Reference RDK Alexa Echo Dot Voice Application (In Progress)

Modified	28 Jan 2020 by **SHIBUTHOMAS**
Version	Future Release
Contact	
Jira	

Summary	Created	Updated	Due	Assignee	P	Status	Resolution

1. Objective

To demonstrate a method to integrate Alexa Echo Dot with RDK 4.0 by developing Alexa skills and integrate on RDK 4.0 reference platforms:-

- Raspberry Pi
- Arris Accelerator - Commscope

Following sections will give the architecture details and end to end flow diagram

2. Amazon Alexa Voice Control Reference High Level Architecture

2.1.1. Architecture (End Product Reference Architecture for Alexa voice Integration)

Account linking flow
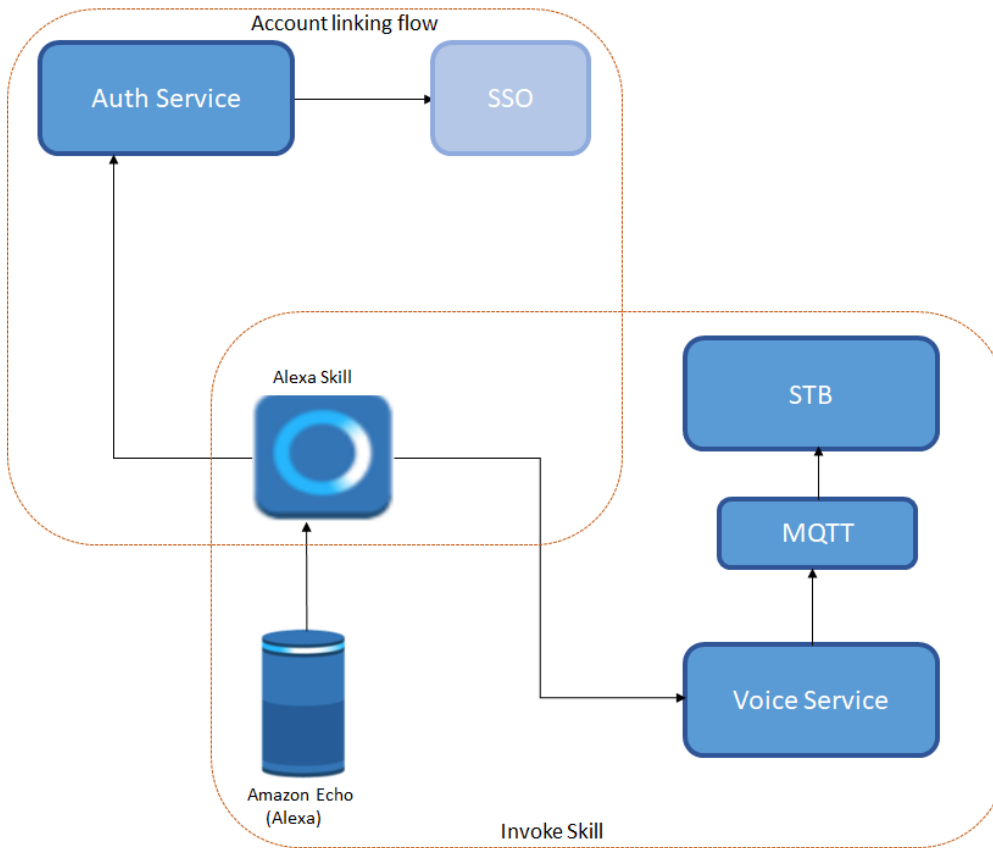
Note: Above is the stable state end goal of the product architecture. Rest of the section is for demonstration purposes as an example and not realizing the full scope of the above architecture as end goal.

2.1.2. Architecture used for proof of integration concepts.
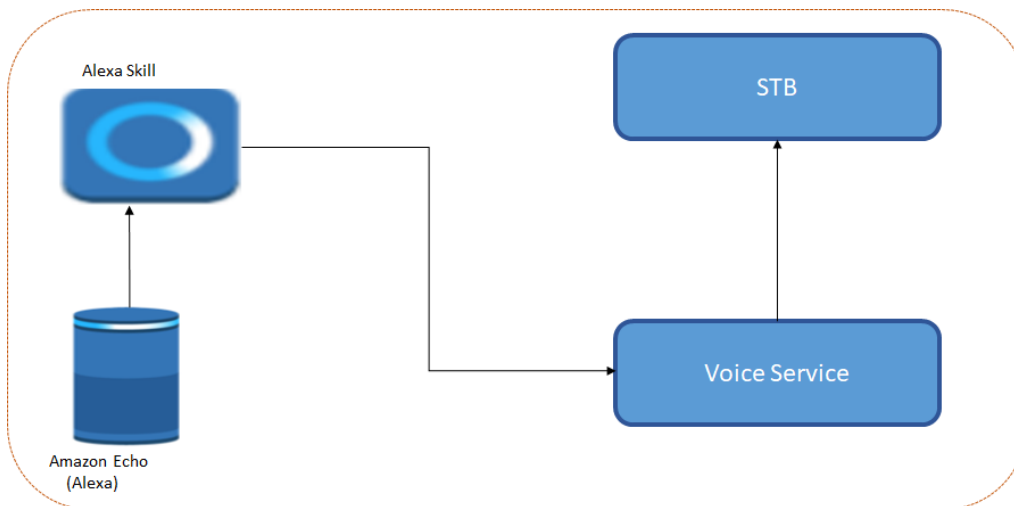

This above mentioned application helps to demonstrate E2E working of Alexa voice and can easily extend to a more scalable, performant and SPOF (Single point of failure) handling design.

3. RDK Alexa Voice Integration - Amazon Voice Server (AVS) setup and configurations

3.1. Alexa Skill Creation

3.1.1. Step one: got to https://developer.amazon.com/alexa/console/ and select your skill name
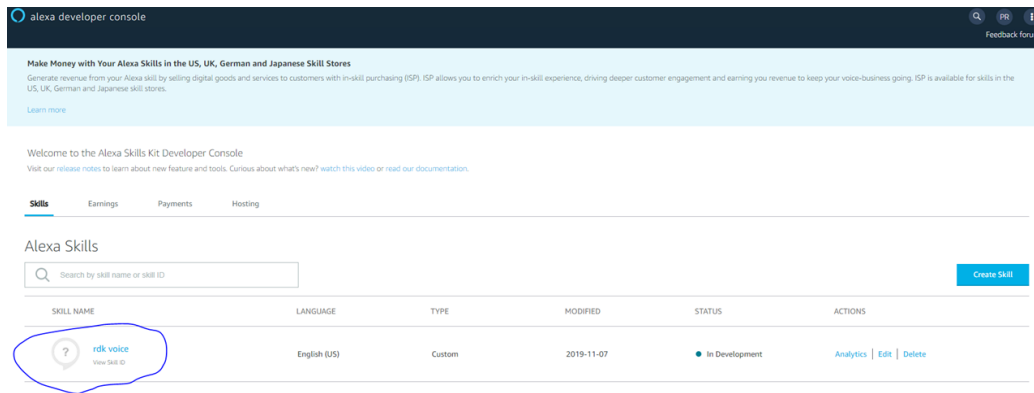
3.1.2. Step two: got to test tab
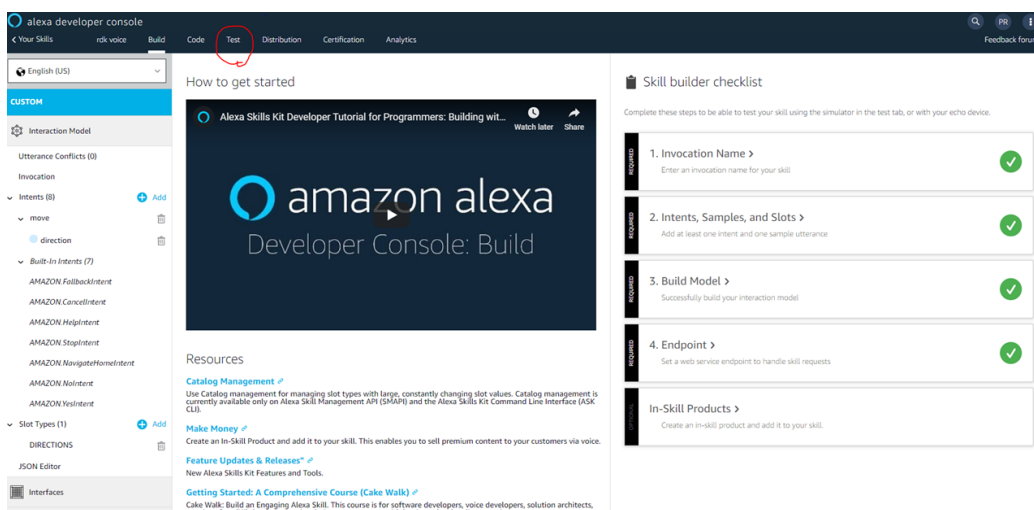
3.1.3. Step Three: Type or say your test command, eg: Alexa, tell stb voice move right

3.1.4. Step four: You will get a response, please note the user id – this maps our account to Django server later

4. RDK Alexa Voice Integration - Alexa Server configuration details

4.1. Configure Alexa User Account and STB account

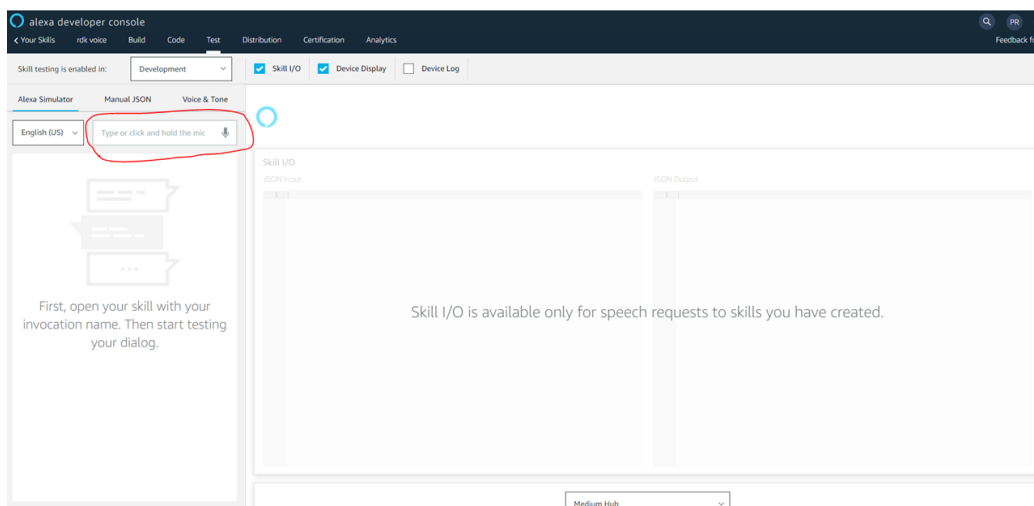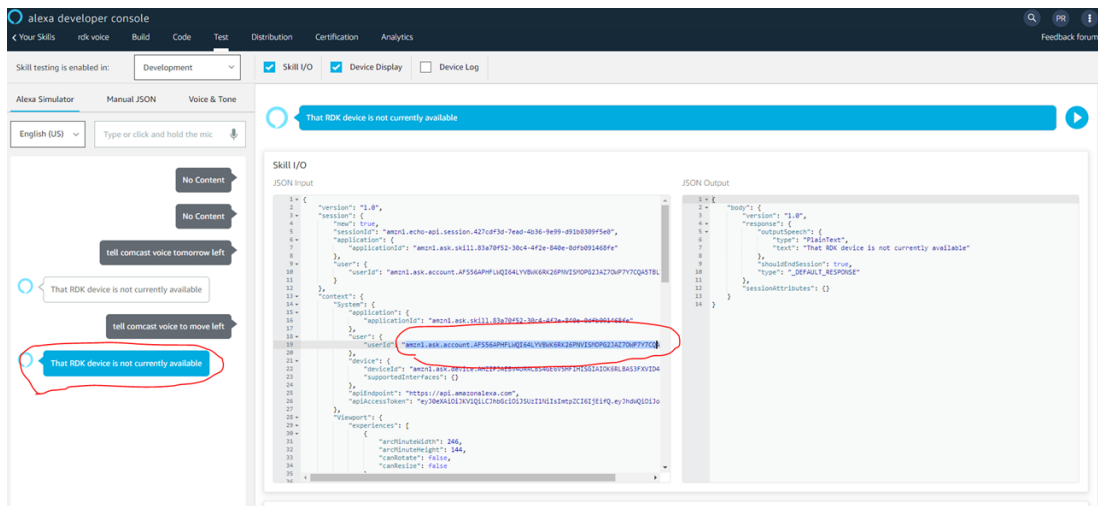It is customer responsibility to create Alexa Use account and STB account once Alexa voice feature has enabled. Following steps explains how can we create a user account. STB account has to mapped with amazon user id and this mapping will ensure that respective alexa voice will reach to the right STB

4.1.1. Step one: Go to https://rdkvoice.iptn.io/admin
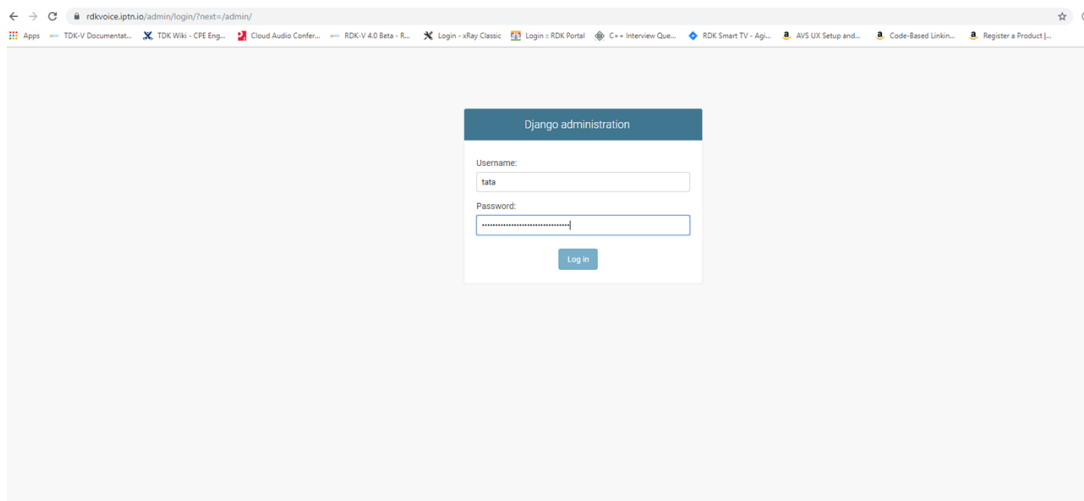
4.1.2. Step two: Go to first users link

4.1.3. Step Three: create a user account

4.1.4. Step Four: user creation

4.1.5. Step Five: STB account creation

4.1.6. Step Five: STB account creation
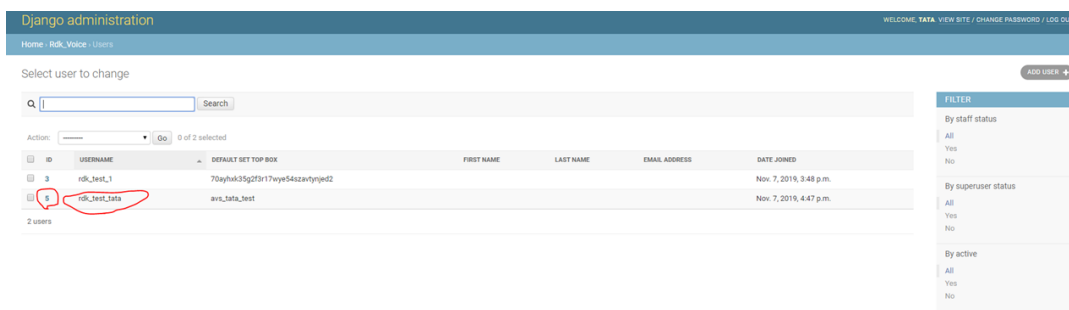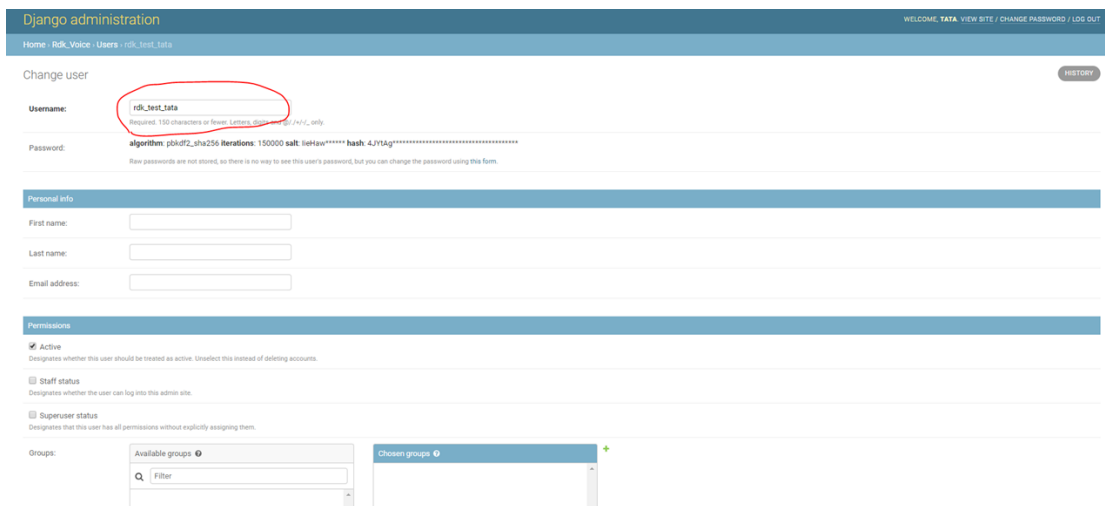

5. E2E Sequence Diagram

6. Client side architecture
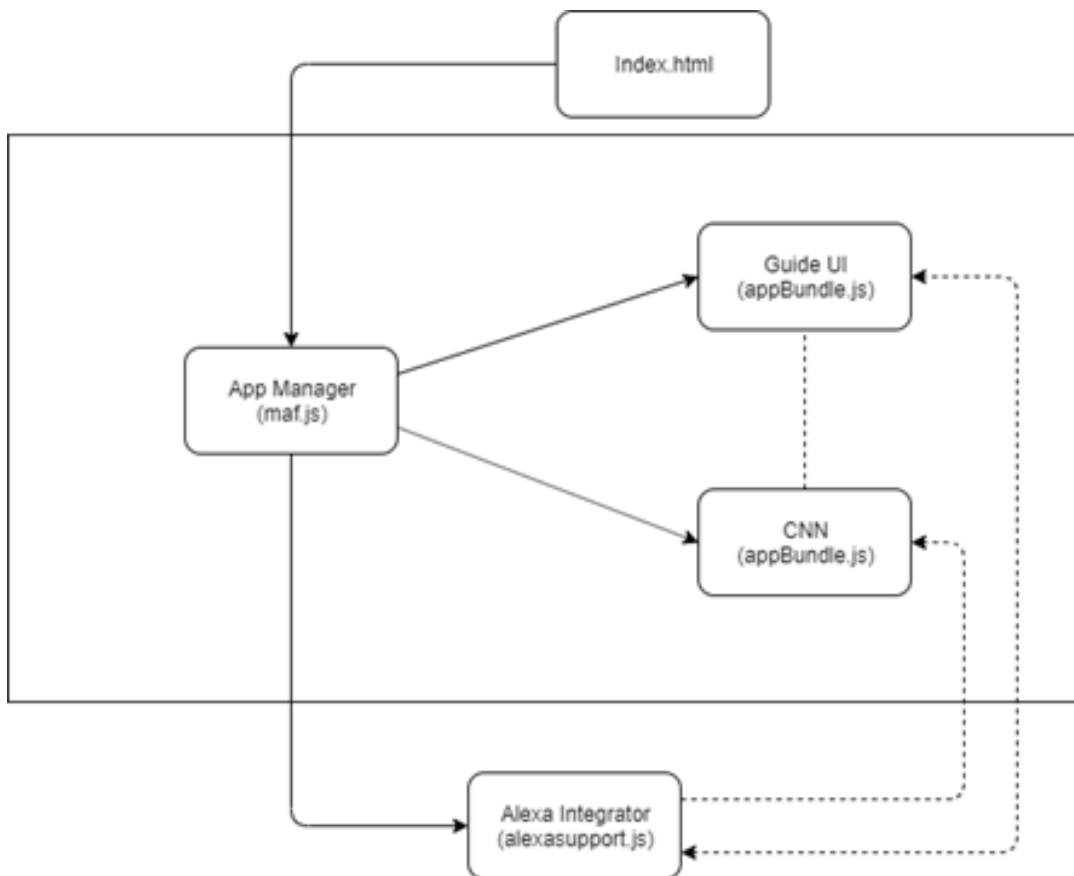
Below section give the details of the each components involved:-

- html – This is the main HTML file which have the instructions to load the app-manager js file (maf.js)
- App Manager – This block contains the logic to handle guide and app launch. The App manager will start the web socket by using the Alexa Integrator once the guide has finished launching.
- Guide UI – This file exports a lightning app object which has all the required lightning components to show the guide
- CNN – This file exports a lightning app object which has all the required lightning components to show the CNN app
- Alexa Integrator – This file contains a function which will export the class, whose object can be used to integrate Alexa.

7. API definition

7.1. method: "RDK.RegisterNewDevice"

7.1.1. Input:

- "id": "1234567890"

7.1.2. Output:

- "id": "1234567890"

7.2. method: "RDK.CompleteRegistration"

7.2.1. Input:

- "id": "1234567890"

7.2.2. Output:

- "id": "1234567890"

7.3. method: "RDK.finalizeRegistration"

7.3.1. Input:

- "id": "1234567890"
- availableApplications:[{}]

7.3.2. Output:

- "id": "1234567890"
- result: {available applications in the box}

7.4. method: "RDK.LaunchApplication"

7.4.1. Input:

- "id": "1234567890"
- appURI: string
- contentID: string
- searchString: string

7.4.2. Output:

- result : Boolean

7.4.3. Errors:

- error code
  - 1 Application not available
  - 2 content not available

7.5. method: "RDK.GetCurrentApplication"

7.5.1. Input:

- None

7.5.2. Output:

- The appURI of the current application

7.6. method: "RDK. Navigate"

7.6.1. Input:

- direction: String

7.6.2. Output:

- Result: success

7.7. method: "RDK. MediaControl"

7.7.1. Input:

- action: String

7.7.2. Output:

- Result: success

8. How to configure server

Follow the process outlined in the attached document to configure the server.

Deployment guide.docx

9. FAQ

8.1 Is this a ready to deploy service integration ?

No, this is not. It is only meant for demonstration of capability purposes.

8.1 What is pending from this proof of concept to actually integrate echo dot as part of RDK 4.0 Voice ecosystem ?

Scalability and security of product deployment is missing in this implementation.

8.3 Do I have to sign any special license or pay a fee to use this service if I have Alexa Echo Dot configured in my home Wifi ?

No.

8.4 Who shall we contact in case we have to integrate more skills to this mode ?

TBD

8.5 Which version of Alexa devices are supported ? What about STB models ? RDK Software versions ?

TBD

8.6 What happens if my echo device does not recognize the commands in the prescribed formats of the API ? Which are the areas I should look for triaging ?

TBD

8.7 How scalable is the current solution ?

End state architecture has scalability built into it, current PoC is only for demonstration purposes.

8.8 How secure is the current solution ?

End state architecture has security built into it, current PoC is only for demonstration purposes.

8.9 Is there a plan in RDK community to realize the end state architecture ?

Currently No. We will revisit the plan based on the users feedback and interest in using this service.