

Smart home is a house that uses currently released IOT technology to automate different activities of home. IOE devices connected to the internet, to allow the distant monitoring and controlling of different home appliances such as lighting, heating, cooling, and alarming.

There are basically two ways to go through this:

1. **Smart Home installation system:** Here you need to have devices with that support smart home- automation features.
2. **Home Automation:** Try to convert existing connections and building a smart home- automation system.

I used bits of both to do this project.

Equipment and tech stack used to make smart home using existing electrical connections and conventional electrical devices:

1. NodeMCU-esp8266 as the main microcontroller to control devices.
2. Relays to control the electrical connections.
3. Arduino- to code the MCU.
4. Node Red as the pathway between client and MCU
5. AWS EC2 Ubuntu Engine as the Node Red server.
6. HIVEMQ as the broker for setting up a topic

NodeMCU- esp8266:

I used Arduino to code the MCU to:

- a. Connect to the home WIFI to access the internet.
- b. Send and receive data to the server using mqtt protocol.
- c. Use relays as per directed by the server.

I installed Node Red in the server using Node red's documentation and guidelines.

I used Node Red to:

- a. Facilitate the communication between client and MCU.
- b. Used mqtt protocol to send and receive data.
- c. Created a dashboard to control the turning on and off of appliances.
- d. Used Ngrok to convert a locally hosted website to a global website.

Equipment and tech stack used to add smart devices to the current framework:

1. Node Red
2. AWS EC2

I used AWS EC2 to:

- a. Read data from the smart devices.
- b. Send data to Node RED using HIVEMQ mqtt broker.

I used Node Red to:

- a. Read data from the smart devices using HiveMQ mqtt broker.
- b. Describe functions to do specifies tasks as assigned by user.

To demonstrate the working of this, let's say I have automated my windows to open automatically at 6:00 AM everyday and speakers to turn on and play music, AC to turn off beyond a certain temperature, say 15 deg C in room and to turn on, say 29 deg C at a certain temperature in and lights to go out in the morning.

Let's assume I have a smart thermostat that constantly updates the temperature in a specified website. Let's assume I don't have a smart AC or smart lights and I have a sound detector.

The following are the steps that would be performed using the setup:

1. Thermostat sends me data continuously using its own company website.
2. Data is being fed into Node Red in the cloud server.
3. Node Red is programmed in a such a way that once the said temperature is met, a certain signal is sent to the MCU.
4. Once the threshold reaches, say 29 deg C, Node red sends a signal to Node MCU to turn on the AC.
5. Similarly, at 6:00 in the morning when my speakers automatically start on, my sound detector picks it up.
6. It sends a signal to Node red in the server.
7. Node Red upon receiving the signal, is programmed in a way to send signal to the MCU to turn off the lights in my room.

References:

Attached a cpt file to demonstrate the working in a very rough way.

Attached the Arduino Code for the NodeMCU.

Attached the json files for the working of Node Red and deployment of dashboard.