Amazon Alexa: Building a skill

Conor Corcoran – 20067633

Advanced Media Technologies 2018

Introduction

Alexa is Amazon's virtual assistant device, designed to perform tasked such as music playback, setting alarms, providing weather updates and other voice interaction capabilities (Amazon, 2018). Alexa also allows for the use of third party applications such as Spotify and RTÉ News. On Alexa, these applications are called skills. This document will detail how to create an Alexa skill to playback an MP3 from an online source.

Requirements

The following is what is needed to create this skill:

- Amazon account
- Amazon Echo/ Dot
- Amazon Web Services account
- Amazon Developer account
- A Digital Audio Workstation (Audacity, Adobe Audition)
- Code Editor
- Node.js Installed on your PC

Development

Multiple components from different sources go into the creation of an Alexa skill. To start, log onto your Amazon Developer account.

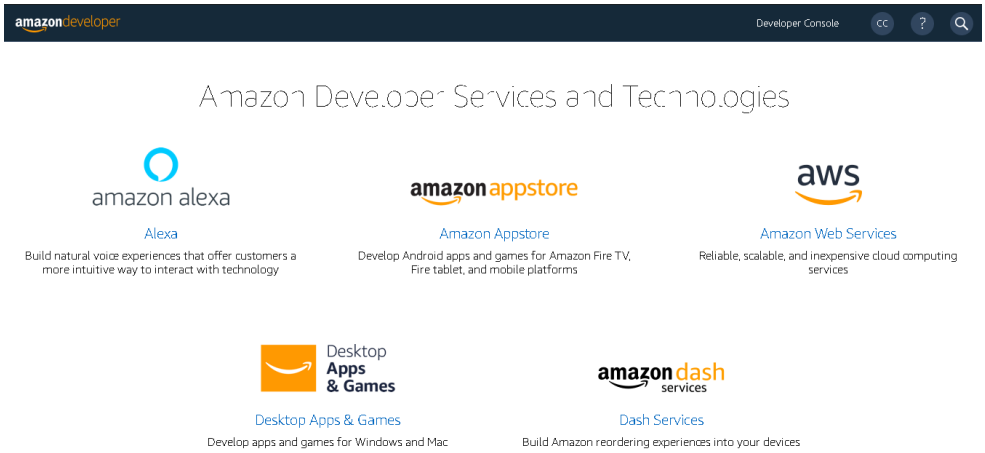

Figure 1 Amazon Developer front page

Once there, click into the 'amazon alexa' section and then go into 'Your Alexa Console' and 'Skills'.

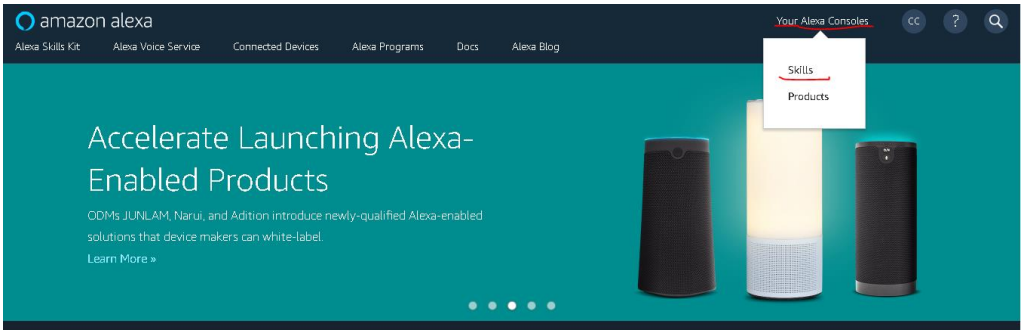

Figure 2 Amazon ALexa Developer Page

Click on the 'Create Skill' button and you will be brought through a series of steps to create your skill. First thing to do is to give your skill a name. For this example, I'll be making a skill for the radio station, Beat and will call it 'Beat News'.

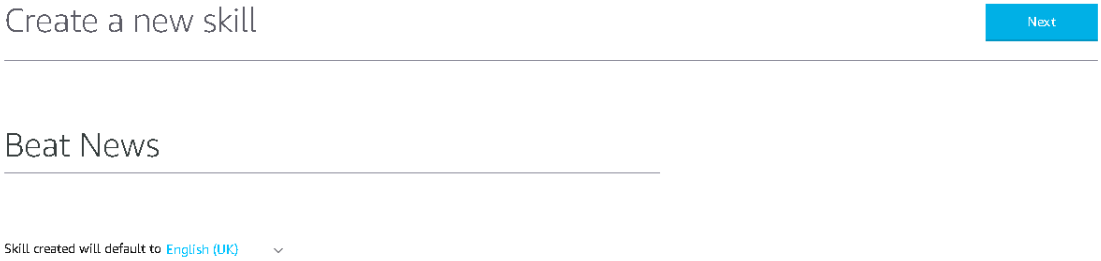

Figure 3 Naming the skill

Notice underneath the name the sentence 'Skill created will default to English (UK)'. This determines what language your skill is created in. For this, English has a few different options. It is important to have the same version of English set across the board. This means that if your Alexa is set to English (UK), you must set your skill to English (UK).

The next page will ask you to choose a model for your skill. This offers different templates for Skills such as Flash Briefings and Smart Home Skills. In this case, select Custom and then hit 'Create Skill'.

You will now be met with the build page. Here, you will be given a checklist of what must be done before you can test your skill. The first thing it will ask is to give your skill an invocation name. This is what you ask Alexa to do to open you app.

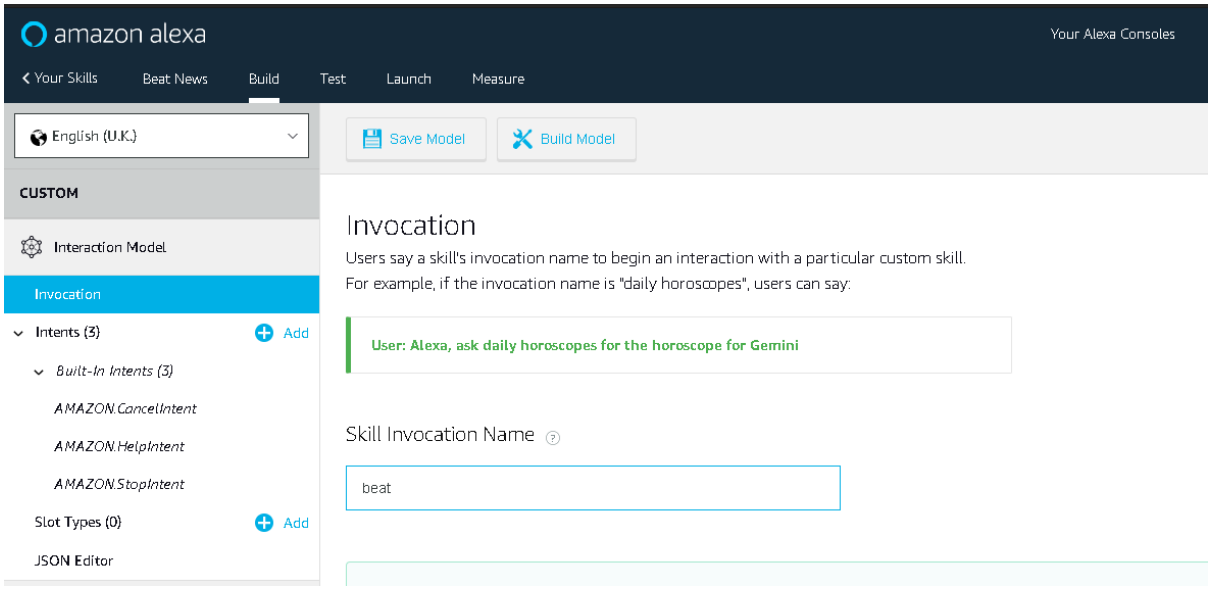

Figure 4 Invocation

It is important to think about how invocation name. Make sure it is not too hard for Alexa to understand. This may be discovered when testing. For more information on invocation name requirements, visit [Invocation help].

Next, add intents. An intent is like a function in programming. It is called on and whatever code is behind it will be run. In this case, we can add pre-made intents for audio playback. This is done from the 'Interfaces' tab.

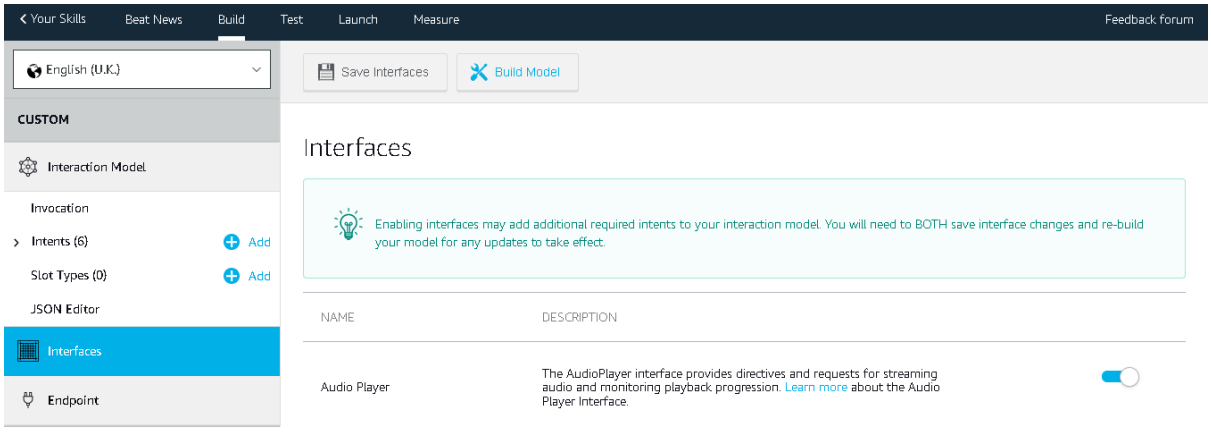

Figure 5 Interfaces tab

For the next step, setting the Endpoint of your will need to move over to AWS (Amazon Web Services). The endpoint of the Alexa skill can be set to a AWS Lambda, which is where the processing of your skill is done.

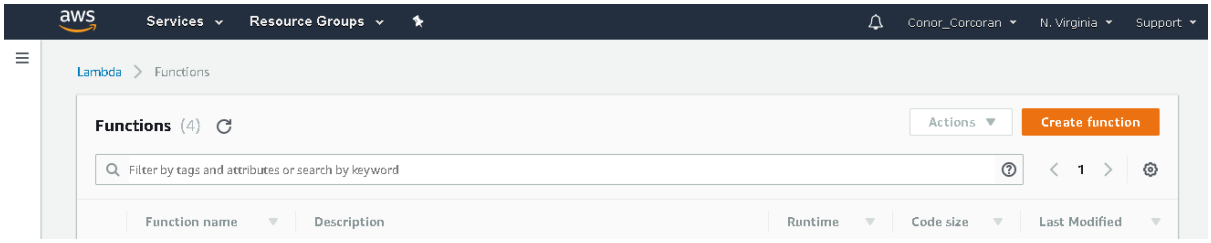

Figure 6 Creating a Lambda Function

On AWS, click into the Lambda section and then create function.

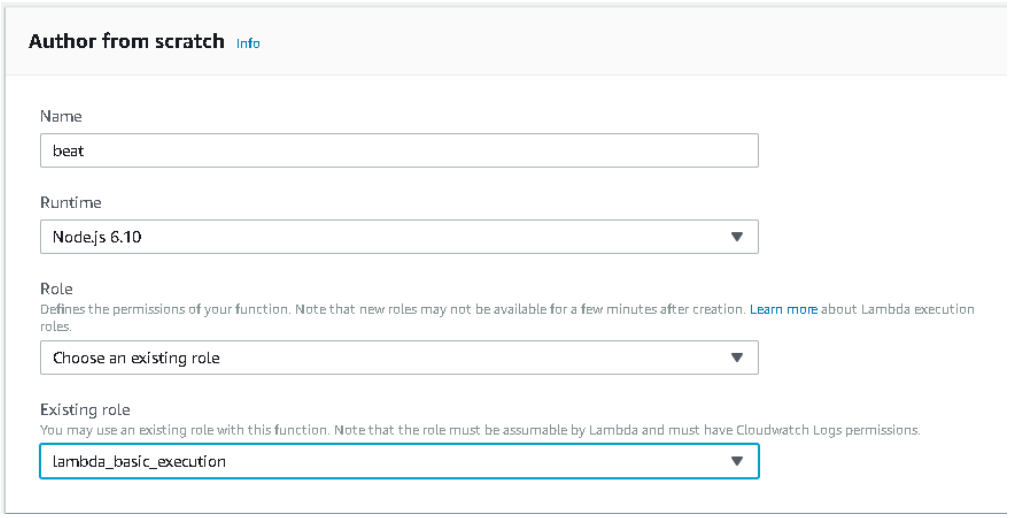

Figure 7 Creating function from scratch

Inside the create function section, you can give your function a name, set the language it will be created in and choose a role that defines the permissions the of the function.

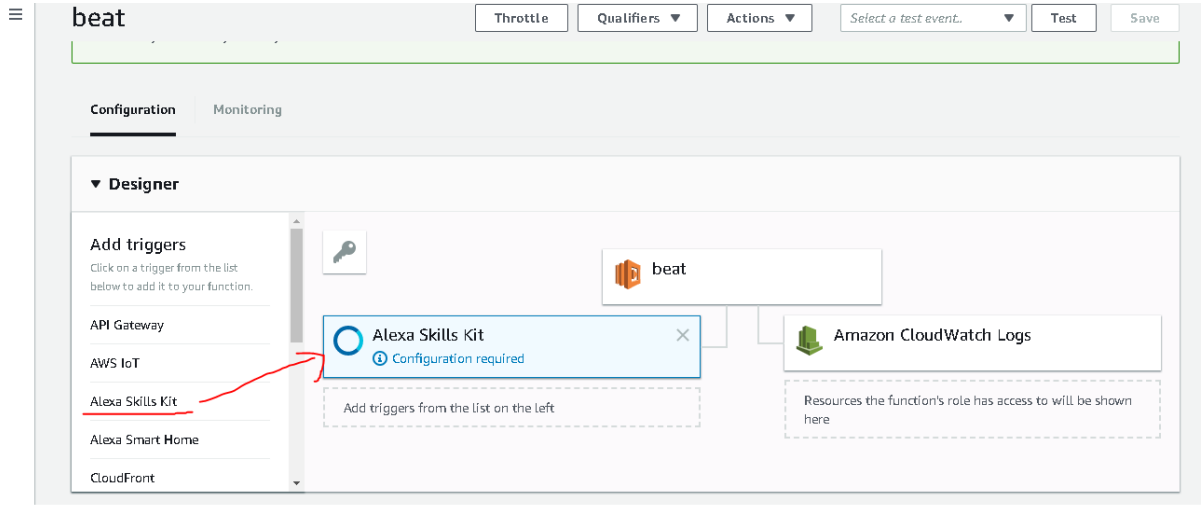

Figure 8 Addition of Alexa Skills Kit

You must add the Alexa Skills Kit to your function to make it work with Alexa. Do this by selecting from the trigger section on the top of the page (See Figure 8).

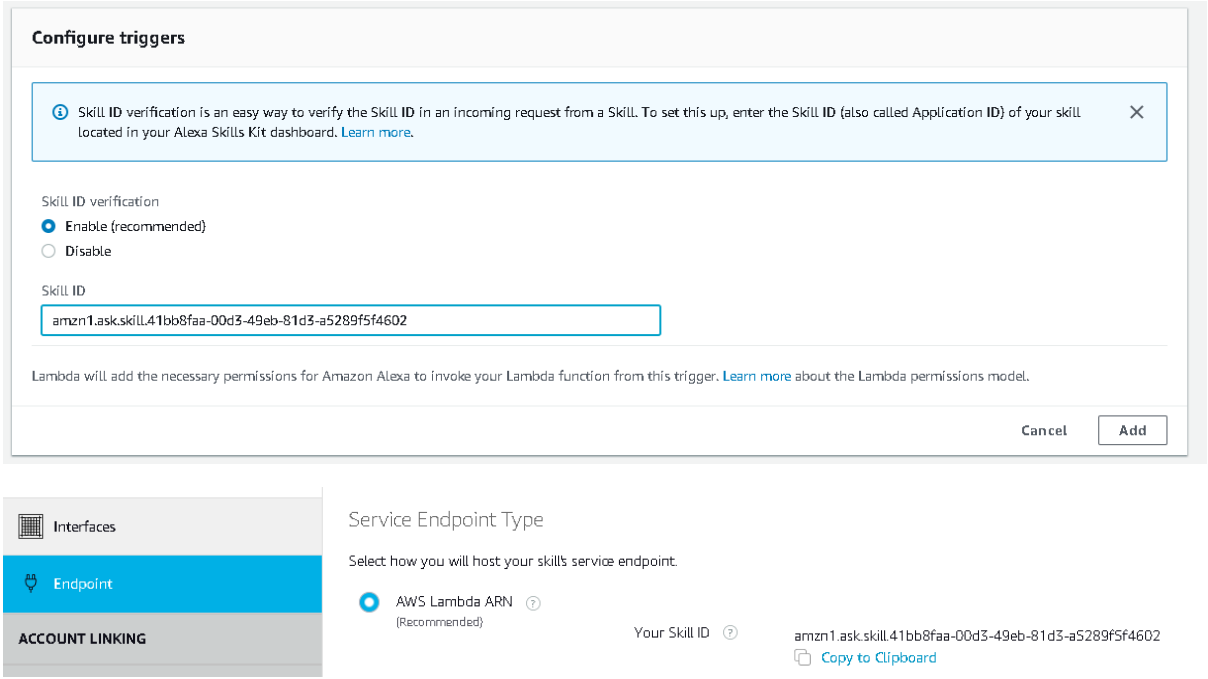

Figure 9 Added Skill ID

When configuring the triggers, you must set the Skill ID for verification purposes. This ID is found back on the Alexa skill builder in the 'Endpoint' section.

Now it's time to create the code to run when this lambda function is called. Since we set Node.js as our runtime we must create the code using that.

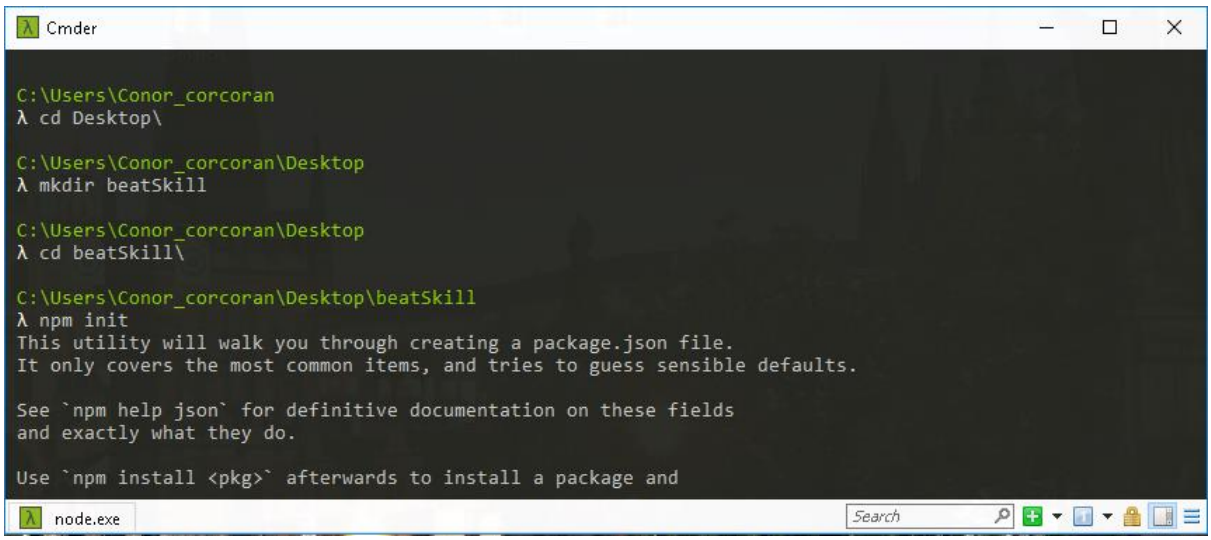

Figure 10 Creating a new file and initialising Node

Above (See Figure 10) is the commands to create a new directory and initialise it with Node. You will need to have Node installed on your PC to do this.

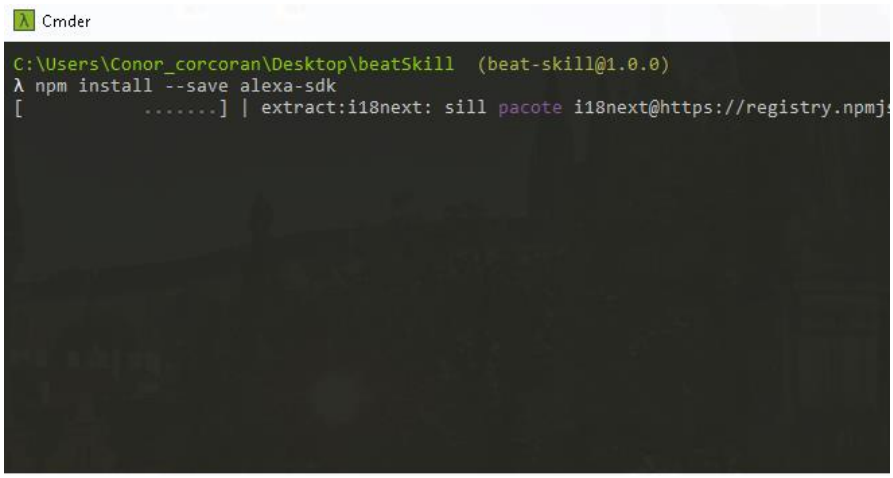

Figure 11 Installing Alexa SDK

You will also need to install the Alexa SDK inside this directory. See Figure 11 on how to do this.

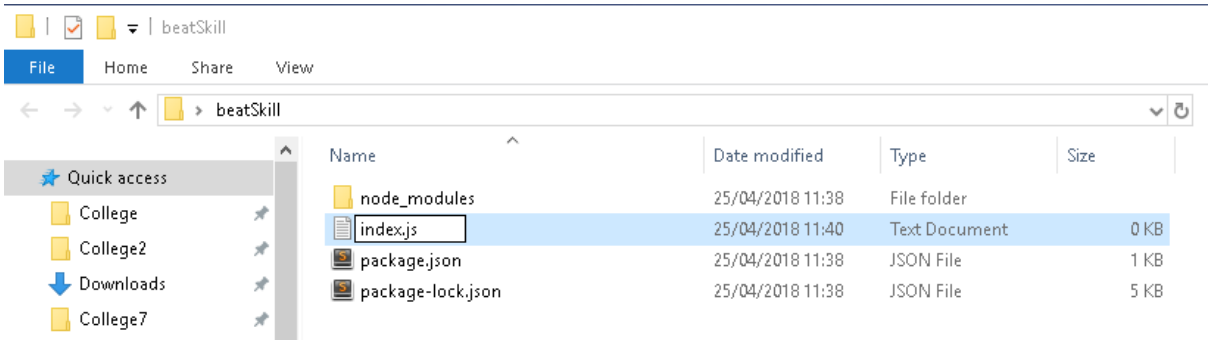

Figure 12 Creating the JavaScript File

Inside of the directory which should be populated with a node modules folder and 2 json files, you must create a JavaScript file. This file will contain the code to be run.

Before writing the code, we are going to need an MP3 to play when our skill is called. There are certain rules that need to be followed when using an MP3 file in a skill.

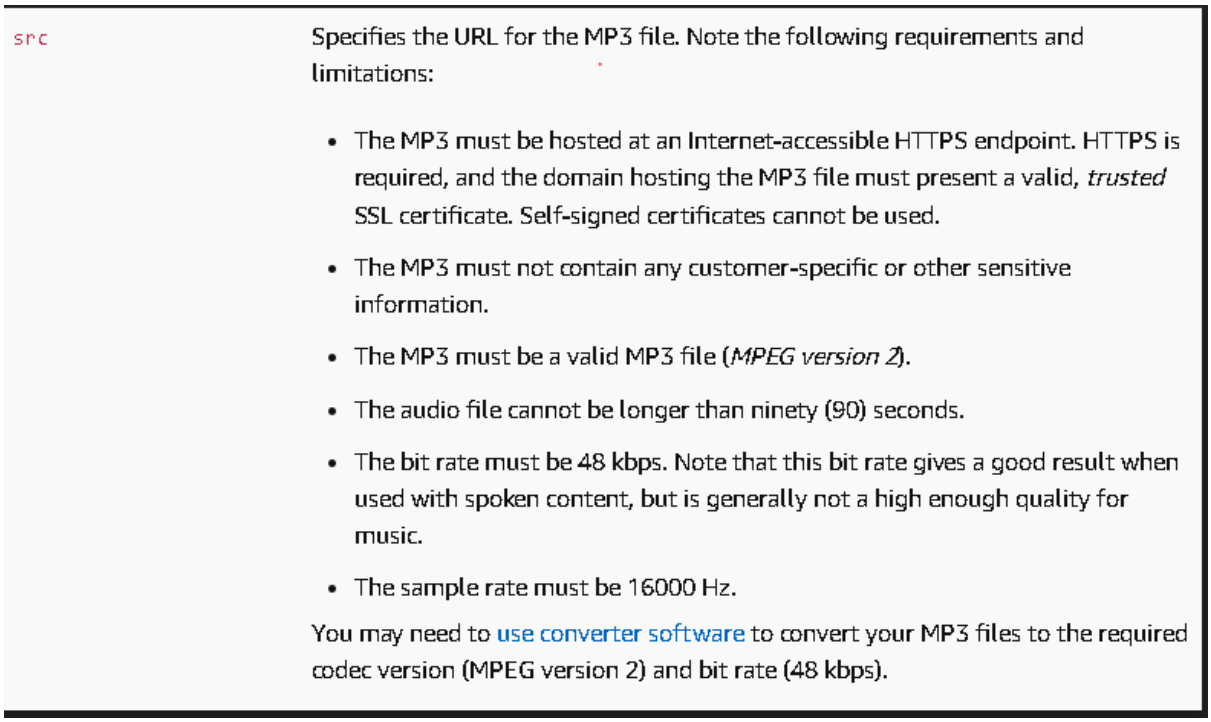

Figure 13 MP3 format rules

The main points to be taken from the above rules (See Figure 13) are the 90 seconds or less length, the sample rate + bit rate and where the file is hosted. To make sure the file is in the right format, Audacity, a free piece of audio editing software was used. This enables you to shorten the length of a clip if needed and export it with the correct bit and sample rate.

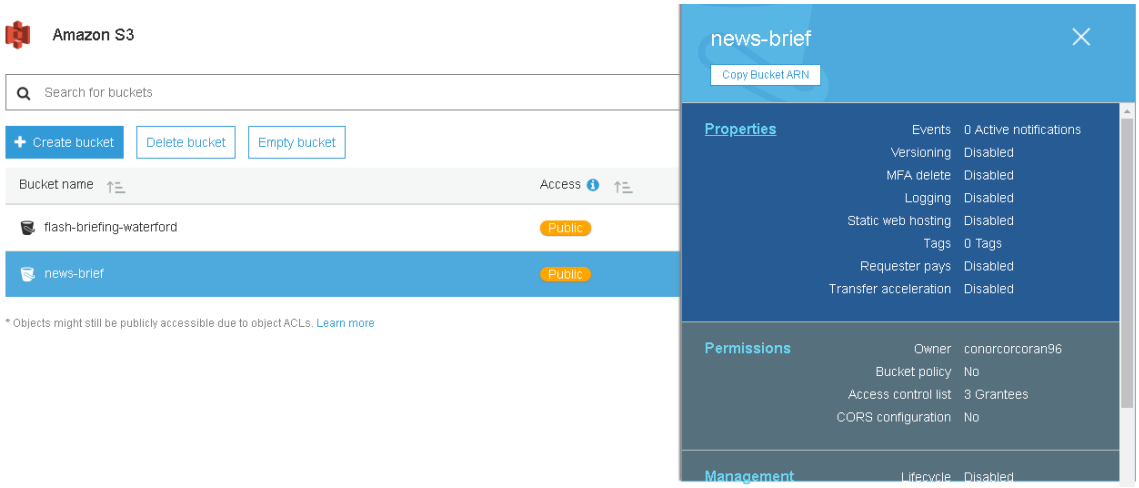

Figure 14 Bucket created in S3

To host the file, AWS was used again. The S3 storage service be used to host files and make them accessible to the public. You first need to create a bucket with the correct permissions and then you can store your file there. Take note of the URL of the file.

This URL can then be placed into the code that is inside your index.js file.

```
var Alexa = require('alexa-sdk');

exports.handler = function(event, context, callback) {
    var alexa = Alexa.handler(event, context);

    alexa.registerHandlers(handlers);
    alexa.execute();
};

var handlers = {
    'LaunchRequest': function() { //Executes when a new session is launched
        this.emit('LaunchIntent');
    },

    'LaunchIntent': function() {
        this.emit(':tell', "Here is the latest news from Beat <audio src='https://s3.amazonaws.com/news-brief/test_clip.mp3'/>");

    }
};
```

Figure 15 Index.js code

The code above (See Figure 15) will emit the short message "Here is the latest news from Beat" and then play the audio file that is linked when the user asks Alexa to “Open Beat News”. This is because the response that we want is located inside the Launch Intent.

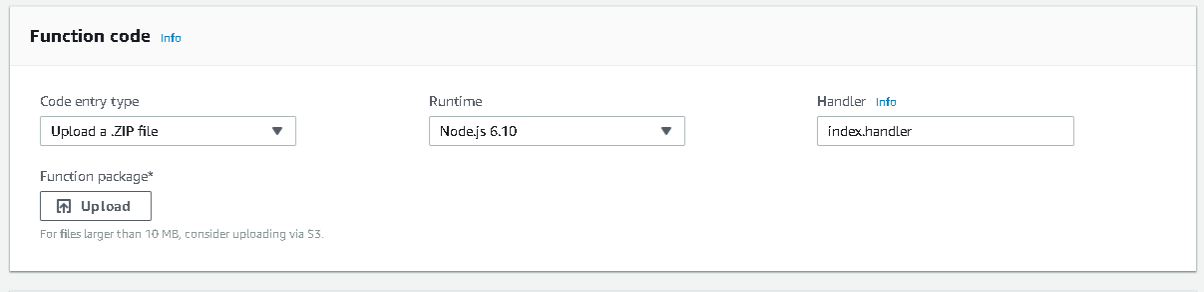

Figure 16 upload area for zipped file

Save down this code and then zip it up along with the node modules and the package.json file. This zipped file can then be uploaded onto the lambda function in the Function Code section (See Figure 16).

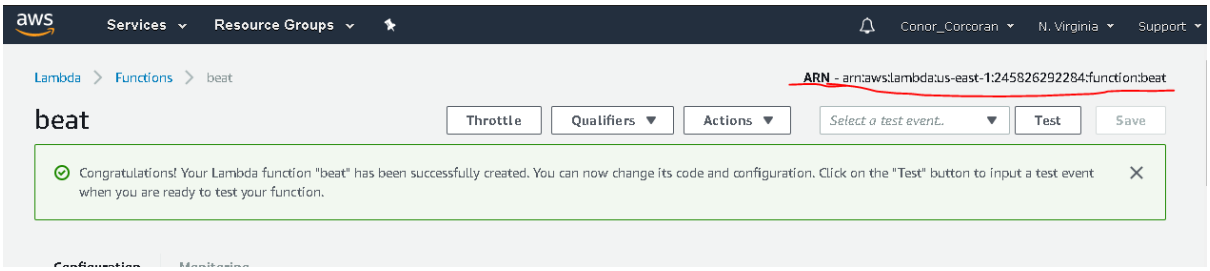

Figure 17 ARN location

You can now save down this Lambda function and copy it's ARN (Amazon Resource Name) from the top of the page (See Figure 17).

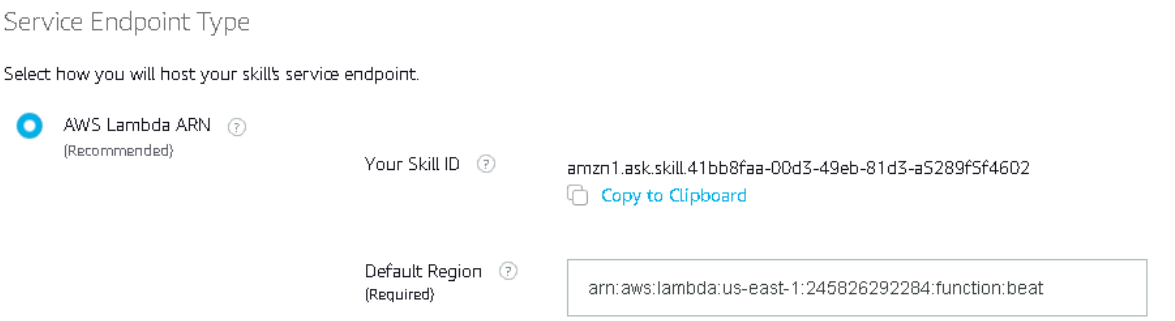

Figure 18 End point set to the ARN

Now you can put this ARN into the endpoint of your skill as seen in Figure 18. Once you have it copied in you can you can now build your skill.

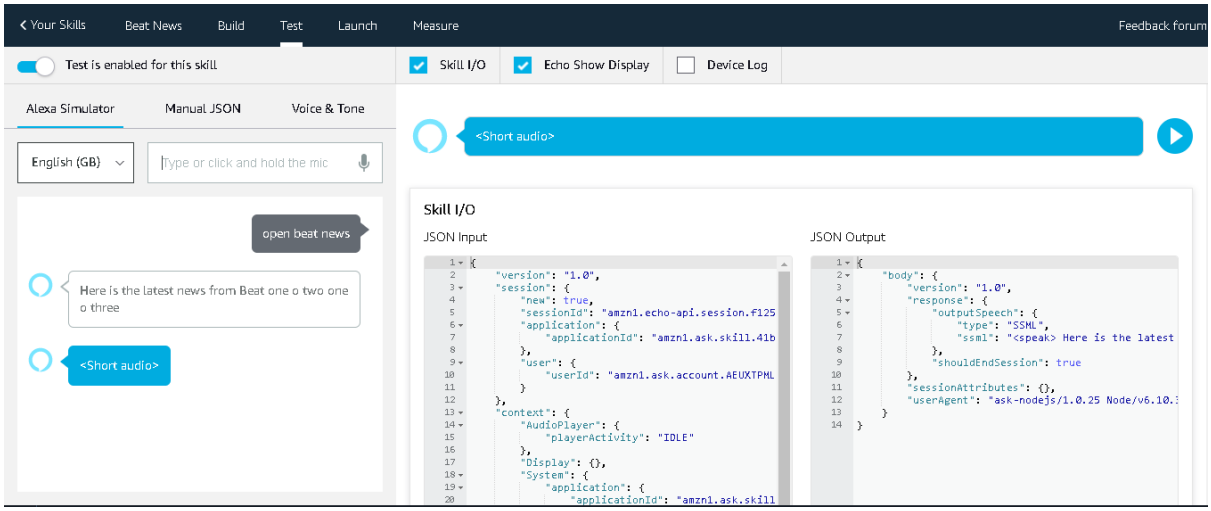

Figure 19 Test section on the Alexa skill

Moving over to the test section, you can type in your invocation name and test to see if the skill is working. You can see the JSON response on the right and the Speak response on the left (See Figure 19). If it is working as expected then the last thing to do is to go to either your Alexa mobile app or web app, select Your Skills and enable it.

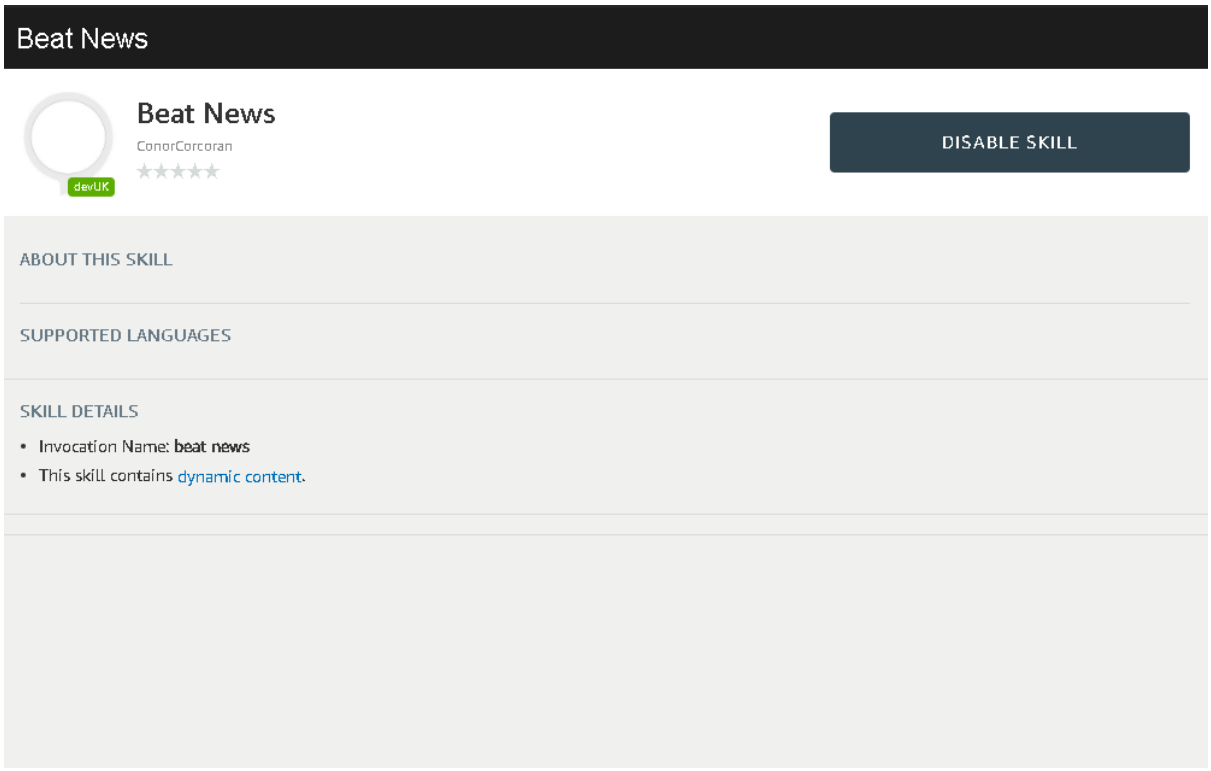

Figure 20 Skill enabled in the Alexa Web app

You should now be able to access your skill on the Amazon Dot device.

References

Amazon, 2018. *Amazon Echo Dot.* [Online]
Available at: https://www.amazon.co.uk/Amazon-Echo-Dot-Generation-Black/dp/B01DFKBL68/ref=lp_10983873031_1_1?s=amazon-devices&ie=UTF8&qid=1524646821&sr=1-1
[Accessed 20 April 2018].

Lab, D., 2017. *A custom Amazon Alexa Skill in 14-minutes.* [Online]
Available at: https://www.youtube.com/watch?v=RMUM0hQOq8Q
[Accessed 21 March 2018].

StudyZone, 2017. *Amazon Alexa Skills - Play Audio File tutorial.* [Online]
Available at: https://www.youtube.com/watch?v=QQcRgnWoyoM&t=2s
[Accessed 6 April 2018].