OPITZ CONSULTING

■■■ Überraschend mehr Möglichkeiten

Alexa – your new coworker?

OC MeetUp Munich 09.01.2018

Tomasz Krajewski

Informationsklassifikation:
Interner Gebrauch

Agenda

1 Short Introduction

2 How to Write a Skill

3 Let‘s Try it out Yourself!

VUI - Voice User Interface

- VUI ~ GUI

- Input: Voice
- Output: Voice

- A well known example:

  A Hotline

Today

- 3 tech giants fight crucial battle over voice recognition

What about Microsoft?

Are voice Chatbots new Apps? Don’t you believe?

- There is no possibilty that this device will feel more comfortable to the executive than the keyboard. Because of its ‘rollability‘, the mause has the aura of a gimmick
  - COMPUTERWORLD, 10/1983

- “mice are nice ideas, but of dubious value for business users.“
  - PC WEEK, 04/1984

TODO List

Today (yesterday?)

1 Take your smartphone

2 Unlock it

3 Find the right app

4 Go to tab todo list

5 Press add a new task

6 Type a name

7 Press save button

8 Put your smartphone away

Tomorrow? (today?)

1 Be in the room where Alexa is

2 Alexa, add „prepare my presentation for Alexa event on Tuesday 11.02.2018 and add it to my outlook calendar“

3 Done

Alexa for business

NEW!
Alexa for Business
Your intelligent assistant for work
Generally available today

https://www.youtube.com/watch?v=ViB3XhsTLuo

Webinar: Building Skills with Alexa for Business

You can register now and save the date Tuesday, January 16, 20189 AM-10 AM PT. There is a special webinar by Amazon about new alexa features

http://dev.amazonappservices.com/alexa-for-business-web-reg.html

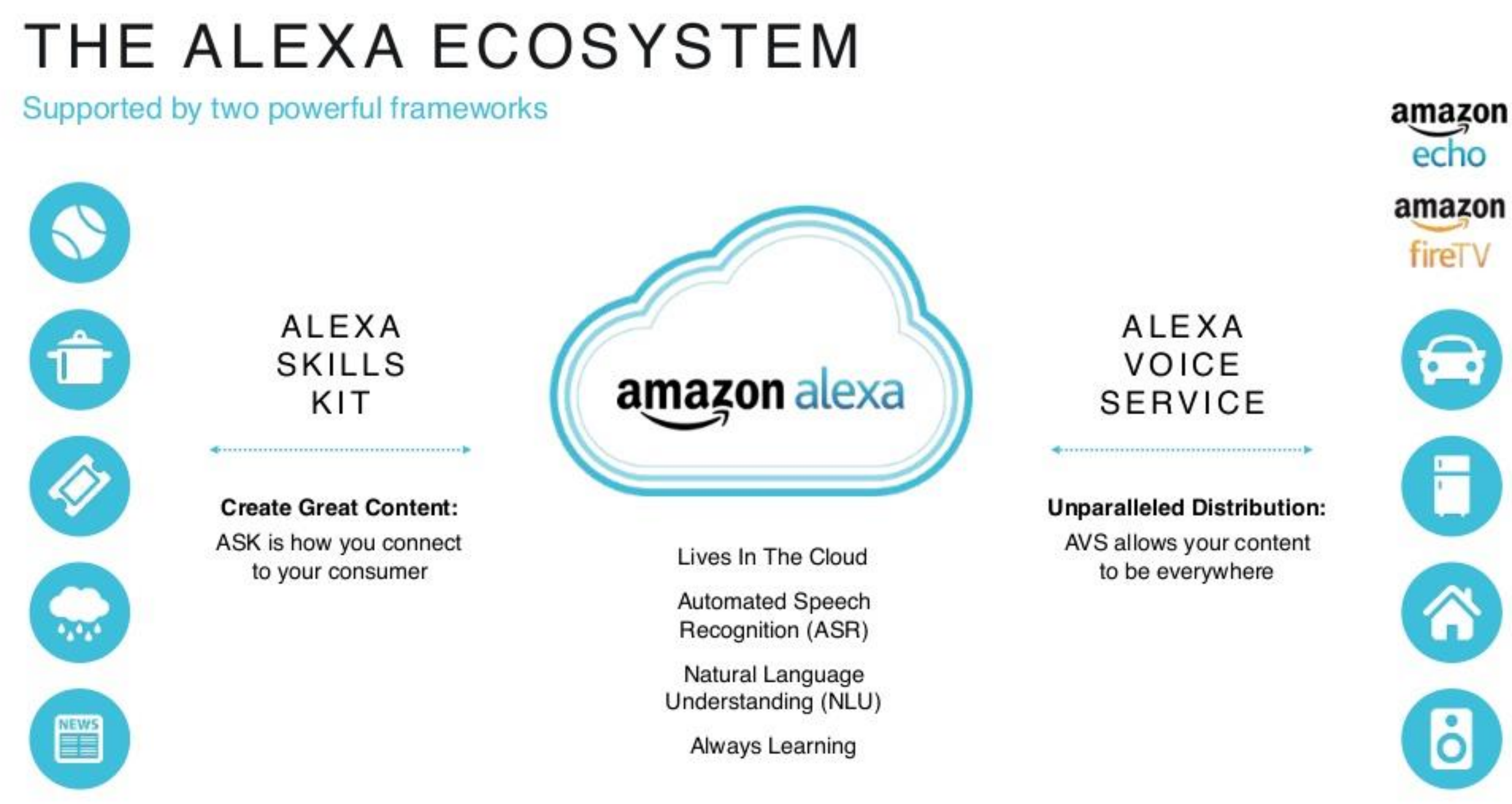
What is Alexa?

amazon alexa

Cloud-based intelligence

Automated Speech-Recognition (ASR)

Natural Language Procession (NLU)



THE ALEXA ECOSYSTEM

Supported by two powerful frameworks

ALEXA SKILLS KIT

Create Great Content:
ASK is how you connect to your consumer

Lives In The Cloud

Automated Speech Recognition (ASR)

Natural Language Understanding (NLU)

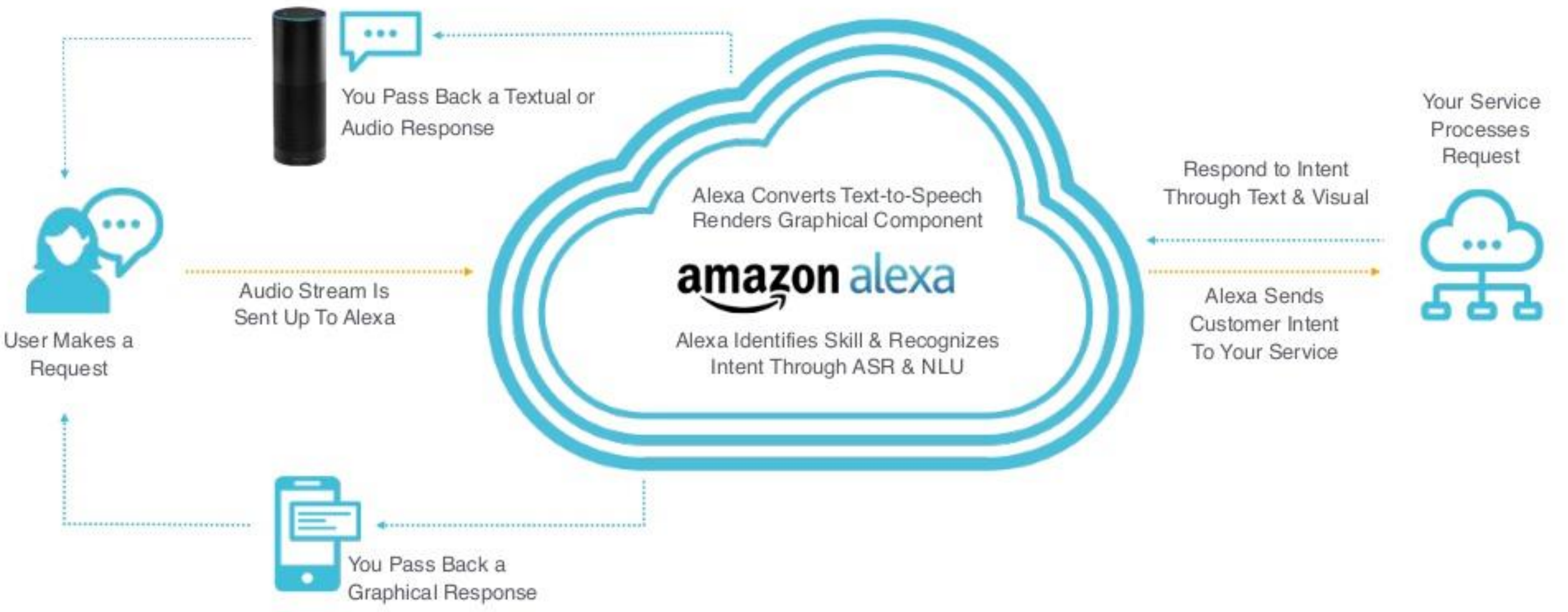
Always Learning

ALEXA VOICE SERVICE

Unparalleled Distribution:
AVS allows your content to be everywhere

UNDER THE HOOD OF ASK

A closer look at how the Alexa Skills Kit process a request and returns an appropriate response



User Makes a Request

Audio Stream Is Sent Up To Alexa

Alexa Identifies Skill & Recognizes Intent Through ASR & NLU

Alexa Sends Customer Intent To Your Service

Your Service Processes Request

Respond to Intent Through Text & Visual

Alexa Converts Text-to-Speech Renders Graphical Component

You Pass Back a Textual or Audio Response

You Pass Back a Graphical Response

Meet the Alexa Family

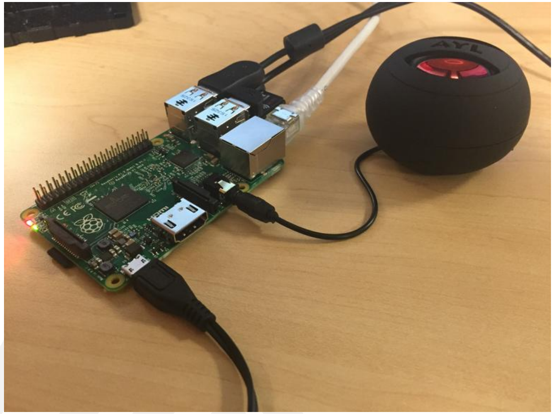
Alexa Skill Testing Tool

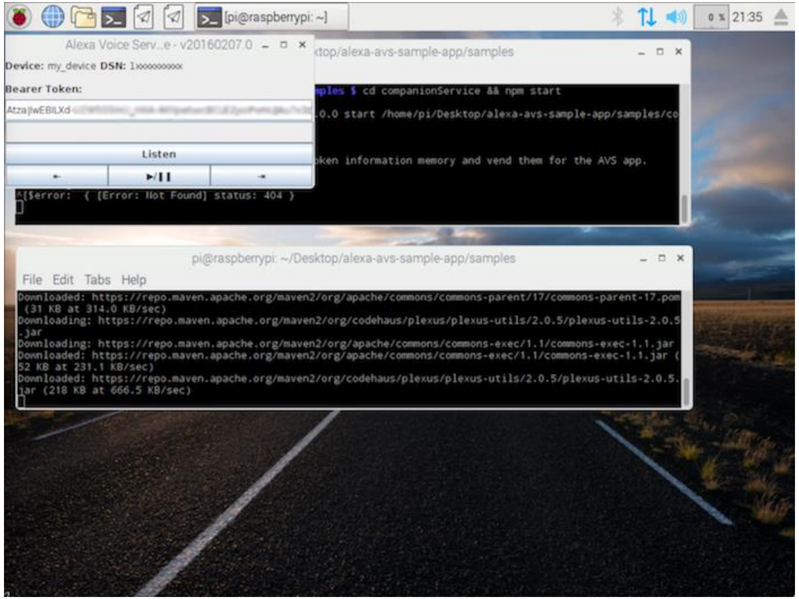
Click and hold to speak your command.

Ready...

Alexa with Raspberry Pi

- https://github.com/alexa/alexa-avs-sample-app
- https://github.com/alexa-pi/AlexaPi





[pi@raspberrypi: ~]
21:35
Alexa Voice Serv...e - v20160207.0
Listen
⏵/❙❙
pi@raspberrypi: ~/Desktop/alexa-avs-sample-app/samples
File
Edit
Tabs
Help

Device: my_device DSN: 1xxxxxxxxxx

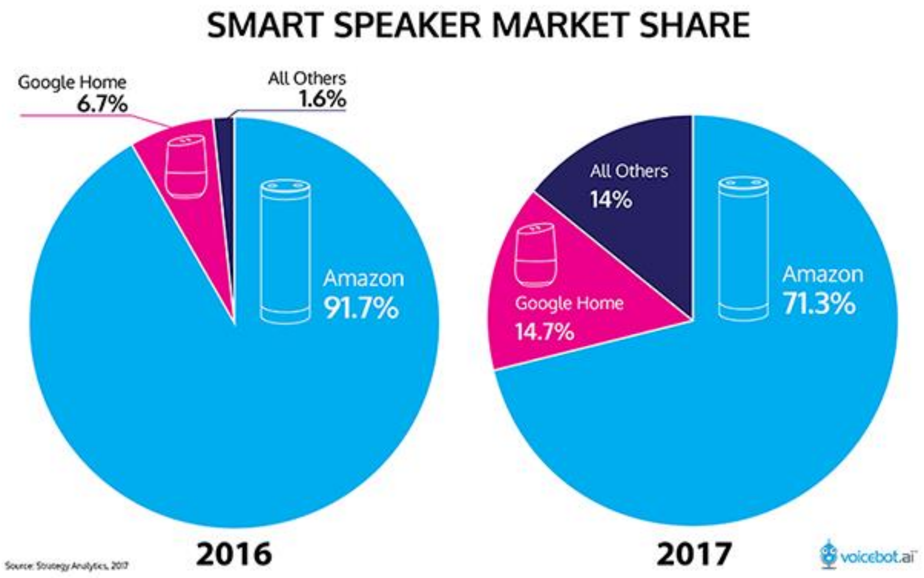
Bearer Token:

Atza|wEBILXd[redacted]

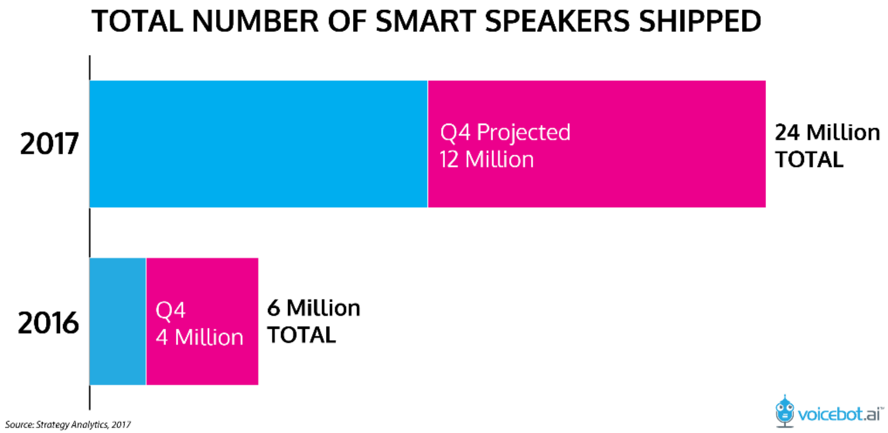
```
ktop/alexa-avs-sample-app/samples
mples $ cd companionService && npm start
.0.0 start /home/pi/Desktop/alexa-avs-sample-app/samples/co
oken information memory and vend them for the AVS app.
^[$error:  { [Error: Not Found] status: 404 }
```

```
Downloaded: https://repo.maven.apache.org/maven2/org/apache/commons/commons-parent/17/commons-parent-17.pom
 (31 KB at 314.0 KB/sec)
Downloading: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/2.0.5/plexus-utils-2.0.5
.jar
Downloading: https://repo.maven.apache.org/maven2/org/apache/commons/commons-exec/1.1/commons-exec-1.1.jar
Downloaded: https://repo.maven.apache.org/maven2/org/apache/commons/commons-exec/1.1/commons-exec-1.1.jar (
52 KB at 231.1 KB/sec)
Downloaded: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/2.0.5/plexus-utils-2.0.5.
jar (218 KB at 666.5 KB/sec)
```

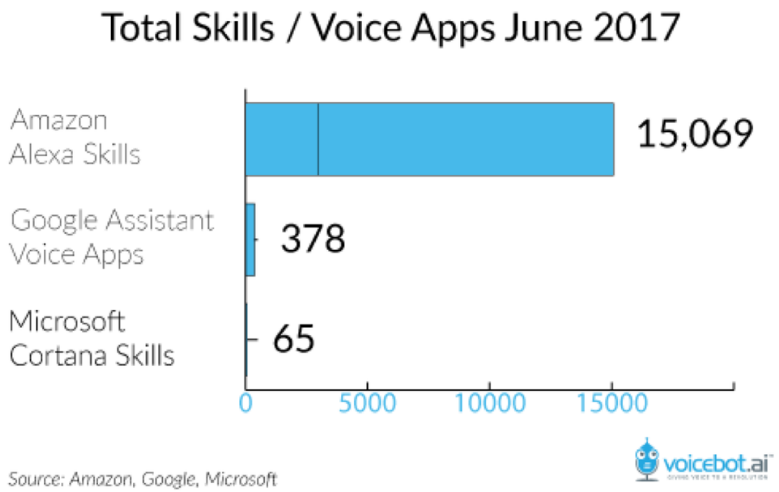
Market


SMART SPEAKER MARKET SHARE

Source: Strategy Analytics, 2017


TOTAL NUMBER OF SMART SPEAKERS SHIPPED

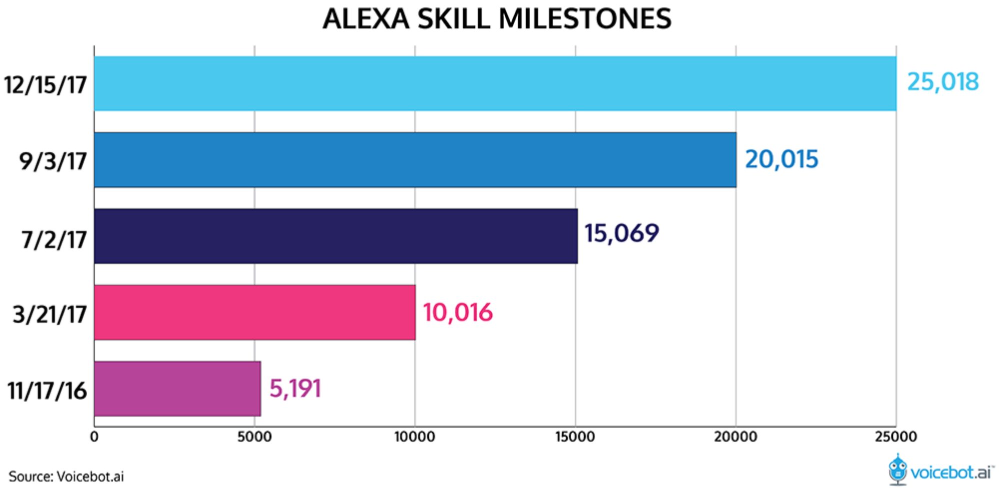
Source: Strategy Analytics, 2017


Total Skills / Voice Apps June 2017

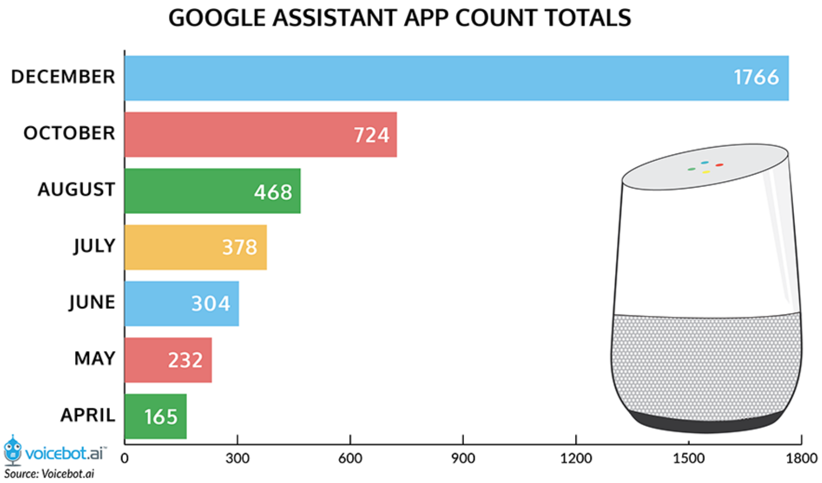
Source: Amazon, Google, Microsoft

Source: https://www.voicebot.ai/amazon-echo-alexa-stats/

Alexa vs Google Home Skills fight in the USA market (December 2017)



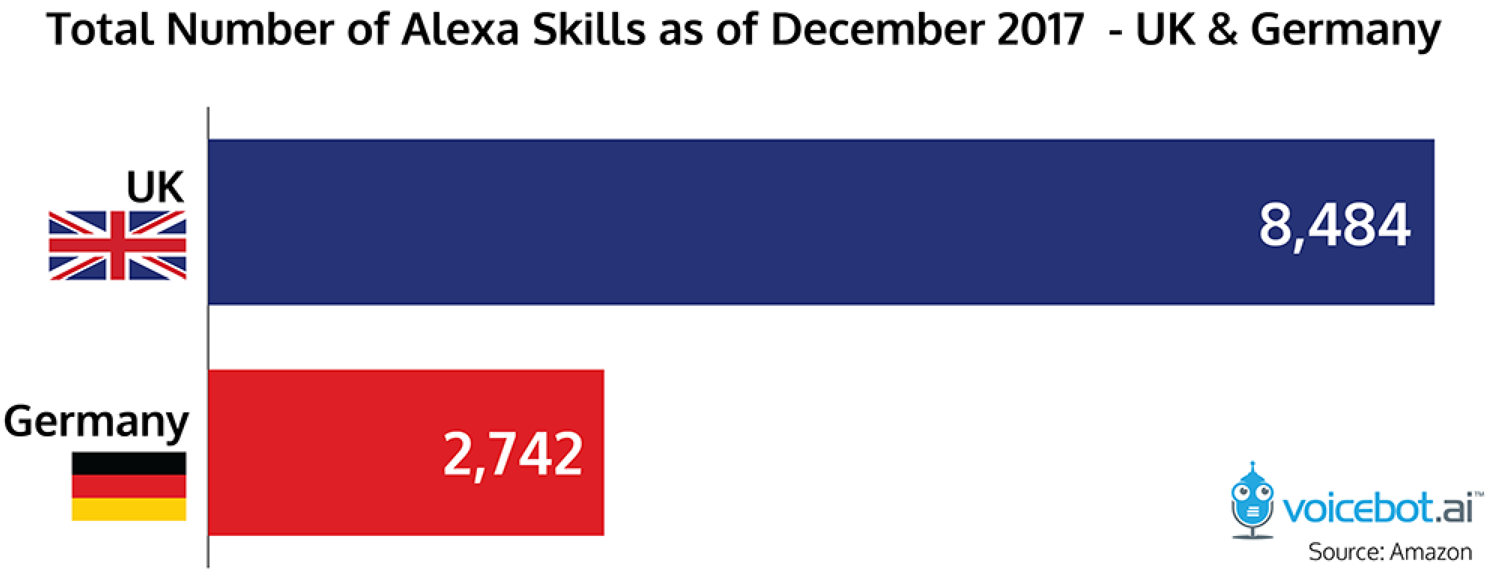
Source: Voicebot.ai



Source: Voicebot.ai

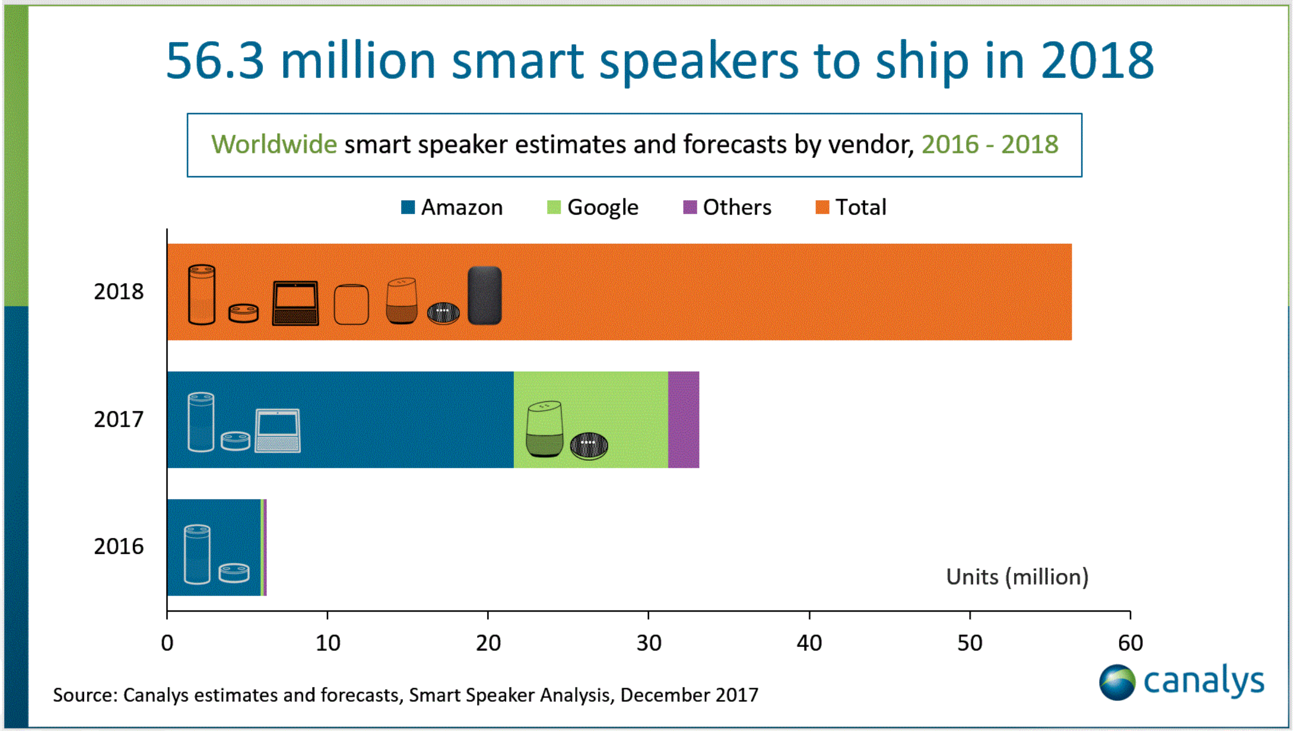
Source: https://www.voicebot.ai/

What about German or UK market?



Source: https://www.voicebot.ai/

56 Million Smart Speaker Sales in 2018 Says Canalys



56.3 million smart speakers to ship in 2018

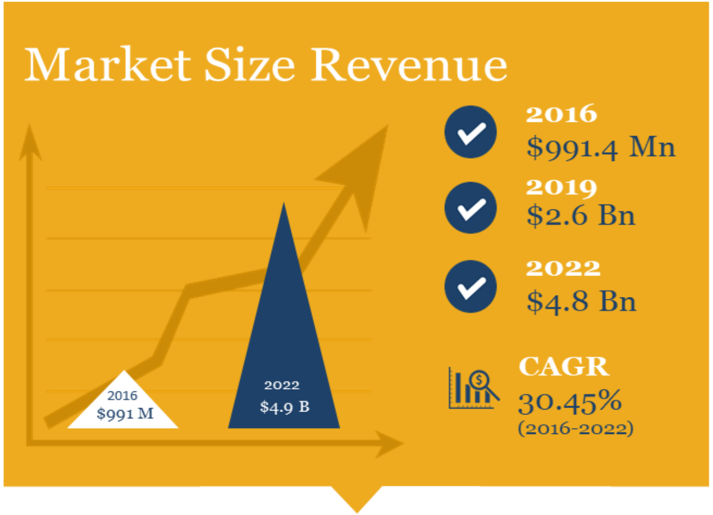
Worldwide smart speaker estimates and forecasts by vendor, 2016 - 2018

Source: Canalys estimates and forecasts, Smart Speaker Analysis, December 2017

Source: https://www.voicebot.ai/

Arizton: Smart Speaker Market $4.8 Billion in 2022

- Chicago-based research firm Arizton has a new report revealing that the smart speaker market was $991 million in 2016 and will grow to $4.8 billion in 2022



Source: https://www.voicebot.ai/

How to write an Amazon skill?

What technology can I use?

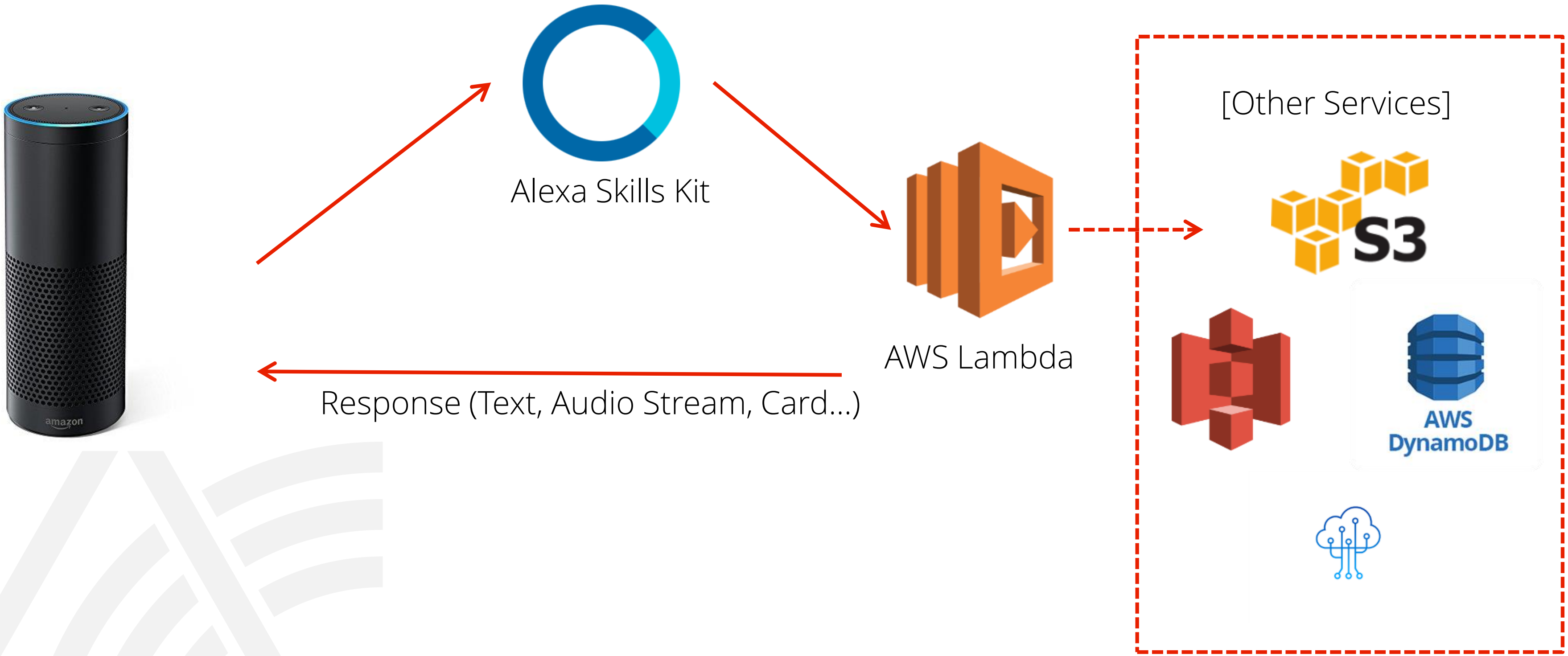
- Theoretical any programming language
- My recommendations
  - Python (python with flask-ask https://github.com/johnwheeler/flask-ask and https://www.pythonanywhere.com)
  - Node.js (javascript and heroku.com)
  - AWS? Yes but not always

  - Pro: https://developer.amazon.com/de/alexa-skills-kit/alexa-aws-credits
  - Pro: free tier, which offers one million AWS Lambda requests and up to 750 hours of Amazon EC2 compute time per month at no cost

Development Setup @ Amazon

2 Accounts at Amazon. You need to add your Credit Card for Amazon Web Services account!!

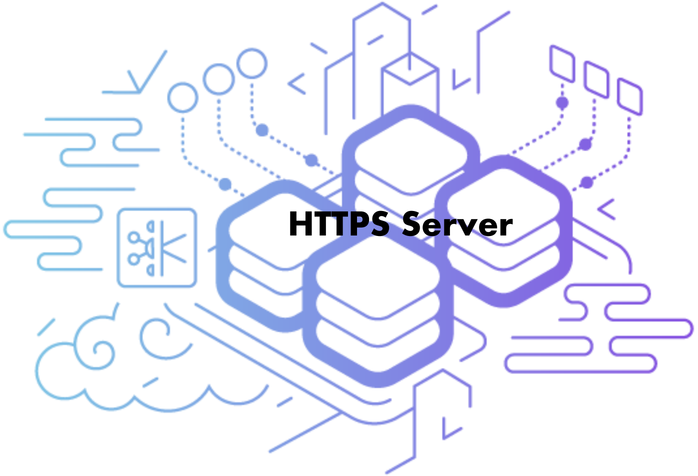
You can use our Alexa template for developing skills at AWS. Just follow 3 short steps from my repository.
https://github.com/falent/Alexa_universal_skill_template_VM/tree/master/dist/app/src


Alexa Skills Kit
AWS Lambda
[Other Services]
S3
AWS DynamoDB
amazon
Response (Text, Audio Stream, Card...)

Development Setup @ HTTPS server

Amazon Developer Account and your own https server (any public node.js provoder, like Heroku).

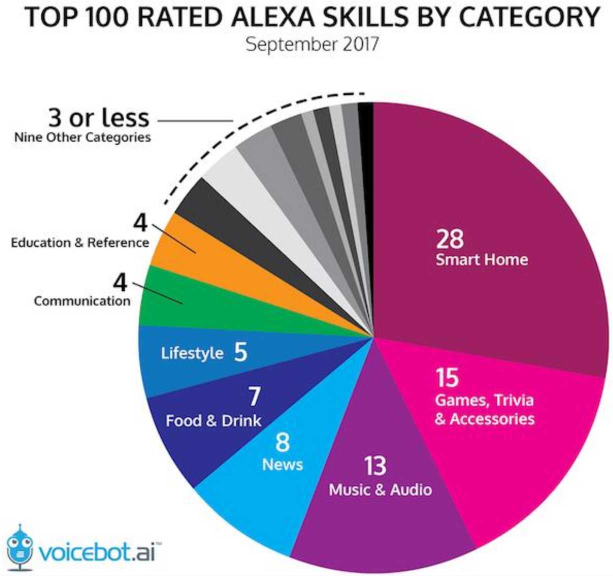
amazon
Developer



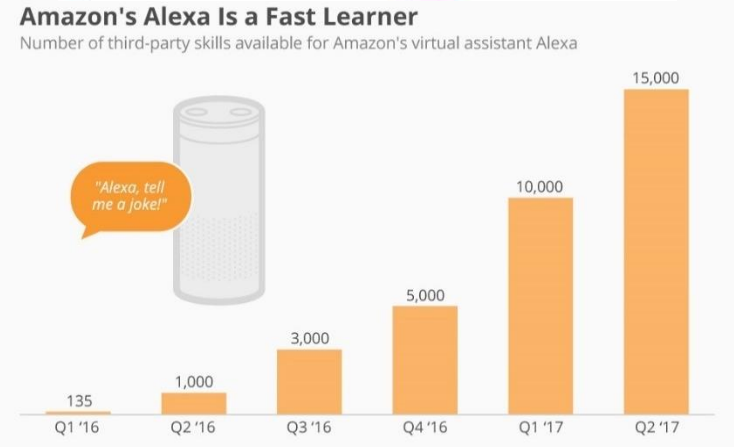
Basics

Skills

- Skills are built in capabilities for Alexa to give new customer experiences.
- You can treat it as a normal program or a script
- Find skills for your Alexa https://www.amazon.de/b?node=10068460031


TOP 100 RATED ALEXA SKILLS BY CATEGORY

September 2017


Amazon's Alexa Is a Fast Learner

Number of third-party skills available for Amazon's virtual assistant Alexa

Get your Alexa dot for free!

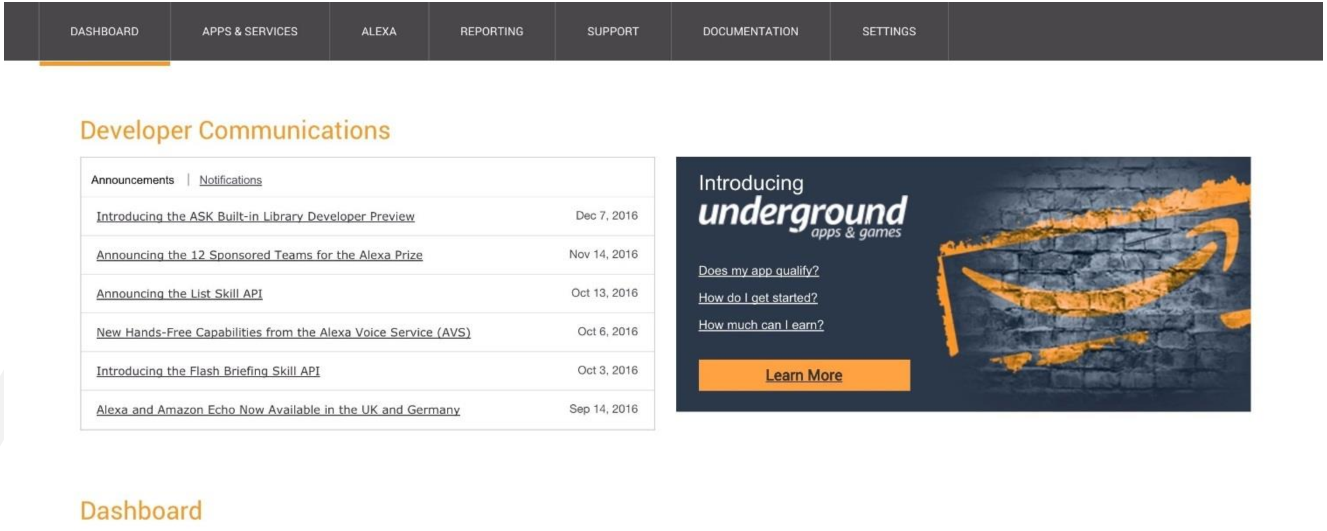
Build a simple skill :)

- https://developer.amazon.com/de/a-skills-kit/alexa-developer-skill-promotion

Let’s configure your new skill!

Go to Amazon Developer Portal

- 1. Let's go to https://developer.amazon.com/ and create a new account (you can use your current amazon account. Amazon developer portal is for free). Once you're in, you should see something that looks like this:



Developer Communications

Announcements | Notifications

Introducing the ASK Built-in Library Developer Preview	Dec 7, 2016
Announcing the 12 Sponsored Teams for the Alexa Prize	Nov 14, 2016
Announcing the List Skill API	Oct 13, 2016
New Hands-Free Capabilities from the Alexa Voice Service (AVS)	Oct 6, 2016
Introducing the Flash Briefing Skill API	Oct 3, 2016
Alexa and Amazon Echo Now Available in the UK and Germany	Sep 14, 2016

Does my app qualify?

How do I get started?

How much can I earn?

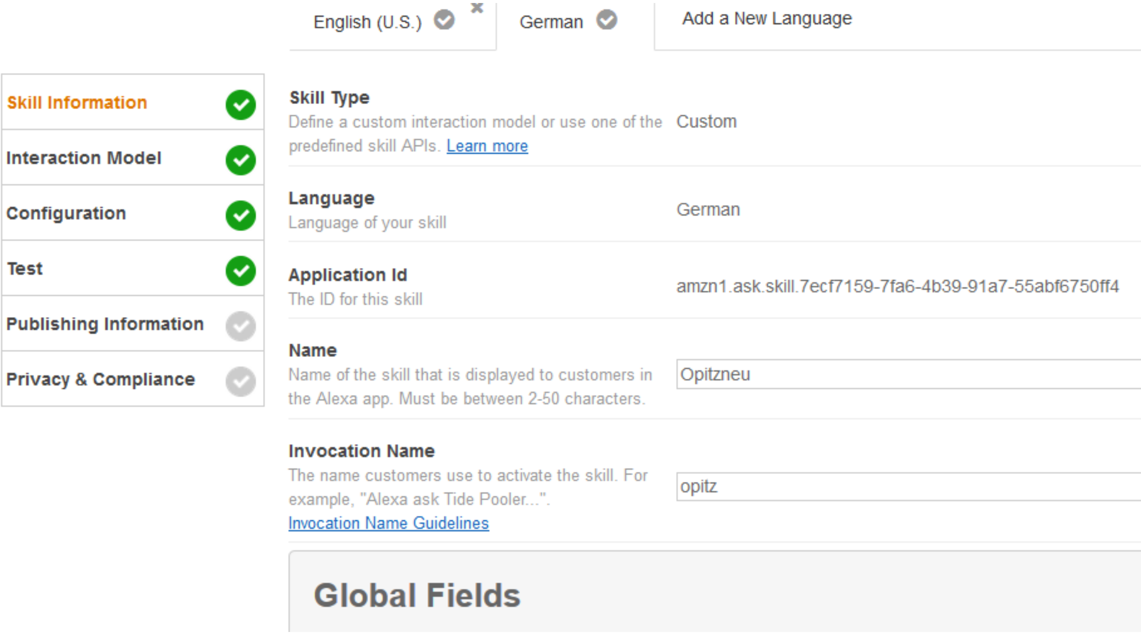
Dashboard

- 2. Click “Alexa” at the top and then click Alexa Skills Kit.

Dashboard

- Click "Add a New Skill" on the right and we'll go ahead and start creating our new skill.
- Add name skill and invocation name.

  It should be unique and easy to say.



English (U.S.) ✓ ✖ | German ✓ | Add a New Language

Skill Information	✓
Interaction Model	✓
Configuration	✓
Test	✓
Publishing Information	✓
Privacy & Compliance	✓

Skill Type
Define a custom interaction model or use one of the predefined skill APIs. Learn more
Custom

Language
Language of your skill
German

Application Id
The ID for this skill
amzn1.ask.skill.7ecf7159-7fa6-4b39-91a7-55abf6750ff4

Name
Name of the skill that is displayed to customers in the Alexa app. Must be between 2-50 characters.
Opitzneu

Invocation Name
The name customers use to activate the skill. For example, "Alexa ask Tide Pooler...".
Invocation Name Guidelines
opitz

Global Fields

Invocation Name

- With invocation name, Alexa identifies what the user wants to do. You can use a such sentence to start your skill **Alexa, ask <your skill invocation name>**. In our case it is:

“Alexa, **ask Opitz**“ or…

“Alexa, **start Opitz**” or…

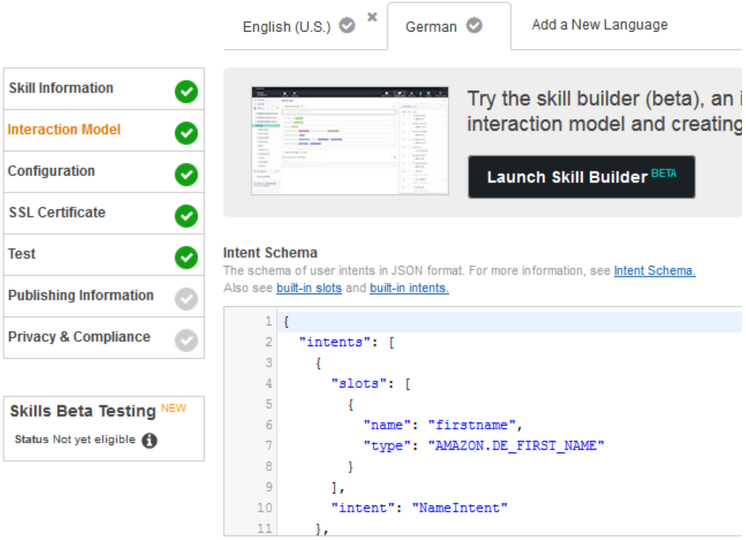
“Alexa, **begin Opitz**” or…

- Click “next“ in the developer portal and go to interaction Model

Interaction Model: Intents

- You can imagine Intent like a trigger function for your code.
- For example, NameIntent is responsible for getting a name from the user. If you want to get some data from the user you need slots.
- Of course you can have intents which have no slots or many of them

Lets see some examples



Try the skill builder (beta), an i
interaction model and creating

Intent Schema

The schema of user intents in JSON format. For more information, see Intent Schema.
Also see built-in slots and built-in intents.

```
{
  "intents": [
    {
      "slots": [
        {
          "name": "firstname",
          "type": "AMAZON.DE_FIRST_NAME"
        }
      ],
      "intent": "NameIntent"
    },
```

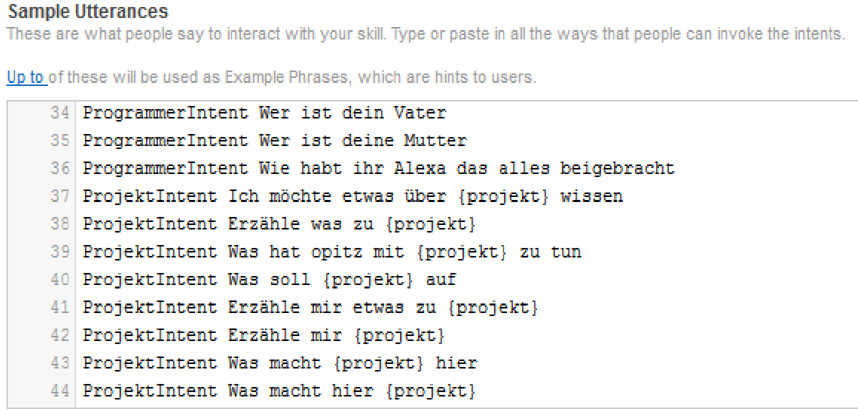
Interaction Model: Built-in Intents

- There are also built-in intents like AMAZON.YesIntent or AMAZON.HelpIntent. They were designed to react to most used cases like: “Alexa, stop” (AMAZON.YesIntent)! Built-in intents already have utterances (user sentences) assigned to them. We can map more utterances to these built-in intents as necessary.
- Please remember built-in intents must be included into Interaction model if you want to use them! You don‘t need to define special utterance, but of course you can do it. For example if you want that your user close your skill, add:

  “AMAZON.StopIntent I dont want to talk with you any more
- More: https://developer.amazon.com/de/docs/custom-skills/standard-built-in-intents.html

Interaction Model: Utterances

- We need to set sample of utterances (your questions, answers) to improve accurancy of Alexa. Some utterances might include:
- ProjectIntent Tell me something about project?
- ProjectIntent Tell me about **{projekt}**
- What is strange value **{projekt}** ? It is a slot value which is like a variable that it has a specific value after we add plenty of similar meanings.



Sample Utterances

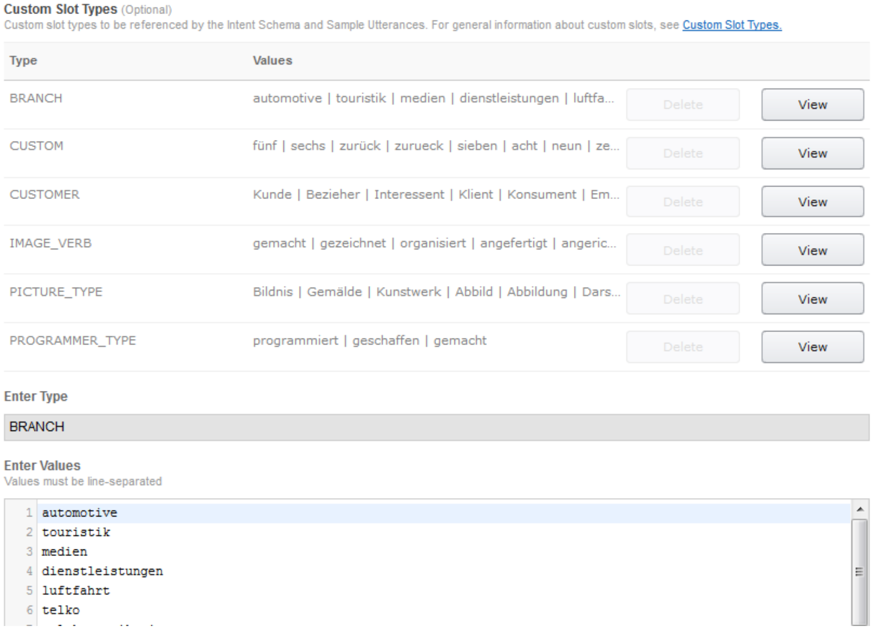
These are what people say to interact with your skill. Type or paste in all the ways that people can invoke the intents.

Up to of these will be used as Example Phrases, which are hints to users.

```
ProgrammerIntent Wer ist dein Vater
ProgrammerIntent Wer ist deine Mutter
ProgrammerIntent Wie habt ihr Alexa das alles beigebracht
ProjektIntent Ich möchte etwas über {projekt} wissen
ProjektIntent Erzähle was zu {projekt}
ProjektIntent Was hat opitz mit {projekt} zu tun
ProjektIntent Was soll {projekt} auf
ProjektIntent Erzähle mir etwas zu {projekt}
ProjektIntent Erzähle mir {projekt}
ProjektIntent Was macht {projekt} hier
ProjektIntent Was macht hier {projekt}
```

Interaction Model: Custom Slots

- A custom slot is like a variable. It can be asked by different words defined by you
- Tell me about your **{projectType}** project done for **{customer}**
- **{projectType}** can be: IoT, Big Data etc
- **{customer} can be** Lufthansa, GEMA etc



Custom Slot Types (Optional)
Custom slot types to be referenced by the Intent Schema and Sample Utterances. For general information about custom slots, see Custom Slot Types.

Type	Values
BRANCH	automotive | touristik | medien | dienstleistungen | luftfa...
CUSTOM	fünf | sechs | zurück | zurueck | sieben | acht | neun | ze...
CUSTOMER	Kunde | Bezieher | Interessent | Klient | Konsument | Em...
IMAGE_VERB	gemacht | gezeichnet | organisiert | angefertigt | angeric...
PICTURE_TYPE	Bildnis | Gemälde | Kunstwerk | Abbild | Abbildung | Dars...
PROGRAMMER_TYPE	programmiert | geschaffen | gemacht

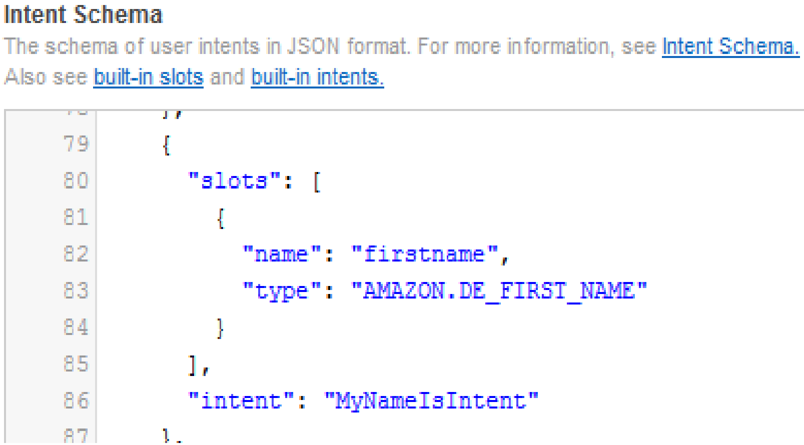
Enter Type
BRANCH

Enter Values
Values must be line-separated

```
automotive
touristik
medien
dienstleistungen
luftfahrt
telko
```

Interaction Model: Build-in Slot Types

- Alexa supports several slot types that define how data in the slot is recognized and handled. The provided types fall into the following general categories: Numbers, Dates, and Times lists of Items
- F.e: AMAZON.DATE Converts words that indicate dates (“today”, “tomorrow”, or “july”) into a date format (such as “2015-07-00T9”).
- AMAZON.DE_FIRST_NAME so you don’t need define any custom slots values for users names



Intent Schema

The schema of user intents in JSON format. For more information, see Intent Schema. Also see built-in slots and built-in intents.

```
    },
    {
      "slots": [
        {
          "name": "firstname",
          "type": "AMAZON.DE_FIRST_NAME"
        }
      ],
      "intent": "MyNameIsIntent"
    },
```

More: https://developer.amazon.com/de/docs/custom-skills/slot-type-reference.html

The Whole Interaction Model

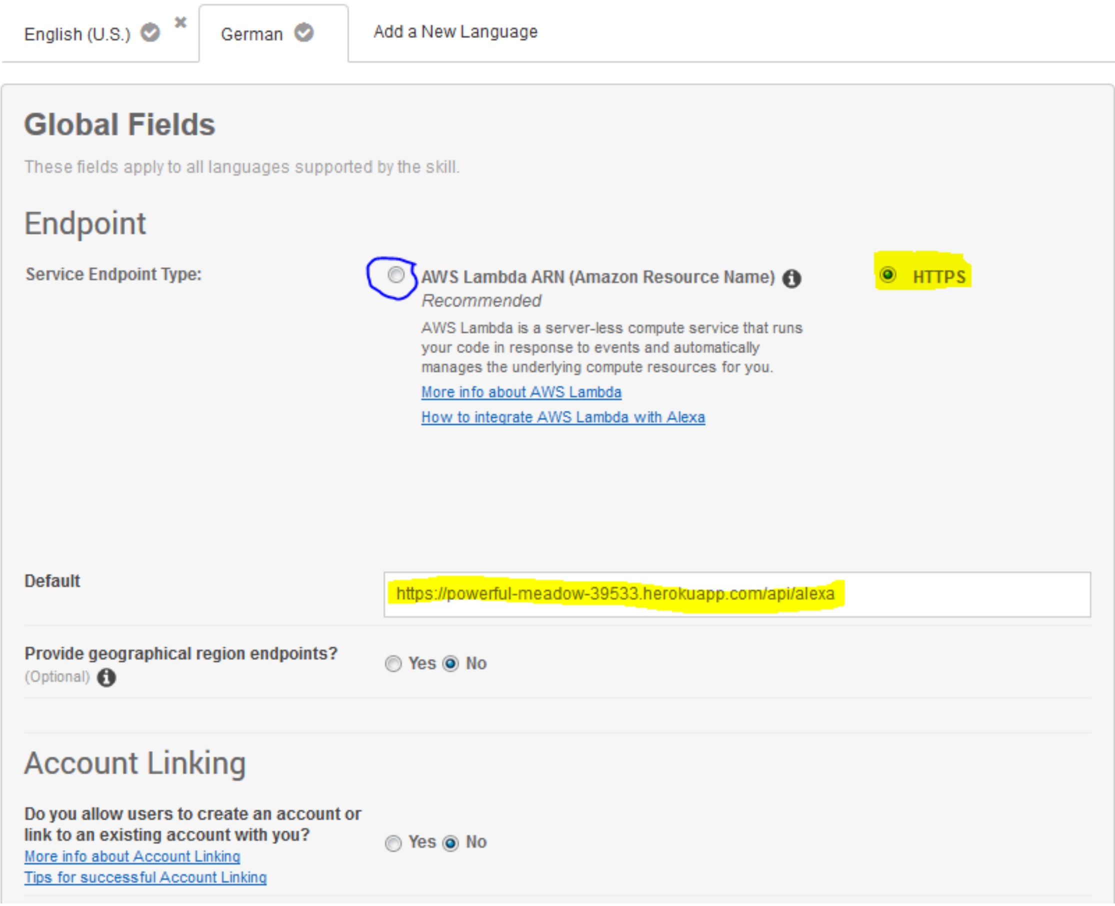
- You can find it in the templates in VM I prepared. Folder speech_assets
- Please visit our opitz git repository https://git.opitz-consulting.de/projects/ALEXAWIM where you can find lots of extended interactions models

Alexa without interaction model is…

amazon echo
Clever
Stupid

Skill Configuration

- Let’s go back to the amazon developer portal, our skill configuration section
- We will program skill locally. That is the reason we need to choose Endpoint as HTTPS (our skill code is almost the same for AWS Lambda function)



Global Fields

These fields apply to all languages supported by the skill.

Endpoint

Service Endpoint Type: ○ **AWS Lambda ARN (Amazon Resource Name)** ● **HTTPS**

Recommended

AWS Lambda is a server-less compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

More info about AWS Lambda

How to integrate AWS Lambda with Alexa

Default https://powerful-meadow-39533.herokuapp.com/api/alexa

Provide geographical region endpoints? (Optional) ○ Yes ● No

Account Linking

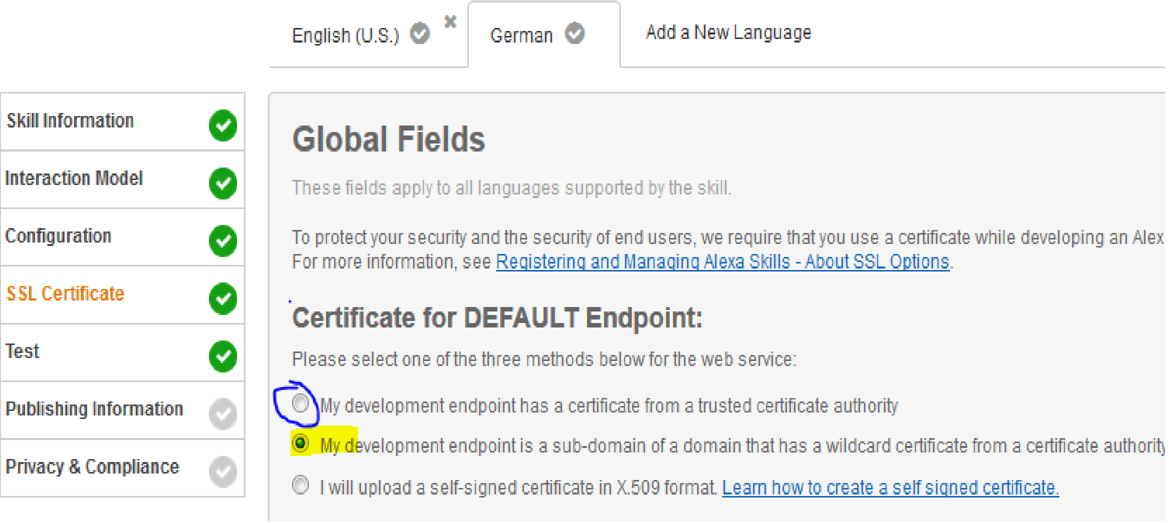
Do you allow users to create an account or link to an existing account with you? ○ Yes ● No

More info about Account Linking

Tips for successful Account Linking

Last Step... SSL Certificate

- For HTTPS endpoint, choose the second option (yellow marked)
- For AWS endpoint, the blue one



Global Fields

These fields apply to all languages supported by the skill.

To protect your security and the security of end users, we require that you use a certificate while developing an Alex
For more information, see Registering and Managing Alexa Skills - About SSL Options.

Certificate for DEFAULT Endpoint:

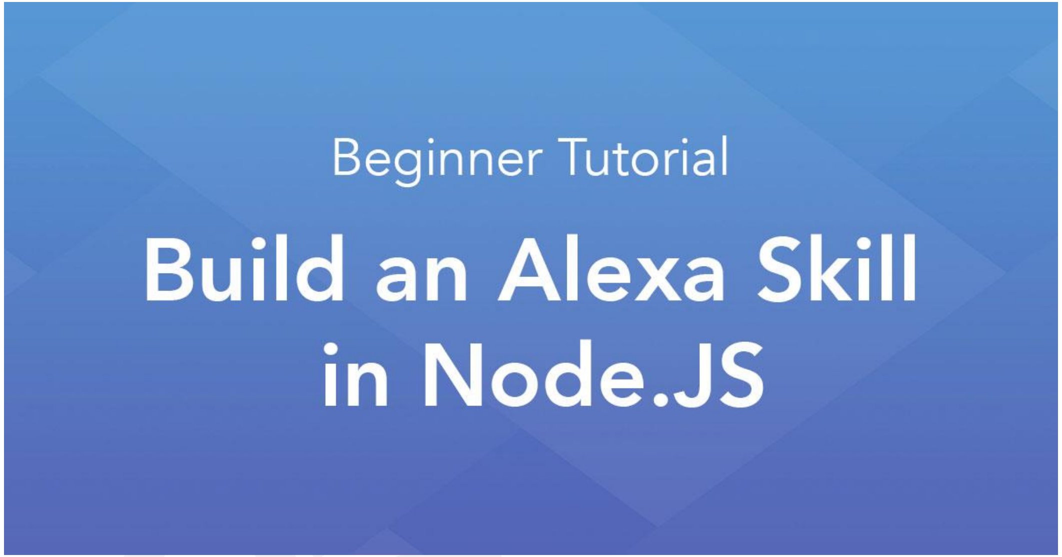
Please select one of the three methods below for the web service:

○ My development endpoint has a certificate from a trusted certificate authority

● My development endpoint is a sub-domain of a domain that has a wildcard certificate from a certificate authority

○ I will upload a self-signed certificate in X.509 format. Learn how to create a self signed certificate.

Node.js with Alexa



Beginner Tutorial

Build an Alexa Skill in Node.JS

Before We Start

- Be sure you have downloaded a right VM image:

https://github.com/falent/Alexa_universal_skill_template_VM

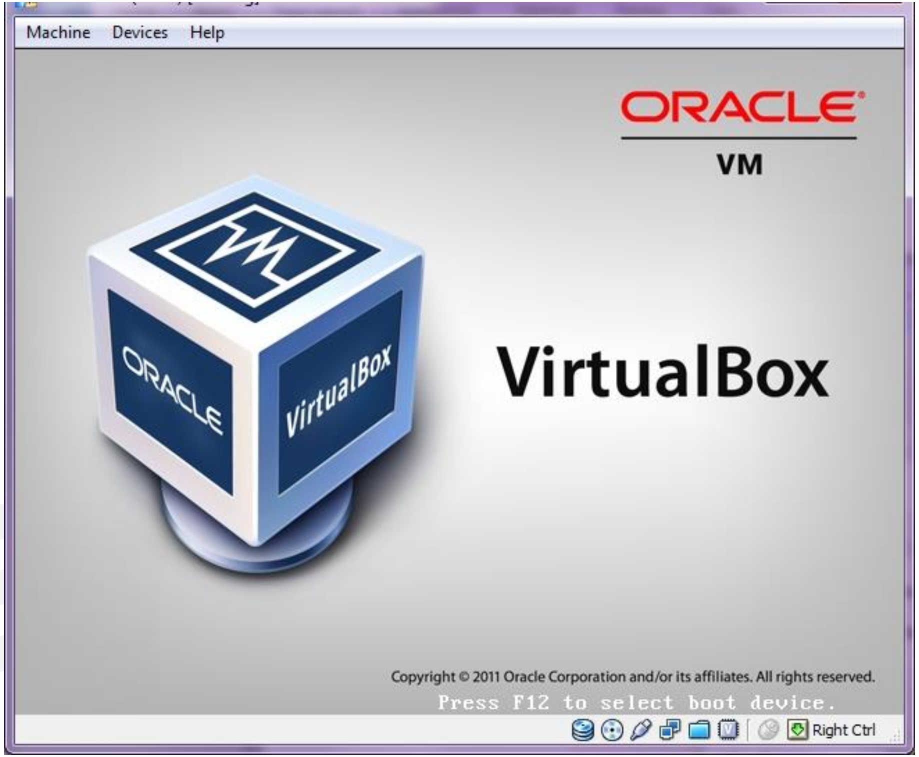
or copy it from my USB Stick.

1. Open https://www.npmjs.com/package/alexa-sdk Alexa SDK Documentation. It helps you a lot later ☺

2. Start Alexa OC Virtual Machine

3. Open eclipse or atom. The templates has been already included in the VM Desktop Directory: „Template“

Virtual Machine

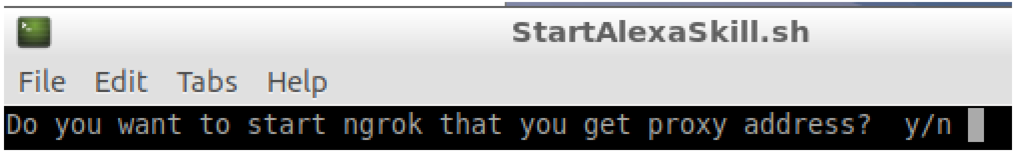


Copyright © 2011 Oracle Corporation and/or its affiliates. All rights reserved.

```
Press F12 to select boot device.
```

Test Your Skill Locally (VM 1)

1. After you start my VM. My script will show up with a question:



```
Do you want to start ngrok that you get proxy address?  y/n
```

We need to start ngrok if we want to test our skill locally. Please type y

2. „Do you want to updated script from Tomasz's GitHub" Please answer only once y , my script will update template code from my github

3. Please copy the last line you get from the script terminal. It is your endpoint url. In my case was: https://0031f3d4.ngrok.io your will be different but in this convention!

4. Your skill is ready to use. Please keep the tab opened the whole time.

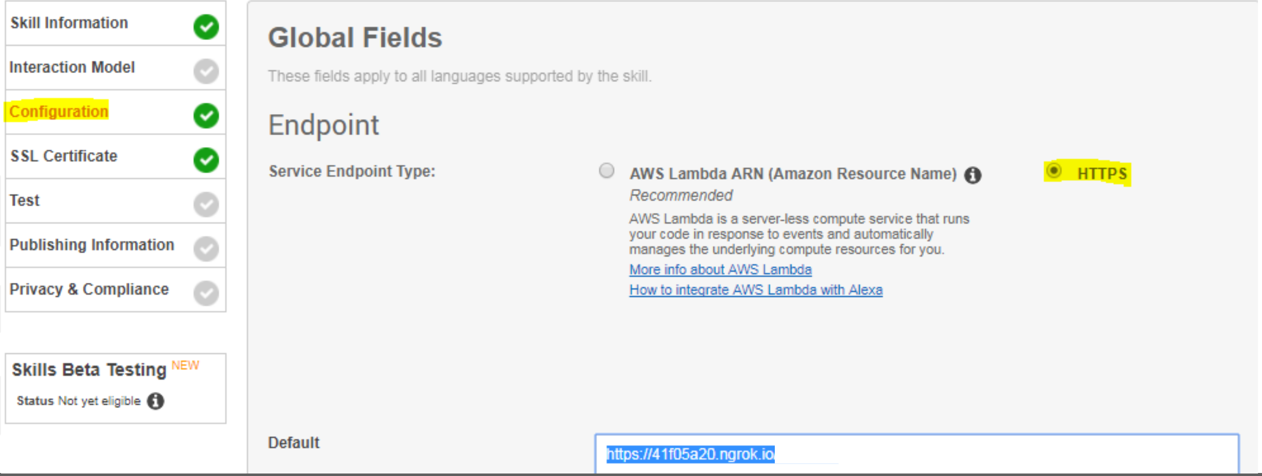
Test Your Skill Locally (VM 2)

5. Go to amazon developer portal and click edit in your skill

   https://developer.amazon.com/edw/home.html#/skills

6. Click configuration in your skill and put your individual https Endpoint (from step nr 3. Your address should look similar to mine)

   https://41f05a20.ngrok.io/



Global Fields

These fields apply to all languages supported by the skill.

Endpoint

Service Endpoint Type: ○ AWS Lambda ARN (Amazon Resource Name) *Recommended* ● HTTPS

AWS Lambda is a server-less compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

More info about AWS Lambda

How to integrate AWS Lambda with Alexa

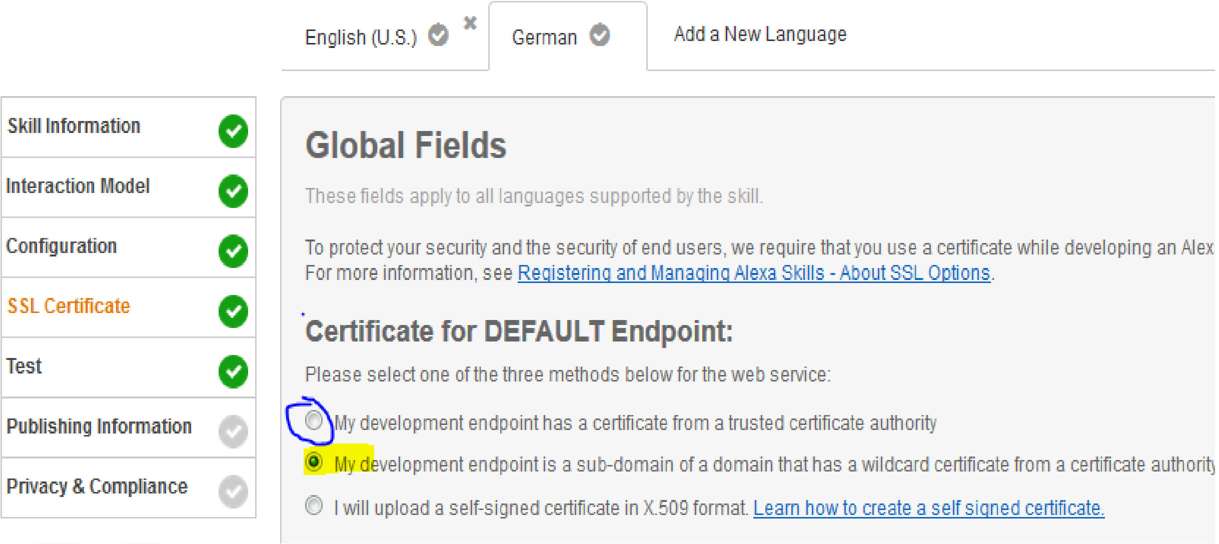
Default: https://41f05a20.ngrok.io

Test Your Skill Locally (VM 3)

7. Save your configuration and in the next step choose second option

   My development endpoint is a sub-domain of a domain that has a wildcard certificate from a certificate authority

   Click next.



Global Fields

These fields apply to all languages supported by the skill.

To protect your security and the security of end users, we require that you use a certificate while developing an Alex
For more information, see Registering and Managing Alexa Skills - About SSL Options.

Certificate for DEFAULT Endpoint:

Please select one of the three methods below for the web service:

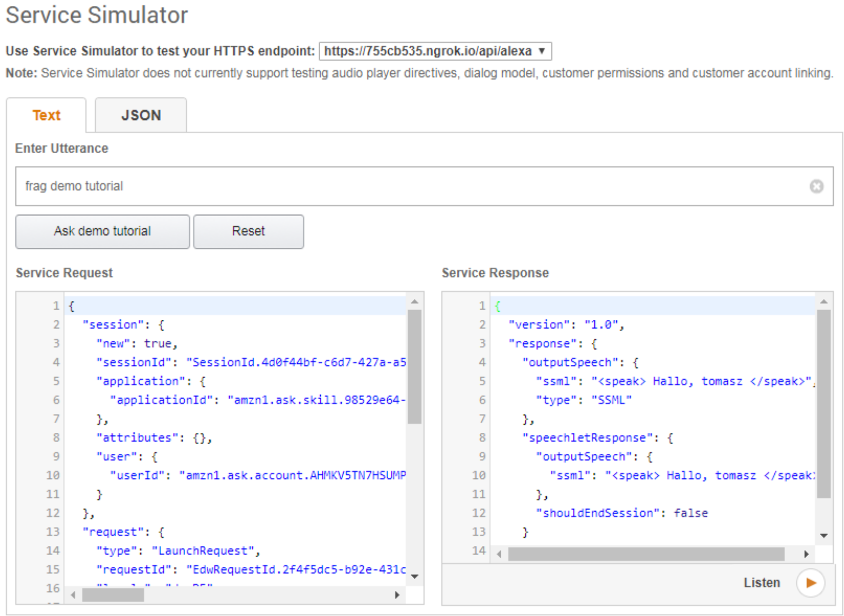
○ My development endpoint has a certificate from a trusted certificate authority

● My development endpoint is a sub-domain of a domain that has a wildcard certificate from a certificate authority

○ I will upload a self-signed certificate in X.509 format. Learn how to create a self signed certificate.

Test Your Skill Locally (VM 4)

8. You can test your skill in service Simulator. You should see Service Response on the right



Service Simulator

Use Service Simulator to test your HTTPS endpoint: https://755cb535.ngrok.io/api/alexa ▾

Note: Service Simulator does not currently support testing audio player directives, dialog model, customer permissions and customer account linking.

Enter Utterance

frag demo tutorial

Service Request

```
{
  "session": {
    "new": true,
    "sessionId": "SessionId.4d0f44bf-c6d7-427a-a5
    "application": {
      "applicationId": "amzn1.ask.skill.98529e64-
    },
    "attributes": {},
    "user": {
      "userId": "amzn1.ask.account.AHMKV5TN7HSUMP
    }
  },
  "request": {
    "type": "LaunchRequest",
    "requestId": "EdwRequestId.2f4f5dc5-b92e-431c
```

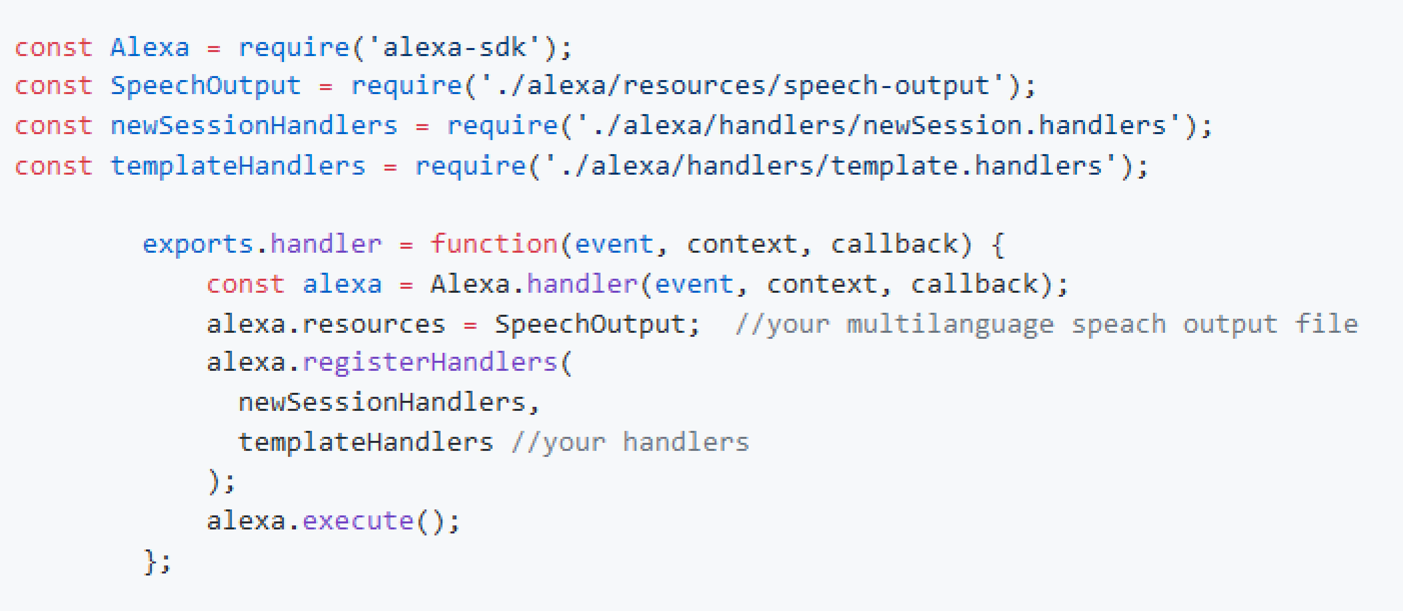
Service Response

```
{
  "version": "1.0",
  "response": {
    "outputSpeech": {
      "ssml": "<speak> Hallo, tomasz </speak>",
      "type": "SSML"
    },
    "speechletResponse": {
      "outputSpeech": {
        "ssml": "<speak> Hallo, tomasz </speak
      },
      "shouldEndSession": false
    }
```

You can also use Docker

Please follow instructions in readMe file:
https://github.com/falent/Alexa_universal_skill_template

Set Entry Point (this example is for AWS)



```
const Alexa = require('alexa-sdk');
const SpeechOutput = require('./alexa/resources/speech-output');
const newSessionHandlers = require('./alexa/handlers/newSession.handlers');
const templateHandlers = require('./alexa/handlers/template.handlers');

        exports.handler = function(event, context, callback) {
            const alexa = Alexa.handler(event, context, callback);
            alexa.resources = SpeechOutput;  //your multilanguage speach output file
            alexa.registerHandlers(
              newSessionHandlers,
              templateHandlers //your handlers
            );
            alexa.execute();
        };
```

Please compare it to HTTP solution:
https://github.com/falent/Alexa_universal_skill_template/blob/master/dist/app/src/index.js What are differences?

Example Conversation for us

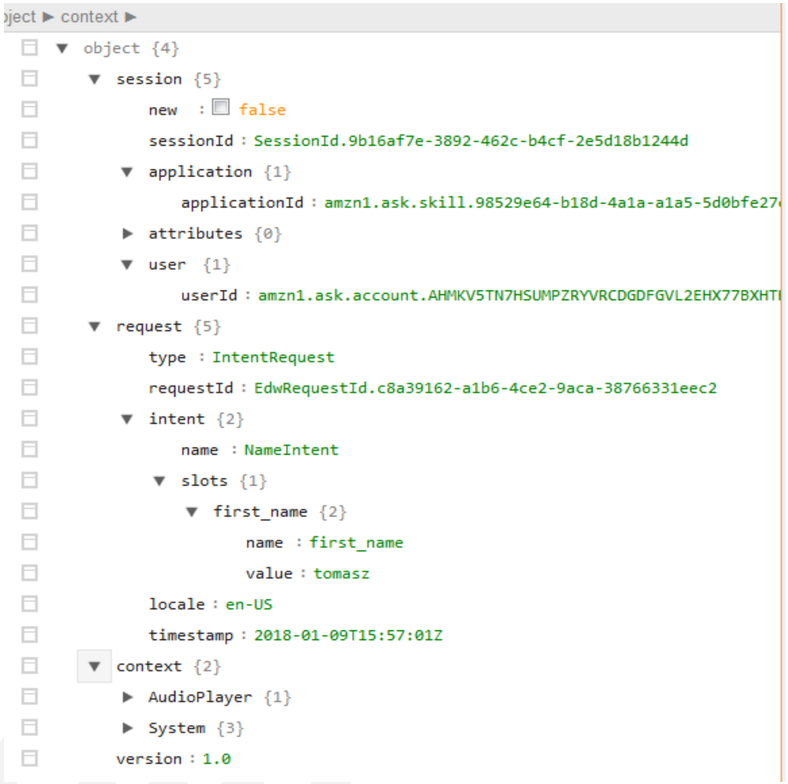
Alexa: „What is your name?“

Me: „Tomasz“

Alexa: „Hopla, what a pleasure Tomasz“

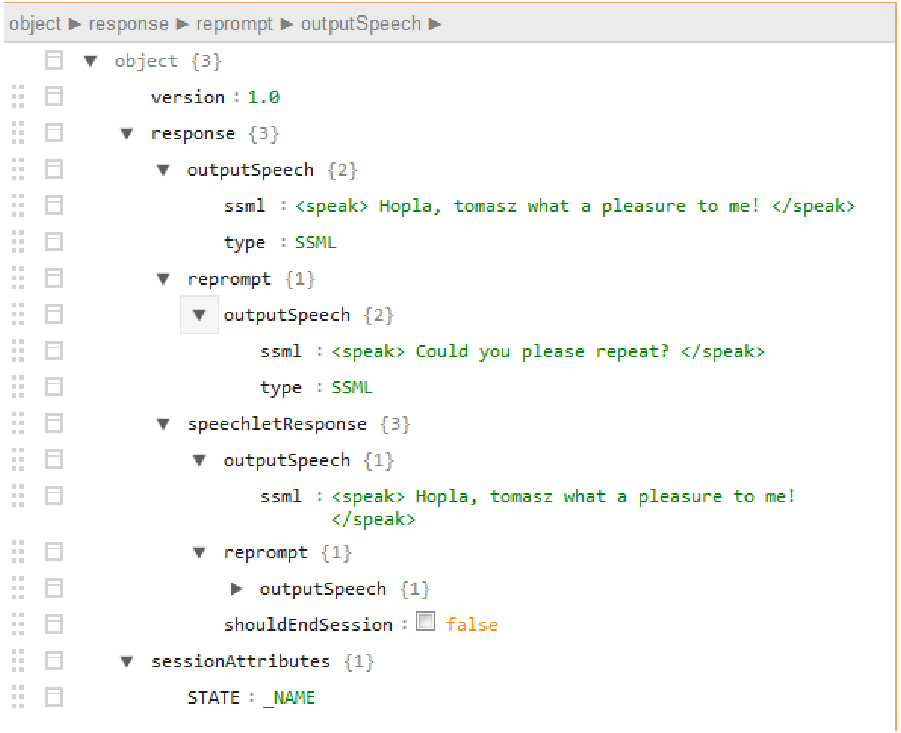
Example Conversation for Alexa

Service Request (Tomasz)



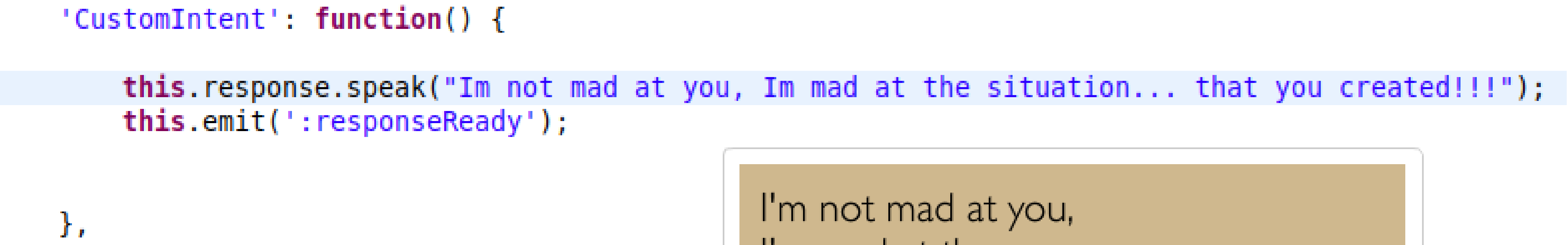
```
ject ▶ context ▶
▼ object {4}
  ▼ session {5}
      new  : □ false
      sessionId : SessionId.9b16af7e-3892-462c-b4cf-2e5d18b1244d
    ▼ application {1}
        applicationId : amzn1.ask.skill.98529e64-b18d-4a1a-a1a5-5d0bfe27
    ▶ attributes {0}
    ▼ user  {1}
        userId : amzn1.ask.account.AHMKV5TN7HSUMPZRYVRCDGDFGVL2EHX77BXHT
  ▼ request {5}
      type  : IntentRequest
      requestId : EdwRequestId.c8a39162-a1b6-4ce2-9aca-38766331eec2
    ▼ intent {2}
        name  : NameIntent
      ▼ slots {1}
        ▼ first_name {2}
            name  : first_name
            value : tomasz
      locale : en-US
      timestamp : 2018-01-09T15:57:01Z
  ▼ context {2}
    ▶ AudioPlayer {1}
    ▶ System {3}
    version : 1.0
```

Service Response (Alexa)



```
object ▶ response ▶ reprompt ▶ outputSpeech ▶
▼ object {3}
    version : 1.0
  ▼ response {3}
    ▼ outputSpeech {2}
        ssml  : <speak> Hopla, tomasz what a pleasure to me! </speak>
        type  : SSML
    ▼ reprompt {1}
      ▼ outputSpeech {2}
          ssml  : <speak> Could you please repeat? </speak>
          type  : SSML
    ▼ speechletResponse {3}
      ▼ outputSpeech {1}
          ssml  : <speak> Hopla, tomasz what a pleasure to me!
                  </speak>
      ▼ reprompt {1}
        ▶ outputSpeech {1}
        shouldEndSession : □ false
  ▼ sessionAttributes {1}
      STATE : _NAME
```

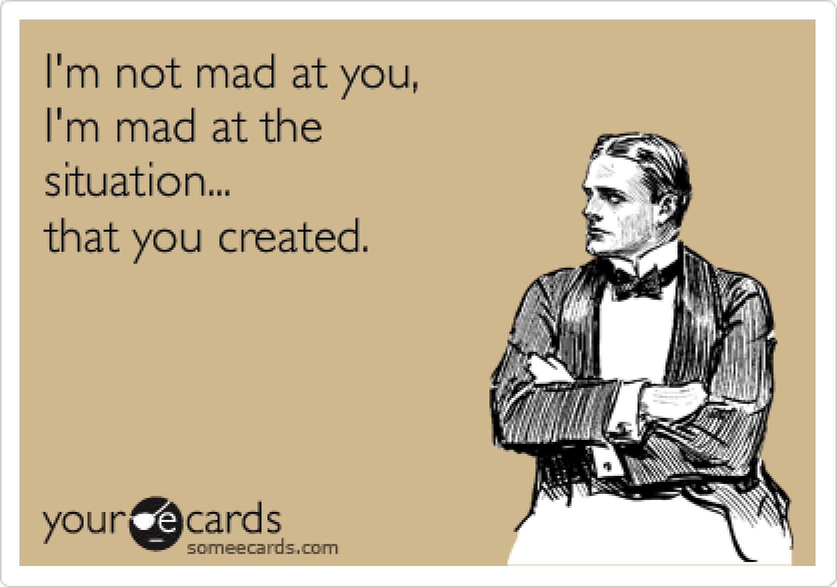
Custom Intents. Tell what you want and end!



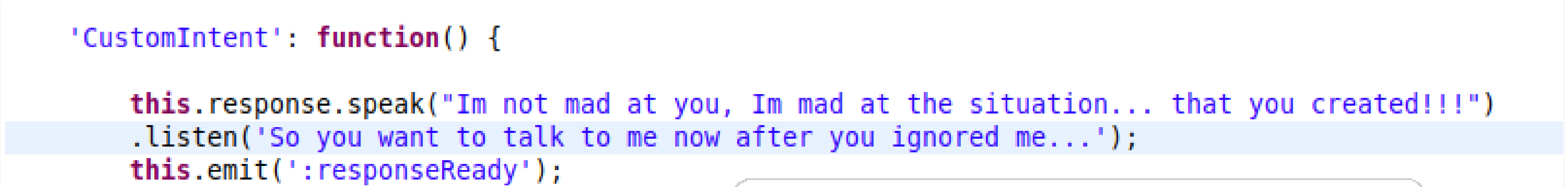
```
'CustomIntent': function() {

    this.response.speak("Im not mad at you, Im mad at the situation... that you created!!!");
    this.emit(':responseReady');


},
```



Custom Intents. Tell and expect somethig from my user!



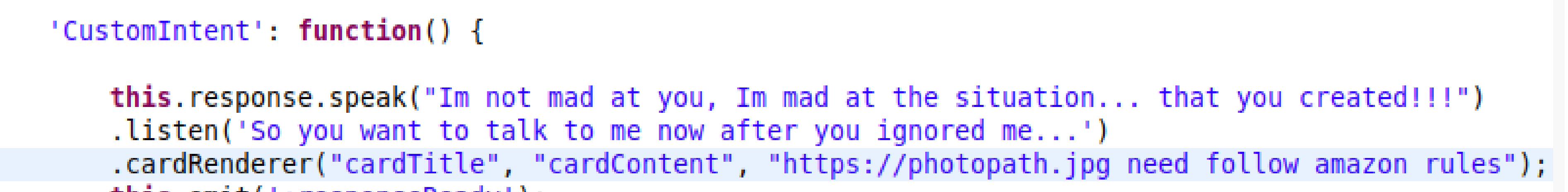
```
'CustomIntent': function() {

    this.response.speak("Im not mad at you, Im mad at the situation... that you created!!!")
    .listen('So you want to talk to me now after you ignored me...');
    this.emit(':responseReady');


},
```

So you want to talk to me now after you ignored me and passed me over for someone else?
someecards
user card

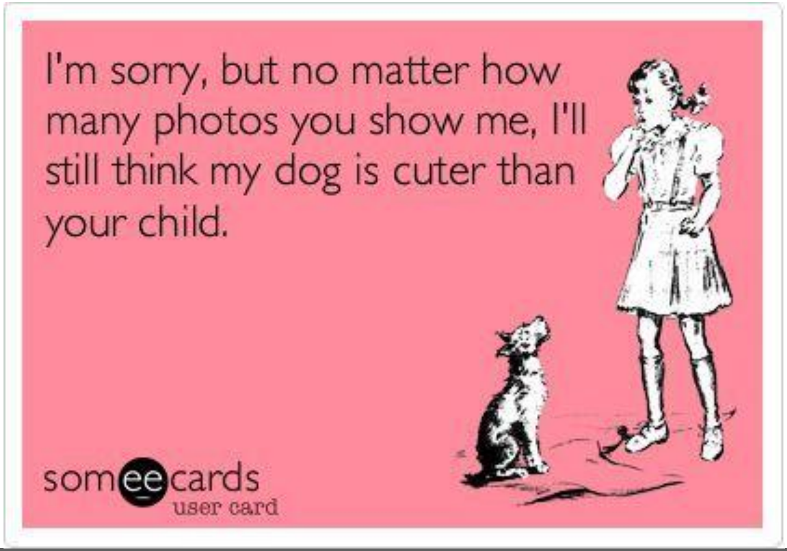
Custom Intents. Tell and expect somethig from my user! Show him what you have as a card!



```
'CustomIntent': function() {

    this.response.speak("Im not mad at you, Im mad at the situation... that you created!!!")
    .listen('So you want to talk to me now after you ignored me...')
    .cardRenderer("cardTitle", "cardContent", "https://photopath.jpg need follow amazon rules");
    this.emit(':responseReady');


},
```


I'm sorry, but no matter how many photos you show me, I'll still think my dog is cuter than your child.
someecards
user card

More more more examples…. In a short alexa bible ;)

Persisting Skill Attributes through DynamoDB

https://www.npmjs.com/package/alexa-sdk

Device Address Service

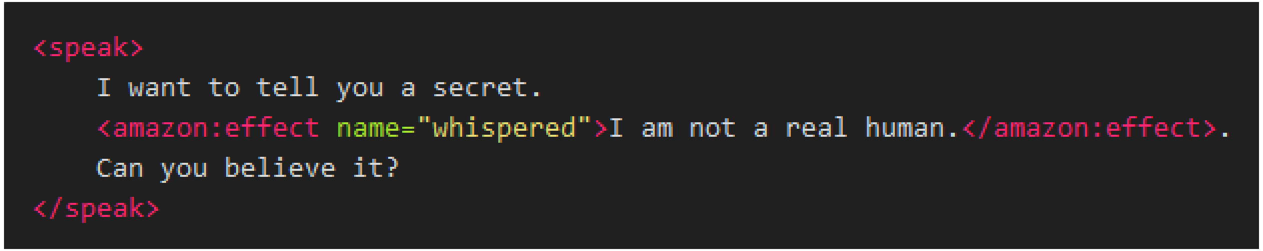
Adding Multi-Language Support for Skill

Playback Controller Interface

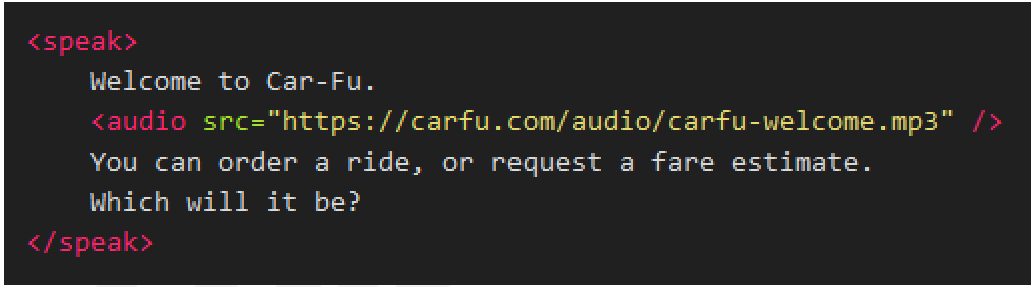
Skill State Management

Speech Synthesis Markup Language

https://developer.amazon.com/de/docs/custom-skills/speech-synthesis-markup-language-ssml-reference.html#amazon-effect

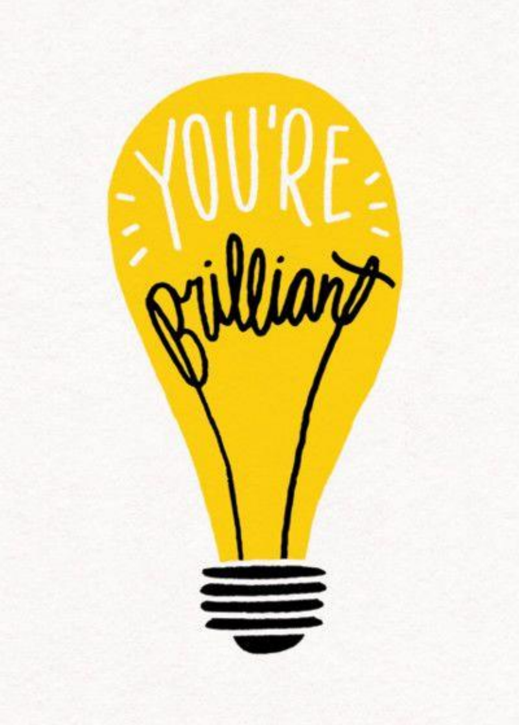


```
<speak>
    I want to tell you a secret.
    <amazon:effect name="whispered">I am not a real human.</amazon:effect>.
    Can you believe it?
</speak>
```



```
<speak>
    Welcome to Car-Fu.
    <audio src="https://carfu.com/audio/carfu-welcome.mp3" />
    You can order a ride, or request a fare estimate.
    Which will it be?
</speak>
```

Congratulation, let‘s start to go step by step through my sample code!



Please clone First Example

https://github.com/falent/Alexa_Event_Exercises

And lets code together :)

API with Alexa Exercise (1)

You know how to program Alexa or at least you can use my template

I prepared for you a simple API REST Server. Imagine you need to read temperature from a given conference room using alexa (1 Intent) and try to change temperature in that room. Your user need to tell alexa this value

You will program in 2 person team. Every time will get one virtual conference room with id

API with Alexa Exercise (2) GET

Example to read all values from your conference rooms

https://a049d2cb.ngrok.io/api/conferenceRoom/5a54016c5038c071a3efcd53

API with Alexa Exercise (2) PUT

Operation: PUT

Headers: Content-Type: application/json

Body: {

{"temperature": 10}

}

https://a049d2cb.ngrok.io/api/conferenceRoom/5a54016c5038c071a3efcd53

Please don‘t be quiet, coding session has been just started…

Creating an Alexa Skill

Voice User Interface + Programming Logic

Some References for Skill Developing

1. https://www.npmjs.com/ - npm is the package manager for JavaScript and the world’s largest software registry
2. https://github.com/alexa/alexa-cookbook - lots of examples how to build skills in Node.js
3. https://git.opitz-consulting.de/projects/ALEXAWIM - Christian Ochsenkühn and my work for Opitz
4. https://www.tutorialspoint.com/nodejs/ - Tutorial
5. https://developer.amazon.com/documentation - Official Alexa Doc
6. https://developer.amazon.com/de/alexa-skills-kit/tutorials - Official Amazon tutorials
7. https://www.youtube.com/watch?v=vsEaGjPPLqM - Youtube videos
8. https://www.codecademy.com/learn/learn-alexa-skills-kit - A complete "from the basics" intro to Alexa and Lambda

Deploying Your Skill to Heroku (1)

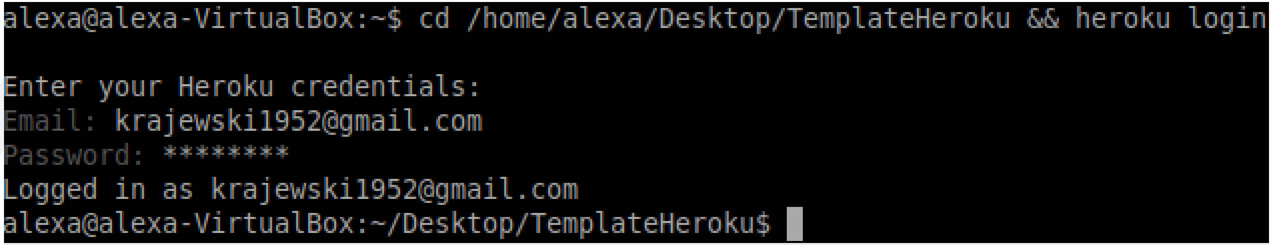
You can read about heroku: https://en.wikipedia.org/wiki/Heroku

1. You need to register https://signup.heroku.com/ :)

2. Open terminal in your OC Alexa VM and go to your skill directory

   cd /home/alexa/Desktop/Template/Alexa_universal_skill_template_VM && heroku login

3. Enter your heroku credentials



```
alexa@alexa-VirtualBox:~$ cd /home/alexa/Desktop/TemplateHeroku && heroku login

Enter your Heroku credentials:
Email: krajewski1952@gmail.com
Password: ********
Logged in as krajewski1952@gmail.com
alexa@alexa-VirtualBox:~/Desktop/TemplateHeroku$
```

`Be sure your project is fresh and ready :) rm -rf .git`

Deploying Your Skill to Heroku (2)

4. Create a new heroku app. Please save your app name. I got “rocky-river-24493” you will get another random funny name

```
cd /home/alexa/Desktop/TemplateHeroku && heroku apps:create --region eu
```

5. Init a new git app

```
git init
```

6. Connect your app to heroku

```
heroku git:remote –a <your app name from step 4>
```

5. Set app to dev

```
heroku config:set NPM_CONFIG_PRODUCTION=false
```

6. Add files to git reposidory

```
git add . && git commit -m "my first commit"
```

Deploying Your Skill to Heroku (3)

7. Push your files

    git push heroku master

8. Set your app to be a web app

    heroku ps:scale web=1

8. Create a mongoDB to your app. Go to page:

    https://elements.heroku.com/addons/mongolab

8. Click install and choose your app name from step 4.

    https://elements.heroku.com/addons/mongolab

9. Choose sandbox. It is free and you have around 500 MB capacity. Click provision

Deploying Your Skill to Heroku (4)

10. In your terminal in VM, get your heroku mongoDB address

```
heroku config:get MONGODB_URI
```

11. Copy your mongodb and paste to your /home/alexa/Desktop/TemplateHeroku/dist/app/routes/alexa/models/user.js

f.E

```
mongodb://heroku_6b9rh667:f7bcbsgbkcildjoo22rd3vpmmr@ds133465.mlab.com:33465/heroku_6b9rh667
```

12. Add your changes in heroku and push it

```
git add . && git commit -m "I added my personal Mongodb" && git push heroku master
```

Deploying Your Skill to Heroku (5)

13. Check if your app works

```
heroku logs --tail
```

14. Put your heroku app address to your amazon developer Alexa configuration tab

    https://<your heroku app name>.herokuapp.com/

    In my case it is:

    https://rocky-river-24493.herokuapp.com/

Questions?

Thanks for your attention!

Tomasz Krajewski

Associated Developer

Weltenburgerstarsse 4,
51647 München

Tomasz.krajewski@opitz-consulting.com
+49 173 5479-333

WWW.OPITZ-CONSULTING.COM

@OC_WIRE

OPITZCONSULTING

opitzconsulting

opitz-consulting-bcb8-1009116