

ACKNOWLEDGEMENT

We are privileged to have this opportunity to express our gratitude and acknowledge everyone's never ending support and valuable contributions for our project.

Prima facie, We would like to express our sincere gratitude to our advisor **Prof. Drashti Garadharia, Prof. Gaurang Patel and Prof. Shivam Ribadiya** for the continuous support of our project and related research, motivation and immense knowledge.

Our sincere thanks also goes to **Dean Dr. Amit Ganatra** who provided us an opportunity to work on a project and be able to present the same.

Last but not the least, we would like to thank our friends and family for supporting us spiritually throughout this project and for always being a constant source of inspiration.

We also place on record, our sense of gratitude to one and all, who directly or indirectly, have lent their hand in this venture.

ABSTRACT

Comfort is becoming a major priority in the 21st century. So the revolutions of computing and smart environment came into existence. Some technologies like Ubiquitous/pervasive and ambient intelligence satisfy the maximum need of smart world but these technologies are not tightly coupled with the internet, so the people need another technology extension. Internet of Things (IoT) is an ideal buzzing technology to influence the internet and communication technologies. IoT allows people and things to be connected anytime, anyplace, with anything and anyone, by using ideally in any path/network and any service. Home automation system achieved great popularity in the last decades and it increases the quality of life. In this report, an overview of home automation systems is discussed. This report presents a design and prototype of the Home Automation system. The main part is NodeMCU, which has inbuilt Wi-Fi module, which will help in controlling devices over the Internet. It supports a wide range of home automation devices like power management systems. In addition to this, other third party services like IFTTT and Adafruit MQTT are used which will allow the user to control their devices using the Google Assistant.

TABLE OF CONTENTS

Abstract.....	1
Acknowledgement.....	2
Chapter 1: PROJECT DEFINITION.....	4
Chapter 2: DESCRIPTION	5
2.1 Project Summary	5
2.2 Purpose	5
2.3 Scope	5
2.4 Objective	6
Chapter 3:PROJECT MANAGEMENT.....	7
3.1 Project Planning	7
3.1.1 Project Effort and Time	7
3.1.2 Roles and Responsibilities.....	7
Chapter 4: SYSTEMS REQUIREMENTS STUDY.....	8
4.1 User Characteristics	8
4.2 Hardware Requirements	8
4.3 Software Requirements	10
Chapter 5: MAJOR FUNCTIONALITY.....	12
Chapter 6: FLOWCHART	13
Chapter 7: SCREENSHOTS OF THE PROJECT OUTPUT.....	14
Chapter 8: LIMITATIONS.....	19
Chapter 9: OUTCOME.....	20
Chapter 10: FUTURE ENHANCEMENTS	21
Chapter 11: BIBLIOGRAPHY	22

Chapter 1: PROJECT DEFINITION

This project presents a proposal for home automation using voice via Google Assistant. Home automation or domotics(i.e. Domestic + Robotics) a term for home automation coined by Jim Hill has been evolving drastically. We saw many home automation technologies introduced over these years from Zigbee automation to Amazon Echo, Google Home and Home from Apple. It has become a craze these days.

Human assistants like housekeepers were a way for millionaires to keep up their homes in the past. Even now when technology is handy enough only the well to do people of the society are blessed with these new smart home devices, as these devices costs are a bit high. However, not everyone is wealthy enough to be able to afford a human assistant, or some smart home kit. Hence, the need for finding an inexpensive and smart assistant for normal families keeps growing.

This project proposes such inexpensive system. It uses the Google Assistant, the IFTTT application, the Adafruit application and the NodeMCU microcontroller as the major components along with a relay board comprising of 2 channel. Natural language voice is used to give commands to the Google Assistant. All of the components are connected over the internet using Wi-Fi which puts this system under the IoT.

Chapter 2: DESCRIPTION

2.1 Project Summary

Since the current world strives for easier and better comfort levels life, technologies are sharpening their edges to automate home appliances for saving user's time.

Modern systems generally consists of switches and sensors connected to a central hub called a gateway from which the system is controlled with a user interface that is interacted either with a mobile phone software , tablet, computers or a web interface ,often but not always via internet cloud services. World's demand for electricity had grown 85% between 2010 and 2018. This increase is more than today's total use of electricity in India, USA, Japan , Australia combined. We can't decrease the electricity gross electricity wasted each year by turning off our home appliances when not in use.

Imagine a life where it would be better if everything such as warming bath water and adjusting the room temperature were already done before we reach our home just by giving a voice command. So, when people would arrive home, they would find the room temperature, the bathwater adjusted to their suitable preferences, and they could relax right away and feel cosier and rather, feel more homely. In this way both time and energy would be saved.

Human assistants like housekeepers were a way for millionaires to keep up their homes in the past. Even now when technology is handy enough only the well to do people of the society are blessed with these new smart home devices, as these devices costs are a bit high. However, not everyone is wealthy enough to be able to afford a human assistant, or some smart home kit. Hence, the need for finding an inexpensive and smart assistant for normal families keeps growing.

This report exhibits one such inexpensive system using

- **Google Assistant**
- **IFTTT**
- **Adafruit MQTT**
- **NodeMCU ESP8266 microcontroller**

as the major components along with supporting components like connecting wires, batteries, leds etc. Natural language voice is used to give commands to the Google Assistant. All of the components are connected over the internet using **WiFi** which puts this system under the **IoT**.

In our project using the above mentioned components when user gives command like "Turn on light" to Google Assistant then led would glow, same is true vice-versa. According to the command required values would be passed IFTTT which is the connecting link between Google assistant and Adafruit and data would be passed to NodeMCU and respective output is generated.

2.2 Purpose

The purpose of this project is to create and learn Home Automation System using IoT and Google Assistant. With the help of this project lots of time and energy could be saved thereby enhancing and making life easier. The project also aims at making it more user friendly and cost efficient.

2.3 Scope

The project will achieve the task of controlling lights/leds using Google Assistant and NodeMCU ESP8266 at real time.

The project will be able to process the given commands into required formats to be able to pass and understood by the third party applications used as an interface.

2.4 Objective

The objective of this project was to implement and explore the already known concept of using Arduino and IoT and also to learn to manage different kinds of errors.

Idea of creating an inexpensive time saving and energy efficient project motivated us to switch to IoT Home Automation using Google Assistant.

Chapter 3: PROJECT MANAGEMENT

3.1 Project Effort and Time

The participants of this project have given about 2-3 hours per 3 weeks to accomplish this project. Also we took knowledge from various sites and help to rectify the errors from IoT faculties.

3.2 Roles and Responsibilities

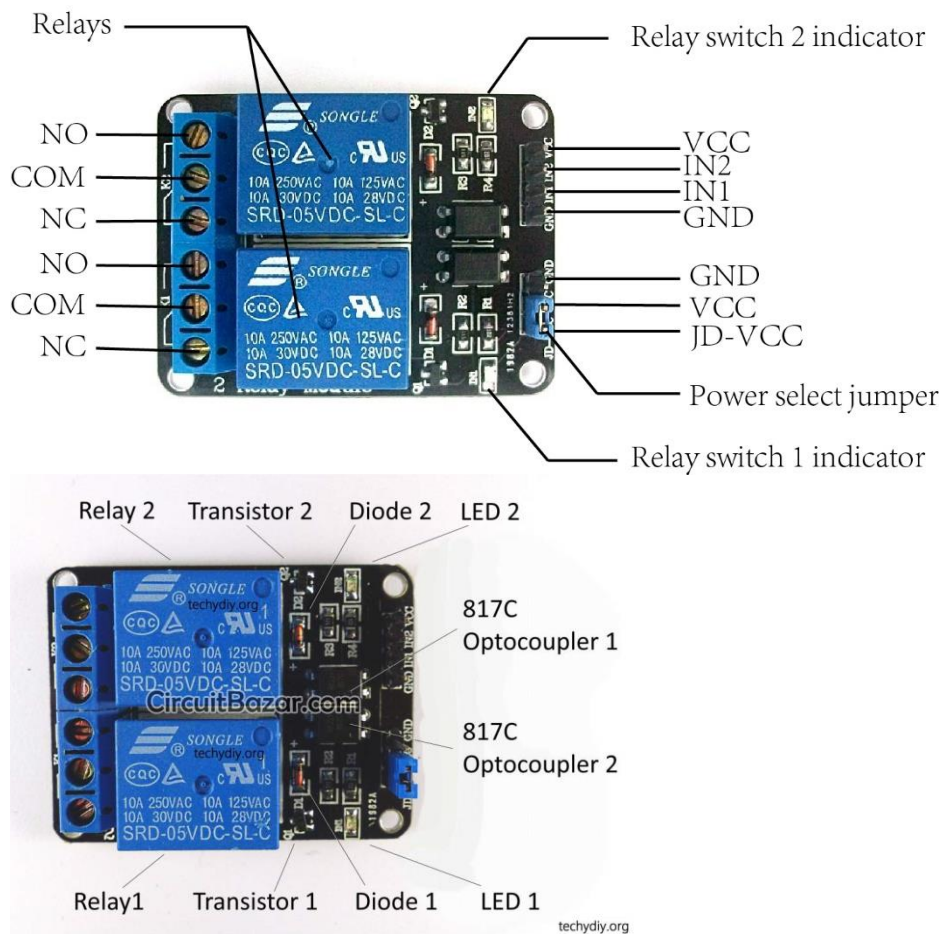
Following Team Members have worked effectively and have undertaken the below mentioned tasks:

- Mansi Nakrani :- Handling and solving errors, Testing the application for vulnerabilities and bugs, PPT, Report.
- Rutika Mehta :- Coding the logic, Creating a Design of flow, PPT, Report.
- Varada Nair :- Researching in the related field of work, Making connections, PPT, Report.

- Inter-IC sound interface with sharing pins with general purpose input output pins
- 10 bits Analog to digital converter

• 2-Channel Relay module

A relay is an electrically operated switch. Many relay use an electromagnet to, mechanically operates a switch, but other operating principles are also used, such as solid state relay. Relay are used where it is necessary to control a circuit by a separate low power signal, or where several circuits must be controlled by one signal. Relay were used extensively in telephone exchange and early computers to perform logical operation.



- **LEDs**
- **Connecting Wires**
- **Bread Board**
- **Android Phone(With Google Assistant)**
- **Electrical Appliances**

4.3 Software Requirements

To build home automation application, We used four different platforms :

- **Arduino IDE – 1.6.4**

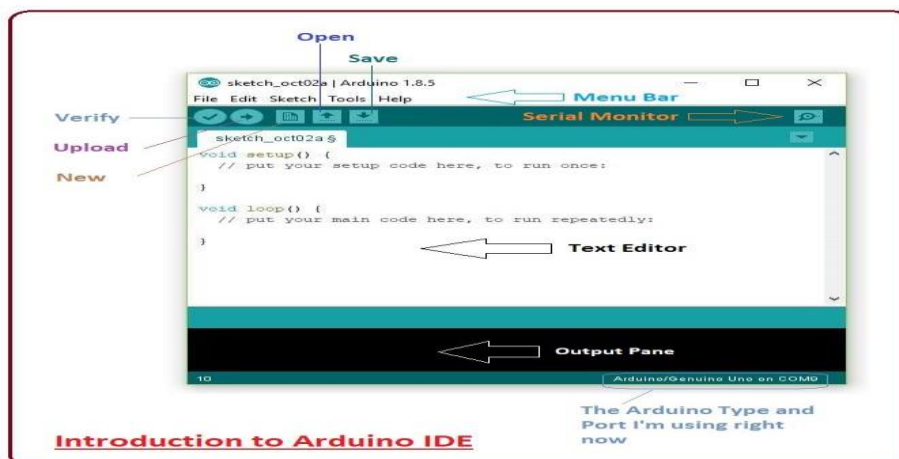
Arduino is an open-source platform used for building electronics projects consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

A minimal Arduino C/C++ sketch, as seen by the Arduino IDE programmer, consist of only two functions.

- **Setup ():** This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch.
- **Loop ():** After setup () has been called, function loop () is executed repeatedly in the main program. It controls the board until the board is powered off or is reset.

Most Arduino boards contain a light-emitting diode (LED) and a load resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions. A typical program for a beginning Arduino programmer blinks a LED repeatedly.

This program uses the functions pin Mode (), digital Write (), and delay (), which are provided by the internal libraries included in the IDE environment. The program is usually loaded in the Arduino by the manufacturer. Arduino IDE and C language allow the programming of the low level registers in the atmega328P. Instructions like `DDRB=0b00000001` for changing PORTB input/output pins are allowed. Fig 4.1 shows the basic program of Arduino IDE.



- **Google Assistant**

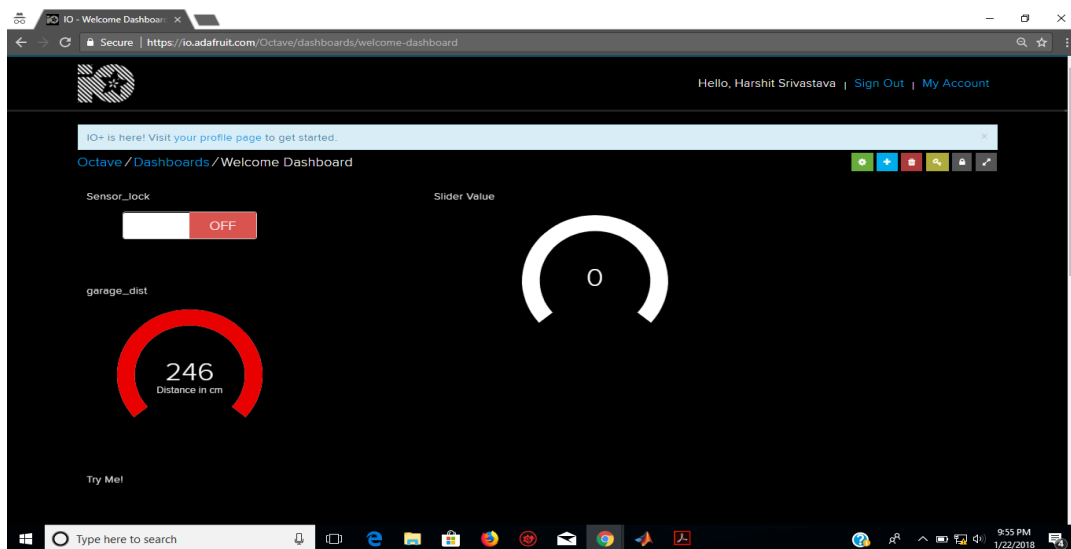
The Google Assistant is a software which allows its users to control all the apps in their device to be controlled directly through it. It allows the users to control and command most of the apps in their devices using voice commands. This provides more convenience to the people as they only have to command the google assistant thorough voice command.



- IFTTT for android 4.0.2+



- Adafruit MQTT

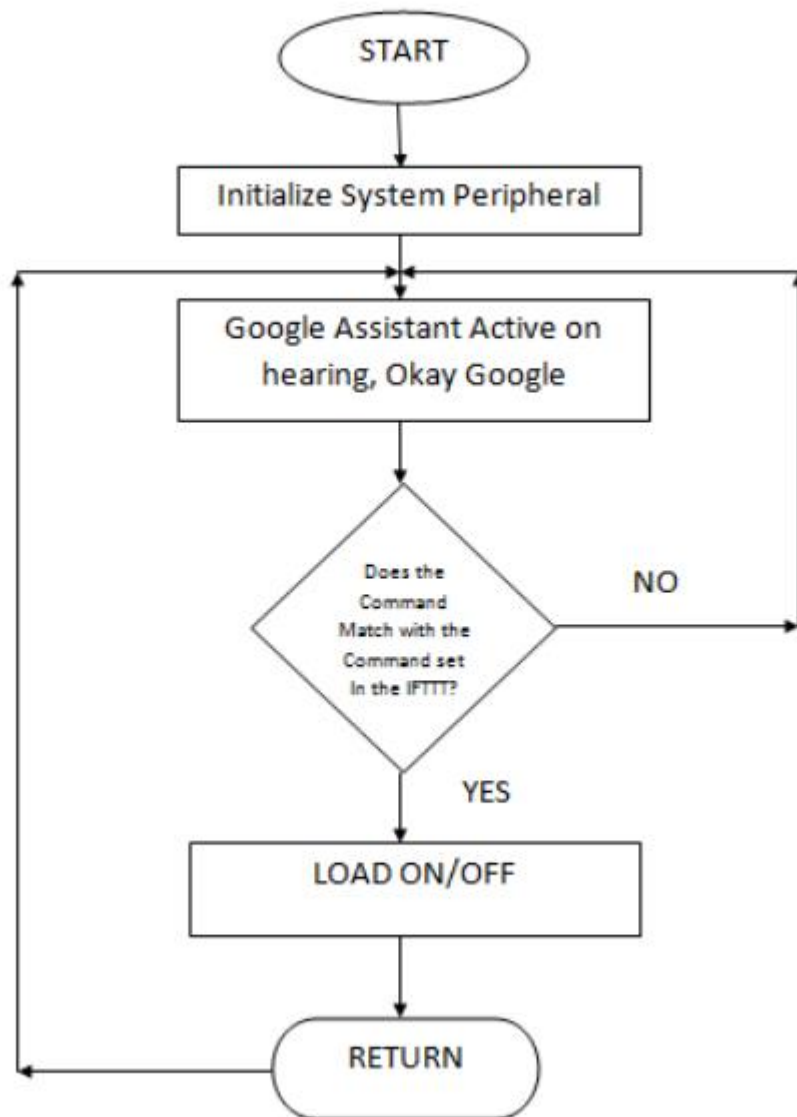


We used IFTTT to use google assistant service and Adafruit service in chain. So, when I use google assistant to control light of my home by saying Ok google, turn the light ON or OFF. Then IFTTT interpret the message and can send it to Adafruit's dashboard as a understandable command to the created feed.

Chapter 5 :MAJOR FUNCTIONALITY

Our project aims to provide the most easy and efficient way to interact with home appliances by giving voice commands in human (natural) language. We plan on eliminating the tedious process of clicking through various application screens with just one voice command. The natural language processing in the project provides a personal connection with our system. The user sends a voice command to the mobile device through which its google account is linked, which interprets the message and sends the appropriate command to the specific application i.e. IFTTT. The voice command given by the user is interpreted by the mobile device using Natural Language processing. The mobile device acts as a central console; it determines what operation must be done by which appliance to fulfill the user's request. The IFTTT then sends the request to Adafruit and checks the trigger action given by the user and works accordingly and sends the output to IFTTT. IFTTT further sends the output to google assistant and thus the user gets the desired output. The appliances are connected to the mobile device through an Arduino Board that establishes the concept of Internet of Things. The Arduino Boards are interfaced with the appliances and programmed in such a way that they respond to mobile inputs. Our project automates the operation of every single appliance connected with the arduino in the house, which greatly reduces the power consumption due to excess use/wastage of the appliance's services.

Chapter 6 : FLOW CHART (GRAPHICAL REPRESENTATION OF OUR PROJECT)



Chapter 7 : SCREENSHOTS OF YOUR PROJECT OUTPUT



The screenshot shows the Arduino IDE interface with the file 'GoogleAssitance_Automation_' open. The code is written in C++ and includes several preprocessor directives for configuration. The code is as follows:

```
#include <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
/*
#define Relay1          5

#define WLAN_SSID       "Co-Hive L5"           // Your SSID
#define WLAN_PASS       "P!!c@123"           // Your password

*/
/*
#define WLAN_SSID       "Rut"                 // Your SSID
#define WLAN_PASS       "password"           // Your password
*/
#define WLAN_SSID       "Mansi"              // Your SSID
#define WLAN_PASS       "mdsa62564"         // Your password

/***** Adafruit.io Setup *****/

#define AIO_SERVER       "io.adafruit.com"
#define AIO_SERVERPORT   1883                // use 8883 for SSL
#define AIO_USERNAME     "Mansi06"           // Replace it with your username
#define AIO_KEY           "bc792973d5364bd59fadca3fdcac92c8" // Replace with your Project Auth Key

/***** Global State (you don't need to change this!) *****/

// Create an ESP8266 WiFiClient class to connect to the MQTT server.
WiFiClient client;
```

The IDE interface includes a menu bar (File, Edit, Sketch, Tools, Help) and a toolbar with icons for saving, running, and other functions. The status bar at the bottom shows the line number 14 and the word 'Node'.

```

// Create an ESP8266 WiFiClient class to connect to the MQTT server.
WiFiClient client;
// or... use WiFiClientSecure for SSL
//WiFiClientSecure client;

// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);

/***** Feeds *****/

// Setup a feed called 'onoff' for subscribing to changes.
Adafruit_MQTT_Subscribe Light1 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME"/feeds/LED"); // FeedName

void MQTT_connect();

void setup() {
  Serial.begin(115200);

  pinMode(5, OUTPUT);

  // Connect to WiFi access point.
  Serial.println(); Serial.println();
  Serial.print("Connecting to ");
  Serial.println(WLAN_SSID);

  WiFi.begin(WLAN_SSID, WLAN_PASS);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();

  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());

  // Setup MQTT subscription for onoff feed.
  mqtt.subscribe(&Light1);
}

```

```

void loop() {

    MQTT_connect();

    Adafruit_MQTT_Subscribe *subscription;
    while ((subscription = mqtt.readSubscription(5000))) {
        if (subscription == &Light1) {
            Serial.print(F("Got: "));
            Serial.println((char *)Light1.lastread);
            int Light1_State = atoi((char *)Light1.lastread);
            digitalWrite(5, !(Light1_State));

        }
    }
}

void MQTT_connect() {
    int8_t ret;

    // Stop if already connected.
    if (mqtt.connected()) {
        return;
    }

    Serial.print("Connecting to MQTT... ");

    uint8_t retries = 3;


    while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected
        Serial.println(mqtt.connectErrorString(ret));
        Serial.println("Retrying MQTT connection in 5 seconds...");
        mqtt.disconnect();
        delay(5000); // wait 5 seconds
        retries--;
        if (retries == 0) {
            // basically die and wait for WDT to reset me
            while (1);
        }
    }
    Serial.println("MQTT Connected!");
}

```

In the above code the user sends the data as voice command and the Arduino IDE further sends the request to IFTTT via Adafruit using MQTT server and user gets the desired output.


[Shop](#) [Learn](#) [Blog](#) [Forums](#) [Videos](#) [Adabox](#) [IO](#)

Mansi06 ▾

 [Profile](#) [Feeds](#) [Dashboards](#) [Triggers](#) [Services](#) AIO Key

Mansi06 / Feeds


Actions ▾

Search 

Group / Feed


Last value

☐ Default

☒ LED  1








[Shop](#) [Learn](#) [Blog](#) [Forums](#) [Videos](#) [Adabox](#) [IO](#)

Mansi06 ▾

 [Profile](#) [Feeds](#) [Dashboards](#) [Triggers](#) [Services](#) AIO Key

Mansi06 / Dashboards / RVM

RVM

Led

OFF



My Applets

**If You say "Turn Off Led",
then Send data to LED
feed**

By mansi060501

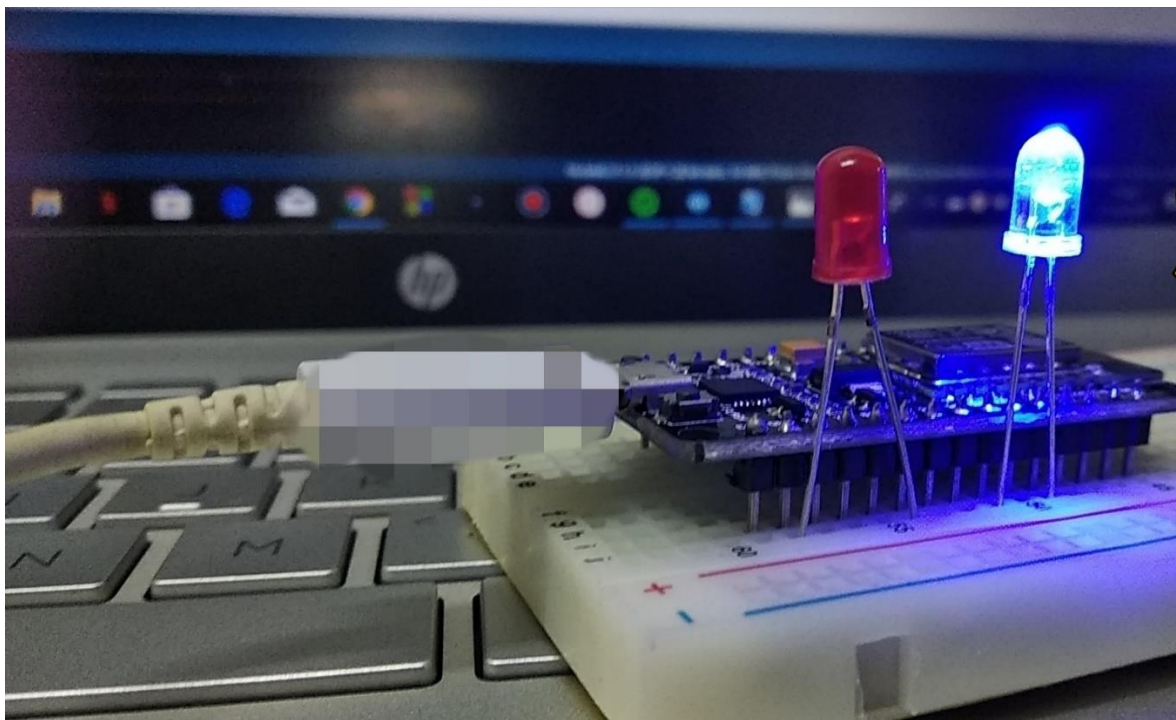
☐ Connected



**If You say "Turn On Led",
then Send data to LED
feed**

By mansi060501

☐ Connected



Chapter 8 : LIMITATIONS OF PROJECT

In many ways, voice assistants simply replace button pushes with voice commands. It is still up to consumers to consider the context of an action. When multiple devices are involved in achieving a high-level objective, users have to orchestrate the device actions with each other, and with other activities required to reach the goal. Consider these points:

- **Learning Curve:**
An automated home comes with a few drawbacks. For a start, owning a Smart House means having to learn how to use it. You will have to adapt to new technology, whether you are comfortable with it or not. Once you do, access to every device you need at your fingertips. The benefits of convenience are yours, but you need to learn how to reap them.
- **Reliability:**
Technology can create disasters if it falls into the wrong hands. Hackers can break into your home once they obtain the password to your lock. Further, you may encounter privacy issues if your camera captures unwanted footage of your neighbour. To avoid such problems, transfer your video via wireless to other areas which you can monitor.
- As more flexible natural language understanding technology is becoming available, interpretations of speech commands may become ambiguous. With commands resulting into actions, misunderstandings can be risky.
- Voice assistants support only one-way “conversations”. The appliances cannot talk back, asking for clarification of intent.
- They depend on an Internet connection, and the obstacles it has in each home that can make it less than reliable.

These shortcomings limit voice assistants from elevating their status from transactionary to really helpful. They need a semantic level of interaction to support more complex activities.

Chapter 9 : OUTCOME

The aim of this project was to propose a cost effective voice controlled (Google Assistant) home automation controlling general appliances found in one's home. The approach discussed in the paper was successful as GACHA's (Google Assistant Controlled Home Automation) design was successfully implemented. This system is highly reliable and efficient for the aged people and differently abled person on a wheel chair who cannot reach the switch for the switching ON/OFF the device and are dependent on others.

Chapter 10 : FUTURE ENHANCEMENT

Android app will also develop for easily use. In Android app there will be direct buttons for ON or OFF the system or to receive the OTP. For more security purpose camera module can also be implemented on the system. If any person attempt to enter in home with more than three time wrong password then at that time camera module will be activate. And camera module will capture the image of person who trying to attack on system. It can use antivirus so that hacking of the system can be difficult.

Google Assistant is a fast-emerging technology. Currently most of the smartphone market is leading in Android which comes with Google Assistant technology. Even iPhones can be installed with Google Assistant from the App Store. This helps the device to be future-proof without becoming easily obsolete. However, there are constant updates coming to the Google Assistant SDK, making it much easier to register more and more devices in the cloud console without putting in a lot of effort.

Chapter11 : BIBLIOGRAPHY

http://ijariie.com/AdminUploadPdf/Home_Automation_Using_Google_Assistant_ijariie8241.pdf
<https://www.irjet.net/archives/V5/i5/IRJET-V5I5395.pdf>
<https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/>
<https://electropeak.com/learn/speak-to-arduino-control-all-parts-by-google-assistant/#>
<https://www.instructables.com/id/Control-House-Lights-With-Google-Assistant-Using-A/>
<https://www.viralsciencecreativity.com/tutorials/google-assistant-home-automation-iot-nodemcu-esp8266>
<https://www.viralsciencecreativity.com/tutorials/google-assistant-home-automation-iot-nodemcu-esp8266>
<https://www.hackster.io/xerax/google-assistant-to-command-esp8266-with-nodemcu-lua-ifttt-51ac4e>
<https://create.arduino.cc/projecthub/pawan-kumar3/home-automation-using-google-assistant-88c5c8>
<https://www.electronicwings.com/users/lokeshc/projects/control-home-appliances-using-google-assistant>
<https://www.electronics-lab.com/project/home-automation-using-nodemcu-esp8266-board/>
<https://www.ijarnd.com/manuscripts/v3i2/V3I2-1181.pdf>
<https://iarjset.com/wp-content/uploads/2018/06/IARJSET-3.pdf>
<http://ijesc.org/upload/a623d731289508e4d1d46bc38584ab8d.IoT%20Based%20Home%20Automation%20using%20NODE%20MCU.pdf>
<https://core.ac.uk/download/pdf/55305294.pdf>