

An Introduction to Designing Voice Driven Experiences



DAVE ISBITSKI

CHIEF EVANGELIST, ALEXA AND ECHO

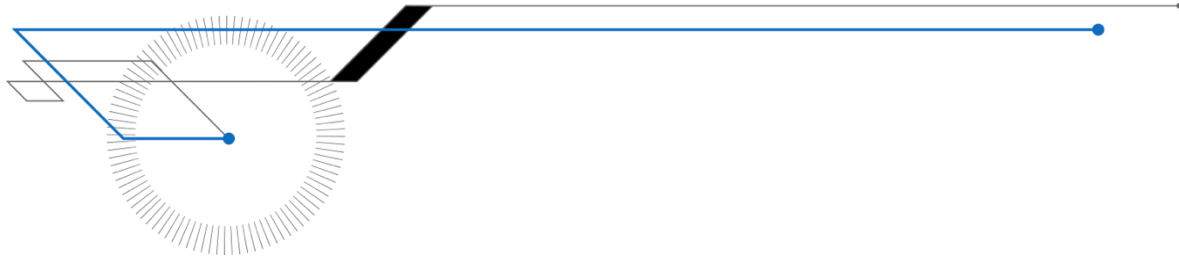


@TheDaveDev



isbitski@amazon.com

What is Alexa?





Alexa, Hello.

The **Alexa** Platform

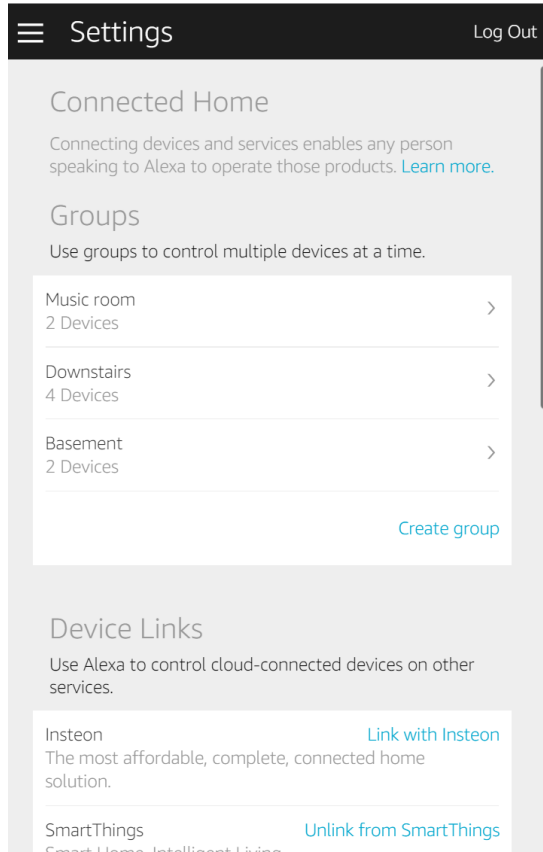


Connected Home (CoHo) and Lighting API



INSTEON®

Alexa App



<http://alexa.amazon.com>

ALEXA SKILLS KIT (ASK)

<https://developer.amazon.com/ask>



Alexa Skills Kit (ASK) Developer Preview

A free SDK that lets you easily add new voice capabilities

Alexa, the voice service that powers Echo, provides a set of built-in abilities, or skills, that enable customers to interact with devices in a more intuitive way using voice. Examples of these skills include the ability to play music, answer general questions, set an alarm or timer and more. With the Alexa Skills Kit, you can easily build and add your own skills to Alexa. Customers can access these new skills simply by asking Alexa a question or making a command.

Quickly Build Skills with the Alexa Skills Kit (ASK)

The Alexa Skills Kit is a collection of self-service APIs, tools, documentation and code samples that make it fast and easy for you to add skills to Alexa. Using ASK, you can teach new skills to Alexa in just a few hours. No prior experience with speech recognition or natural language understanding is required. All of the code runs in the cloud — nothing is installed on any user



Get Started

Alexa Voice
Service

Alexa Fund

GETTING STARTED

- [Getting Started with the Alexa Skills Kit](#)

ALEXA VOICE SERVICE (AVS)

<https://developer.amazon.com/avs>



Alexa Voice Service (AVS) Developer Preview

Bring voice capabilities to your connected device

Introducing the Alexa Voice Service (AVS)

If you're a hardware maker and your connected device has a microphone and a speaker, the new Alexa Voice Service (AVS) [developer preview](#) enables you to add voice-powered experiences to your connected devices. Your customers can simply speak to Alexa through the microphone on your device and Alexa will respond through your device's speakers. This gives your customers access to Alexa's skills and capabilities, including built-in skills and those created by you or other developers using the [Alexa Skills Kit \(ASK\)](#). Examples of Alexa skills include the ability to answer general knowledge questions, provide weather forecasts, query Wikipedia and much more.



Get Started

[Alexa Skills Kit](#)

[Alexa Fund](#)

GETTING STARTED

■ [Getting Started with the Alexa](#)



<http://developer.amazon.com/ask>

<http://developer.amazon.com/blog>

amazon echo

Always ready, connected, and fast. **Just ask.**



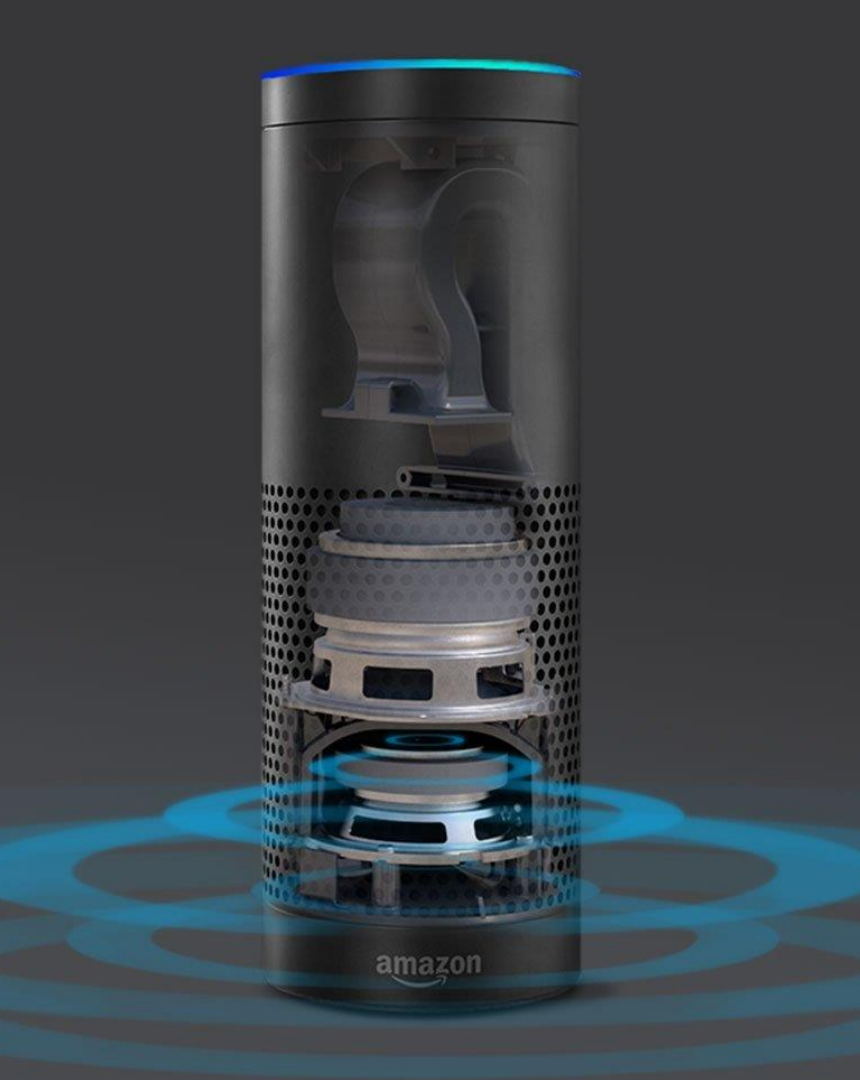
Customer Expectations for **ALEXA SKILLS**

Users can speak to Alexa naturally and spontaneously.

She understands most requests.

She responds in an appropriate way, either by executing the command, or informing the user why she can't.

As you look to create your own skills you should ensure all three of these core user experiences are met.



Key Design Principles for **ALEXA SKILLS**

- ✓ Skills Should Provide High Value
- ✓ A Skill Should Evolve Over Time
- ✓ Users Can Speak to Your Skill Naturally and Spontaneously
- ✓ Alexa Should Understand Most Requests to Your Skill
- ✓ A Skill Should Respond in an Appropriate Way

High Utility

Low Utility

Doing

Performs a Task

“Alexa, ask Scout to arm away mode.”

“Away mode armed. You have 45 seconds to leave the house.”

Searching

Identifies specific info

“Alexa, ask Vendor if there are Madonna tickets available for this weekend.”

“There are a limited amount of tickets, ranging from \$49 to \$279.”

Telling

Provides a quick reference point

“Alexa, tell me a cat fact.”

“It is well known that dogs are superior to cats.”

Browsing

Gives info on a broad subject

“Alexa, ask Amazon what’s on sale.”

“The following items are on sale right now...”

Skills Should Provide High Value

Voice is conversational. Very different than touch driven experiences. Less is more.

A large majority of the types of skills submitted today can grow with the user over time. Aim for skills that perform tasks on behalf of the user and learn as time goes on.

This will provide a much better experience and lead to more complex interactions.

A Skill Should Evolve Over Time

Voice user interfaces work well when they are focused, and give quick responses.

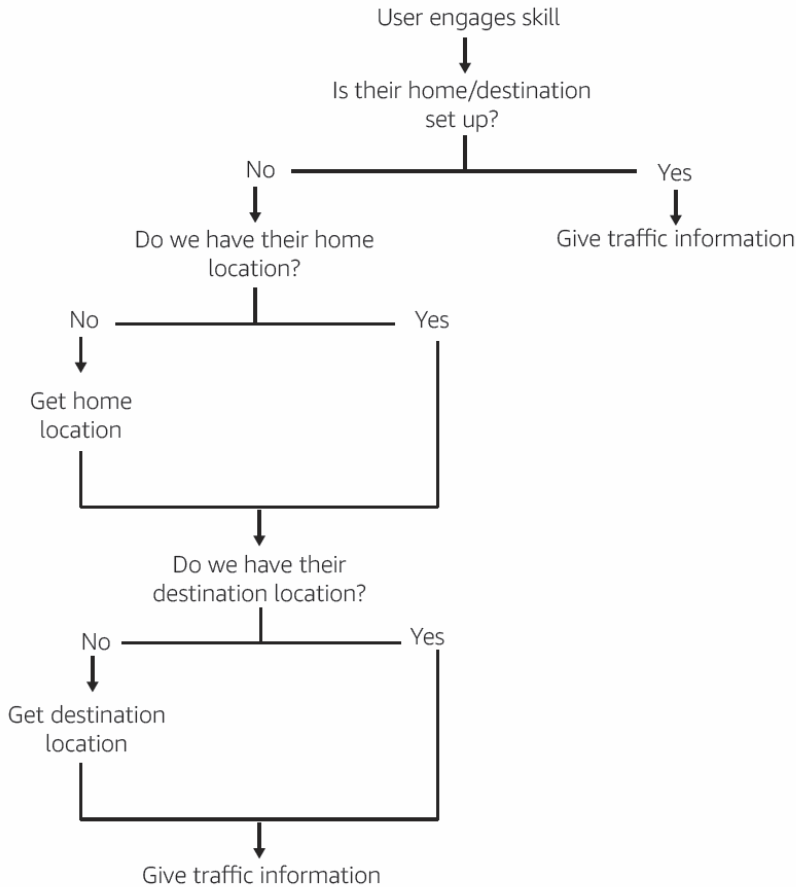
Start with a primary use case that both communicates your business case, but is also a clear winner for a voice user interface.

Let's do one thing well, and add in capabilities allowing it to get smarter over time.

This follows the current model we have with Alexa. She is getting smarter over time.

Example of Automatic Learning

ALEXA SKILL



Alexa, launch Travel Buddy

Hi, I'm travel buddy. I can easily tell you about your daily commute. Let's get you set up. Where are you starting from?

Philadelphia

Ok, and where are you going?

Boston

Great, now whenever you ask, I can tell you about the commute from Philadelphia to Boston. The current drive time is five hours and twenty three minutes. There is an accident on I95 near Hartford.

Alexa, launch Travel Buddy

Your commute is currently five hours and two minutes.

Account Linking or Creation *

Do you allow users to create an account or link to an existing account with you? [Learn more](#)

Yes No

Authorization URL *

The url where customers will be redirected in the companion app to enter login credentials.

Client Id *

Unique public string used to identify the client requesting for authentication.

Domain List

The list of domains that the authorization URL will fetch content from. You can provide up to 15 domains.

Scope

List of permissions to request from the skill user. You can provide up to 15 scopes.

Redirect URL

HTTP's redirection endpoint url you want to direct to after completing the authorization interaction with user.

Customer friendly with **ACCOUNT LINKING**

- Allow your customers to link their existing accounts with you, to Alexa.
- Customers are prompted to log in to your site using their normal credentials with webview url you provide.
- You authenticate the customer and generate an access token that uniquely identifies the customer and link the accounts.

Users Can Speak to Your Skill Naturally and Spontaneously

The experience of using your Alexa skill should allow users to not have to think about what to say and allow them to not remember how to say it.

They should be able to converse with Alexa just as they would another human.

All they need is a rough idea of what Alexa can do (e.g. playing music, setting a timer, etc.), and they just ask her to do it.

This is the real value of voice interface, but this value can quickly erode in a skill that forces users to interact in unnatural ways.

Users Can Speak to Your Skill Naturally and Spontaneously

You should try to remove artificial skill syntax and make interactions within your skill as natural as possible.

Allowing your users to make simple requests without having to think about the format those requests should be in, will create a much better experience.



Example of a Conversation in **ALEXA SKILLS**

Odd Phrasing: Very odd and/or lengthy invocations that inhibit using the skill in a conversational and spontaneous way.

Alexa, ask [davefacts] for a fact when the fact is of type davefact.

Alexa, ask [dave] for a [fact].

Lengthy Invocations: The combination of skill name with the task is often difficult to remember the exact syntax .

Alexa, ask [transportation service alerts] for the [current status] of [the monorail A].

Alexa, ask [trafficbuddy] about [monorail A]



Example of a Conversation in **ALEXA SKILLS**

Repetitive Invocations: Some invocations (particularly those that would not necessarily need an intent) are not optimized for the “ask” model and may result in repetitive phrasing.

Alexa, ask [developerinfo] for a [developer info].
Alexa, ask [developerinfo].



Having a Good Conversation in an **ALEXA SKILL**

- Makes It Clear that the User Needs to Respond
- Doesn't Assume Users Know What to Do
- Clearly Presents the Options
- Keeps It Brief
- Avoids Overwhelming Users with Too Many Choices
- Offers Help for Complex Skills
- Asks Users Only Necessary Questions
- Uses Confirmations Selectively
- Obtains One Piece of Information at a Time
- Makes Sure Users Know They are in the Right Place
- Avoids Technical and Legal Jargon

*Write for the **Ear**, not the **Eye**!*

Alexa Should Understand Most Requests to Your Skill

In the core Alexa experience, most requests are understood and acted on. The same experience should be provided within your own skill without numerous attempts to invoke your skill failing for the end user.

Currently, the biggest contributor for requests to your skills not being consistently understood is a lack of sample utterances in your interaction model.

When skills do not work as consistently and reliably as the core Alexa experience, users will become frustrated.

```
{
  "intents": [
    {
      "intent": "GetComputer",
      "slots": [
        {
          "name": "Model",
          "type": "MODELS"
        },
        {
          "name": "Date",
          "type": "AMAZON.DATE"
        }
      ]
    }
  ]
}
```

Building an Alexa Skill **HOSTED SERVICE**

- You **define** interactions for your Alexa Skill through **Intent Schemas**
- Each intent consists of two fields. The **intent field** gives the name of the intent. The slots field lists the slots associated with that intent.
- Slots can be any internal types such as `AMAZON.LITERAL`, `AMAZON.NUMBER`, `AMAZON.DATE`, `AMAZON.US_CITY` etc. or they can be ones you create.

- “what is...”
- “what’s...”
- “tell me...”
- “give...”
- “give me...”
- “get...”
- “get me...”
- “find...”
- “find me...”

Building an Alexa Skill

HOSTED SERVICE

- The mappings between intents and the typical utterances that invoke those intents are provided in a tab-separated text document of sample utterances.
- Each possible phrase is assigned to one of the defined intents.
- GetHoroscope what is the horoscope for {pisces|Sign}
- GetHoroscope what will the horoscope for {leo|Sign} be {next tuesday|Date}

Custom Slot Types

Custom slot types to be referenced by the Intent Schema and Sample Utterances

For more information, see [Defining the Voice Interface for an Alexa skill](#).

Example: TOPPINGS - cheese | onions | ham

Adding slot type

Enter Type *

MODELS

Enter Values *

Values must be line-separated

```
1 A500
2 A1000
3 A2000
4 A2000HD
5 A2500
6 A3000
7 A600
8 A1200
9 A4000|
```

Increasing Accuracy with **CUSTOM SLOTS**

- Created inside Interaction Model page once in the Developer Portal
- Greatly reduces the number of sample utterances required
- Can define as many as you need with values line separated
- Can be combined with existing AMAZON internal types

```
{
  "intents": [
    {
      "intent": "GetFirstEventIntent",
      "slots": [
        {
          "name": "day",
          "type": "AMAZON.DATE"
        }
      ]
    },
    {
      "intent": "GetNextEventIntent"
    },
    {
      "intent": "AMAZON.HelpIntent"
    },
    {
      "intent": "AMAZON.StopIntent"
    }
  ]
}
```

Increasing Accuracy with **Built-In Intents**

AMAZON.CancelIntent

- Called when the user says “cancel”, “nevermind”, “forget it” or something similar.
- This Intent will let the user cancel the current task but remain in the skill, or exit the skill completely.

```
{
  "intents": [
    {
      "intent": "GetFirstEventIntent",
      "slots": [
        {
          "name": "day",
          "type": "AMAZON.DATE"
        }
      ]
    },
    {
      "intent": "GetNextEventIntent"
    },
    {
      "intent": "AMAZON.HelpIntent"
    },
    {
      "intent": "AMAZON.StopIntent"
    }
  ]
}
```

Increasing Accuracy with **Built-In Intents**

AMAZON.HelpIntent

- Called when the user says “help”, “help me”, “can you” or “help me.”
- This skill provides a way for you to return help on how to use your skill and can be customized.


```
{
  "intents": [
    {
      "intent": "GetFirstEventIntent",
      "slots": [
        {
          "name": "day",
          "type": "AMAZON.DATE"
        }
      ]
    },
    {
      "intent": "GetNextEventIntent"
    },
    {
      "intent": "AMAZON.HelpIntent"
    },
    {
      "intent": "AMAZON.StopIntent"
    }
  ]
}
```

Increasing Accuracy with **Built-In Intents**

AMAZON.StopIntent

- Called when the user says “stop”, “off”, “shut up” or something similar.
- This Intent will let the user stop an action but remain in the skill or exit the skill completely.

A Skill Should Respond in an Appropriate Way


- An Alexa skill should provide adequate error handling for unexpected or unsupported utterances.
- A user should never be exposed directly to a skill's error handling. Instead Alexa should respond with a request for more information from the user or simply that she is unable to do the current task.
- When an error does occur it should be clear to the user what went wrong and where it occurred.
- Since Alexa will not be doing any client side checking of slot values being sent with your Intents you should check for missing values and value types server side within your service.
- If you find any missing information you should respond to the Alexa service with a reprompt inside the OutputSpeech object.

Changing Alexa's inflection with **SSML**

- Alexa automatically handles normal punctuation, such as pausing after a period, or speaking a sentence ending in a question mark as a question.
- Speech Synthesis Markup Language (SSML) is a markup language that provides a standard way to mark up text for the generation of synthetic speech.
- Tags supported include: speak, p, s, break, say-as, phoneme, w and audio.

```
1 < speak >
2   Here is a number read as a cardinal number:
3   < say-as interpret-as="cardinal">12345</ say-as >.
4   Here is the same number with each digit spoken separately:
5   < say-as interpret-as="digit">12345</ say-as >.
6   Here is a word spelled out: < say-as type="spell-out">hello</ say-as >
7 </ speak >
```

This Alexa skill is ready for testing

Enable This skill is enabled for testing on this account. 

Once you have completed testing on your device, please complete the Publishing Information tab, then submit the skill for certification.

If it passes Amazon's testing and certification process, it will become available to Alexa end users.

Try this on your Echo: Alexa ask developer day

Service Simulator

Use Service Simulator to test your lambda function.

Text **Json**

Enter Utterance *

when is dave speaking

Ask Amazon Developer Day

Reset

Lambda Request

```
{
  "session": {
    "sessionId": "SessionId.b2b555fd-2aa9-44b7-abbe-826d63ddcb01",
    "application": {
      "applicationId": "amzn1.echo-sdk-ams.app.ec9314c5-7e9a-45c2-8318-4b3fe492b8d7"
    },
    "attributes": null,
    "user": {
      "userId": "amzn1.account.AGSRDOGKYM4N2ZYD7GKWH3XAQ5A"
    },
    "new": true
  },
  "request": {
    "type": "IntentRequest",
    "requestId": "EdwRequestId.70ef270c-c860-4f5b-a72a-f22a05a084a2",
    "timestamp": 1441396052972,
    "intent": {
```

Lambda Response

```
{
  "version": "1.0",
  "response": {
    "outputSpeech": {
      "type": "plain text",
      "text": "dave will be presenting An Overview of the Amazon Devices and Services for Mobile Developers at 10am. He will also be presenting An Introduction to Using Amazon Web Services and the Alexa Skills Kit to Build Voice Driven Experiences at 10:30am"
    },
    "reprompt": {
      "outputSpeech": {
        "type": "plain text",
        "text": null
      }
    },
    "shouldEndSession": true
  },
  "sessionAttributes": {}
}
```

Testing Your Skill

SERVICE SIMULATOR

- Enabled once a Skill has been configured in the Developer Portal
- Use spoken utterances to generate ad hoc results
- Use JSON to verify requests
- Combine with AWS Lambda Unit Tests to verify both client and service side Alexa end points

Digging Deeper into Voice Design

Alexa Skills Kit Voice Design Best Practices - <http://bit.ly/voicedesign>

Alexa Skills Kit Voice Design Handbook - <http://bit.ly/voicehandbook>

Wired for Speech: How Voice Activates and Advances the Human-Computer Relationship, by Nass and Brave

The Elements of VUI Style: A Practical Guide to Voice User Interface Design, by Bouzid and Ma

Don't Make Me Tap!: A Common Sense Approach to Voice Usability, by Bouzid and Ma

The Voice in the Machine: Building Computers That Understand Speech, by Pieraccini

Voice User Interface Design, by Cohen, Giangola, and Balogh

An Introduction to Designing Voice Driven Experiences



DAVE ISBITSKI

CHIEF EVANGELIST, ALEXA AND ECHO



@TheDaveDev



isbitski@amazon.com