A Project Report on

IoT Enabled Smart Laboratory

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

in

Information Technology

by

Prathmesh Pande (16104020)
Ritesh Shetty (16104048)
Bhavana Kondurkar (16104066)

Under the Guidance of

Prof. Vishal Badgujar

Department of Information Technology
NBA Accredited
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2020-2021

Approval Sheet

This Project Report entitled ***"IoT Enabled Smart Laboratory"*** Submitted by ***Prathmesh Pande(16104020), Ritesh Shetty (16104048), Bhavana Kondurkar (16104066)*** is approved for the partial fulfillment of the requirement for the award of the degree of ***Bachelor of Engineering*** in ***Information Technology*** from ***University of Mumbai***.

Prof. Vishal Badgujar	Prof. Kiran Deshpande
Guide	Head of Department of Information Technology

Place: A.P. Shah Institute of Technology, Thane

Date:

CERTIFICATE

This is to certify that the project entitled ***"IoT Enabled Smart Laboratory"*** submitted by ***Prathmesh Pande(16104020), Ritesh Shetty (16104048), Bhavana Kondurkar (16104066)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Information Technology***, to the University of Mumbai, is a bonafide work carried out during academic year 2020-2021.

Prof. Vishal Badgujar
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Acknowledgement

We have great pleasure in presenting the report on **IoT Enabled Smart Laboratory.** We take this opportunity to express our sincere thanks towards our guide **Prof. Vishal S. Badgujar**, Department of I.T., APSIT Thane, for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project. We thank **Prof. Kiran B. Deshpande**, Head of Department of I.T., for his encouragement during progress meeting and providing guidelines to write this report. We thank **Prof. Vishal S. Badgujar** B.E. Project Co-ordinator, Department of I.T. for being encouraging throughout the course and for guidance. We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Prathmesh Pande
16104020

Ritesh Shetty
16104048

Bhavana Kondurkar
16104066

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Prathmesh Pande
16104020

Ritesh Shetty
16104048

Bhavana Kondurkar
16104066

Date:

Abstract

Internet of Things is defined as a system wherever appliances are embedded with software system, sensors and actuators. The devices enable information transfer over a network and additionally communicate with one another. This system is incorporated in our labs to create the appliances convenient and automatic. During this epoch of Automation, it's been ascertained that a lot of the University and faculty Labs do not have IT automation capabilities up to the present industrial trendy standards. The most recent technologies are not being enforced in faculties and so the fellow students cannot enjoy them. Automation is a section that is gaining attention progressively day by day since last few years. The aim behind our project is to assist appliances to not only connect with one another, but also in a user friendly manner. Here we have a tendency to look forward a system which might offer the user complete management over all remotely governable aspects of the Laboratories. One can accomplish laboratory automation by merely connecting the appliances to a central network or cloud storage.

Contents

List of Figures

List of Abbreviations

IoT: Internet of Things

MQTT: Message Queuing Telemetry Transport

NORA: NOde-Red home Automation

GPIO: General-Purpose Input/Output

AC: Air Conditioner

Chapter 1

Introduction

There are several devices in an exceedingly laboratory, like Lamps, Fans, Air Conditioners and Projectors. Laboratory automation will be represented as a technology that is employed within the campus setting to supply comfort, convenience and energy potency to its user and students. Automation may be a conception that involves real time management and observance of multiple appliances. Today, there's an increasing demand of machine-controlled systems so human intervention is reduced. The user will build use of this technique to regulate switch on of lights, fan, AC, etc. mechanically. The user will access complete IoT system from anyplace exploiting web. Raspberry-Pi which is a tiny sized computer accordingly acts as a server for the system. A smart laboratory could be a place that has automatic systems for lighting and temperature, laboratory appliances, and security and lots of alternative functions. Through IoT, each appliances of our labs will be connected to the internet. IoT plays a very important role in building good laboratory.

As the field grows each year, new management issues and energy problems arises. Managing the resources within the field has become a true drawback. Controlling and monitoring of the unused devices that are left running, thereby consuming power in the absence of human is additionally a serious issue. It is troublesome to watch all sub-systems like Lighting, Projectors and Air-conditioner system. The present usual way to manage the appliances within the work is to manually toggle switches on the switch board. However, that in itself may be a long task as someone needs to be out there to do by themselves. Our planned system is geared towards developing an automatic resolution wherever even though the end user is at a location away from the area, the appliances will be turned ON and OFF. The web interface essentially converts smartphones and computers into a remote control for all appliances. The user will access complete IoT system from any place with the help of internet. Therefore, the decisive motive behind making the system like laboratory automation is that you simply will how the devices should respond. All the appliances awill be joined along through relays and nodes in NodeRED and devices can also be controlled via MQTT protocol based on ESP8266 via Node-RED. Raspberry-Pi is tiny sized computer that acts as a main server as well as MQTT Broker for this system.

Cloud will be one of the most promising and cost-efficient answer to connect, manage and track the IoT ecosystem. Cloud computing could be a model for on-demand access to a shared pool of configurable computing resources (example - networks, servers, storage, applications, services, and software) that may be simply provisioned as and once required. Cloud computing provides an abstracted interface aggregates the resources to achieve efficient resource utilization and permit users to rescale to resolve larger science issues. It permits the system software system to be designed as needed for individual application needs. For analysis teams, cloud computing can offer convenient access to reliable, high performance clusters and storage, without the necessity to buy and maintain subtle hardware.

Research has thrived from the rapid climb in machine power. With this, comes increased pressure on labs as information storage facilities ought to house the exponentially-increasing quantities of huge information sets, as huge information becomes an integral part of analysis. This poses an important drawback for labs, during which they will need an entire area dedicated to its on-the-spot storage. And thereupon comes maintenance, resulting in hefty up-front IT infrastructure prices. Cloud computing has helped to alleviate this burden, thereupon comes maintenance, resulting in hefty up-front IT infrastructure prices.

Cloud computing has helped alleviating this burden, by removing the necessity for corporations to host their own data center. Instead, the information will be kept on the cloud. For laboratories, cloud computing centralizes information, assuring security, while facilitating collaboration. One among the advantages of cloud computing is unifying information. Cloud computing permits labs to collaborate without the excess cost and complexity of running onsite server rooms. Changing from an onsite resolution to the cloud alleviates the prices of IT infrastructure. Integrating all the devices, and appliances of the laboratory with the cloud infrastructure is another advantage which cloud computing provides. One amongst the most widespread product within the market is Cubus science lab, a plug-and-play resolution that is a laboratory execution system and collects instrument information in real time, remotely. Another necessary good thing about cloud computing is information safety. It may be possible that one may lose their laboratory information because of outages or cyber attacks, or worst-case, a natural disaster. Deploying systems within the Cloud escapes the matter of data loss. If a cascading failure were to occur, the information will not be lost as most files are at different location. What appears to be holding back several labs is that the concern of losing information by change to cloud computing. In reality, it is a lot safer and reliable than on-site information storage. As expressed by Ranjan, Praful [6], the market for the home automation can increase based on many key enhancements within the technology offered in automation, like enhancements in wireless automation solutions as well as lowering of costs because the market simply begins to adopt home automaton usage in larger volumes.

Chapter 2

Objectives

- To minimize, monetary costs, user discomfort, delays, utilization of resources.
- To automatize the appliance dominant of Labs.
- To reduce the power consumption by economical usage of the appliances.
- To provide cloud access so that any device can access remotely regardless of being in same network or not.
- To attempt integration of laboratory time scheduling with the system.

Chapter 3

Literature Review

M. Poongothai, A. Rajeswari and P. Muthu Subramanian [1] talks about the benefits of IoT in general, how automation can be useful, and also the use of MQTT and Node-RED for automation. They stated that more than 85 percent of systems remains unconnected, and do not share information with one another or the cloud. One such technology that facilitates the interconnection is eventually Internet of Things. Application code written for interfacing IoT sensible hardware kit and MQTT broker, and for observance temperature, wetness and lightweight intensity within the laboratory. Developed Dashboard and mobile application by using Node-RED and Android Studio. A database set has been created for a paradigm switch to look at standing history. IoT reduces the human intervention by introducing device to device interaction. By using the projected system, the whole energy consumption is reduced in our field.

In the paper by R. K. Kodali and A. Anjum [2], an efficacious home automation system through affordable Wi- Fi development boards is projected. They explained how Node-RED can be used as a visual wiring tool that helps in associating gadgets simply bringing concerning quick and easy association setups. Gadgets are connected along to using Node-RED and an association is setup for remote ESP8266 and a mosquito primarily based MQTT broker observance and management. Node-RED could be a programming tool for wiring along hardware devices, on- line services and APIs. Further they [2] explained: Varied parts in Node-RED are connected along to form a flow within the Node-RED editor. Message Queuing telemetry Transport (MQTT) provides a light-weight electronic messaging protocol that uses a publish/subscribe model attributable to that it's worthy to be used on all devices from low power boards to servers. A consumer will publish or take a subject or do each. Whereas as a broker receives all the messages, filters them and send them to the signed consumer. Node-Red is therefore an economical platform to link variety of IoT gadgets and may be controlled from any a part of the planet.

T. Malche and P. Maheshwary [3] bring into light about what is a smart home and what role it plays in the society. According to them, a smart home is a connected home (or E-Home), consisting of many technologies which enhances the quality of living, an area that has extremely advanced automatic systems for various functions of controlling and sensing, such as alert, monitor, control, Intelligence, and a system with IoT infrastructure as a prime base. Further they [3] talked more in depth about it as follows: An IoT primarily based smart home is rising as a very important a part of the sensible and intelligent cities that are being projected and developed round the world. the aim of a wise house is to boost living customary, security and safety still as save energy and resources. The smart home plays a very important role in development of society.

S. Somani, P. Solunke, S. Oke, P. Medhi and P. P. Laturkar [4] focuses on a system that gives options of Home Automation using Internet of Things via an android application, where an it (android application) acts as an interface between user and the IoT system to control and manage devices. The core components of this system are Raspberry-Pi, various sensors and appliances, where Raspberry-Pi acts as a server which accepts requests from user, logs in the user by forming a client socket between the two, and in return, the server responds back to the user if correct details are provided.

In the paper by H. K. Singh, S. Verma, S. Paland and K. Pandey [5], they aim to develop an IoT based home automation system which is primarily based on Wi-Fi based microcontroller. Here, NodeMCU (ESP8266) microcontroller together with Relays is employed to manage electrical switches remotely from the server that is constructed on Node.js. User will manage switches employing an internet Application when authenticating. The projected system consists of internet server, internet interface, database, NodeMCU and Solid-State Relays. Server controls and monitors appliance state and user command, and designed to handle additional hardware interface module. The web server is running on NodeJS that in turn running on AWS (Amazon web Services). which may be accessed from a web browser remotely over internet. Furthermore, the log file will be generated as a result of dynamical state of appliances

Chapter 4

Project Design

Existing System Architecture

The usual way to control the devices in a laboratory is to manually operate them using physical switches on the switch board. The existing system consists of the administrator being supplied with a distant access to the web server, which the user has to control using an android application, which requires separate programming as well as limits the cross-platform compatibility to a single platform. Following are the typical components included in these kind of systems:

- **Android Application:**
  The Administrator use an Android smartphone, wherever accessible to any or all the Computers, Lights, Fans, Air Conditioners and Projectors that are connected in network.

- **IoT Kit:**
  Typically includes an embedded board with WiFi support, which hosts the Android Application and necessary functions.

- **Appliances:**
  Lights, fans, projectors, air-conditioners and PC switches are connected through IoT kit to the entire system.

- **Other connected electronic devices:**
  This may include typical home devices such as:
  - HVAC
  - Smart Locks
  - Washing Machine.
  - Oven

  and so on.

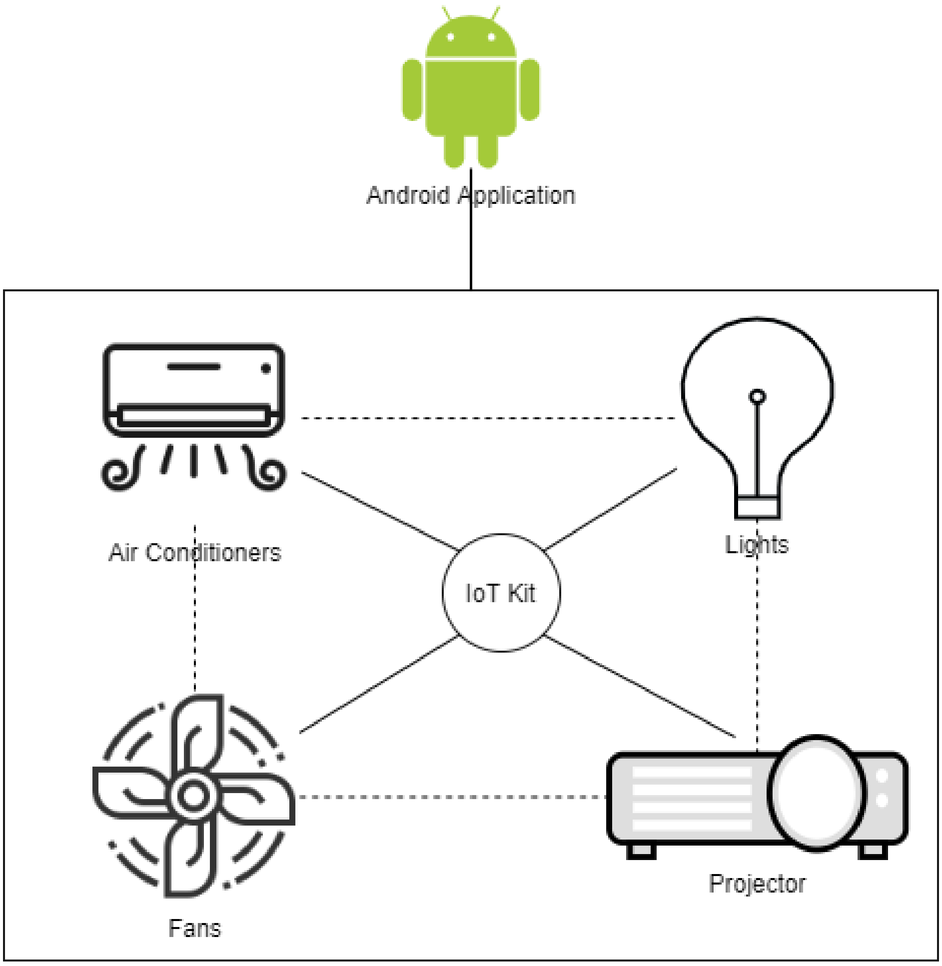

Figure 4.1: Existing System Architecture

Creating an application for a specific platforms separately, for the same functionality, not only requires a lot of effort and skills, but also makes it tedious to maintain them individually and keeping them on-par with each other in a programmatic stand point.

Proposed System Architecture

In our proposed system, rather than building an android application, we are using a programming tool called Node-RED, which helps us establish a link between our connected devices and our automation system. Not only that, it also has a built-in dashboard functionality, and everything is accessible via a browser, which ensures cross-platform compatibility.

Fig. 4.2 shows the planned system architecture style of IoT enabled smart laboratory system. All the appliances among the laboratory are connected to Relays, which are controlled using Node-RED and MQTT broker, hosted on Raspberry-Pi.

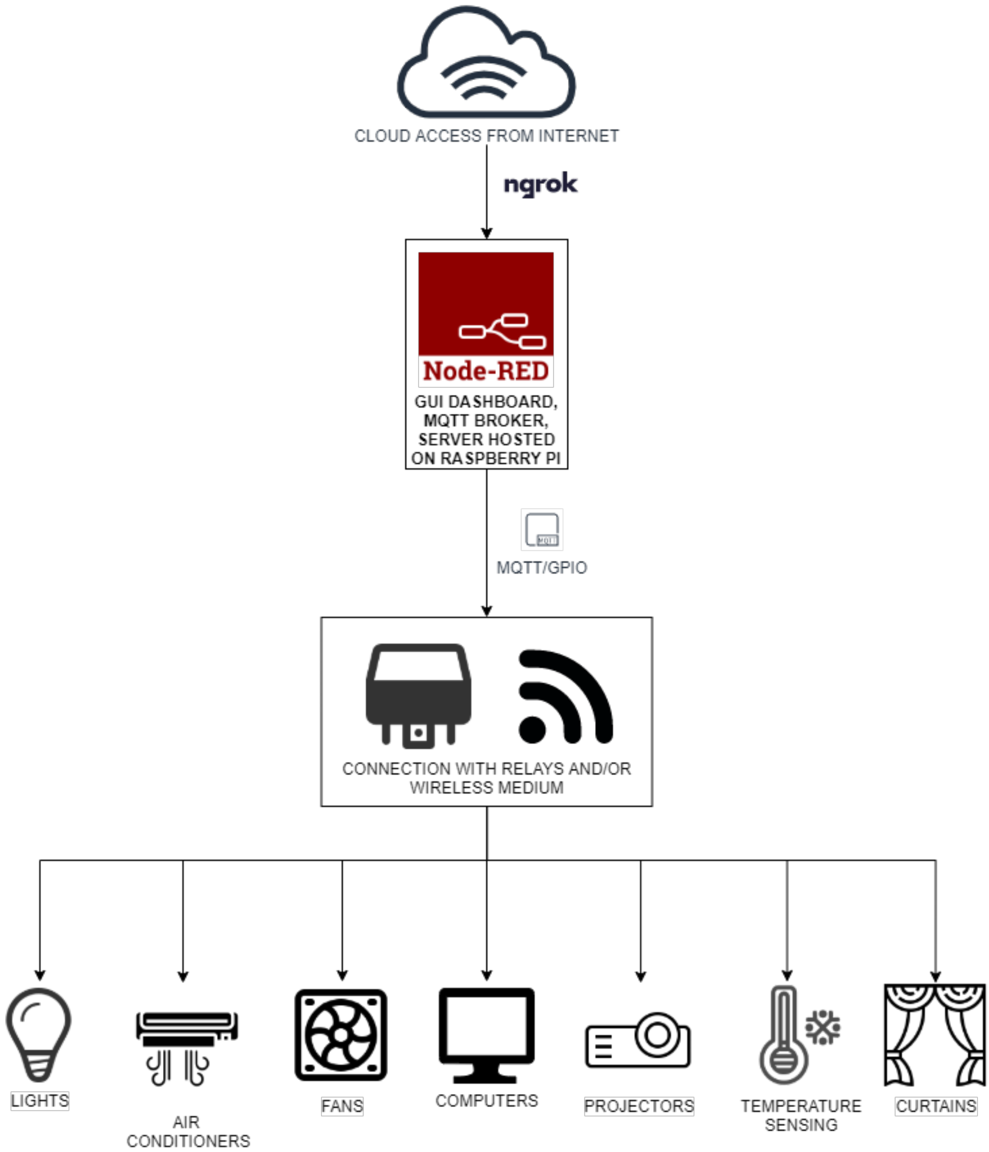

Figure 4.2: Proposed System Architecture

The components of this architecture are as follows:

- **Node-RED:**
  In our proposed system, rather than building an android application, Node-RED programming tool is used for wiring along hardware devices, APIs and online services. It provides us a browser-based editor which simplifies wiring along flows using the big selection of nodes within the palette that may be deployed to its run-time during a single- click which could also be deployed to the run-time. Along with this, using Node-RED, the system will have a Graphical, User-Friendly interface dashboard, which is accessible via a browser, for managing and controlling the devices.Taking inspiration from Sfikas, Giorgos and Akasiadis, Charilaos and Spyrou, Evaggelos. [8], we can look into categorising various parts of our system supported sensors (S–type services) and actuators (A–type services), physically put in within the space, yet as processing units (P–type services) running within the cloud, and also locally.

- **MQTT:**
  For wireless devices, we will use MQTT as our protocol, which will be used to receive all messages and command from the client devices like lights, AC, Computers, Fan, curtains, etc. (called as MQTT Clients), which are connected to Raspberry-Pi using Relays.

- **MQTT Broker:**
  To actually implement MQTT Protocol, we need a MQTT Broker. For that, Node-RED also provides an MQTT broker, both for publish and subscribe as nodes, so we do not need to implement it separately. The ESP8266 based devices will be able to communicate with Node-RED wirelessly via WiFi.A real-time alarm system [7] that uses WiFi as a communication medium between the Raspberry-Pi and also the Node-Red platform is one of the examples of a ZigBee MQTT client.

- **Raspberry-Pi:**
  The Raspberry-Pi will be our choice of embedded board because of its versatility at a reasonable price. Here the Node-RED will be hosted on top of Raspberry-Pi OS, as well as it will act as a MQTT server for the connected appliances. In our system, we have chosen Raspberry-Pi 3 Model B.

- **Cloud Access via NGROK:**
  NGROK is a program which allows us to expose the local host to the public over secure tunnels. With NGROK, we can expose our Raspberry-Pi, which is hosting Node-RED and all the connected appliances, to the internet and will allow the user to have access to the dashboard and editor from anywhere, being in any network. Users can use smart phones, laptops or computers to monitor and control lab device from anywhere. The main models of cloud

- **Smart NORA:**
  Node-RED home Automation (or NORA) is a Node-Red integration service for Google Home. With NORA, we can integrate our smart lab automation flows with google home, via google assistant. Due to its limited capability of handling multiple devices,

Smart NORA has born. Smart NORA provides the same simplicity in setup but now with a completely rewritten back-end based on Google Cloud that can support tens of thousands of active users and a new client integration package that provides better stability and connectivity. It will allow the devices to simply able to be controlled via the Google Home application as well.

- **Relay Modules:**
  Relays are a type of switches that open and close circuits electromechanically or electronically. Relays control one electrical circuit by opening and closing contacts in another circuit. The appliances will be connected to these Relays and then controlled by Node-RED. There are many types of Relay Modules available in the market for cheap, like 2 channel, 4 channel, 16 channel, as we as relays with built-in ESP8266 module integrated to it. Depending on the number of devices, we can choose the type of relays accordingly.

- **Time Scheduler Node:**
  The Time Scheduler Node named "node-red-contrib-ui-time-scheduler" allows us to schedule activation and deactivation of various. It can be used to switch devices ON or OFF between a given time period on a daily or weekly basis, as well as a one time event only.

Chapter 5

Project Implementation

To implement our system, we are using a basic setup of four 2-channel relays, which allows us to wire upto 8 devices together. These relays are then connected to raspberry pi and interfaced via GPIO pins present on this embedded board. Through Node-RED, we are creating a dashboard which consists of two separate flows for 2 Laboratories, namely ’Lab1’ and ’Lab2’, which in this scenario represents 2 laboratories respectively. We allocated 2 relay module for each of the labs so effectively each lab has 4 switches. Fig. 5.1 is the breadboard circuit diagram which shows how the 2 channel relays are connected with raspberry-pi. The Figure 5.2 is the schematic of the circuit.

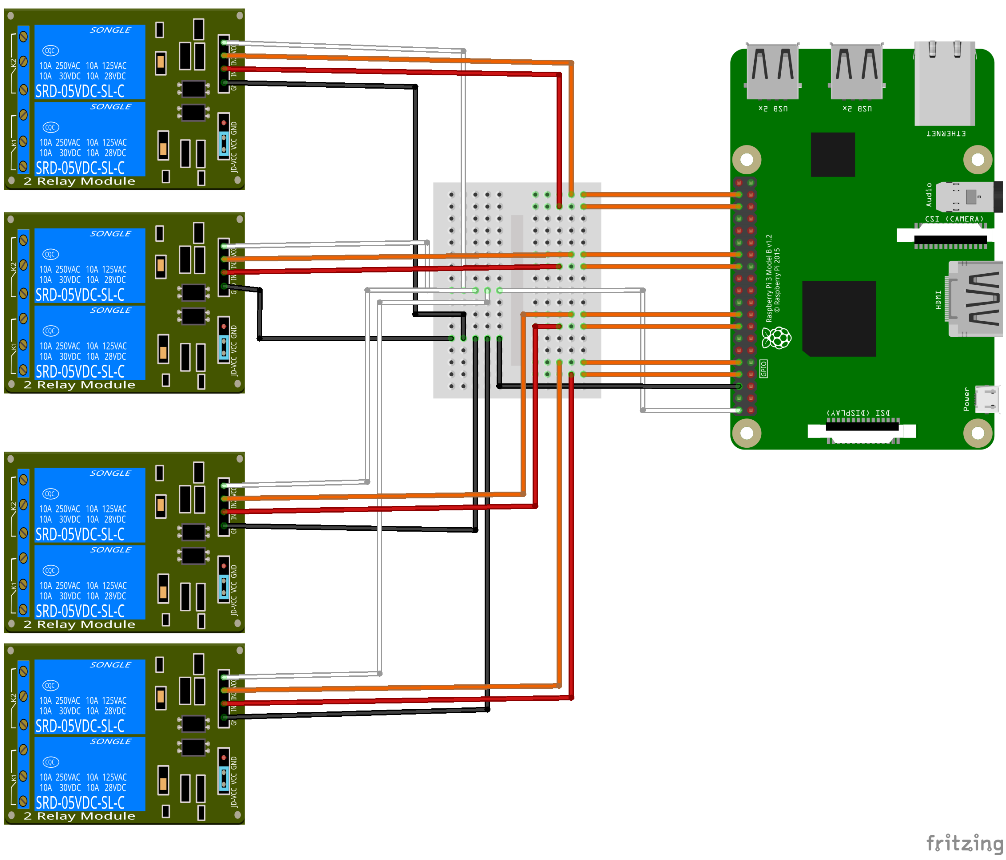

Figure 5.1: Relay Connection with Raspberry-Pi

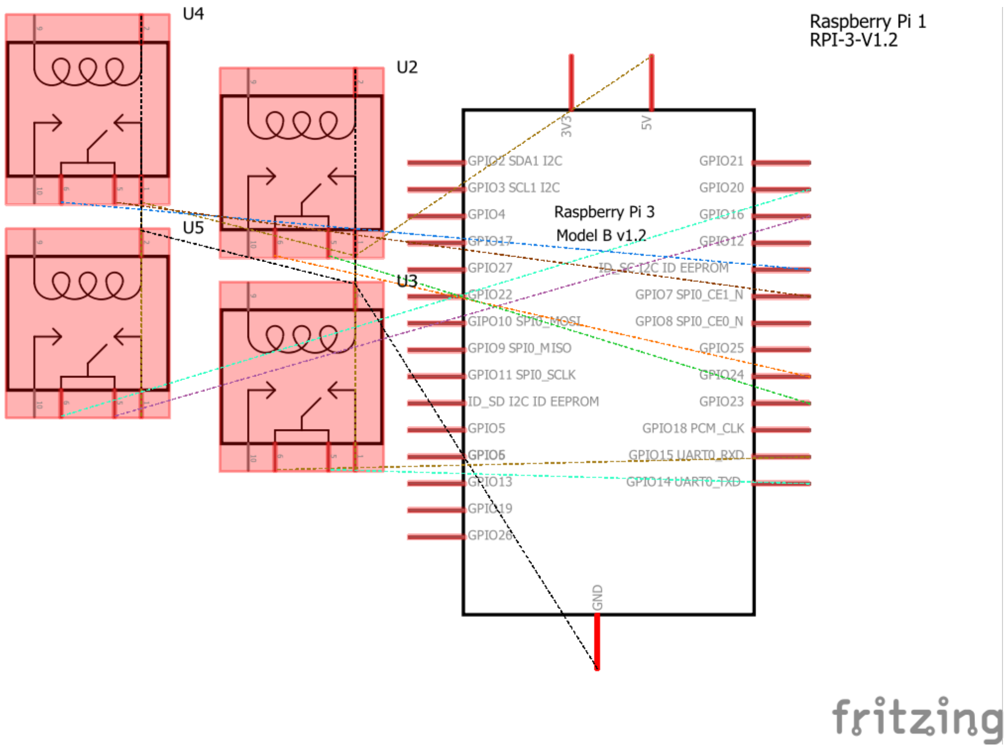

Figure 5.2: Schematic Diagram of the Circuit Connections.

In the above figures, the black connections correspond to the ground. The White connections correspond to 5V power (i.e. VCC). The Red connections represent Input-1 of all the relays. Similarly, the orange connections represent Input-2 of each of the 2-Channel Relay Modules.

The relays are powered by Raspberry-Pi itself, though it is advisable to power them externally if there are large number of them. All the Inputs of the relay modules are connected to the GPIO Pins of the Raspberry-Pi in order to send the signals from Node-RED on Raspberry-Pi to the input of the relays. The Node-RED Editor allows us to insert various palettes and connect them to create a flow of functions.

In order to allow our system (more specifically Node-RED on our Raspberry-Pi) to be accessible via internet anywhere, we are using an application called ngrok, which is a cross-platform application that allows us to expose a web server running on a local machine to the internet, in our case, it is the Raspberry-Pi based server on which Node-RED is hosted. We'll go over it in more detail in the coming pages.

Before using Node-RED, it is recommended to create an account on ngrok to generate the authentication key, which will be needed for tunneling, via the ngrok node in Node-RED as shown in the figure below.

The ngrok node takes two inject nodes to turn the tunneling ON or OFF. When we inject the ON node, in the debug window, we can see a URL which is being generated. Anyone having access to that URL can obtain cloud access and control to the Node-RED Editor as well as Dashboard. The URLs will also be available through the ngrok dashboard itself.

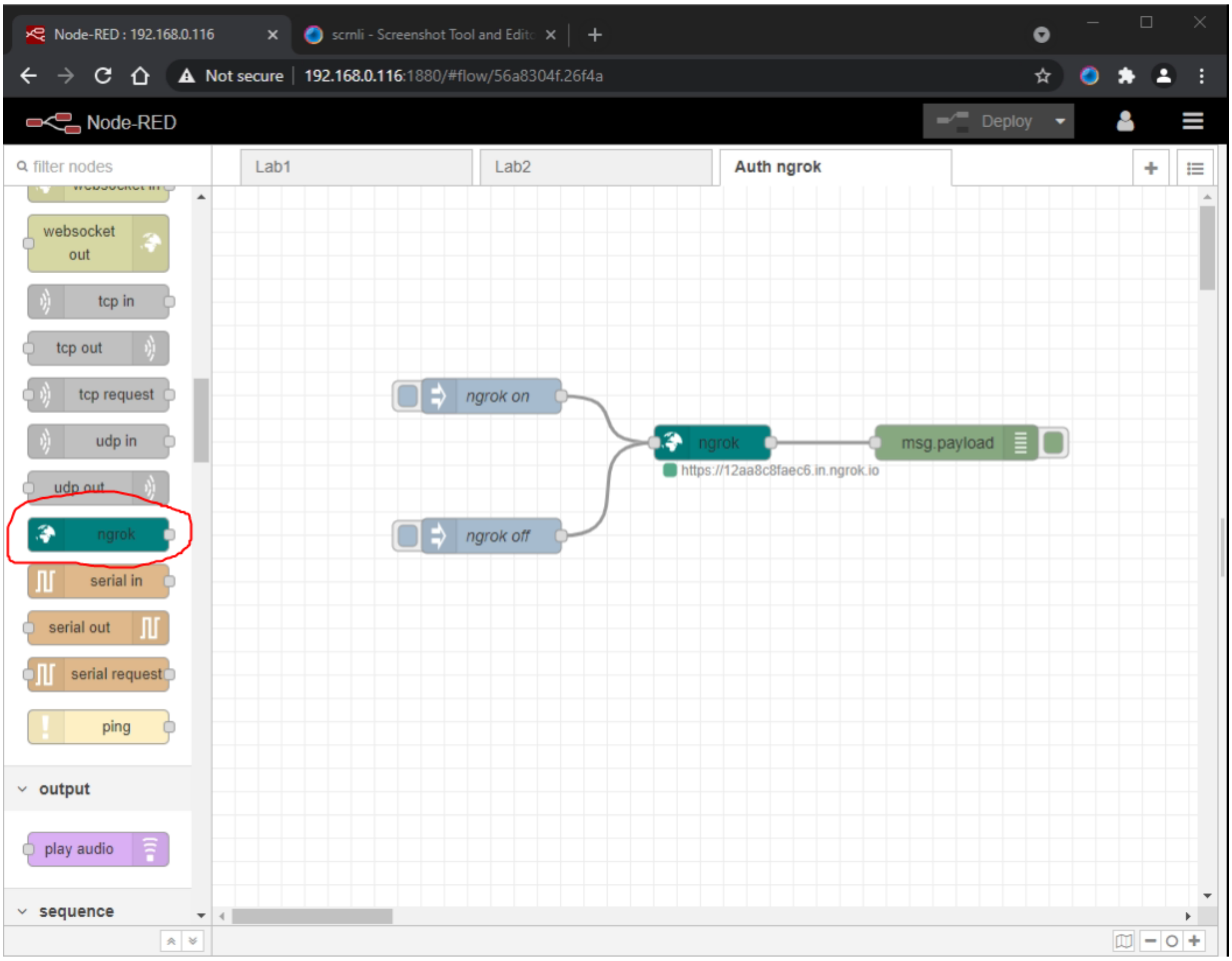

Figure 5.3: The ngrok flow in Node-RED.

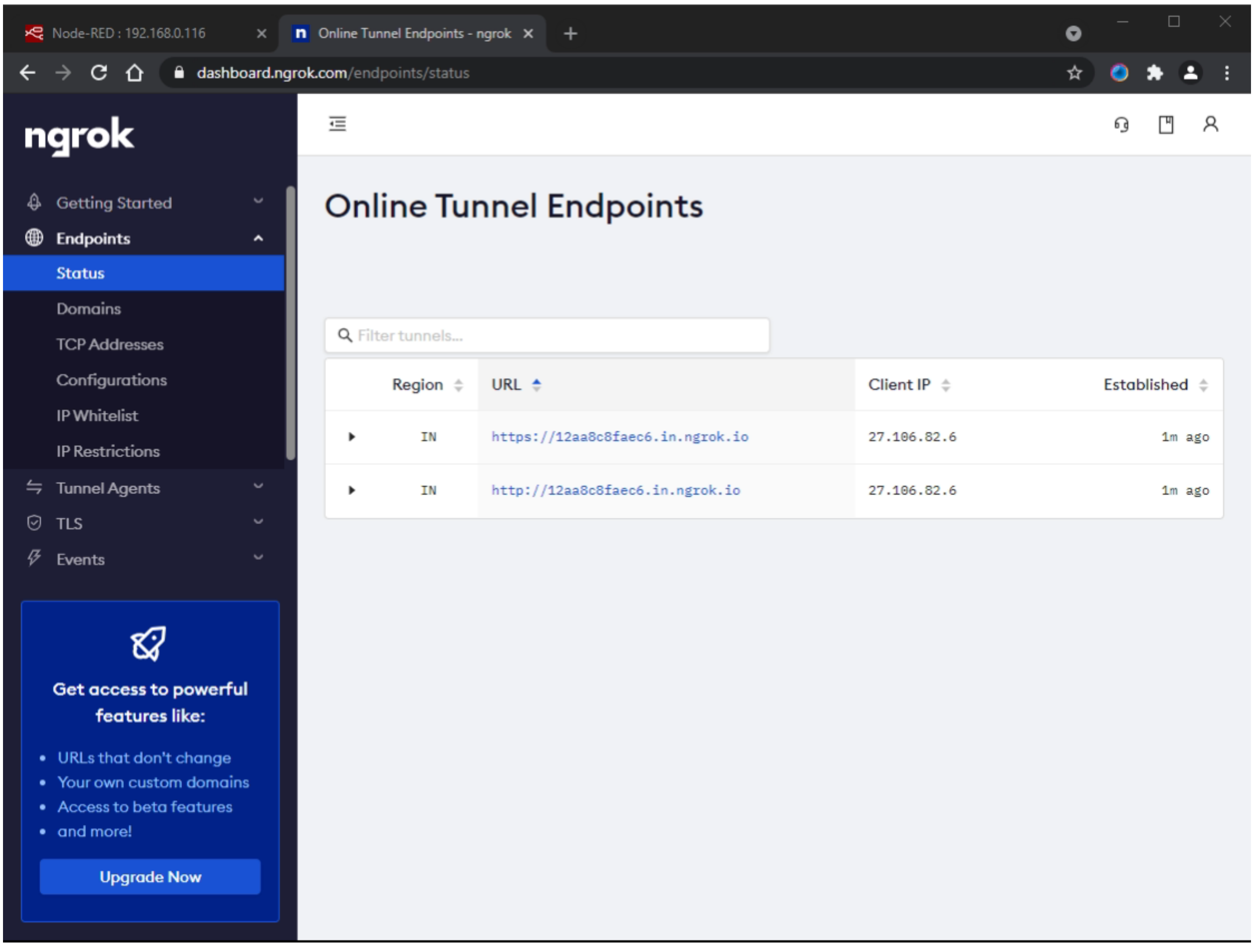

Figure 5.4: The ngrok Dashboard.

Chapter 6

Testing

For our testing, the relays which are connected to Raspberry-Pi board via GPIO pins, are being controlled via switch nodes present on the Node-RED Editor, which will be visible on the dashboard as toggle switches, as shown in Fig 6.1. The nodes with the Google Assistant icons are the nodes provided by NORA, which will allow all the appliances to be able to communicate with the Google Home application as well as controlled via voice. Each Lab can have its own separate flow to avoid any confusion and keep things organized.

The final circuit after implementation is shown in the Fig. 6.2. Here we also connected a bulb on one of the relays of the 2-Channel relay modules, which we will be controlling via the Node-RED Dashboard

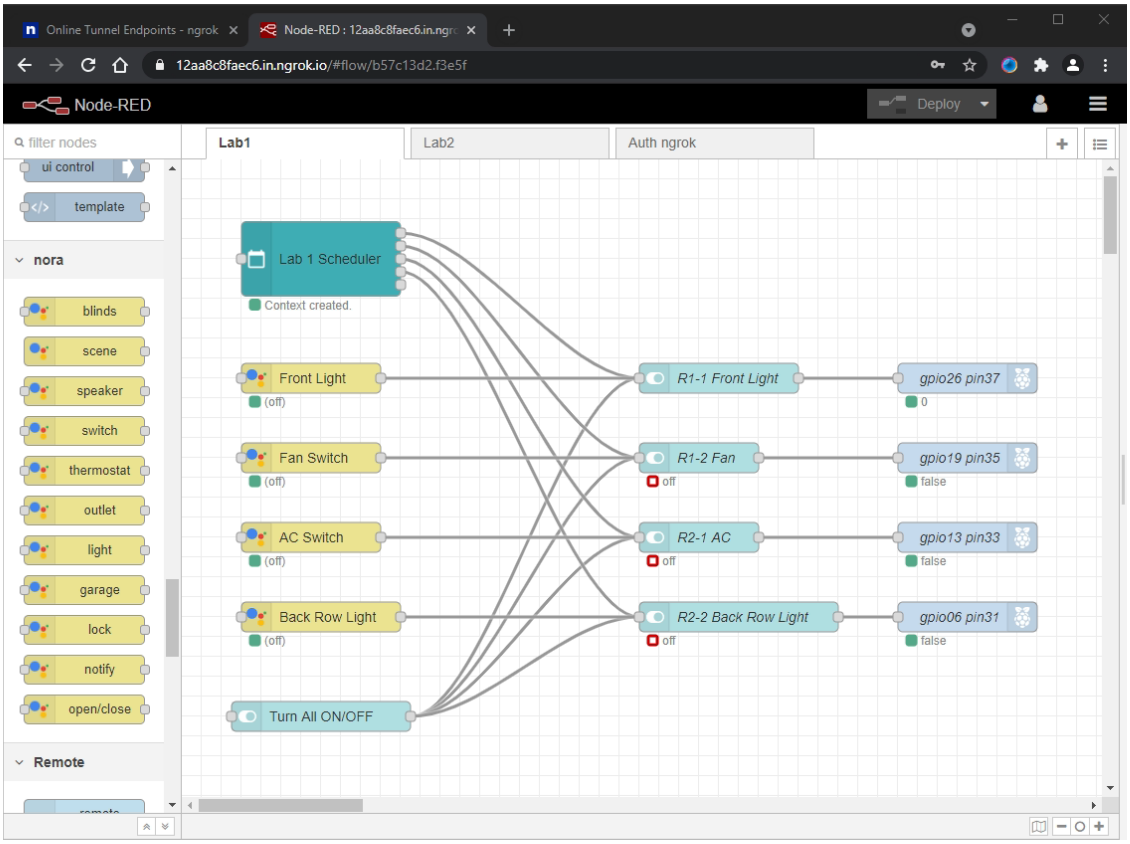

Figure 6.1: Node-RED Editor.

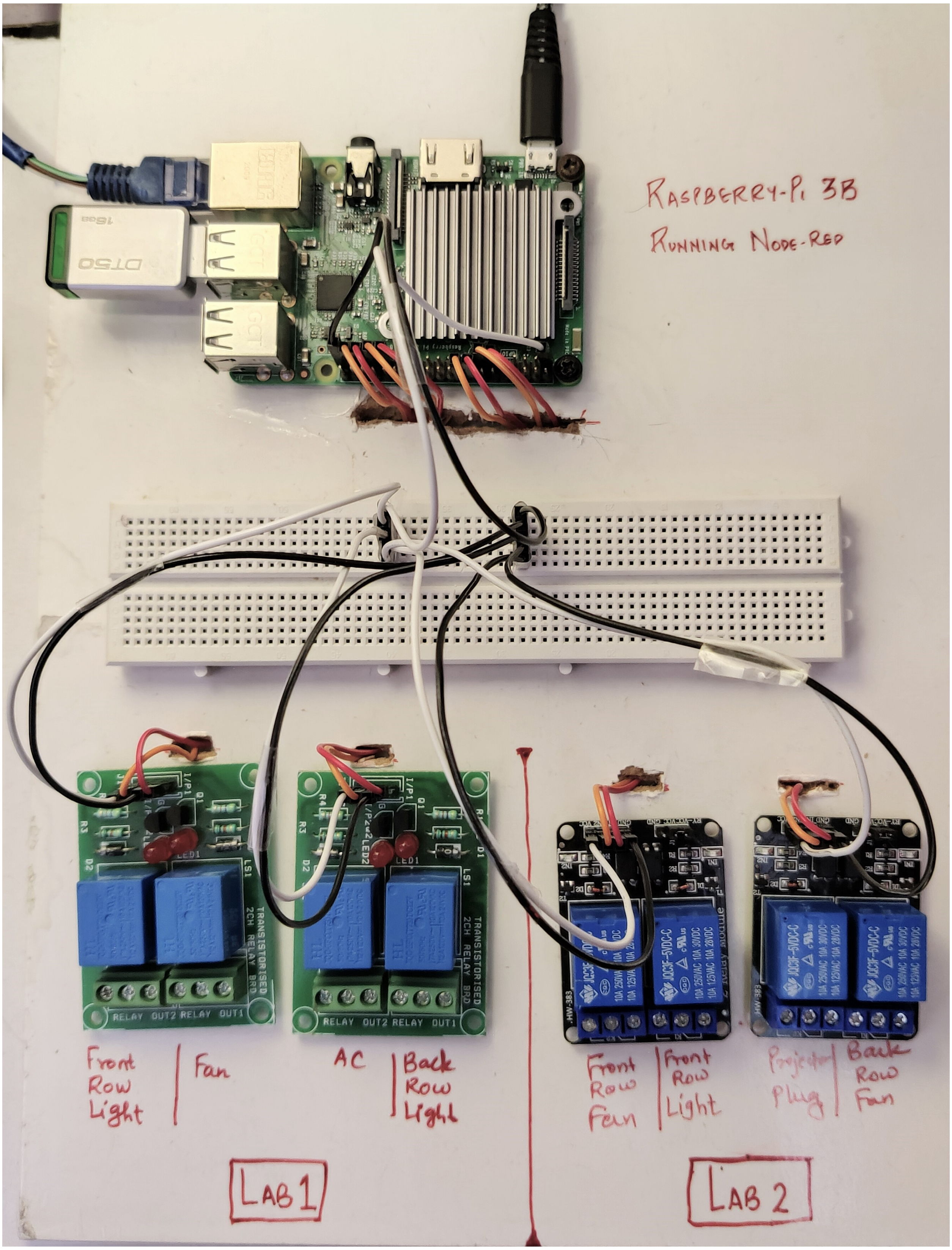

Figure 6.2: Implementation of the circuit.

Chapter 7

Result

With this system, we have much more flexibility to adjust the system as per the requirements. We are able to control all the devices added and connected to Node-RED. With Time Scheduler, we are able to schedule a time period for activation and deactivation of the devices as per the time period and days assigned.

The dashboard shown in Fig 7.1 shows us the available toggle switches which were added by using switch nodes on the Node-RED Editor. User can switch between different Laboratories from the tabs present on the Left Hand Side of the Node-RED Dashboard.

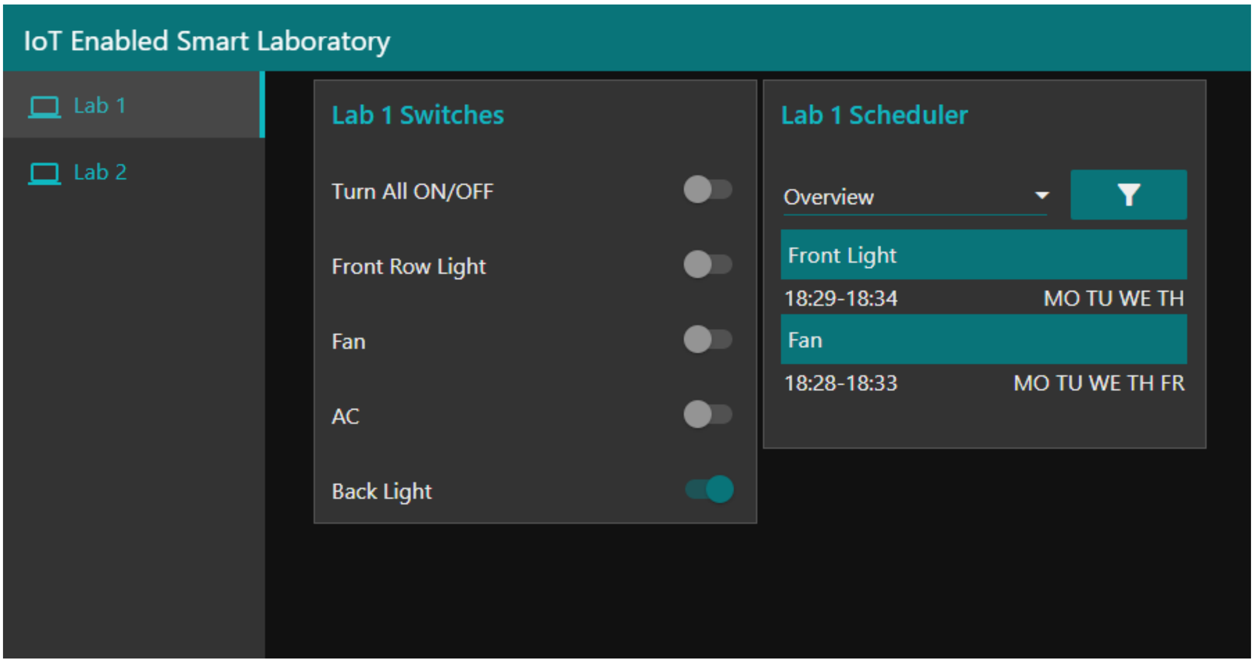

Figure 7.1: Node-RED Dashboard.

The time scheduler allows the user to schedule a time period and days when the devices get triggered on or off. The devices can be selected from the drop down menu, after which the time schedule and the days can be selected. Once done, user can save the changes and all the active schedules will be displayed in the overview pane, as shown in Fig. 7.2.

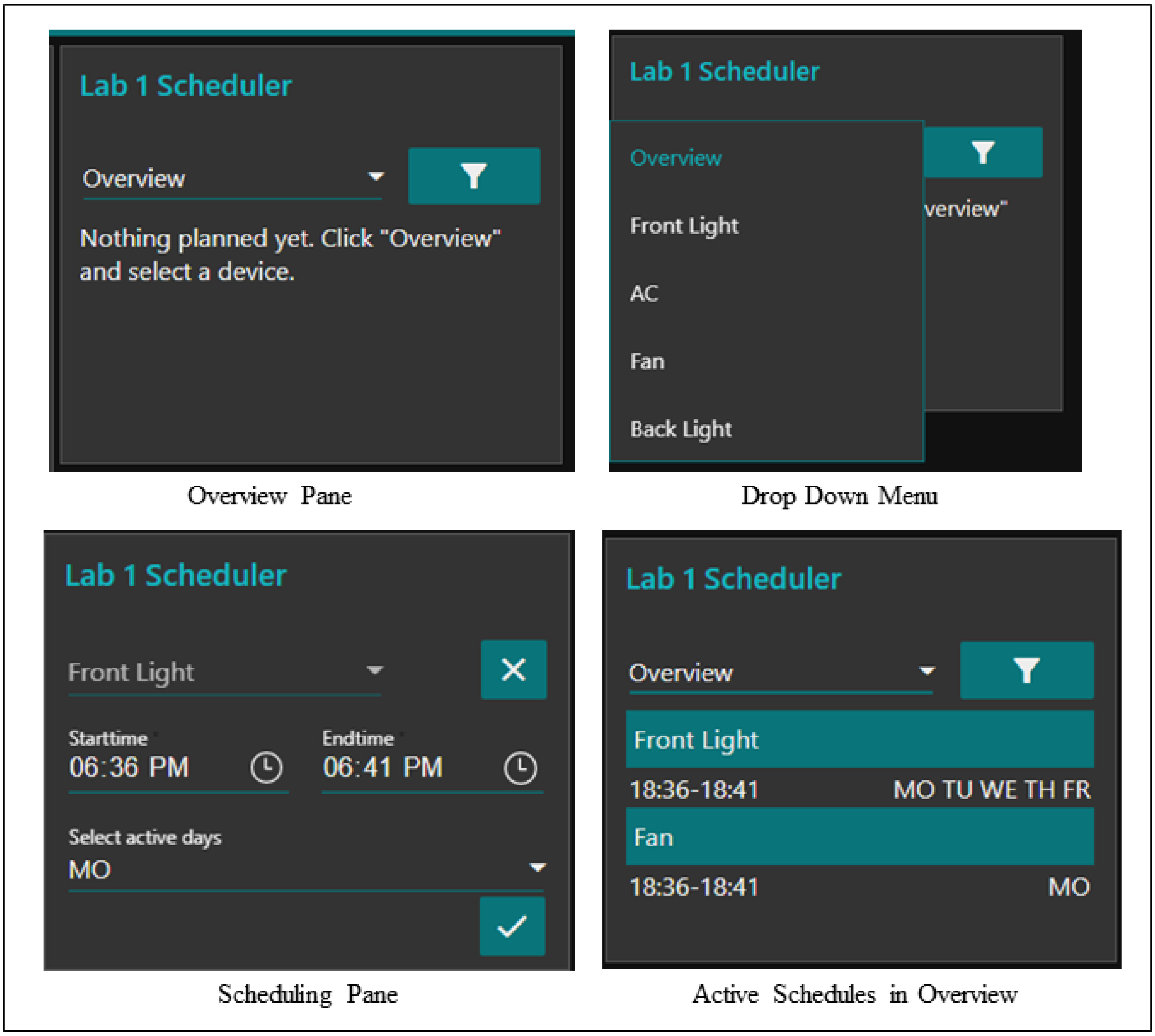

Figure 7.2: Time Scheduler UI.

The Smart NORA nodes allowed us to access the devices in the Google Home app as shown below, where we can control them via voice or manually, as shown below. Node-RED is a flexible tool which also supports integration with various industry standards like MQTT, ZigBee, Alexa voice controls and many more, which allows integration of many services for a cohesive environment.

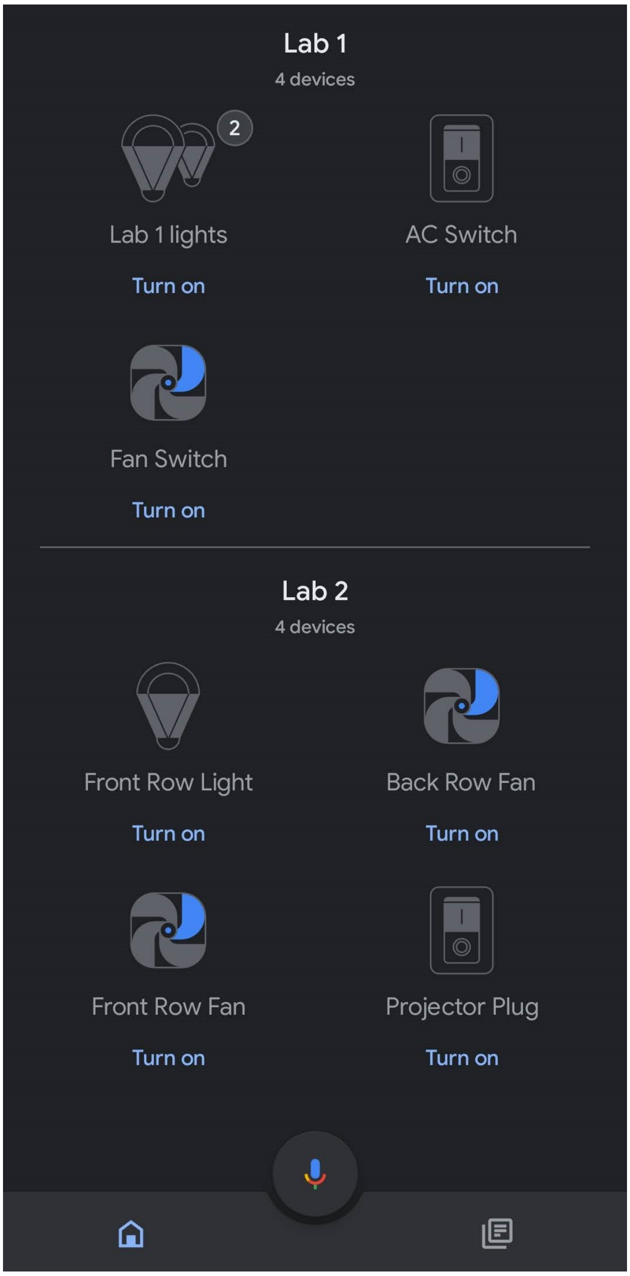

Figure 7.3: Google Home Integration.

Chapter 8

Conclusions and Future Scope

Considering the same issues in our university, we've listed the problems and located out the solutions to reduce the inconvenience for the research lab assistants and college members to the utmost. The Node-RED user interface dashboard is accessible via browser, it ensures cross-platform compatibility. Adding some 'smarter' parts to our life makes our lives easier, and a lot of versatile in terms of day to day usage. the main objective of our project is to cut back the human efforts in labs, that we predict will be achieved using laboratory automation.

The Expected system can ease the complete automation method in labs and create the lab management easily. The projected system not only be used for schools and universities, but also at homes and studios. There are some limitations of our system which include constantly connected internet service required for remote access. Also, the system requires continuous power to be supplied in order to function optimally. Node-RED by its nature, does not support separate flow access in the form of individual flows for multiple user account handling, and uses BASIC authentication as the default authentication standard, though multiple users can be created and given the permissions of read, write, or both, to access and control all the flows.

Bibliography

[1] M. Poongothai, P. M. Subramanian and A. Rajeswari, "Design and implementation of IoT based smart laboratory," 2018 5th International Conference on Industrial Engineering and Applications (ICIEA), Singapore, 2018, pp. 169-173.

[2] R. K. Kodali and A. Anjum, "IoT Based HOME AUTOMATION Using Node-RED," 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), Bangalore, India, 2018, pp. 386-390.

[3] T. Malche and P. Maheshwary, "Internet of Things (IoT) for building smart home system," 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, 2017, pp. 65-70.

[4] S. Somani, P. Solunke, S. Oke, P. Medhi and P. P. Laturkar, "IoT Based Smart Security and Home Automation." 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-4.

[5] H. K. Singh, S. Verma, S. Pal and K. Pandey, " A step towards Home Automation using IOT," 2019 Twelfth International Confernce on Contemporary Computing (IC3), Noida, India, 2019, pp. 1-5.

[6] Ranjan, Praful. (2017). Home Automation Using IOT. 10.21742/ijsh.2017.11.09.01.

[7] Kumar Selvaperumal, Assoc. Prof. Dr. Sathish and Al-Gumaei, Waleed and Abdulla, Raed and Thiruchelvam, Vinesh. (2019). Integrated Wireless Monitoring System Using LoRa and Node-Red for University Building. Journal of Computational and Theoretical Nanoscience. 16. 3384-3394. 10.1166/jctn.2019.8297.

[8] Sfikas, Giorgos and Akasiadis, Charilaos and Spyrou, Evaggelos. (2016). Creating a Smart Room using an IoT approach.

Appendices

Appendix-I: Installing Raspberry-Pi OS and Node-RED

The Raspberry-Pi team and the Node-RED Team has an excellent step-by-step guide documented for the installation of Raspberry-Pi OS and Node-RED respectively.

1. Follow the steps documented by the Raspberry-Pi Team at the following URL: `https://www.raspberrypi.org/documentation/installation/installing-images/`

2. To install NodeRED on Raspberry Pi, the instructions are written at this page: `https://nodered.org/docs/getting-started/raspberrypi`

3. Once Node-RED is installed, open terminal and type **node-red-start**. This will start the Node-RED service. You will see a URL like **http://127.0.0.1:1880**. This is the URL through which you can access Node-RED in the browser. Open the browser and paste the link in the address bar and proceed. You will be redirected to the Node-RED Editor.

4. To access the Node-RED Dashboard, add **/ui** at the end of the obtained URL so it looks like: **http://127.0.0.1:1880/ui**. This will redirect the browser page to the Node-RED Dashboard.

Appendix-II: Installing Smart NORA nodes in Node-RED

In the Node-RED Editor Page:

1. Click on the hamburger menu in the top-right corner and select **Manage palette**. Select **Install**, then type **node-red-contrib-smartnora** and click the install button.

2. Once the installation is done, in your Raspberry-Pi, open a seperate terminal window, and type **node-red-restart**. This will restart the NodeRED service.

3. Now the Smart NORA nodes will be visible in the left side of the editor page under **NORA** section.

4. To configure NORA, follow this complete step-by-step guide at this URL: `https://github.com/andrei-tatar/node-red-contrib-smartnora/blob/master/doc/setup/README.md`

Appendix-III: Installing Time Scheduler node in Node-RED

In the Node-RED Editor Page:

1. Click on the hamburger menu in the top-right corner and select **Manage palette**. Select **Install**, then type **node-red-contrib-ui-time-scheduler** and click the install button.

2. Once the installation is done, in your Raspberry-Pi, open a seperate terminal window, and type **node-red-restart**. This will restart the NodeRED service.

3. Now the Time Scheduler node will be visible in the left side of the editor page.

4. For configuration of Node-RED Time Scheduler, follow this complete step-by-step guide at this URL provided by the author of this node: `https://flows.nodered.org/node/node-red-contrib-ui-time-scheduler`

Appendix-IV: Setting up ngrok.

1. Sign up for ngrok at `https://dashboard.ngrok.com/signup`. Once done, login to ngrok account.

2. Once done, tou will be presented with the ngrok dashboard. On the left side pane, go to 'Get Started', then 'Your Authtoken'. Here you will find an authtoken generated for you. We will need this token in the coming steps.

3. Click on the hamburger menu in the top-right corner and select **Manage palette**. Select **Install**, then type **node-red-contrib-ngrok** and click the install button.

4. Once the installation is done, in your Raspberry-Pi, open a seperate terminal window, and type **node-red-restart**. This will restart the NodeRED service.

5. Now the ngrok node will be visible in the left side of the editor page, drag it onto the editor. Open up the ngrok node properties and specify the port number. In case of Node-RED, the default value is 1880.

6. In the authtoken field, click on the edit button and paste the auth token which you received in step 2.

7. Finally create 2 inject nodes, ON and OFF. The ngrok node takes the inject node inputs to start and stop tunneling.

For more information or help relate to ngrok configuration, visit `https://dev.to/vonagedev/getting-started-with-ngrok-in-node-red-4cn7`

Publication

Paper entitled **“IoT Enabled Smart Laboratory”** is accepted at **“ICT4SD 2021 Sixth International Conference on ICT for Sustainable Development”** by **“Prathmesh Pande”**, **“Ritesh Shetty”**, **“Bhavana Kondurkar”** and **“Prof. Vishal Badgujar”**.

ICT4SD - 2021 : Notification for Paper : 46 : IoT Enabled Smart Laboratory External Inbox ×

ICT4SD Conference <support@ict4sd.org> Fri, Apr 30, 9:16 AM (10 days ago)
to rrshetty, me, bskondurkar, vsbadgujar

Dear Prathmesh Pande, Ritesh Shetty, Bhavana Kondurkar, Vishal Badgujar,

Ref : 46

Title : IoT Enabled Smart Laboratory

Congratulations! On behalf of the Program Committee of ICT4SD 2021 – Goa, India, I am happy to inform you that your above mentioned paper has been ACCEPTED for oral presentation in ICT4SD 2021 and publication in Springer LNNS series subject to fulfillment of Guidelines by Springer. An accepted paper will be published in the Springer proceedings LNNS only if the final version is accompanied by the payment information (i.e transaction reference number) subject to quality check as per Springer Guidelines.

Important Notification COVID 19 - As we are passing through unprecedented times, and India is going through a second wave of COVID19, we as organizers understand it well and are trying to monitor the update from the Central and State Governments in India. Also taking into consideration of current government guidelines we are also trying our best to make this conference beneficial for all authors. The conference will be organized in physical Mode during 5 - 6 August 2020 at Goa subject to SOP's and Government guidelines near to the dates. Further for those authors who do not wish to attend the conference physically have been relaxed and can present their papers in digital sessions which will be organized via ZOOM. In both the cases conference papers will be moved to publication. Link : https://ict4sd.org/participation-policy.php

For all updates and details please visit - https://ict4sd.org/COVID-19.php