|  |  |
| --- | --- |
| Amazon Alexa | Abstract  This document helps to create custom skills for Alexa. While building a custom skill, one creates an invocation name, intents, utterances and configure that to bring all together for Alexa to route requests to the service of skill. |

Introduction

“Amazon Alexa” (Alexa) is a virtual personal assistant, designed by Amazon’s secretive Lab126. Alexa is capable of listening our voice commands, process it and responds with contextual responses to get our job done. In this project, we are trying to build a small conversation with Alexa where a user asks Alexa to give the summary of certain blogs.

Contribution

People contributed in the making of this MVP is Manish Maheswaran, Manisha, Are Sudheer Babu and Vishal Kumar under the guidance of Venkata Vignesh, Prem and Gayathri N V S.

Requirements

* Amazon Developer account
* Amazon Web Services account

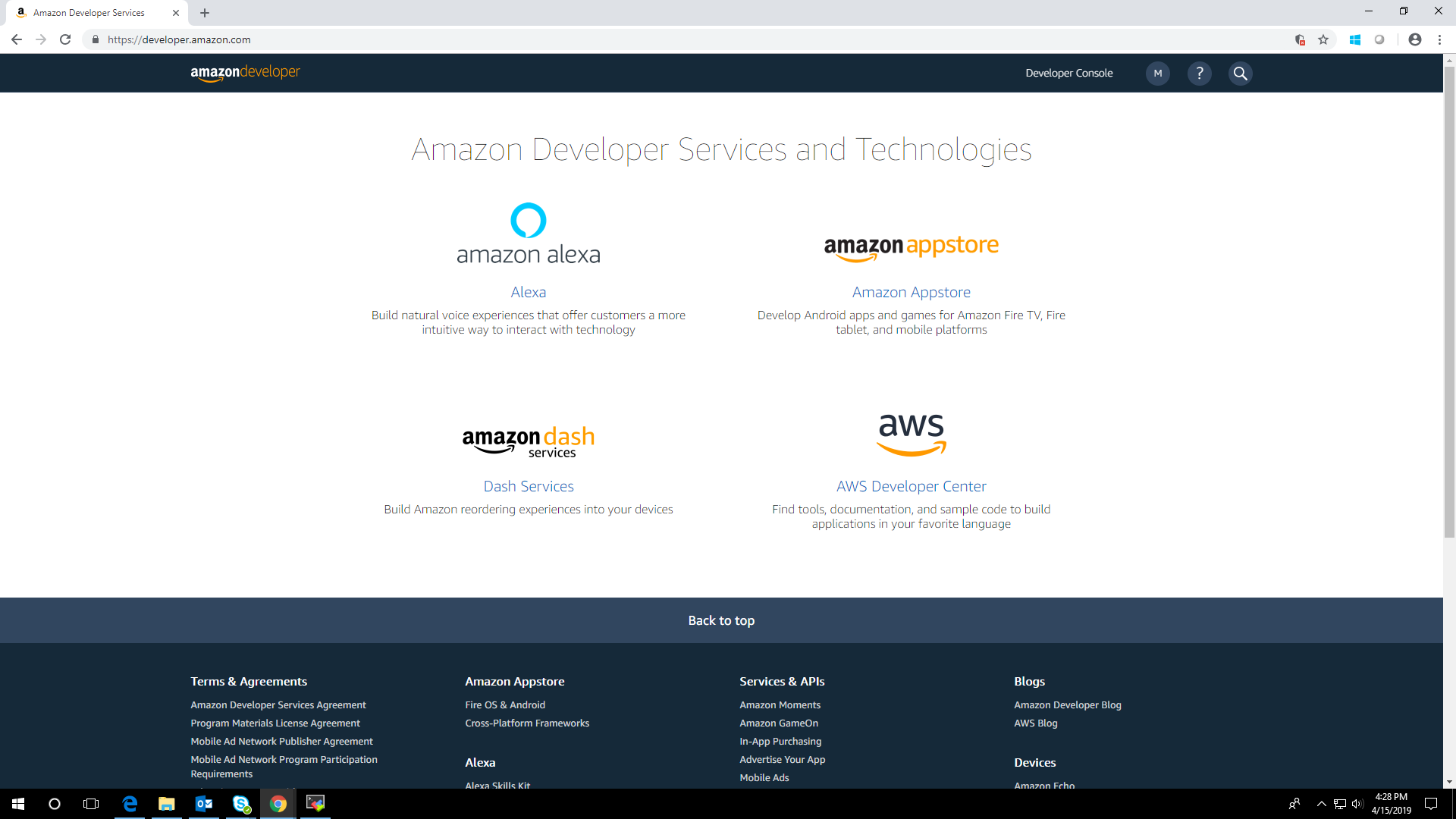
To build a small conversation application with Alexa, one must have an account on Amazon Developer and AWS Management Console. For this project, we have used python but one can use any language like Node.js, java, C# as per their convenience.

***Note***: Make sure to use the same account for both developer and for setting up alexa.

Setting up Accounts

* If one has an Amazon shopping account then he/she can access to <https://developer.amazon.com> and sign with same account otherwise he/she is required to create one at <https://www.amazon.com>.
* After signing it, fill the profile information and accept app’s distribution agreement. One will be directly taken to his/her Dashboard.

Amazon Developer Account

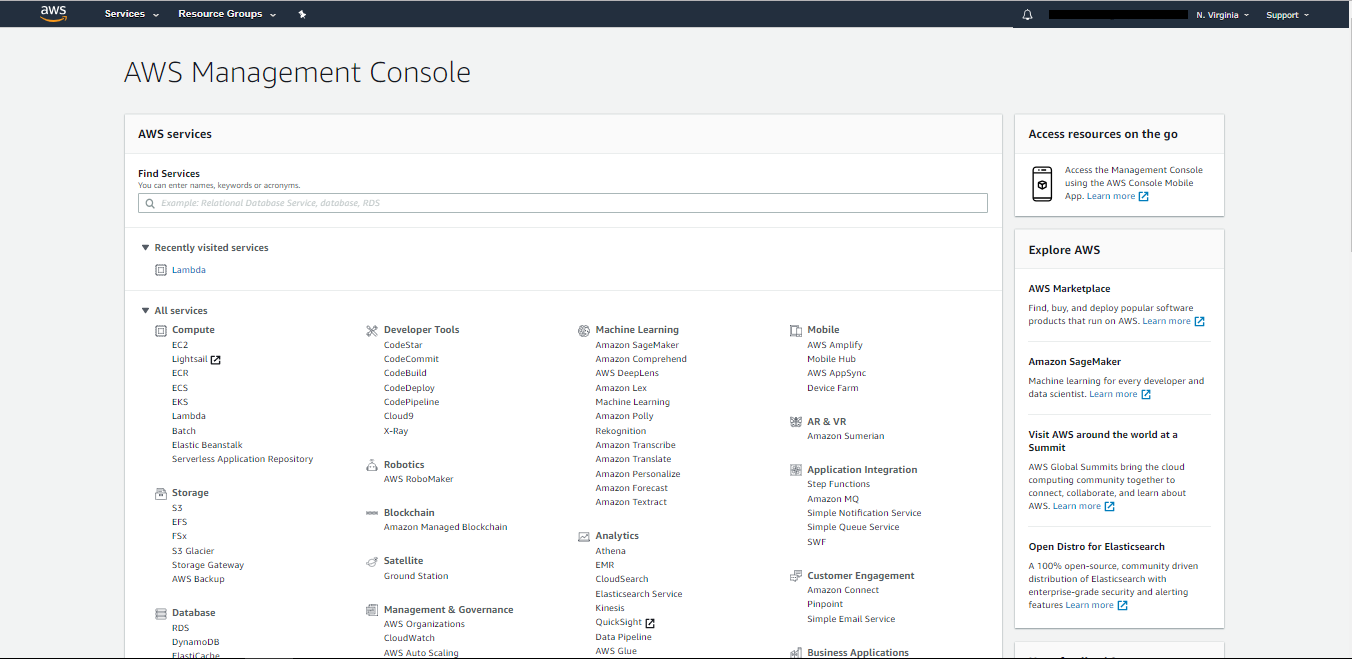


* Again, if one has AWS account then he/she can sign in and head over

to AWS Console otherwise first create one AWS account at <https://aws.amazon.com/console>.

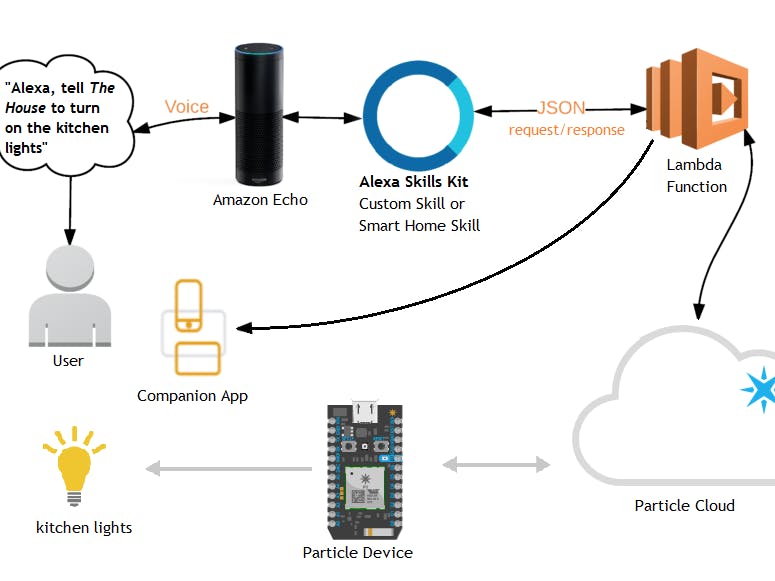
* One can create an AWS free account, but few things are limited.
* AWS Account Includes 12 Months of Free Tier Access
* Including use of Amazon EC2, Amazon S3, and Amazon DynamoDB
* For example, AWS Lambda has 1-million free requests per month.
* Visit [https://aws.amazon.com/free/](https://aws.amazon.com/free/%20)for full offer terms

AWS Management Console



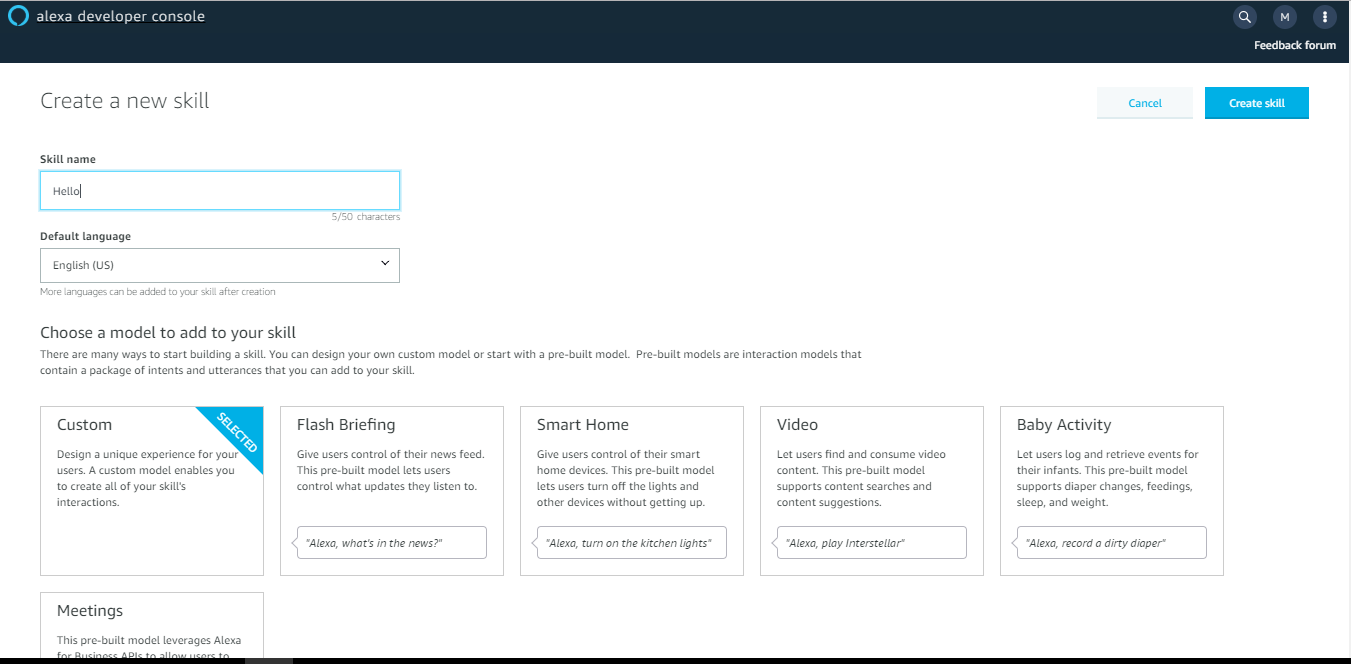
* AWS Management console have many services like EC2, lambda, IAM, Elastic Beanstalk and many more.
* Here, we will use lambda services to communicate with Alexa but one can choose any service on which he/she wants to work with (i.e Either Lambda or HTTPS).

Working Flow



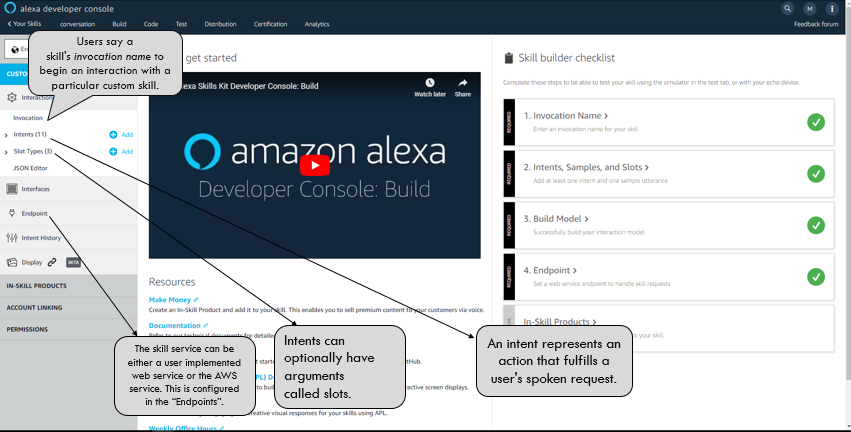
* In Amazon Developer Dashboard, access to Alexa Skills then create skills.
* To create a new skill, one need a Skill Name and a Language depending on skill one want to build.

Creating custom skill



* Choose a model to start building your skill. We usually go for custom interaction model because it is easy to understand the models and define the structures we need.
* We have different types of models like
* Custom
* Flash briefing
* Smart home
* Video
* Baby activity
* Meetings

Alexa Dashboard

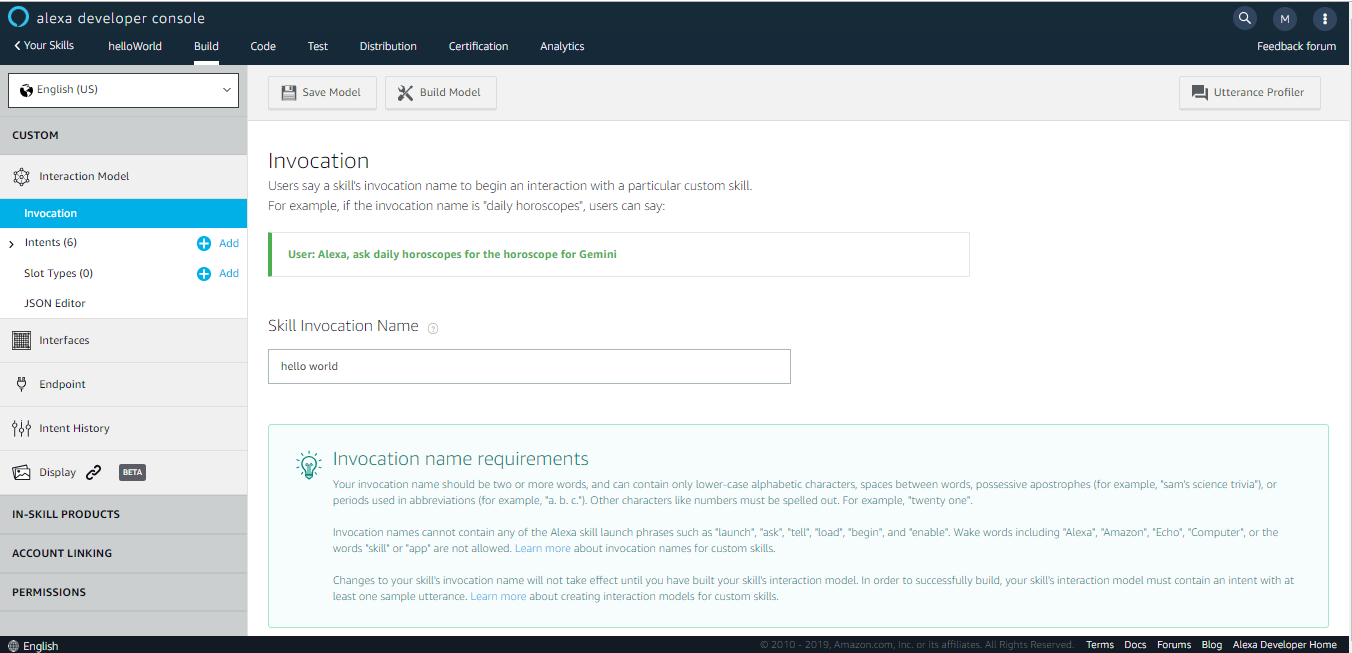


* To build the interaction model, only 4 steps are required.
* Find an Invocation Name
* Intents, Samples and Slots
* Build Model
* Endpoints

1. Creating Invocation Name

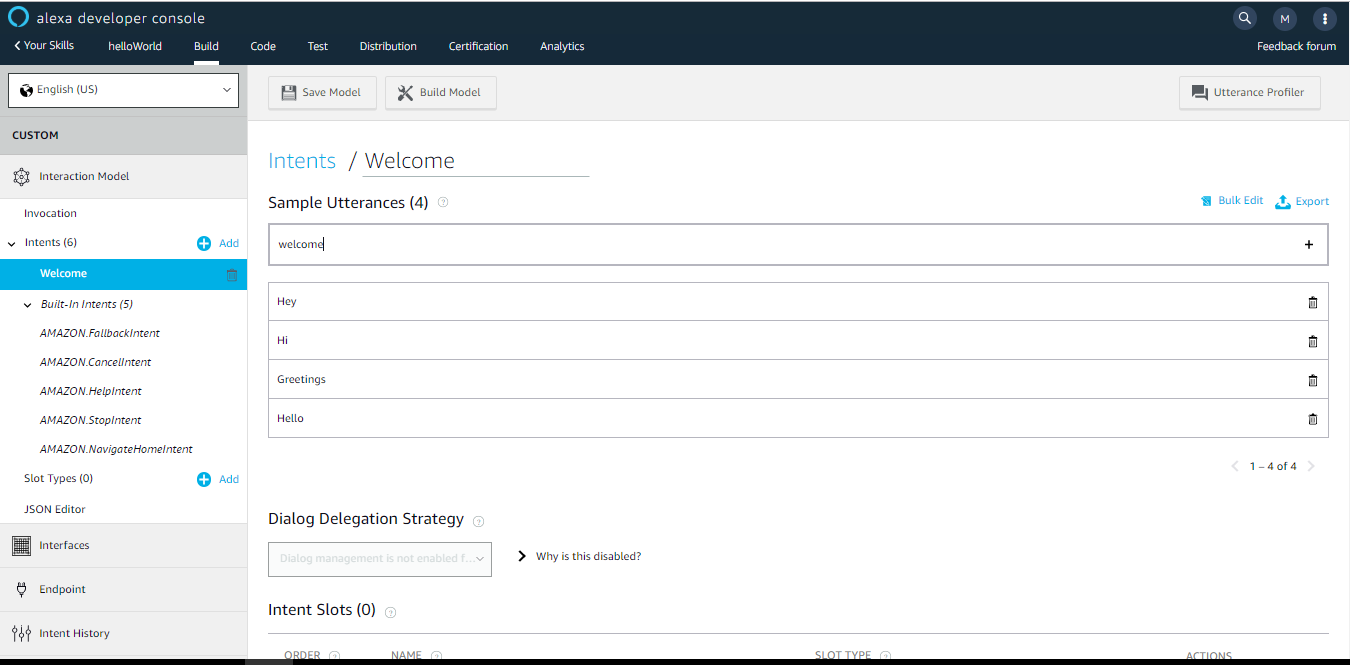
* Invocation Name: Invocation Name is the identifier used to trigger the Skill.

For example: if the invocation name is “hello world", then users can say: “Alexa open hello world.”



1. Assigning Utterances to Intent

* Intents, Samples and Slots:  User’s spoken input will be mapped to intents one defined in own Skill. They reflect the meaning of what user wants to say. Samples are a set of sentences mapped to an intent.
* They are used by The Natural Language Processing System to help Alexa learn user’s intents. Slots are types of items that are not covered by Amazon (entities).
* One can create own custom Intent and set up utterances or use the in-built intent such as: cancel, help or stop.

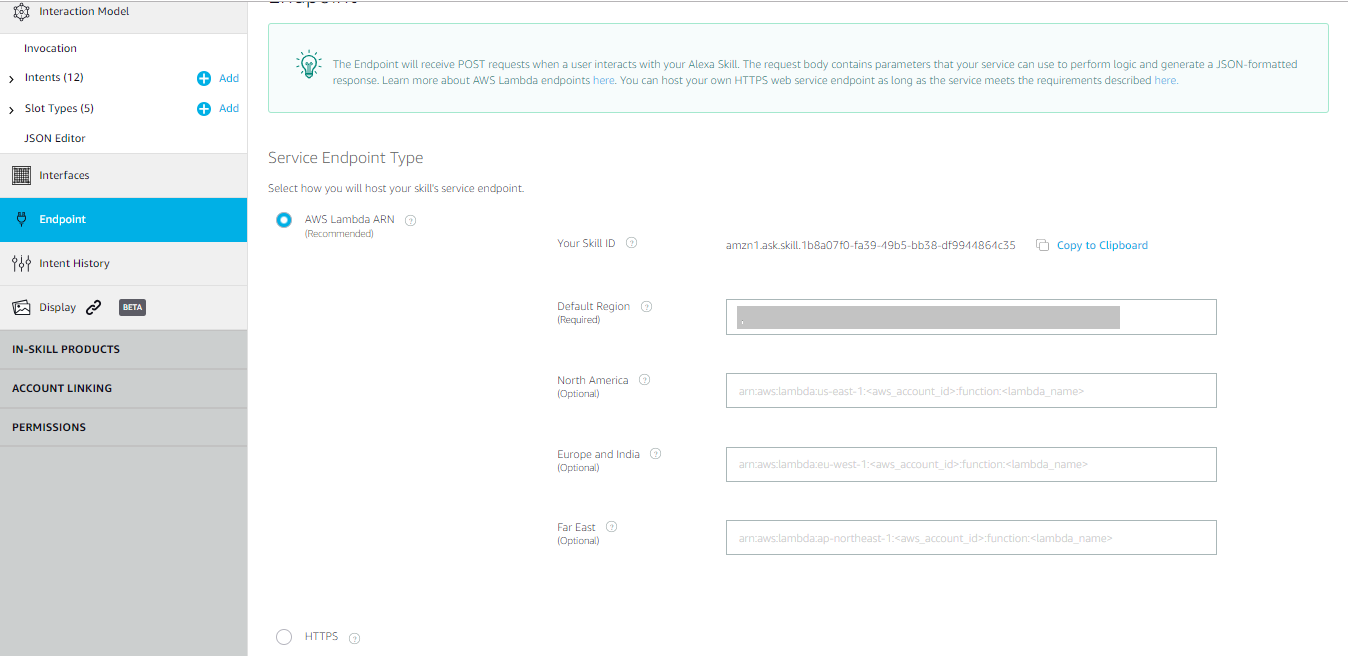


1. Build Model:

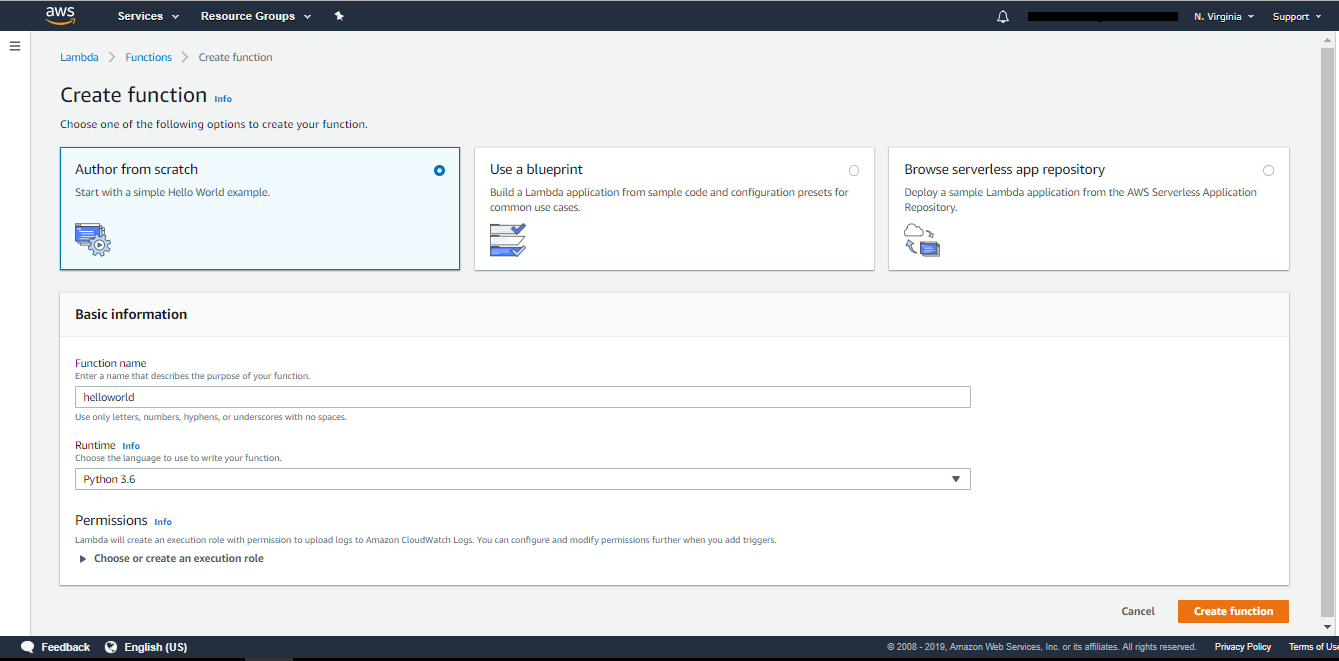
* Once done with creating the intent and giving the utterances save the model and Build it.
* Then it will give a Notification stating that the model is successfully build or if there is an error it will through an error
* If there is a problem in building the model, then check with the configurations that are configured properly or not.

1. Endpoints:

* The Endpoint will receive POST requests when a user interacts with Alexa Skill. For this we need to create Lambda Function to handle skill’s intent One can link his/her Skill with Lambda or can host endpoint using an HTTPS web service that one manages.



Creating Lambda Function:



* AWS Lambda is a compute service. It takes one’s code, runs it in AWS without requiring any servers’ provisioning or management and executes the code only when needed and scales automatically from a few requests per day to thousands per second. One pay only for the compute time he/she consumes - there is no charge when the code is not running. With AWS Lambda, one can run code for virtually any type of application or backend service - all with zero administration.
* Search for Lambda in AWS Management Console and create a new Lambda Function from scratch.
* In the Lambda function we are going to write a logic for each intent that we have created in the developer console.
* We can choose the platform whatever we need, they have provided some platforms like python, node.js, etc.

**Note:**Our Lambda Function will be triggered via the Alexa Skill Kit. Make sure that one is working on these regions as these components are available only in: US East (N. Virginia), US West (Oregon), EU (Ireland), Asia Pacific (Tokyo).

Identity Access Management:

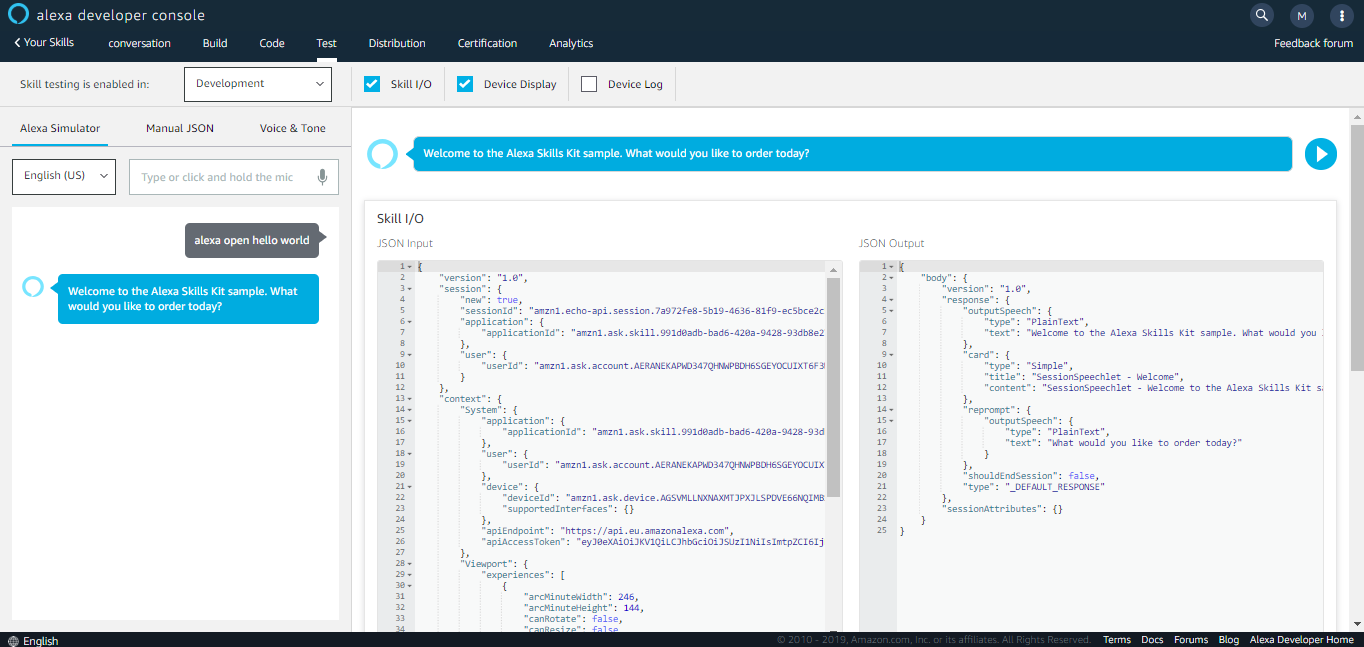
* An IAM role is an AWS Identity and Access Management (IAM) entity with permissions to make AWS service requests. IAM roles cannot make direct requests to AWS services; they are meant to be assumed by authorized entities, such as IAM users, applications, or AWS services such as EC2.
* For sample custom skill we can use the basic Lambda Execution role  
  (default role).
* Create roles
* Attach the polices (whatever the permission required for your skill according to the requirement.)

Configuring Skill Id and Lambda ARN:

* From the trigger, add Alexa Skills Kit and configure the Alexa skill id from the amazon developer account.
* Once the skill id is configured, copy the ARN id from AWS console and paste it in the endpoint services in the amazon developer account.
* Now that we set the endpoints, the connection between our Alexa Skill and Lambda Function should be fine.
* One can also build [server less](https://aws.amazon.com/serverless) applications composed of functions that are triggered by events and automatically deploy them using Code Pipeline and AWS Code Build.

Test:

* Go back to Alexa Skills Kit, and re-build the Model. Once done, Test the skill on the Test Tab.
* The developer can test the skill in the developer console itself.
* If it works fine, then it will display both JSON input and JSON output.



* The left side represents the JSON Input, what the user asks Alexa (Request).
* The right side represents the JSON output hitting from the lambda function (Respond).

**References**

* <https://developer.amazon.com/docs/custom-skills>
* <https://medium.com/@rania.zyane/amazon-alexa-build-a-new-skill-with-aws-lambda-to-automatically-build-a-circleci-bot-project-fb65835b7a47>
* https://www.youtube.com/watch?v=sj7NqS7yytw&pbjreload=10
* https://www.youtube.com/watch?v=-MOvJ3eklr0