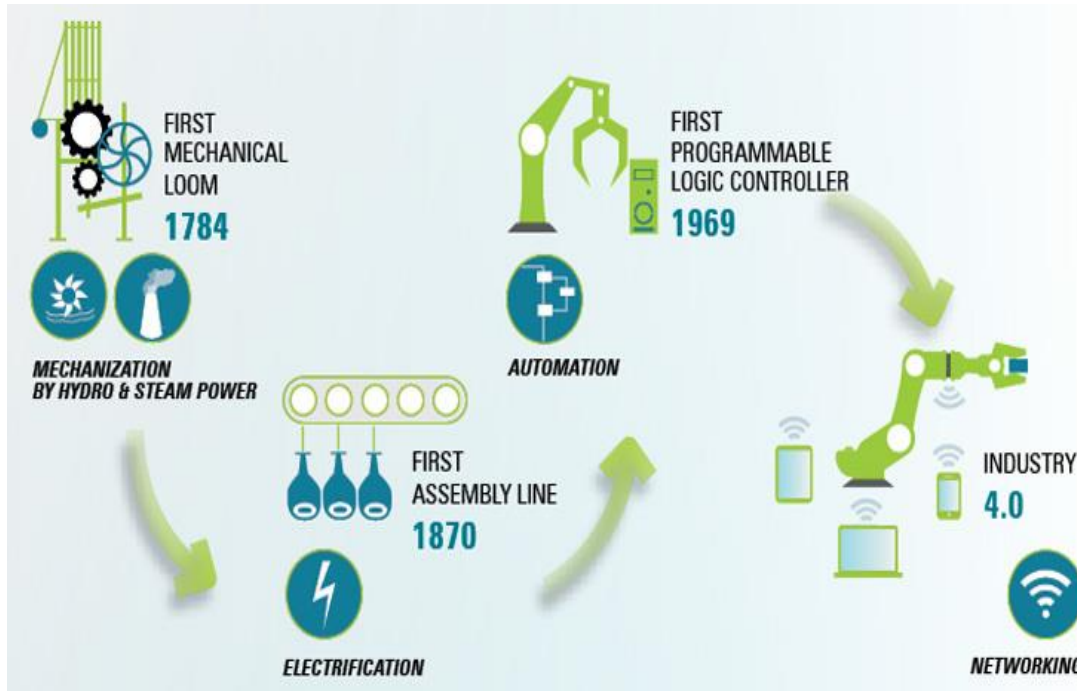
## **INTRODUCTION**

### **1.1 AUTOMATION**

Automation is the technology by which a process or procedure is performed without human assistance. It refers to the state of being automatically operated devices or machines and is the step beyond mechanization where it requires physical labor to perform the functions manually. With the automating the machines, manual labor is reduced by using computer aided and automated machinery. Automation system operation is integrated with the use of sensory systems, feedback control loops and automatic actuating devices. Although the term mechanization is often used to refer to the simple replacement of human labor by machines, automation generally implies the integration of machines into a self-governing system.

In general usage, automation can be defined as a technology concerned with performing a process by means of programmed commands combined with automatic feedback control to ensure proper execution of the instructions. The resulting system is capable of operating without human intervention. The development of this technology has become increasingly dependent on the use of computers and computer-related technologies. Consequently, automated systems have become increasingly sophisticated and complex. Advanced systems represent a level of capability and performance that surpass in many ways the abilities of humans to accomplish the same activities.

“Automation puts the people out of work” is a common misconception of automation. But the reason for putting the process operation to be automated is to increase the productivity and product quality, reduce manual/periodic checking, to improve safety, to reduce the production cost and to be operator friendly. It gives the powerful control solutions to wide variety applications like industries, automobiles, aircraft, and domestic control, etc. In automation control, advanced controllers like Programmable Logic Controllers (PLCs) are deployed for various operations. These controllers are programmed with advanced control loops which are an intrinsic part of it and some of these control structures are feedback, feed forward, cascade, ratio control loops and so on.



**Fig 1.1:** History of trends in technology

The automation system replaces the conventional hard-wired relay system by automating the process functions with the use of various hardware and software tools with minimal human intervention. There are different types of automation systems are used in different applications, but all automated systems are not similar. Depends on application area and control nature, these automated systems are classified. Some of these automation types are:

- (i) Industrial Automation
- (ii) Home Automation

### 1.1.1 Industrial Automation

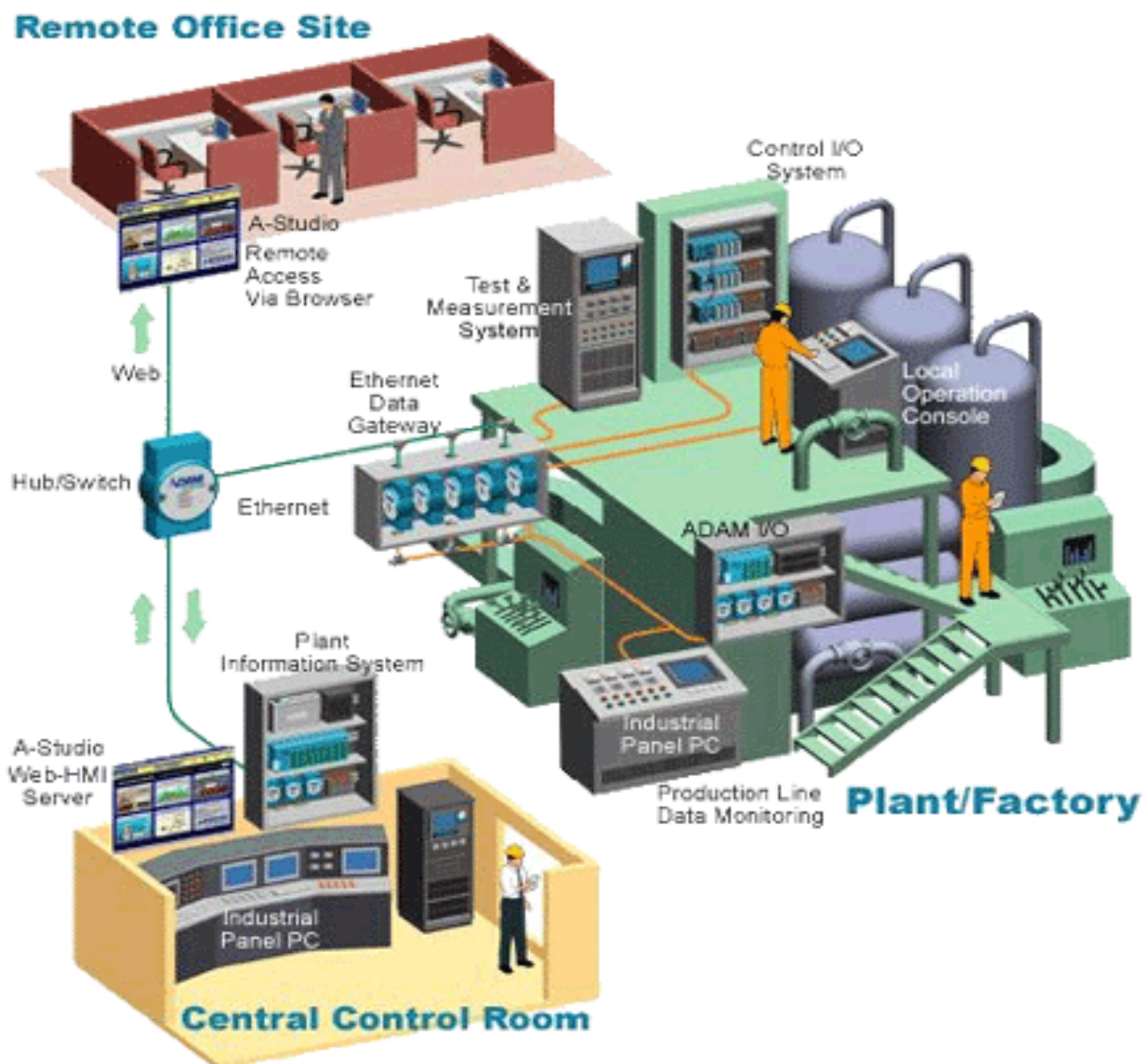
Industrial automation is the process of making industrial production processes more flexible and simpler at higher efficiencies. Automation integration to the industries results intelligent manufacturing solutions at improved product quality and productivity with reduced downtime and wastes. Industrial automation involves in usage of usage of various control-devices such as PC/PLC's, various sensors and actuators, communication buses/modules, machine drives, HMI (Human Machine Interface) systems and other control appliances. This type automation is most popularly used in automotive, computer and electronics, medical,



telecommunication, consumer goods and other industrial applications. The automation systems can be a fixed, programmed, flexible and integrated systems.

- **Numerically Controlled Machines**

These machines are of computer controlled machines which uses computers to perform the control operations by acquiring, processing, calculating and controlling the process variables. This automation is a programmed version of machine tools and also called as Computerized Numerical Controlled (CNC) Machines. These CNC machines are used in cutting and milling applications for high accuracy and accurate precision operation.



**Fig 1.2:** Example for Industrial Automation

- **Computer – Aided Manufacturing (CAM)**

In this, the entire manufacturing process (includes production, planning and control) is automated with the use of numerically controlled machines, industrial robots and other types of automation devices. These automation systems also make use of computers to plan, design and layout the various products. Examples of this automation systems are computer-aided design (CAD), computer-aided design and drafting (CADD) and computer-aided process planning (CAPP).

- **Industrial Robots**

These are a type of automated machines or equipment's that can perform the different tasks for longer duration. These are mostly implemented in the areas that are highly dangerous or hazardous for humans.

- **Flexible Manufacturing Systems**

This automation is of fully automated one. Starting from planning and designing process to dispatching of products, the whole system is completely integrated to be automated. This automation combines numerically controlled machines, industrial robots and other automation equipment's into one integrated system.

Fig; 1.2 depicts how industrial automation is used inside a factory. It also shows how process takes place and relation between various units in factory.

### **1.1.2 Home Automation**

As the technologies are rapidly advancing that affects the human daily lives more flexible, particularly in controlling home appliances. Electrical installations are the heart of every building, therefore intelligent building control absolutely ensures the safety and efficient control and eventually saves the electric power consumption and human energy. Home automation system does the operations like lighting control and regulation, regulating HVAC (Heating, Ventilation and Air Conditioning) equipment's, energy and load management, security and monitoring, audio/video systems, HMI's and other control tasks.

Home automation uses the various sensors to sense parameters like temperature, pressure, motion, illumination, etc. and sends these signals to central controllers. These controllers are programmable devices like PLCs which are programmed according to the application it is

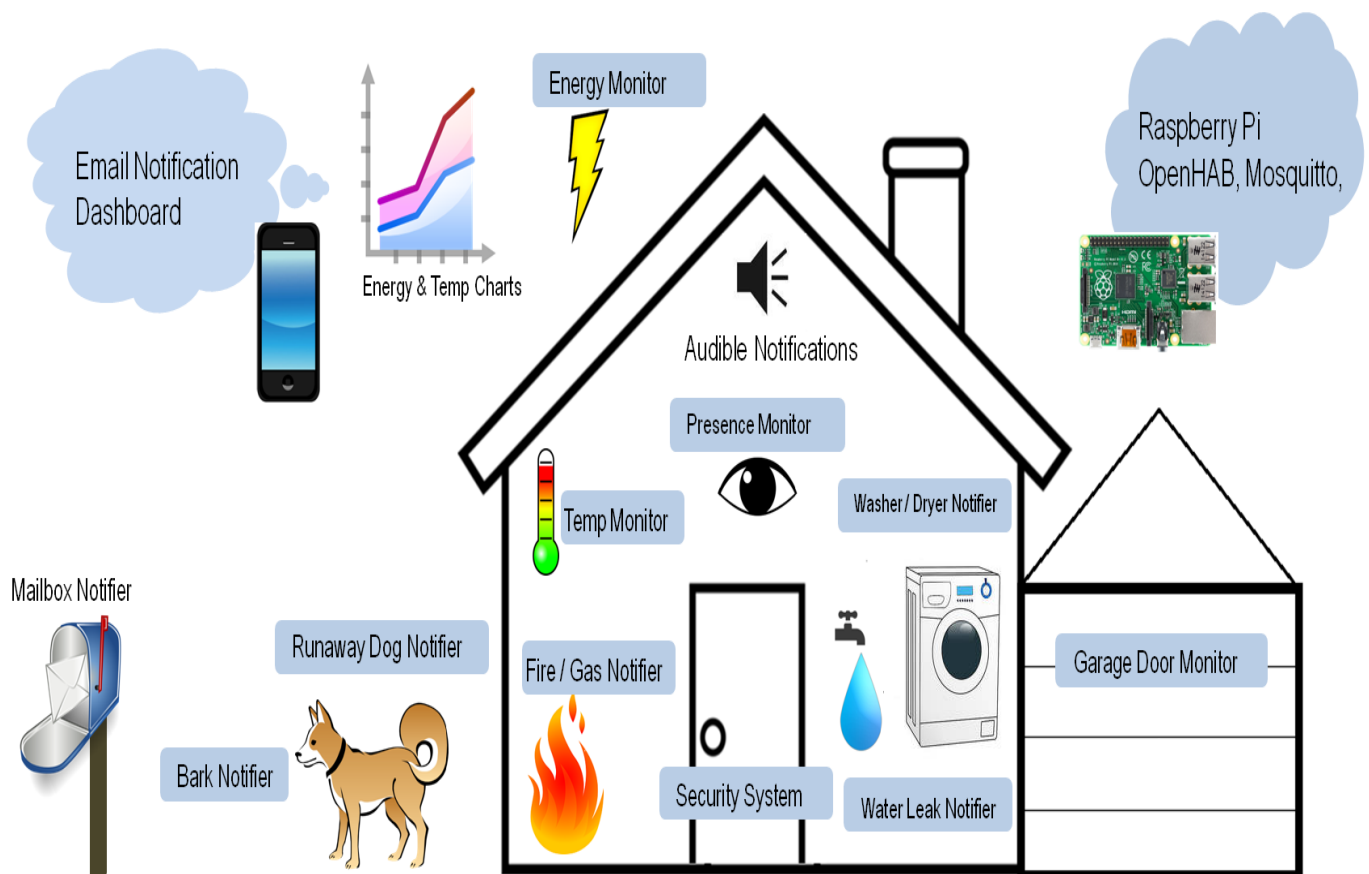
employed for. Therefore, these controllers get the inputs from sensors and correspondingly sends the control signals to actuating devices like relays.

- **Power Line Home Automation System**

This automation is of low cost since it uses power lines to transfer the information or data therefore it doesn't require additional cables for transferring the information. However, this system is of little bit complex one that necessitates additional converter circuits.

- **Wired Home Automation System**

This type of automation uses a common communication bus to which all the appliances in homes are connected. This communication bus or cable is connected to a main controller (programmable logic controller) in order to acquire input signals and to send control command signals to the actuators.



**Fig 1.3:** Home Automation

- **Wireless Home Automation**

This is an advanced automation technology as compared with wired automation. This automaton uses wireless technologies like RF, Zigbee, GSM, Wi-Fi and Bluetooth for achieving remote control operation. This is a flexible type of automation requires negligible wires to connect different appliances in homes to the central control unit.

Fig 1.3 shows a pictorial representation of home automation system.

These are the two types of automation system that has been implementing in most of the industries and homes. Overall quality, accuracy, productivity, reduced labor costs and safety are better arguments for the automation systems

## **1.2 WHY AUTOMATION**

Automation, in any field, brings in the benefits of increased productivity and reduced costs. The process of controlling various operating equipment, machinery, factory operations, etc., automatically (sometimes remotely) using control systems is the demand in the current market. Automation is an efficient method to use in every field such that to reduce manpower, energy usage and also for improving the quality and efficiency of any system. Still some other reasons that show why automation forms an integral part in any development process and business solutions.

**1. Reduce Worker Fatigue and Effort or Labor Intensive Operation** – Typically, humans dislike banal, repetitive tasks. However, computer systems perform them without complaint. Tasks that lack variability provide a place for automated systems to shine, but this also holds true for systems utilizing advanced sensors and integration. If the task requires conditions not suited to human comfort or focus, consider automation.

**2. Prevent Products or Materials from Being Damaged or Destroyed** – Humans make mistakes when they fatigue. This embodies the sentiment of the “human condition.” Mistakes using tools mean damaging raw materials, components, assemblies, and end products.

**3. Prevent Non-conforming Product from Shipping** –Computers controlling robots do not forget steps. Neglecting to put in a screw requires a human touch. A machine not doing it yields an error to be addressed. Does the process require doing something in a specific order to improve yield? Automated systems will not violate the instruction set. Moreover, automated systems may

employ inspection capabilities. Tune the system and allow the data to roll in without preference or bias.

4. **Increase Efficiency** – Improving processes for efficiency makes a company more competitive, but do people always do the same thing in the same way every time they do it? No, human variation exists. Automated systems allow for improvements that benefit from consistent execution. Perfect planning and training does not defend against the human touch.

5. **Collect Better Data** – Remove the accidental data entry or missed data point from logging. Make the method of collecting sensor and process data regulated.

6. **Improve Metrics** – Sending reliable data directly to a database provides an ongoing resource. Does the process improve with changes? Why do I see more failures now than in the past? Leveraging data can provide these answers beyond a simple list of pass/fail statistics from the past. Correlation of associated process data with pass/fail records provides insight rather than guessing “what is causing this?”.

7. **Devise the Right Process Improvements** – Automated systems now collect reliable data. The database provides a searchable forum. What comes next? Equipped with copious amounts of reliable data, engineers make the most of this information. Where problems existed, light shines on the problem. Rather than just changing to seek “continuous improvement,” make changes with better information.

8. **Save Money** – Why instrument that test stand? Why log that data? Why spend the money now? Simply, investing in industrial automation yields cost savings through making processes more regular and collecting data for making confident decisions.

Coming to the need for home automation,

9. **Times Are Changing-** Times change and things that were once considered luxuries are now absolute necessities. For example, in 1995, only 10% of homes had the internet. Now, a short time later, nearly not only does nearly every home have it, our society depends on it for just about every function. While home automation was convenient, the services it provides makes it increasingly indispensable to running a household.

10. **Resources Management is Increasingly Important-** Not only is it true that as a society we have become more conscious about conserving our resources, we also understand that it is increasingly important to do so because of the political and economic climate we live in. Home automation provides one of the best tools available for conserving our resources.

**11. Time Management is Important-** As we move to a more information drive society, how we spend our time is more important. Home automation provides the tools we need to increase productivity and to do more, more effectively. It also allows us to multi-task much more effectively than ever before.

**12. It Allows Us to Improve Our Caregiving-** Today, it is not uncommon to have both small children and aged adults in the home requiring some special oversight. Home automation allows you to provide that without compromising a loved one's dignity but allowing you to check in remotely and receive custom alerts for activity that might indicate trouble. Home automation also allows you to provide care to those living outside the home, too. With the ability to check in via security camera or keep track of activity through sensors, you can call for help the minute you suspect a problem.

### **1.3 PROJECT OBJECTIVE**

- To develop a system that provides, user to control all type of devices in his/her home using IoT.
- To control the devices both in online and offline mode.
- To provide security while controlling devices at home/industry.
- To make device, that responds to various physical parameters like temperature etc.
- To create an user-friendly GUI so that everything can be managed easily.
- To know the status of device whether it is in ON state or in OFF state.

## **CHAPTER-2**

### **LITERATURE SURVEY**

In this era of digitization and automation, the life of human beings is getting simpler as almost everything is automatic, replacing the old manual systems. Nowadays humans have made internet an integral part of their everyday life without which they are helpless. Internet of things (IoT) provides a platform that allows devices to connect, sensed and controlled remotely across a network infrastructure. In the paper [1] focuses on home automation using smart phone and computer. The IoT devices controls and monitors the electronic electrical and the mechanical systems used in various types of buildings. The devices connected to the cloud server are controlled by a single admin which facilitate a number of users to which a number of sensor and control nodes are connected. The admin can access and control all the nodes connected to each user but a single user can control only the nodes to which the user itself is connected. This whole system using Internet of Things (IoT) will allow mobile devices and computers to remotely control all the functions and features of home appliances from anywhere around the world using the internet connection. The system designed is economical and can be expanded as it allows connection and controlling of a number of different devices.

The fast growing world needs various technologies to improve quality of life. The Internet of Things is one such technology impacting human lives. The Internet of Things is a network exchanging the data/information between the devices having Internet connectivity, sensing and computing capabilities. The work [2] uses the ultra- low power consumption ESP8266 system on chip board operating 2.4 GHz ISM band for wireless connectivity. The relay circuitry is interfaced to ESP8266 NodeMCU board using the GPIO pins in order to control 220 V supply power outlet, wherein domestic appliances such as lights, fans, air conditioning system, and so on may be controlled remotely. The WLAN SSID & password allow only authorized persons from the home to control such appliances. The Android app has been designed and generated making use of the open source web application MIT App Inventor-2 to control the domestic appliances using any

Android Smart phone. The Android app created provides graphical user interface supports two options to control appliances:

- (i) the user can use icons or graphical buttons created for respective home appliances and
- (ii) they can use specific voice commands to control those home appliances using Android Smart phone. The main objective of this work is to have a system which is cost effective, robust and also scalable.

In recent years, home automation has become so popular due to its numerous advantages. The home environment has witnessed a rapid introduction of network enabled digital technology. This technology comes with new and exciting opportunities to increase the connectivity of different devices within the home for the purpose of home automation. The paper [3] aims at designing a basic home automation system of controlling multiple appliances which can be monitored and accessed from anywhere in the world with very low cost. The technology incorporates Raspberry Pi and the web server. The Raspberry Pi and Arduino integrated with Nrf modules are used to monitor the home environment appliances, and the readings are passed to the web server designed. The parameters or commands sent through web page are monitored frequently and if any threats found the mobile connected to this web server is alerted through an alarm or message. The user can access this application from anywhere in the world. The result produced is low cost advantageous and absolute. Performance Analysis of different protocols (MQTT, HTTP and CoAP) is estimated using visualizations

The article deals with usage of PLC (Programmable Logic Controller) and PIC (Peripheral Interface Controller) for realization of intelligent family house model. The aim was to design intelligent family house model with help of microelectronic components and compare advantages and disadvantages of PLC Simatic 314C-2DP and PIC 16F873A in this branch. In this thesis [4] were made for example modules serving for temperature control, DC motor control, lights control or modules for security system. Data transfer between PIC and PC is realized by serial communication and between PLC and PC is OPC (OLE for Process Control) communication. Everything is controlled and monitored from SCADA/HMI system called Control Web 6.



Application is accessible via Internet browser in local network. The view of application is adapted to viewing through mobile phone.

Internet of Things (IoT) conceptualizes the idea of remotely connecting and monitoring real world objects (things) through the Internet. When it comes to house, this concept can be aptly incorporated to make it smarter, safer and automated. This IoT project [5] focuses on building a smart wireless home security system which sends alerts to the owner by using Internet in case of any trespass and raises an alarm optionally. Besides, the same can also be utilized for home automation by making use of the same set of sensors. The leverage obtained by preferring this system over the similar kinds of existing systems is that the alerts and the status sent by the Wi-Fi connected microcontroller managed system can be received by the user on his phone from any distance irrespective of whether his mobile phone is connected to the internet. The microcontroller used in the current prototype is the TI-CC3200 Launchpad board which comes with an embedded micro-controller and an onboard Wi-Fi shield making use of which all the electrical appliances inside the home can be controlled and managed.

Home automation system achieved great popularity in the last decades and it increases the comfort and quality of life. In the paper [6] an overview of current and emerging home automation systems is discussed. Nowadays most home automation systems consist of a smartphone and microcontroller. A smart phone application is used to control and monitor the home appliances using different type of communication techniques. In this paper the working principle of different type of wireless communication techniques such as ZigBee, Wi- Fi, Bluetooth, EnOcean and GSM are studied and their features are compared with each other so the users can choose their own choice of technology to build home automation system. Moreover in this research work the survey of different home automation systems is discussed and their advantages and drawbacks are also highlighted.

<b>S. No.</b>	<b>System</b>	<b>Communication Interface</b>	<b>Controller</b>	<b>User Interface</b>	<b>Applications</b>	<b>Benefits</b>
-------------------	---------------	------------------------------------	-------------------	---------------------------	---------------------	-----------------

1.	Wi-Fi based using Arduino Microcontroller	Wireless LAN and Wi-Fi shield	Hardware interface module	web based Application.	Temperature and humidity, Motion detection, Fire detection, Door status, Light level ,Video monitoring, Controlling appliances	Low cost, Secure, Ubiquitously accessible, Auto-configurable, Remotely controlled
2.	Web service and android app Based using Raspberry pi	Web server and interface card	Raspberry pi	Android application	Controlling shutter of window	Autonomous, and Quite scalable
3.	Cloud Based Using Hadoop System	Cloud based data server uses Hadoop Technology	Home gateway and Router	Smart device	Monitoring and Controlling Home Appliances	Effectively manage Semi structured and unstructured data, Reduce computational burden of smart devices
4.	Email Based using Raspberry pi	Internet Modem	Raspberry pi	E-mail	Switching LED	Smart, Economic and Efficient
5.	Cloud Based Using Zig Bee Microcontroller	Zig bee wireless Network	Smart Socket	PC or Android Phone	entrance control management, monitoring the power consumption, temperature and humidity	Convenience, safety, and Powersaving
6.	Smart Task Scheduling	Wired X10 and Wireless Zig bee	Arduino	Android Application	Energy	Energy-efficient and

	Based using Arduino and Android				Management and task scheduling with power and cost	Highly scalable
7.	Wireless Sensors Based with mobile Technology	cloud-based data server	PCB circuits	Mobile Application	monitor the home conditions and power consumption of appliance	Low power consumption And system cost efficiency.
8.	Android based using Arduino	Micro Web Server	Arduino Mega 2560 and the Arduino Ethernet shield	Android App	Light switches, Temperature ,Humidity sensors, Intrusion detection,, Smoke/Gas sensor	Feasibility and Effectiveness
9.	Konnex-Bus based using raspberry pi	SIP Provider	Raspberry pi and Konnex Bus	Mobile App	Lights Control, Temperature Monitoring	Performance improved ,energy consumption could be Reduced.
10.	By Using DTMF	DTMF Receiver	Logical Controller with I/O interface	Computer	Authentication of user by ringing line	Secure and allow International Standard
11.	GSM Based Using PIC Microcontroller	SMS	PIC16F887 microcontroller	Mobile phone	Control appliances	High availability, coverage and Security but costs for the SMS.
12.	GSM Based Using Arduino	SMS	Arduino	Smartphone App	Control appliances	Simplicity
13.		Bluetooth	Arduino		controlling	

	Bluetooth Based using Arduino			Python supported mobile		Secured and Low cost
--	-------------------------------------	--	--	-------------------------------	--	-------------------------

**Table 2.1:** Comparison

## **CHAPTER-3**

### **SYSTEM ANALYSIS AND IMPLEMENTATION**

#### **3.1 PROBLEM IDENTIFIED**

Home automation systems face four main challenges, 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 and maintain the home automation system. The proposed system has a great flexibility by using Wi-Fi 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.

#### **3.2 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.

### **3.3 PROPOSED SYSTEM**

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 appliances and actuators. 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 through server real IP (internet IP). Wi-Fi technology is selected to be the network infrastructure that connects server and hardware interface modules. 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).

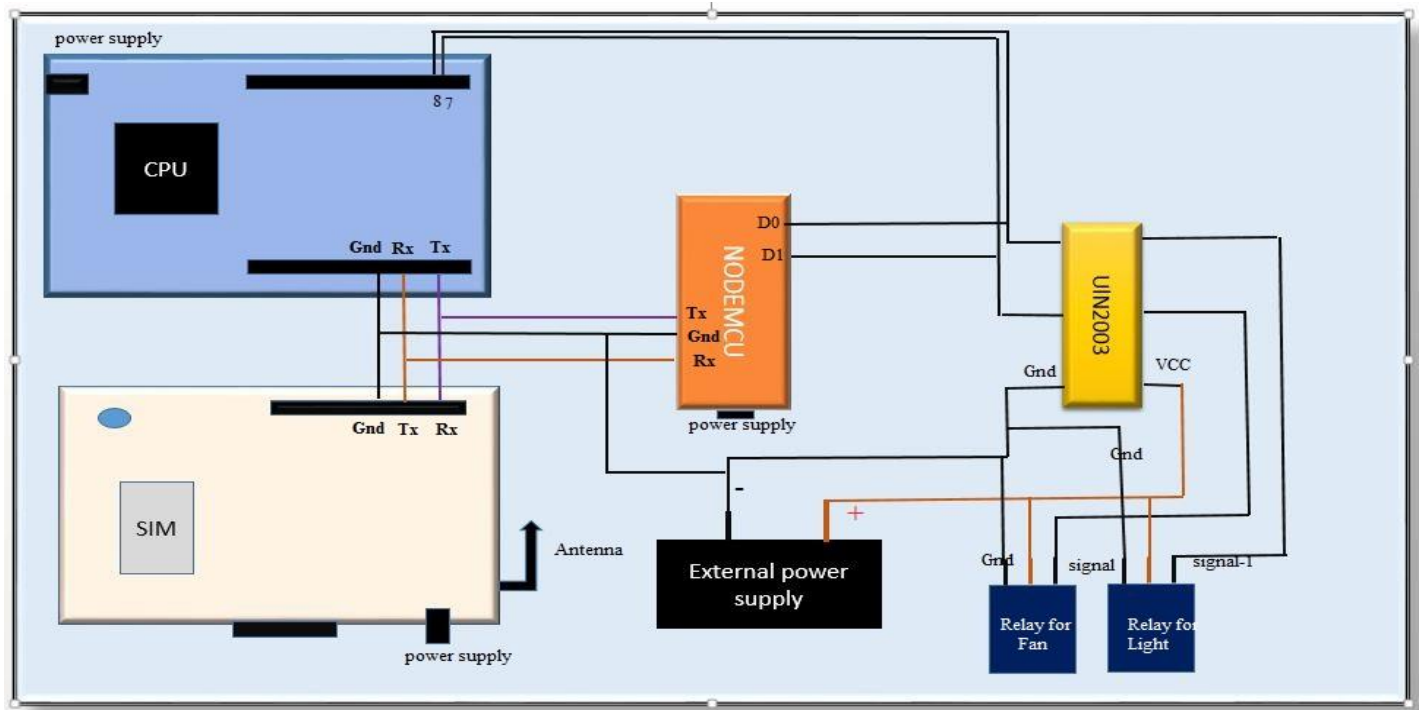
Similarly in places where internet is not available the user can control our devices in offline mode using GSM module. SMS technology used in the system is easy to use, learn and can be accessed easily when needed. A microcontroller can be controlled and monitored from any GSM phone that supports SMS. The system is universal as most of the GSM phones support SMS i.e. the remote to this system can be any basic GSM phone. The range of GSM is global so the user can use the system from any corner of the world. User and/or system administrators are more likely to have their phones with them at all times than they are likely to physically be in front of their computers. After the desired operation of the system is performed the system provides the user with an acknowledgement. SMS is used as the main communication medium. Abnormal

conditions like power failure or the malfunction of the device is also monitored and informed by the installed system to the user who is remotely located. The construction of this system is easy, cheap and versatile, and can be used in any process industry with a little modification. But in GSM based automation system, there should be the user intervention for controlling home appliances wherein the user has to send some commands like “#A.light on\*”, “#A.light off\*” etc. After receiving these commands by the GSM module, the microcontroller send signal to relays, to switch ON or OFF the home appliances using a relay driver.

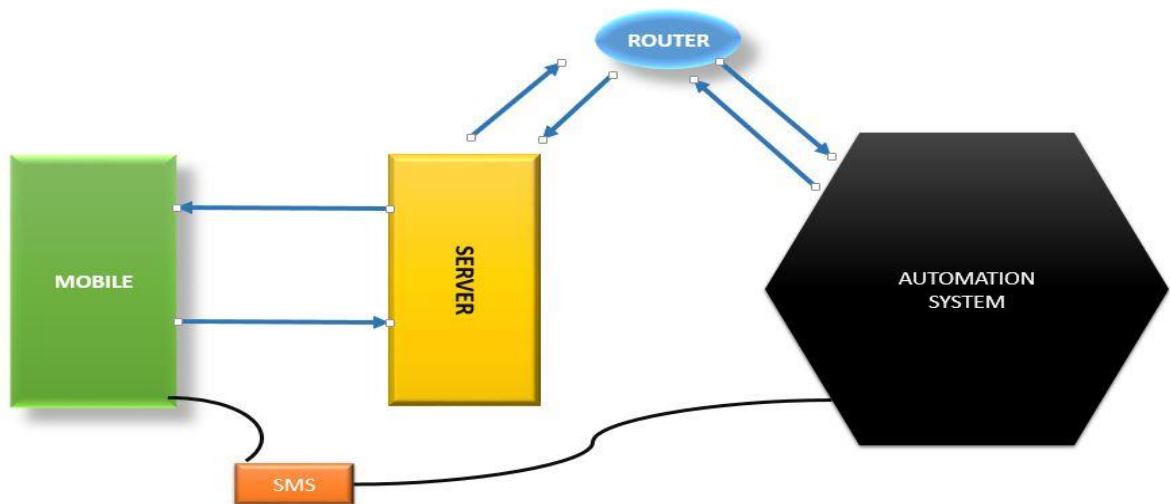
### **3.4 SYSTEM DETAILS**

#### **3.4.1 Block Diagram**

The system has four main parts: Server, Gsm, Arduino, NodeMCU. A local network will be established in order to make them communicate with each other. Once the local network is established NodeMCU and server tries to connect with local network. After connecting with the local network, an IP address is generated for both server and NodeMCU, based on those IP addresses communication will be established between them. Once all steps mentioned above are completed, user can connect to that local network through IP address of server and enter his/her details in order to login into the application so far developed. After a successful login, user can control his devices by giving appropriate commands with the help of GUI provided in the control panel in order to control his devices. In offline mode, the devices can be controlled by GSM module with the help of Arduino Uno. The block diagram is shown below in Fig 3.1.



**Fig 3.1:** Block diagram of system



**Fig 3.1.2:** Work Flow Of System

## CHAPTER-4

### HARDWARE REQUIREMENTS

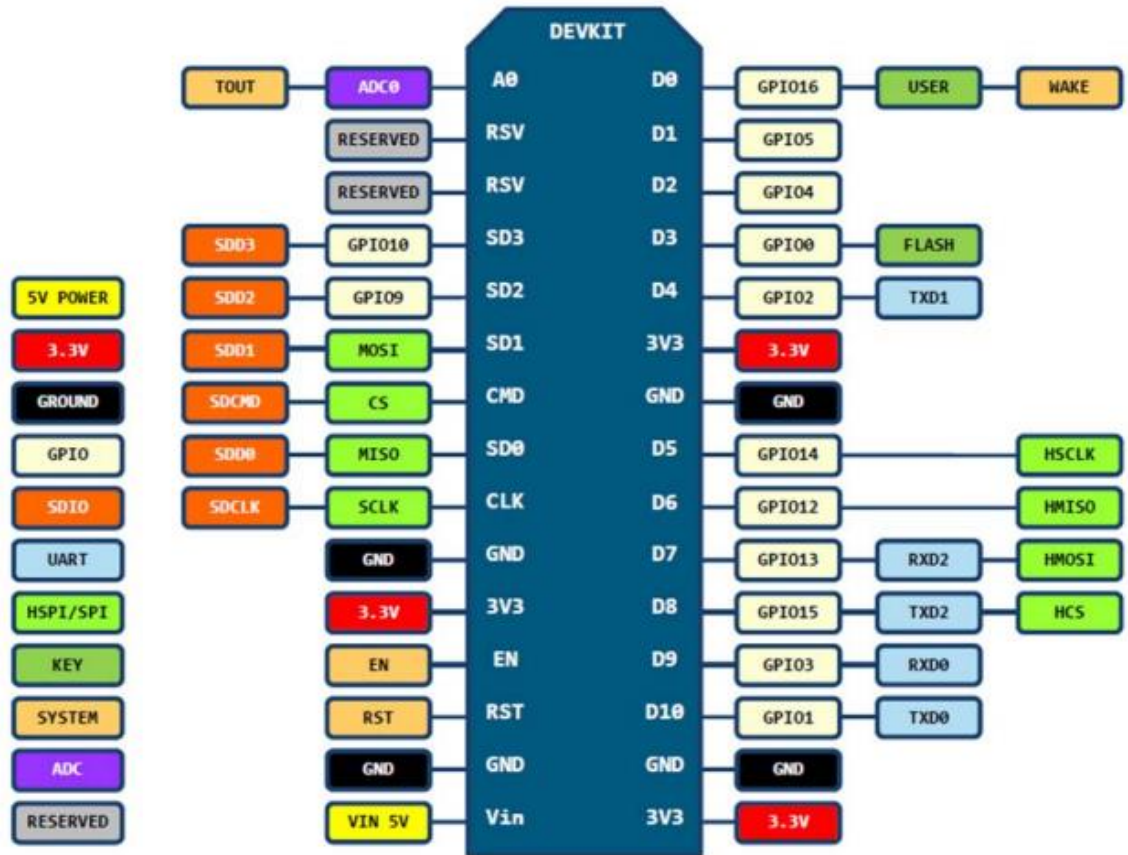
#### 4.1 NODEMCU

NodeMCU Development Kit/Board consist of ESP8266 wifi chip. ESP8266 chip has GPIO pins, serial communication protocol, etc. features on it. ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. The features of ESP8266 are extracted on NodeMCU Development board. NodeMCU (LUAbased firmware) with Development board/kit that consist of ESP8266 (wifi enabled chip) chip combines NodeMCU Development board which make it stand-alone device in IoT applications.

NodeMCU is an open source Lua based firmware for the ESP8266 Wi-Fi SOC from Espressif and uses an on-module flash-based SPIFFS file system. NodeMCU is implemented in C. The NodeMCU programming [7] model is similar to that of Node.js, only in Lua. It is asynchronous and event-driven. It is a single board microcontroller type operated on XTOS with ESP8266 CPU. It has a storage of 4MB and Memory 128KB. It has 12 GPIO pins which can act as input or output. It can be powered with USB or by using external voltage source. NodeMCU can act as an access point or it can be made to connect to an access point.

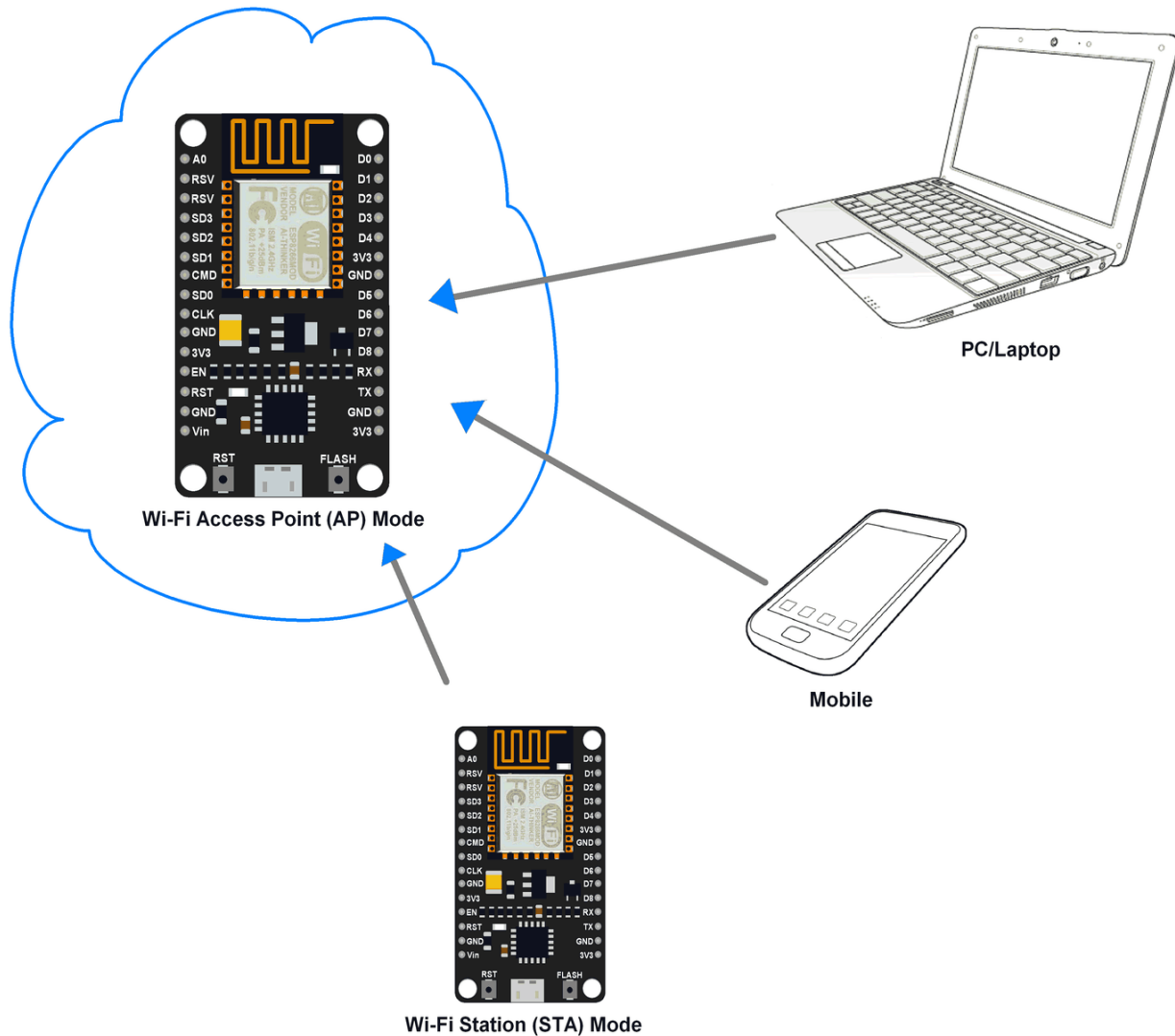
NodeMCU Dev Kit has **Arduino like** Analog (i.e. A0) and Digital (D0-D8) pins on its board. It supports serial communication protocols i.e. UART, SPI, I2C etc. Using such serial protocols we can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards etc.





**Fig 4.1:** NodeMCU pin diagram

NodeMCU is based on ESP8266 system on chip which combines the feature of Wi-Fi and microcontroller to satisfy the need of prototyping IoT applications within less time and with few lines of Lua Scripts. Wi-Fi is wireless LAN technology used for short distant wireless networking applications. It is based on IEEE 802.11 standards. NodeMCU firmware provides Event driven APIs for network applications. NodeMCU wi-fi networking can be used to connect, fetch or upload data to internet. NodeMCU Wi-Fi subsystem is running in background tasks periodically. If any function takes more than 15 milliseconds, it may cause the wi-fi subsystem to crash. To handle such functionalities NodeMCU has their APIs through which we can control this subsystem.



**Fig 4.2:** NodeMCU Wi-Fi Station and Access Point Mode

NodeMCU has four Wi-Fi modes as,

#### 1. Station (STA) Mode:

In this mode, NodeMCU joins the existing networks. The NodeMCU connects the existing wi-fi router. This provides the internet access to the NodeMCU through the wi-fi router.

#### 2. Access Point (AP) Mode:

In this mode, NodeMCU creates its own network and others can join this network. Here it has local IP address with which other devices can connect to it. NodeMCU assigns next available IPs to the other devices.

### 3. Station + Access Point (BOTH) Mode:

This is the mode, where it creates its own network while at the same time being joined to another existing network.

### 4. Wi-Fi OFF Mode:

In this mode, wi-fi remains OFF.

As shown in above Fig 4.2 NodeMCU (AP mode) creates Wireless LAN to which other Wi-Fi enabled devices like PC/Laptop, smart mobile phones, NodeMCU (STA mode) etc. can connect. NodeMCU (AP mode) assigns local IP address to each device connected to it.

#### 4.1.1 Specifications:

- Voltage: 3.3V
- Wi-Fi Direct (P2P), soft-AP
- Current consumption: 10uA~170mA
- Flash memory attachable: 16MB max (512K normal)
- Integrated TCP/IP protocol stack
- Processor: Tensilica L106 32-bit
- Processor speed: 80~160MHz
- RAM: 32K + 80K. • GPIOs: 17 (multiplexed with other functions)
- Analog to Digital: 1 input with 1024 step resolution
- +19.5dBm output power in 802.11b mode
- 802.11 support: b/g/n
- Maximum concurrent TCP connections: 5

#### 4.1.2 Features

- Finally, programable WiFi module.
- Arduino-like (software defined) hardware IO.
- Can be programmed with the simple and powerful Lua programming language or Arduino IDE.
- USB-TTL included, plug & play.

- 10 GPIOs D0-D10, PWM functionality, IIC and SPI communication, 1-Wire and ADC A0 etc. all in one board.
- Wifi networking (can be used as access point and/or station, host a web server), connect to internet to fetch or upload data.
- Event-driven API for network applications.
- PCB antenna.

#### **4.1.3 Working with NODEMCU:**

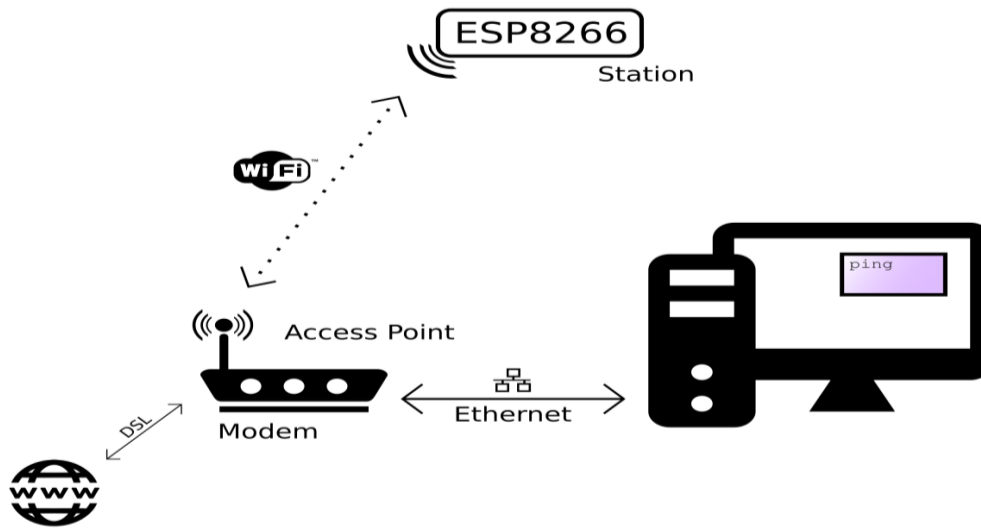
1. NODEMCU can be easily used Arduino IDE and install NodeMCU libraries using the link [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json).
2. Open Arduino IDE go to tools menu, then select NodeMCU board and port and set the baud rate as per the latest NodeMCU specification.
3. Include “ESP8266WiFi.h” in the program.

#### **4.1.4 Connection setup:**

1. NodeMCU can be programmed to connect to a network by providing SSID and password, whenever NodeMCU detect the network with the specified SSID, then nodeMCU connects to that network.
2. When NodeMCU connects to a network an IP address is generated and this ip address is the main thing to locate the ESP8266 over the entire network

#### **4.1.5 Communication with NodeMCU:**

1. NodeMCU can communicate to the devices connected over the entire network using ip address of other devices similarly other devices can communicate with NodeMCU with its IP address.
2. NodeMCU can support HTTP protocol. NodeMCU can support various types of requests of HTTP protocol but mainly GET and POST requests.



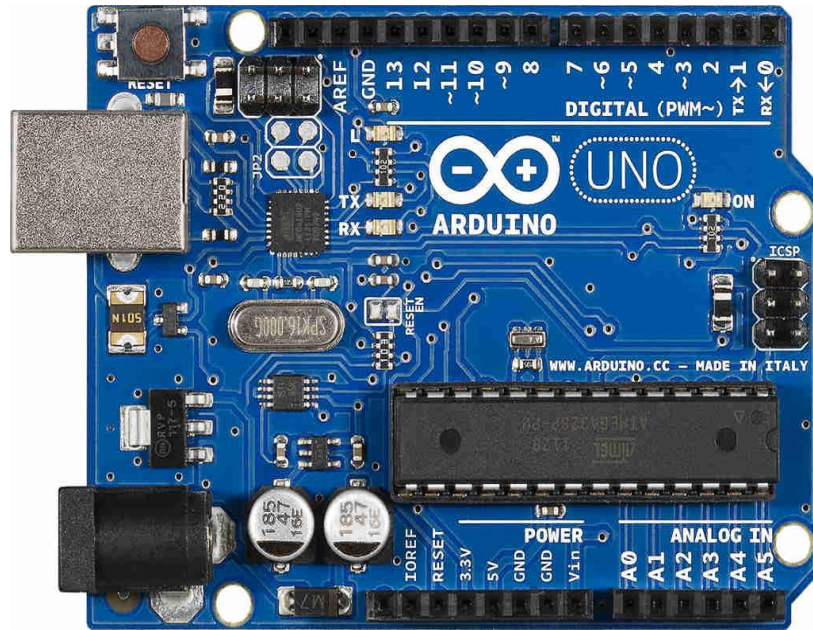
**Fig 4.3:** Communication between Server and NodeMCU

## 4.2 ARDUINO UNO

Arduino Uno is a microcontroller board developed by Arduino.cc which is an open-source electronics platform mainly based on AVR microcontroller Atmega328. The current version of Arduino Uno comes with USB interface, 6 analog input pins, 14 I/O digital ports that are used to connect with external electronic circuits. Out of 14 I/O ports, 6 pins can be used for PWM output. It allows the designers to control and sense the external electronic devices in the real world. This board comes with all the features required to run the controller and can be directly connected to the computer through USB cable that is used to transfer the code to the controller using IDE (Integrated Development Environment) software, mainly developed to program Arduino. IDE is equally compatible with Windows, MAC or Linux Systems, however, Windows is preferable to use. Programming languages like C and C++ are used in IDE. Apart from USB, battery or AC to DC adopter can also be used to power the board.

The UNO board can be powered through

- A USB cable
- An external AC-DC adapter (output voltage of the adapter must be fixed and within the range of 7V to 12V). Adapter needs to be plugged into the power jack.
- A battery (Fixed voltage, the voltage must be in the range of 7V to 12V). Battery terminals must be connected between VIN and GND pins on the board.



**Fig 4.4:** Arduino Uno board

The above Fig 4.4 shows the Arduino UNO R3 board. Arduino UNO R3 uses ATmega328P microcontroller as the main control and processing unit of the board. The ATmega328P has 32kB In-System Programmable Flash Memory, 1kB EEPROM, and 2kB internal SRAM. The UNO board provides access to 14 Digital I/O pins (0-13 in above image). Of these 14 digital I/O pins, 6 pins can be used as PWM pins (pins 3, 5, 6, 9, 10 and 11). The PWM pins are indicated by ~ before the pin number on the Arduino UNO boards (E.g. ~3). Digital pins 0 and 1 are the serial communication pins RX and TX respectively. 6 pins are available for use as Analog input pins (A0-A5). The analog pins can be used as digital I/O pins as well if there is no analog input to be measured.

Arduino Uno boards are quite similar to other boards in Arduino family in terms of use and functionality, however, Uno boards don't come with FTDI USB to Serial driver chip. There are many versions of Uno boards available, however, Arduino Nano V3 and Arduino Uno are the most official versions that come with ATmega328 8-bit AVR Atmel microcontroller where RAM memory is 32KB. When nature and functionality of the task go complex, Micro SD card can be added in the boards to make them store more information.

The below Table 4.1 gives the description of each pin in Arduino Uno.

Pin Category	Pin Name	Details
Power	Vin, 3.3V, 5V, GND	<p>Vin: Input voltage to Arduino when using an external power source.</p> <p>5V: Regulated power supply used to power microcontroller and other components on the board.</p> <p>3.3V: 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA.</p> <p>GND: ground pins.</p>
Reset	Reset	Resets the microcontroller.
Analog Pins	A0 – A5	Used to provide analog input in the range of 0-5V
Input/ Output Pins	Digital Pins 0 - 13	Can be used as input or output pins.
Serial	0(Rx), 1(Tx)	Used to receive and transmit TTL serial data.
External Interrupts	2, 3	To trigger an interrupt.
PWM	3, 5, 6, 9, 11	Provides 8-bit PWM output.
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Used for SPI communication.
Inbuilt LED	13	To turn on the inbuilt LED.
TWI	A4 (SDA), A5 (SCA)	Used for TWI communication.
AREF	AREF	To provide reference voltage for input voltage.

**Table 4.1:** Arduino Uno pin description

Arduino Uno comes with an ability of interfacing with other Arduino boards, microcontrollers and computer. The Atmega328 placed on the board provides serial communication using pins like Rx and Tx. The Atmega16U2 incorporated on the board provides



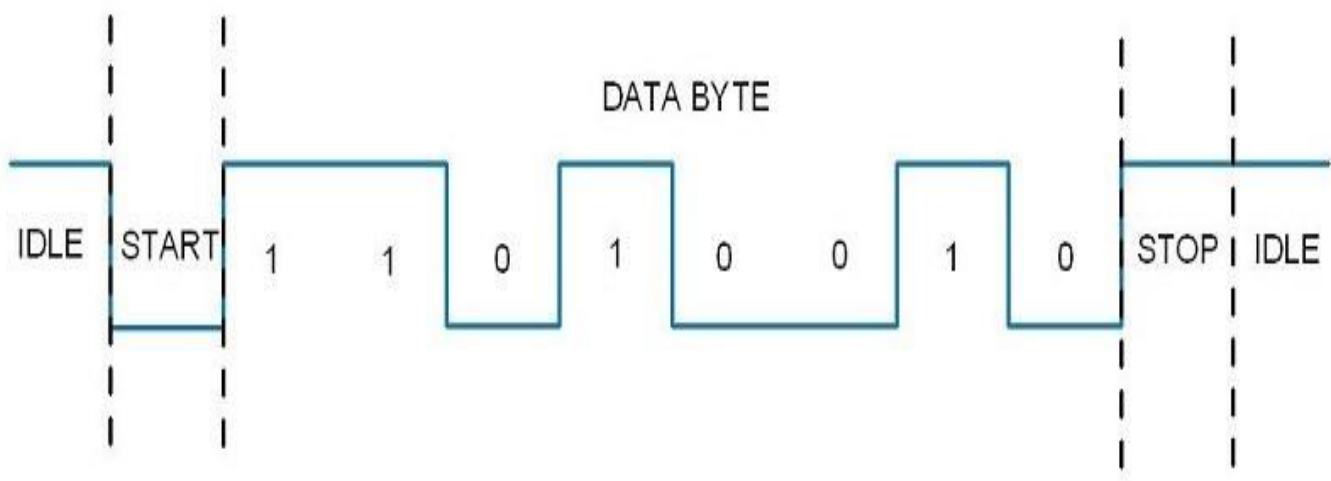
a pathway for serial communication using USB com drivers. Serial monitor is provided on the IDE software which is used to send or receive text data from the board.

#### 4.2.1 USART in Arduino Uno:

USART (Universal Serial Asynchronous Receiver Transmitter) is a serial communication protocol used to transmit/receive data serially at a specific baud rate. Several devices such as GPS, GSM, RFID, sensors, etc need to communicate with the microcontroller for transmitting or receiving information. To communicate with the microcontroller, several communication protocols are used such as RS232, SPI, I2C, CAN etc.

With the help of USART, we can send / receive data to a computer or other devices. USART is also used in interfacing microcontroller with various modules like Wi-Fi (ESP8266), Bluetooth, GPS, GSM, etc.

**Asynchronous Communication:** Asynchronous operation means that a process operates independently of other processes. Asynchronous communication means each character (data byte) is placed in between the start and stop bits. The start bit is always 0 (low) and the stop bit is always 1 (high).



**Fig 4.5:** Basic Frame Structure

**Bit Rate & Baud Rate:** The rate of data transfer in serial data communication is stated in bps (bits per second). Another widely used terminology for bps is baud rate; means, number of changes in signal per second. Here the signal is in bits, therefore bit rate = baud rate.



### 4.3 GSM MODULE

Global System for Mobile communication (GSM) is digital cellular system used for mobile devices. It is an international standard for mobile which is widely used for long distance communication. There are various GSM modules available in market like SIM900, SIM700, SIM800, SIM808, SIM5320 etc. SIM900A module allows users to send/receive data over GPRS, send/receive SMS and make/receive voice calls. The GSM/GPRS module uses USART communication to communicate with microcontroller or PC terminal. AT commands are used to configure the module in different modes and to perform various functions like calling, posting data to a site, etc.

GSM/GPRS module is used to establish communication between a computer and a GSM-GPRS system. Global System for Mobile communication (GSM) is an architecture used for mobile communication in most of the countries. Global Packet Radio Service (GPRS) is an extension of GSM that enables higher data transmission rate. GSM/GPRS module consists of a GSM/GPRS modem assembled together with power supply circuit and communication interfaces (like RS-232, USB, etc) for computer. GSM/GPRS MODEM is a class of wireless MODEM devices that are designed for communication of a computer with the GSM and GPRS network. It requires a SIM (Subscriber Identity Module) card just like mobile phones to activate communication with the network. Also they have IMEI (International Mobile Equipment Identity) number similar to mobile phones for their identification. A GSM/GPRS MODEM can perform the following operations:

1. Receive, send or delete SMS messages in a SIM.
2. Read, add, search phonebook entries of the SIM.
3. Make, Receive, or reject a voice call.

The MODEM needs AT commands, for interacting with processor or controller, which are communicated through serial communication. These commands are sent by the controller/processor. The MODEM sends back a result after it receives a command. Different AT commands supported by the MODEM can be sent by the processor/controller/computer to interact with the GSM and GPRS cellular network.

For the purpose of project, GSM Sim900A (as shown in Fig 4.5) is used and its features are:

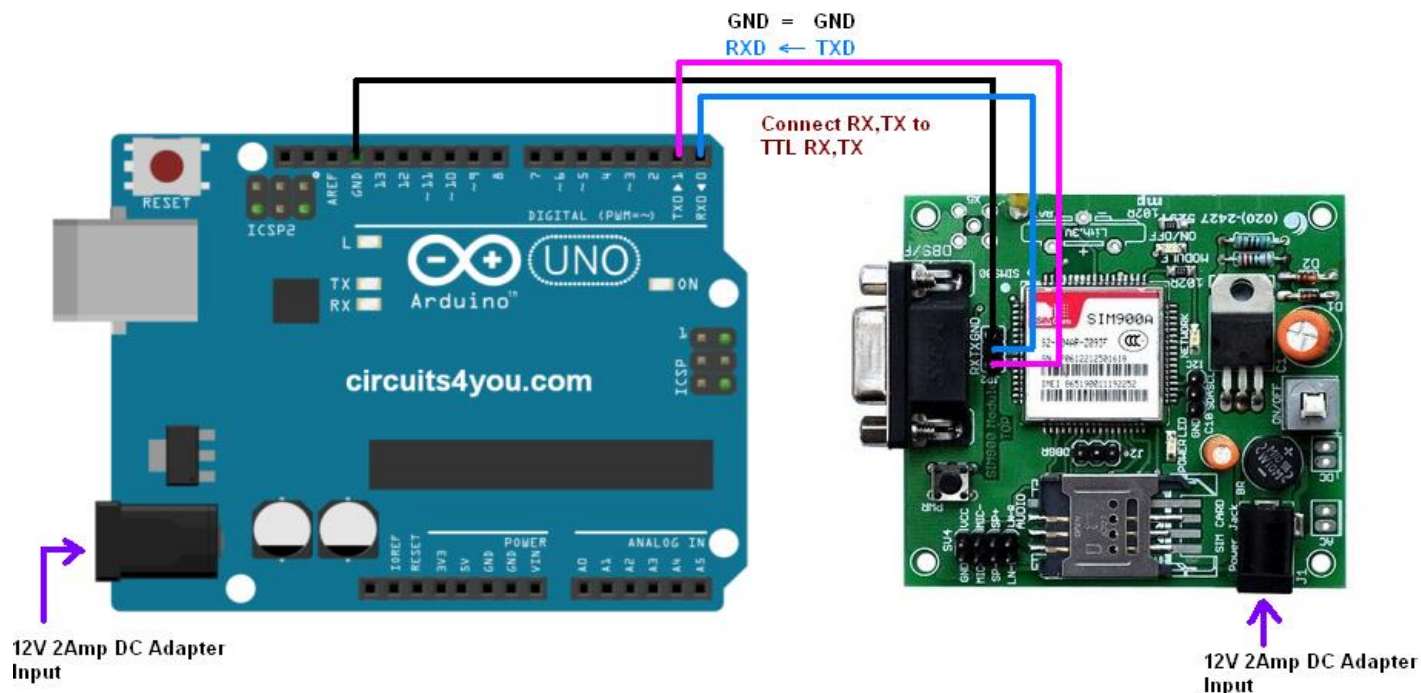
- Quad-Band 850/ 900/ 1800/ 1900 MHz
- Dual-Band 900/ 1900 MHz
- GPRS multi-slot class 10/8GPRS mobile station class B
- Compliant to GSM phase 2/2+Class 4 (2 W @850/ 900 MHz)
- Class 1 (1 W @ 1800/1900MHz)
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- Low power consumption: 1.5mA(sleep mode)
- Operation temperature: -40°C to +85 °C



**Fig 4.6:** GSM Sim900A module

#### **4.3.1 Connection of GSM Module with Arduino Uno:**

Connecting GSM modem with Arduino Uno is very simple just connect RX Line of Arduino to TX Line of GSM Modem and vice versa i.e. TX of Arduino Uno to Rx of GSM modem. Make sure use TTL RX, TX lines of GSM modem. Give 12V 2Amp power supply to GSM modem, Use of less current power supply can cause reset problem in GSM modem, give sufficient current to GSM modem.



**Fig 4.7:** Arduino Uno and GSM module interface diagram

#### 4.3.2 AT Command

There some basic AT commands which are used as follows,

Command	Description	Response
AT	Checking communication	OK
ATE0 or ATE1	Turn off /ON echo	OK
ATI	Product Identification	e.g.: SIM900 R11.0 OK
AT+GSN	Product serial number identification (IMEI)	IMEI no.
AT+GMI	Manufacturer Name	Manufacturer Name E.g. SIMCOM_LTD
AT+GMM	Model Identification	Model no. E.g. SIMCOM_900A

AT+CSMINS?	SIM Inserted Status Reporting	+CSMINS: <n>, <SIM inserted>  0: not inserted  1: inserted
AT+CSPN?	Get Service Provider Name from SIM	Service Provider Name  E.g. +CSPN: “!dea”, 1

**Table 4.2:** Basic AT commands

To use calling service following AT commands are used.

Command	Description	Response
ATD<Mob. No.>;	Dial / call a number	OK (if successful), BUSY (if busy), NO CARRIER (if no connection)
ATA	Answer a call	OK
ATH	Hang up call	OK
ATDL	Redial Last Telephone Number Used	OK (if successful), BUSY (if busy), NO CARRIER (if no connection)

**Table 4.3:** AT commands for calling via GSM module

To use SMS service following AT commands are used.

Command	Description	Response
AT+CMGF=<index> index- 0: PDU     1: Text	Select message format	OK
AT+CMGS=”9881xxxxxx”	Send message	> ”Type message here” press ‘ctrl+z’ to end msg or ‘ESC’ to exit without sending  OK
AT+CMGR=<index>	Read message at that index	+CMGR: “Message Header”  Message Body  OK

AT+CMGD=<index>	Delete message at that index	OK (if present at that index)
AT+CMGDA="DEL ALL"	Delete all SMS	OK

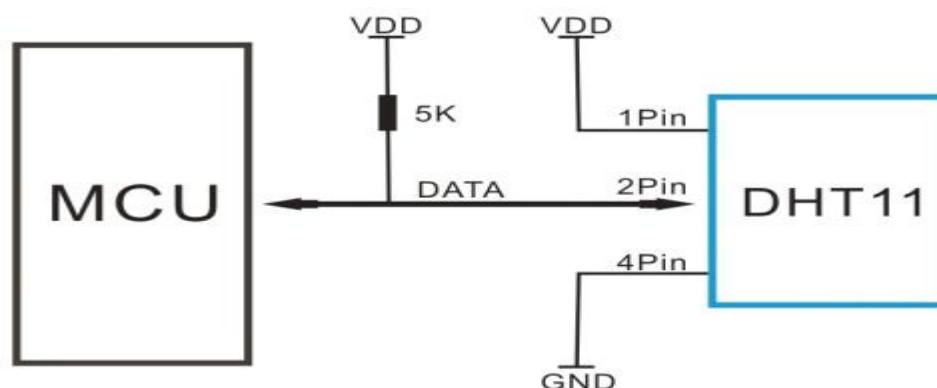
**Table 4.4:** AT commands sending SMS via GSM module

#### 4.4 TEMPERATURE AND HUMIDITY SENSOR (DHT 11):

Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high performance 8-bit microcontroller [8], offering excellent quality, fast response, anti-interference ability and cost-effectiveness.

The DHT11 is a basic, low cost digital temperature and humidity sensor.

- DHT11 is a single wire digital humidity and temperature sensor, which provides humidity and temperature values serially with one-wire protocol.
- DHT11 sensor provides relative humidity value in percentage (20 to 90% RH) and temperature values in degree Celsius (0 to 50 °C).
- DHT11 sensor uses resistive humidity measurement component, and NTC temperature measurement component.



**Fig 4.8:** Interfacing with NodeMCU

Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package
DHT11	20-90%RH 0-50 °C	± 5%RH	± 2°C	1	4 Pin Single Row

**Table 4.5:** Specifications of DHT 11

#### 4.5.1 Characteristics:

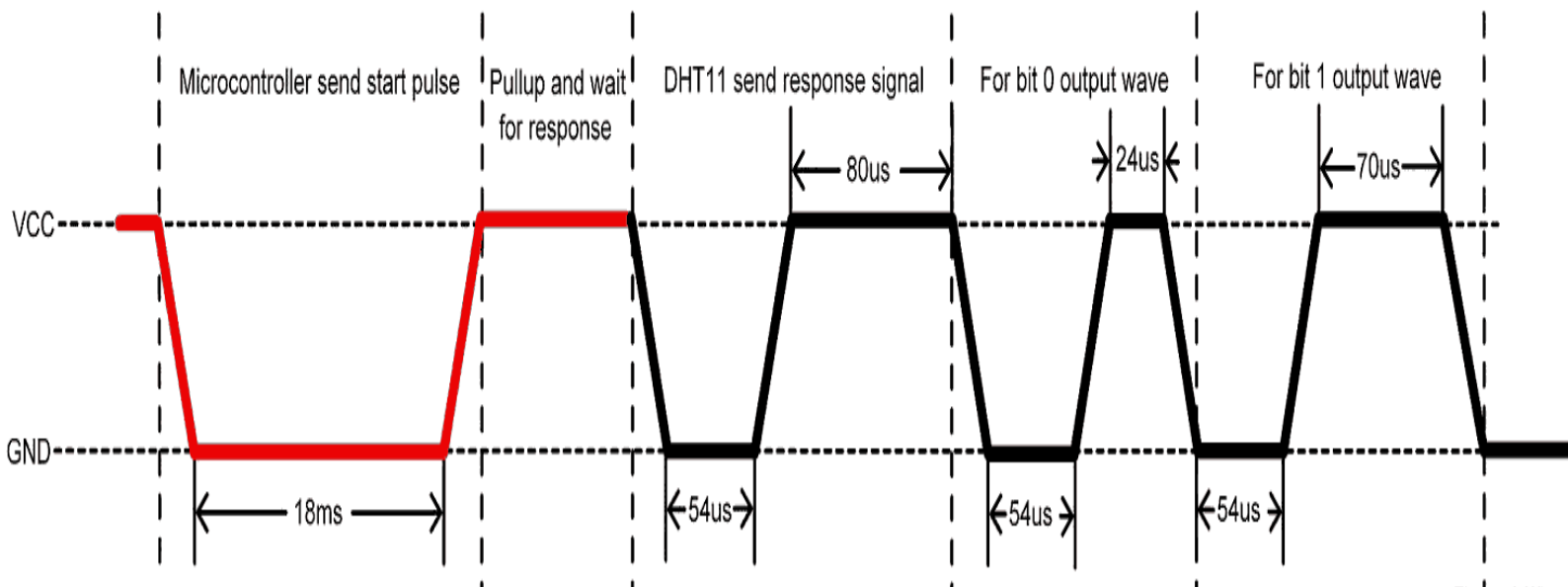
- Ultra-low cost
- to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50°C temperature readings  $\pm 2^{\circ}\text{C}$  accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- pins with 0.1" spacing

DHT11 is a part of DHTXX series of Humidity sensors. The other sensor in this series is DHT22. Both these sensors are Relative Humidity (RH) Sensor. As a result, they will measure both the humidity and temperature. Although DHT11 Humidity Sensors are cheap and slow, they are very popular among hobbyists and beginners. The DHT11 Humidity and Temperature Sensor consists of 3 main components. A resistive type humidity sensor, an NTC (negative temperature coefficient) thermistor (to measure the temperature) and an 8-bit microcontroller, which converts the analog signals from both the sensors and sends out single digital signal.

DHT11 Humidity Sensor consists of 4 pins: VCC, Data Out, Not Connected (NC) and GND. The range of voltage for VCC pin is 3.5V to 5.5V. A 5V supply would do fine. The data from the Data Out pin is a serial digital data. The following image shows a typical application circuit for DHT11 Humidity and Temperature Sensor. DHT11 Sensor can measure a humidity value in the range of 20 – 90% of Relative Humidity (RH) and a temperature in the range of 0 – 50°C.

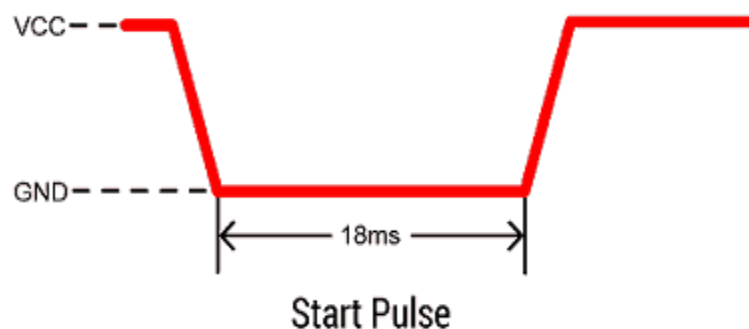
### 4.5.2 Communication with Microcontroller

- DHT11 uses only one wire for communication. The voltage levels with certain time value defines the logic one or logic zero on this pin.
- The communication process is divided in three steps, first is to send request to DHT11 sensor then sensor will send response pulse and then it starts sending data of total 40 bits to the microcontroller.



**Fig 4.9:** Communication process

#### Start pulse (Request)



**Fig 4.10:** Request

- To start communication with DHT11, first we should send the start pulse to the DHT11 sensor.

- To provide start pulse, pull down (low) the data pin minimum 18ms and then pull up, as shown in diag.

### Response

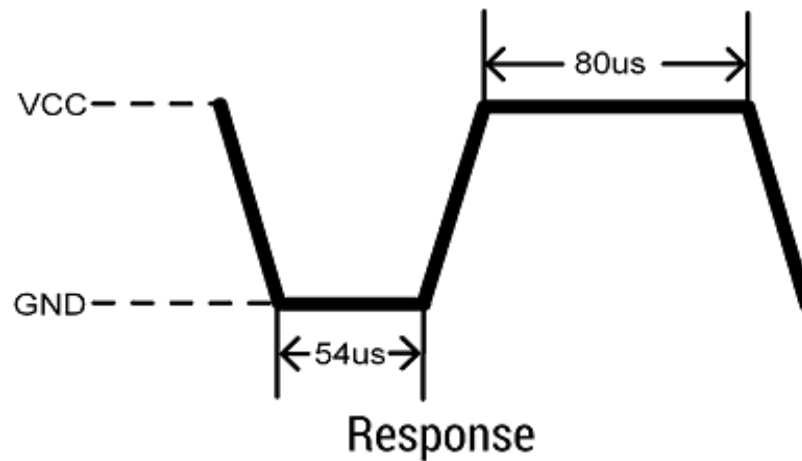


Fig 4.12: Response

- After getting start pulse from, DHT11 sensor sends the response pulse which indicates that DHT11 received start pulse.
- The response pulse is low for 54us and then goes high for 80us.

### Data

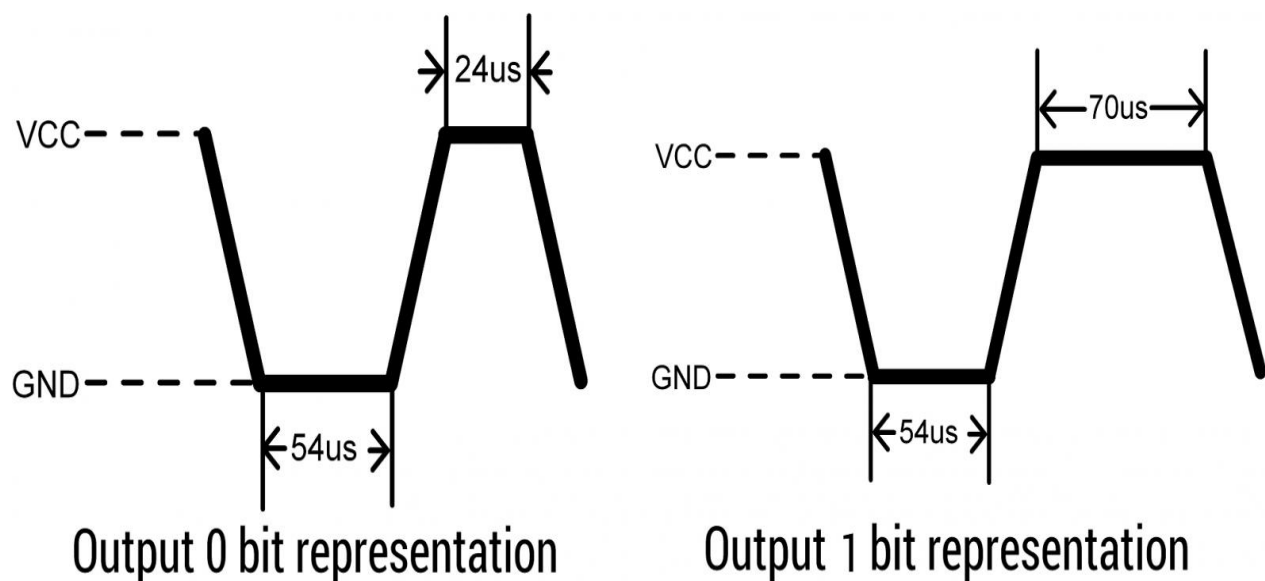
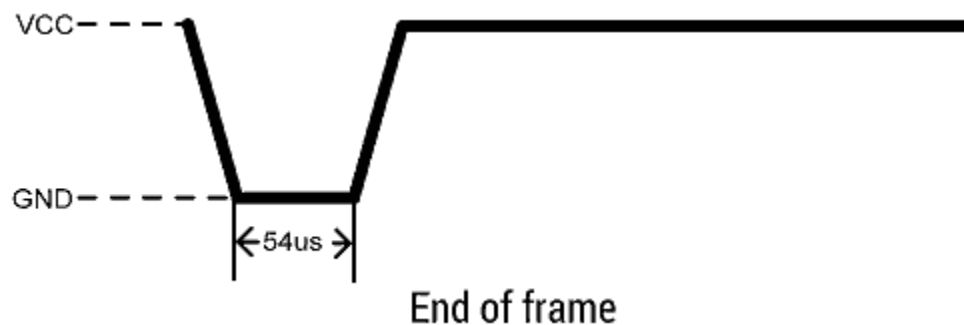


Fig 4.13: Data representation



- After sending the response pulse, DHT11 sensor sends the data, which contains humidity and temperature value along with checksum.
- The data frame is of total 40 bits long, it contains 5 segments (byte) and each segment is 8-bit long.
- In these 5 segments, first two segments contain humidity value in decimal integer form. This value gives us Relative Percentage Humidity. 1st 8-bits are integer part and next 8 bits are fractional part.
- Next two segments contain temperature value in decimal integer form. This value gives us temperature in Celsius form.
- Last segment is the checksum which holds checksum of first four segments.
- Here checksum byte is direct addition of humidity and temperature value. And we can verify it, whether it is same as checksum value or not. If it is not equal, then there is some error in the received data.
- Once data received, DHT11 pin goes in low power consumption mode till next start pulse.

#### End of frame



**Fig 4.14:** End of frame

- After sending 40-bit data, DHT11 sensor sends 54us low level and then goes high. After this DHT11 goes in sleep mode.

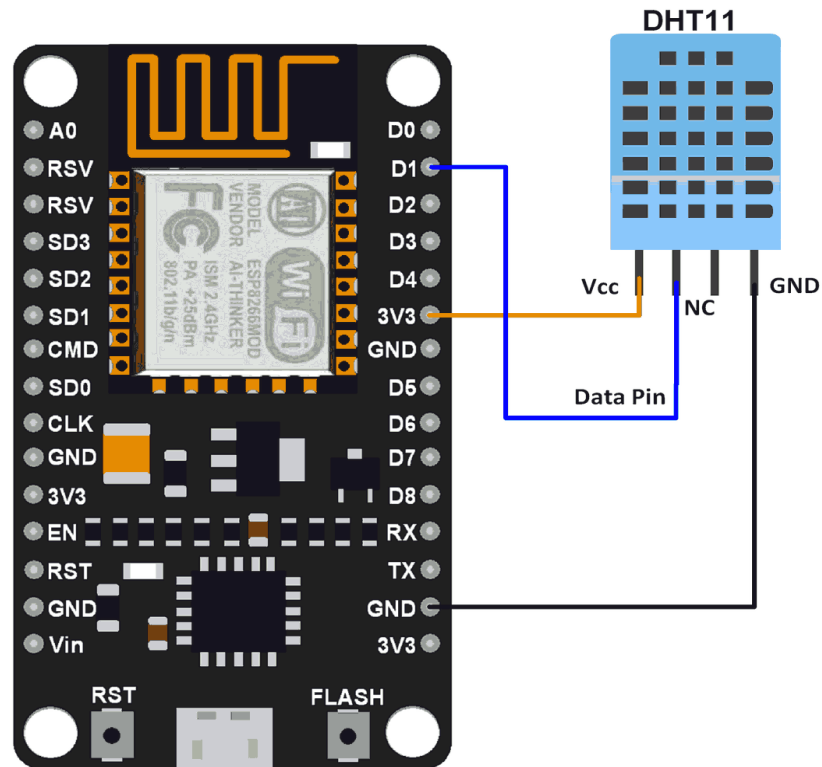


Fig 4.15: DHT11 interface with NodeMCU

## 4.6 RELAY BOARD

A relay is an electromagnetic switching device consisting of an armature which is moved by an electromagnet to operate one or more switch contacts. Some advantages of relays are that they provide amplification and isolation and are straight forward. Here we are using 5v 4-channel relay interface board, and each channel needs a 15-20mA driver current. it can be used to control various appliances and equipment with large current relays that work under AC250V 10A or DC30V 10A. it has a standard interface that can be controlled directly by microcontroller.

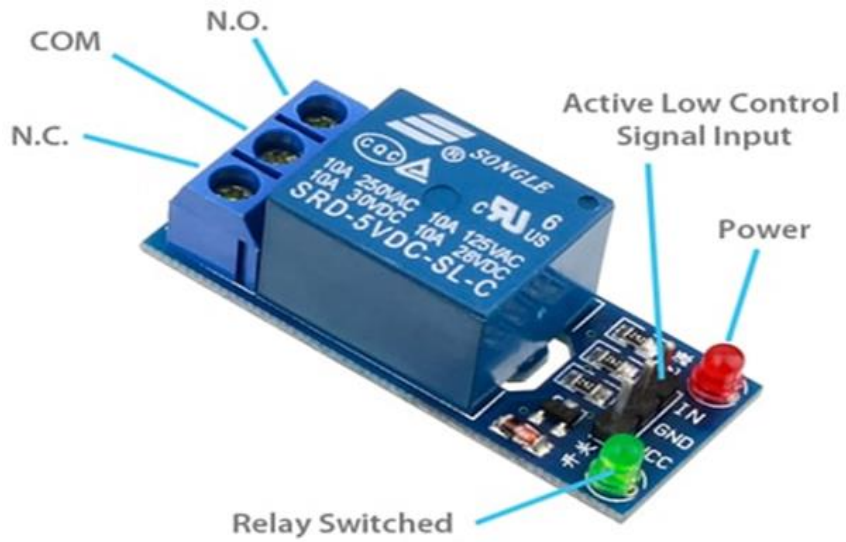
Pin Number	Pin Name	Description
1	Coil End 1	Used to trigger(On/Off) the Relay, Normally one end is connected to 5V and the other end to ground

2	Coil End 2	Used to trigger(On/Off) the Relay, Normally one end is connected to 5V and the other end to ground
3	Common (COM)	Common is connected to one End of the Load that is to be controlled
4	Normally Close (NC)	The other end of the load is either connected to NO or NC. If connected to NC the load remains connected before trigger
5	Normally Open (NO)	The other end of the load is either connected to NO or NC. If connected to NO the load remains disconnected before trigger

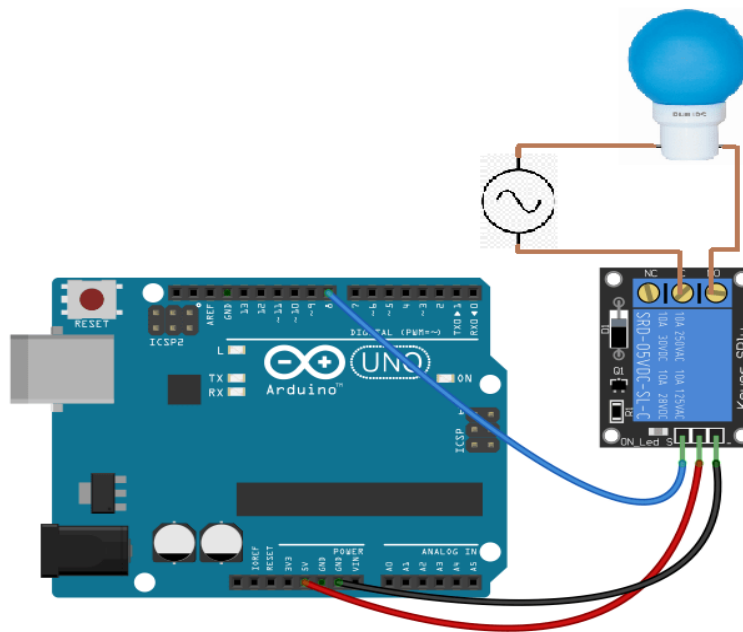
**Table 4.3:** Relay pin configuration

#### **4.6.1 Features of 5-Pin 5V Relay**

- Trigger Voltage (Voltage across coil) : 5V DC
- Trigger Current (Nominal current) : 70mA
- Maximum AC load current: 10A @ 250/125V AC
- Maximum DC load current: 10A @ 30/28V DC
- Compact 5-pin configuration with proper insulation
- Operating time: 10msec Release time: 5msec
- Maximum switching: 300 operating/minute (mechanically)



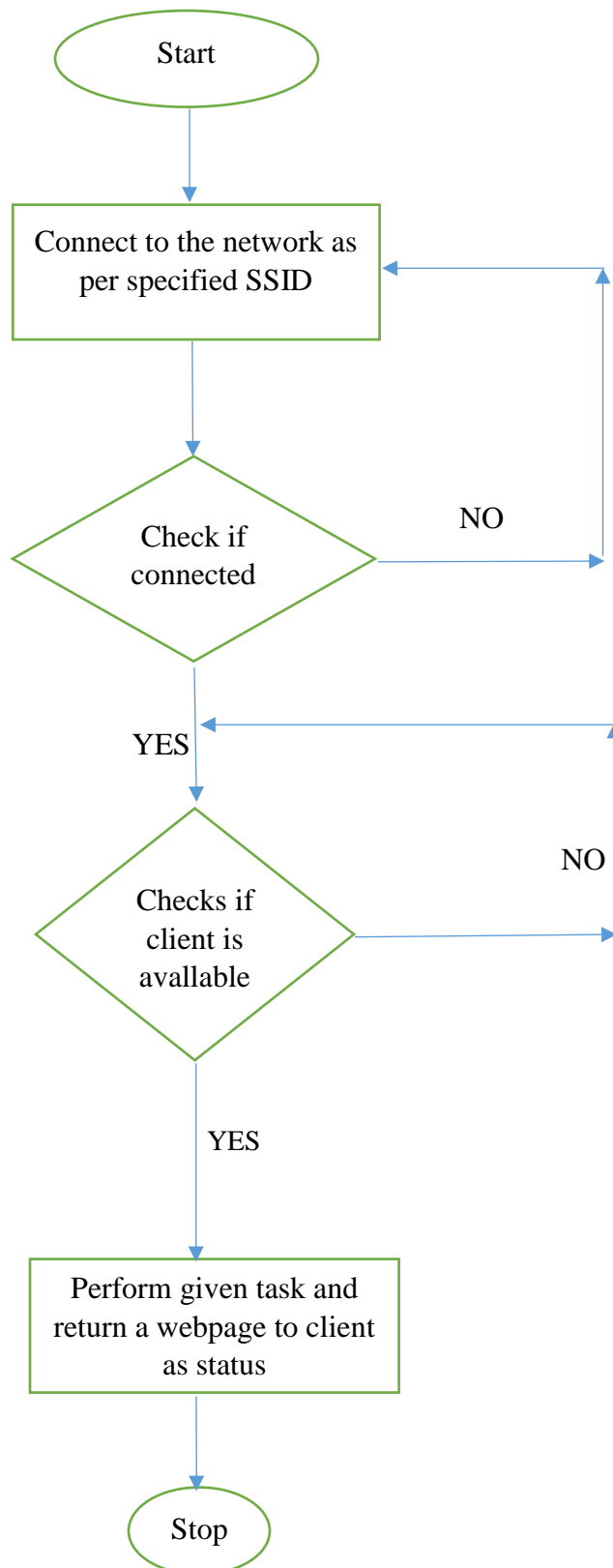
**Fig 4.16:** Relay Board



**Fig 4.17:** Relay Board interface with Arduino Uno and appliance

## 4.7 FLOWCHART AND ALGORITHM

### 4.7.1 Flow chart



#### **4.3.2 Algorithm of NodeMCU:**

1. Start
2. Connect NodeMCU to the given network using SSID and password
3. Checks for the client requests
4. If requests available process the requests, if not go to step 2.
5. Perform the task as per request and return a web page to client.
6. End

Note: Refer to file -1 in appendix for NodeMCU code

## CHAPTER-5

### SOFTWARE REQUIREMENTS

#### 5.1 XAMPP PACK

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

<b>X</b>	Refers To Cross-Platform
<b>A</b>	Apache or its expanded form, Apache HTTP Server
<b>M</b>	MariaDB (formerly: MySQL)
<b>P</b>	PHP
<b>P</b>	PERL

**Table 5.1:** XAMPP Abbreviation

Officially, XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their own computers without any access to the Internet. To make this as easy as possible, many important security features are disabled by default. XAMPP has the ability to serve web pages on the World Wide Web. A special tool is provided to password-protect the most important parts of the package. XAMPP also provides support for creating and manipulating databases in MariaDB and SQLite among others.

Once XAMPP is installed, it is possible to treat a localhost like a remote host by connecting using an FTP client. Using a program like FileZilla has many advantages when installing a content management system (CMS) like Joomla or WordPress. It is also possible to connect to localhost via FTP with an HTML editor.

### **5.1.1 Working with XAMPP:**

1. After installation of XAMPP pack in the PC, XAMPP control panel can be found in PC, where different Servers, Database, PHP and PerL can be found installed. One can start and stop those servers.
2. Once a server started in the PC, a port is assigned to the Server and Server runs in that port.
3. We can execute all application deployed in the server in the following way:
  - i) Start server
  - ii) Open browser
    - (1) Type <http://> then type servername followed by colon (":") followed by port number and then type application name.
    - (2) Example: let servername is "localhost" and portnumber is "80" and desired application name is "index.html" ,to execute that application type following in address bar of a browser  
<http://localhost:80/index.html>
    - (3) After Step 2 index.html page is displayed in browser.

## **5.2 ARDUINO IDE**

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino board. The Arduino IDE employs the program avrdude [9] to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

### **5.2.1 Working with Arduino IDE:**

1. Install Arduino IDE from Arduino cc official site.
2. Install required libraries from official site.

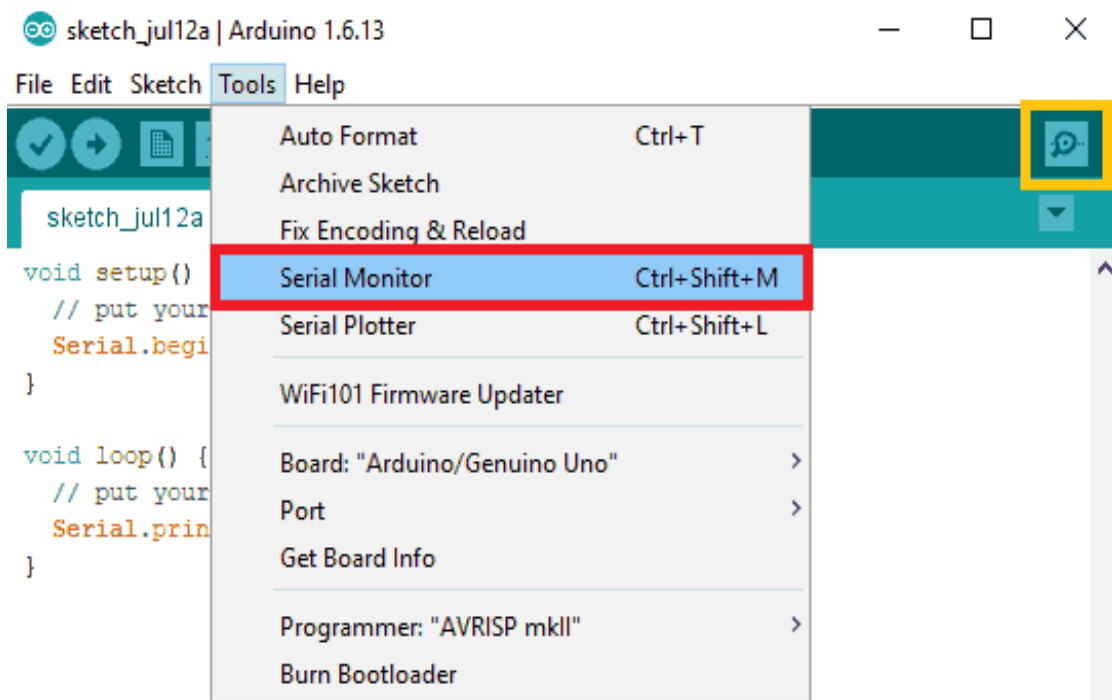


3. In the tools folder select the appropriate board and set the baud rate of board for perfect communication between PC & board.
4. See the output in the Serial monitor.

### 5.2.2 Arduino Serial Monitor:

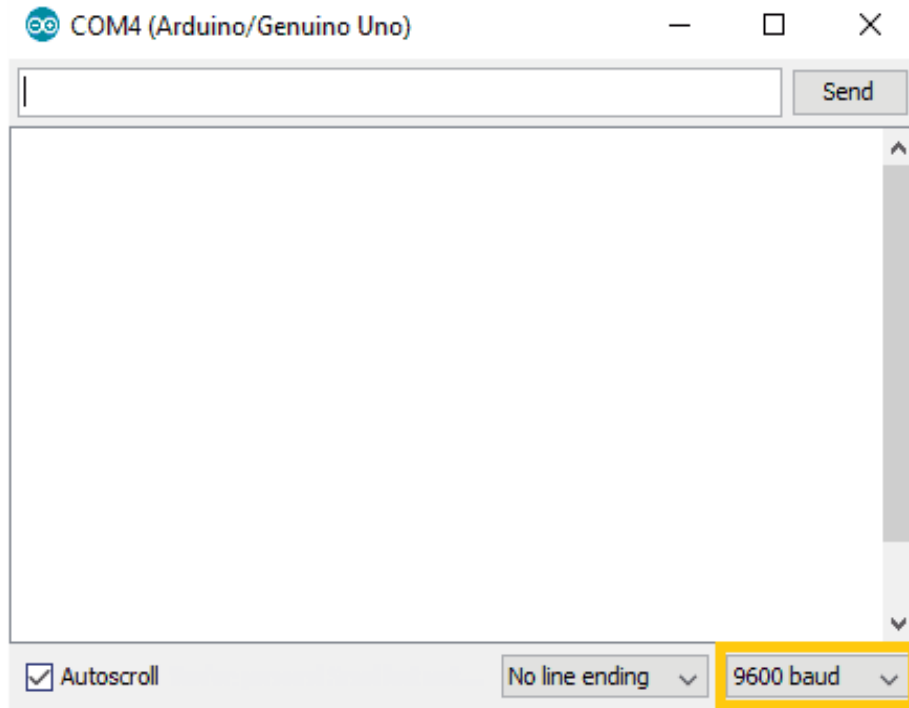
Arduino IDE has integrated serial monitor which can be used to watch serial data. After uploading program to the Arduino, open Serial Monitor to watch Serial data. Serial monitor as shown below in Fig 5.1 can be found the following path.

Go to **Tools --> Serial Monitor**



**Fig 5.1:** Opening of serial monitor in Arduino IDE

One can also access serial monitor by clicking on icon present on top-right corner of Arduino IDE shown in above image. After selecting Serial monitor, you will get serial monitor window as shown below in Fig 5.2,



**Fig 5.2:**Serial monitor window

One can change baud rate by selecting one of value highlighted in above image. If baud rate of two devices mismatched, then we will get nothing or garbage value.

### 5.2.3 Arduino Serial Plotter

Arduino Serial Plotter is a very cool feature present in Arduino ide. It visualizes the data receiving on computer serial port in the form of waveforms. It is very useful in visualizing data, debugging code and visualizing our variables as waveforms. Serial Plotter in Arduino IDE can be found as shown below in Fig 5.3

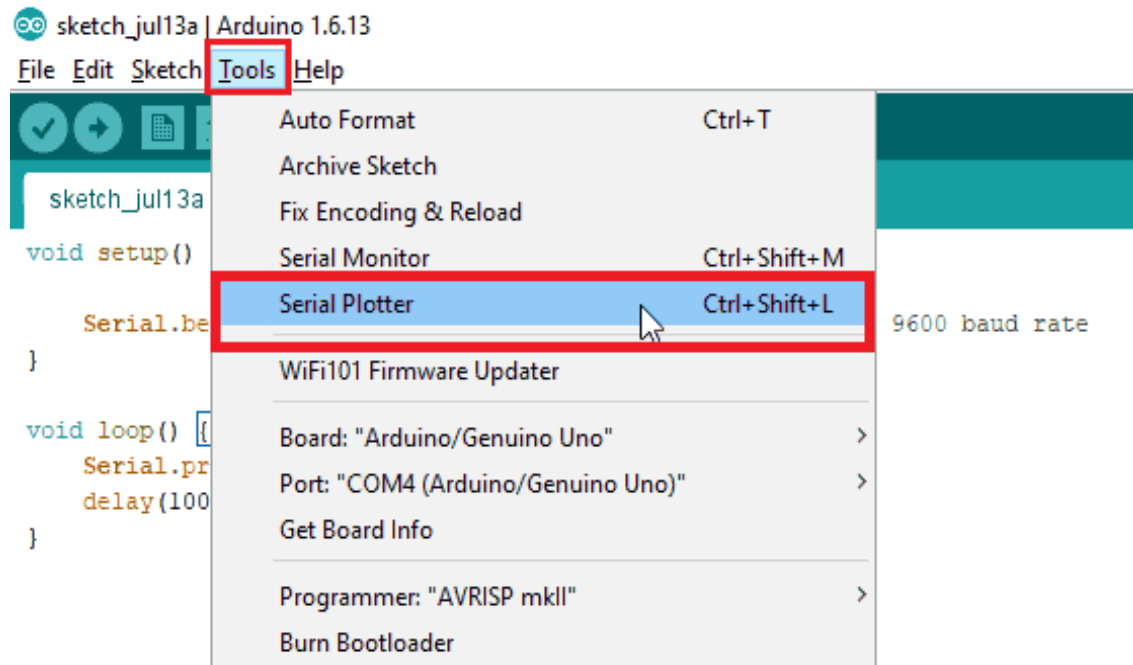


Fig 5.3: Path for Serial plotter in Arduino IDE

### 5.3 OTHER TOOLS

The entire project is developed by using the following:

1. HTML
2. CSS
3. JavaScript
4. JQuery
5. PHP

## CHAPTER-6

### SETTING UP THE PROJECT

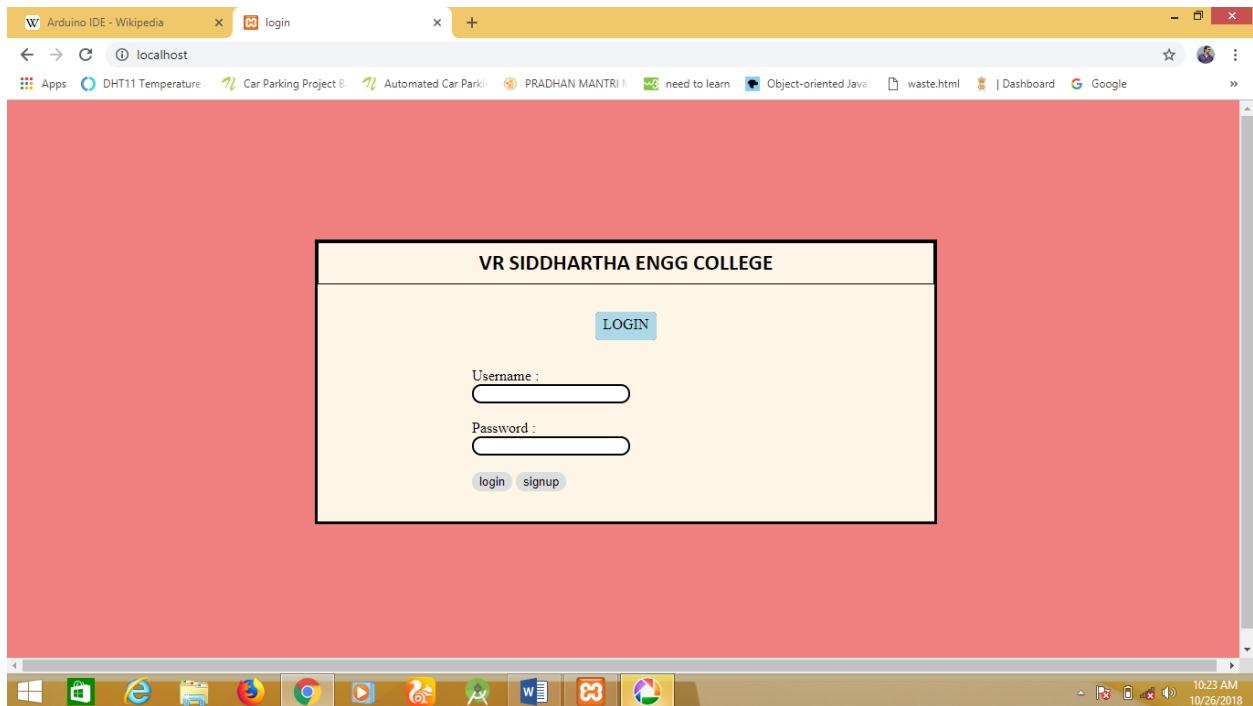
#### 6.1 STEP WISE PROCEDURE

Step 1: In this step, security for the entire system can be provided by Authentication system and it is prepared one for login and other for signup.

Step 2: In this step user entered data is checked with the database, if everything is correct then System control panel will be displayed.

Implementation of Step 1 and Step 2:

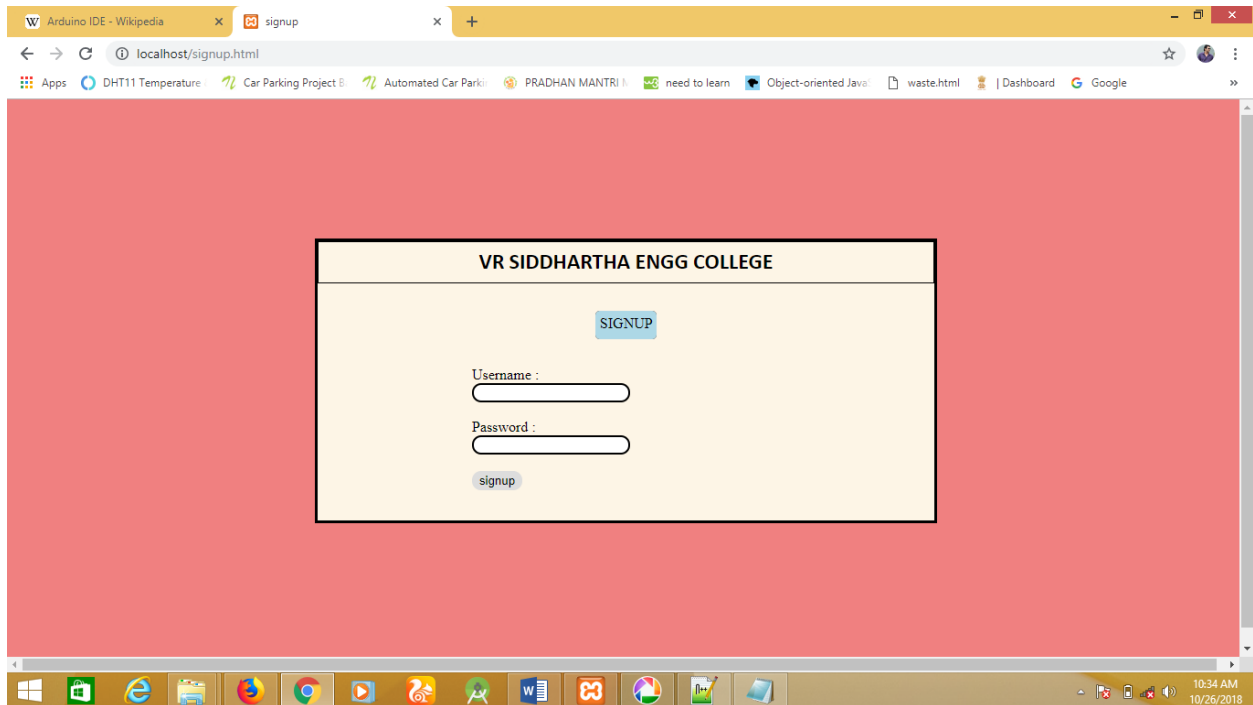
#### Preparing Login Form:



**Fig 6.1:** Login page

Refer index.php (File-2) in the appendix

The above file is to login into the application if user was already registered, if user is a new one ,then user must be signup an account for this Signup application must be designed as below:

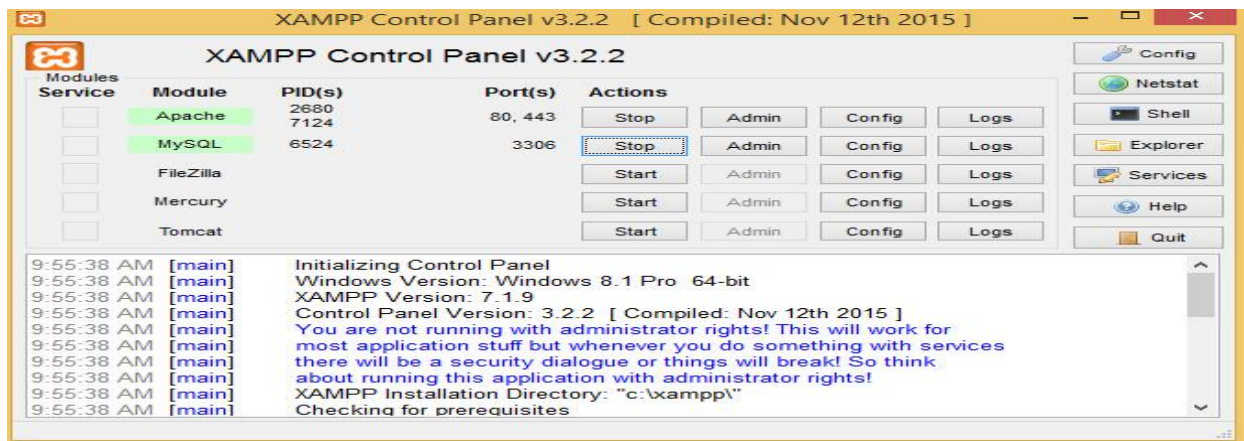


**Fig 6.2:** Signup page

Refer to File 3 in appendix

After entering the user details, when user clicks Signup button, a new account will be created and corresponding Username and Password are stored in a database. For this a database must be prepared and here are the steps to be followed to create a data base:

Step 1: Type <http://localhost/phpmyadmin> in the address bar after starting Apache and MYSQL in the XAMPP control panel.



**Fig 6.3:** XAMPP control panel

Step2: After entering the above URL in the address bar, then below screen will be displayed.

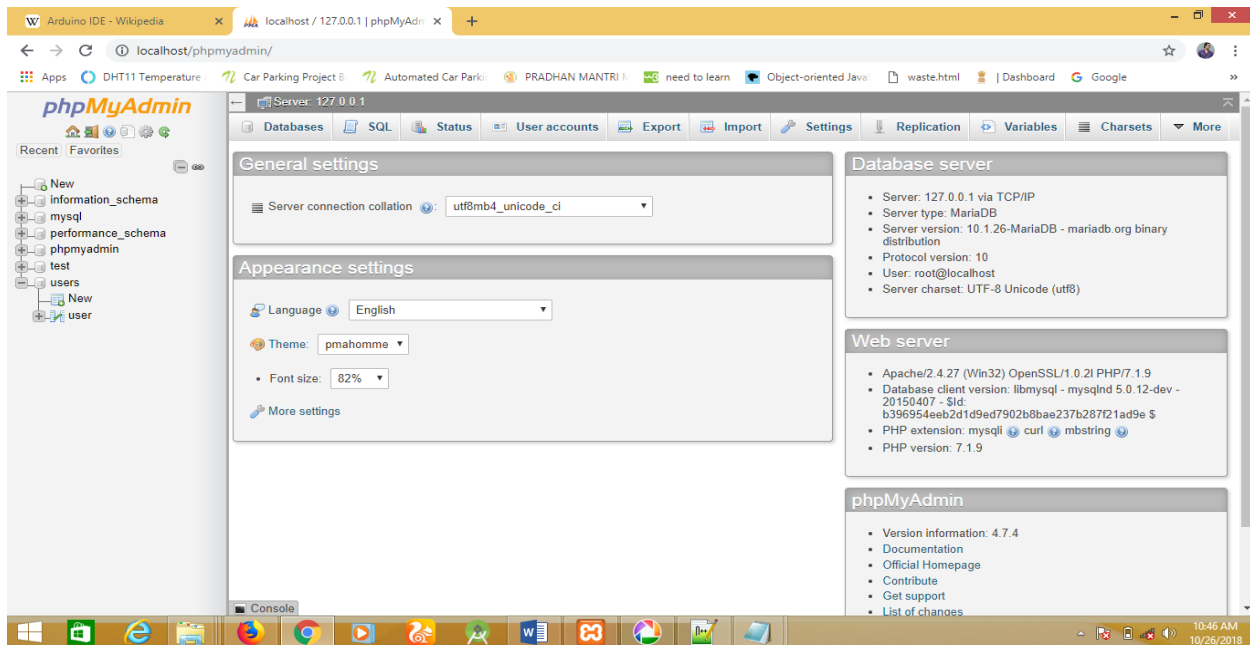


Fig 6.4: phpMyAdmin page

Step 3: Then click “New” and enter database name, here it is “users” and click Create button .Then a new data base will be created.

Step 4: Create a XAMPP user account i.e, Server name and password as follows. Enter the URL specified in step 1 in the browser’s address bar once again, then go to privileges click “Add user account”

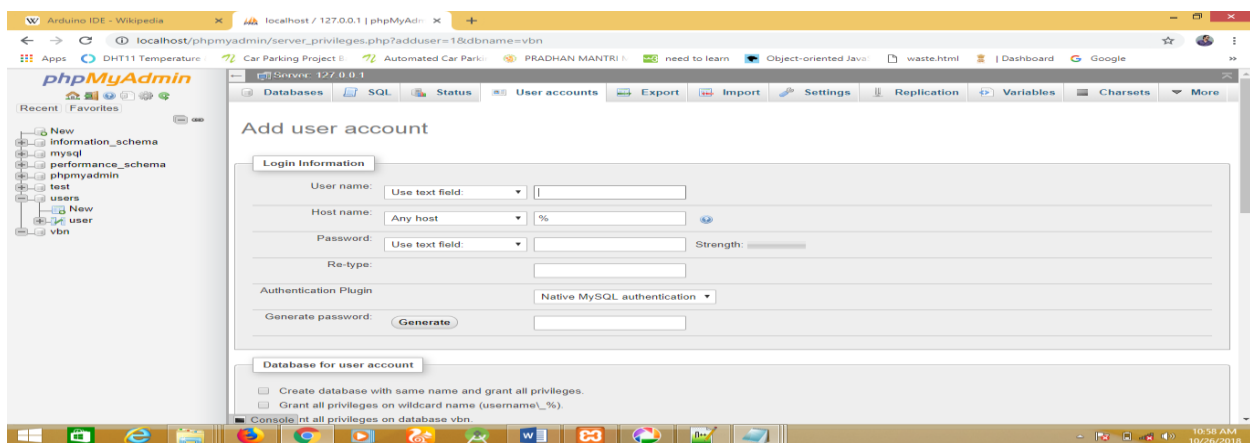


Fig 6.5: Server account page

Step 5: Enter Username, hostname, Password and check all checkboxes and then click “Go”. Then a XAMPP account is created.

Step 6: Then create a table in the database created using the following code

Refer to File- 4 for creating Table

Note: Care should be taken that servername, Username, Password, databasename entered in program matches with the ones we created previously.

Then a new table is created in the “users” data base with the name “users”.

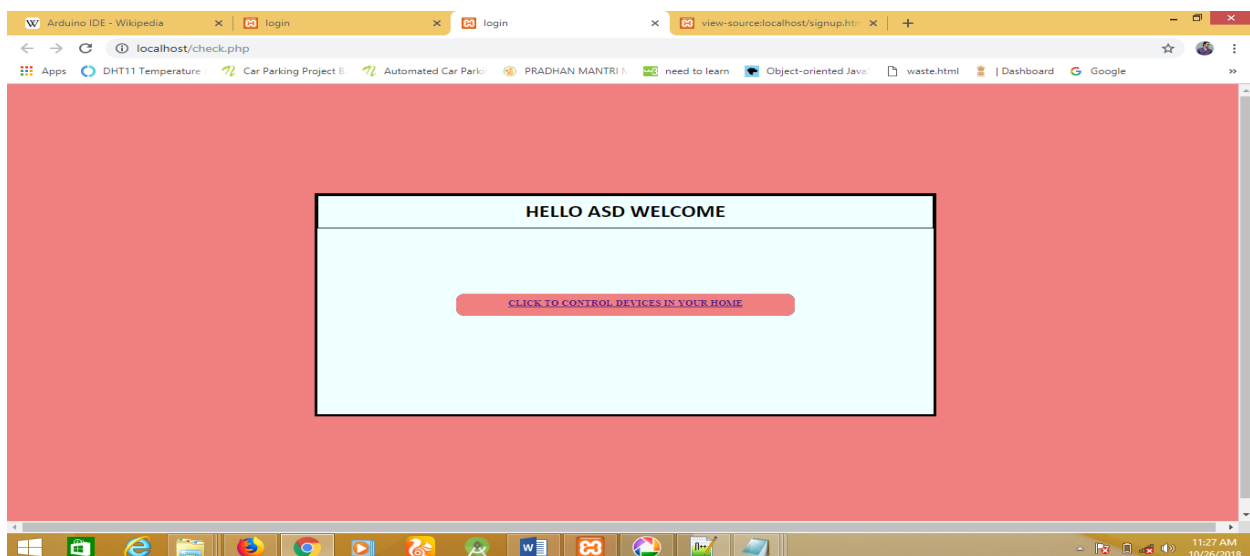
Step 7: When User enter details in Signup.html then that details will be stored in the table “users” we created already in the database “users”. For this a new file is required which is below:

Refer to File- 5 to create a file to upload details into the database:

Step 8 : When user enter details in login form to login ,then that details will be compared with already registered details which are stored in database. For this a new file is required which is given below:

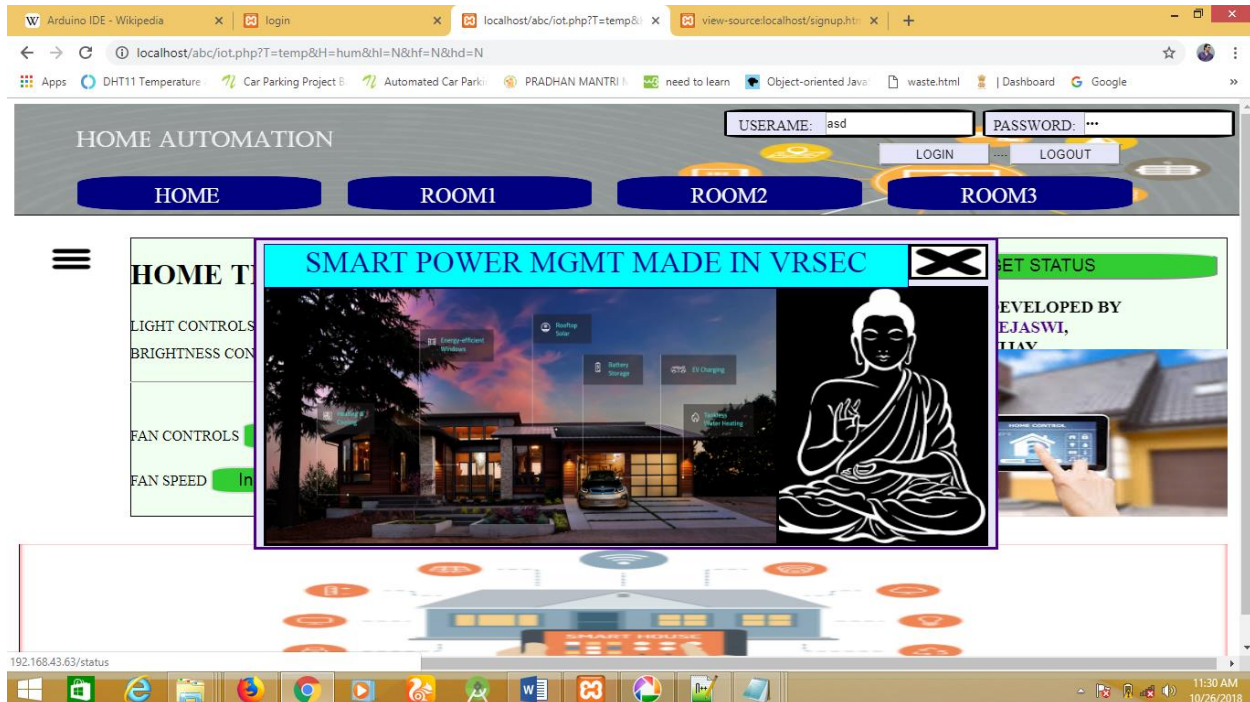
Refer to File- 6: Code for comparing user entered details with data stored in the database.

If all details are correct then redirecting page will be displayed as follows:



**Fig 6.5:** Redirecting page

Then click button to display the System control panel



**Fig 6.6:** System control panel

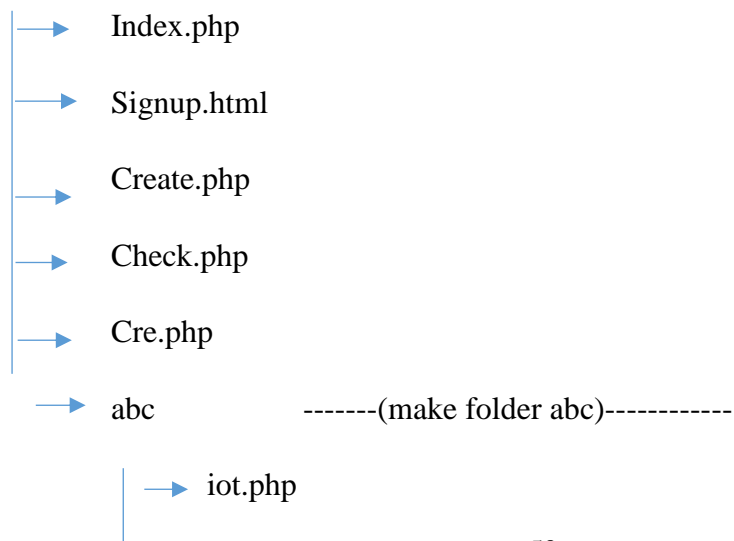
Then click the GUI buttons to control the devices

Refer to File-7: code for main application is given below:

File directory structure:

Go to htdocs folder in XAMPP folder in program files store files as shown below

Htdocs:





Note: There are many files to be stored in abc folder like images and other files they are provided in CD just copy the htdocs folder present in cd in XAMPP folder.

## **6.2 OVERALL WORKING**

1. In order to occur communication between NodeMCU and XAMPP's Apache server a local network is created using router or hotspot.
2. After creation of a local network, NodeMCU and PC in which XAMPP server is installed (in which we copy the htdocs folder in XAMPP folder) are made to connect to that local network.
3. When NodeMCU and PC connected to a network both will generate ip addresses and both knows others IP address as already entered in the program.
4. Start the server and MYSQL in XAMPP control panel
5. Connect the device (from which we want to control devices) to the local network created and type IP address of computer in the address bar of the device's browser
6. Enter details if already signed up and click login else signup and then login, if all details are correct system control panel will be displayed in the device.

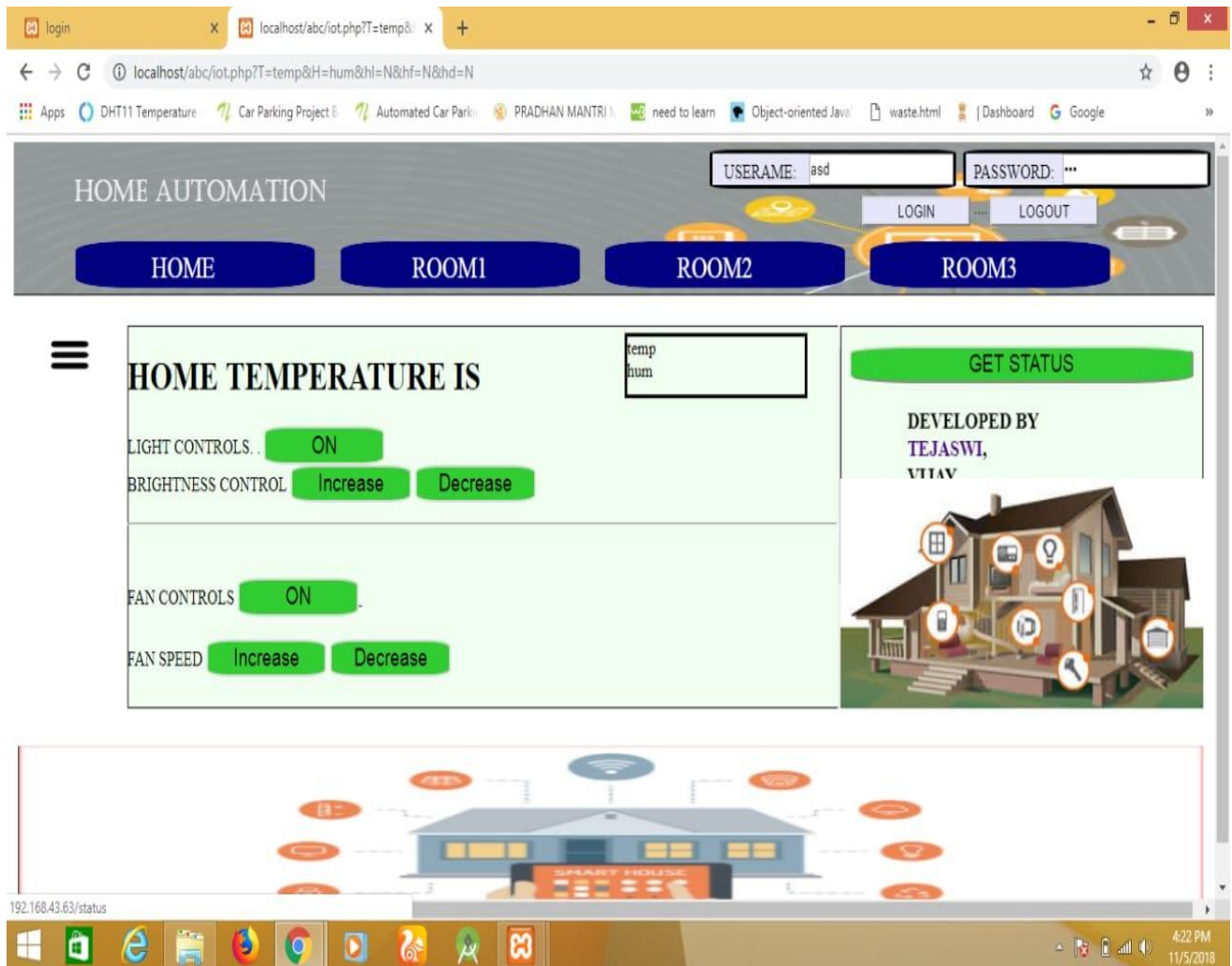
Note: Sometimes IP-addresses in the program written and IP- addresses actually generated by PC and NODEMCU may be different so enter actually generated IP- addresses in the program.

## CHAPTER-7

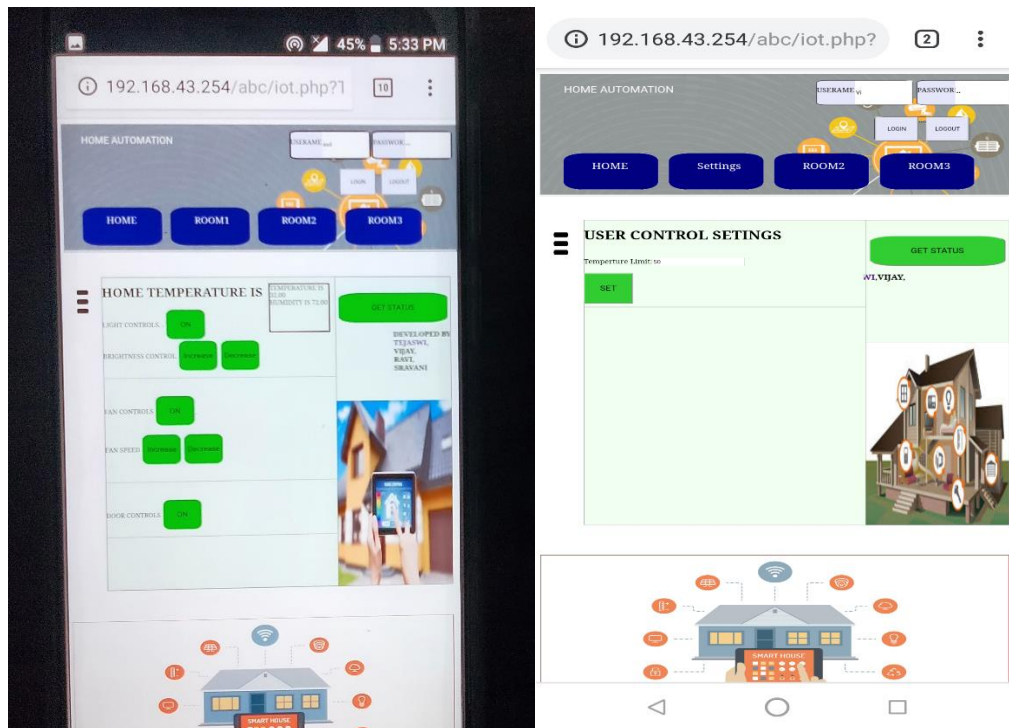
### RESULTS AND CONCLUSION

#### 7.1 RESULTS AND DISCUSSION

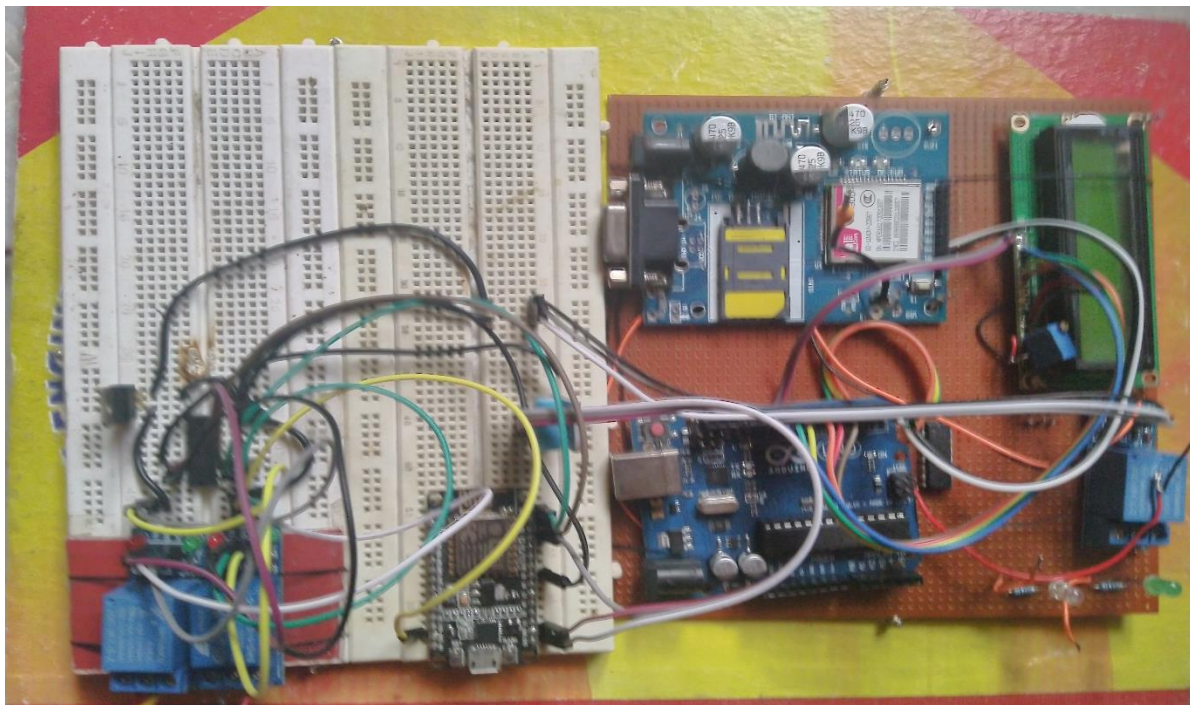
After the successful connection to the server, the data of sensor are sent to the web server for monitoring of the system. The Fig: 7.1 and Fig: 7.2 shows the web server page (when it is opened on PC or laptop and on a mobile phone) which will allow user to monitor and control the system. By entering the assigned IP- address in the web browser this web server page will appear. The web server gives the information about the temperature in different places of the house and motion state in the house. It also gives the status of the various electrical appliances like light, fan etc. which can be controlled remotely. This system also aims to make device that responds to various physical parameters like temperature, light etc .



**Fig 7.1:** Web server page (As viewed on PC)

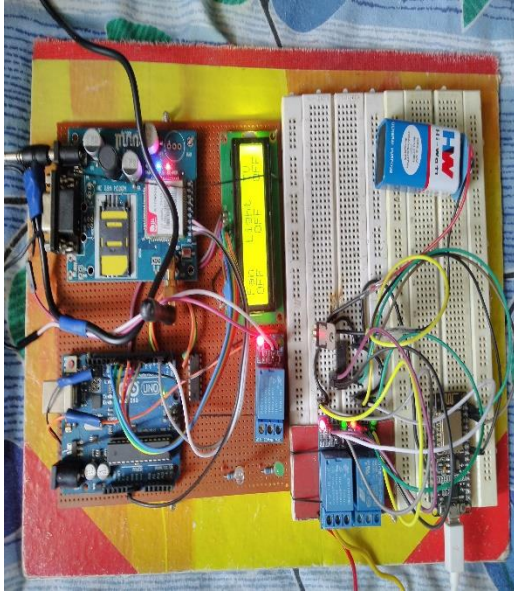


**Fig 7.2:** Web server page (As viewed on mobile) and temperature management control.

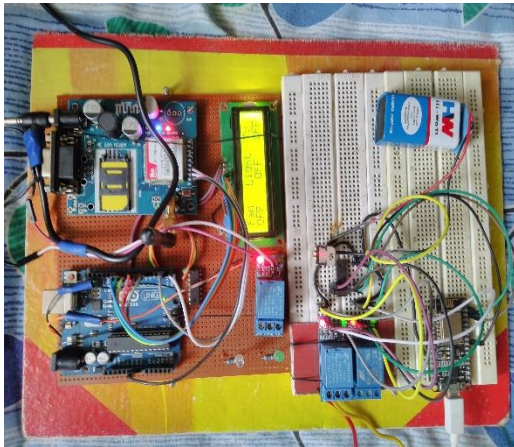


**Fig 7.3:** Implemented system





**Fig 7.4:** Controlling Fan



**Fig 7.5:** Controlling Light

Fig: 7.3 shows the initial state of the system. After the user clicks on the button on the graphical user panel i.e. on web page, the corresponding action takes place. If the user clicks on

the Fan control button then output as shown in Fig: 7.4 is observed. If the user clicks on the Light control button then output as shown in Fig: 7.5 is observed.

## **7.2 LIMITATIONS**

- Server may down sometimes in such cases service may interrupted.
- Sometimes requests may be failed in such cases appropriate task will not be performed.
- Only limited number of appliances can be connected to this system.
- It does not implement the automatic detection of faults.

## **Chapter - 8**

### **8.1 CONCLUSION**

The home automation using Internet of Things has been experimentally proven to work satisfactorily by connecting simple appliances to it and the appliances were successfully controlled remotely through internet. The designed system not only monitors the sensor data it can be easily integrated with different sensors like temperature, gas, light, motion sensors and actuates a process according to the requirement, for example switching on the light when it gets dark. It also stores the sensor parameters in the database in a timely manner. This will help the user to analyze the condition of various parameters in the home anytime anywhere.

It is evident from this project work that individual home automation system can be cheaply designed from the low cost components and can control a wide range of appliances (from simple fan to big appliances). This helps improve the living standards and also provide a safe environment where one can rest without any worries of break-ins or any kind of mishaps.

## Chapter - 9

### 9.1 FUTURE WORK

The system can be expanded to include various other options which could make home intelligent. They are:

- ✓ Energy and power can be managed more efficiently.
- ✓ Identification faults in the appliances.
- ✓ Modifying user interface which can provide a better look and feel and makes user task easier.
- ✓ Real the webpage in realtime servers and can provide more security to users.
- ✓ Other sensors like IR, Gas sensors will make the system better.
- ✓ A database which can store all the sensor data and which stores the data with respect to time, can be helpful to analyse the data and to take decisions on its own.
- ✓ Backup servers can be maintained to overcome server failures.
- ✓ Email and messaging services can integrated with the system for better error handling and also to send notifications to the user.

## REFERENCES

- [1] Weimei Zhang, "Study about IOT's application in Digital Agriculture construction", Electrical and Control Engineering (ICECE), 2011 International Conference, Yichang, IEEE, 2011, pp. 2578-2581.
- [2] Sushant Kumar, S.S Solanki, "Voice and Touch Control Home Automation", in 3rd International Conference on Recent Advances in Information Technology (RAIT), 2016
- [3] Ying-Tsung Lee, Wei-Hsuan Hsiao, Chin-Meng Huang and Seng-Cho T. Chou, "An Integrated Cloud-Based Smart Home Management System with Community Hierarchy", IEEE Transactions on Consumer Electronics, Vol. 62, No. 1, February 2016
- [4] M. Pawlenka, "Use of PLC and PIC for Realization of Intelligent Family House": Diploma Thesis. Ostrava: VSB - Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation, 2014, Thesis head: Skuta, J.
- [5] M. N. N. A. Asghar, M.H., "Principle application and vision in Internet of Things (IoT)," in Communication Technologies (GCCT), 2015 Global Conference on, may 2015.
- [6] M. Kuzlu, M. Pipattanasomporn and S. Rahman, "Review of communication technologies for smart homes/building applications," Smart Grid Technologies - Asia (ISGT ASIA), 2015 IEEE Innovative, Bangkok, 2015, pp. 1-6.
- [7] NodeMCU documentation-<https://nodemcu.readthedocs.io/en/master/>,  
<https://github.com/nodemcu/nodemcu-firmware>.
- [8] <https://github.com/tardate/LittleArduinoProjects/tree/master/ESP8266/Sensors/DHT11>.
- [9] <https://www.arduino.cc/>
- [10] <https://www.electronicwings.com/>



## APPENDIX:

### File-1:

#### Code for NodeMCU:

```
#include <ESP8266WiFi.h>

#include "DHT.h"      // including the library of DHT11 temperature and humidity sensor
#define DHTTYPE DHT11 // DHT 11
#define dht_dpin 0
DHT dht(dht_dpin, DHTTYPE);
void func(void);

const char* ssid = "vijay";
const char* password = "123456789";
int c=0;

int ledhl = D0;
int ledhf = D1;// GPIO13
int ledhd = D5;

String
hl="N",hf="N",hd="N",r1l="N",r1f="N",r1d="N",r2l="N",r2f="N",r2d="N",r3l="N",r3f="N",
",
r3d="N",r4l="N",r4f="N",r4d="N",r5l="N",r5f="N",r5d="N";

String temp;String hum;String k;
WiFiServer server(80);

void setup() {
  Serial.begin(115200);
  delay(10);
  dht.begin();
  pinMode(ledhl, OUTPUT);
  digitalWrite(ledhl, LOW);
```

```

    pinMode(ledhf, OUTPUT);
    digitalWrite(ledhf, LOW);
pinMode(ledhd, OUTPUT);
    digitalWrite(ledhd, LOW);
    // Connect to WiFi network
    Serial.println();
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    // Start the server
    server.begin();
    Serial.println("Server started");
    // Print the IP address
    Serial.print("Use this URL to connect: ");
    Serial.print("http://");
    Serial.print(WiFi.localIP());
    Serial.println("/");
}

void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    Serial.print("Current humidity = ");
    Serial.print(h);

```

```

Serial.print("% ");
Serial.print("temperature = ");
Serial.print(t);
Serial.println("C ");
delay(800);
// Check if a client has connected
WiFiClient client = server.available();
if (!client) {
    return;
}
// Wait until the client sends some data
Serial.println("new client");
while(!client.available()){
    delay(1);
}
// Read the first line of the request
String request = client.readStringUntil('\r');
Serial.println(request);
client.flush();
// Match the request
temp=String(t);hum=String(h);
//-----
if (request.indexOf("hl=ON") != -1) {
    digitalWrite(ledhl, !digitalRead(ledhl));
    if(!digitalRead(ledhl)) {
        hl="N";
    }else{hl="F";}
    func();
    client.println("<html><head><script>function
func(){ window.open("+k+");}</script></head><body onload='func()'></body></html>");

```

```

    }
    if (request.indexOf("hf=ON") != -1) {
        digitalWrite(ledhf, !digitalRead(ledhf));
        if(!digitalRead(ledhf)) {
            hf="N";
        }else{hf="F";}
        func();
        client.println("<html><head><script>function
func(){ window.open("+k+");}</script></head><body onload='func()'></body></html>");
    }
    if (request.indexOf("hd=ON") != -1) {
        digitalWrite(ledhd, !digitalRead(ledhd));
        if(!digitalRead(ledhd)) {
            hd="N";
        }else{hd="F";}
        func();
        client.println("<html><head><script>function
func(){ window.open("+k+");}</script></head><body onload='func()'></body></html>");
    }
    //-----
    if (request.indexOf("status") != -1) {
        func();
        client.println("<html><head><script>function
func(){ window.open("+k+");}</script></head><body onload='func()'></body></html>");
    }
    //-----
    Serial.println("Client disconnected");
    Serial.println("");
}
void func(){

```

```

k="\http://192.168.43.254/abc/iot.php?T=TEMPERATURE IS
"+temp+"&H=HUMIDITY IS
"+hum+"&hl="+hl+"&hf="+hf+"&hd="+hd+"\n";Serial.println(k);
}

```

## File- 2

-----index.php-----

```

<html>
<head>
<title>login</title>
<meta name="viewport"content="width=device,initial-scale=1">
<style type="text/css">
body{ width:100%;height:100%;background-color:lightcoral}
div.a { box-sizing:border-box }
div.main      { position:absolute;top:25%;left:25%;width:50%;height:50%;border-
style:solid;background-color:oldlace}
div form { position:absolute;top:45%;left:25% }
div.b{ position:absolute;top:25%;left:45%;width:10%;height:10%;border-style:solid;background-
color:lightblue;text-align:center;box-sizing:border-box;border-width:.1px;padding:5px;border-
radius:5px}
form input { border-style:solid;border-radius:10px;}
@media(max-width:650px){
div.main{ position:absolute;top:15%;left:10%;width:79%;height:70%;border-style:solid;border-
radius:10px}
div.a{ font-size:1em !important;padding:4% !important;color:indigo}
div form { position:absolute;top:35%;left:10% }
div form label { color:Green}
div.b { position:absolute;top:25%;left:35%;width:20%;height:5%;border-style:solid;background-
color:lightcoral;text-align:center;box-sizing:border-box;border-width:.1px;padding:3px;border-
radius:5px}
}
</style>

```

```

<script type="text/javascript">
function func()
{
window.open("signup.html","_self");
}
function fun()
{
sessionStorage.username=document.getElementById("us").value;
sessionStorage.pwd=document.getElementById("pw").value;
//document.write(sessionStorage.username+sessionStorage.pwd);,"_blank"
}
</script>
</head>
<body>
<div class="main">
<div class="a" style="position:absolute;width:100%;height:15%;font-weight:bold;font-size:1.5em;font-family:calibri,algerian;border-style:solid;border-width:.1px;text-align:center;padding:1%">VR SIDDHARTHA ENGG COLLEGE</div>
<div class="b">LOGIN</div>
<form id="signup" target="_blank" action="check.php" method="post">
<label>Username :</label><br/><input type="text" id="us" name="username"/><br/><br/>
<label>Password :</label><br/><input type="password" id="pw" name="pwd"/><br/><br/>
<input class="side" type="submit" value="login" onclick="fun()"/>
<input class="side" type="button" value="signup" onclick="func()"/>
</form>
</div>
</body>
</html>

```

### File- 3:

----signup.html-----

```
<html>
<head>
<title>signup</title>
<meta name="viewport"content="width=device,initial-scale=1">
<style type="text/css">
body{ width:100%;height:100%;background-color:lightcoral}
div.a {box-sizing:border-
box }div.b{position:absolute;top:25%;left:45%;width:10%;height:10%;border-
style:solid;background-color:lightblue;text-align:center;box-sizing:border-box;border-
width:.1px;padding:5px;border-radius:5px }
div.main      {position:absolute;top:25%;left:25%;width:50%;height:50%;border-
style:solid;background-color:oldlace}
div form {position:absolute;top:45%;left:25% }
form input {border-style:solid;border-radius:10px;}
@media(max-width:650px){
    div.main{position:absolute;top:15%;left:10%;width:79%;height:70%;border-
style:solid;border-radius:10px}
    div.a{font-size:1em !important;padding:4% !important;color:indigo}
    div form {position:absolute;top:35%;left:10% }
    div form label {color:Green}
    div.b {position:absolute;top:25%;left:35%;width:25%;height:5%;border-
style:solid;background-color:lightcoral;text-align:center;box-sizing:border-box;border-
width:.1px;padding:3px;border-radius:5px} } }
</style>
<script type="text/javascript">
</script>
</head>
<body>
<div class="main">
<div class="a"style="position:absolute;width:100%;height:15%;font-weight:bold;font-
size:1.5em;font-family:calibri,algerian;border-style:solid;border-width:.1px;text-
align:center;padding:1%">VR SIDDHARTHA ENGG COLLEGE</div>
```

```

<div class="b">SIGNUP</div>

<form id="signup" action="create.php" method="post"><label>Username :</label><br/><input
type="text" id="us" name="username"/><br/><br/>
<label>Password :</label><br/><input type="password" id="pw" name="pwd"/><br/><br/>
<input class="side" type="submit" value="signup"/>
</form>
</div>
</body>
</html>

```

#### File- 4:

-----cre.php-----

```

<?php $servername = "localhost";
$username = "vijay";
$password = "php@123";
$dbname = "users";

$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) { die("Connection failed: " . $conn->connect_error); }
// sql to create table
$sql = "CREATE TABLE user (username VARCHAR(30) PRIMARY KEY NOT NULL,
password VARCHAR(30) NOT NULL)";
if ($conn->query($sql) === TRUE) { echo "Table MyGuests created successfully"; }
else { echo "Error creating table: " . $conn->error; }
$conn->close();
?>

```

#### File – 5:

-----create.php-----

```

<?php
$name=$_POST["username"];
$pass=$_POST["pwd"];
$host="localhost";

```



```

$user="vijay";
$password="php@123";
$db="users";
$conn= new mysqli($host,$user,$password,$db);
if($conn->connect_error)
{
die("TRY AGAIN SERVER PROBLEM <br/>".$conn->connect_error);
}
if((!$name=="")&&(!$pass==""))
{
$sql="INSERT INTO user(username,password) VALUES('$name','$pass')";
if($conn->query($sql)===TRUE)
{
echo "Registered Successfully";
Include "index.php";
}
else
echo "Not regitered ".$conn->error;
}
else
echo "FIELDS cant be empty";
$conn->close();
?>

```

## File- 6:

-----check.php-----

```

<html>
<head>
<title>login</title>
<meta name="viewport"content="width=device,initial-scale=1">

```

```

<style type="text/css">
body{ width:100%;height:100%;background-color:lightcoral}

div.a { box-sizing:border-box;position:absolute;width:100%;height:15%;font-weight:bold;font-
      size:1.5em;font-family:calibri,algerian;border-style:solid;border-width:.1px;text-
      align:center;padding:1% }

div.main      { position:absolute;top:25%;left:25%;width:50%;height:50%;border-
      style:solid;background-color:azure}

div.c { position:absolute;top:45%;left:22.5%;font-weight:bold;text-
      align:center;width:55%;height:10%;border-style:solid;font-size:.7em;border-
      width:.1px;box-sizing:border-box;padding:7px;border-radius:10px;background-
      color:lightcoral}

form input {border-style:solid;border-radius:10px;}

@media(max-width:650px){
      div.main{ position:absolute;top:15%;left:10%;width:79%;height:70%;border-
      style:solid;border-radius:10px}

      div.a{ font-size:1em !important;padding:4% !important;color:indigo}

      div form { position:absolute;top:35%;left:10% }

      div form label {color:Green}

}

</style>
</head>
<body>
<div class="main">
<?php
$name=$_POST["username"];
$pass=$_POST["pwd"];
$host="localhost";
$username="vijay";
$passw="php@123";
$db="users";
$xyz=strtoupper($name);
$conn= new mysqli($host,$username,$passw,$db);

```

```

if($conn->connect_error)
{
die("TRY AGAIN SERVER PROBLEM <br/>".$conn->connect_error);
}if((!$name=="")&&(!$pass==""))
{
$query1 ="Select * FROM user WHERE username='$name' and password='$pass'";
$result=$conn->query($query1);
$n=$result->num_rows;
if($n==1)
{
echo "<div class=\"a\">HELLO ".$xyz." WELCOME</div>";
echo "<br/><br/><div class=\"c\"><a
        href='/abc/iot.php?T=temp&H=hum&hl=N&hf=N&hd=N'>CLICK TO
        CONTROL DEVICES IN YOUR HOME</a></div>";
}
else
echo "Invalid user name and password";
}
else
echo "ENTER NAME OR PASSWORD";
$conn->close();?>
</div></body></html>

```

## File- 7:

File-5 -----iot.php-----

```

<html>
<head>
<style type="text/css">
img.micon {position:absolute;top:25%;left:2% }
div.foot {position:absolute;top:80%;left:1%;border-
        style:solid;height:25%;width:98%;background-color:aquamarine;border-
        width:.01em}

```

```

div.butt {position:absolute;top:65%;left:2%;height:35%;width:90%;}
div.more{position:absolute;top:25%;left:2%;height:7%;width:10%;}
div.roomname {display:inline-block;position:relative;height:85%;width:22%;border-
width:.01em;border-style:solid;border-radius:20%;margin-left:2%;text-
align:center;background-color:navy;color:white}
div.heading {position:absolute;border-width:.01em;top:1%;height:20%;width:100%;text-
align:center;border-style:solid;z-index:1}
div.content {border-width:.01em;position:absolute;top:25%;left:10%;border-
style:solid;height:50%;width:60%;display:none;overflow-y:scroll;background-
color:honeydew}
div.temp {position:absolute;top:2%;left:70%;border-style:solid;height:15%;width:25%}
button {height:10%;width:17%;font-size:1.2em;background-color:limegreen;}
div.test {width:30%;height:50%;top:20%;left:30%;position:relative;font-size:1.5em}
div.mark{position:absolute;top:25%;right:1%;height:50%;width:30%;border-style:solid;border-
width:.01em;background-color:honeydew;}
div.slide{position:absolute;border-width:.01em;height:60%;width:100%;top:40%;border-
style:solid}
div.mark marquee {position:absolute;top:15%;right:1%;text-indent:justify;}
button.led{border-style:solid;border-radius:20%}
button.ledh{border-style:solid;border-radius:20%}
div.auth{border-style:solid;height:20%;width:20%;background-color:LAVENDER}
div label{position:absolute;left:0%;width:40%;top:25%}
input.ll1 {height:100%;width:45%;position:absolute;left:1%;background-color:LAVENDER}
input.ll2{height:100%;width:45%;position:absolute;right:1%;background-color:LAVENDER}
</style>
<script src="jquery-3.2.1.js"></script>
<script type="text/javascript">
var i,z=0;var pl;
function func(n)
{
var r=document.getElementsByClassName("content");
var a=document.getElementById("home");

```

```

a.style.display="none";
var sd=document.getElementById("us").value=sessionStorage.username;
var pd=document.getElementById("pw").value=sessionStorage.pwd;
if((sd=="undefined")||(pd=="undefined"))
{
    var spp=document.getElementsByTagName("body");
    spp[0].style.display="none";
}
for(i=0;i<r.length;i++)
{
    r[i].style.display="none";
}
r[n].style.display="inline-block";
var xx=document.getElementsByClassName("infor");
var ll=document.getElementsByClassName("ledh");
for(ab=0;ab<xx.length;ab++){
    document.write(ab+ll[ab].innerHTML);
        if((xx[ab].innerHTML=="N")){
            ll[ab].innerHTML="ON";
        }
        if((xx[ab].innerHTML=="F")){
            ll[ab].innerHTML="OFF";
        }
    }
}
$(document).ready(function(){
    Var hug=document.getElementById("imp").innerHTML;
    //var mpmp=document.getElementById("xxs").innerHTML;
    if(hug.indexOf("TEMPERATURE")==-1)
    {

```

```

setTimeout(adver,1500);
function adver()    {

    $("div.advert").show(1500);

    //document.write("aWEDF");

    setTimeout(cont,4000);

    function cont(){
document.getElementById("p").innerHTML="SMART POWER MGMT MADE IN VRSEC";
document.getElementById("p").style.color="DarkBlue ";
document.getElementById("ppp").src="a.jpg";
    }}

    }

/*if((mpmp.indexOf("TEMPERATURE")!=-1)|| (mpmp.indexOf("temp")!=-1))
{}
else
{ var bbb=document.getElementsByTagName("body");
    bbb[0].style.display="none";
    }*/

$("div.more").click(function(){
    $("div.lmenu").show();
    $("img.micon").hide();

    $("div.content").animate({left:'30%'},10);

    $("div#home").animate({left:'30%'},10);
    $("div.mark").hide();

});
$("img.close").click(function(){

    $("div.lmenu").hide();

```

```

                                                                    $("img.micon").show();

                                                                    $("div.content").animate({left:'10%'},10);

                                                                    $("div#home").animate({left:'10%'},10);

                                                                    $("div.mark").show();

                                                                    });

$("div.clrs").click(function(){ $("div.advert").hide(1000);});
});

var myIndex = 0;

setTimeout(carousel, 2000);

function carousel() {
    var x = document.getElementsByClassName("mySlides");
    var q;
    for (q= 0; q < x.length; q++) {
        x[q].style.display = "none";
    }
    myIndex++;
    if (myIndex > x.length) { myIndex = 1 }
    x[myIndex-1].style.display = "block";
    setTimeout(carousel, 2000); // Change image every 2 seconds
    //document.write(x.length);
}

function log(){
    sessionStorage.username=document.getElementById("us").value;
    sessionStorage.pwd=document.getElementById("pw").value;
}

```

```

function log1(){
    //document.write("in log1");
    sessionStorage.username="";
    sessionStorage.pwd="";
    window.close();
}
</script>
</head>
<body onload="func(0)">
<div class="more">
</div>
<div class="foot"></div>
<div class="heading">
    
    <p style="position:absolute;top:1%;left:5%;font-size:1.5em;color:white;font-
family:Castellar,algerian,Andalus">HOME AUTOMATION</p><div
class="butt">
        <div class="roomname"onclick="func(0)"><div
class="test">HOME</div></div>
        <div class="roomname"onclick="func(1)"><div
class="test">ROOM1</div></div>
        <div class="roomname"onclick="func(2)"><div
class="test">ROOM2</div></div>
        <div class="roomname"onclick="func(3)"><div
class="test">ROOM3</div></div>
    </div><form method="post"action="check.php"target="_blank">
    <div class="auth"style="position:absolute;top:5%;right:22%;border-
radius:10%;"><label>USERNAME:</label><input
name="username"style="position:absolute;right:0%;width:60%;height:100%"typ
e="text"id="us"/></div><div
class="auth"style="position:absolute;top:5%;right:1%;border-
radius:10%;"><label>PASSWORD:</label><input
name="pwd"style="position:absolute;right:0%;width:60%;height:100%"type="pa
ssword"id="pw"/></div>

```



```

<div style="position:absolute;top:35%;height:20%;width:20%;border-
width:.01em;right:10%;border-style:solid;border-radius:10%;"><input
type="submit" value="LOGIN" class="l1"onclick="log()"/>....<input
type="reset" value="LOGOUT" class="l2"onclick="log1()"/></div></form>

</div>

<div class="content" >

<h1>HOME TEMPERATURE IS</h1>

<p>LIGHT CONTROLS.      . <a href="http://192.168.43.63/h1=ON"target="_self"><button
class="ledh" id="">ON</button></a>

      <br/>BRIGHTNESS CONTROL <button
id="" class="led">Increase</button>

      <button id="" class="led">Decrease</button></p><!-->

<hr/><br/>

<p>FAN CONTROLS  <a href="http://192.168.43.63/hf=ON"target="_self"><button
class="ledh" id="">ON</button> </a><!-->

<br/><br/>FAN SPEED <button id="" class="led">Increase</button> <button
id="" class="led">Decrease</button></p>

<br><hr/><br>

<p>DOOR CONTROLS  <a href="http://192.168.43.63/hf=ON"target="_self"><button
class="ledh" id="">ON</button> </a>

<hr/><br/>

<div class="temp" id="imp">

<?php
$temp=$_GET["T"];
$hum=$_GET["H"];
if(empty($temp)||empty($hum))
echo "<p>temp is<br>hum is</p><script>funv();function funv(){ var
bbb=document.getElementsByTagName('body');

/*bbb[0].style.display='none';*/}</script>";

else
echo "$temp <br/> <span id='sds'>$hum</span>";

?>

```

```

</div>

</div>

<div class="content" >

<h1>ROOM 1 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS  <a href="http://192.168.43.63/LEDr1=ON"target="_self"><button
    class="led"id="">ON</button> </a><a
    href="http://192.168.43.63/LEDr1=OFF"><button
    class="led"id="">OFF</button> </a> <br/><br/>BRIGHTNESS CONTROL
    <button class="led"id="">Increase</button> <button
    class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS  <br/><button class="led"id="">ON</button>  <button
    class="led"id="">OFF</button><br/><br/>COOLING CONTROL <button
    class="led"id="">Increase</button> <button
    class="led"id="">Decrease</button></p>

<br/><hr/><br/>

<p>MOTOR CONTROLS  <button class="led"id="">ON</button>  <button
    class="led"id="">OFF</button>  </p>

<hr/><br/>

<div class="temp"><?php
$temp=$_GET["T"];
$hum=$_GET["H"];
if(empty($temp)||empty($hum))
echo "<p>temp is<br>hum is</p>";
else
echo "$temp<br/> $hum";
?></div>

</div>

<div class="content">

<h1>ROOM 2 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS  <br/><button class="led"id="">ON</button>  <button
    class="led"id="">OFF</button>  <br/><br/>BRIGHTNESS CONTROL

```

```

        <button class="led"id="">Increase</button> <button
        class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS    <br/><button class="led"id="">ON</button>    <button
        class="led"id="">OFF</button><br/><br/>COOLING CONTROL <button
        class="led"id="">Increase</button> <button
        class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS    <button class="led"id="">ON</button>    <button
        class="led"id="">OFF</button>    </p>

<hr/><br/>

<div class="temp"></div>

</div>

<div class="content" >

<h1>ROOM 3 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS    <br/><button class="led"id="">ON</button>    <button
        class="led"id="">OFF</button>    <br/><br/>BRIGHTNESS CONTROL
        <button class="led"id="">Increase</button> <button
        class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS    <br/><button class="led"id="">ON</button>    <button
        class="led"id="">OFF</button><br/><br/>COOLING CONTROL <button
        class="led"id="">Increase</button> <button
        class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS    <button class="led"id="">ON</button>    <button
        class="led"id="">OFF</button>    </p>

<hr/><br/>

<div class="temp"></div>

</div>

<div class="content" >

<h1>ROOM 4 TEMPERATURE IS</h1>

```

<p>LIGHT CONTROLS   <br/> <button class="led"id="">ON</button>   <button  
class="led"id="">OFF</button>   <br/><br/>BRIGHTNESS CONTROL  
<button class="led"id="">Increase</button>   <button  
class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS   <br/><button class="led"id="">ON</button>   <button  
class="led"id="">OFF</button><br/><br/>COOLING CONTROL   <button  
class="led"id="">Increase</button>   <button  
class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS   <button class="led"id="">ON</button>   <button  
class="led"id="">OFF</button>   </p>

<hr/><br/>

<div class="temp"></div>

</div>

<div class="content" >

<h1>ROOM 5 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS   <br/> <button class="led"id="">ON</button>   <button  
class="led"id="">OFF</button>   <br/><br/>BRIGHTNESS CONTROL  
<button class="led"id="">Increase</button>   <button  
class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS   <br/><button class="led"id="">ON</button>   <button  
class="led"id="">OFF</button><br/><br/>COOLING CONTROL   <button  
class="led"id="">Increase</button>   <button  
class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS   <br/><button class="led"id="">ON</button>   <button  
class="led"id="">OFF</button>   </p>

<hr/><br/>

<div class="temp"></div>

</div>

<div class="content" >

<h1>ROOM 6 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS <br/> <button class="led"id="">ON</button> <button  
class="led"id="">OFF</button> <br/><br/>BRIGHTNESS CONTROL  
<button class="led"id="">Increase</button> <button  
class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS <br/> <button class="led"id="">ON</button> <button  
class="led"id="">OFF</button><br/><br/>COOLING CONTROL <button  
class="led"id="">Increase</button> <button  
class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS <button class="led"id="">ON</button> <button  
class="led"id="">OFF</button> </p>

<hr/><br/>

<div class="temp"></div>

</div>

<div class="content" >

<h1>ROOM 7 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS <br/> <button class="led"id="">ON</button> <button  
class="led"id="">OFF</button> <br/><br/>BRIGHTNESS CONTROL  
<button class="led"id="">Increase</button> <button  
class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS <br/> <button class="led"id="">ON</button> <button  
class="led"id="">OFF</button><br/><br/>COOLING CONTROL <button  
class="led"id="">Increase</button> <button  
class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS <button class="led"id="">ON</button> <button  
class="led"id="">OFF</button> </p>

<hr/><br/>

<div class="temp"></div>

</div>

```

<div class="content" >

<h1>ROOM 8 TEMPERATURE IS</h1>

<p>LIGHT CONTROLS  <br/><button class="led"id="">ON</button>  <button
    class="led"id="">OFF</button>  <br/><br/>BRIGHTNESS CONTROL
    <button class="led"id="">Increase</button>  <button
    class="led"id="">Decrease</button></p>

<hr/><br/>

<p>AC CONTROLS  <br/><button class="led"id="">ON</button>  <button
    class="led"id="">OFF</button><br/><br/>COOLING CONTROL  <button
    class="led"id="">Increase</button>  <button
    class="led"id="">Decrease</button></p>

<br><hr/><br>

<p>MOTOR CONTROLS  <button class="led"id="">ON</button>  <button
    class="led"id="">OFF</button>  </p>

<hr/><br/>

<div class="temp"></div>

</div>

<div id="home"style="position:absolute;top:25%;left:10%;border-
    style:solid;height:50%;width:60%">home</div>

<div class="lmenu"style="position:absolute;background-color:grey;border-
    width:.01em;top:25%;left:2%;border-
    style:solid;height:50%;width:25%;display:none;overflow-y:scroll">

    <div class="close"style="float:right;height:15%;width:25%;">

    </div>

    <div style="border-
    width:.01em;position:absolute;top:15%;left:12%;height:60%;width:75%">

        <div class="roomname"onclick="func(4)"style="border-
        width:.5em;postion:relative;height:20%;width:100%;border-style:solid;margin-
        top:15%;dispal:y:block"><div class="test">ROOM4</div></div>

```

```

        <div class="roomname" onclick="func(5)" style="border-
width:.5em;postion:relative;height:20%;width:100%;border-style:solid;margin-
top:15%;dispaly:block"><div class="test">ROOM5</div></div>

        <div class="roomname" onclick="func(6)" style="border-
width:.5em;postion:relative;height:20%;width:100%;border-style:solid;margin-
top:15%;dispaly:block"><div class="test">ROOM5</div></div>

        <div class="roomname" onclick="func(7)" style="border-
width:.5em;postion:relative;height:20%;width:100%;border-style:solid;margin-
top:15%;dispaly:block"><div class="test">ROOM6</div></div>

        <div class="roomname" onclick="func(7)" style="border-
width:.5em;postion:relative;height:20%;width:100%;border-style:solid;margin-
top:15%;dispaly:block"><div class="test">ROOM7</div></div>

    </div>

</div>

<div class="mark"><marquee ><h3>DEVELOPED BY <span style="color:indigo;font-
size:bold"><BR/>TEJASWI</span>,<BR/>VIJAY,<br/>RAVI,<br/>SRAVANI
</h3></marquee>

<div class="slide">

</div>

<div class="instat" style="height:10%;width:95%;position:absolute;top:5%;left:2.5%"><a
href="http://192.168.43.63/status" target="_self"><button
style="height:100%;width:100%;border-radius:20%">GET
STATUS</button></a></div>

</div>

<div class="advert" style="position:absolute;border-color:indigo;background-
color:lavender;height:55%;width:60%;top:25%;left:20%;border-
style:solid;display:none">

    <div class="clrs" style="position:absolute;background-
color:white;height:10%;width:10%;top:1%;right:1%;border-style:solid">

</div>

```

```

<div style="position:absolute;background-
color:white;height:83%;width:30%;top:15%;right:1%;border-style:solid">

</div>

<div style="position:absolute;background-
color:white;height:83%;width:69%;top:15%;left:1%;border-style:solid">

</div>

<div id="p"style="position:absolute;background-color:aqua;height:14%;text-
align:center;font-size:25pt;width:87%;top:1%;left:1%;border-style:solid;border-
width:1px">
MINI PROJECT D6 BATCH</div>

```

```

</div>
<div id="inf"style="display:none">
<?php
$hl=$_GET["hl"];$hf=$_GET["hf"];$hd=$_GET["hd"];
echo"<p class='infor'>$hl</p><p class='infor'>$hf</p><p class='infor'>$hd</p>";
?>
</div>
</body></html>

```