

Einrichtung Amazon Account zur Nutzung eines Echo' Dot's

Stand: 01.12.2016, Version 2.1

Vorabinfo:

- 1) Bestimmte Installationsanweisungen kommen aus dem Fhem- Forum (<http://www.fhemwiki.de/wiki/Alexa-Fhem>). Diese sind hier zusammengefasst und ggf. erweitert, erneuert (gab Änderungen bei Amazon!!).
- 2) **WICHTIG:** Macht euch eine einfache Textdatei auf, worin ihr eure eigenen wichtigen ID's eintragt!!!
- 3) Wenn man bei Amazon das AWS- Konto nutzen will, dann muss man bei der Anmeldung eine Kreditkarte bzw. die entsprechenden Kenndaten haben.
- 4) Diese Anleitung ist einmal für die Nutzung in einen Gira HS geschrieben, also sind entsprechen ab und an Beispiele zum LBS des Gira HS vorhanden.
- 5) Des weiteren muss eine z.B. von <https://account.dyn.com/dns/dyndns/> DynDns Adresse vorhanden sein.

1) Die Installation

Die Grundvoraussetzung ist ein gültiges Amazon- Konto.
Neben dem wahrscheinlich schon vorhandenen Amazon- Konto muss man sich bei zwei weiteren Amazon- Diensten anmelden:

Dienst 1: Das kostenlose Amazon Developer Konto

Dienst 2: Das kostenlose Amazon AWS-Konto¹

Die Kommunikation zwischen den Diensten und z.B. dem Gira HS kann man sich dann wie folgt vorstellen:

Alexa <-> Amazon Developer Konto <-> Amazon AWS-Konto (GiraHS.js Lambda Function)
<-> über Portforwarding im eigenen Router <-> in den Gira HS, bzw. zum Logikbaustein

1.1) Dienst 1: Grundanmeldung beim Amazon- Developer Konto

Als erstes müssen die Anmeldungen/ Einrichtungen bei Amazon erfolgen.

Die erfolgt über die Webseite: <https://developer.amazon.com/>

Info: Wenn man ein gültiges Amazon Konto hat, dann kann man sich auch damit einloggen.

Also Anmelden bzw. später einloggen:

Info: Bei der Anmeldung mit einem vorhandenen Amazon Konto werden dann noch mal die Nutzerdaten abgeglichen

Achtung: Es müssen drei Felder zusätzlich, bzw. angepasst, ausgefüllt werden:

Registration

1. Profile Information 2. App Distribution Agreement 3. Payments

* Indicates a required field.

Country/Region * Germany

First name *

Last name *

Email address *

Phone number *
e.g. 212-555-1212, +44 0161 715 3369

Fax number

Developer name or company name *
Displayed on your apps at Amazon.com Required field

Developer description
Maximum characters: 4000, Remaining: 4000

Address 1 * Required field

Address 2

City *

State/Province/Region * Required field

Zip code/Postal code * **NOTE**

Customer support email address

Customer support phone

Customer support website

Cancel Save and Continue

Danach bestätigen wir das mit **“Save and Continue”**

Dann noch die weiteren Registerkarten akzeptieren:

1. Profile Information 2. App Distribution Agreement 3. Payments

English | 中文 (Chinese)* | 日本語 (Japanese)*

Und ich denke nicht, dass wir irgendwas verkaufen wollen:

1. Profile Information 2. App Distribution Agreement 3. Payments

* Indicates a required field.

Do you plan to monetize apps by charging for apps or selling in-app items? *

☒ No ☐ Yes

Do you plan to monetize apps by displaying ads from the Amazon Mobile Ad Network or Mobile Associates? *

☒ No ☐ Yes

Note: You may still monetize later if you select "No" by entering payment and tax information from the Settings menu.

Cancel Save and Continue

So damit haben wir die Anmeldung bei Amazon- Developer abgeschlossen.

1.1.1) Amazon-Developer Punkt: Apps & Services

Nun geht es nach der Anmeldung in den Bereich APPS & SERVICES:

amazon // DEVELOPER CONSOLE

DASHBOARD APPS & SERVICES ALEXA REPORTING SUPPORT DOCUMENTATION SETTINGS

My Apps App Testing Service Promotions Security Profiles Login with Amazon Dash Replenishment Service Alexa New GameCircle PC/Mac & Web Instant Access

Tester Management Advertise Your App Mobile Ads

Introducing **underground** apps & games

Get Paid for Every Customer, Every Minute.

Does my app qualify?
How do I get started?
How much can I earn?
Learn More

0 apps under this account

Nothing Found

Add a New App

Im Bereich APPS & SERVICES suchen wir den Punkt „Security Profiles:

APPS & SERVICES ALEXA REPORTING

App Testing Service Promotions **Security Profiles**

Jetzt legen wir ein neues Security Profil an:

Security Profile Management

[More Information](#)
[Login with Amazon](#)
[GameCircle](#)
[Device Messaging](#)

Security Profiles

Use this tool to create security profiles for your apps. A profile consists of OAuth2 credentials and API keys. Each profile can be shared by one or more apps. To use some features, your app must securely identify itself with Amazon. Use this tool to create, view, or edit a security profile.

Create a New Security Profile

Info: Man kann mehrere Sicherheitsprofile für unterschiedliche Anwendungen erstellen.

z.B. für verschiedene Apps oder Nutzungen.

Da das ja für einen Gira HS- Logikbaustein gedacht ist, nehme ich z.B.:

Security Profile Name: Gira HS

Security Profile Description: Sicherheitsprofil für den Gira Homeserver Logikbaustein

Nach der Bestätigung und Speicherung kannst Du die wichtigen Kenndaten (z.B. Client ID und Client Secret) sehen:

Security Profile Management

[More Information](#)
[Login with Amazon](#)
[GameCircle](#)
[Device Messaging](#)

Gira HS - Security Profile

General Android/Kindle Settings iOS Settings

These settings apply to all the apps using this security profile. Your security profile credentials — Client ID and Client Secret — allow your app to securely identify itself to Amazon services. [Learn More](#)

Security Profile Name	Gira HS
Security Profile Description	Sicherheitsprofil für den Gira Homeserver Logikbaustein
Security Profile ID	[REDACTED]
Client ID	[REDACTED]
Client Secret	[REDACTED]

Edit

Delete Security Profile

Jetzt wird im Fenster oben rechts auf: **Login with Amazon** geklickt:

[More Information](#)
[Login with Amazon](#)
[GameCircle](#)
[Device Messaging](#)

Es erscheint das folgende Fenster:

Sign Up

Auf der neu geladenen Seite klickst du dann auf **"Sign up"**

Wähle anschließend im Dropdown Menü das vorher angelegte Profil (bei diesem Beispiel Gira HS) aus und bestätige das anschließend mit *Confirm*

Login with Amazon

Login with Amazon allows users to login to registered third party websites or apps ('clients') using their Amazon user name and password. Clients may ask the user to share some personal information from their Amazon profile, including name, email address, and zip code. To get started, select an existing Security Profile or create a new Security Profile. [Learn More](#)

The screenshot shows the 'Select a Security Profile' dropdown menu. The 'Gira HS' option is selected and highlighted in blue. A blue arrow points from this option to the 'Confirm' button in the subsequent screenshot.

Im folgenden Fenster gibst du dann die Adresse:

The screenshot shows the 'Enter Consent Screen Information' form. The 'Consent Privacy Notice URL' field is filled with 'https:// or http://'. The 'Consent Logo Image' field has an 'UPLOAD IMAGE' button. The 'Save' button is highlighted in orange.

The screenshot shows the 'Enter Consent Screen Information' form with the 'Consent Privacy Notice URL' field filled with 'https://www.amazon.com/gp/help/customer/display.html?nodeId=468496'. The 'Save' button is highlighted in orange.

<https://www.amazon.com/gp/help/customer/display.html?nodeId=468496> ein und bestätigst die Eingabe mit *Save*

Klicke dann bei dem neu angelegten Eintrag auf der rechten Seite auf das Zahnrad und wähle *Web Settings* aus:

Login with Amazon

Login with Amazon allows users to login to registered third party websites or apps ('clients') using their Amazon user name and password. Clients may ask the user to share some personal information from their Amazon profile, including name, email address, and zip code. To get started, select an existing Security Profile or create a new Security Profile. [Learn More](#)

The screenshot shows the 'Login with Amazon' configuration page. The 'Manage' button is highlighted in orange. A blue arrow points from the 'Manage' button to the 'Web Settings' dropdown menu, which is open and shows 'Web Settings' as the selected option.

Im neu geladenen Fenster klicken wir auf *"Edit"*:

Security Profile Management

The screenshot shows the 'Gira HS - Security Profile' page. The 'Web Settings' tab is selected. The 'Edit' button is highlighted in orange. A blue arrow points from the 'Edit' button to the 'Edit' button in the subsequent screenshot.

Nun erscheint ein Fenster wo wir die Webseiten eintragen müssen die sich sozusagen zurückmelden dürfen. Dazu ist ein Informationsaustausch vom **Amazon Developer Konto** <-> **Amazon AWS-Konto** notwendig

Info: Die Verbindung des **Amazon Developer Konto** <-> **Amazon AWS-Konto** ist ein bisschen umständlich, da man Informationen von der jeweiligen Gegenseite eintragen muss.
Also erst die eine Seite erstellen, dann die andere und dann nochmal zurück zur ersten um die fehlenden Infos zu ergänzen.

Also legen wir in **“Allowed Return URLs”**, also die „Erlaubten Rücksende URL's, die zwei Seiten vom **Amazon Developer Konto** an:

Seite 1: <https://pitangui.amazon.com/api/skill/link/xxx>

Seite 2: <https://layla.amazon.com/api/skill/link/xxx>

ACHTUNG: **xxx** sind die Daten die beim Erstellen des Skill Kits unter:
Reiter: Alexa -- Gira HS – Logikbaustein – Configuration -- **Redirect Urls**
...ergänzt werden müssen. (siehe Seite 6, da „entstehen die Werte“)

So sieht das dann in etwas aus (hier noch mit den drei Seiten aus dem Fhem Forum!!):

Gira HS - Security Profile

General Web Settings Android/Kindle Settings iOS Settings

To use Login with Amazon with a website, you must specify either an allowed JavaScript origin (for the Implicit grant) or an allowed return URL (for the Authorization Code grant). If you are using Pay with Amazon, you must specify an allowed JavaScript origin. [Learn More](#)

Allowed Origins ?
[Add Another](#)

Allowed Return URLs ?

[Add Another](#)

[Cancel](#) [Save](#)

Wir beenden die Einrichtung wieder mit **“Save”**.

Gira HS - Security Profile

General Web Settings Android/Kindle Settings iOS Settings

To use Login with Amazon with a website, you must specify either an allowed JavaScript origin (for the Implicit grant) or an allowed return URL (for the Authorization Code grant). If you are using Pay with Amazon, you must specify an allowed JavaScript origin. [Learn More](#)

Allowed Origins ?
[Add Another](#)

Allowed Return URLs ?

[Add Another](#)

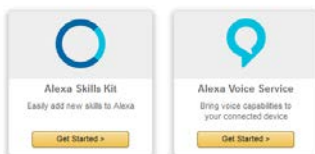
[Edit](#)

1.1.2) Amazon- developer Punkt: Alexa

Nun fügen wir dem Alexa Skills Kit eine neue Sprachfunktion zu.

Get started with Alexa

Add new voice-enabled capabilities using the Alexa Skills Kit, or add voice-powered experiences to your connected devices with the Alexa Voice Service.



Also ab auf: Alexa Skills Kit „Get Started“:



Es erscheint das Fenster worin wir eine neue Sprachfunktion anlegen:

Building Alexa Skills with the Alexa Skills Kit

[Add a New Skill](#)

To learn more about building Alexa skills, see [Getting Started with the Alexa Skills Kit](#). To start building an Alexa skill for free using AWS Lambda, see [Creating an AWS Lambda Function for a Custom Skill](#). We encourage you to visit the [Alexa Developer Forum](#) to collaborate with Alexa team members and fellow Alexa developers.

Your skills		Skill Metrics				
Skill Name	Language	Skill Type	Modified	Status	Actions	

Mit [Add a New Skill](#) wird die neue Sprachfunktion angelegt

Es müssen die drei folgenden Kenndaten eingetragen/ gesetzt werden:

- 1) Skill Type: Smart Home Skill API
- 2) Language: German
- 3) Name: z.B. Gira HS - Logikbaustein

So sieht das dann aus:

Create a New Alexa Skill

Skill Information	<input checked="" type="checkbox"/>	Skill Type Define a custom interaction model or use one of the predefined skill APIs. Learn more <input type="radio"/> Custom Interaction Model <input checked="" type="radio"/> Smart Home Skill API <input type="radio"/> Flash Briefing Skill API
Interaction Model	<input checked="" type="checkbox"/>	
Configuration	<input checked="" type="checkbox"/>	Language Language of your skill: German
SSL Certificate	<input checked="" type="checkbox"/>	
Test	<input checked="" type="checkbox"/>	Name Name of the skill that is displayed to customers in the Alexa app. Must be between 2-50 characters. Gira HS - Logikbaustein
Publishing Information	<input checked="" type="checkbox"/>	
Privacy & Compliance	<input checked="" type="checkbox"/>	

Save Next

Über **Next** werden die Eingabe bestätigt.

Dadurch werden auch die grünen Haken gesetzt/ angezeigt:

German ☒ Add New Language

Skill Information	<input checked="" type="checkbox"/>
Interaction Model	<input checked="" type="checkbox"/>
Configuration	<input checked="" type="checkbox"/>
Test	<input checked="" type="checkbox"/>
Publishing Information	<input checked="" type="checkbox"/>
Privacy & Compliance	<input checked="" type="checkbox"/>

Save Submit for Certification Next

Your skill uses Smart Home Skill API
The interaction model is already built into the Smart Home Skill API and you can find the supported utterances [here](#). You just have to handle the directives sent by Alexa in your Smart Home adapter (Lambda function)

Den Punkt 2 "Interaction Model" wird gleich durch das drücken auf **Next** bestätigt.

Auf der Seite **Configuration** sind nun einige Eintragungen zu machen:

ACHTUNG: **xxx** Hier werden die Werte angelegt die weiter oben durch die drei Kreuze vorgehalten wurden.

Die entsprechenden Daten finden sich im Eintrag: **Redirect URLs (Optional)** und sehen z.B. so aus:

Redirect URLs (Optional)
The list of valid HTTPS redirection endpoints that could be requested during authorization to redirect the user back to after the authorization process.
[Learn More](#)

<https://layla.amazon.com/api/skill/link/M10...>
<https://pitangui.amazon.com/api/skill/link/M10...>

Info: Auch wenn hier nur zwei Webseiten angezeigt werden, so sind die **xxx** Werte für alle Webseiten gleich. Somit können wir die Schlüsselwerte in die eventuelle vorhandenen weiteren Webseiten vom Punkt weiter oben eintragen werden.

z.B. **M10**

Im Fhem Forum sind die drei Seiten vorhanden/ aufgeführt. Hier für das Beispiel des Gira HS Bausteins reichen die zwei Seiten!!!

Der Reihe nach:

A) Fest einzutragen Daten:


Eintrag 1: In **Authorization URL** gibt man die URL: <https://www.amazon.com/ap/oa> an.

Eintrag 2: Unter **Scope** wird ein neuer „Skillbenutzer“ angelegt. Eintrag: `profile:user_id`

Eintrag 3: In **Access Token URI** gibt man die URL: <https://api.amazon.com/auth/o2/token> an

Eintrag 4: In **Privacy Policy URL** gibt man die URL: <https://www.amazon.com/gp/help/customer/display.html?nodeId=468496> an

B) Fest individuell einzutragende Daten:

Eintrag 1: Bei **Service Endpoint Type:** muss man bei Europa den Haken setzen 
Dadurch entsteht ein Textfeld mit der Überschrift Europa

☐ North America ☒ Europe
Europe

Dieser Eintrag kann erst vervollständigt werden, wenn wir eine AWS Lambda Funktion² angelegt haben (wird weiter dann unter den Punkt AWS Lambda Funktion anlegen erklärt).

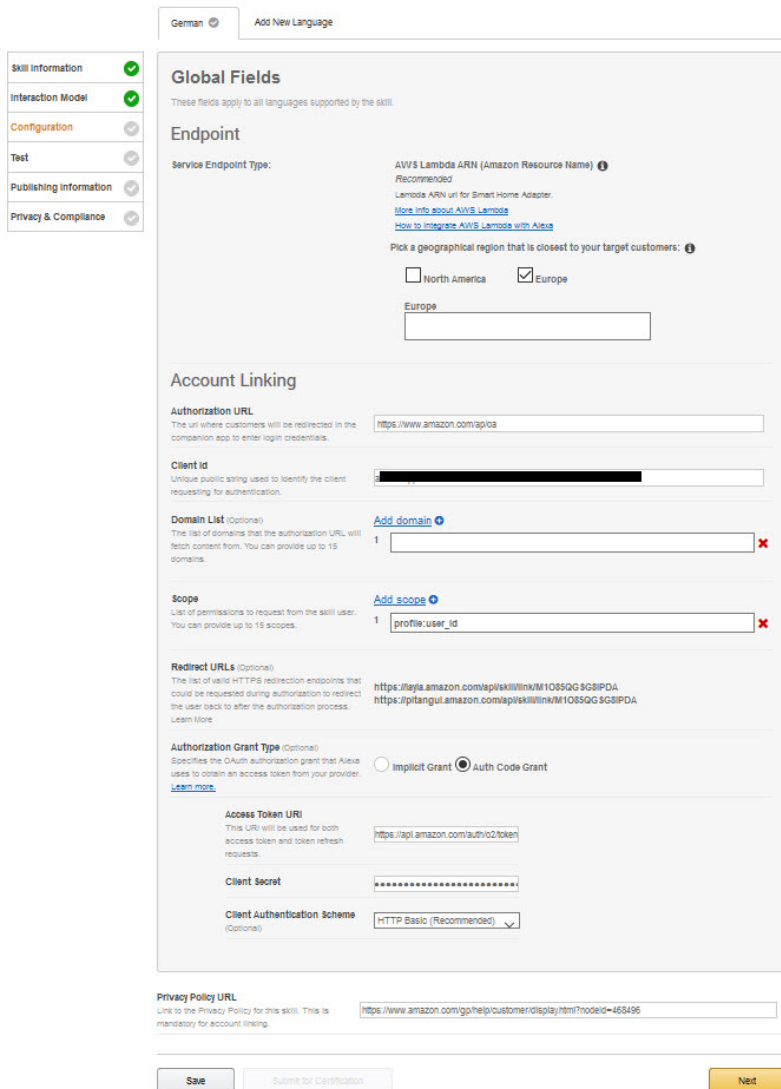
Eintrag 2: **Client Id** Die Daten hierfür haben wir weiter oben beim Gira HS – Security Profil bekommen, also aus dem Punkt **“Login with Amazon“**:

Client ID 

Eintrag 3: **Client Secret** ..dieser Wert ist auch bekannt und wurde weiter oben Gira HS – Security Profil erzeugt, also aus dem Punkt **“Login with Amazon“**:

Client Secret 

So sieht dann alles eingetragen aus:



The screenshot shows the 'Global Fields' configuration page in the AWS IAM console. On the left, there is a sidebar with navigation links: Skill Information (checked), Interaction Model (checked), Configuration (selected), Test, Publishing Information, and Privacy & Compliance (checked). The main content area is titled 'Global Fields' and contains several sections:

- Endpoint:** Service Endpoint Type is set to 'AWS Lambda ARN (Amazon Resource Name)'. Below it, there are checkboxes for 'North America' and 'Europe' (checked). A text field below the checkboxes contains the word 'Europe'.
- Account Linking:** Authorization URL is set to 'https://www.amazon.com/ap/oa'. Client Id is a redacted field. Domain List (Optional) has one entry with a red 'x' icon. Scope has one entry 'profile:user_id' with a red 'x' icon. Redirect URLs (Optional) are listed as 'https://alexa.amazon.com/api/skill/link/M1085QG9GBIPDA' and 'https://pitangul.amazon.com/api/skill/link/M1085QG9GBIPDA'. Authorization Grant Type is set to 'Auth Code Grant' (selected over 'Implicit Grant'). Access Token URI is 'https://api.amazon.com/auth/o2/token'. Client Secret is a redacted field. Client Authentication Scheme is set to 'HTTP Basic (Recommended)'. Privacy Policy URL is 'https://www.amazon.com/gp/help/customer/display.html?nodeId=465496'.

At the bottom, there are three buttons: 'Save', 'Submit for Certification', and 'Next'.

Achtung:

Hier konnte ich den Punkt erst nicht abschließen:

Es wurde immer der Fehler: **Error: Please provide a valid endpoint**


angezeigt. Denn man braucht erst die **ARN (Amazon Resource Name)** Nummer

Aber ganz oben auf der Seite steht die wichtige *Application Id*, die man zum Anlegen der Lambda Funktion braucht (also immer mit zwei Fenstern arbeiten und sozusagen zwei Schritte ausführen)



ACHTUNG:

Man benötigt hier eine **ARN (Amazon Resource Name)** Nummer, diese bekommt man aber erst Wenn man einen AWS Lambdadienst eingerichtet hat!!

Ziemlich blöd das einzurichten, also das Fenster offen lassen und sich weiter durcharbeiten!!

Mit  wird der dritte bearbeitungspunkt abgeschlossen

Es folgt folgendes Bild:

Die letzten zwei Punkte sind optional (also die Punkte: **Publishing Information**  und **Privacy & Compliance** )

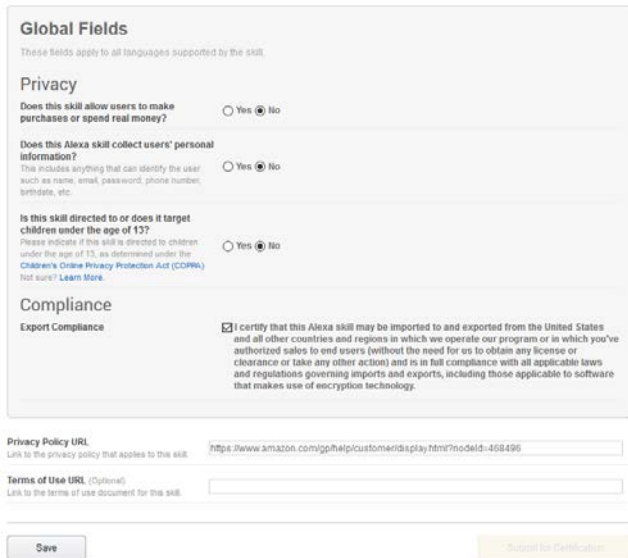
Diese zwei Punkte wären nur wichtig wenn man den Skill für alle Alexa User bei Amazon veröffentlichen wollte.

Dies hört sich im ersten Moment super an, dann könnten doch alle sich die komplizierte Einrichtung sparen?

Aber, nachdem es für Amazon-Echo jedoch keinen Cloud-Service von Gira für den Homeserver gibt, muss jeder Benutzer quasi seinen eigenen Cloud-Service einrichten.

Das einzige Gimick was man hier nutzen kann ist das hochladen eigener Icons die in der Alexa App beim Verknüpfen des Skills angezeigt werden.

Im letzten Punkt habe ich dann sicherheitshalber die **Privacy** auf No gesetzt.



Global Fields
These fields apply to all languages supported by the skill.

Privacy

Does this skill allow users to make purchases or spend real money? ☐ Yes ☒ No

Does this Alexa skill collect users' personal information?
This includes anything that can identify the user such as name, email, password, phone number, birthdate, etc. ☐ Yes ☒ No

Is this skill directed to or does it target children under the age of 13?
Please indicate if this skill is directed to children under the age of 13, as determined under the [Children's Online Privacy Protection Act \(COPPA\)](#).
Not sure? [Learn More](#). ☐ Yes ☒ No

Compliance

Export Compliance

☒ I certify that this Alexa skill may be imported to and exported from the United States and all other countries and regions in which we operate our program or in which you've authorized sales to end users (without the need for us to obtain any license or clearance or take any other action) and is in full compliance with all applicable laws and regulations governing imports and exports, including those applicable to software that makes use of encryption technology.

Privacy Policy URL
Link to the privacy policy that applies to this skill:

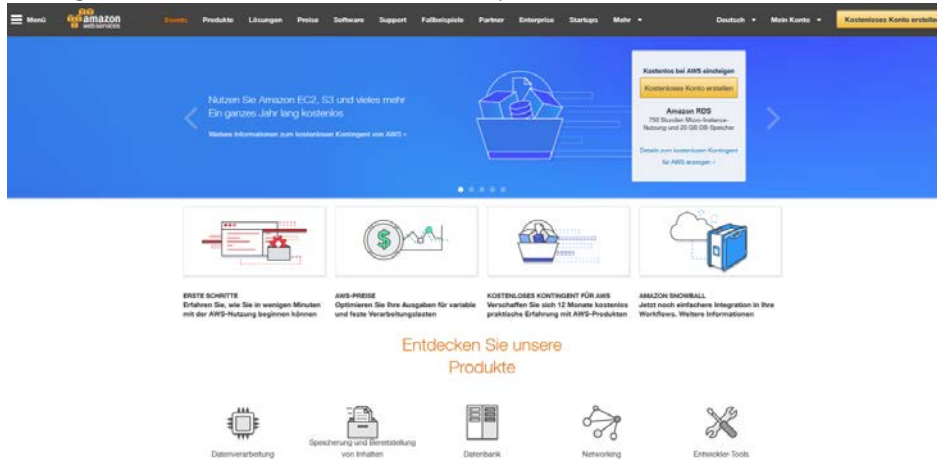
Terms of Use URL (Optional)
Link to the terms of use document for this skill:

Save **Submit for Certification**

1.2) Dienst 2: Grundanmeldung beim Amazon AWS-Konto

Nun wenden wir uns dem Punkt AWS- Konto zu.

Dazu gehen wir auf die Webseite: <https://aws.amazon.com/de/>



Jetzt gehen wir auf:

Hier können wir uns [Kostenloses Konto erstellen](#) Kontendaten einloggen, müssen diese aber vervollständigen:

Kontaktinformationen

☒ Firmenkonto ☐ Persönliches Konto

* Pflichtfelder

Vollständiger Name*

Firmenname*

Land*

Adresse*

Wohnung, Bürotage, Einheit, Gebäude, Etage usw.

Stadt*

Bundesstaat/Provinz/Region*

Postleitzahl*

Telefonnummer*

Sicherheitsprüfung ©

Bitte geben Sie die oben abgebildeten Zeichen ein

AWS-Kundenvertrag

☐ Klicken Sie auf dieses Kästchen, um zu bestätigen, dass Sie die Bedingungen des AWS-Kundenvertrags gelesen haben und ihnen zustimmen.

[DID aktualisieren](#)

[Konto erstellen und fortfahren](#)

Info: Die zwei 3- Buchstaben/ Zahlenkombinationen zusammenschreiben

Info: Haken setzen nicht vergessen

Dann auf [Konto erstellen und fortfahren](#) klicken um zum nächsten Fenster zu kommen

Nun kommt ein, für mich jedenfalls, großes Problem:

Man benötigt eine Kreditkarte!!! und muss die Daten entsprechend eintragen!!!

Zahlungsinformationen

Bitte geben Sie Ihre Zahlungsinformationen unten ein. Sie haben die Möglichkeit, eine große Auswahl von AWS-Produkten über das kostenlose Kontingent gratis zu testen. Ihre Kredit- oder Debitkarte wird nur für die Nutzung belastet, die nicht durch unser kostenloses Kontingent abgedeckt ist.

+ Häufig gestellte Fragen

Kreditkarten-/Debitkartennummer

Ablaufdatum

91 2016

Name des Karteninhabers

☒ Meine Kontaktadresse verwenden

(L)

☐ Neue Adresse verwenden

Fortfahren

Info zum Thema Anmeldung mit Kreditkarte, bzw. Infos vom Projektteam:

Im Augenblick lässt sich der Alexa Smart-Home-Skill nur mit einer AWS-Lambda Funktion verknüpfen. Auf der Konfigurationsseite für den Smart-Home-Skill steht zwar „AWS-Lambda Recommended“, aber eine andere Wahl hat man hier nicht.

Der AWS-Service ist als Dienst für Firmen gedacht, welche Ihre Web-Services in einer Cloud betreiben wollen. Also kein eigenes Rechenzentrum bzw. Server wollen die sie betreuen und warten müssen. Allerdings steht auf irgendeiner Alexa-Seite auch explizit, dass „Smart-Home-Enthusiasts“ (also wir ;-)) herzlich willkommen sind Ihre eigenen Skills zu entwickeln.

Nachdem wir von den angebotenen Diensten nur die Lambda-Funktionalität verwenden, haben wir im Monat 1 Millionen Anforderungen bzw. 3,2 Mio. Sekunden Rechenzeit pro Monat (bei 128 MB Speicherzuteilung) frei. Dieses freie Kontingent läuft auch nach einem Jahr nicht aus. Jede weitere 1 Mio Anforderungen kosten 0,20 USD. Wenn man die 3,2 Mio Sekunden Rechenzeit überschreitet kostet das GB/Sekunde 0,0001667 USD.



AWS Lambda

Datenverarbeitungsservice, der Ihren Code beim Eintreten bestimmter Ereignisse ausführt und automatisch die Datenverarbeitungsressourcen verwaltet

[Weitere Informationen »](#)

1 000 000 kostenlose Anforderungen pro Monat

Bis zu 3,2 Mio. Sekunden Rechenzeit pro Monat

Läuft nicht am Ende Ihres 12-monatigen kostenlosen AWS-Nutzungskontingents aus.

Um euch einen Überblick zu geben, hier die aktuellen Verbrauchswerte für mein AWS-Konto (zu finden unter Billing):

Wichtigste Services des kostenlosen Kontingents nach Nutzung			Alle anzeigen
Service	Aktueller Monat bis heute Nutzung/Grenze kostenloses Kontingent	Voraussichtlich bis zum Ende des aktuellen Monats Nutzung/Grenze kostenloses Kontingent	
Lambda - Requests	0.06% (621.00/1,000,000 Requests)	0.07% (665.36/1,000,000 Requests)	
Lambda - GB Second	0.02% (85.13/400,000 Second)	0.02% (91.21/400,000 Second)	

Also selbst bei der Monatsprognose ist mein Verbrauch noch weit unter 1% von dem was ich kostenlos verbrauchen darf.

Ich kann mir beim besten Willen nicht vorstellen, wie ein User über 100% kommen könnte - es sei den Alexa läuft „Amok.“

..danach erfolgt noch eine Identitätsüberprüfung

Identitätsüberprüfung

Um Ihre Identität zu überprüfen, werden Sie nach Eingabe Ihrer Telefonnummer umgehend von einem automatisierten System angerufen und aufgefordert, die angegebene PIN Nummer einzugeben.

1. Eine Telefonnummer angeben
Geben Sie Ihre Informationen unten ein, und klicken Sie auf die Schaltfläche 'Mich jetzt anrufen'.

Sicherheitsprüfung

Bitte geben Sie die oben abgebildeten Zeichen ein:

Landesvorwahl: Telefonnummer: Durchwahl:

[Mich jetzt anrufen](#)

2. Laufender Anruf

3. Identitätsüberprüfung abgeschlossen

Die läuft wie folgt ab:

- 1) Man wird über die angegebene Telefonnummer angerufen.
- 2) Man bekommt einen PIN den man über die Zifferntasten eingibt.

Info: Die automatische Stimme war in meinem Fall sogar in Deutsch!!

Wenn alles erfolgreich abgelaufen ist kommt man zum Punkt:

Support- Programm



Support-Programm

AWS Support bietet verschiedene Nutzungspläne, die Ihren Anforderungen gerecht werden. Alle Pläne bieten Kundenservice rund um die Uhr und geben Ihnen Zugriff auf Dokumentationen zu AWS, Whitepapers und Support-Foren. Damit Sie technischen Support in Anspruch nehmen können und Zugriff auf zusätzliche Ressourcen haben, die Ihnen helfen, Ihre AWS-Umgebung zu planen, bereitzustellen und zu optimieren, empfehlen wir Ihnen, einen Support-Plan auszuwählen, der Ihrer AWS-Nutzung angemessen ist.

Ein Programm auswählen

☒ **Basic**
Beschreibung: Kundenservice bei Konto- und Fakturierungsfragen und Zugang zu den AWS-Communityforen.
Preis: Inbegriffen

☐ **Developer**
Anwendungsfall: Erste Experimente mit AWS
Beschreibung: Ein primärer Kontakt kann technische Fragen im Support-Center stellen. Während der vor Ort geltenden Geschäftszeiten erhalten Sie innerhalb von 12 bis 24 Stunden eine Antwort auf Ihre Anfrage.
Preis: Ab 29 USD/Monat (je nach Nutzung)

☐ **Business**
Anwendungsfall: Nutzung von AWS in der Produktion
Beschreibung: Telefonischer und Chat-Support rund um die Uhr, Hilfe innerhalb 1 Stunde bei dringenden Support-Vorgängen und Hilfe bei der Nutzung von gängiger Drittanbieter-Software. Ungehinderter Zugang zu AWS Trusted Advisor zwecks Optimierung Ihrer AWS-Infrastruktur und Zugriff auf die AWS Support-API zur Automatisierung von Support-Vorgängen und zum Abrufen der Ergebnisse von Trusted Advisor.
Preis: Ab 100 USD/Monat (je nach Nutzung)

☐ **Enterprise**
Anwendungsfall: Geschäftskritische Verwendung von AWS
Beschreibung: Alle Funktionen des Business-Support-Plans sowie ein zugewiesener technischer Account-Manager (TAM), der proaktive Hilfestellung und bewährte Methoden bereitstellt, um bei der Planung, Entwicklung und Ausführung Ihrer AWS-Lösungen Unterstützung zu leisten, ein Support-Concierge, der Rechnungs- und Kontoanalyse und -unterstützung bietet, Zugriff auf Infrastructure Event Management, um die Einführung neuer Produkte und saisonale Angebote/Ereignisse zu unterstützen, und eine 15-minütige Reaktionszeit bei kritischen Fällen für den Support mit bevorzugter Bearbeitung von Fällen.
Preis: Ab 15.000 USD/Monat (je nach Nutzung)

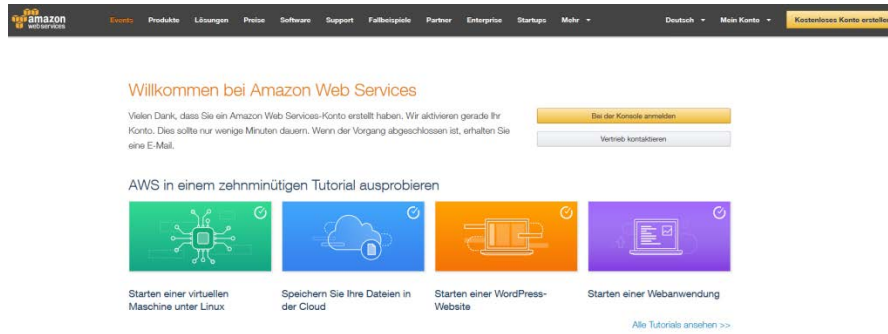
Wenn Sie sich für diese Option entscheiden, erhalten Sie innerhalb von 48 Stunden einen Anruf vom Kunden-Support. Bei diesem Anruf können Sie Ihre Anforderungen beschreiben und die Anmeldung abschließen. Sie haben Zugriff auf die Support-Ressourcen, sobald die Anmeldung abgeschlossen ist. Bis zu diesem Zeitpunkt entstehen Ihnen keine Kosten.

[Klicken Sie hier](#), um eine Übersicht aller vom AWS Support angebotenen Funktionen und Vorteile, einen Vergleich der verschiedenen Pläne und Preisbeispiele zu sehen.

[Weiter](#)

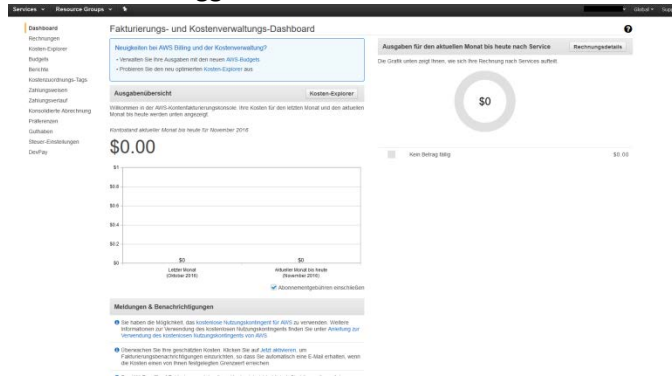
Hier habe ich den Kostenlosen Basic- Zugang genommen

Wenn das alles geklappt hat erschein folgendes:



...also es dauert etwas und man bekommt eine E-Mail.

Nach dem einloggen sieht das dann so aus:



1.2.1) Amazon AWS-Konto den Punkt: Lambda Funktion anlegen

Jetzt geht es endlich los.

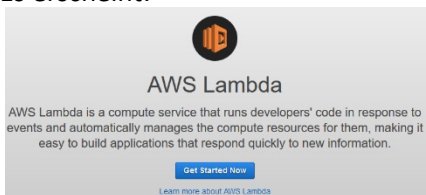
Durch da klicken auf  **Services** kommt man auf die Homeseite.

Console Home

Im Punkt AWS Service findet man unter Compute den Punkt Lambda:

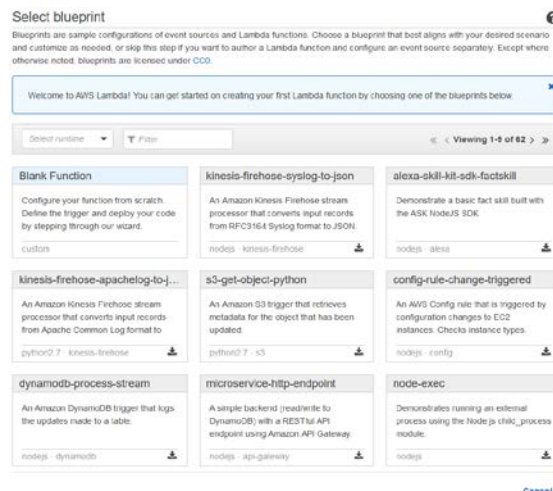
Auf diesen springen wir mal:

Es erscheint:



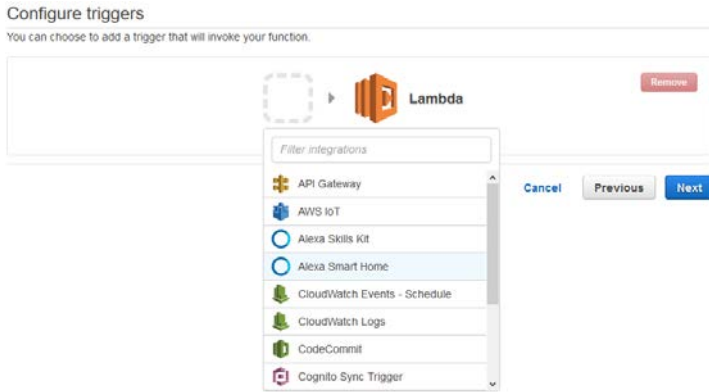
Mit  geht es weiter.

..es kommt das Fenster:



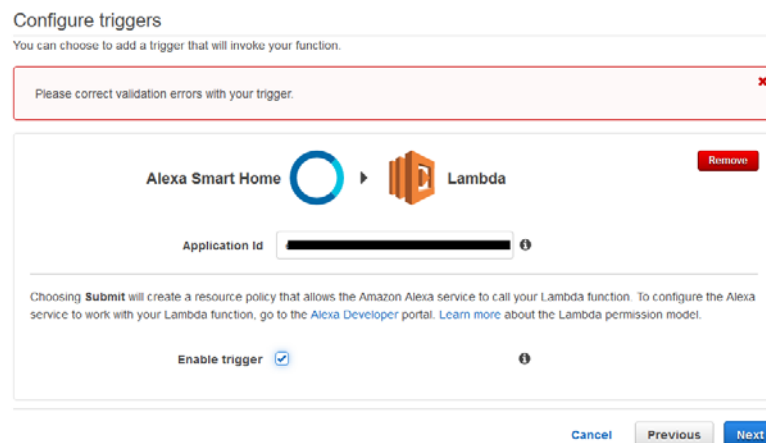
Hier den Blueprint: Blank Function auswählen

Auf der folgenden Seite in das gestrichelte Kästchen klicken:



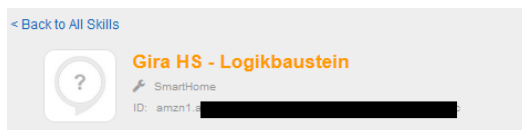
ACHTUNG WICHTIG: Wenn ihr Alexa Smart Home nicht auswählen könnt, dann müsst ihr durch ändern des Landes (keine Ahnung warum ich Nigeria hatte) es möglich machen. Mit Ireland als ausgewähltes Land funktioniert die Anmeldung!! Ireland ▾ Support ▾

Auch Super Wichtig: Die *Application Id* (siehe weiter oben) braucht man hier:



Also zwei Einträge/ bzw. Haken müssen hier gesetzt werden:

Eintrag 1: Die Applications ID
Diese haben wir ja auf der Webseite von <https://developer.amazon.com/> bekommen (siehe auch Seite 8):



Eintrag 2: Den Haken bei **Enable trigger** ☒ setzen

und Alexa Smart Home auswählen und mit Next bestätigen.

Jetzt kommen wir zum Punkt:

Lambda > New function

Select blueprint
Configure triggers
Configure function
Review

Configure function

A Lambda function consists of the custom code you want to execute. [Learn more](#) about Lambda functions.

Name*

Description

Runtime*

Lambda function code

Provide the code for your function. Use the editor if your code does not require custom libraries (other than the aws-sdk). If you need custom libraries, you can upload your code and libraries as a .ZIP file. [Learn more](#) about deploying Lambda functions.

Code entry type

```
1 exports.handler = (event, context, callback) => {  
2   // TODO implement  
3   callback(null, 'Hello from Lambda');  
4 };
```

Hier bitte die folgenden Einträge vornehmen:

Eintrag 1: Unter Name: **Name*** z.B. Gira eintragen

Eintrag 2: Bei **Runtime*** wählen

Eintrag 3: Bei **Role*** auswählen.

Info: Es wird ein neues Webbrowser- Fenster aufgemacht:

Services Resource Groups

AWS Lambda requires access to your resources

AWS Lambda uses an IAM role that grants your custom code permissions to access AWS resources it needs.

▼ Hide Details

Role Summary

Role Description	Lambda execution role permissions
IAM Role	<input type="text" value="Create a new IAM Role"/>
Role Name	<input type="text" value="lambda_basic_execution"/>

► View Policy Document

IAM Management Console

In diesem Fenster wird eine IAM Role erzeugt.

Wir lassen die Einstellungen so und bekommen somit eine :

zugewiesen

Mit **Don't Allow** **Allow** (ganz unten rechts) übernehmen wir

die Werte in die Seite **Configure function**

Durch diesen Schritt wird der folgende Eintrag gemacht:

Existing role*

Info: Bei der Beschreibung der Anmeldung unter Fhem steht noch die Information: **Choose an existing role wählen**, es muss wohl bei Amazon eine Änderung Bezüglich der Anmeldung stattgefunden haben, also ist hier im Dokument der zurzeit gültige Anmeldevorgang beschrieben (also nicht wundern warum das anders beschrieben wird!!).

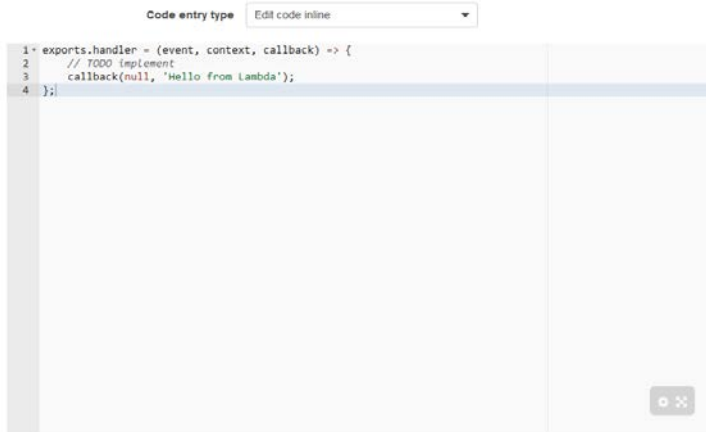
Eintrag 5: **Achtung:**

Hier kommt die größte Änderung zum Fhem Forum rein:

Role name*

Wir editieren jetzt den Quellcode der Datei: **GiraHS.js** (das ist ja unsere Beispiel, kann/ muss ja je nach Projekt angepasst, bzw. erstellt werden)

Am Anfang sieht das Editierfeld noch so aus:



```
1 exports.handler = (event, context, callback) => {
2   // TODO implement
3   callback(null, 'Hello from Lambda');
4 };
```

Für den Gira HS LBS ist die folgende Webseite Pflicht:

<https://github.com/Picpol/HS-AmazonEcho>

hier die GiraHS.js suchen:



also die Webseite: <https://github.com/Picpol/HS-AmazonEcho/blob/master/GiraHS.js>

Auszug aus der Datei:



```
1 var REMOTE_CLOUD_BASE_PATH = '/';
2 var REMOTE_CLOUD_HOSTNAME = '[DYNDNS]'; // Dynamic DNS Hostname ohne http/https z.B.: alexa.meinDomainName.de
3 var REMOTE_CLOUD_PORT = [PORTXXXXXX]; // zu verwendender Port (> 1024), gleicher Int-Wert wie im Logikbaustein Eingang 2
4 var ACCESS_TOKEN = '[TOKEN]' // zu verwendendes Access Token, gleicher String-Wert wie im Logikbaustein Eingang 3
5
6 var https = require('https');
7
8 /**
9  * Main entry point.
10  * Incoming events from Smart Home API are processed via this method.
11  */
12 exports.handler = function (event, context) {
13   log('Input', event);
14
15   switch (event.header.namespace) {
```

Info: Eine Gira HS Wiki Seite ist auch am entstehen.

Wenn jemand Lust hat kann er gerne mitarbeiten:

<https://github.com/Picpol/HS-AmazonEcho/wiki>

Zurück zum Quellcode aus GiraHS.js:

Es müssen noch zwei Einträge auf euren Zugang angepasst werden:

Eintrag 1: In Zeile 2: Euren Dyn- DNS Hostnamen:

```
2 var REMOTE_CLOUD_HOSTNAME = '[DYNDNS]'; // Dynamic DNS Hostname ohne http/https z.B.: alexa.meinDomainName.de
```


Eintrag 2: Info: Jetzt kommt der erste Querverweis zum Gira HS, bzw. zum Einrichten in den Gira Experten: **Eingang E2 Htps-Server Port** 

Info: Auf wie wird ein LBS für den Gira Experten angelegt möchte ich hier nicht eingehen.

Nur Kurzinfo: Von der Webseite den Quellcode in z.B. Node ++

[https://github.com/Picpol/HS-AmazonEcho/blob/master/10013_Amazon%20Echo%20\(10013\)_V0.1.hsl](https://github.com/Picpol/HS-AmazonEcho/blob/master/10013_Amazon%20Echo%20(10013)_V0.1.hsl)

eingeben und unten dem Dateinamen:

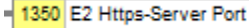
 **10013_Amazon Echo_V0.1.hsl** abspeichern.

Nun im Gira HS Experten eine Logikseite erstellen und den LBS

Einfügen (dieser ist unter WJL zu finden)

Nun entsprechend den Htps- Server- Port eingeben.

Info: Einen Portwert größer > 1024

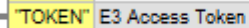


Jetzt auch der Wert in den Quellcode (Zeile 3) eintragen

```
3 var REMOTE_CLOUD_PORT = 1350 ; // zu verwendender Port (> 1024), gleicher Int-Wert wie im Logikbaustein Eingang 2
```


Eintrag 3: Hier muss auch derselbe Wert angegeben werden, den wir dann auch im Gira HS im Eingang 3 eingetragen haben.

Also eine „Art“ Token, eine Verschlüsselung, würde sagen wie eine W- LAN WPA2 Verschlüsselung. Hier im Beispiel habe wir die schlechte Verschlüsselungsnummer „TOKEN“ genommen



Der Wert im Quellcode (in Zeile 4) sieht dann wie folgt aus:


```
4 var ACCESS_TOKEN = 'TOKEN' // zu verwendendes Access Token, gleicher String-Wert wie im Logikbaustein Eingang 3
```



Wenn alles fertig ist mit  bestätigen


Es kommt folgendes Fenster:

Review

Please review your Lambda function details. You can go back to edit changes for each section. When you are ready, click **Create function** to complete the setup process.

Triggers 

 Alexa Smart Home
Application id:  Enabled

Lambda function 

Name: Gira

Description:

Runtime: Node.js 4.3

Environment variables:

Handler: index.handler





Existing role: service-role/lambda_basic_execution

Memory (MB): 128

Timeout: 3

VPC: No VPC

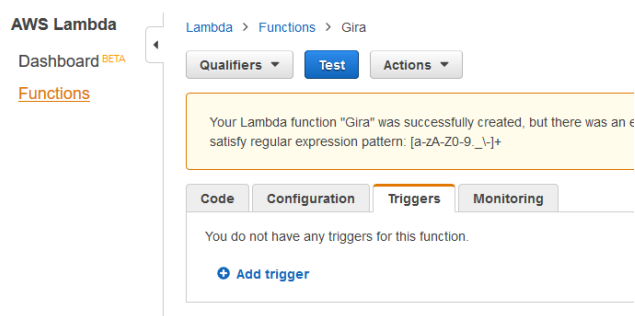
KMS key: (default) aws:lambda

Auf dieser Übersichtsseite über **Create function** die Funktion erstellen

...das dauert ein wenig

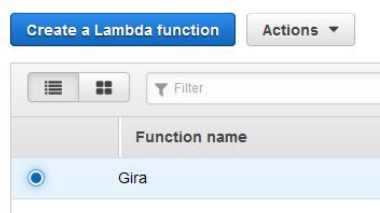
Bei dem neuen Fenster:



Oben links auf Functions **AWS Lambda** klicken

Dashboard **BETA**
Functions

In diesem Fenster die Funktion, z.B. Gira aktivieren:



Dann erst kann man über **Actions** den Punkt Show ARN auswählen



In diesem Fenster wird nun eine ARN- Adresse angezeigt, die man weiter oben benötigt:

So jetzt haben wir die Echo Einrichtung geschafft!!!

Legende/ Quellennachweis:

¹ **Amazon AWS-Konto:** Auszug aus WIKIPEDIA: https://de.wikipedia.org/wiki/Amazon_Web_Services
Amazon Web Services (AWS) ist eine Sammlung verschiedener [Online-Dienste](#) des Unternehmens [Amazon.com](https://amazon.com), die im Jahr 2006 gestartet sind.

² **AWS Lambda-Funktion** Auszug aus Amazon web services:
Der Code, den Sie auf AWS Lambda ausführen, wird als "Lambda-Funktion" hochgeladen. Jeder Funktion sind Konfigurationsinformationen zugeordnet, etwa der Name der Funktion, ihre Beschreibung, ihr Einstiegspunkt und ihre Ressourcenanforderungen. Der Code muss in einem "zustandslosen" Stil geschrieben sein, also davon ausgehen, dass es keine Affinität zur zugrunde liegenden Infrastruktur gibt. Der lokale Zugriff auf das Dateisystem, untergeordnete Prozesse und ähnliche Elemente bestehen möglicherweise nicht über die Laufzeit der Anforderung hinaus. Jeder persistente Zustand sollte in Amazon S3, Amazon DynamoDB oder einem anderen im Internet verfügbaren Speicherservice gespeichert werden. Lambda-Funktionen können Bibliotheken enthalten, auch native.
Webseite dazu die Hilfe gibt: <https://aws.amazon.com/de/lambda/faqs/>