Manual for senior design project

Ndubuisi Iwuala & Kingsley Asonye

University of Maryland Eastern Shore

Senior Design II

Table of Contents

[1. How to Access Alexa with Raspberry Pi 2](#_Toc8766589)

[2. Light Control 3](#_Toc8766590)

[3. Temperature Control 4](#_Toc8766591)

[4. Television Control 5](#_Toc8766592)

[5. Door Sensor 6](#_Toc8766593)

# How to Access Alexa with Raspberry Pi

A screenshot of a computer

Description automatically generatedIn order to access Amazon’s Alexa interface with the Raspberry Pi, you need to connect to the home Wi-Fi and open the command window for the Raspberry Pi. Ensure that the mouse, keyboard, display, and microphone are all connected to the Raspberry Pi. Once the command window is open, type in “cd /home/pi/l3-37-alexa\_patchbots” and then enter. Then type in “./l337.sh run” and press enter. This should bring up this window seen below.

This window shows that Alexa is currently running, and the user can now speak commands to the microphone.

# Light Control

In order to operate the light control outlet supply, first the user will have to have the Arduino IDE installed on a computer. Next you will need to go to the GitHub repository where the code lies. That repository is provided [here](https://github.com/Niwuala/Senior-Design-1.git). Within that repository, there is a file named “NodeMCU Source Code for Light Control.” Copy and paste this code into the Arduino IDE. In the 7th line of the code, there are the Wi-Fi credentials. The figure below shows that. Change the Wi-Fi SSID and password to match your own. Save the new sketch, connect the NodeMCU to the computer and upload the sketch to the NodeMCU. The NodeMCU is now configured for your Wi-Fi. The user can say commands to the like “Alexa. Tell light switch to turn the light on.” Or “Alexa. Tell light switch to turn the light off.”

# Temperature Control

In order to operate the temperature control, first the user will have to have the Arduino IDE installed on a computer. Next you will need to go to the GitHub repository where the code lies. That repository is provided [here](https://github.com/Niwuala/Senior-Design-1.git). Within that repository, there is a file named “NodeMCU Source Code for Temperature Control.” Copy and paste this code into the Arduino IDE. In the 7th line of the code, there are the Wi-Fi credentials. The figure below shows that. Change the Wi-Fi SSID and password to match your own. Save the new sketch, connect the NodeMCU to the computer and upload the sketch to the NodeMCU. The NodeMCU is now configured for your Wi-Fi. The user can say commands to the like “Alexa. Tell thermostat to set the heat to seventy-five.” Or “Alexa. Tell thermostat to set the AC to sixty-eight.”

# Television Control

In order to operate the television control, first the user will have to have the Arduino IDE installed on a computer. Next you will need to go to the GitHub repository where the code lies. That repository is provided [here](https://github.com/Niwuala/Senior-Design-1.git). Within that repository, there is a file named “NodeMCU Source Code for Television Control.” Copy and paste this code into the Arduino IDE. In the 7th line of the code, there are the Wi-Fi credentials. The figure below shows that. Change the Wi-Fi SSID and password to match your own. Save the new sketch, connect the NodeMCU to the computer and upload the sketch to the NodeMCU. The NodeMCU is now configured for your Wi-Fi. The user can say commands to the like “Alexa. Tell remote to go to channel four.” Or “Alexa. Tell remote to turn TV On/Off.”

# Door Sensor

In order to operate the door sensor, first the user will have to have the Arduino IDE installed on a computer. Next you will need to go to the GitHub repository where the code lies. That repository is provided [here](https://github.com/Niwuala/Senior-Design-1.git). Within that repository, there is a file named “NodeMCU Source Code for Door Sensor.” Copy and paste this code into the Arduino IDE. In the 7th line of the code, there are the Wi-Fi credentials. The figure below shows that. Change the Wi-Fi SSID and password to match your own. Keep the Arduino IDE open.

Once that is complete go to ifttt.com and create a free IFTTT account. Create a New Applet by creating clicking “New Applet.” Search for the “Webhooks” service and save the event name as “Door\_Open.” Click on the icon +that and search “Andriod SMS.” Click connect and enter your phone number and click create action. Once that is done, click on “My Applets” and open the newly created applet. Click on the webhooks icon. When the page opens click on documentation. There will be a URL there with an empty space that says {event}. In that space type in “Door\_Open” and copy the URL. Go back to the Arduino IDE. On the 69th line there will be a URL in quotation marks.

Replace that URL with the one that was just copied. Save the new sketch, connect the NodeMCU to the computer and upload the sketch to the NodeMCU. The NodeMCU is now configured.