

DevNet Workshop

Extreme automation: Meraki APIs and the Alexa virtual assistant.

Session ID

DEVNET-2129

Speakers:

Juan Antonio Castilleja

Uriel Arriaga Alvarado

Learning Objectives

Upon completion of this lab, you will be able to:

- Understand the flow of Alexa request
- How to create python code into Lambda
- Create new skills to call personalized code

Scenario

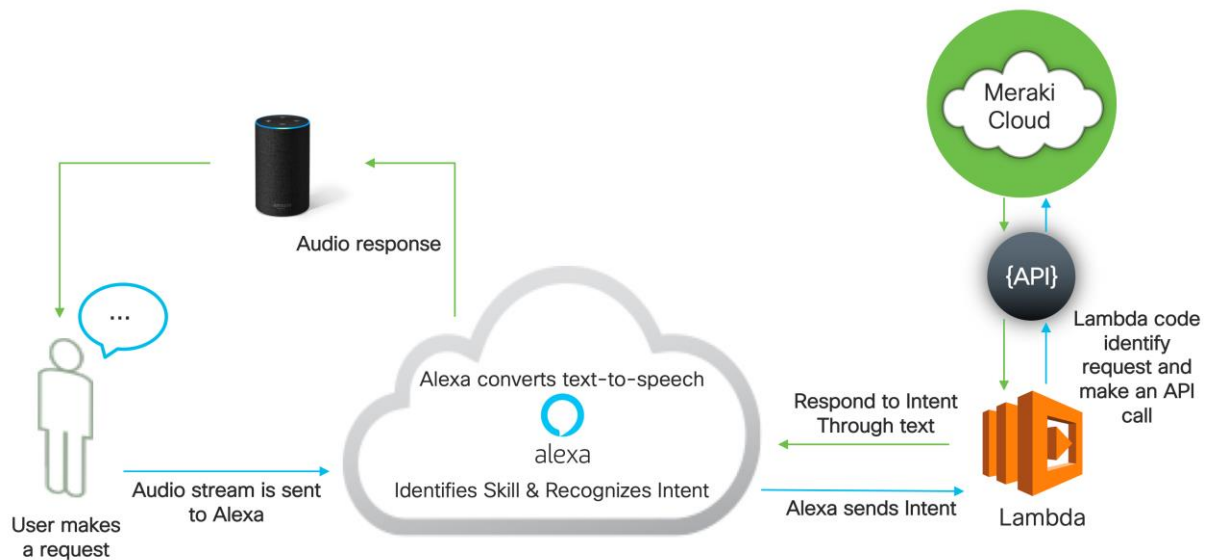


Figure 1. Lab scenario

Prerequisites

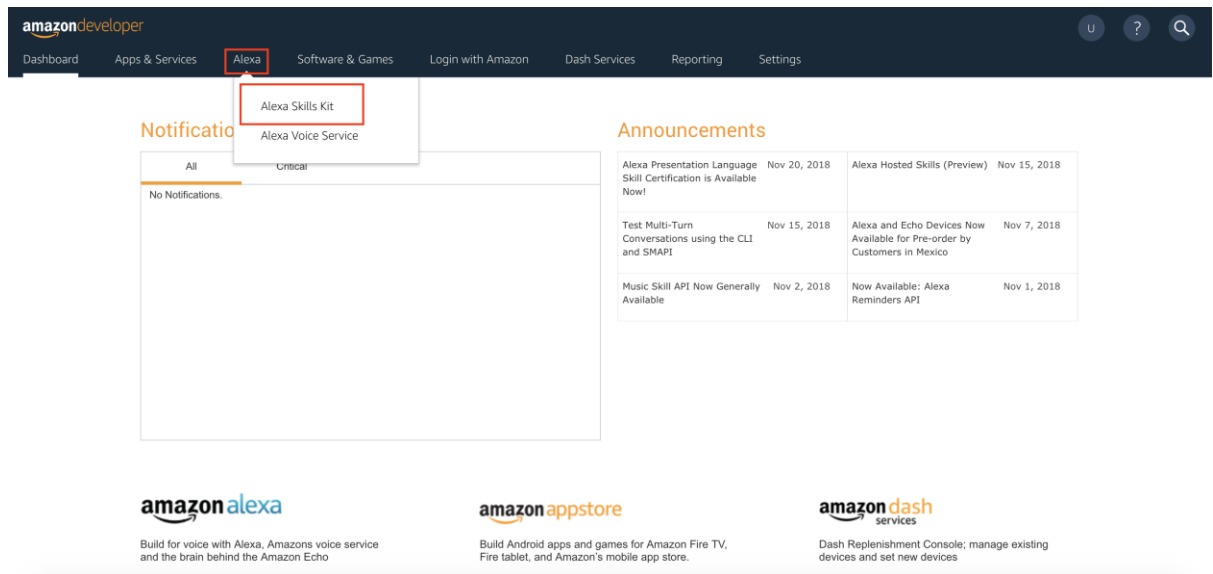
Have experience on:

1. Meraki APIs Knowledge
2. Basic Programing skills

Account details and logins will be provided during the lab.

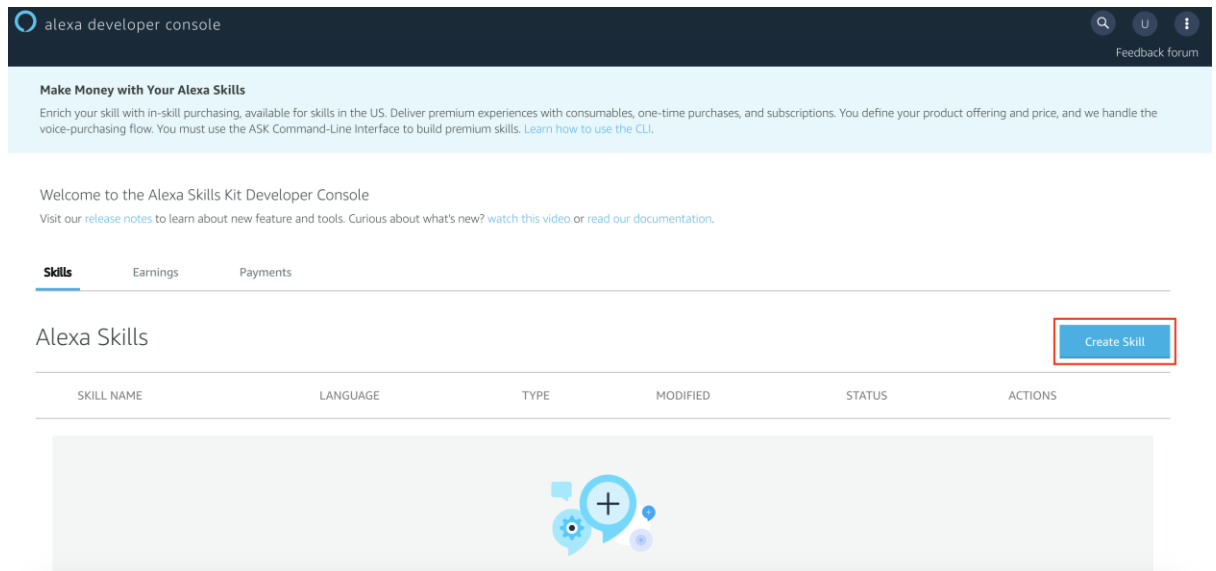
Task 1: Create an Alexa Skill

1. Go to developer.amazon.com
2. Click in "Developer Console"
3. Login as Prompted. Credentials will be given to you by the instructor.
4. Click on Alexa, then Click on Alexa Skill Kit



Announcement	Date	Announcement	Date
Alexa Presentation Language Skill Certification is Available Now!	Nov 20, 2018	Alexa Hosted Skills (Preview)	Nov 15, 2018
Test Multi-Turn Conversations using the CLI and SMAPI	Nov 15, 2018	Alexa and Echo Devices Now Available for Pre-order by Customers in Mexico	Nov 7, 2018
Music Skill API Now Generally Available	Nov 2, 2018	Now Available: Alexa Reminders API	Nov 1, 2018

5. Click the create skill button



Welcome to the Alexa Skills Kit Developer Console

Visit our [release notes](#) to learn about new feature and tools. Curious about what's new? [watch this video](#) or [read our documentation](#).

Skills | Earnings | Payments

Alexa Skills

[Create Skill](#)

SKILL NAME	LANGUAGE	TYPE	MODIFIED	STATUS	ACTIONS
					

6. Fill the “Skill Name” with “Meraki”, put the “Default language” to “English (US)” and select the “Custom” Option As a model for the skill. Then Click “Create skill”

Create a new skill

Cancel

Create skill

Skill name

Meraki

6/50 characters

Default language

English (US)

More languages can be added to your skill after creation

Choose a model to add to your skill

There are many ways to start building a skill. You can design your own custom model or start with a pre-built model. Pre-built models are interaction models that contain a package of intents and utterances that you can add to your skill.

Custom

Design a unique experience for your users. A custom model enables you to create all of your skill's interactions.

SELECTED

Flash Briefing

Give users control of their news feed. This pre-built model lets users control what updates they listen to.

"Alexa, what's in the news?"

Smart Home

Give users control of their smart home devices. This pre-built model lets users turn off the lights and other devices without getting up.

"Alexa, turn on the kitchen lights"

Video

Let users find and consume video content. This pre-built model supports content searches and content suggestions.

"Alexa, play Interstellar"

7. Choose “Start from Scratch” as a template, an click “choose”



Choose a template

Choose

Select a quick start template to get started with a predefined skill or simply "Start from scratch"

Start from scratch

Design a unique experience for your users and define your custom model from scratch.

SELECTED

Fact Skill

Provided a list of interesting facts about a topic, Alexa will select a fact at random and tell it to the user when the skill is invoked. Includes 1 custom intent, and 4 built-in intents.

Quiz Game Skill

Provided a list of interesting facts about a topic, Alexa will quiz a user with facts from the list. Includes 1 custom intent with 1 slot, and 6 built-in intents.

High-Low Game Skill

Try to guess the target number. Alexa tells the player if the target number is higher or lower than their current guess. Includes 2 custom intents with 5 slots, and 5 built-in intents.

English

© 2010 - 2018 Amazon.com, Inc. or its affiliates. All Rights Reserved. Terms Docs Forums Blog Alexa Developer Home



8. Click "JSON Editor"

The screenshot shows the Alexa Skills Kit Developer Console. In the left sidebar, under the 'CUSTOM' section, the 'JSON Editor' tab is highlighted with a red box. The main content area is titled 'How to get started' and features a video player for 'Amazon Alexa Developer Console: Build'. To the right, a 'Skill builder checklist' is visible, listing four steps: 1. Invocation Name (checked), 2. Intents, Samples, and Slots (checked), 3. Build Model (checked), and 4. Endpoint (checked).

9. Copy and Paste the JSON Available in intentJSON.txt within <https://github.com/Momoyactly/Extreme-automation> and click "Save Model"

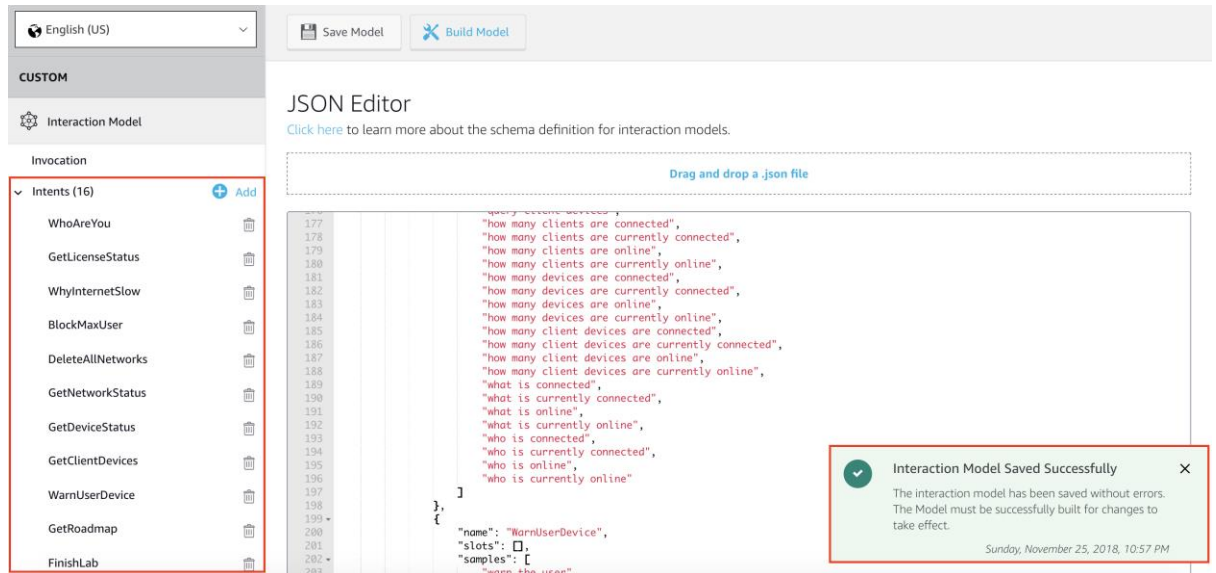
The screenshot shows the 'JSON Editor' tab in the Alexa Developer Console. A red box highlights the 'Save Model' button at the top. The main area displays the JSON code for the interaction model, which is pre-filled with sample intents. The JSON code is as follows:

```

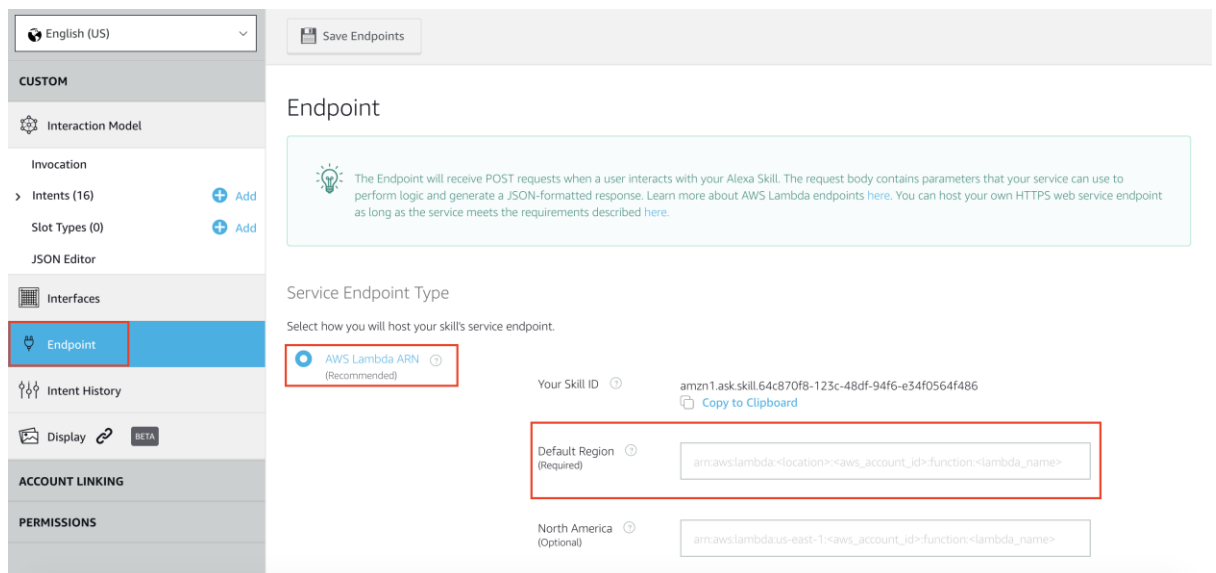
1 {
2   "interactionModel": {
3     "languageModel": {
4       "invocationName": "meraki",
5       "intents": [
6         {
7           "name": "AMAZON.FallbackIntent",
8           "samples": []
9         },
10        {
11          "name": "AMAZON.CancelIntent",
12          "samples": []
13        },
14        {
15          "name": "AMAZON.HelpIntent",
16          "samples": []
17        },
18        {
19          "name": "AMAZON.StopIntent",
20          "samples": []
21        },
22        {
23          "name": "AMAZON.NavigateHomeIntent",
24          "samples": []
25        }
26      ],
27      "types": []
28    }
29  }
30 }

```

10. if everything ins Ok you should see a green message in the inferior right corner, and in the left side the “Intent tree” Should look like in the image below. (Checkpoint)



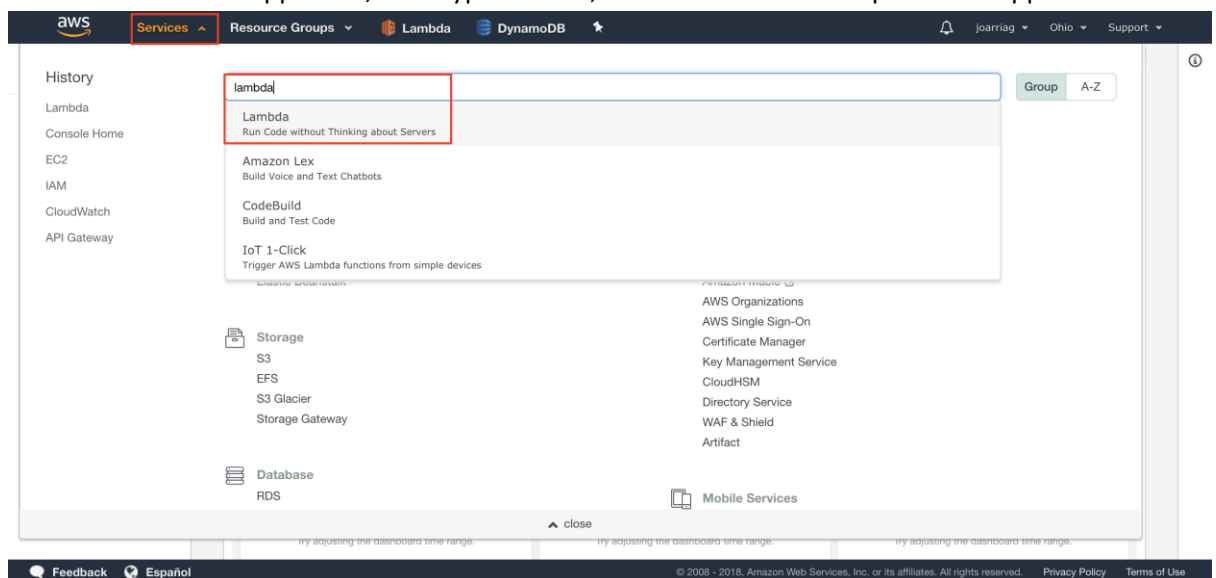
11. Click Endpoint in the right Panel, then click “AWS Lambda ARN” In the “Default Region” field we will copy the ARN given to us by Lambda in the next task, so we will come back to this Screen.



Task 2: Create the Lambda function

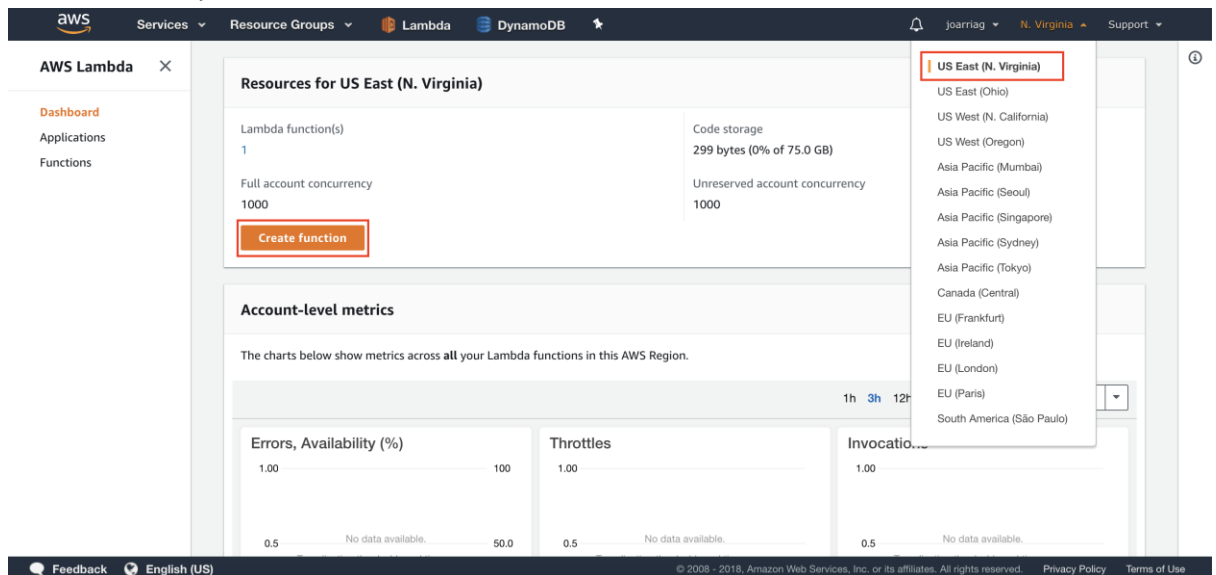
Nam nec lacus luctus, vestibulum ex non, convallis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ultrices lorem mauris

1. Login to AWS.amazon.com
2. Click Services in the upper bar, then type lambda, and click the lambda option that appears.

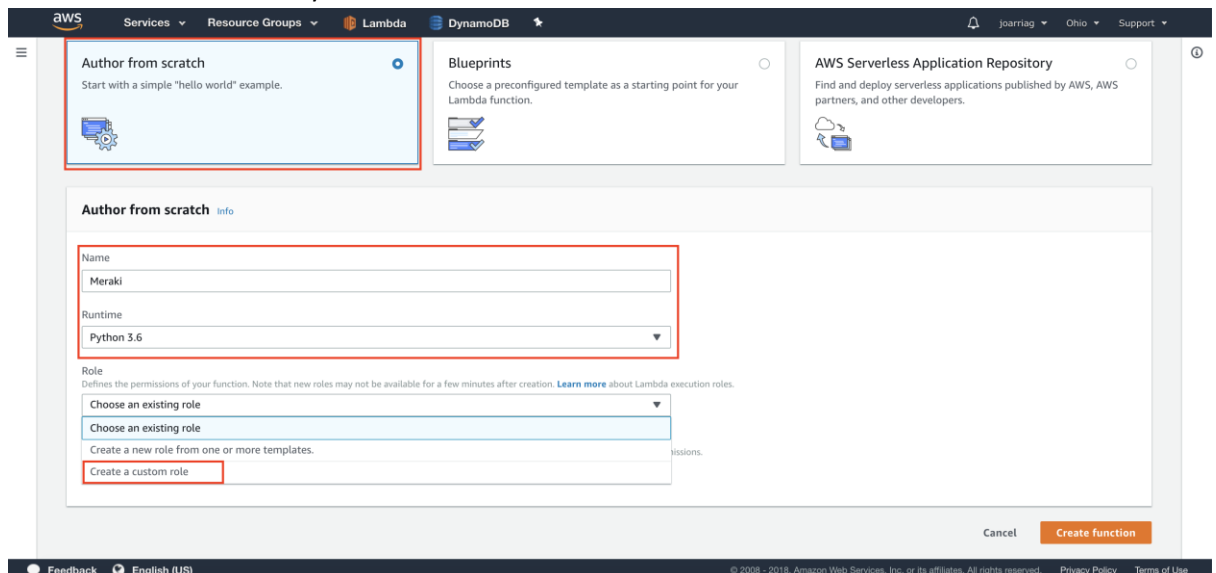




3. Make sure that you are in US East. Then click “Create a function”



4. Choose “Author from scratch”, write Meraki in the “Name” field, select “Python 3.6” in “Runtime”. Then click “Create a custom role” in the “Role” menu. You will be taken to another window and the you will come back.



5. Select a “Create a new IAM Role”, and write “lambda_basic_execution” in the “Role name”. Then click “Allow”

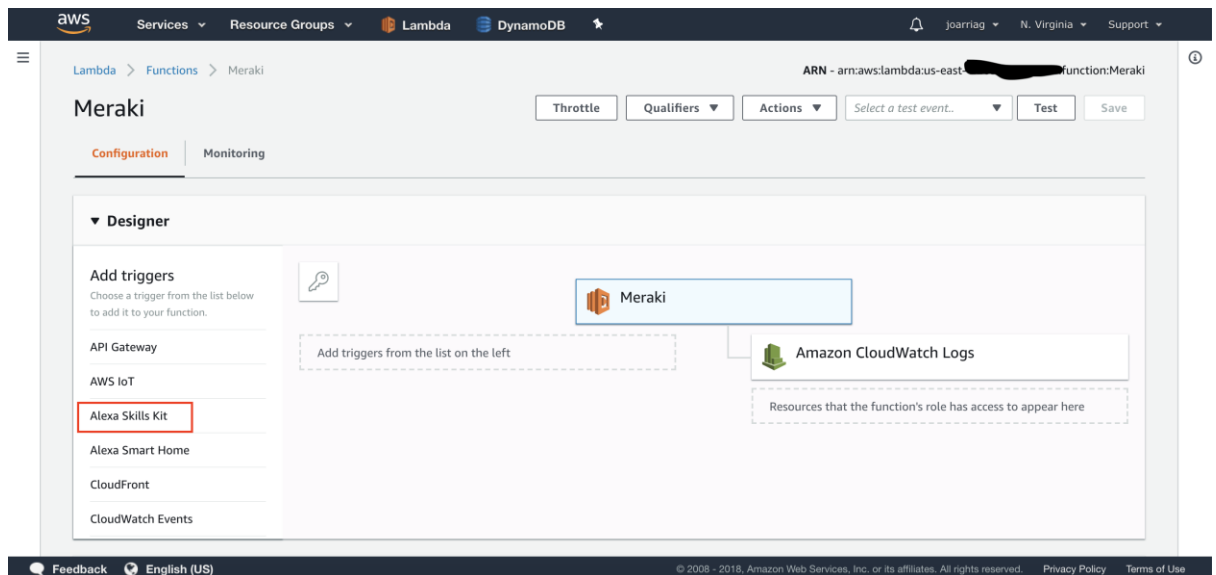
The screenshot shows the AWS IAM console interface. At the top, there's a navigation bar with 'aws', 'Services', 'Resource Groups', 'Lambda', and 'DynamoDB'. Below this, a message states 'AWS Lambda requires access to your resources'. The main section is titled 'Role Summary' and shows 'Role Description' as 'Lambda execution role permissions'. Under 'IAM Role', there's a dropdown menu set to 'Create a new IAM Role'. The 'Role Name' field is filled with 'lambda_basic_execution'. At the bottom right, there are 'Cancel' and 'Allow' buttons, with the 'Allow' button highlighted by a red rectangle.

6. Choose “Choose an existing role” in the “Role” menu, and choose “lambda_basic_execution” ,that we just created, from the “Existing role” menu. Then click “Create function”.

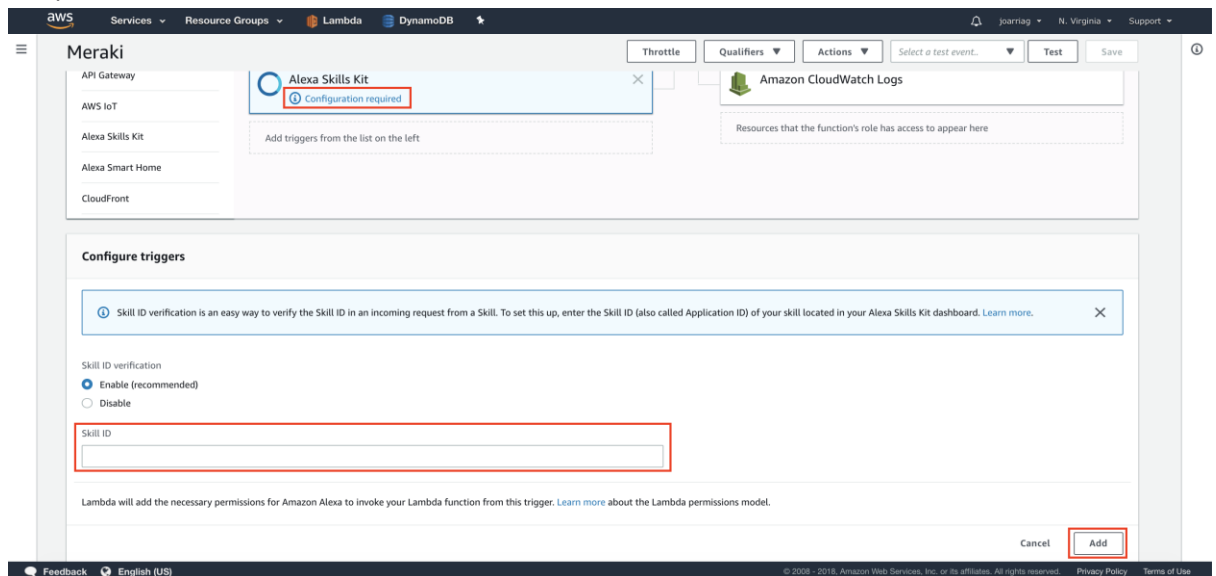
The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with 'aws', 'Services', 'Resource Groups', 'Lambda', and 'DynamoDB'. Below this, there are three tabs: 'Author from scratch', 'Blueprints', and 'AWS Serverless Application Repository'. The 'Author from scratch' tab is selected. In the 'Author from scratch' section, there's a form with fields for 'Name' (filled with 'Meraki'), 'Runtime' (set to 'Python 3.6'), and 'Role'. The 'Role' dropdown is set to 'Choose an existing role'. Below this, there's an 'Existing role' dropdown set to 'lambda_basic_execution'. At the bottom right, there are 'Cancel' and 'Create function' buttons, with the 'Create function' button highlighted by a red rectangle.



7. Click in “Alexa Skill Kit”

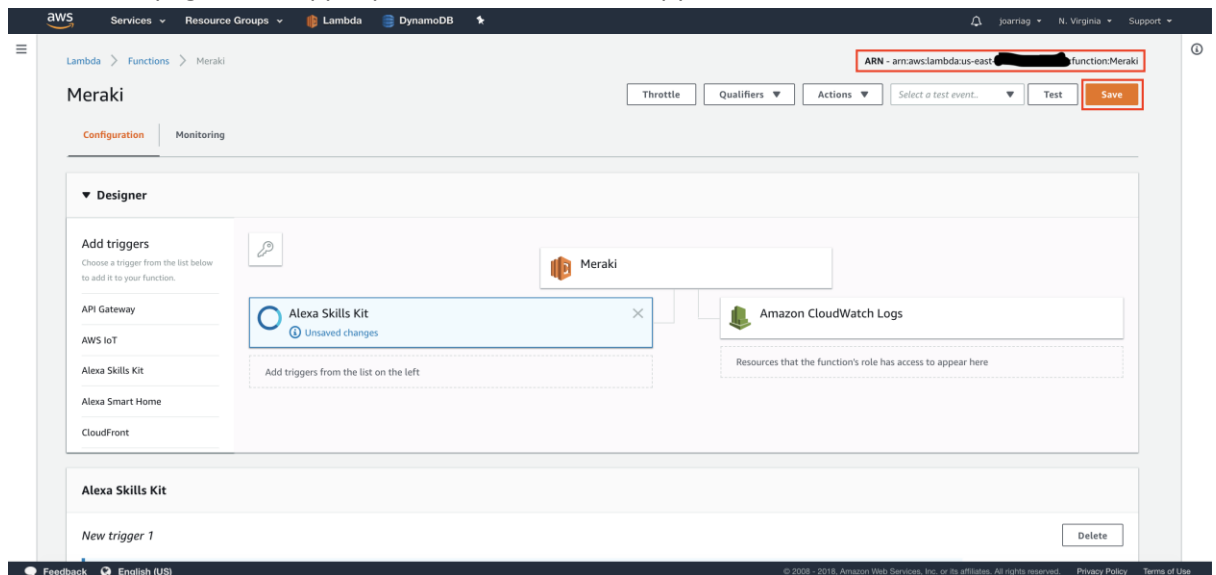


8. Click where it says “Configuration required” and the field “Skill ID” copy the value we get in the previous task. The click “ADD”

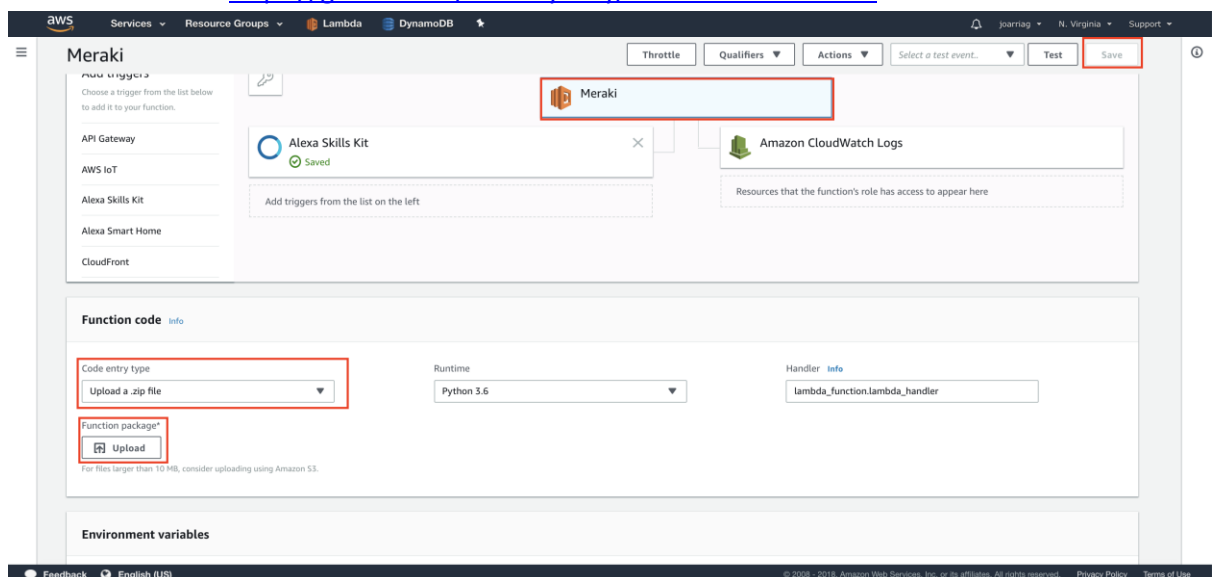




9. In the same page in the upper part, click “SAVE” the copy the ARN value



10. Click in the lambda button of “Meraki”, choose “Upload a .Zip file” and click Upload to add the code available in: <https://github.com/Momoyactly/Extreme-automation> Then click “Save”



Task 2: Integrate Alexa with Lambda

Nam nec lacus luctus, vestibulum ex non, convallis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ultrices lorem mauris

1. Return to developer.amazon.com in the menu “Endpoints” and paste the ARN value in the “Default Region” field

English (US) Save Endpoints

CUSTOM

Interaction Model

Invocation

Intents (16) + Add

- WhoAreYou
- GetLicenseStatus
- WhyInternetSlow
- BlockMaxUser
- DeleteAllNetworks
- GetNetworkStatus
- GetDeviceStatus
- GetClientDevices
- WarnUserDevice
- GetRoadmap
- FinishLab

Endpoint

The Endpoint will receive POST requests when a user interacts with your Alexa Skill. The request body contains parameters that your service can use to perform logic and generate a JSON-formatted response. Learn more about AWS Lambda endpoints [here](#). You can host your own HTTPS web service endpoint as long as the service meets the requirements described [here](#).

Service Endpoint Type

Select how you will host your skill's service endpoint.

☒ AWS Lambda ARN (Recommended)

Your Skill ID [Copy to Clipboard](#)

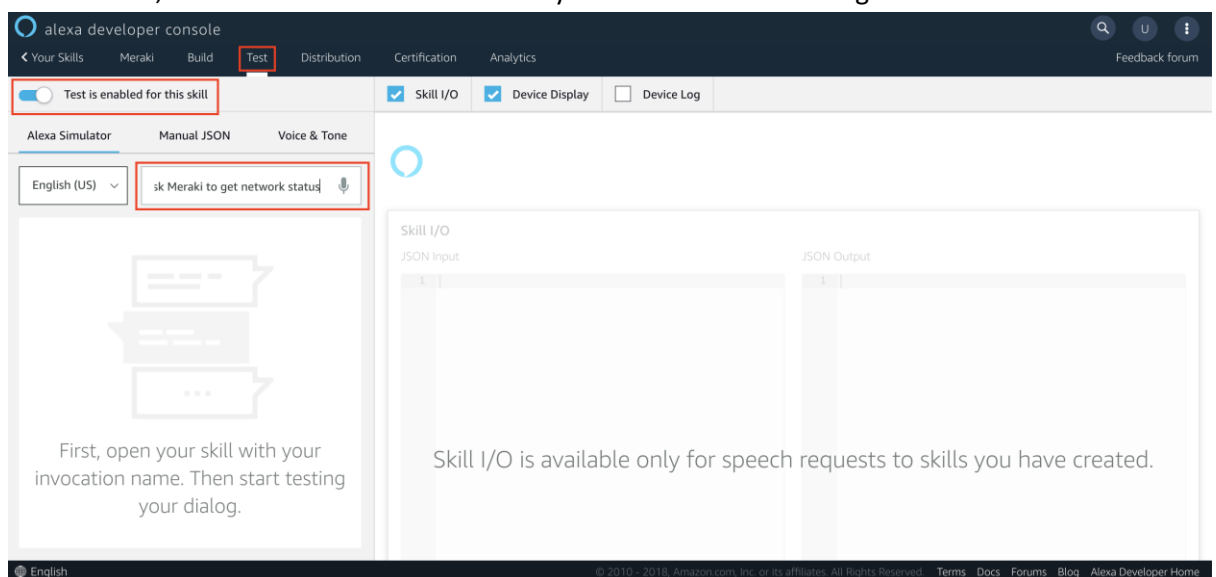
Default Region (Required)

North America (Optional)

Task 3: Test with Alexa!

Nam nec lacus luctus, vestibulum ex non, convallis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ultrices lorem mauris

1. Go to “Test”, “Enable test for this skill” and say or write “ask Meraki to get network status”



Challenge: Add a new skill to create a new network

All the documents mentioned are available in: <https://github.com/Momoyactly/Extreme-automation>

An updated version with the new skill is available in the folder newSkillVersion.

1. Modify the Json in the “Alexa Developer Console” with the code in jsonNewSkill.txt. Then build and save the skill
2. Add the method available in the updateLambdaFunction.txt in the lambda_Fuction.py in lambda. Then Save the function.
3. Modify the on_intent method in the lambda_Fuction.py in lambda. An example is available in updateLambdaFunction.txt. Then Save the function.
4. Try the new Skill.