B.Sc. Eng. Final Year Project

Smart Home Automation Using Raspberry Pi with Smart Phone

Shimaa Zakaria Abu Qassim,
Abdulazez K. AboEljoben

Palestine Technical College – Deir El-Balah,

State of Palestine
Ministry of Higher Education & Scientific Research
Palestine Technical College - Deir El Balah

دولة فلسطين
وزارة التعليم العالي والبحث العلمي
كلية فلسطين التقنية - دير البلح

B.Sc. Eng. Final Year Project

Title:

Smart Home Automation Using Raspberry Pi with Smart Phone

Graduation Project submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Systems Engineering at the Palestine Technical College – Deir El-Balah.

Prepared by:

Shimaa Zakaria Abu Qassim 1649011005.
Abdulazez Khaled Abu Eljoben 1649011017.

Supervisor:

Eng. Said Abu Al-Rous.

2021

CERTIFICATE OF ORIGINALITY

This is to certify that we are responsible for the work submitted in this project, that the original work is our own except as specified in the references and acknowledgements, and that original work contained herein have not been taken or done by unspecified source, or person.

ACKNOWLEDGMENT

Praise and thanks to Allah first and foremost, whose blessing enabled our to accomplish this project.

We wish to express our deepest appreciation to our supervisor Eng. Said Abu Al-Rous for relentless guidance, helpful suggestion, close supervision and moral encouragement to complete this project.

Special thanks to our parents and to all our teachers we have had.

Our sincere thanks to all those who directly or indirectly helped us to complete this project.

ABSTRACT

This project focuses for control monitoring home appliances, providing security.
This project presents "discuss" a smart home automation using raspberry pi and smart phone to access and control appliances remotely.
The proposed system used to control a lot of devices such as light, fan, electrical appliances. For example, user can turn light on/off using home assistant app which used as mobile app.
Through this project we used google assistant and google home as a tool to control these app by voice.
The results indicate that our project is very good and able to control all sensors and loads very well.

Keyword:
Smart Home, Automation, Raspberry Pi, Smart Phone, Home Assistant, Google Home, Google Assistant, control, monitoring, Home,

Table of Contents

LIST OF FIGURES

s

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

ACRONYM	Definition of Acronym
RasPi	Raspberry pi
HDMI	High Definition Multimedia Interface
Wi-Fi	Wireless Fidelity
App	Application
API	Android Application Interface
IOT	Internet of Things
PIR	Passive infrared sensor
MQ	Gas Sensor
DHT11	Temperature & Humidity Sensor
ID	Identifier
UI	User Interface
SD	Secure Digital
I/O	Input/output
IP	Internet Protocol
CPU	Central processing unit
RAM	Random Access Memory
USB	Universal Serial Bus
OS	Operating system
HASSIO	

Chapter One

INTRODUCTION

1. INTRODUCTION

1.1. Overview

Home automation is using pc or more than one to control and monitor home appliances automatically or using smart phone. Home automation systems offer a variety of services and functions, Fire monitoring, Remote lighting control, Thermostat control, Appliance control, Home automation security systems and cameras, Alarm systems and Voice-activated control. home automation includes lighting, reminder systems, security system, protection and entertainment appliances.

Various communication methods are used for home automation such as WIFI, Bluetooth, GSM, and ZigBee.

The purpose of this project is to design an intelligent system application to control the home remotely using a smartphone. This system is a WIFI based Android or IOS Application.

Home automation based on Raspberry pi or Arduino. Raspberry pi provides a low-caste platform, Perform multiple tasks at the same time, powerful than Arduino, suitable for complex project, faster than Arduino.

Home assistant is an Open source home automation that considers local control and privacy first. Perfect to run on a Raspberry Pi.

Here we will detail the drivers and problems that led us to develop this control system. In the methodology section, we will be more detail and how to solve through the practical part of our system which is a control circuit connected to a smartphone application via Wi-Fi, then we will present our results.

1.2. Problem Statement and Importance of the Project

This part belongs to the introduction, Home automation refers to the control of home appliances using computer technology. Also, the presence of smart devices capable of sensing physical phenomena and translating them into a flow of information data, as well as devices capable of triggering actions, increases safety, security, comfort and energy savings, thus becoming more and more common. It also provides a remote interface for home appliances to provide control and monitoring on a web browser.
The importance of Home Automation is to improve life style of user such as

I. Measuring home condition using sensors.
II. Managing home appliances.
III. Controlling home access.

So, our project is helpful for disabled, elderly people.
These people can perform their activities easily using mobile Apps and internet.
Therefore, the proposed solution to help these people using any android or IOS phone to control any device which connected to our system. On the other hand, control and monitor all of appliance in the home such as saving power protecting it from theft.
So, we use RPI with Internet connected to home devices to perform all required functions

1.3. Objectives

The aim of this project is to implement the following tasks:

- Design and build the following electronic circuit to:
  - Controlling the light, Fans, TV, Air Condition.
  - Controlling the doors, main door and garage door.
  - Controlling indoor (gas sensor, night light sensor, motion sensor, fire detector sensor, temperature and humidity sensor)
- Web page customization.
- Connect your Raspberry Pi IP to the mobile app.
- Testing and result.

1.4. Project Elements

There are many different devices in the house that differ from one house to another and need automatic control without any effort. We have implemented the project on a miniature model that simulates a house and contains some devices. server architecture of the system is shown in the **Fig 1.1** below

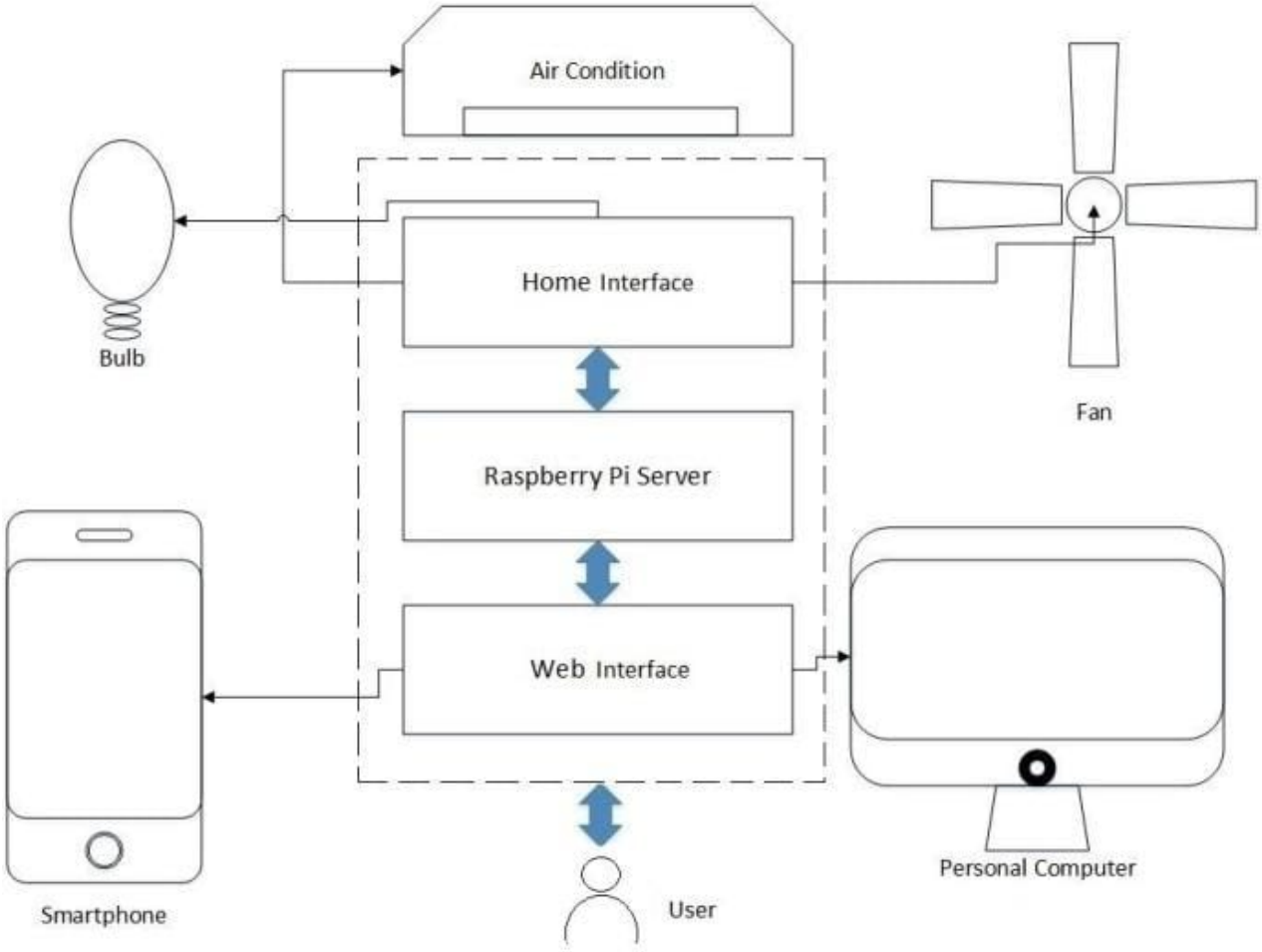


Fig 1. 1: server architecture of the system.

Next, we mention the hardware pieces used in the project. **Fig 1. 2:** shows project element are used.

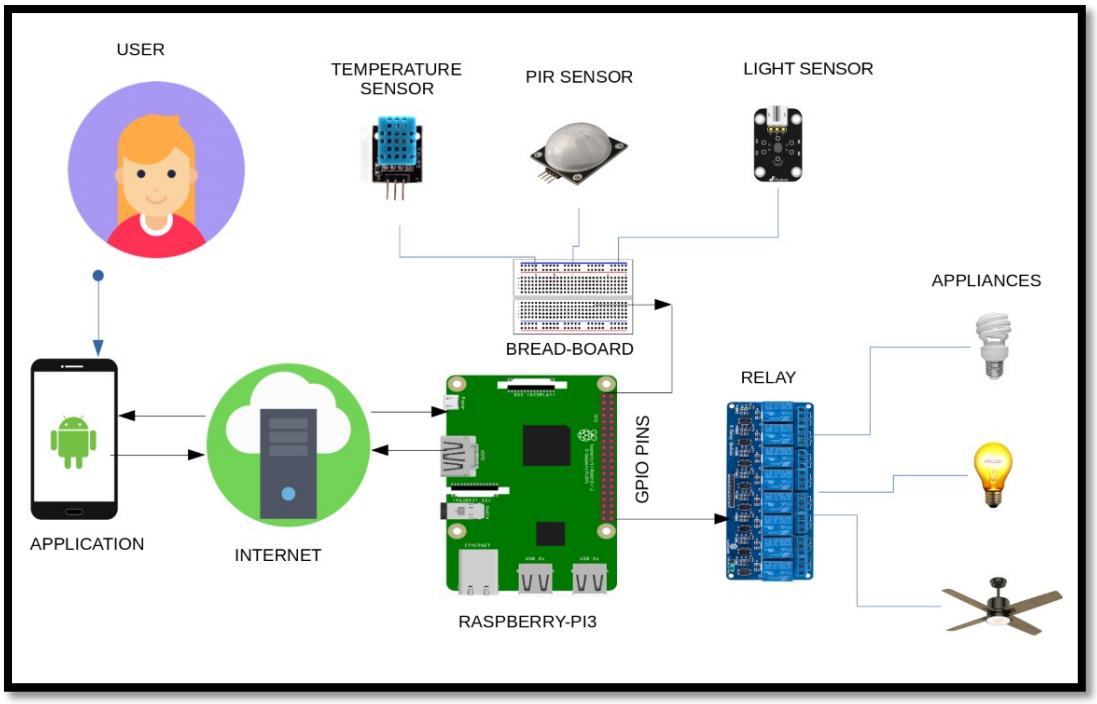


Fig 1. 2: project element.

In this proposal, two modes of operation are designed. The first is the manually automated mode where the device is monitored and accessed manually using the cell phone. The proposed flow chart for this process. In this case, the hardware detection state is performed by Raspberry Pi. The user can select the required devices using a smartphone. The selected devices can be turned on / off according to the appropriate resolution.
We also covered the method of information collection, system analysis and design. Implementation steps are as shown in the **Fig 1. 3**

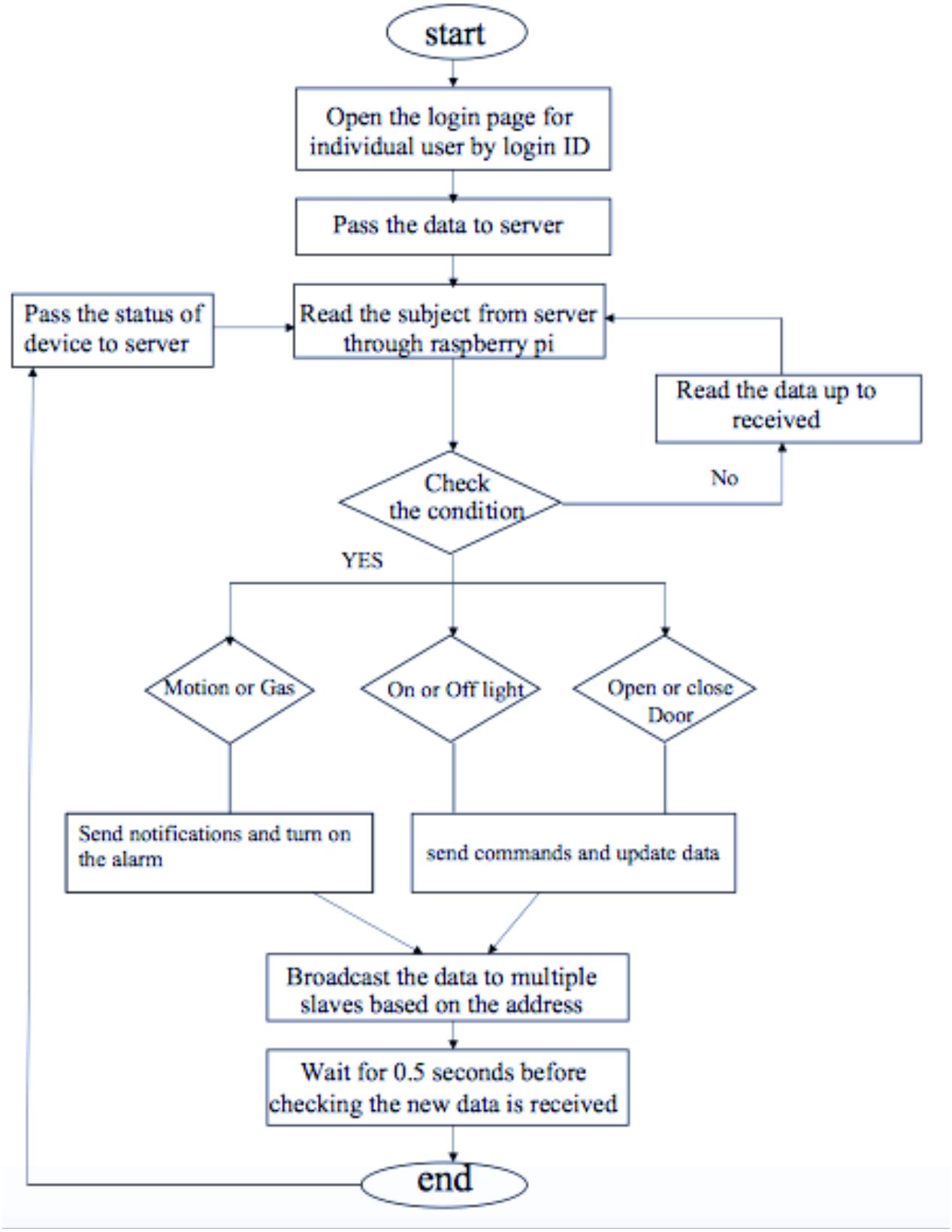


Fig 1. 3: Flow Chart Description of Proposed System.

1.5. Related Work

In this section we discuss the most important work and research related to controlling appliances in the home. We will focus only on systems that are very similar to our control system. Using a Raspberry Pi in home control is not new a vast number of publications have been covered this issue.

In 2013. Rozita Teymourzadeh Provide analysis and implementation of home automation technology using a Global System for Mobile Communications (GSM) modem to control home appliances such as light, police system and SMS security system. The proposed research work focuses on the functionality of the GSM protocol, which allows the user to control the target system away from residential areas using bandwidth. The concept of serial communication and AT commands has been applied in the development of the GSM-based smart home automation system. Homeowners will be able to receive feedback status of any home appliances under control whether they are turned on or off remotely from their mobile phone [1].

Muhammad Asadullah and Khalil Ullah, they proposed a remote-controlled home automation system using an Arduino board, a Bluetooth module, a smartphone, an ultrasonic sensor and a humidity sensor. A smartphone app is used in the proposed system which allows users to control up to 18 devices including home appliances and sensors using Bluetooth technology [2].

In 2017, Chandramohan he provides a flexible, low-cost and flexible home control and monitoring system with the help of a small integrated web server with Internet Protocol (IP) communication to access and control equipment and devices remotely using an Android-based smartphone application [3].

In 2016, Kolhe Ujvala S. developed a system that provides remote control of home appliances and also provides protection against intrusion when the home host is not at home. Its purpose is to provide electrical energy and human energy. This project was created with the help of a microcontroller and a Raspberry Pi. Various devices are connected to the microcontroller and the sensor is connected using a wireless network. [4].

There are other apps that rely on Raspberry Pi and Wi-Fi mainly to build a smart home.

1.6. Overview of Report

This section outlines the overall structure of the report, and provide a brief description for each chapter.

Chapter two, provides basic information that helps the reader understand the basics of our project. Moreover, it includes the definition, concept and limitations of Home assistant, Raspberry Pi, as well as the definition of all electronic parts, i.e. practical parts that we need to build the control circuit.

The third chapter includes the techniques used in the analysis and design of this project, a schematic diagram of user interfaces for mobile applications to fit the modern form of advanced mobile application technology to work well for additions.

Chapter Four, includes the methods and techniques used in implementing this project, such as the tools used to implement this project, and user interfaces for mobile applications.

Finally, Chapter Five, summarize the work presented in this project and indicate suggestion for future work.

Chapter two

System Design and Integration

2. SYSTEM DESIGN AND INTEGRATION

2.1. Overview

This Smart Home system allows users to control electrical appliances and lighting remotely at all times, depending on the control circuit in addition to relying on Raspberry Pi supported by a home assistant app, the control circuit is connected to the mobile device via Wi-Fi where a home assistant app will be connected to a website to record all devices that we want to control remotely, in addition to lighting, electricity and water meters.

Difference systems, techniques and software packages related to the smart home automation were reviewed and evaluated based on performance criterion the project was divided into different phases.

- Build an RPI controlled home automation system with sensor and actuations.
- Design a web interface/mobile App.

2.2. System description

This paper presents design and implementation concepts for a wireless real-time home automation system based on Raspberry pi microcontroller as central controllers. We used the Home Assistant application as a user interface that enables it to control devices with ease and is suitable for all ages, in addition to the Google Home application to send voice commands and control the house from outside. The system controls lighting, fans, electrical appliances, main doors and garage door, and we can know the temperature and humidity of the house, in addition to detecting movement, fire, or gas leakage, issuing alarms and sending notifications to the user of the application.

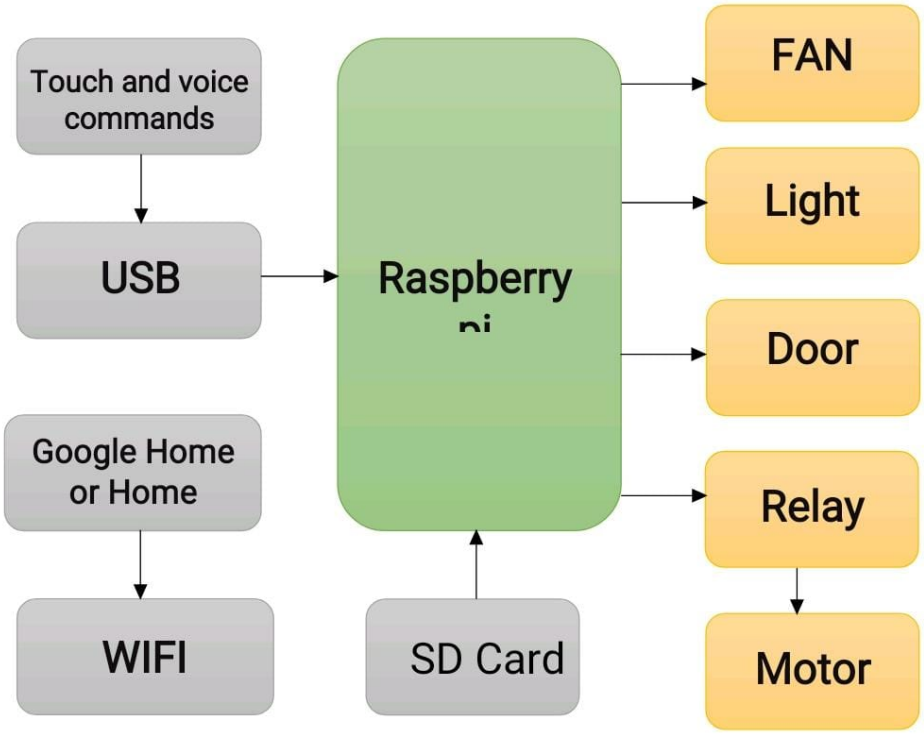


Fig 2. 1: Block diagram for home automation system.

2.3. Hardware Design

2.3.1. Raspberry pi

The Raspberry Pi is a small, powerful, cheap, hackable, and education-oriented computer board introduced in 2012. It works the same way as a standard PC, requiring a keyboard for entering commands, a display unit and a power source. This credit card sized computer with many offers and at an affordable price of $100-150 is an ideal platform for interacting with many devices. The vast majority of system components - its CPUs and graphics, audio and communications hardware along with a 2 GB memory chip, are built on a single component. The Raspberry Pi board shown in Figure 2_1 and Figure 2_2 contains basic elements (processor, graphics chip, program memory - RAM) and other optional hardware (different interfaces and connectors for peripheral devices). The Raspberry Pi processor is a 64-bit, 1.5GHz system on a chip. The SD Flash memory acts as a hard drive for the Raspberry Pi processor. The unit is powered via micro USB connector while internet connection may be via Ethernet cable or via USB dongle (Wi-Fi connection) .

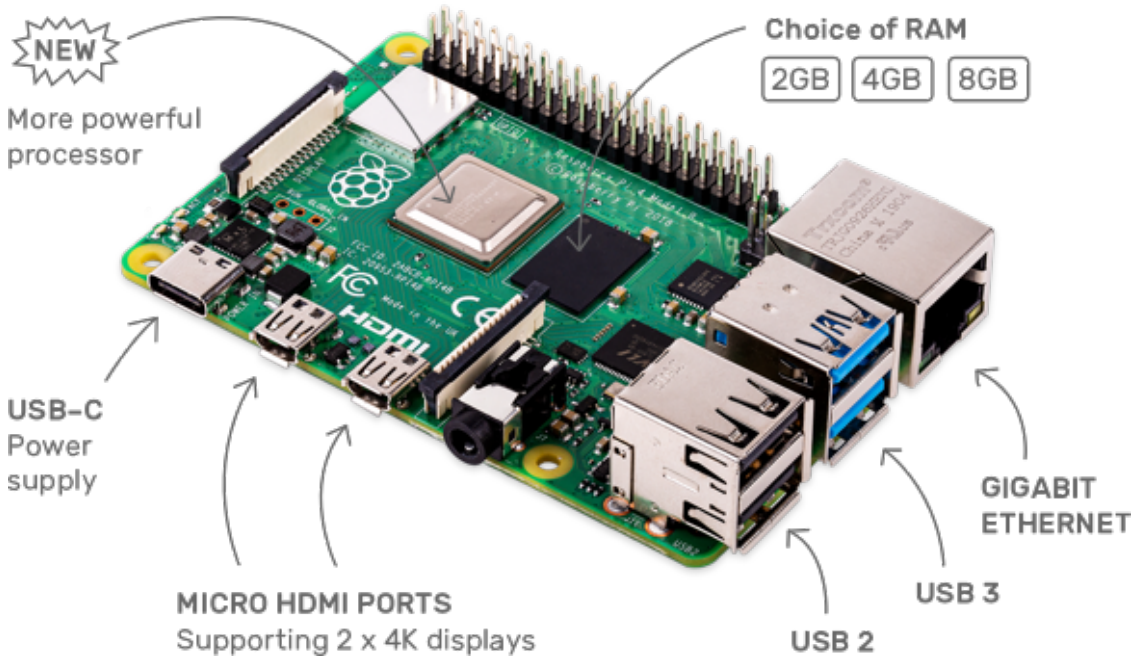


Fig 2. 2: Raspberry pi 4 module b.

The Raspberry Pi is most likely the top device for newcomers as well as experienced users to run Home Assistant on. It's small, cheap, quiet, and doesn't use much electricity. A Raspberry Pi was my gateway drug into home automation and self-hosted software.
Home Assistant currently recommends you run the software on either a Raspberry Pi 3 Model B, a Raspberry Pi 3 Model B+, or Raspberry Pi 4 Model B.
It is not recommended that you attempt to run Home Assistant on anything older than the Raspberry Pi 3 Model B. This includes the original Raspberry Pi and the Raspberry Pi 2. And while the Raspberry Pi Zero might be a popular platform for applications such as Pi-hole, it just isn't powerful enough to handle Home Assistant.
So, we used Raspberry pi 4 module b in this project.

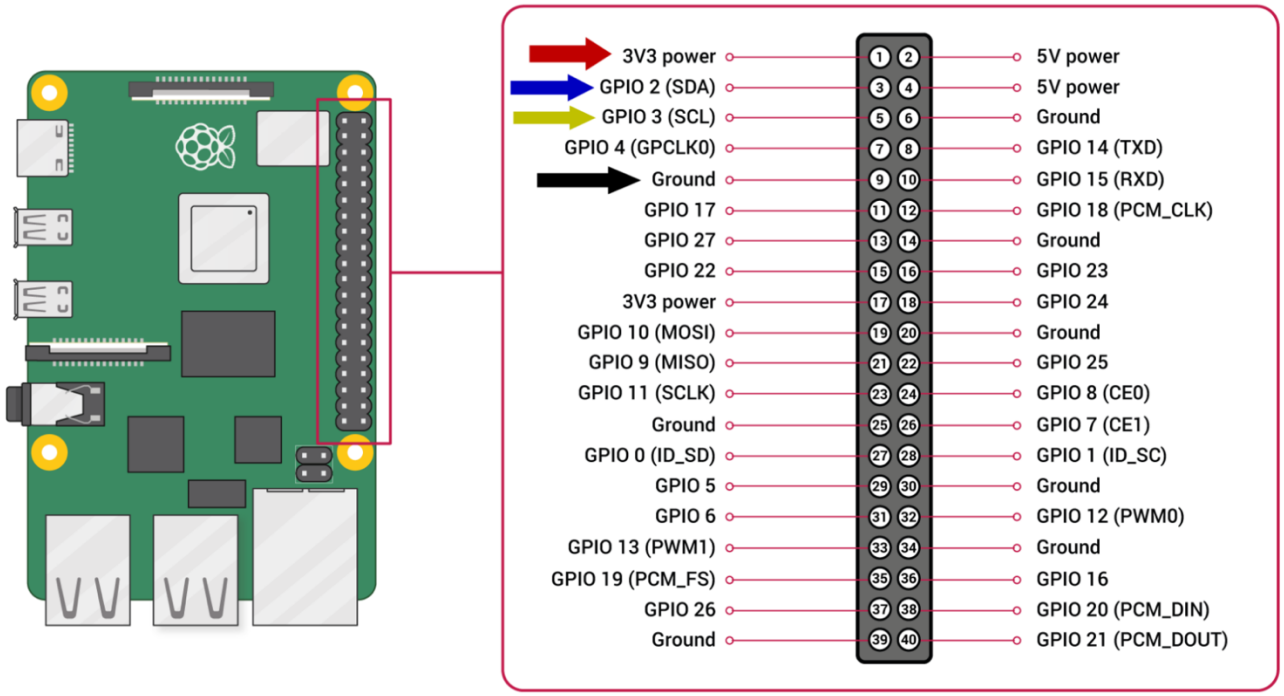


Fig 2. 3: GPIO pins in RPI.

13. Fan.
15. Living room.
19. Bed room.
21. Main room.
29. Lamb-2.
31. Night light.
33. Open Garage.
35. Close Garage.
16. DHT11.
18. RIP sensor.
22. MG-5 sensor.
38. Open main door.
40. Close main door.

2.3.2. Sensors

Automation sensors are the eyes and ears of your home, notifying you and other smart devices of changes in status 24/7. Automation sensors control your lights and appliances so you can customize how and when to control appliances around your home.

Sensors are a basic platform component in Home Assistant. They monitor the states and conditions of a variety of entities. An entity can be many things. This can include a physical device like a motion sensor that reports the battery level, a web service that retrieves the weather temperature, a built-in function that calculates the sun's elevation relative to your GPS position, or even a custom sensor you may have created to report the free space on your laptop. These are all things reporting different types of information.

Some of these sensors are built-in to Home Assistant, some are created automatically when you add an integration, and some can be created manually.

The sensors that were used in the project:

- Temperature sensor: **DHT11** is a basic and very low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and broadcasts a digital signal on the data pin. It's fairly easy to use, but it requires a precise timing for data acquisition that you can get new data from once every two seconds.

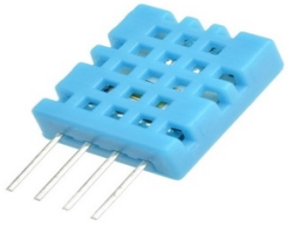


Fig 2. 4: DHT11 Humidity & Temperature Sensor.

- MQ-5 sensor: The Grove - Gas Sensor (**MQ-5**) unit is useful for detecting gas leaks in the home. It is suitable for the detection of liquefied petroleum gas, natural gas and coal gas. Due to its high sensitivity and speed of response, measurements can be made as quickly as possible. The sensitivity of the sensor can be adjusted using a potentiometer.

Fig 2. 5: Gas Sensor (MQ5).

- PIR sensor: The passive infrared (**PIR**) sensor recognizes infrared light emitted by nearby objects, it is a thermal collector consisting of a heat-sensitive crystal. When electromagnetic radiation falls on it, it causes a temperature difference on its surface. The temperature difference generates a current flowing from the hot end to the hot end.

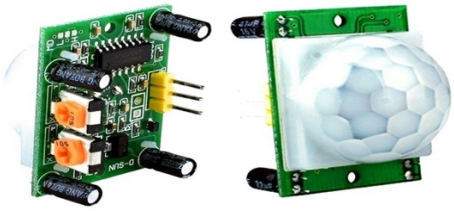


Fig 2. 6: PIR sensor.

2.3.3. Relay Module

Relay is an electromechanical device that uses an electric current to open or close the contacts of a switch. A single channel relay unit is much more than a regular relay, it consists of components that make switching and connection easier and act as indicators to show whether the unit is powered and whether or not the relay is active.

In our project, we used a single relay module, shown in the figure below, which corresponds to every Raspberry Pi entry for electrical appliances. Security in system

All data between your browser and your local instance is encrypted. The local instance has created and owns the certificate, so only the local instance will be able to decrypt the incoming traffic.

Once the user is connected to their Home Assistant instance, they will have to log in with their local credentials. These credentials are only stored locally and cannot be impersonated by anyone.

Fig 2. 7: Relay Module.

- ✓ **How Does A Relay Work?**

The relay uses an electric current to open or close the contacts of a switch. This is usually done using the help of a coil that attracts the contacts of a switch and pulls them together when activated, and a spring pushes them apart when the coil is not energized.

There are two advantages of this system – First, the current required to activate the relay is much smaller than the current that relay contacts are capable of switching, and second, the coil and the contacts are galvanically isolated, meaning there is no electrical connection between them. This means that the relay can be used to switch mains current through an isolated low voltage digital system like a microcontroller.

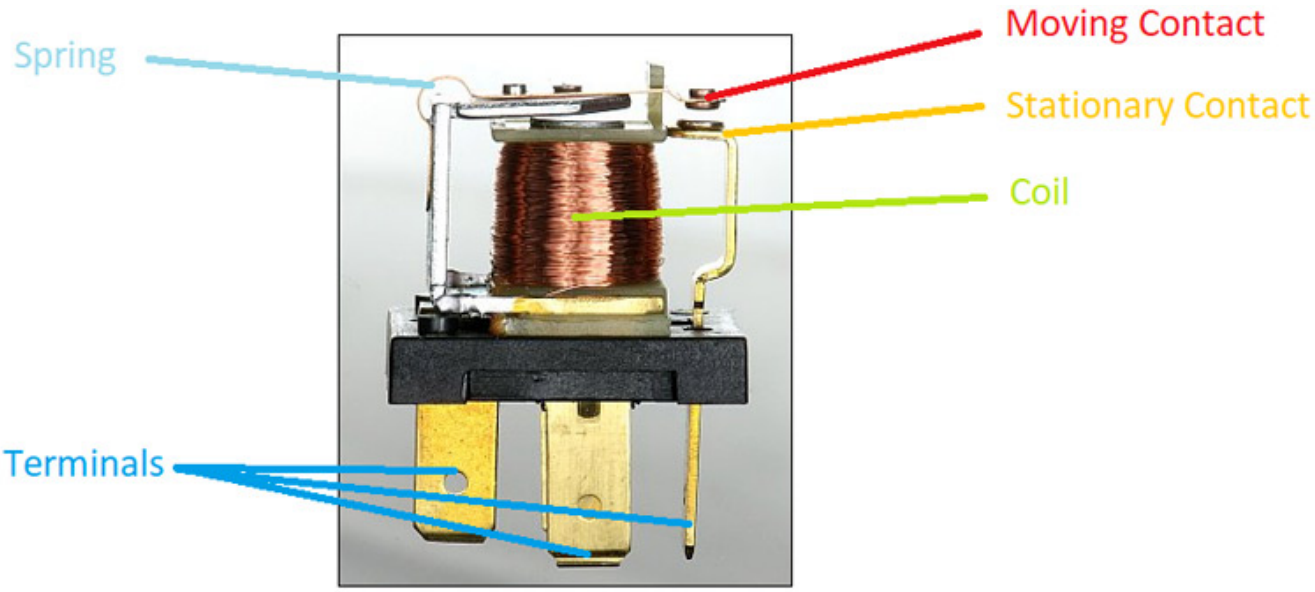


Fig 2. 8: How Does A Relay Work

✓ **Internal Circuit Diagram for Single Channel Relay Module**

The circuit on the PCB is quite simple.

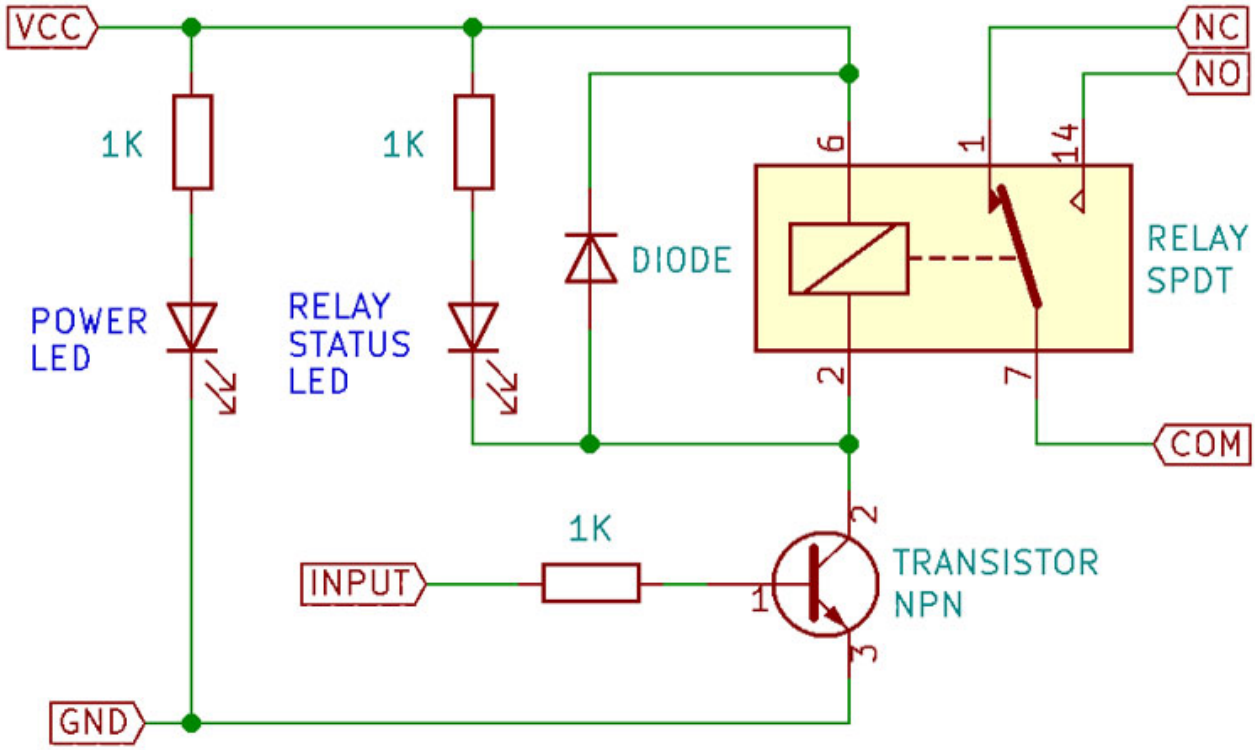


Fig 2. 9:Relay Basic Schematic.

2.4. Software Design

2.4.1. Home Assistant

Home assistant, a Home Automation Hub or Software, is an open-source and free application that is used for home automation. It is a complete local alternative to solutions like SmartThings and HomeBridge. Home assistant enabled the local control of smart home devices, offering better security, greater reliability, and more flexibility than many cloud-connected devices.

It offers you the control of home automation without the need for cloud, thus your home automation system is not dependent on remote servers or internet connection. It will result in fast operation and consistency. It is flexible and supports a majority of smart devices and doesn't use your network.

The home assistant lets you set up schedules and individual smart home devices work together and let the devices as smart as they need to be. You will let Home Assistant find all the smart devices on your Wi-Fi network, link with the device it can control, and provides a clear interface to let them work together based on how and when you want the single command to be executed. It is essential to understand that the Home Assistant cannot control anything itself. A home assistant is just like a master gadget that can command other smart devices to perform the task. A home assistant will be useless and won't do anything if you don't have any smart home device in your home. However, if you are having one or more smart home devices a Home Assistant can be a great way to enhance the functionality and features that will not cost you a fortune. Home Assistant is not very difficult to use and configuring it is inexpensive.

As it is a local service, it means it is not going to deliver data to the cloud, even if it is required to recover data from the internet. The commands and schedules you set up are only for you. It is quite easy to perform setting and programming routines with the help of a web interface hosted by Home Assistant locally. It is important as such services can get very difficult if you are willing to program a wide range of routines, so the simple and nice interface makes it a lot easier to do it.

The home assistant runs on a local server or a Raspberry Pi. You are required to connect Raspberry Pi to the router Ethernet and install Hassio. Hassio only runs this home automation software on the Pi, enabling the software to use its full resources.

A local home automation hub like a Home Assistant has several benefits as compared to cloud-connected devices. Here are some of the advantages of Home Assistant for home automation.

- Speed: When you trigger an activity or issue a command, all the processing occurs in your home as a home assistant is a local controller. There will be nothing sent over the webserver. Also, you won't have too many devices competing for the bandwidth of your home network. Smart devices can be kept in subnets and only connect gadgets that are directly interacting with.
- Compatibility: You don't have to fear the company such as Amazon forbidding your Google devices if you don't use their cloud service to start with.
- Security: When there are few devices connected to the internet, there will be fewer devices allowing the internet into your network. You are allowed to decide what can access your data and how they interact.
- Control: Cloud offers only limited services to the users that their owners want. Using local hubs and controllers, you are free to decide what to do with it. You can have full control over devices and automation and involve as deeply as you want.
- Reliability: What will happen if the cloud service is out of business? Or if servers just go down leaving thousands of users with doors locked or without AC as the home automation hub isn't working? What if service owners stop supporting your specific device? Imagine you lose the internet? Many circumstances lead to your cloud-connected devices getting offline. But if you use a local home automation hub to maintain and control your devices, you don't have to worry about such problems.

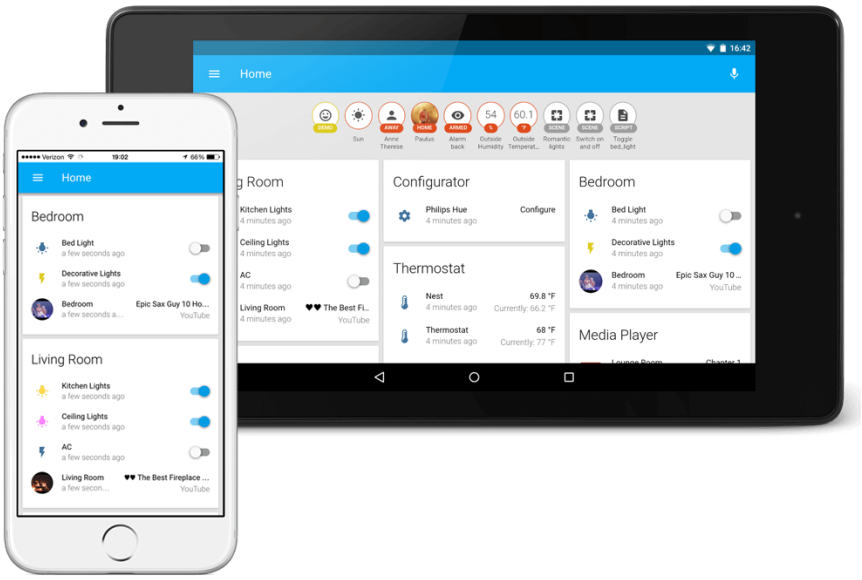


Fig 2. 10: Screenshots of the new UI for home assistant.

2.4.1.1. Architecture

Home Assistant provides a platform for home control and home automation. Home Assistant is not just an application: it's an embedded system that provides an experience like other consumer off-the-shelf products: onboarding, configuration and updating is all done via an easy to use interface.

The operating system provides the bare minimal Linux environment to run Supervisor and Core.

The Supervisor manages the operating system.

The Core interacts with the user, the supervisor and IoT devices & services.

2.4.1.2. Home Assistant Cloud:

No longer worry if you left the garage door open. Quickly access an instance of Home Assistant from your phone, your favorite coffee shop, or at work. All data is fully encrypted between your device and your Home Assistant instance. No snooping.

With Home Assistant Cloud, you can connect your Home Assistant instance with just a few simple clicks to both Google Assistant.

If you can connect it to the Home Assistant, you can now control it with your voice using your Google Home or Android phone.

2.4.1.3. Home Assistant Core

The Home Assistant Core consists of four main parts. On top of this it includes many helper classes to deal with common scenarios, like providing an entity or dealing with locations.

- Event Bus: facilitates the firing and listening of events -- the beating heart of Home Assistant.
- State Machine: keeps track of the states of things and fires a state_change event when a state has been changed.
- Service Registry: listens on the event bus for call_service events and allows other code to register services.
- Timer: sends a time_changed event every 1 second on the event bus.

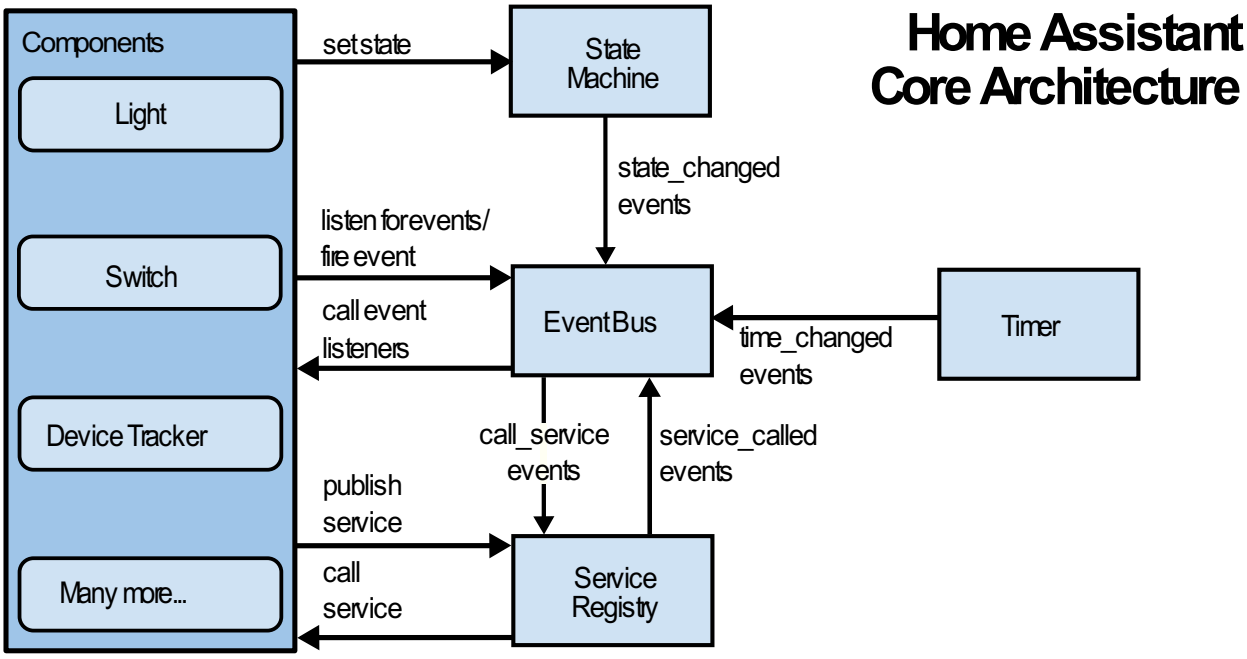


Fig 2. 11: Home Assistant Core Architecture.

2.4.2. Google Home

The Google Home app lets you set up, manage and control Google Home, Google Nest, and Chromecast devices, as well as thousands of app-related home products like lights, cameras, electrical appliances, and more.
Find out what's going on in your home at a glance The Google Home app is designed to show you what's going on in your home and keep you up to date on things you've missed. You can keep track of what's happening in your home at any time and get a summary of the latest happenings in it.
we can control the home appliances from outside the house, and if there is no network on the phone, through the phone data we can control it, also through it we control the house by voice commands
Home Assistant is linked with Google Home, and home appliances are recognized by Home Assistant cloud.

2.4.3. The configuration.yaml file

In Home Assistant, the configuration.yaml file saves data in a hierarchical format to let Home Assistant know what you want to do. This file stores what components you're using, how they are organized and how to automate with them.
Go to the HASSIO shared file, select the configuration folder and open the configuration.yaml file.

2.4.4. Smartphone

Here we describe the modern smartphone and its basic sensors. It is a handheld personal computer with a mobile operating system and an integrated mobile broadband cellular network connection for voice, SMS, and Internet data communication; most, if not all, smartphones also support Wi-Fi. Smartphones are typically pocket-sized, as opposed to tablet computers, which are much larger. They are able to run a variety of software components, known as "apps".

2.5. System Integration

HW and SW is integrated as completed system.

2.5.1. Connection between Project Element

In this system we use Raspberry Pi and establish internet connection for the purpose of automation using IoT by accessing IP address. All home appliances are connected to the Raspberry Pi, and the mobile phone and Raspberry Pi are connected via Wi-Fi. The Raspberry Pi is used as an onboard controller to connect devices via the I/O port.

After completing the work on controlling the house using Raspberry Pi and the smartphone, the project supervisor must give the user a password and the IP of the system, which will enable him to control and monitor the house without trouble and for the time required without delay, we will apply the system to a small geometry simulating a house, as shown in the figure.

Fig 2. 12: small simulating a home.

On the other hand, the application must be open and connected to the Internet so that the process of control is successful, information is recorded and the signal is sent to cut to carry out orders. If the application is not connected to the Internet, the Internet does not store information on the site.

2.5.2. Hardware part

Here we show the sequence of steps we took to accomplish the most important part of our project, the control part. we designed and built the electronic circuit to connect the lights, fans and doors. Tested before connecting to Raspberry Pi. Scheme of the basic control circuit as shown in Figure 4_ 3:

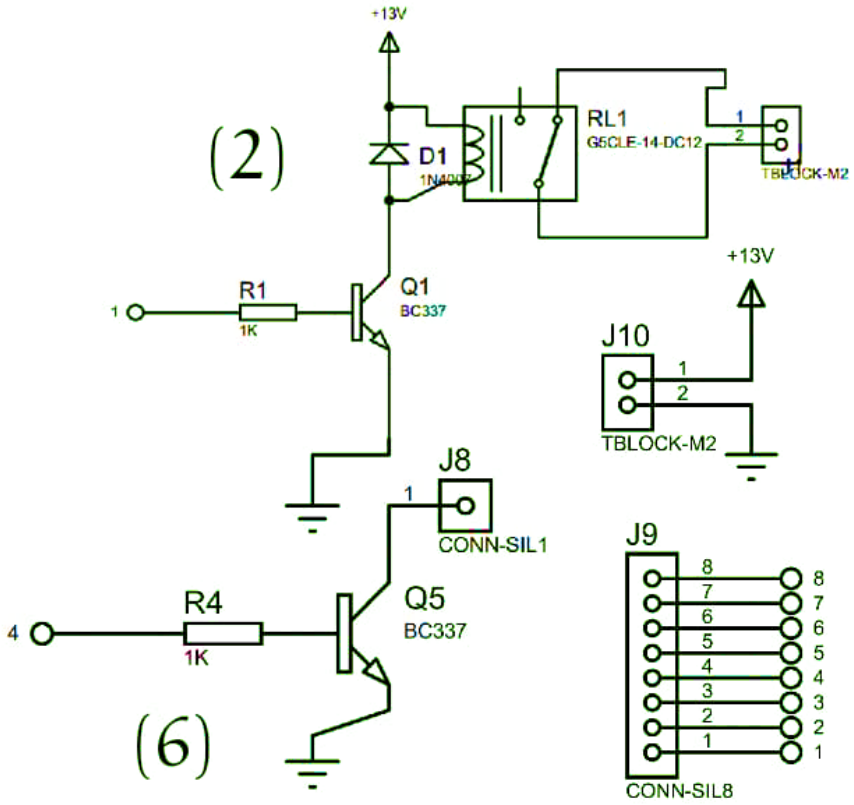


Fig 2. 13: Scheme of the basic control circuit.

After that, we initially designed using a proteus simulation program relay model for the required circuits to control the house, these parts give a clear reaction if any command is given, we learned how to print the paintings and acidification, as in Figure below:

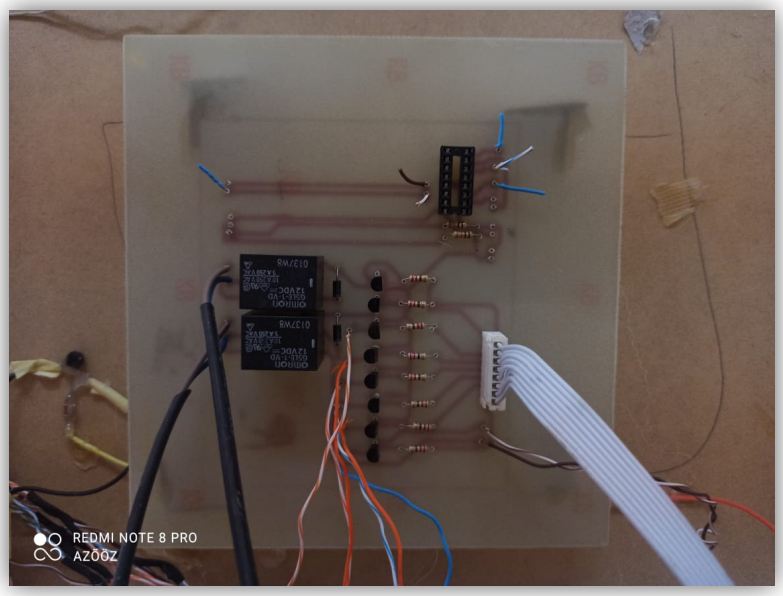


Fig 2. 14: interface circuit modules simulation by proteus.

Next, we made a circuit to control the doors, opening and closing the door. We controlled two doors, the first is the main door of the house, when given the command to open the door, the door pulls to the right and vice versa. The other door is the garage door. When an open command is issued, it pulls the door up, and vice versa. We considered the presence of movement while closing the door, the closing process stops permanently until the movement stops, and the required process is completed. The circuit is as shown in the figure:

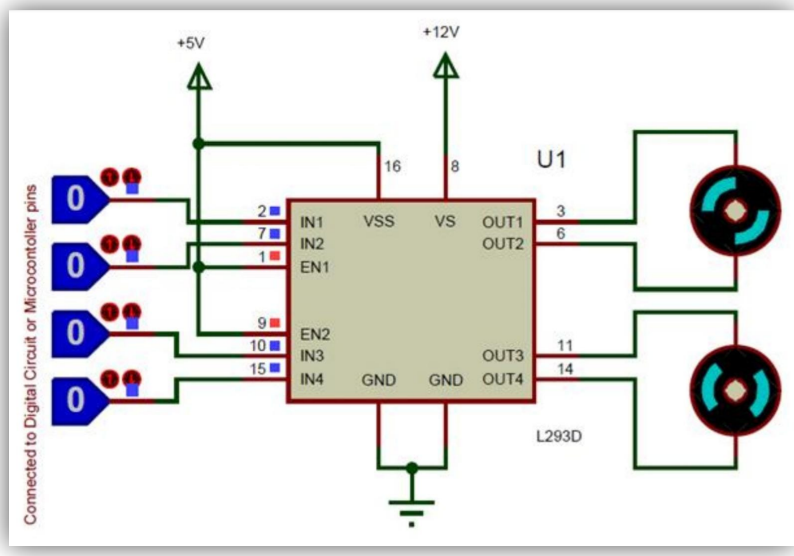


Fig 2. 15: Connected to digital circuit or microcontroller pins.

2.5.3. Software part

I. Installation

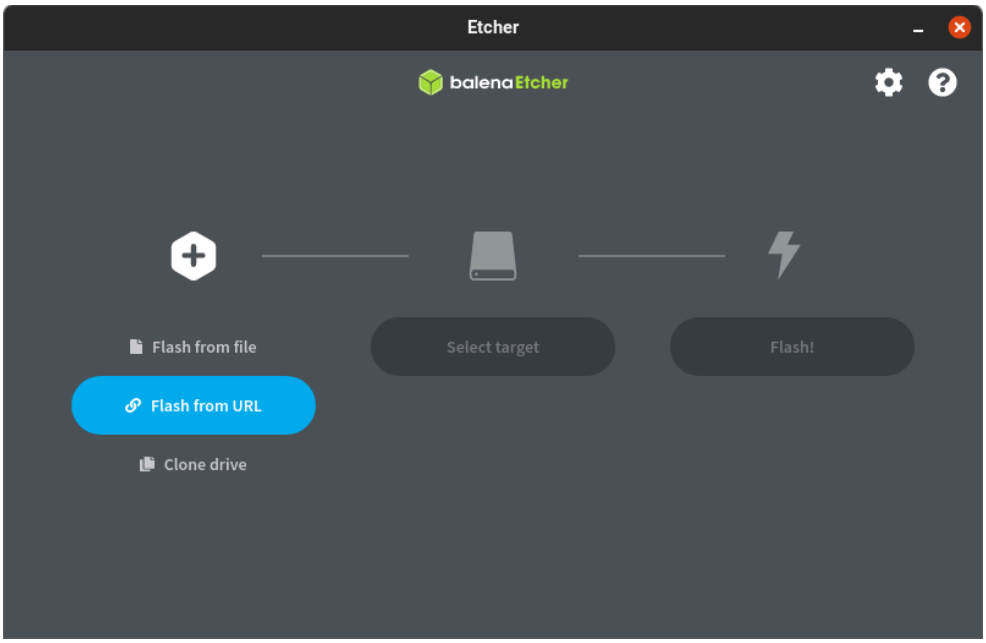
Raspberry pi setup: you will need to set up a Raspberry Pi in order to project. we will show you how to set up a RasPi 2 Model B as a workstation that will host the applications required to implement a variety of High Availability solutions. the RasPi 4 Model B used in this project.

What will we need

A. power supply: For power socket connection, all Raspberry Pi models have a USB port (the same is found on many cell phones)
B. microSD card: Raspberry Pi needs an SD card to store all its files and the Raspberry Pi OS operating system.
C. keyboard and a mouse: To start using the Raspberry Pi, we need a USB keyboard and a USB mouse.
D. TV or computer screen: To view a Raspberry Pi OS desktop environment, you need a monitor and cable to connect the display and your Raspberry Pi.
E. HDMI: Raspberry Pi has an HDMI output port that is compatible with the HDMI port of most modern TVs and computer monitors.

Set up SD card

- Attach the installation media (SD card) to your computer
- Download and start Balena Etcher
- Select “Flash from URL”

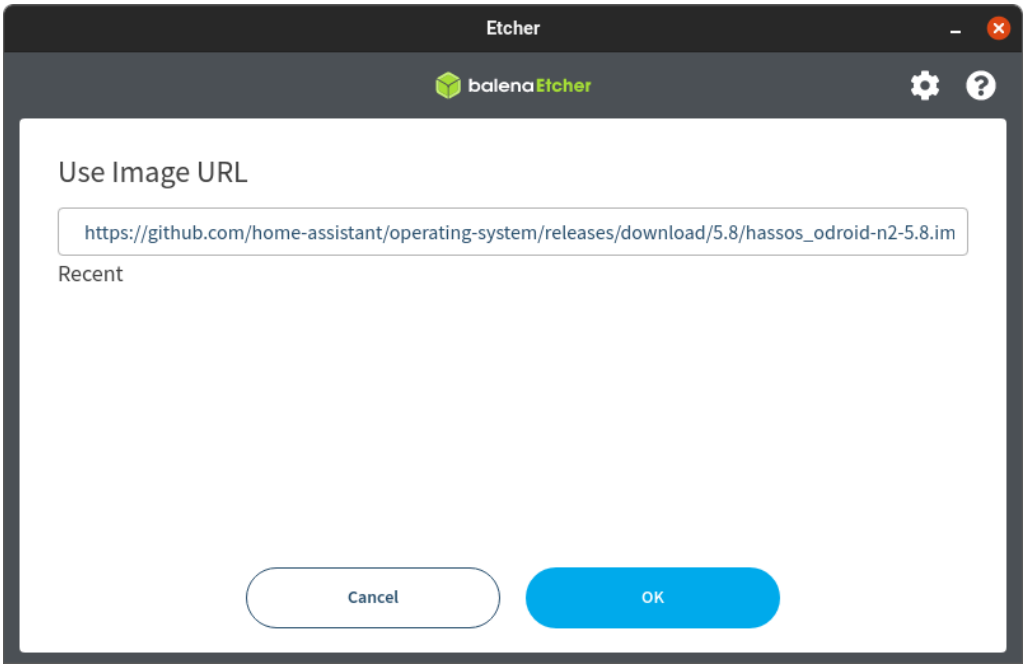


- Get the URL for your Raspberry Pi:

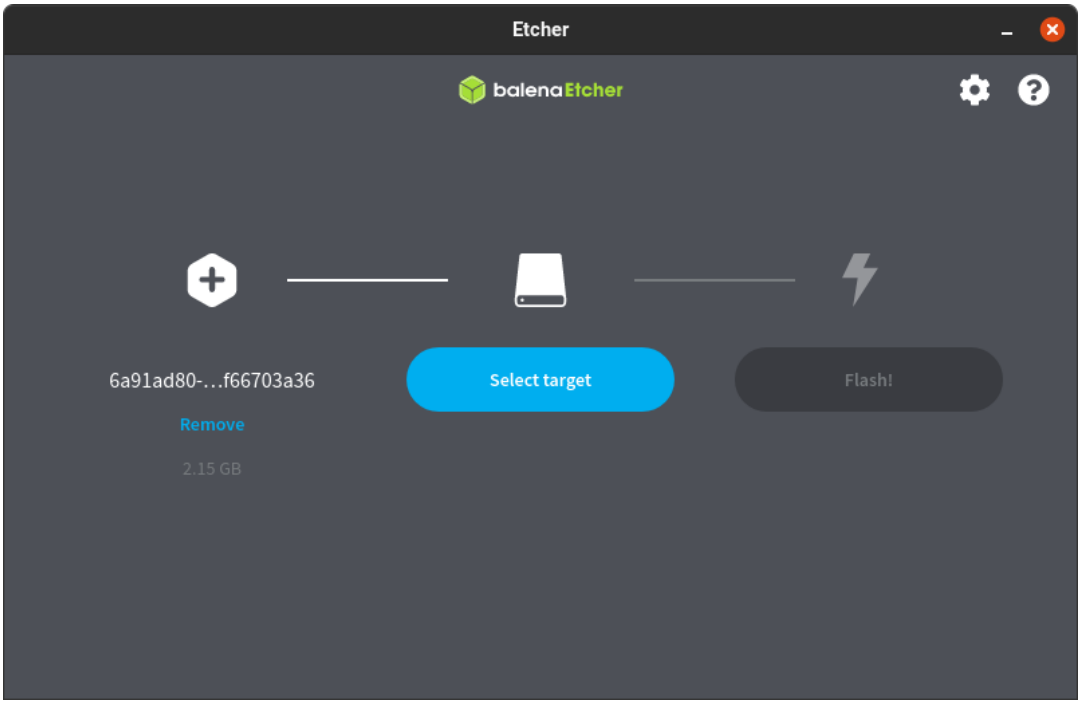
https://github.com/home-assistant/operating-system/releases/download/5.13/hassos_rpi3-5.13.img.xz

- Select and copy the URL

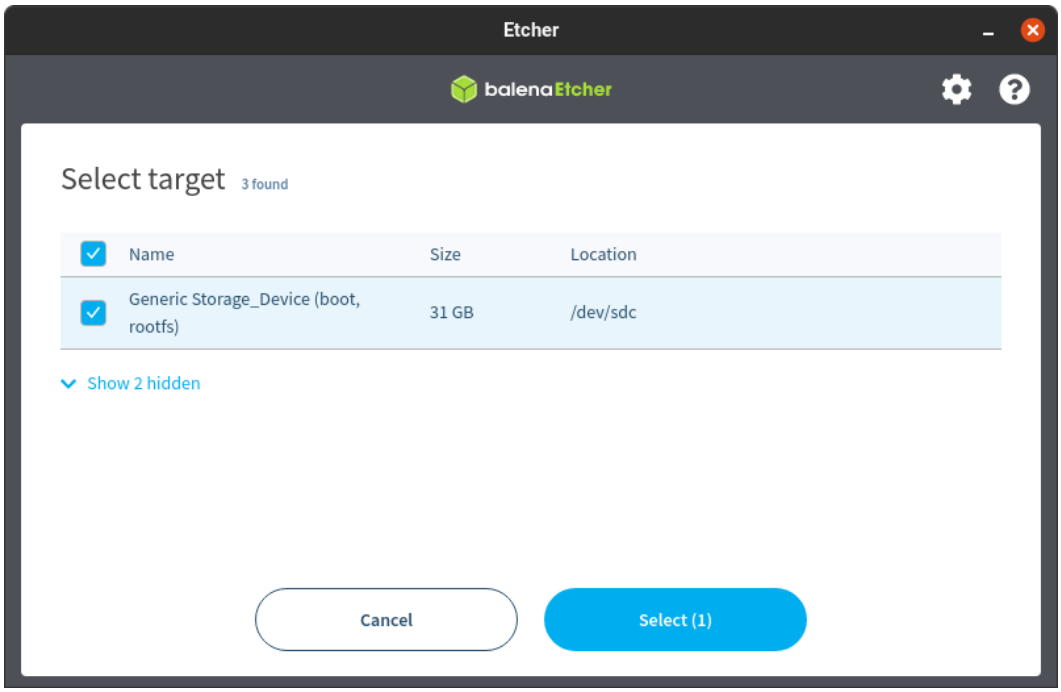
1. Paste the URL of the Raspberry Pi into Balena Etcher and click OK



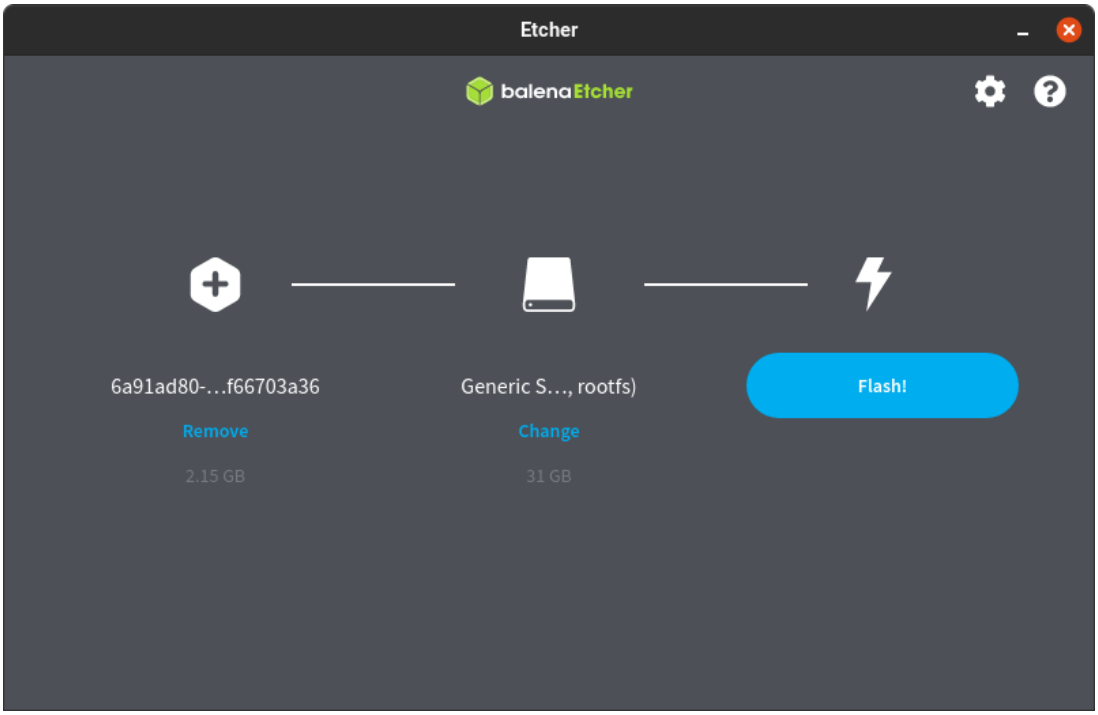
2. Balena Etcher will now download the image, when done, click 'Set target'



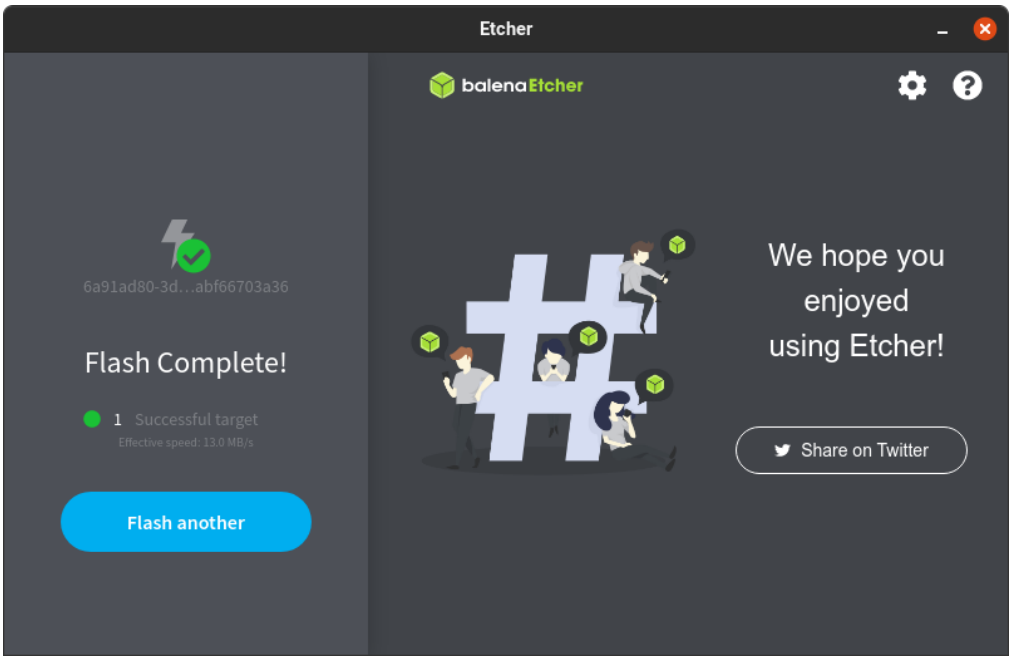
3. Select the SD Card you want to use for your Raspberry Pi



4. Click "Flash!" To start writing the image



5. When Balena Etcher finishes writing the photo, you'll get this confirmation



Start up your RASPBERRY PI

Insert the installation media (SD card) you just created, attach an ethernet cable for network and attach a cable for power.

Within a few minutes you will be able to reach Home Assistant on homeassistant.local:8123 or http://X.X.X.X:8123 (replace X.X.X.X with your Raspberry Pi's IP address).

II. Get Started

After you have installed Home Assistant, now is the time to configure it. We create the owner account for the Home Assistant. Enter the required and click "Create Account". As shown in the Figure 4_ 6. Then, you can enter a name for your home and set your location and unit system. Click "Detect" to find your location and set the time zone and unit system based on that location.

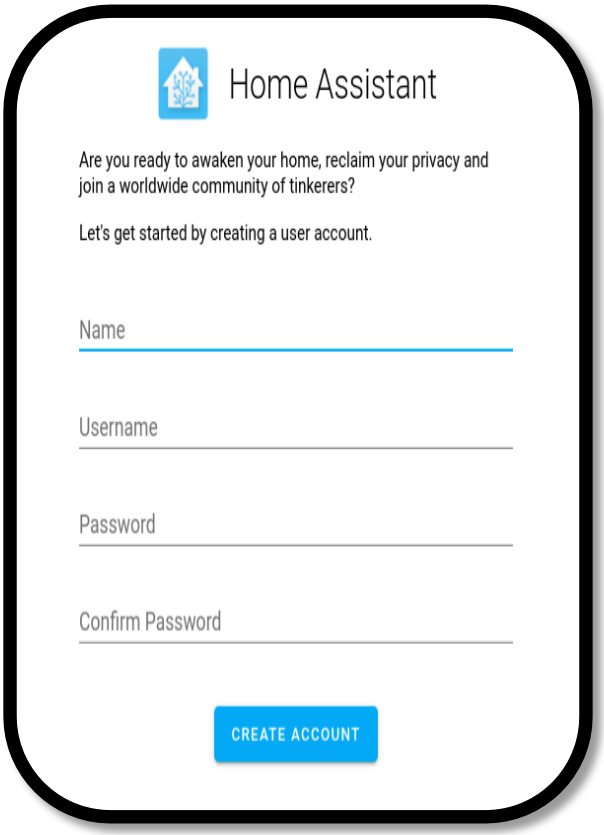


Fig 2. 16: Set your username and password.

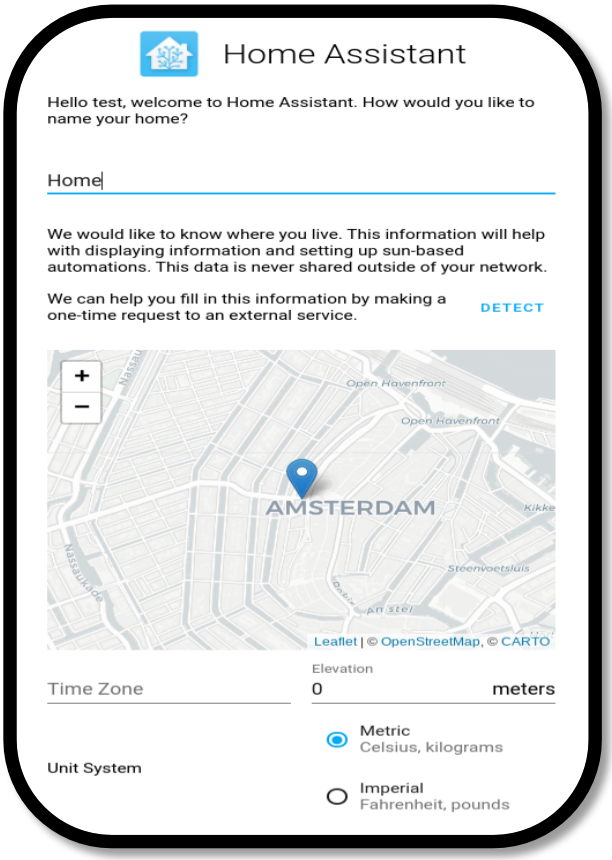


Fig 2. 17: Set your location, time zone, and unit system.

click Next.
In this screen, Home Assistant will show any devices that it has discovered on your network. In the absence of all devices, you can manually add devices later. Finally, click Finish.
Now you are taken to the Home Assistant web interface. This screen will display all of your devices.

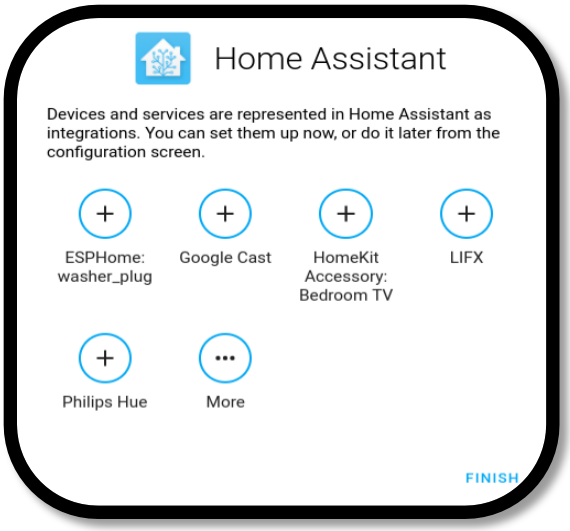


Fig 2. 18: Discovery of devices on your network.

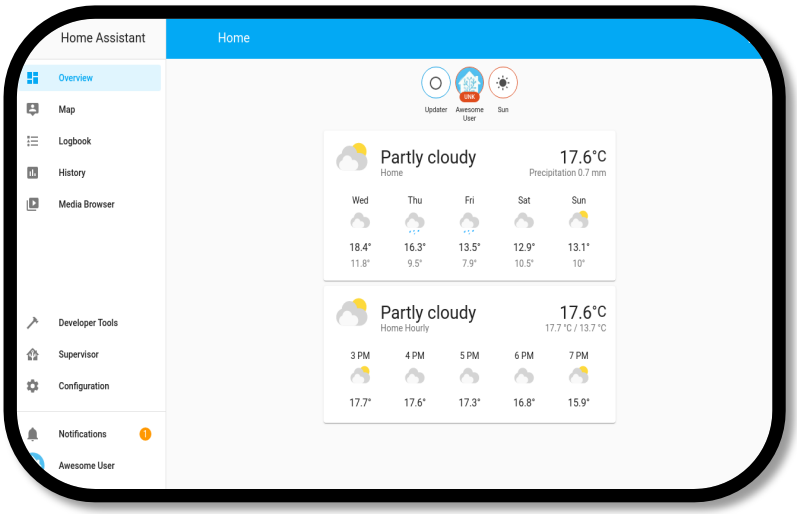


Fig 2. 19: The Home Assistant user interface.

Click on Configuration.

On the next screen, the integrations page in the configurations panel shows you all your configured integrations.

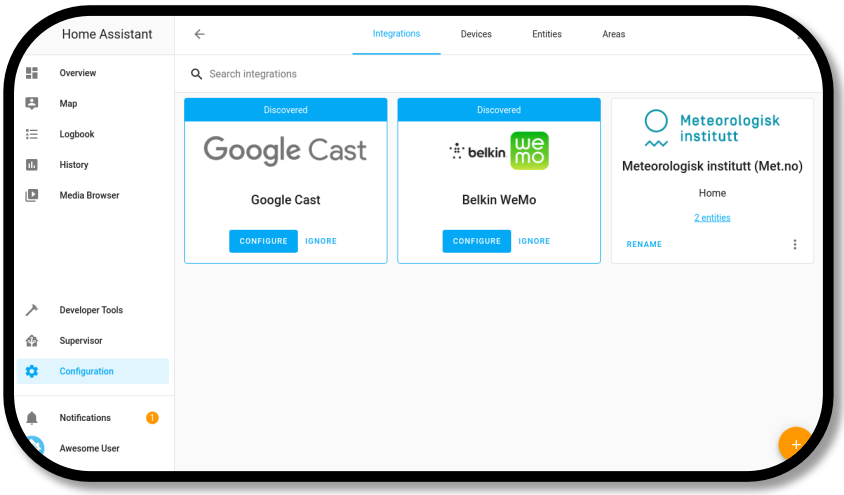


Fig 2. 20: The integrations page.

When you're done, go back to the web interface and voila, your devices will be ready to be controlled.

III. Automation

In the user interface, click Configuration, then click Automation. You will now see the automation screen from which you can manage all the automations in Home Assistant. Click the orange button at the bottom right to create a new automation.

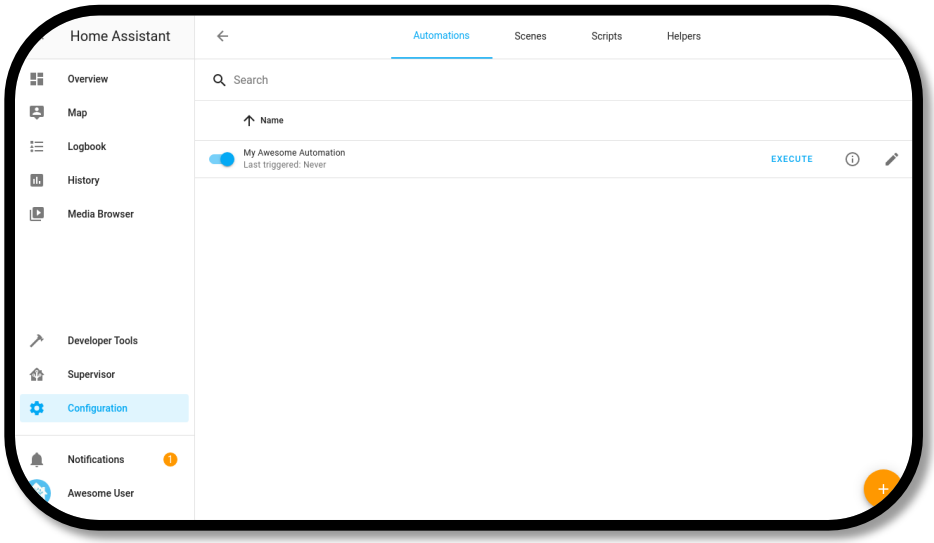


Fig 2. 21: The automation editor.

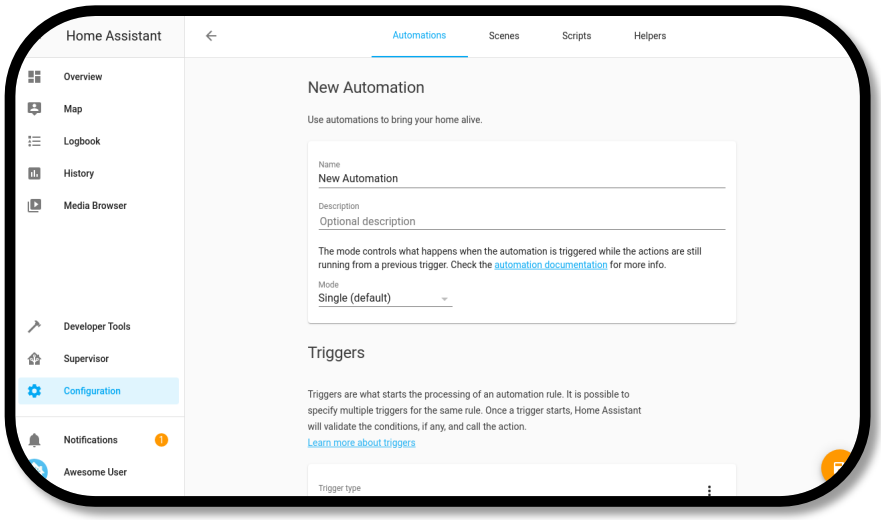


Fig 2. 22: The start of a new automation.

The first thing we will do is set a name. Enter “Turn Lights on at Sunset”. The second step is to decide what to run our automation. In this case, we want to use the sunset event to start our automation. However, if we turn on the lights when the sun has already set, it will be too late because it is already dark while it is setting. So, we'll add offset.

In the Player section, click on the dropdown menu and change the Player type to "Sun". We are allowed to choose sunrise or sunset, so go ahead and choose sunset. As we've discussed, we want the automation to run a little before sunset, so let's add -00:30 as an offset. This indicates that the automation will be triggered 30 minutes before sunset actually. salary!

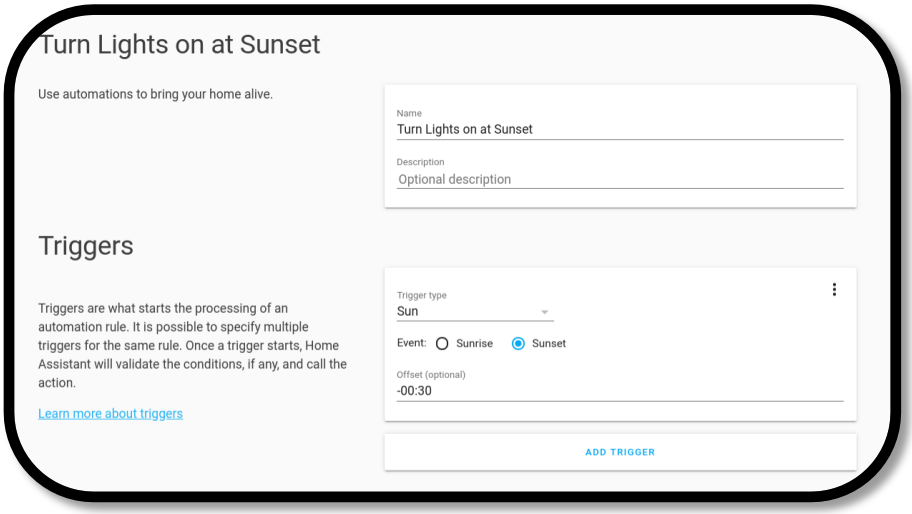


Fig 2. 23: A new automation with a sun trigger filled in.

Once we have defined our trigger, scroll down to the action section. Make sure the action type is set to "Call service," and change the service to light.turn_on. For this automation we're going to turn on all lights, so let's change the service data to: entity_id: all

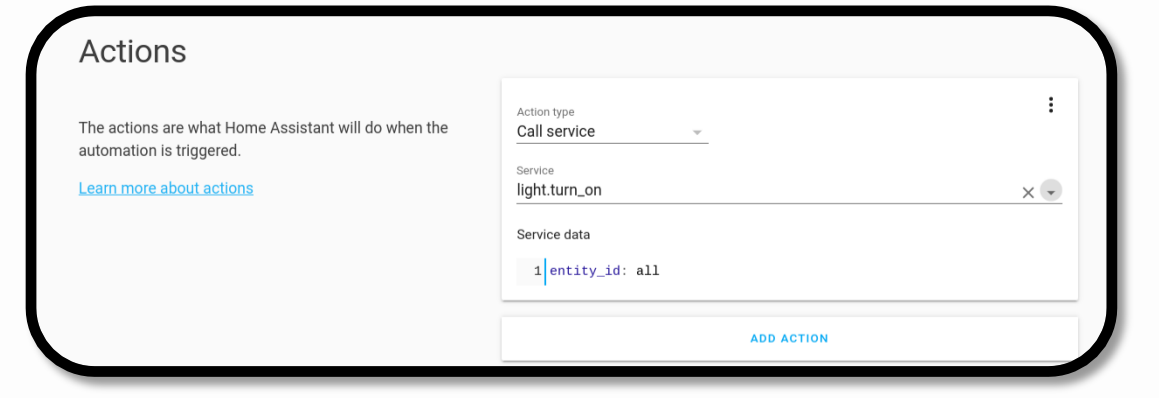


Fig 2. 24: A new automation with the action set up to turn on the lights.

Click the orange button to save the automation and see your automation.

IV. Advanced Configuration

The setup process takes care of the initial setup of the Home Assistant, such as naming your house and choosing your location. After the initial setup, these options can be changed in the user interface by clicking Configure in the sidebar and clicking General, or by manually editing them in the Home Assistant configuration file called config.yaml.

From the Supervisor main panel, open the add-on store.

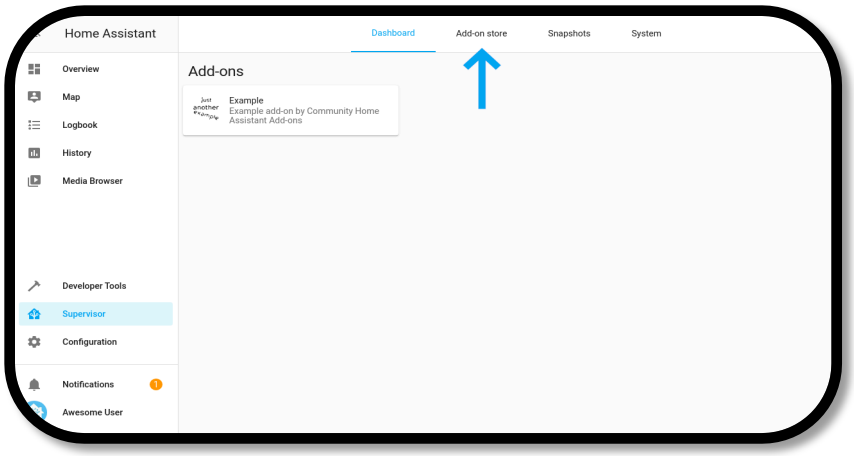


Fig 2. 25: From the Supervisor main panel, open the add-on store.

Under the "Official add-ons" section you will find the File editor add-on.

- Click on File Editor and click on INSTALL. When installation is complete, the UI will go to the add-on details page for the file editor.
- Now start the add-on by clicking on START.
- Open the user interface by clicking on OPEN WEB UI.

Now let's make a change using the file editor: we are going to change the name, location, unit system, and time zone of your Home Assistant installation.

- Click the folder icon in the top left of the file editor window to open the file browser sidebar.
- Click the configuration.yaml file (in the /config/ folder) to load it into the main file editor window.
- Add the following to this file (preferably at the very top, but it ultimately doesn't matter):

```
home assistant:
name: Home Azoz
latitude: xx. xxxx
longitude: xx. xxxx
unit system: imperial
time zone: Gaza/Deir El_Balah
```

- Click the save icon to commit changes.
- Screenshot of the "General" page in the configuration panel.
- Now Restart Home Assistant using the “RESTART” button in the Server management section on the same page.

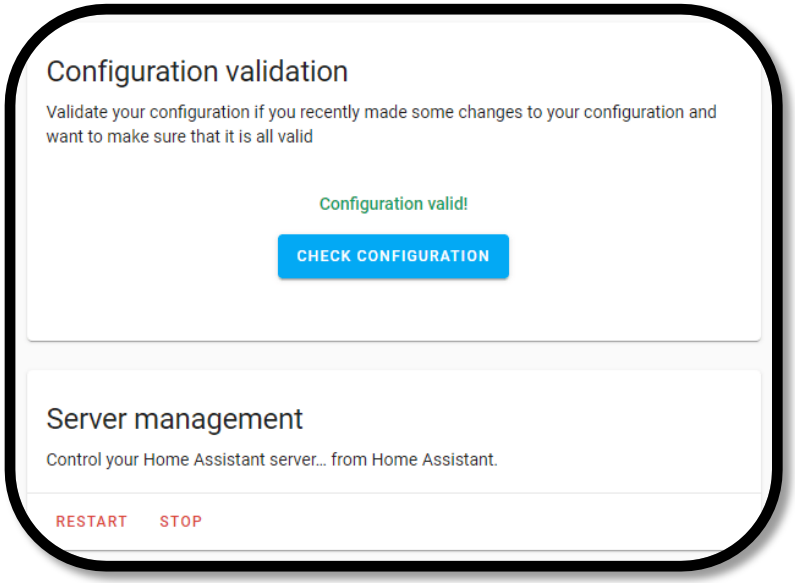


Fig 2. 26: Screenshot of the "General" page in the configuration panel.

2.5.4. Google Assistant

Google Assistant integration allows you to control things via Google Assistant on your mobile phone, tablet, or Google Home device. Google Assistant integration requires a bit more setup than most due to the way Google requires you to set up Assistant apps.

1. Configuration of the google cloud platform
2. Create a new project in the Actions on Google console.

- Click New Project and give your project a name.
- Click on the Smart Home, then click the Start Building.
- Click on the Overview tab
- Click Build your Action, then click Add Action(s).
- Add your Home Assistant URL: https:// 192.168.1.117/8123.
- Click Save.

3. Account linking is required for your app to interact with Home Assistant.

- Start by going back to the Overview tab.
- Click on Setup account linking.
- If asked, leave options as they default No, I only want to allow account creation on my website and select Next.
- Then if asked, for the Linking type select OAuth and Authorization Code. Click Next
- Enter the following in Figure:

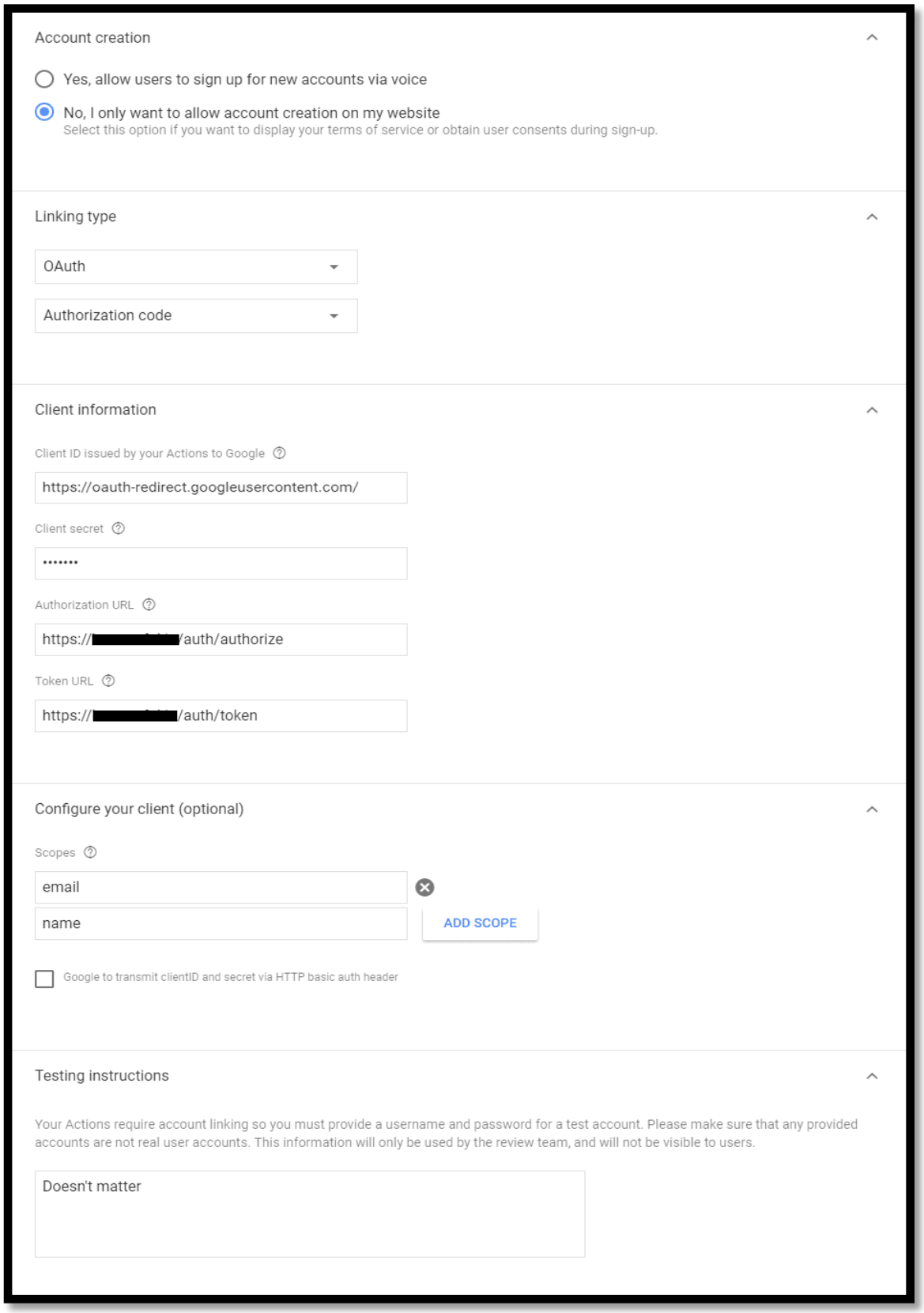


Fig 2. 27: Account linking

- Click Next, then click Save
- Select the Develop tab, then select the Test button to generate the draft version Test App.
- Add the google_assistant integration configuration to your configuration.yaml file and restart Home Assistant following the configuration guide.

3.1. Available domains

The following domains are available for use with the Google Assistant, listed with their default types :

Table 1: list of available domains with Google Assistant

alarm_control_panel	arm/disarm
camera	streaming, requires compatible camera
group	on/off
input_boolean	on/off
input_select	option/setting/mode/value
scene	on
script	on
switch	on/off
fan	on/off/speed
light	on/off/brightness/rgb color/color temp
lock	on/off/set position
cover	
media_player	on/off/set volume (via set volume)/source (via set input source)/control playback
climate	temperature setting, hvac_mode
vacuum	dock/start/stop/pause
sensor	temperature setting for temperature sensors and humidity setting for humidity sensors

2.5.5. Google Home

Share home and devices in Google Home Application, to organize and manage your devices in the Google Home app, you can create a home. Then, you can invite the household members who will be involved in controlling the house and its appliances.
For example, you can place sensors in the living room or kitchen, bedroom lights, and doors in the house, and invite your family to be members of that house.

2.5.5.1. Manage home members

To let other people, like family members or roommates, use the devices in your home, add them as home members. Only invite people you trust to become home members.

- **Invite members to a home**
  - Open the Google Home app
  - At the top, tap Add + , then click Invite home member , then click Invite person.
  - Enter the email address of the person you want to invite to your home and then tap Next.
  - See what's shared when you add someone to your home and click Next.
  - Review the person's access to your home, and then click Send.
  - The person you invite will receive an email with instructions.

When accepting the request, the member can share your home information

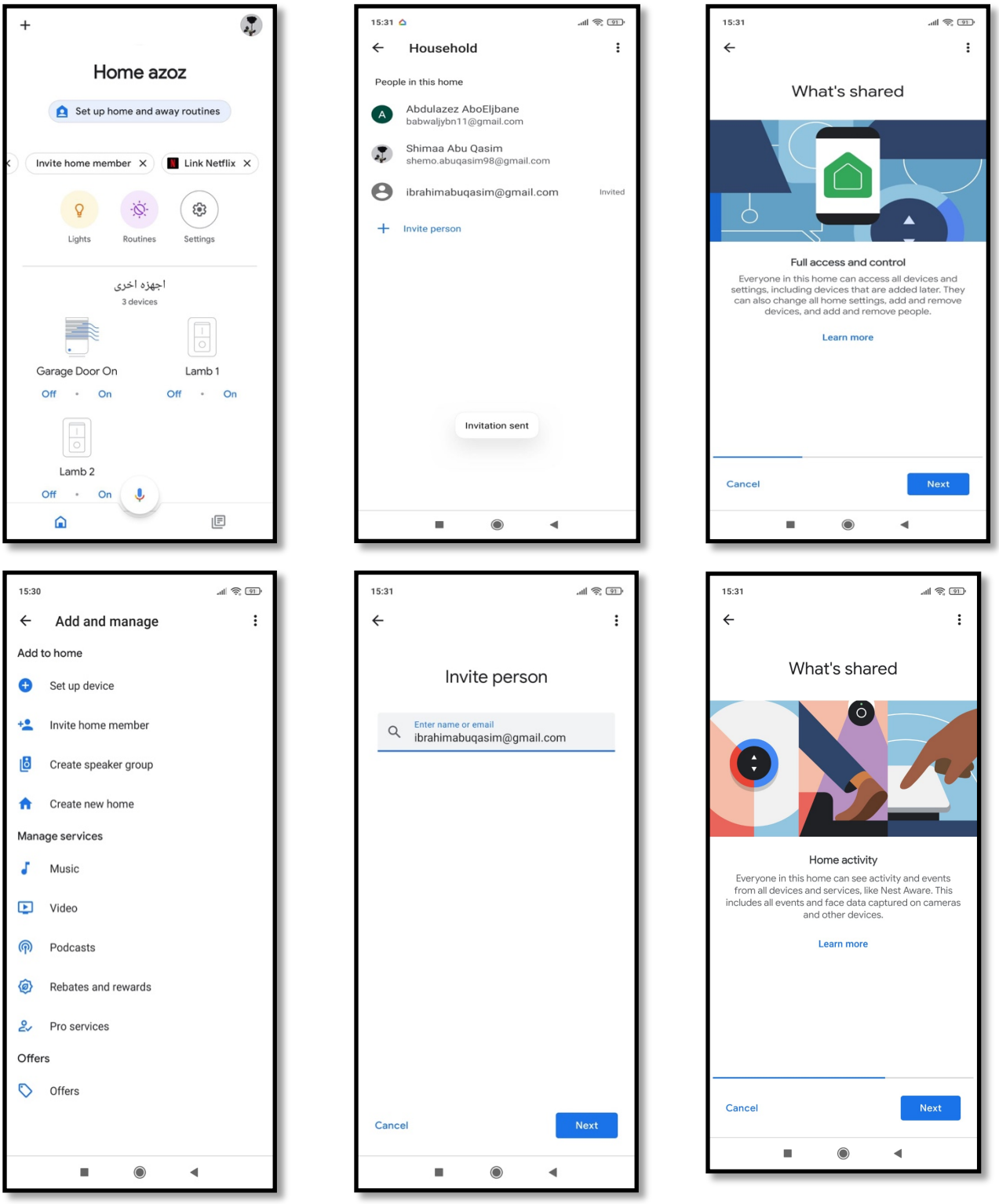


Fig 2. 28: Invite members to a home.

2.5.5.2. Create and manage rooms

- **Create a new room**
  - Open the Google Home app
  - At the top, select the home you want to add a room to.
  - At the top right, tap your account.
  - Tap Assistant settings, then Home control.
  - Tap any of your devices, then Room.
  - Scroll and tap Add new room, enter a room name, final Ok.

Note: The device you choose from its current room will be removed and added to a new one. You can move the device back to the first room after creating a new room.

- **Rename a room**
  - Open the Google Home app
  - At the top, select the home with the room you want to rename.
  - Tap the name of the room you want to change.
  - Tap Settings, Name, enter a name, then Save.
- **Delete a room**
  - Open the Google Home app
  - At the top, select the home with the room you want to delete.
  - Tap the name of the room you want to delete.
  - Tap Settings, delete room, Remove.

Note: Any devices in the room will be removed from the room. You can add them to a different room.

2.5.5.3. Assign devices to a room

To control your devices more easily, assign them to a room in a home. You can add each device to one room only.

- **Add the device to a room in your current home**
  - Open the Google Home app
  - Tap the device you want to add to a room, then Settings, Room, choose a room, then Save.
- **Change a device's room**
  - Open the Google Home app
  - Tap the device you want to device, then Settings, Room, choose a room, then Next.
- **Add the device to a new room in your current home**
  - Open the Google Home app
  - Tap the device you want to add to a room, then Settings, Room, scroll to the bottom and tap Add a custom room, enter the room name, then Save.

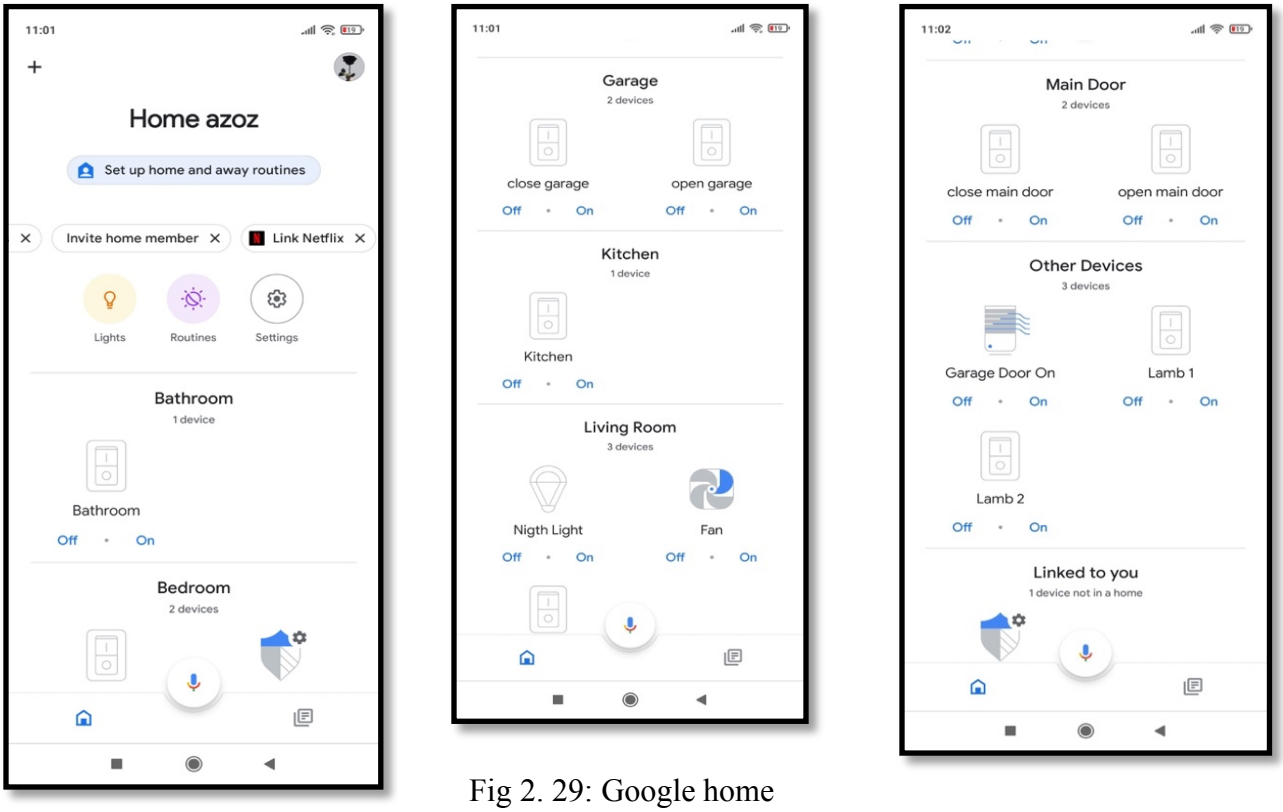


Fig 2. 29: Google home interface.

Chapter three

SYSTEM ANALYSIS

3. SYSTEM ANALYSIS

3.1. Software Development Life Cycle

Software development lifecycle process is the kind of structure or framework used in the development of any software product. There are many different models' specific lifecycles such as Waterfall model, spiral model and prototyping model. Each model is described through a series of activities. The development steps or the activities may vary in each model, but all the models include planning, requirement, analysis, design etc.

Although there are many different methods used in the field of software development, we consider the waterfall model. The waterfall model is the classical model of software engineering. This model is one of the oldest models and is widely used in government projects. This model includes requirement gathering, planning, requirement's & design, development, testing and deployment.

Test Automation is applied with the structural sequence of testing process, that can have all the stages of testing life cycle with additional phases for including the test preparation and automation implementation processes. This entire process is known as Automation Testing Life Cycle.

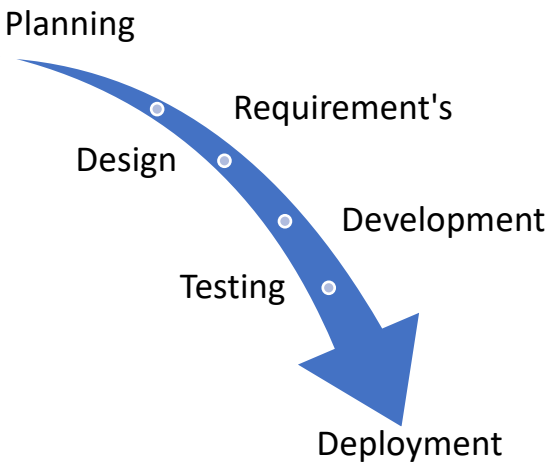


Fig 3. 1: Waterfall model diagrams.

1. **Planning**

   Planning is the process of thinking about the activities required to achieve a desired goal. It is the first and foremost activity to achieve desired results. We have determined the devices and motors that we will control and the time required to implement the project.

2. **Requirement's**

   In this step we list all the requirements including the material needs and all the functions necessary to design our system. The necessary parts and the material cost, and determining the available ones and the best ones.

3. **Design**

   At this stage, prototype is designed (in papers) for the screens, and menus which will be inside the application. We designed the control circuits.

4. **Development**

   Here the application programming process starts with an appropriate software environment which helps in building the application. Designing the user interface and adding devices and sensors.

5. **Testing**

   Below is a beta version of the application made for testing and implementation on the selected model or house

6. **Deployment**

   The trial version is published after ensuring that it works correctly and properly.

3.2. Task Management

Project management is a methodical approach to planning and guiding project processes from start to finish. Project management can be applied to almost any type of project and is widely used to control the complex processes of software development projects. Gantt chart of the phases of the project and their corresponding estimated time.

Table 2: Estimating time to phases of the project.

Task Name	Start	End	Duration (days)
Project Planning	25/10/20	28/11/20	33
Project Analysis	28/11/20	20/12/20	22
Project Design and Simulation	20/12/20	30/02/21	70
Project implementation	28/02/21	20/05/21	80
Project Testing and Result	01/05/21	20/05/21	20
Documentation	20/05/21	25/06/21	35

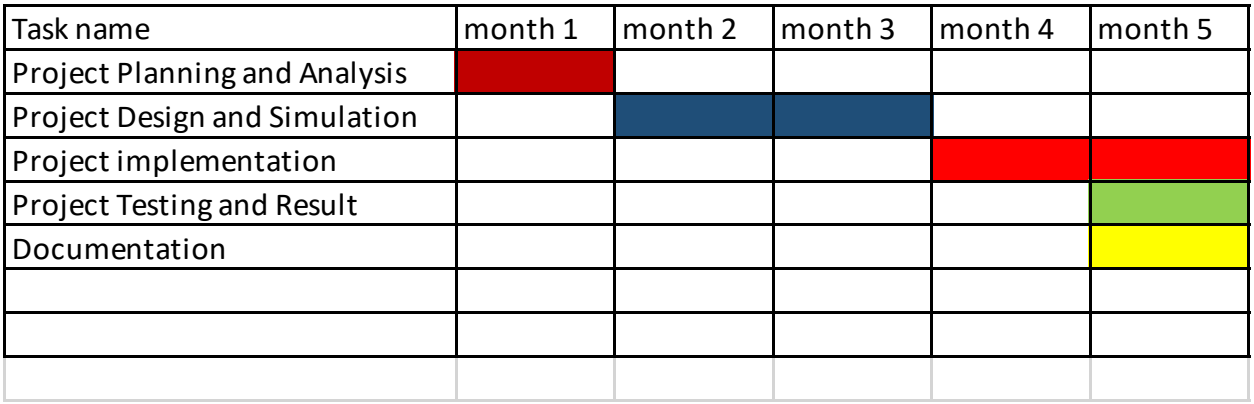


Fig 3. 2: Gantt chart to phase of project.

Chapter four

RESULT AND DISCUSSION

4. RESULT AND DISCUSSION

4.1. Overview

The steps required for the family member to monitor, the controller devices are as following, first he has to login using IP and password, if the user enters the password correctly, the application records the available devices and their status and displays the control keys for lighting, fans and other devices if any are available and the values of the sensors. In this thesis, we suggest the following main steps to implement the system as shown in **Fig 4. 1**.

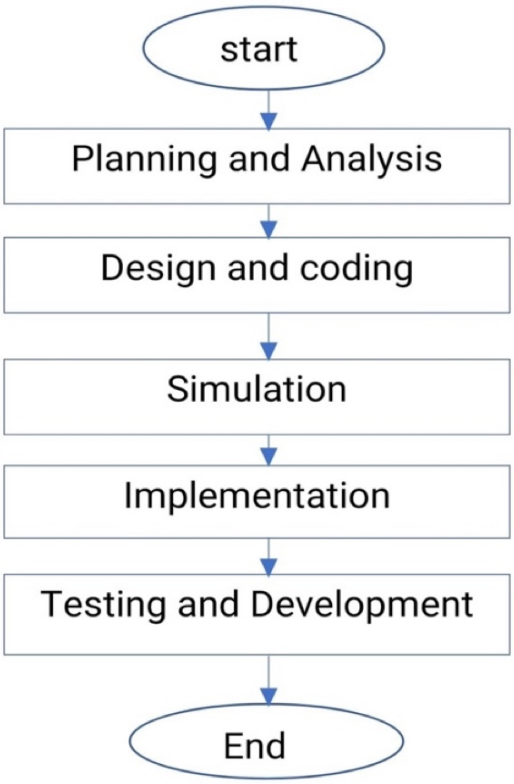


Fig 4. 1: Main steps of implementation the project.

4.2. Graphical User Interface (Mobile Application)

The official app for Home Assistant. Home Assistant for Android lets you control all your devices in your Home Assistant instance.
Home Assistant for Android requires that you have access to a Home Assistant instance. First, download the APP and log into the account using your IP.

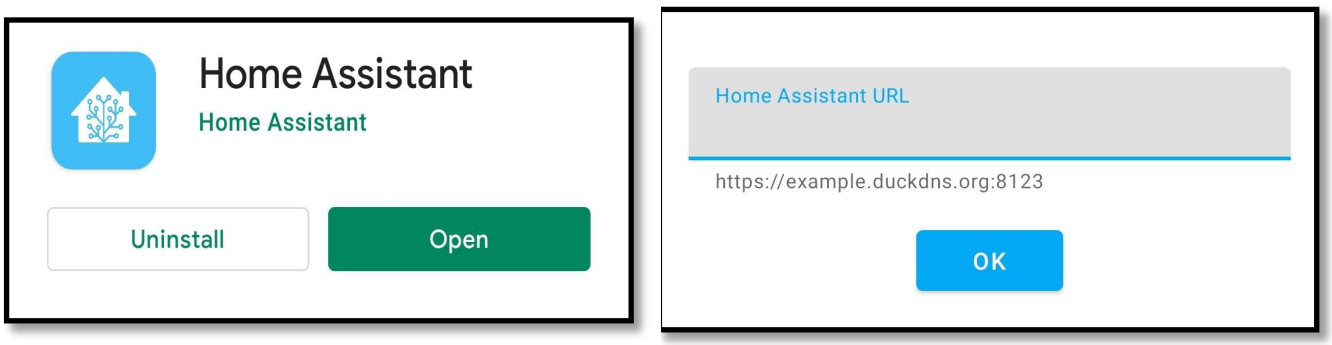


Fig 4. 2: Download Home Assistant from the store and enter your IP.

If your entry is correct, you will go directly to the main screen and the home interface, which consists of three sections: the first section is the lighting control section, electrical appliances and fans, as shown in the **Fig 4. 3.** the second section is the door control section, the main door and the garage door, shown in **Fig 4. 4**. the last section is the sensors that displays the temperature and humidity in addition to motion and gas sensors. As shown in **Fig 4. 5**.
Home Assistant enables us to know the status of the sensors and the history of the sensor through a sub-interface, when you click on the sensor, as shown in **Fig 4. 6** and **Fig 4. 7**.
In the main interface, when we click on a map, at the top screen we can move to another interface that contains a map of the home, through which we can dispense with the switch just by click on any icon in the home, as shown in **Fig 4. 8.**

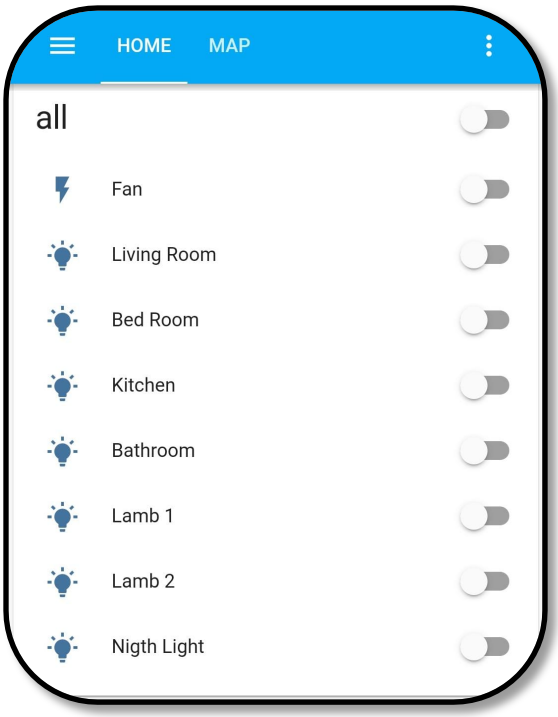


Fig 4. 3: Lighting and fan control interface.

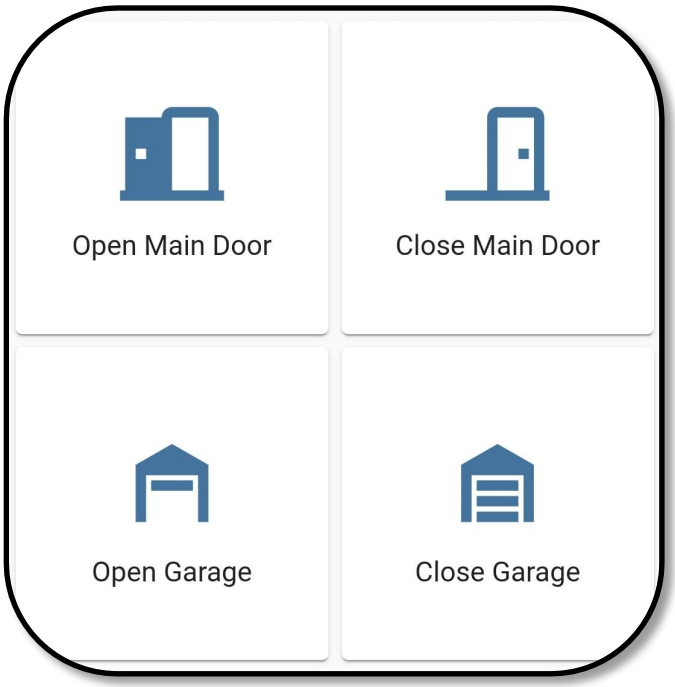


Fig 4. 4: door control interface.

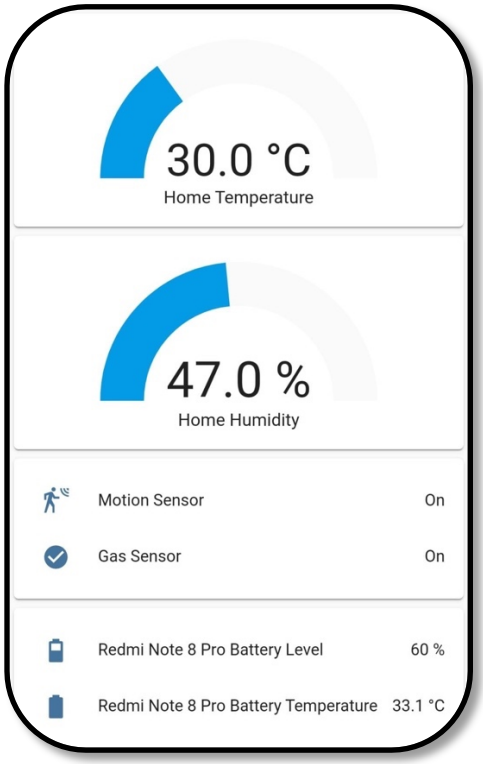


Fig 4. 5: Sensors interface.

Fig 4. 6: MQ-5 office.

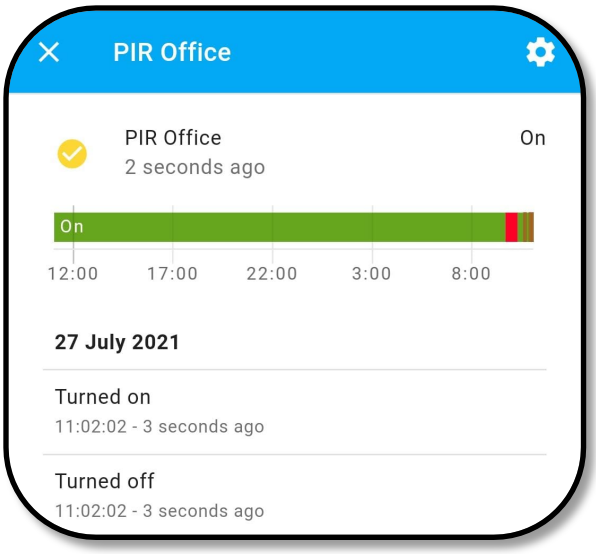


Fig 4. 7: PIR office.

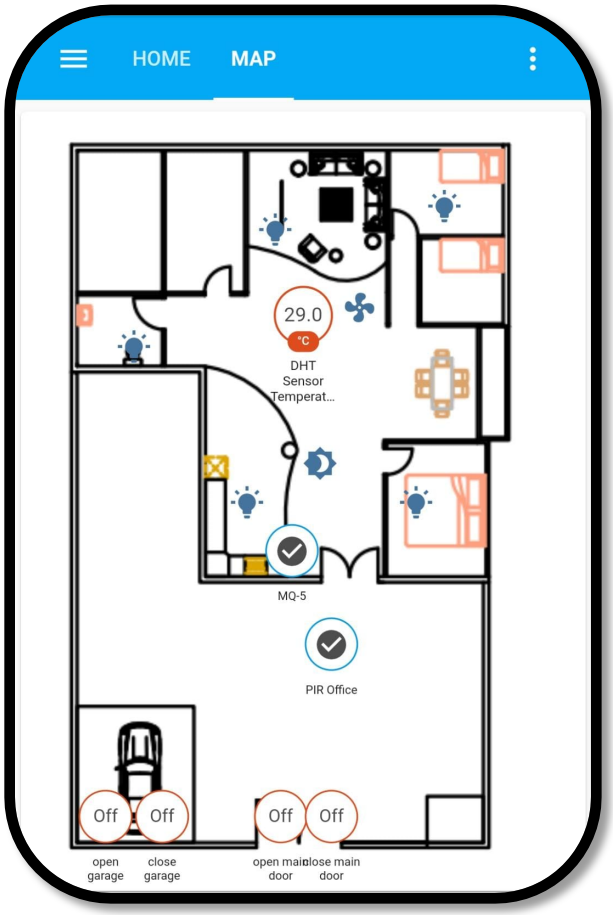


Fig 4. 8: Home Map interface.

4.3. Graphical User Interface (Website)

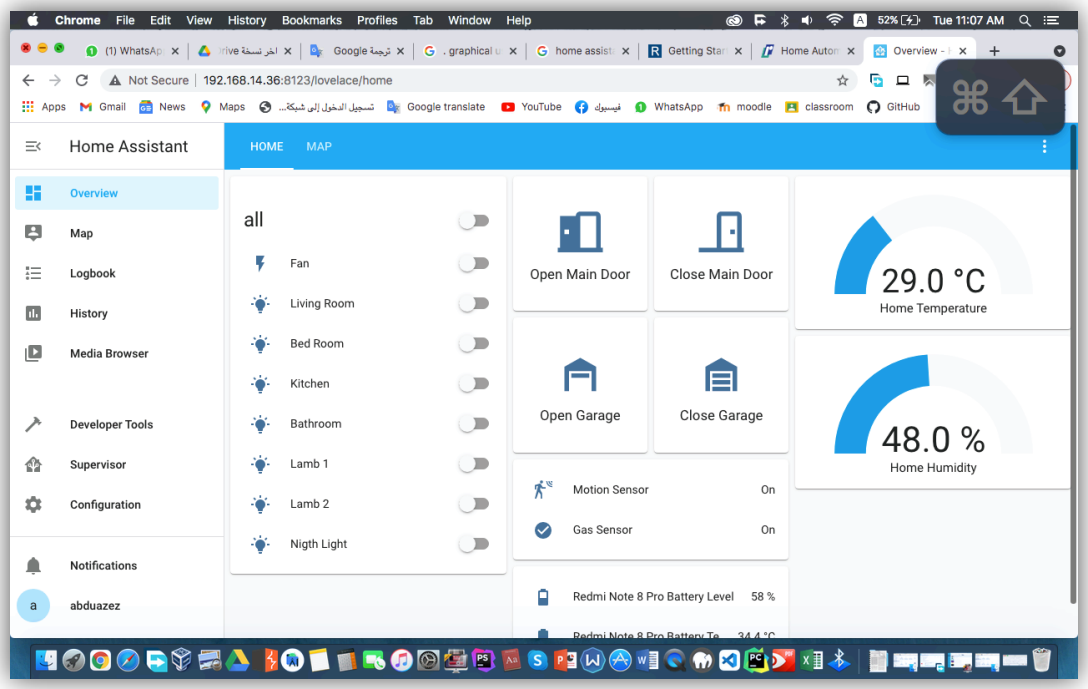


Fig 4. 9: control web interface.

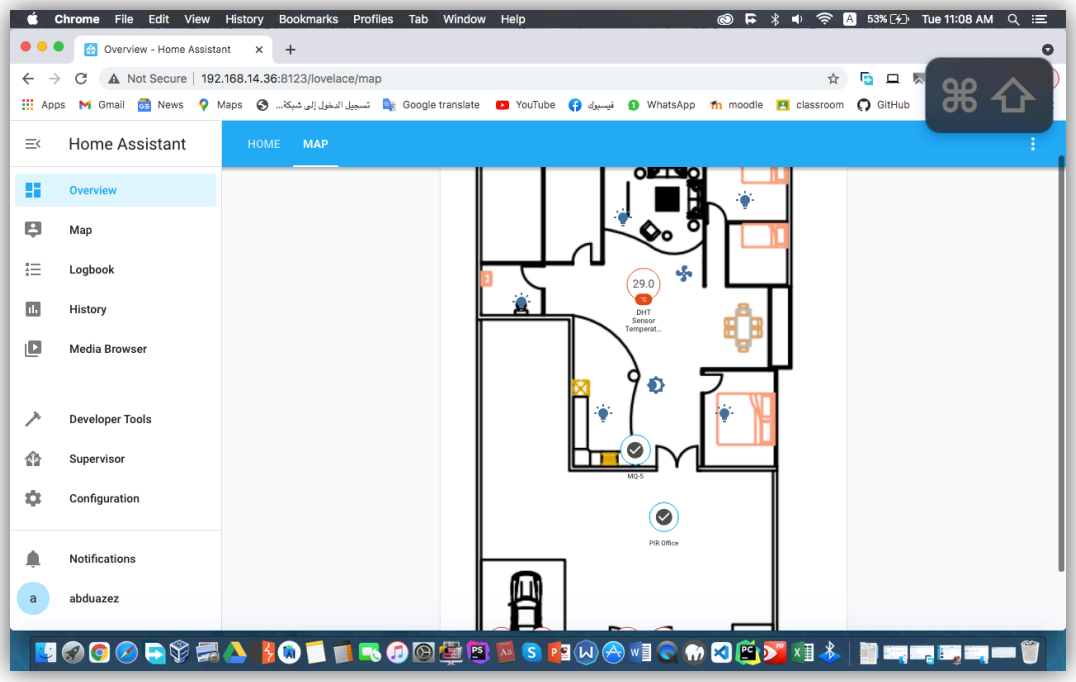


Fig 4. 10: Map of the home web interface.

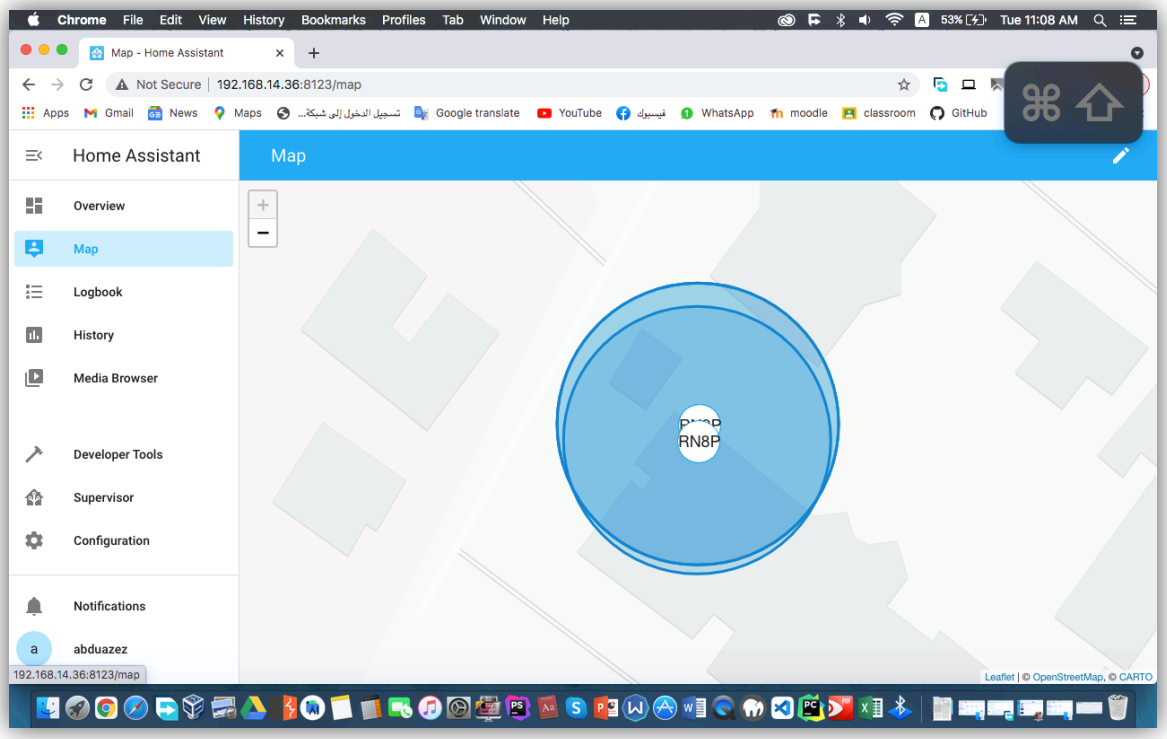


Fig 4. 11: Map of Member location web interface.

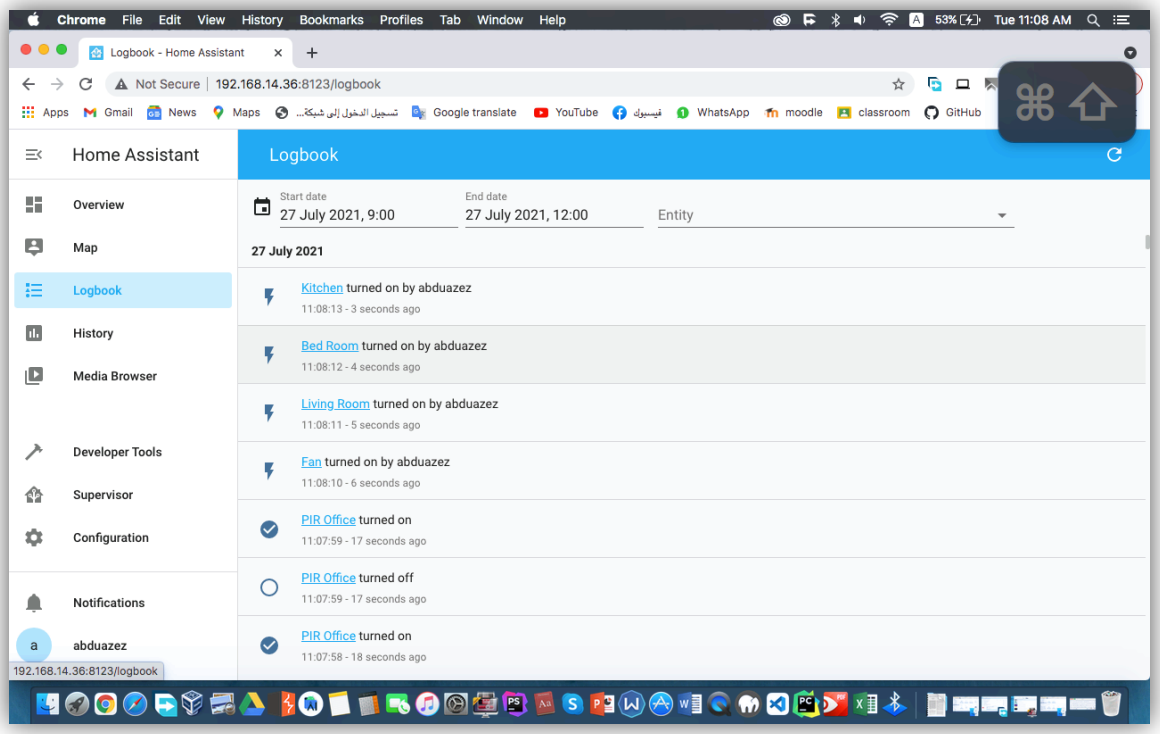


Fig 4. 12: Logbook web interface.

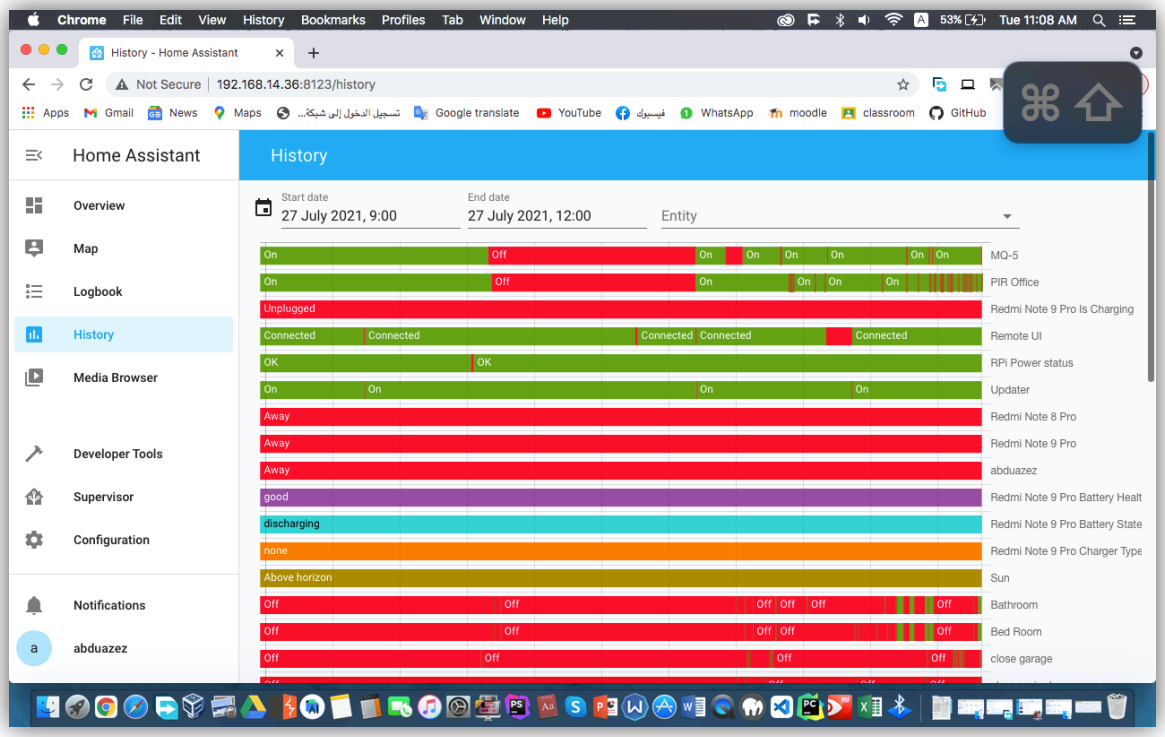


Fig 4. 13: History web interface.

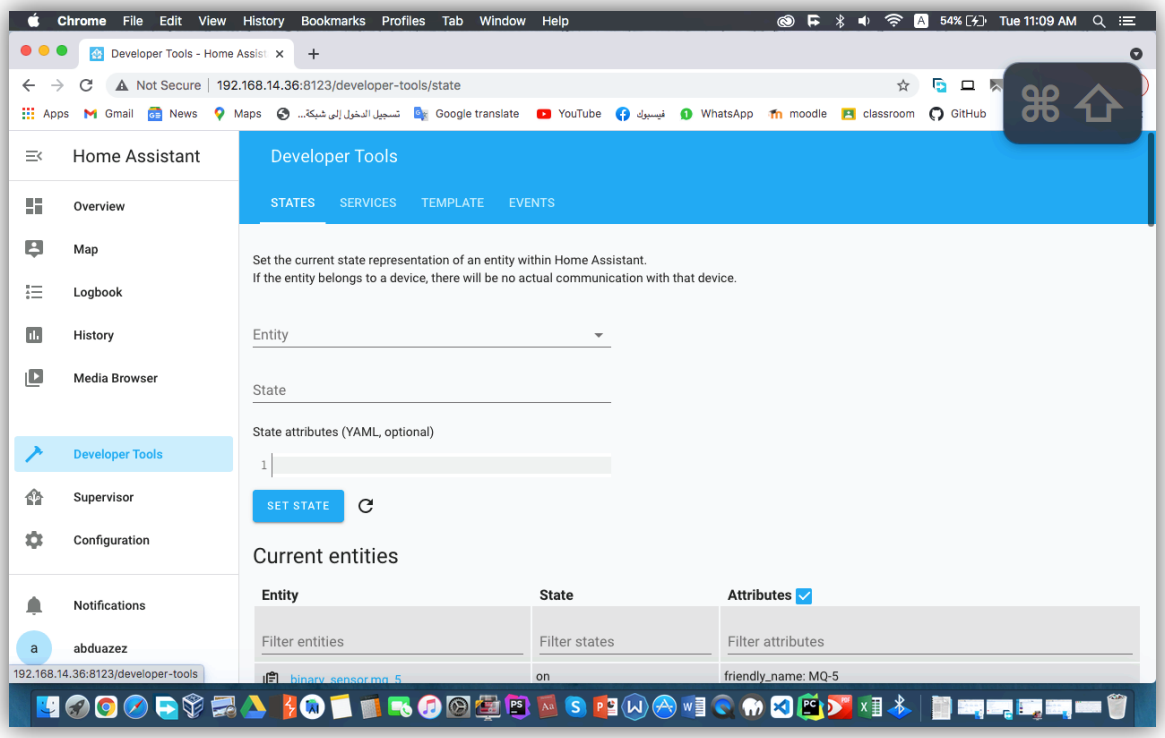


Fig 4. 14: Developer tools web interface.

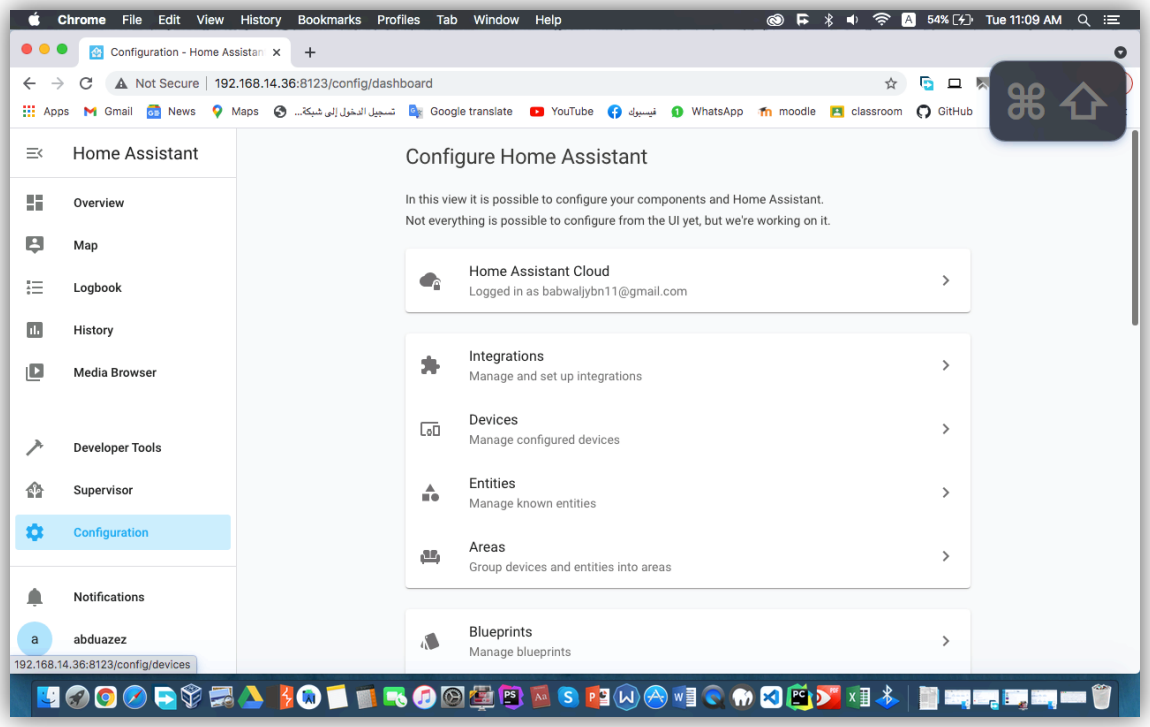


Fig 4. 15: Configuration web interface.

4.4. Testing and Results

We have brought all the basic parts that represent our system, a home simulator on which the circuit and electricity connection will be applied, the computer with the website and the user supervising the system to follow the success of the project. Where we run the application, enter the IP and password, connect the pieces and wires, then control and give commands from the phone. The application shows the temperature and humidity of the house and we control the doors, lighting and electrical appliances.

Thus, the final form of the house is as shown in the figure below and injects us with the success of the system and the tasks required of it as expected of it, and thus it has achieved the goals and specific objectives of our project.

Fig 4. 16: Smart home final form.

Chapter Five

CONCLUSION AND FUTUREWORK

5. CONCLUSION AND FUTUREWORK

5.1. Conclusion

This project, proposed a solution to control electrical appliances, lights, control doors, read house temperature and humidity, and detect fire to protect appliances and tools, save human energy and electric energy, and protect the house from theft.

Our system helps people track and monitor their homes from outside using GUI without the cost of going home, triggering fire alarms when gas leaks or there are fires.

This project was designed system that communication with smart phone via RPI to control and monitor home Appliances.

This system helps people track and monitor their homes using from outside.

Manual control has been dispensed with the android app, website replacement and modifications made to it from sandbox restrictions that limit who can interact with the system.

We can control the devices via touch commands by opening the application and giving the command, or by voice commands, as this is one of the features.

Sure, a lot of improvements can be made after getting feedback from users.

5.2. Future Work

Currently, our system has been implemented as a trial version with lighting, fans, door control and some sensors, but in the near future and since the system has achieved a certain success, we will apply the system to

- Monitoring the power and energy.
- video call.
- increasing system safety.
- using HD camera with cloud storage upload picture to cloud.
- sensors "reading analog data ".
- Using surface mount components SMD.
- communication protocol protection from hacking so implement a security layers in communication protocol.

we want His later achievement is to make the system ready for sale in specialized stores, considering the appropriate price for the system.

Our project has covered the application of the Raspberry Pi in home automation and how we can leverage the Raspberry Pi's abilities. As newer and more powerful versions are released, we believe the future for this technology is, indeed, very bright.

This idea opens the door to future application in hotels and small businesses, as it enables them to control and monitor their companies. Employees can also feel less worried if they forget devices or lights on, as the application can control them.

REFERENCES

[1] **Smart GSM based Home Automation System,** Rozita Teymourzadeh; Salah Addin Ahmed; Kok Wai Chan; Mok Vee Hoong.

[2] Smart home automation system using Bluetooth technology, Muhammad Asadullah, Khalil Ullah.

[3] Intelligent Smart Home Automation and Security System Using Arduino and Wi-fi, Chandramohan, R.Nagarajan, K.Satheesh Kumar, AND.Ajith Kumar, P.A.Gopinath, S.Ranjith Kumar.

[4] **Raspberry PI and Wi-Fi Based Home Automation System**, Kolhe Ujvala S.

[5] **Raspberry Pi as Internet of Things hardware: Performances and Constraints** Mirjana Maksimović, Vladimir Vujović, Nikola Davidović, Vladimir Milošević and Branko Perišić

[6] **Home Automation with Raspberry pi**, Donald Norris.

[7] Home Assistant, home page https://www.home-assistant.io/

[8] **Raspberry Pi home page** https://projects.raspberrypi.org/en

[9] **IoT Based Home Automation Using Raspberry Pi,** Rajalingam Balakrishnan, K. Venkatesh, P. Rajkumar, S. Hemaswathi.

[10] **IOT Based Web Controlled Home Automation Using Raspberry PI**, G. Joga Rao, A. Vinod, N. Priyanka, Ch. Siva Hari Kumar. K

[11] Home Automation Systems - A Study, Satish Palaniappan, Naveen Hariharan, Naren T Kesh, Vidhyalakshimi S, Angel Deborah S.

[12] **Internet of Things: Smart Home Automation System using Raspberry Pi,** Smita Mahindrakar, Ravi K. Biradar.

[13] **A Smart Home Automation technique with Raspberry Pi using IoT**, Vamsikrishna Patchava; Hari Babu Kandala; P Ravi Babu

[14] **Raspberry Pi based interactive home automation system through E-mail,** Sarthak Jain; Anant Vaibhav; Lovely Goyal

[15] **IoT Based Home Automation Using Raspberry PI,** Prof B.P Kulkarni 1, Aniket V Joshi 2, Vaibhav V Jadhav3, Akshaykumar T Dhamange4

[16] **IoT Based Home Automation Using Raspberry PIIoT based monitoring and control system for home automation,** D. Pavithra; Ranjith Balakrishnan.

Appendix A

We describe the programs and websites used to complete this project:

1. **Proteus**: Proteus is a software package for computer-aided design, simulation and design of electronic circuits, consist of two main parts, the ISIS, the circuit design environment that even the simulator VSM includes, and the ARES, the PCB designer.
2. **Microsoft Office Word**: Microsoft office word is a word processor developed by Microsoft It was first released in 1983 under the name multi-tool word for Xenix systems and offer enchained feature to create professional-quality documents, easier ways to work together with people and almost, anywhere access to your files.
3. **Visual Studio Code** is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.
4. **Magicplan**: Instantly create and share floor plans, field reports, and estimates on site with one easy-to-use application. Increase productivity on the go and have an easy way to connect to the office. It's lightning fast, powerful, light-weight and affordable.

Appendix B

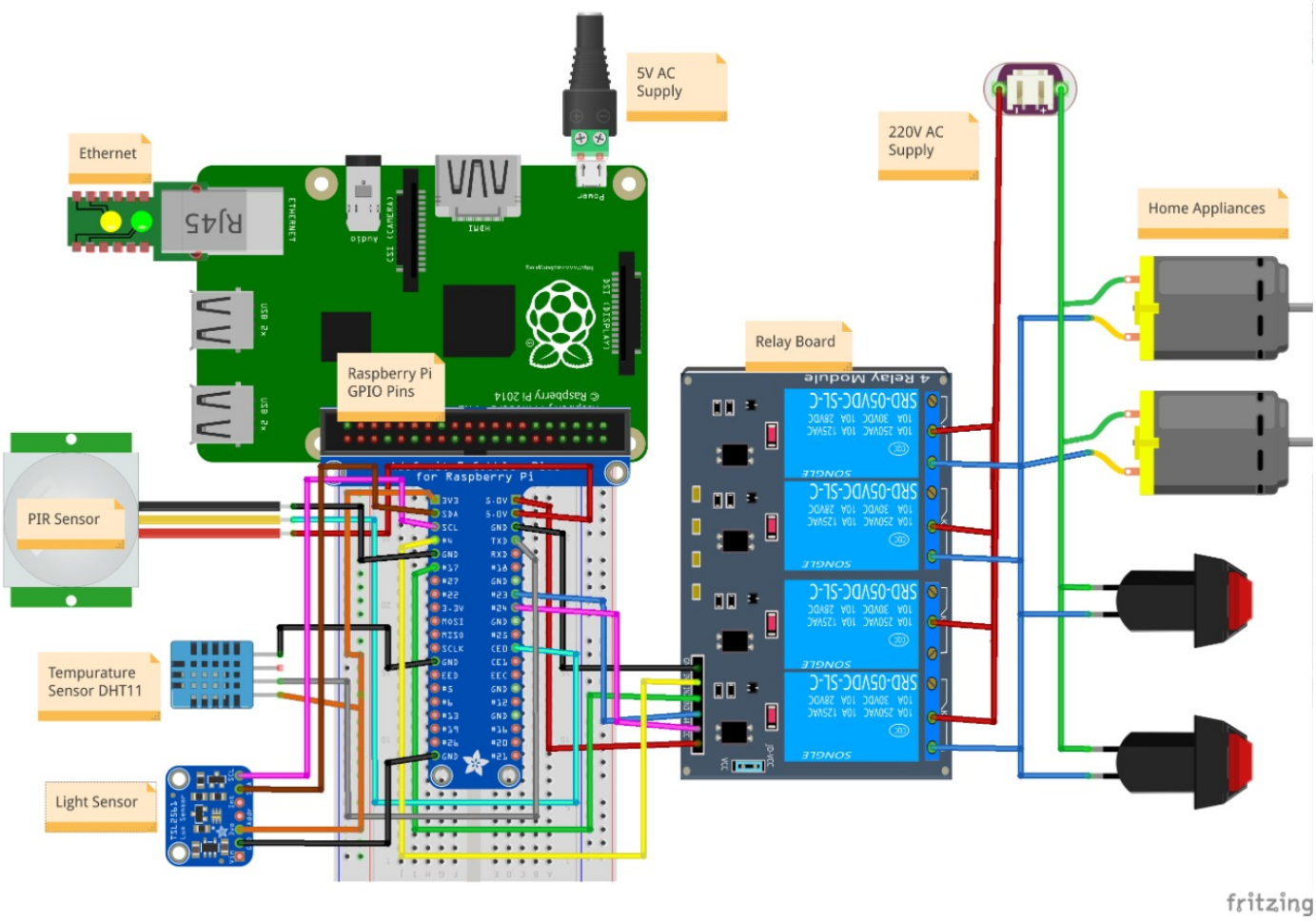


Fig Appendix. 1: Circuit Diagram.

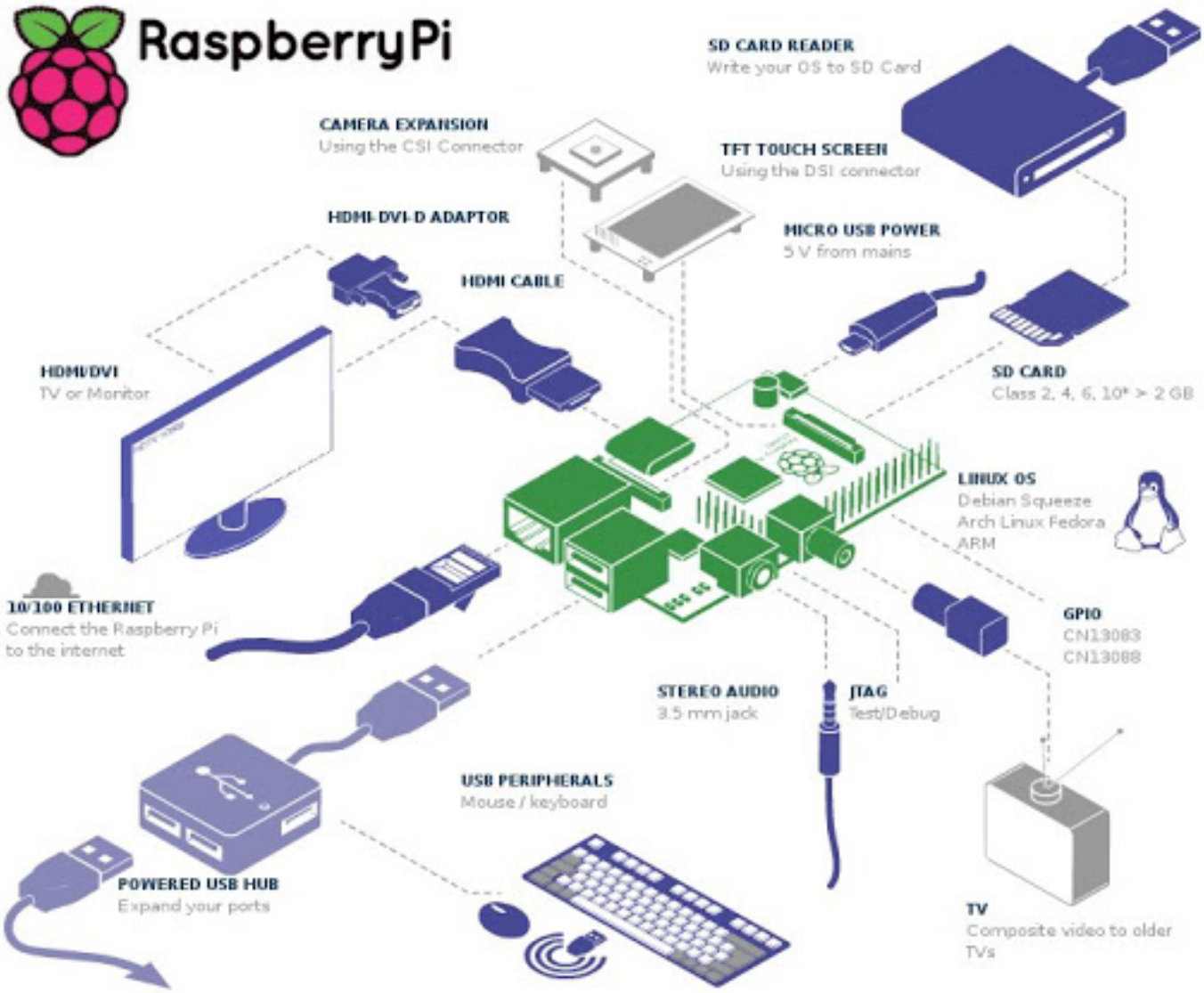


Fig Appendix. 2: raspberry pi specifications.

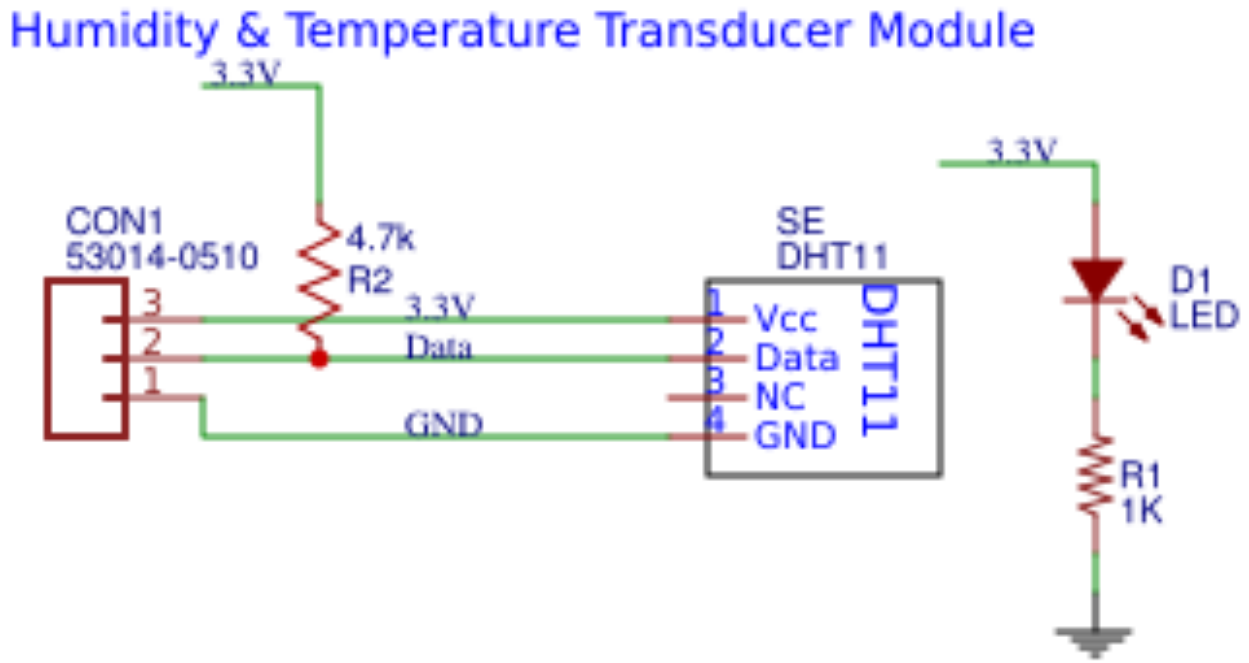


Fig Appendix. 3: Humidity and Temperature Transducer module.

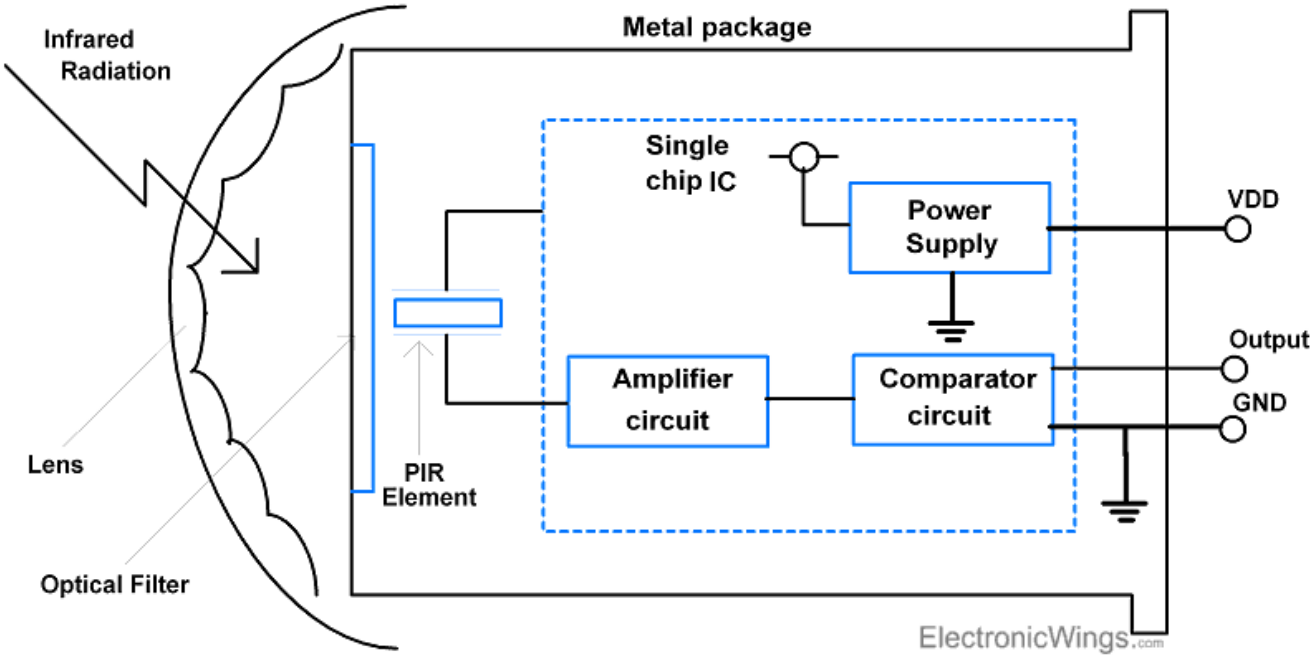


Fig Appendix. 4: Internal PIR Blocks.

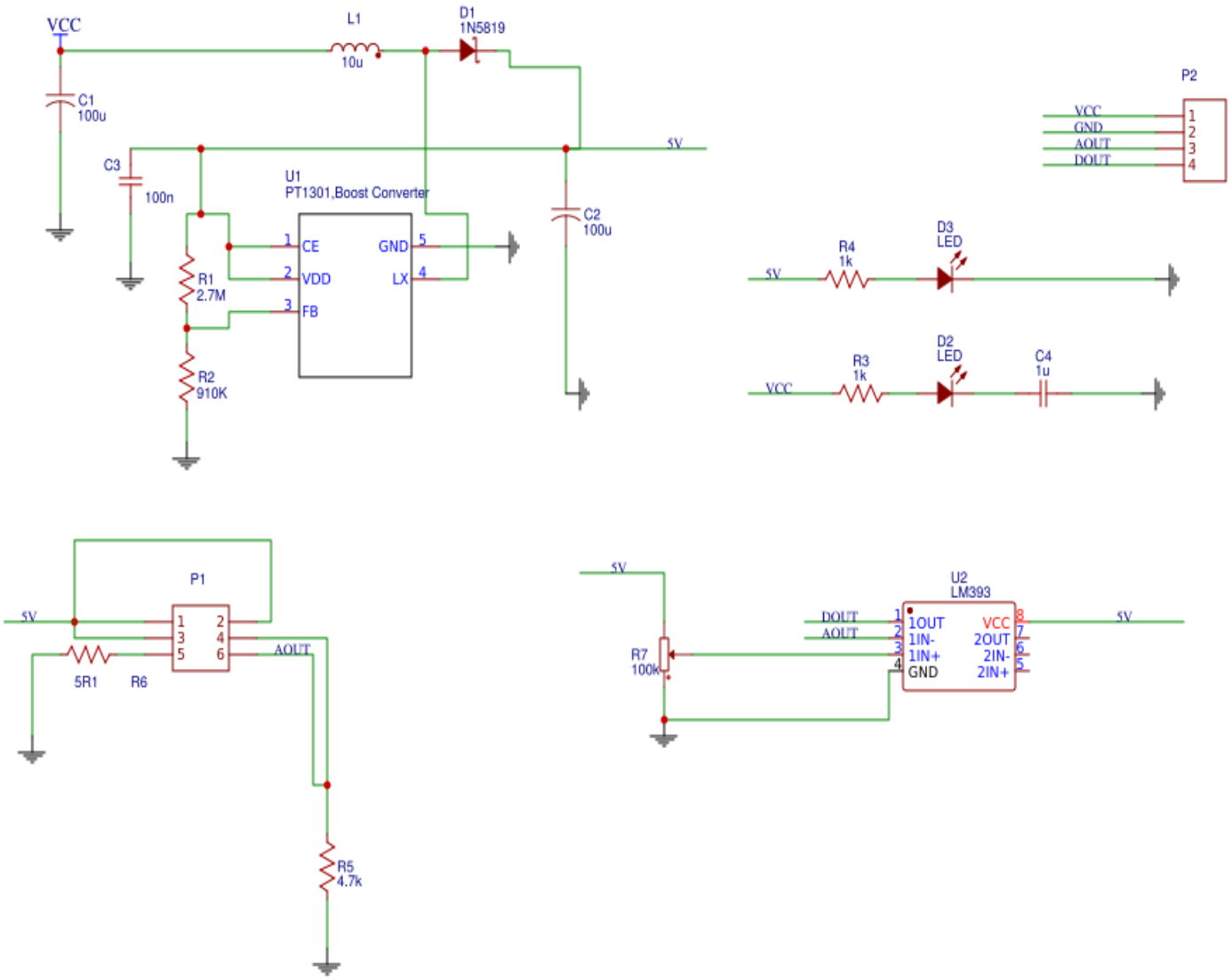


Fig Appendix. 5: MQ5 Schematic.

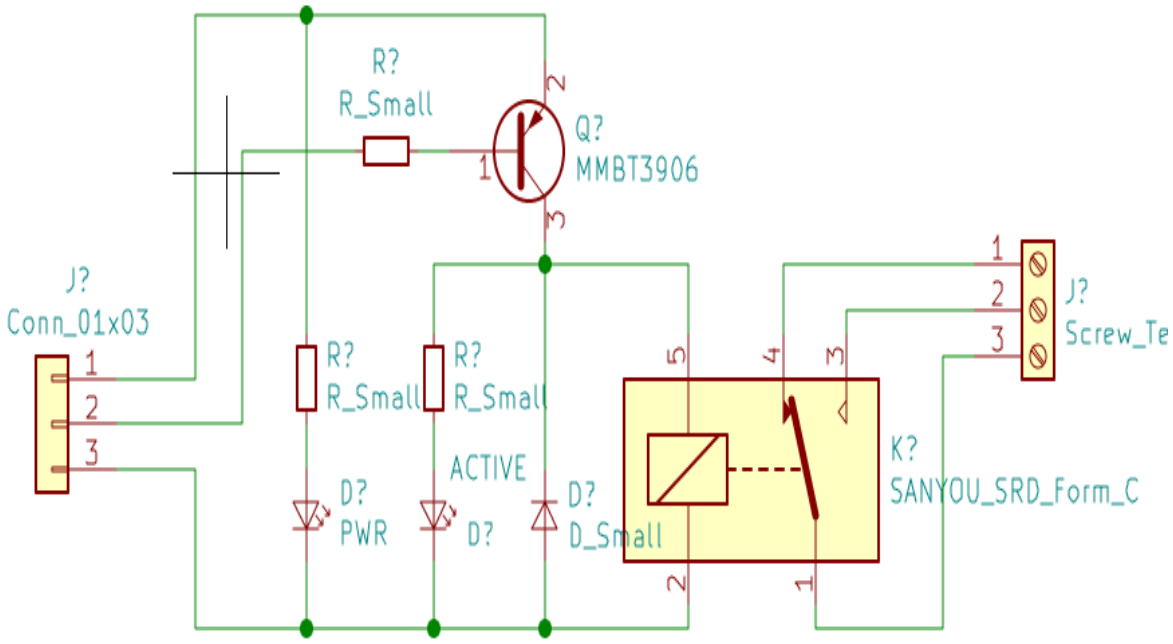


Fig Appendix. 6:Relay module schematic.

Appendix C

Code (configuration.yaml)

```
# Configure a default setup of Home Assistant
(frontend, api, etc)
default_config:

# Text to speech
tts:
  - platform: google_translate

group: !include groups.yaml
automation: !include automations.yaml
script: !include scripts.yaml
scene: !include scenes.yaml

########################################
###########  GPIO Ligth   ##############
########################################

switch:
  - platform: rpi_gpio
    ports:
      17: Garage Door On
      27: Fan
      22: Living Room
      10: Bed Room
      9: Kitchen
      11: Bathroom
      5: Lamb 1
      6: Lamb 2
      13: Nigth Light
      20: open main door
      21: close main door
      19: open garage
```

```
      26: close garage

#####################################
#########   DHT11 Sensor  ###########
#####################################
sensor:
  platform: dht
  sensor: DHT11
  pin: 23
  monitored_conditions:
    - temperature
    - humidity

#####################################
##########   PIR Sensor  ############
##########   MQ-5 Sensor  ###########
#####################################
binary_sensor:
  - platform: rpi_gpio
    ports:
      24: PIR Office
      25: MQ-5
```