

**BULB AND FAN CONTROL BY ARDUINO AND WIFI**

**MINOR PROJECT REPORT**

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR  
THE AWARD OF THE DEGREE OF

**BACHELOR OF TECHNOLOGY**

Information Technology



Submitted By:  
Jaspreet Kaur(1805518)  
Samreet Kaur(1805553)  
Harjot Singh(1805514)

Submitted To:  
Prof. Harjot Kaur  
Assistant Professor  
Minor Project Coordinator

**Department of Information Technology**

**Guru Nanak Dev Engineering College,**

**Ludhiana-141006**

## **Abstract**

Education plays a great role in development of any country. Many of education organizations is trying to increase education quality. We have come up with a new system called Arduino based Bulb and Fan automation. This system is super cost effective and can give the user, the ability to control these device without even spending for a remote control. This system is designed to support and provide support in order to fulfil the needs of elderly and impuissant in home. Also, the smart bulb and fan in the system developed the standard living at home. The main control system equipment's WIFI technology to provide remote access from Android Phone. The design remains the subsist electrical switches and provides more safety control on the switches with low voltage activating method. The switches status is synchronized in all the control system whereby every user interface indicates the real time existing switches status. The system intended to control fan and bulb with relatively low cost design, user-friendly interface and ease of installation.

## Acknowledgement

This has been the light of the day due to invaluable contribution of certain individuals whose constant guidance, support and encouragement resulted in the realization of our project within specific time limit.

We are grateful to our guide Prof. Hanit Karwal and HOD of Information Technology, Dr. Kulwinder Singh Maan for providing us the necessary help and guidance whenever we needed, which has resulted in the success of our project. We take this opportunity to thank Dr. Sehajpal Singh, Principal of our college in providing healthy environment in the college, which helped us in concentrating on our task.

We would like to thank all the staff members of our department, without whose constructive suggestions and valuable advice, the simple idea, which has borne by us, would not have been able to blossom forth to give such a beautiful bloom.

# List of Figures

1	SDLC model used . . . . .	12
2	Basic Topology . . . . .	13
3	Block Diagram . . . . .	15
4	Structured Chart . . . . .	17
5	Use Case Diagram . . . . .	18
6	Class Diagram . . . . .	19
7	Arduino IDE . . . . .	22
8	Gantt Chart . . . . .	24
9	Pert Chart . . . . .	24
10	NodeMCU ESP8266 . . . . .	27
11	Channel Relay . . . . .	28
12	USB Cable . . . . .	29
13	Connecting wires . . . . .	29
14	Blynk installation on smartphone . . . . .	29
15	Bulb used . . . . .	30
16	Fan used . . . . .	31
17	Configuring blynk app . . . . .	33
18	Creating new project . . . . .	34
19	Creating Switches . . . . .	35
20	Code . . . . .	36
21	Final Circuit . . . . .	36

# Contents

<b>List of Figures</b>	<b>3</b>
<b>1 Introduction</b>	<b>5</b>
1.1 Introduction to Project Page . . . . .	5
1.2 Project Category(Internet based, Application or System Development, Research based, Industry Automation, Network or System Administration) . . . . .	5
1.3 Objectives . . . . .	6
1.4 Problem Formulation . . . . .	6
1.5 Identification/Reorganization of Need . . . . .	7
1.6 Existing System . . . . .	7
1.7 Proposed System . . . . .	8
1.8 Unique Features of the System . . . . .	8
<b>2 Requirement Analysis and System Administration</b>	<b>10</b>
2.1 Feasibility Study (Technical, Economical, Operational) . . . . .	10
2.2 Software Requirement Specification Document which must include the following:(Data Requirement, Functional Requirement, Performance Requirement, Dependability Requirement, Maintenance Requirement, Security Requirement, Look and feel requirement) . . . . .	11
2.3 Validation . . . . .	11
2.4 Expected hurdles . . . . .	12
2.5 SDLC Model to be used . . . . .	12
<b>3 System Design</b>	<b>14</b>
3.1 Design Approach (Function oriented or Object oriented) . . . . .	14
3.2 Detail Design . . . . .	14
3.3 SystemDesign using various structured analysis and design tools such as : DFS's, Data Dictionary, Structured charts, Flowcharts or UML . . . . .	16
3.4 User Interface Design . . . . .	17
3.5 Methodology . . . . .	17

<b>4</b>	<b>Implementation, Testing and Maintenance</b>	<b>20</b>
4.1	Introduction to Languages, IDE's, Tools and Technologies used for Implementation . . . . .	20
4.2	Coding standards of Language used . . . . .	22
4.3	Project Scheduling using various tools such as PERT, GANTT charts, Open PROJ etc . . . . .	23
4.4	Testing Techniques and Test Plans . . . . .	23
<b>5</b>	<b>Results and Discussions</b>	<b>26</b>
5.1	User Interface Representation (of Respective Project) . . . . .	26
5.1.1	Brief Description of Various Modules of the system . . . . .	26
5.2	Snapshots of system with brief detail of each . . . . .	32
<b>6</b>	<b>Conclusion and Future Scope</b>	<b>37</b>
6.1	Conclusion . . . . .	37
6.2	Future Scope . . . . .	38
	<b>Bibliography</b>	<b>39</b>

# Chapter 1

## Introduction

### 1.1 Introduction to Project Page

Do you want to automate your home? Ever wished to control your appliances with a swipe on your phone? Have you ever wondered about how home automation can give you the facility of controller? Seeing an increasing demand for smart homes, where appliances react automatically to changing environment conditions and can be easily through one common device. We have come up with a new system called Arduino based Bulb and Fan automation. This system is super cost effective and can give the user, the ability to control these device without even spending for a remote control. This system allows automated Bulb and Fan for better convenience, energy efficiency, and security.

With the help of this system you can control your room bulb and fan from your mobile phone ,you can turn on/off them within the range of Wifi . With Wifi, we are using IOT(internetof things) concecpt, IOT is an umbrella term used for all technologies that enable the connection of a device to the Internet. Secondly we are using DCLD(digital circuits and logic design), the system uses DCLD technology that receives commands from phone and produces digital output which initiates relay driver to switch ON/OFF the bulb and fan using IC's,relays etc .Once the call is received by the Smart phone connected to the system, the user can now send commands to operate the light.

### 1.2 Project Category(Internet based, Application or System Development, Research bsd, Indutry Automation, Network or System Administration)

The Project Category for our project is Internet based , its bsd on IOT concept. In this age the gadgets that we're the utilization are getting to be more brilliant

and littler. They're connecting relatively without issue, and they might demonstrate to us that in almost every and everything of our regular daily existences. The Arduino based Bulb and Fan automation will be based Internet of Things (IoT). The bulb and fan are automated through the internet and are controlled. The user commands over the internet will be obtained by the Wi-Fi modems. The NodeMCU has an interface with this modem. This is a typical IoT based automation project, for controlling fan and bulb. The smart home market is taking off as IoT device prices come down and the general public comes to understand the benefits of these products and from smart fan and bulb, the next logical step is smart home, which would take the IoT to the next level. And yet, these automated devices are just one small part of our daily lives that the Internet of Things will transform in the coming years.

### 1.3 Objectives

1. The objective of this project is to implement a low cost, reliable and scalable automated Bulb and Fan that can be remotely switched on or off .
2. TO PROVIDE ANYTIME AND ANYWHERE ACCESS - Because you can access your smart Bulb and fan switch system from anywhere via your app using wifi, you can simply open your app and turn them off without reaching out it's switches.
3. HELP HANDICAPPED AND OLD -The objective of this project is to help handicapped and old aged people which will enable them to control mostly used appliances like Bulb and fan with ease.
4. ENERGY SAVING - To save the energy or power used in places like once or homes where lighting is very much important for the people and provide an automated life style.
5. TO SETUP EXCELLENT CONTROL OVER SWITCH - After the system is set-up, with the help of a mobile phone and a controller, tests are driven while data is recorded and inspected. Providing the ease to switch on/off light.

### 1.4 Problem Formulation

There is a huge energy exigency in the present situation of our country. Moreover, people become negligent in proper utilization of available energy. People often forget to turn off the light sources and home appliance while getting out from home. Even in those situation, application of home automation makes it possible to control them from a distant place in easy way with our smartphone. People are early and late running from place to place, working to accomplish everything on our never ending to-do list. Because of this project we will never have to take tension about switching off fan and bulb next time . In short we can save valuable time and experiences more daily productivity.



## 1.5 Identification/Reorganization of Need

1).Savings- Energy saving is one of the main goals in today's world. Therefore, all the technological innovations are directed towards the idea of conserving energy. One of the ways through which technology is helping in energy conservation is through home automation. According to [actuatorzone.com](http://actuatorzone.com) linear actuators enables the item to move in a straight direction while motion system create both linear and rotational motion. Therefore, they play a substantial role in home automation because they create motion and help in energy saving. Home automation is one of the essential technologies aimed at energy conservation as discussed below.

2).Control- Many of the todays apparatuses in a household, from broilers and fridge to deadbolts and cooling gadgets, might be controlled naturally by means of projects in PCs, phones and pills. In many occurrences, the control of every one of these gadgets works when you are out of the house as well and can transform them, which implies you could close the entry via the air terminal, check at the pooch from any of the nation, or affirm that you turned off your stove from the commercial center or some other store.

3).Convenience- Having the majority of your lounge and room lightings interchanged as you achieve your property remotely, the home theater and TV machine consequently betting your favored melody and the front entry opens naturally when you approach it with hands total of acquiring stuff, is maybe the end rich highlights of the astute and home. in any case, solace and harmony isn't about sumptuous and simple life, shrewd locks can likewise give you a chance to allow with the privilege of section of the particular people at exact examples and not generally, so you don't must remain at home as well as supply out a key. so also, a sensor lets you know while your fridge vacant or out of stock encourages you to "arrangement" your entrance or leave entryway from wherever inside this world.

## 1.6 Existing System

Most of the Smart home appliances that exist nowadays only have the provision of turning the machines ON and OFF. There are many systems available that aim to automate using NFC, Bluetooth and Wi-Fi. The existing system consists of a procedure to interact with the appliances, which usually includes pressing a button at some location within the application on your smart device. While some also have the provision to do so with the help of specific command. Many existing systems still uses Bluetooth module instead of the Wi-Fi module so the area of operation is greatly reduced. Many smart home appliances are available nowadays, which includes Amazon Echo, Google Home and Apple Home Kit, this is in order to bring home automation to every household.

## 1.7 Proposed System

The aim of the project is to give user both the option to either control various appliances through the app interface or through voice commands, which disregards the need to manually click within the application. The user first needs to authenticate by entering the credentials required, the user is then given the option to either send a voice command or click any of the radio buttons present on app interface, the command is interpreted by the mobile and an appropriate command is sent to the appliance. The application on your smart device acts as a central means through which the user communicates with the appliances. The application sends the signals to the NodeMCU which in turn send the appropriate command to the Relay through which the appliances are controlled, this demonstrates the concept i.e. IOT. The ESP8266 is programmed to send controls to relay which in turn control the appliances. With the help of this project you will be able to automate every appliance, which will greatly help in reducing power consumption. The user was also given the option to control the devices automatically using the input provided by the different types of sensor that are interfaced with the Arduino. The sensors provide the Arduino readings from the surrounding, the Arduino is in turn connected to the Relay which is in turn connected to the appliances, so the devices were able to be controlled automatically using the input from the sensors.

## 1.8 Unique Features of the System

1)MANAGING HOME APPLIANCES SUCH AS FAN AND BULB FROM ON PLACE- The convenience factor here is enormous. Being able to keep all of the technology in your home connected through one interface is a massive step forward for technology and home management. Theoretically, all you'll have to do is learn how to use one app on your smartphone and tablet, and you'll be able to tap into countless functions and devices throughout your home. This cuts way back on the learning curve for new users, makes it easier to access the functionality you truly want for your home.

2).REMOTE CONTROL OF THESE FUNCTIONS- Don't underestimate the power of being able to control your home's functions from a distance. On an exceptionally hot day, you can order your house to become cooler in just enough time before you get home from work. If you're in a hurry to get dinner started but you're still at the store, you can have your oven start to preheat while you're still on your way home. You can even check to see if you left the lights on, who is at your front door, or make sure you turned off all your media while you're away.

3).INCREASE ENERGY EFFICIENCY- Depending on how you use your smart-home technology, it's possible to make your space more energy-efficient. For example, you can have more precise control over the heating and cooling of your home with a programmable smart thermostat that learns your schedule and temperature preferences, and then suggests the best energy efficient settings throughout the day. Lights and motorized shades can be programmed to switch

to an evening mode as the sun sets, or lights can turn on and off automatically when you enter or leave the room, so you never have to worry about wasting energy.

4).IMPROVED BULB AND FAN FUNCTIONALITY- Smart homes can also help you run your appliances better. A smart TV will help you find better apps and channels to locate your favorite programming. A smart oven will assist you with cooking your chicken to perfection – without ever worrying about overcooking or undercooking it. An intelligently designed home theater and audio system can make managing your movie and music collection effortless when entertaining guests. Ultimately, connecting your appliances and other systems with automation technology will improve your appliance effectiveness and overall make your home life much more easier and enjoyable!

## Chapter 2

# Requirement Analysis and System Administration

### 2.1 Feasibility Study (Technical, Economical, Operational)

The project is executed within a one semester frame. The time frame allocated for this project is limited but adequate through proper time management and planning. The aim of this project is to design an open source, easy-to-use and affordable bulb and fan automation. The project can be accomplished with low-cost electronic and software technologies, making it feasible from an economic, technical, and operational standpoint.

1).ECONOMIC FEASIBILITY-This project is based on an Android phone with low-cost electronic components such as my Processor, camera modules, and relay switches which are the most cost-effective structures to construct.

2).TECHNICAL FEASIBILITY-This proposal is based on wireless radios that are incorporated in a way that is reasonably in phase with current technology. As a result, it is highly valued by technology.

3).OPERATIONAL DEASIBILITY-This programme will feature a simple, user-friendly interface that will appeal to anyone with a basic understanding of how to operate an Android phone. Controlling house appliances with a push of a button might be beneficial for physically challenged people as well. As a result, it is possible from a practical standpoint.

## **2.2 Software Requirement Specification Document which must include the following:(Data Requirement, Functional Requirement, Performance Requirement, Dependability Requirement, Maintenance Requirement, Security Requirement, Look and feel requirement)**

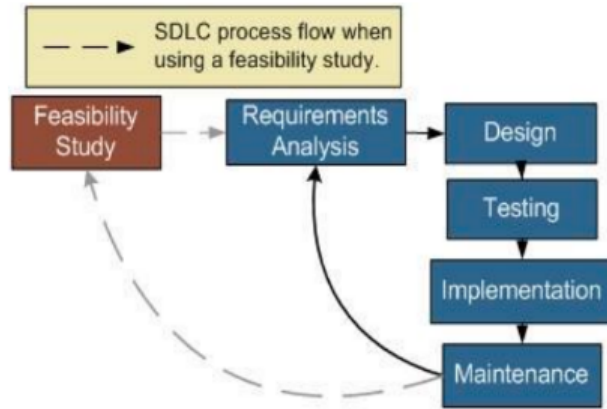
- 1).DATA REQUIREMENT-The data for the fan and bulb automation system must be stored and accessible in a secure and reliable manner. At any one moment, a database will record the usernames and passwords of several users. The database will also keep track of the quantity and status of any automated appliances that are available.
- 2).FUNCTIONAL REQUIREMENT- The functionalities that a developer must incorporate into software to accomplish use cases are referred to as functional requirements. These functions will be built in such a way that vendor independent system-to-system communication is possible.
- 3).PERFORMANCE REQUIREMENT-If the system is not connected, it must not add more than two seconds to the time it takes to accomplish an action. There must be no more than a ten-second delay in the logging of researcher data to the research centre. The speed with which directives are provided to the system will be affected by the efficiency of the software code.
- 4).MAINTAINABILITY REQUIREMENT-The system is as simple to use as feasible, with all capabilities accessible.
- 5).SECURITY REQUIREMENT-Because the system is meant to run on a network like the internet, there are security concerns connected with utilising it. When evaluating the system, the user must ensure that intruders, such as hacker attempts and third-party invasions, are prevented from gaining access.
- 6).LOOK AND FEEL REQUIREMENT-Modularity is incorporated into the system's architecture.

## **2.3 Validation**

To validate the performance of our system a hardware implementation is done.

- 1).MANUAL-SCENARIO IMPLEMENTATION-For the hardware implementation(manual scenario), we can remotely control it by using ON or OFF buttons in our smart phones for fan and bulb.
- 2).AUTOMATIC-SCENARIO IMPLEMENTATION-The arduino uno kit controls the automation of the fan and lamp based on continuous data from the sensors.

Figure 1: SDLC model used



## 2.4 Expected hurdles

There are several external and internal hurdles associated with our project.

### 1).EXTERNAL HURDLES

a).POWER CUTS-The risk of unreliable electrical supply in the household is one of the external obstacles. Even after such an occurrence, the system is capable of working and working properly

b).DAMAGE OF DEVICES-This will be a series condition, and there will be no means of knowing the device's state in this situation.

### 2).INTERNAL HURDLES

a).SOFTWARE-Our project's development will be restricted to Arduino programming languages.

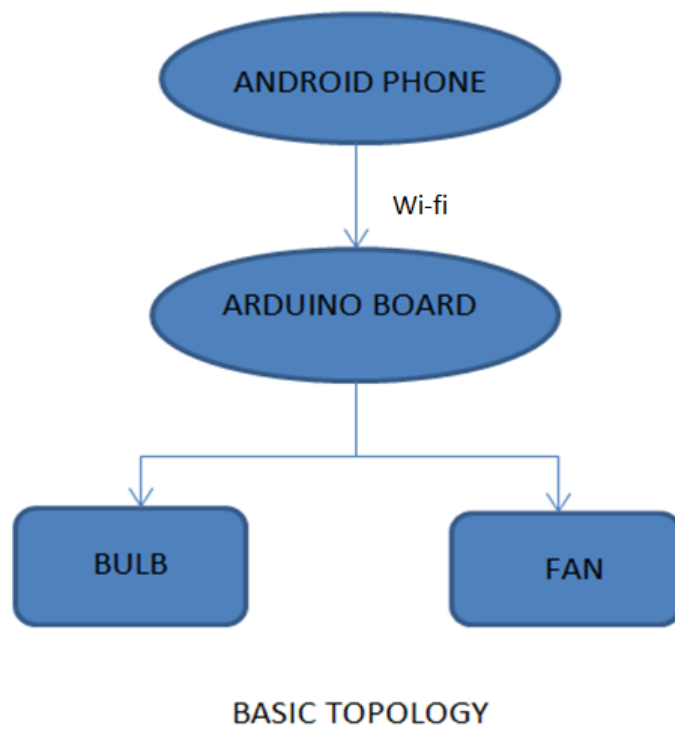
b).FUNCTIONALITY-Because of the problems of high electricity and components that cannot handle the high power, the functionality of our system will be limited to just functioning on leds that will operate as gadgets at home.

## 2.5 SDLC Model to be used

The following figure shows the simplest graphical representation of the SDLC which we have followed- The hardware involved in making the system work from hand to hand, from the Android phone to the Arduino board, and finally to the Home Appliances, is depicted in the diagram below.

Our project's core backbone is this architecture, with the remainder of the connections being made through Wi-Fi.

Figure 2: Basic Topology



## Chapter 3

# System Design

### 3.1 Design Approach (Function oriented or Object oriented)

The focus of the object-oriented approach is on encapsulating the structure and behaviour of information systems into tiny modules that mix data and process. The basic goal of Object Oriented Design (OOD) is to make system analysis and design more accessible in order to increase quality and productivity. In analysis phase, OO models are used to fill the gap between problem and solution. It performs well in situation where systems are undergoing continuous design, adaption, and maintenance. It identifies the objects in problem domain, classifying them in terms of data and behavior.

The Object Oriented model is beneficial in the following ways:

1. It facilitates changes in the system at low cost.
2. It promotes the reuse of components.
3. It simplifies the problem of integrating components to configure large system.
4. It simplifies the design of distributed systems.

### 3.2 Detail Design

DETAIL DESIGN-Several blocks were used to create the system, including an input/output block, a microcontroller block, a networking block, and a controlling/monitoring devices block.

BLOCK DIAGRAM:

#### 1. INPUT/OUTPUT BLOCK

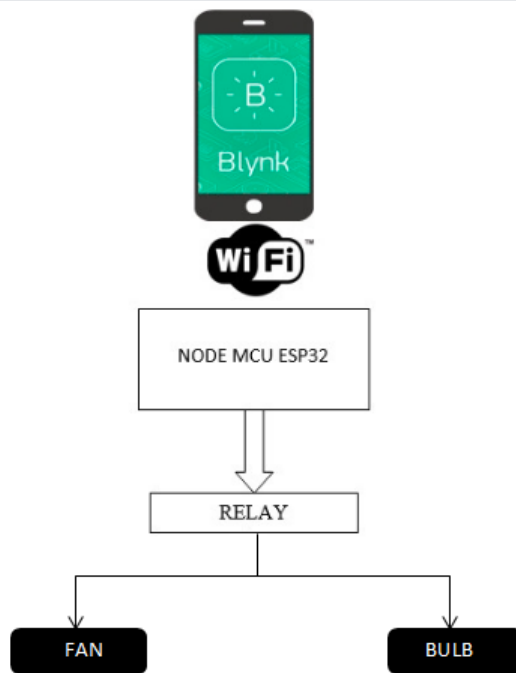
Input block consists of nodemcu esp8266 and blynk application on our smart phone.

Output block consists of relay.

The NodeMcu8266 is an open-source IoT platform with a low-cost develop-



Figure 3: Block Diagram



BLOCK DIAGRAM

ment board incorporated into the ESP8266 WiFi module. The NodeMcu is programmed using the ARDUINO IDE, which is comparable to the Arduino programming. However, the IDE does not come with an ESP-compatible board pre-installed. Before executing the application, we must first add the board attachments to the ARDUINO IDE.

#### FEATURES

1. It is open-source.
2. Simple to use, easily programmable and interactive
3. It is very smart with wi-fi enabled.

Blynk is a new platform that lets you easily create interfaces to manage and monitor your hardware projects from your iOS or Android mobile. You may construct a project dashboard after downloading the Blynk software and arranging buttons, sliders, graphs, and other widgets on the screen. You may use the widgets to switch pins on and off, as well as show data from sensors.

A relay is an electrical device that is commonly used to regulate large voltages with a very low input voltage.

#### 2. INTEGRATED DEVELOPMENT ENVIRONMENT

IDE is a cross-platform application that is written in functions from C and C++ language. The IDE used in our project is Arduino IDE written in C language.

Writing code and uploading it to the board is simple with the open-source Arduino Software (IDE). Any Arduino board may be used with this software.

The Arduino IDE employs the program avrdude 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. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

#### 3. MONITORING/CONTROLLING DEVICE

This block consists of any devices that helps in monitoring/controlling the system. These devices are smart phone and laptop and these are connected to wi-fi. Status of the equipment that monitored and controlled will be seen in the monitoring device. While the monitoring device able to control fan and bulb at any time.

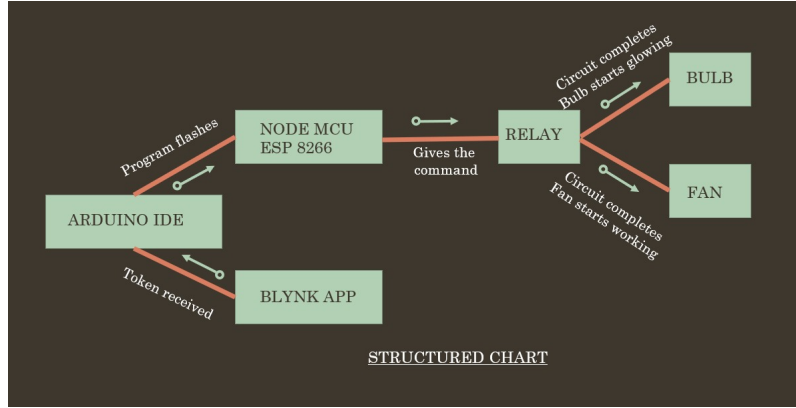
**3.2.4 SYSTEM DESIGNED** It can be seen that the system has one nodemcu esp8266, two relays and connecting wires. All equipment's placed in each planned place. A scenario has been designed to control fan and bulb. This scenario then was written in C-like Arduino language.

### **3.3 System Design using various structured analysis and design tools such as : DFS's, Data Dictionary, Structured charts, Flowcharts or UML**

STRUCTURED CHART:

USE CASE DIAGRAM:

Figure 4: Structured Chart



CLASS DIAGRAM :

### 3.4 User Interface Design

In addition to downloadable apps, most manufacturers of systems offer user interfaces in the form of dedicated touch screens. Due to its ability to be customized and its comfortable

### 3.5 Methodology

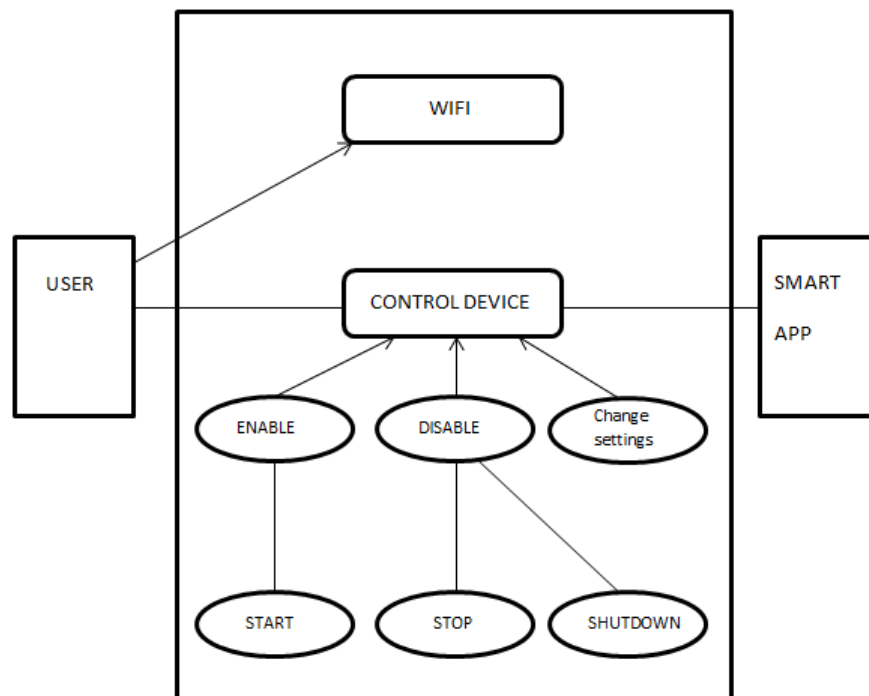
After confirmation of the project title, preliminary research is done to analyze it. Further research is then made to find the most suitable hardware and software in executing the project. The plan for the system's design is then laid out and materials are purchased. The Arduino programming language must be familiarized. The next step involves getting the system to work wirelessly utilizing the Wi-Fi shield and wireless control through web browser to control fan and bulb. After the hardware setup is finalized and once the system is found to be stable, with the help of android application - Blynk app, the system is controlled. 1.A coding is written in the C language, which is then given into the NODE MCU module.

2.The connections are made as per the connection diagram.

3.Source supply is provided to the bulb and fan.

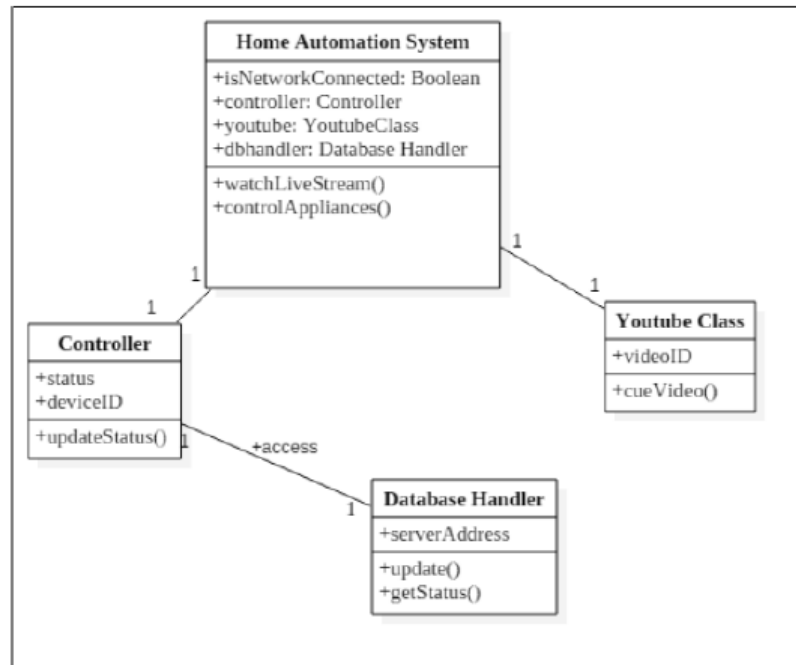
4.With the help of application in the android, the switching of bulb and fan can be controlled.

Figure 5: Use Case Diagram



USE CASE DIAGRAM

Figure 6: Class Diagram



## Chapter 4

# Implementation, Testing and Maintenance

### 4.1 Introduction to Languages, IDE's, Tools and Technologies used for Implementation

#### C PROGRAMMING LANGUAGE-

The C programming language is a procedural programming language. Dennis Ritchie created it as a system programming language for writing operating systems. Low-level memory access, a small collection of keywords, and a clean style are all qualities that make C language excellent for system programming, such as operating system or compiler development. Many following languages have directly or indirectly inherited syntax and features from C. The C language is the foundation for several languages, including Java, PHP, JavaScript, and many more. C++ is a near-superset of the C language (just a few programmes will compile in C, but not in C++). C is a widely used general-purpose programming language that is easy to learn and use. It is a machine-independent structured programming language that is widely used to create a variety of applications, operating systems such as Windows, and other complicated programmes such as the Oracle database, Git, Python interpreter, and others.

#### WHY LEARN C LANGUAGE-

C is a foundational language for numerous programming languages, as we previously discussed. As a result, understanding 'C' as the primary programming language will be useful while learning other programming languages. Data types, operators, control statements, and a variety of other notions are all included. 'C' is a versatile character that may be used in a variety of contexts. It's a straightforward language with quick execution. 'C' is a structured programming language that divides a programme into several modules. Each module may be written alone, but when combined, they constitute a single 'C' programme. This framework facilitates process testing, maintenance, and trou-

bleshooting. 'C' contains 32 keywords, various data types and a set of powerful built-in functions that make programming very efficient. Another property of 'C' programming is its ability to self-extend. Various functions from a library are included in a 'C' programme. We have the option of adding our own features and functions to the library. In our application, we may access and use these functions whenever we want. This capability makes working with sophisticated code much easier.

#### ARDUINO IDE-

To develop computer code and upload it to the physical board, the Arduino IDE (Integrated Development Environment) is utilised. The Arduino IDE is quite straightforward, which is perhaps one of the reasons Arduino has become so popular. We can confidently declare that Arduino IDE compatibility is currently one of the most important requirements for a new microcontroller board. The Arduino IDE is a free and open source programme for developing and compiling code for the Arduino Module. It is official Arduino software that makes code compilation so simple that even a non-technical person may get their feet wet with the learning process. Many valuable features have been introduced to the Arduino IDE over the years, and you can now manage third-party libraries and boards from the IDE while maintaining the board's simplicity. It operates on the Java Platform and is compatible with operating systems such as MAC, Windows, and Linux. It has built-in functions and commands that are useful for debugging, editing, and compiling code in the environment. Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro, and many other Arduino modules are available. On the board of each of them is a microcontroller that has been programmed and takes data in the form of code. The core code, also known as a sketch, written on the IDE platform will eventually generate a Hex File, which will be copied and uploaded to the board's controller. The IDE environment is made up of two parts: an editor and a compiler. The editor is used to write the needed code, while the compiler is used to compile and upload the code to the Arduino Module. Both C and C++ are supported in this environment. The main window of the Arduino IDE is shown below, with the simple simple Blynk example.

#### TOOLS AND TECHNOLOGIES USED FOR IMPLEMENTATION

##### 1. Android development tools (ADT)

To build the android application to send the control signal.

##### 2. Microsoft Visual Studio 2010

The server programme for Windows is written in Microsoft Visual C and built as a normal Windows executable with CLR (Common Language Runtime), which is part of the NET Framework.

##### 3. Arduino IDE For Microcontroller

A text editor for writing code, a message box, a text terminal, a toolbar with buttons for common operations, and a series of menus are all included in the Arduino Software (IDE).

##### 4. PIC PGM Programmer 1.0.1.4

The PIC PGM Programmer software is used to flash the compiled programme to the microcontroller's EEPROM and check that it has been written success-

Figure 7: Arduino IDE



fully.

## 5. WIFI

WIFI is an alternative network to wired network which is commonly used for connecting devices in wireless mode.

## 4.2 Coding standards of Language used

### CODING STANDARDS OF C LANGUAGE

To simplify the definition and enable some freedom in implementation, the C standard leaves some behaviour of many C constructs undefined and others undetermined. In C, for example, using any automatic variable before it has been initialised results in undefined behaviour, and the sequence in which subexpressions are evaluated is not stated. Should such a programme be provided, this expressly frees the compiler to perform whatever is easiest or most efficient.

So what is the conclusion about above two examples?

Consider the first case, "void main()," where the standard specifies the following regarding the prototype of main: ().

The main function is called when the programme starts. This function has no prototype declared in the implementation. It must be defined with an int return type and no arguments.

As a result, the return type void deviates from the standard, yet it is permitted by some compilers.

Let's have a look at the second case. The following sentence is mentioned under



undefined behaviour in the C standard.

In a function call, the order in which the function designator, arguments, and subexpressions inside the arguments are evaluated.

What to do with programs whose behaviour is undefined or unspecified in standard?

It is never a good idea to utilise programming structures whose behaviour is undefined or undetermined as a programmer; such programmes should be avoided at all costs. The compiler and/or hardware may affect the output of such programmes.

#### ANSI STANDARD C

EMBOSS C should follow this C89 standard. The semantics of this are given in the second edition of Kandr and also in Harbison and Steele's C: A Reference Manual.

This standard introduced:

- a) Types of arguments are specified in the function definitions.
  - b) Addition of void's and enum's
  - c) Ability to pass structures to functions and have structures as return values.
- 'a' and 'b' are welcome additions. In EMBOSS, passing whole structures around is deprecated (we have our own way of dealing with structures and, even if we hadn't, passing structures around is wasteful of stack memory and rather slow). If you need to write system dependent code then try to put such functions in (e.g.) ajsys.c to isolate it as much as possible.

### 4.3 Project Scheduling using various tools such as PERT, GANTT charts, Open PROJ etc

### 4.4 Testing Techniques and Test Plans

Each module was put through its paces utilising both black box and white box testing methods. Given a variety of possible inputs, black box testing assures that the system's output is valid. White box testing entails supplying computed inputs in order to ensure that the system's underlying structure is error-free. White box testing of a software module, for example, is supplying various inputs to evaluate all potential code paths within the programme. The modules were created with all conceivable inputs in mind, as well as any incorrect inputs. Any exceptions are dealt with correctly without jeopardising the system's stability. Various challenges, both anticipated and unanticipated, arose during the design process. This chapter discusses these difficulties as well as the solutions that were implemented to address them.

#### 1).SERVER MODULE

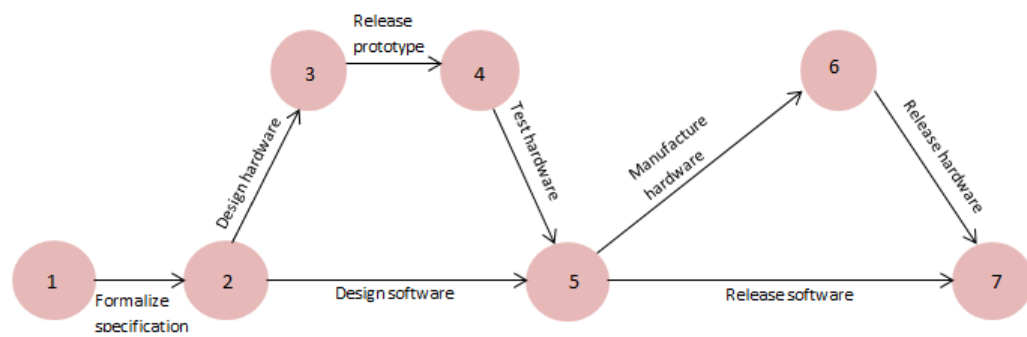
a).ACTIVATION OF TIME BASED PROFILE-By comparing the current time to the profile's activation time, the system determines if it is time to activate the profile. When the current time swings around to 00:00:00 for the following

Figure 8: Gantt Chart

sno	Task name	Duration	March2021				April2021				May2021				June2021			
			1W	2W	3W	4W	1W	2W	3W	4W	1W	2W	3W	4W	1W	2W	3W	4W
1	Project Management	19																
1.1	Initiation	14																
1.2	Planning	14																
2	Analysis	14																
3	Design	25																
4	Implementation and Support	21																
5	Testing	17																

**GANTT CHART**

Figure 9: Pert Chart



**PERT CHART**

day/month/year, however, the comparison might be erroneous. To accommodate for such circumstances, the calculation is changed.

b).PORT SELECTION-When the programme is launched again, it's conceivable that a COM port that was previously used for communication with the microcontroller and recorded in the configuration file no longer exists. This can happen if you plug the microcontroller into a different USB port than you did the last time, in which case Windows will assign it to a new virtual COM port. The COM ports are enumerated to avoid an error, and the last used port is enabled only if it is still present in the current enumeration.

## 2). MICROCONTROLLER AND HARDWARE MODULE

a).QUALITY OF MAIN SUPPLY-To work properly, electromagnetic relays require a consistent +12V DC supply. To produce a +12V DC supply from the mains, a transformer with a rectifier was first employed.

b). ELECTRICAL GROUNDING PROBLEM-During testing, it was discovered that the USB standard required the ground wire to link two end points perfectly. Only the two data connections were linked to the ground wire in this scenario, and the power connector's ground wire was not connected to the PC's ground. To absorb stray currents, the power plug was altered to a three-pin top and a direct connection to the ground pin of the wall outlet was supplied.

## Chapter 5

# Results and Discussions

### 5.1 User Interface Representation (of Respective Project)

#### 5.1.1 Brief Description of Various Modules of the system

Hardware Requirement

##### 1. NODE MCU ESP8266

The NodeMCU ESP8266 development board comes with the ESP-12E module containing ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects.

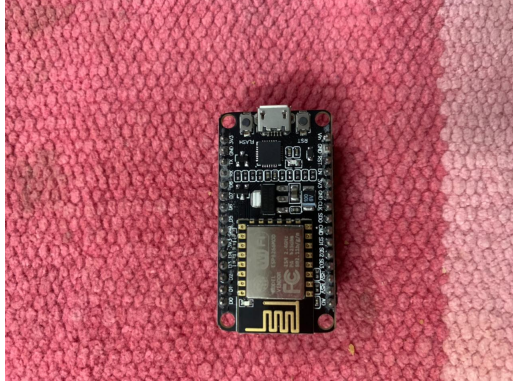
Programming NodeMCU ESP8266 with Arduino IDE:

The NodeMCU Development Board can be easily programmed with Arduino IDE since it is easy to use. Programming NodeMCU with the Arduino IDE will hardly take 5-10 minutes. All you need is the Arduino IDE, a USB cable and the NodeMCU board itself. You can check this Getting Started Tutorial for NodeMCU to prepare your Arduino IDE for NodeMCU.

Uploading your first program:

Once Arduino IDE is installed on the computer, connect the board with the computer using the USB cable. Now open the Arduino IDE and choose the correct board by selecting Tools-Boards-NodeMCU1.0 (ESP-12E Module), and choose the correct Port by selecting Tools¿Port. To get it started with the NodeMCU board and blink the built-in LED, load the example code by selecting Files¿Examples¿Basics¿Blink. Once the example code is loaded into your IDE, click on the ‘upload’ button given on the top bar. Once the upload is finished, you should see the built-in LED of the board blinking.

Figure 10: NodeMCU ESP8266



## 2.CHANNEL RELAY

Relays are most commonly used switching device in electronics. Make sure the load you are using falls into this range.

Relay Working:

The above circuit shows a bare-minimum concept for a relay to operate. Since the relay has 5V trigger voltage we have used a +5V DC supply to one end of the coil and the other end to ground through a switch. This switch can be anything from a small transistor to a microcontroller or a microprocessor which can perform switching operating. You can also notice a diode connected across the coil of the relay, this diode is called the Fly back Diode. The purpose of the diode is to protect the switch from high voltage spike that can produced by the relay coil. As shown one end of the load can be connected to the Common pin and the other end is either connected to NO or NC. If connected to NO the load remains disconnected before trigger and if connected to NC the load remains connected before trigger.

Applications of Relay:

1. Commonly used in switching circuits.
2. For Home Automation projects to switch AC loads.
3. To Control (On/Off) Heavy loads at a pre-determined time/condition.

## 3. USB CABLE

Universal Serial Bus (USB) is an interface to establish communication between devices and a host controller (usually personal computer). An USB system architecture consists of a host controller, a USB ports, and multiple connected devices. Additional USB hubs may be included allowing branching into a tree structure with up to five tier levels.

Use it to connect Arduino Uno, Arduino Mega 2560, Arduino 101 or any board with the USB female A port of your computer. Cable lenght is approximately 1m.

Figure 11: Channel Relay



#### 4. CONNECTING WIRES

Connecting wires allows an electrical current to travel from one point on a circuit to another because electricity needs a medium through which it can move. Most of the connecting wires are made up of copper or aluminum. Connecting wires allows an electrical current to travel from one point on a circuit to another, because electricity needs a medium through which to move. In the case of computers, wires are embedded into circuit boards, carrying pulses of electricity that are interpreted as binary signals of zeros and ones.

#### 5. SMARTPHONE

We need smartphone for installation of blynk application.

#### 6. BULB AND BULB HOLDER

We used 12 Watt LED bulb with bulb holder as a component connected to relay.

Figure 12: USB Cable



Figure 13: Connecting wires

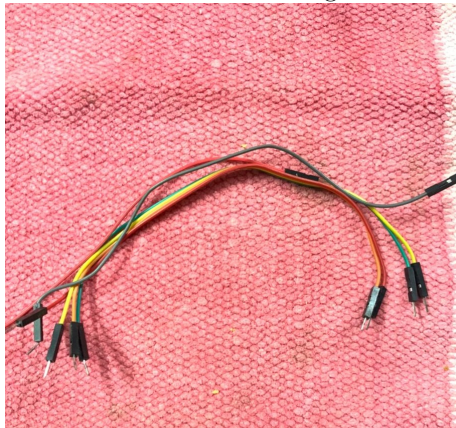


Figure 14: Blynk installation on smartphone



Figure 15: Bulb used



#### 7. FAN

Used Fan as second component connected to relay.

#### 8. POWER SUPPLY-We have 240 watt

### SOFTWARE REQUIREMENT

#### 1. ARDUINO IDE

Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module. It is an official Arduino software,



Figure 16: Fan used



making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.

It is easily available for operating systems like MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment. The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.

This environment supports both C and C++ languages. The Arduino IDE (Integrated Development Environment) is used to write the computer code and upload this code to the physical board. The Arduino IDE is very simple and this simplicity is probably one of the main reason Arduino became so popular. We can certainly state that being compatible with the Arduino IDE is now one of the main requirements for a new microcontroller board.

## 2.WIFI

WiFi-enabled smart devices are usually less expensive compared to their counterparts. It is also much easier to find DIY devices that use WiFi, making it a cheaper option. WiFi is already a necessity and is in most households. It's easier to purchase devices that are already WiFi-enabled. For people who are new to home automation, the protocols like Z-Wave and ZigBee might be totally new and sound too technical. Plus, they require a new hub. No Need for Extra Hardware.

WiFi takes the lead when it comes to simplicity - that is, when you don't need to connect dozens of devices and only a minimal number of devices for a home automation set-up. Since it is very common, you don't need to invest on extra

hardware when you only need the basic setup for your home automation system. WIFI also Spelled as Wi-Fi is a local area wireless technology. It allows an electronic device to transfer data or connect to the internet using ISM radio bands. It is an underlying technology of wireless local area network (WLAN). Wi-Fi allows computers and other devices to communicate over a wireless network. Wi-Fi network components are based on the one of the 802.11 standards developed by the IEEE and adopted by Wi-Fi alliance. It provides a standard way to connect with wireless . network. Wi-Fi is the trademark of the Wi-Fi alliance and used as a brand name for products using the IEEE 802.11 standards.

### 3. BLYNK APP

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things. There are three major components in the platform:

- A).Blynk App - allows to you create amazing interfaces for your projects using various widgets we provide.
- B).Blynk Server - responsible for all the communications between the smart-phone and hardware. You can use our Blynk Cloud .
- C).Blynk Libraries - for all the popular hardware platforms - enable communication with the server and process all the incoming and outgoing commands.

Features of blynk app:

- A).Set of easy-to-use Widgets.
- B).Direct pin manipulation with no code writing.
- C).Easy to integrate and add new functionality using virtual pins.

## 5.2 Snapshots of system with brief detail of each

CONFIGURE THE BLYNK APP:

1. Download the Blynk App from Google play store or App store.
2. Create a new project in the Blynk app. Enter the project name and choose the device. In this IoT project, I have used NodeMCU, so I have selected NodeMCU.

3. After that Blynk will send an Auth Token to the registered email id. The Auth Token will be required while programming the ESP8266.

ADDING WIDGETS IN BLYNK APP:

1.Now to control the 2 relay module you have to add 2 button widgets in Blynk app, Open the project in the Blynk App Click on the “+” icon on the top. select the Button.

2.Click on that button and select the output pin –> D1 and Mode –> Switch.

3.In a similar way create switches with D2 pin to control the second relay.

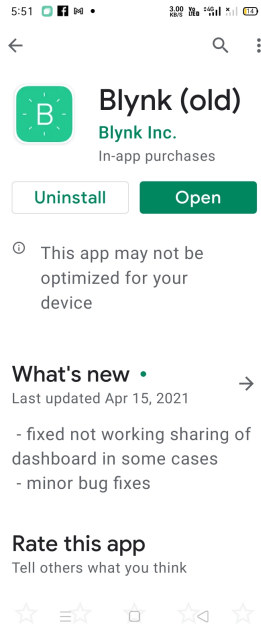
PROGRAM NODEMCU WITH ARDUINO IDE:

1.For this IoT based home automation project, I have used the Arduino IDE to program NodeMCU. First update the Preferences –> Additional boards Manager

2.Then install the ESP8266 board from the Board manager .

3.After that install the Blynk library.

Figure 17: Configuring blynk app



#### CODING PART FOR THIS PROJECT:

Enter the following WiFi credential and Authentication token in the code:

- 1.WiFi Name at “WiFi Name”
- 2.WiFi Password at “WiFi Password”
- 3.Auth Token sent by Blynk at “AUTH TOKEN”
- 4.Then Goto Tools and select the board as “NodeMCU 1.0 ESP-12E” and the proper PORT in Arduino IDE.
- 5.Then click on the upload button to program the NodeMCU board.
- 6.After uploading the code, the blue LED connected with pin D0 should turn on as shown in the picture.
- 7.If the wifi is not available, the blue LED will turn off. Then you can control the relay module with push buttons.
- 8.If the internet comes back the NodeMCU will automatically connect with the WiFi and the blue LED will turn on.

#### CONNECT HOME APPLIANCES WITH RELAY:

- 1.Now, connect FAN AND BULB with relay module as per the above circuit.
  - 2.Please take proper safety precaution while working with high voltage.
- BULB AND FAN AUTOMATION USING ARDUINO AND WIFI SYSTEM IS NOW READY

Figure 18: Creating new project

The screenshot shows a mobile application interface for creating a new project. At the top, there is a green header bar with a back arrow and the text "Create New Project". Below this, there is a text input field labeled "Project Name". Underneath the input field, there is a section titled "CHOOSE DEVICE" with a dropdown menu currently displaying "ESP8266" and a downward arrow. Below the device selection, there is a section titled "THEME" with two buttons: "DARK" (which is highlighted in green) and "LIGHT". At the bottom of the form, there is a large green button labeled "Create". The entire interface is set against a dark background. The status bar at the very top shows the time as 5:51 and various system icons.

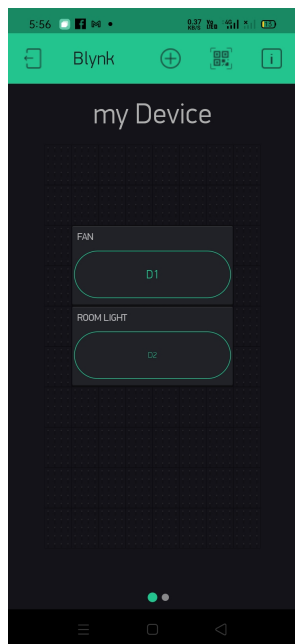


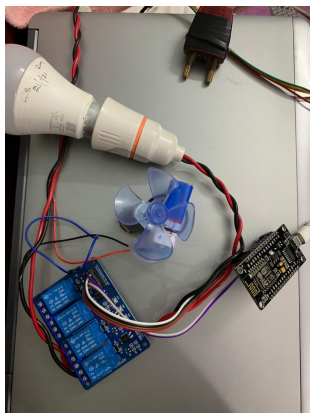
Figure 19: Creating Switches

Figure 20: Code

```
#define BLYNK_PRINT Serial
/* Fill-in your Template ID (only if using Blynk.Cloud) */
// #define BLYNK_TEMPLATE_ID "YourTemplateID"
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "nfV50oiJBnGsZLOXW6VIBbRR8KMj7JuJ";
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "Realme 1";
char pass[] = "12345678";
void setup()
{
  // Debug console
  Serial.begin(9600);

  Blynk.begin(auth, ssid, pass);
  // You can also specify server:
  // Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
  // Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
}
void loop()
{
  Blynk.run();
}
```

Figure 21: Final Circuit



## Chapter 6

# Conclusion and Future Scope

### 6.1 Conclusion

The idea is to create automated bulb and fan. The output of this project is an array of home appliances that are controlled over the smartphone with connection to WIFI. This project is a successful outcome of continuous and tireless effort from all the project members, supervisors, college faculty, colleagues and other helping hands. This project has been a really great experience and opportunity to learn and to experiment. Moreover, the authors got the chance to closely experiment and learn about what goes into designing and developing home automation systems. We are very much delighted that we explored this topic as our minor project. We have created a part of home automation system of our own, and to be closely related with the technology that is of a great interest of study and research today and is sure to revolutionize the way of living of people in the days to come. We were successful in developing a system through which the user was given the ability to control their household appliances through mobile application and wifi. This gives users the ability to automate bulb and fan without the need to buy expensive smart appliances. Many advancements can be added to this project like adding motor to control window drapes, fire sensor to prevent mishaps etc. This specification allows the implementers of the system to keep a clear idea of what tasks need to be accomplished and never get confused by the complexity of the task at hand. If, while constructing the system, any part failed to be realized, it was possible to work around the problem and still manage to implement a system that was working towards the final goal. The lesson learned is that more time spent designing results in less time wondering what went wrong with the implementation. Throughout the project many problems arose that could not be solved during the designated time period.

## 6.2 Future Scope

As mentioned earlier our project is just part of home automation. So there are a lot of other scopes for this project. Future scope for this project involves making homes even smarter. Homes can be interfaced with automating other home appliances, using sensors including motion sensors, light sensors and temperature sensors and provide automated toggling of devices based on conditions. More energy can be conserved by ensuring occupation of the house before turning on devices and checking brightness and turning off lights if not necessary. The system can be integrated closely with home security solutions to allow greater control and safety for home owners. The next step would be to extend this system to automate a large scale environment, such as offices and factories. Home Automation offers a global standard for interoperable products. Standardization enables smart homes that can control appliances, lighting, environment, energy management and security as well as the expandability to connect with other networks.



# Bibliography

- [1] M.Shell.(10 March 2021) Home Automation with Blynk and NodeMCU ESP8266-IoT.[online] Available : <https://iotcircuitHub.com/nodemcu-esp8266-blynk-home-automation/>
- [2] K Dejima and T. Onama.(15 May 2021) Tutorial Arduino.[online] Available : <https://www.arduino.cc/en/Tutorial/HomePage/>
- [3] Yaka Oklada.(20 April 2021) Home Automation using NodeMCU (Part1).[online] Available : <https://www.rs-online.com/designspark/home-automation-using-nodemcu-part1>
- [4] C.Zhinu.(15 March 2021) Home Automation - How to Add Relays to Arduino.[online] Available :<https://www.instructables.com/Home-Automation-How-to-Add-Relays-to-Arduino/>
- [5] S. Zhang.(15 March 2021) Introduction to Home Automation.[online] Available : <https://medium.com/@smarthomeguru/introduction-to-home-automation-6bdd0a4783c>
- [6] Kivabi Soam.(15 March 2021)it's an internet thing!.[online] Available :<https://www.supanet.com/smart-home-automation-benefits-vs-challenges-a19161.html>
- [7] P.K.T. Mok.(15 March 2021)IEEE xplore!.[online] Available :<https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin>