

Voice Home Automation System

Sudeep Gokarn
Computer Science
State University of New York
at Binghamton
sgokarn1@binghamton.edu

Aakash Chavan
Computer Science
State University of New York
at Binghamton
achavan4@binghamton.edu

ABSTRACT

Home automation has gone from a luxury item, to a trend, and now to must-have technology. Many tasks that are repetitive in nature can be accomplished automatically or with fewer steps using home automation. Instead of turning off or dimming four different lights when you want to watch a movie, home automation allows you to accomplish this task with one button.

Many home automation systems are already available in the market, but all of the existing automation systems control devices which are wireless. None of the existing automation system controls old wired devices who don't have wireless connectivity. Most of the them are not cross-platform as they are tightly coupled to a specific operating system (like Android, iOS). Existing Home automation systems do support a specific set of devices but configuring the system to control a newer device is a complex task.

So there is need to develop a system which can connect to wired devices, which is cross-platform and which is easy to install & configure.

KEYWORDS

NeoPixel LEDs, NodeMCU, RaspberryPi, iOS application, TCP Protocol.

1. INTRODUCTION

Home automation involves introducing a degree of computerized or automatic control on certain home appliances. We are proposing a system which would help everyone to get their old wired devices on to a network and allow them to control them through a mobile app. This system doesn't require any expensive hardware and it can be easily manufactured on a large scale will very low cost. That doesn't mean this system won't give the performance which is expected from an automation system. This system uses specialized components designed for a specific task and this combination of components works seamlessly. Distributing the sensor controlling

task to NodeMCU, processing task to powerful yet tiny RaspberryPi and user friendly GUI to mobile app makes this system small and yet smart.

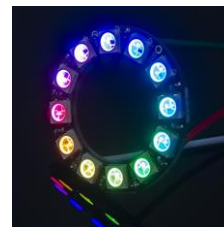
2. DESIGN

The approach we took in designing our system was to divide it into four modules and those modules are Sensors (NeoPixel LEDs), a Sensor Controlling Unit, a Processing Unit and a Front-end(iOS mobile application). We decided to use this modular design because it made it easier for each person to work on each part without affecting the other module. Then we just need to connect the modules together with standard protocols like TCP for communication.

3. IMPLEMENTATION

The implementation of "Voice Home Automation" is four-fold. The Sensors (NeoPixel LEDs), NodeMCU, a Middleware (RaspberryPi) and a Front-end (iOS mobile application). Following sections explain the implementation of each part.

3.1 NeoPixel LEDs



NeoPixel LEDs is digital LEDs compatible with NodeMCU. They are connected to NodeMCU via wire connectors. NeoPixel LEDs will receive a command from the NodeMCU and they will change their states as per commands.

3.2 NodeMCU



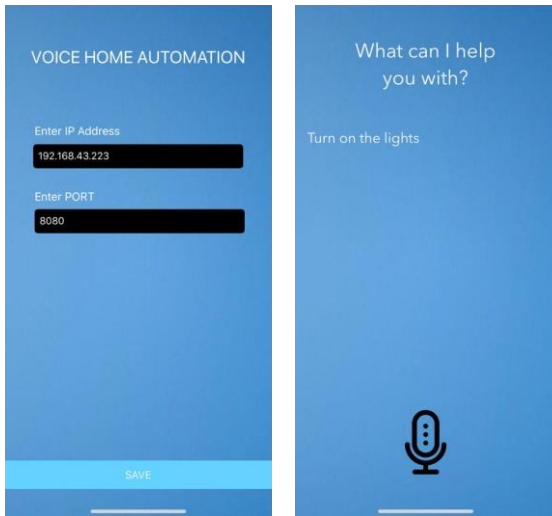
It provides wireless connectivity. NodeMCU is the component which makes wired device wireless. It takes data from Raspberry Pi and controls sensors (like Neopixel LED in this case) as per commands.

3.3 RaspberryPi



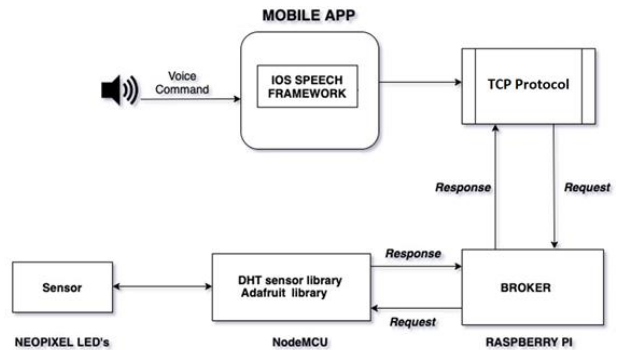
The RaspberryPi is a very important component of the system. It acts as a middleware between NodeMCU and the Mobile Application. It receives voice commands in the form of text from the mobile application. It interprets the text and converts them into commands which can be understood by NodeMCU.

3.4 Mobile Application



The Mobile Application will connect to RaspberryPi using TCP protocol. On the welcome screen, there is a provision of setting information like IP & Port No. Mobile Application has a button to record the voice commands. Once the voice is recorded it is displayed in a mobile application in the form of text and it is also forwarded to RaspberryPi for further processing.

4. System Architecture



System Architecture is made on four components NeoPixel LEDs, NodeMCU, RaspberryPi and the Mobile Application. Any action is always originated from the mobile application. Mobile application records the voice and sends its text equivalent to RaspberryPi. All the interpretation logic resides in RaspberryPi. It maintains different datasets which have lots of keywords. It tries to match keywords from text sentence with the dataset keywords and then it transforms the English text into a command which can be understood by NodeMCU. RaspberryPi sends the commands generated after interpretation to NodeMCU for further processing. NodeMCU has handling for all the commands which it expects from RaspberryPi. It checks the sensor type & action type receive in the command & sends the signals to the respective sensor.

5. EVALUATION

We have successfully evaluated our project by using NeoPixel LEDs, NodeMCU, Raspberry pi and the iOS mobile application. We have tested the system with following scenarios:

- A. Voice command as 'Turn On the light': NeoPixel LEDs are turned white as it was just a 'On' command.
- B. Voice command as 'Turn the lights Red': NeoPixel LEDs changes their color to red.
- C. Voice command as 'The room is very bright': Here the interpretation logic comes into action since it has to understand the sentence and transform it into command. Lights are turned Off on receiving this command which is the expected output.
- D. Voice command as 'Brighten the room please': Again the interpretation logic works fine in this case as well and it turns On the LEDs.
- E. Voice command as 'How are you?': This command has no action compatible with any of the sensors available so the Error Message is displayed on the Mobile application.

Here is a YouTube Link to our demonstration - <https://www.youtube.com/watch?v=VtPLDq7by4o>

6. CONCLUSION

In this paper, we proved that the Voice Home Automation system is a very useful utility which lets you access home appliances quickly and more efficiently. User-friendly iOS UI makes it very easy for a user to control home devices. Our project has exceeded our expectations by use of specialized components designed for a specific task. We could have implemented the interpretation logic on NodeMCU but it wouldn't have given the performance as we are getting with this system. NodeMCU is specialized in controlling sensors but it performs poorly for heavy computation. By keeping the computation logic in RaspberryPi we could able to make the system fast and smart.

7. ACKNOWLEDGMENT

We would like to thank Professor Mo Sha for all the guidance we have got from him throughout the semester. He tested few negative test cases during initial demos which actually opened up a few loopholes in our system. Early identification of such issues helped up in fixing them in time. We also thank our TA Di Mu for his suggestion and feedbacks during demonstrations.

8. FUTURE ENHANCEMENTS

In future, additional devices like Fan, Air conditioner etc. could be added to the system. Currently, the Automation system is compatible with an iOS App. In future, we could develop an Android App/Website to make our system cross-platform. This system could be further enhanced by displaying the devices onto a map so that they could be tracked through the mobile app.