Utilizing Alexa on Raspberry PI to Build Smart Home Automation System

Yuxiao Guo
yxguo@uw.edu

Mengting Weng
mtweng@uw.edu

Yu-Chieh Cheng
ycc91@uw.edu

Abstract

Home automation now is becoming more and more popular because of its numerous advantages. This target can be achieved by not only local networking but also remote control. Our project aims integrate Amazon Alexa Skill with Raspberry PI to create the smart home automation system. We use AWS lambda as our server and AWS IOT core to collect data from Raspberry Pi. Also, we store our data and create visualization UI with IBM Cloud.

Keywords: *Home Automation, Raspberry PI, Alexa Skill Kit, AWS Lambda, AWS IOT, Node-RED, IBM Cloud, Cloudant database.*

1 Introduction

Home automation is building automation for a home and the system will control lighting, climate, entertainment systems, and appliance[1]. Home automation application varies from simple remote control of lighting to complex micro-controller based on networks involving varying degrees of intelligence and automation[2]. The benefits of home automation typically include security, energy efficiency, convenience, comfort, save money and control.

The popularity of network and cloud enabled home automation become a dominating presence for the past several year, along with big company names, such as Amazon, Microsoft, Google, GE, IBM and Apple. Additionally, experts predict that smart home's market share will be worth tens of billions within the next few years[3].

Our project try to utilizing Alexa on Raspberry PI to home automation system and all our code are running in cloud.This project is to present a simulation of home automation system. It includes a Raspberry Pi, Grove Pi, a temperature/ humidity sensor, a LED, and an Amazon Echo/Dot.

We implemented our smart home automation system using Alexa Skills kit, AWS lambda, AWS IoT, IBM Watson, Raspberry Pi/Grove Pi and Node-RED. Specifically, we have used Amazon Alexa to control the light turn on/off or get the current temperature and humidity which collected by GrovePi sensor. Additionally, we store the data in IBM cloud and we try to analyze the data to get the user's pattern.

2 Architecture Design

2.1 Overall

The architecture of our design is as Figure 1 shows, we use Alexa to get the user request through voice. Then the Alexa Skills Kit will parse the intents to AWS Lambda function after it gets the request. AWS Lambda access the resource of AWS IOT and the corresponding topic will be published. On the other hand, Node-red help the communication between AWS IOT and Raspberry PI. For the database, we store the room temperature, humidity and LED on/off status in IBM NoSQL database Cloudant. The time interval for catching temperature and humidity is 1 minute.

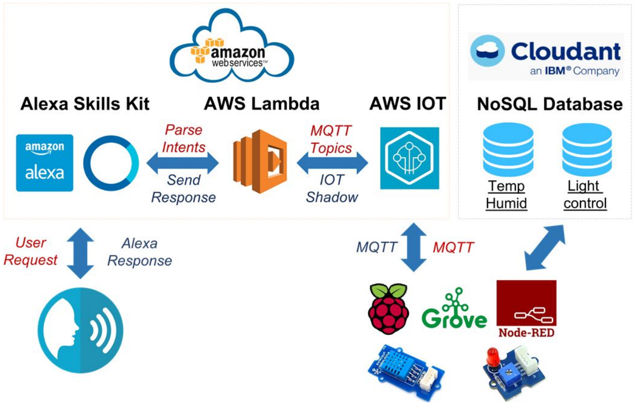


Figure 1. Overall architecture design

2.2 Raspberry PI

Raspberry Pi (as shown in Figure 2) is a credit-card-sized single-board computer developed in the UK by Raspberry Pi foundation with the intention of simulating the teaching of basic computer science in schools[4]. And GrovePi is a electronic board that we can connect to hundreds of different sensors, so that we can program them to monitor, control and automate devices. For our project, the sensor we used is mainly the LED and temperature/humidity sensor.

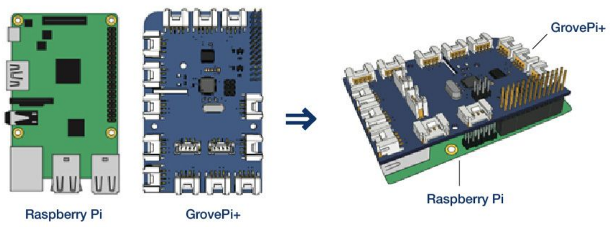


Figure 2. Raspberry Pi and GrovePi+

2.3 Alexa Skills Kit

The Alexa Skills Kit (ASK) is a collection of self-service APIs, tools, documentation, and code samples that makes it fast and easy for us to create customized skill to Alexa. To create a skill, we need to define the invocation name, intents, sample utterance firstly. For next step, build interaction model and set the web service endpoint to handle skill request. Then we can use Alexa Simulator to give Alexa commands to test our skill.

For our project, Table 1 shows the command we use to control LED, temperature and humidity. After the user input the voice command, Alexa Skills Kit will send out request to AWS lambda function. Also. if the Alexa Skills Kit receive the response from AWS lambda, it will be turn into voice response.

Alexa Command	Function
Ask Raspberry to turn LED on.	Turn on LED
Ask Raspberry to turn LED off.	Turn of LED
Ask sensor the room temperature.	Report the current room temperature
Ask sensor the room humidity.	Report the current room humidity
Ask sensor the room condition.	Report the current room temperature and humidity

Table 1. Alexa skill and function

2.4 AWS Lambda

The Alexa Skill Kit provides several samples of custom skills written in Node.js and Java. So an easy way to get started working with it is to deploy and test them as AWS lambda functions on AWS Lambda. AWS Lambda is an event-driven, serverless computing platform provided by Amazon Web Service as a part of the Amazon Web Service. It is a compute service that runs code in response to events and automatically manages the compute resource required by that code.

In our project, as Figure 3 shows, the trigger of AWS Lambda is the Alexa Skills Kit and the resource we connected with Lambda is AWS IoT. For the Lambda function, we use Node.js to write the code to work as our server. When the user talk to Alexa, a request is sent to the Lambda function and it interprets this Alexa JSON request and publishes the request to corresponding AWS IoT topic through MQTT.

Specially, for the LED control system, we just send the LED control signal to Raspberry PI through the IoT topic. For the temperature and humidity control system, Lambda function will retrieved the up-to-date temperature and humidity from the IoT shadow and push it to Alexa Skills Kit.

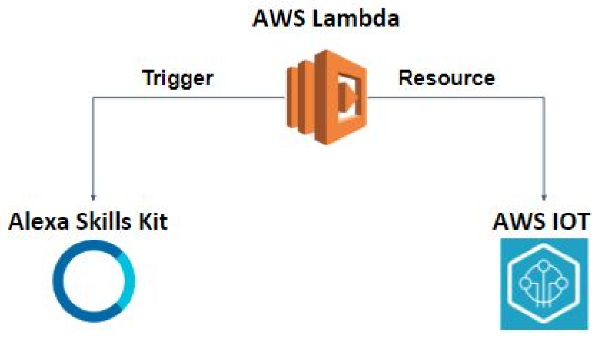


Figure 3. AWS Lambda design

2.5 AWS IOT

AWS IoT is a managed cloud platform that lets connected devices easily and securely interact with cloud applications and other devices. It is a great tool to support messages and process and route those messages to AWS endpoints and to other devices[5]. So for our project, we create two things in the AWS IoT platform. For is for LED control and the other one is for temperature and humidity control. For LED control, we just need to publish the control single (1/0) to Raspberry Pi to control turning on or off the light. For temperature and humidity control, upon GetTemperature or GetHumidity request, Raspberry Pi retrieves the latest data from database and push it into AWS IoT Shadow as JSON format. And Lambda will send execution response or data retrieved from IoT Shadow to Alexa Skills Kit.

2.6 Node-RED

Node-RED is a programming tool for wiring together hardware devices, APIs and online services. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed. The light-weight runtime is built on Node.js. As all the benefits of Node-RED, We found this tool the best choice to deploy our project.

To integrate every nodes in the project, we chose Node-RED to connect all the nodes together (as shown in Figure 4, 5, 6). Once finished the flows, a click on the “Deploy” button will run all the opening flows.

Specifically, for the LED control and temperature or humidity control, we use Node-RED as the tool to interact between AWS IoT and Raspberry Pi. Figure 4 shows the workflow for LED control. For the start point, we connect with AWS IoT and the control signal sends to Raspberry Pi and the database in IBM cloud to record the data for LED status change. Additionally, Figure 6 shows the workflow for temperature and humidity control, when the start point (AWS Iot) sends out the request for temperature and humidity, the data from database is pushed into the readTempData function and the newest data is transfer into the Json format and send to AWS IoT shadow. So the Lambda function can get the temperature and humidity from that shadow.

We also use Node-RED for updating database. As Figure 6 shows, the temperature and humidity data is caught from the Raspberry Pi and be sent to IBM Cloud database. The time interval for this process is one minutes. What's more, we create the visualization by showing two charts of temperature and humidity history recorded every minute.

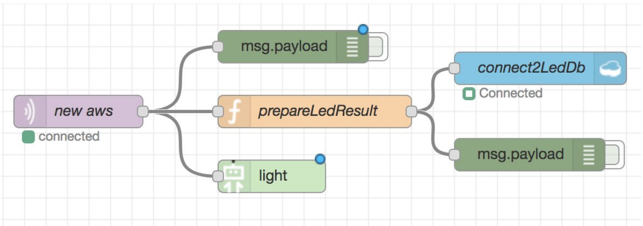


Figure 4. Node-RED workflow for LED control

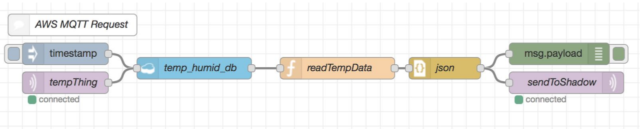


Figure 5. Node-RED workflow for temperature and humidity control

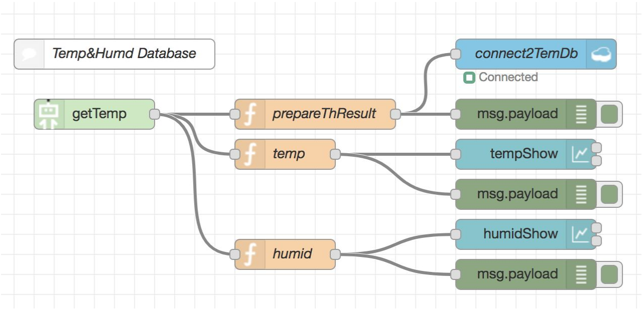


Figure 6. Node-RED workflow for database

2.7 IBM Cloud

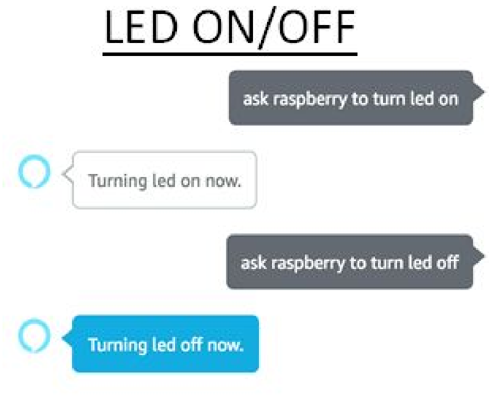
IBM Cloudant is a NoSQL database which offers data access in the flexible JSON format. As we are using free edition, this plan limits us a maximum of 1 GB of data storage. For this project, we use IBM Cloudant to store the data of temperature, humidity and the time users turn on/off the lights.

3 Results

3.1 Alexa Skill Test

We test the Alexa Skill on the Alexa Simulator. Figure 7 shows the test for LED control, when we input the voice request "asl Raspberry to turn led on", the LED connected with Raspberry Pi will be turned on and Alexa will return the response as "Turning led on now". The turning off led function is similar with turning on function.

Also, we test the temperature and humidity function with Alexa. Figure 8 shows the test result, when we ask for the temperature/humidity/condition, the Alexa will return the current temperature and humidity.



LED ON/OFF

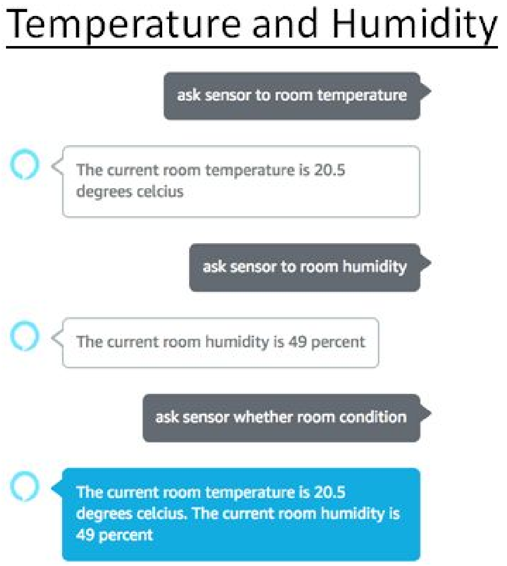
ask raspberry to turn led on

Turning led on now.

ask raspberry to turn led off

Turning led off now.

Figure 7. Alexa test for LED control



Temperature and Humidity

ask sensor to room temperature

The current room temperature is 20.5 degrees celcius

ask sensor to room humidity

The current room humidity is 49 percent

ask sensor whether room condition

The current room temperature is 20.5 degrees celcius. The current room humidity is 49 percent

Figure 8. Alexa test for temperature and humidity control

3.2 Visualization

In order to check the data, we made a user interface by using Node-RED UI. As seen in Figure 9 and 10, temperature and humidity records are updated every minute and shown as two charts. The range of temperature is set from 0 to 50 while the range of humidity is from 0 to 100.

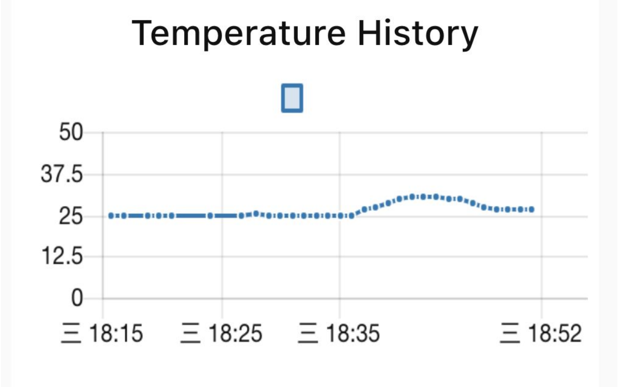


Figure 9. Visualization of temperature

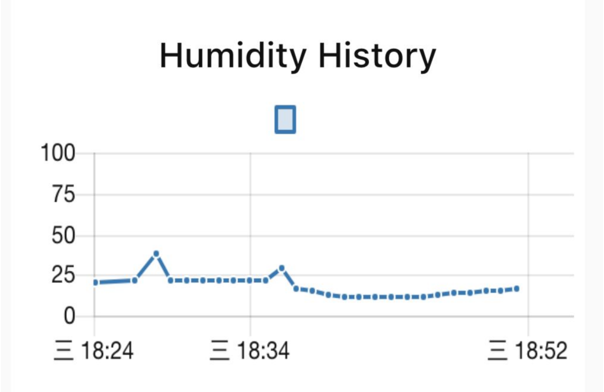


Figure 10. Visualization of humidity

4. Future Work

In the future, to further improve this idea, we will connect the Alexa skill to real Amazon Alexa devices as Echo or Echo dot. Secondly, we are now using LED to represent home lights. We will connect real home lights to the system instead of using LEDs.

Last but not least, for now we have saved the data in a cloud database. We are able to use these data to do further analysis for energy conservation. For example, according to the database, we can get the average time of when people are at home and when are not. If the user forgets to turn off the lights after he/she goes out for work, the system will turn off the lights automatically based on the data.

5. Conclusions

In summary, home automation application varies from lighting, climate, entertainment systems, and appliance. This project is a simulation of a home automation system. In this project, we use hardwares including a Amazon Alexa Dot, a Raspberry Pi, a Grove Pi, a LED, and a temperature/humidity sensor. In software, we use AWS Lambda and AWS IoT to connect Amazon Alexa with Raspberry Pi. On the other hand, we connect Raspberry Pi and IBM Cloudant NoSQL database. We also use Node-RED to deploy the whole system. Thus, we can control the LED and get the room condition by voice commands. After getting commands, we save the result in NoSQL database for future query.

References

[1] Wikipedia contributors. (2018, May 28). Home automation. In *Wikipedia, The Free Encyclopedia*. Retrieved 19:42, June 2, 2018, from https://en.wikipedia.org/w/index.php?title=Home_automation&oldid=843291997

[2] Jain, Sarthak, Anant Vaibhav, and Lovely Goyal. "Raspberry Pi based interactive home automation system through E-mail." *Optimization, Reliability, and Information Technology (ICROIT), 2014 International Conference on*. IEEE, 2014.

[3] Smart Home Market by Product (Lighting Control, Security & Access Control, HVAC, Entertainment & Other Control, Home Healthcare, Smart Kitchen, and Home Appliances), Software & Service (Behavioral, Proactive), and Geography - Global Forecast to 2023 from https://www.marketsandmarkets.com/Market-Reports/smart-homes-and-assisted-living-advanced-technologie-and-global-market-121.html

[4] Upton, Eben, and Gareth Halfacree. *Raspberry Pi user guide*. John Wiley & Sons, 2014.

[5] Guth, Jasmin, et al. "Comparison of IoT platform architectures: A field study based on a reference architecture." *Cloudification of the Internet of Things (CIoT)*. IEEE, 2016.

[6] Aravindan, Vignesh, and Darshan James. "Smart homes using Internet of Things." *Int. Res. J. Eng. Technol* 4.4 (2017): 1725-1729.

[7] Chandrasekaran, Vishal, Gautam Somappa, and K. S. Sivaraman. "peepHole: Securing your homes with Raspberry Pi and Amazon Echo."

[8] Rajalakshmi, Anoja, and Hamid Shahnasser. "Internet of Things using Node-Red and alexa." *Communications and Information Technologies (ISCIT), 2017 17th International Symposium on*. IEEE, 2017.