AWS Chatbot Challenge

Amazon Lex and AWS Lambda

Benjamin Towner

Cloud Solutions Architect / Engineer

[Overview 2](#_Toc487213703)

[AWS Chatbot Challenge 2](#_Toc487213704)

[Eligibility 3](#_Toc487213705)

[Requirements 3](#_Toc487213706)

[Main Requirement: 3](#_Toc487213707)

[Submission Recommendations: 3](#_Toc487213708)

[How to enter 3](#_Toc487213709)

[Judging Criteria 4](#_Toc487213710)

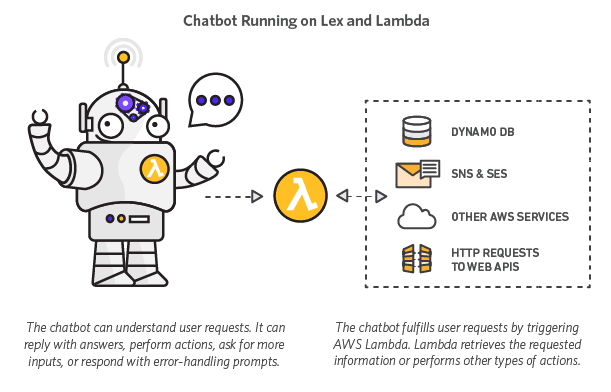
# Overview

## AWS Chatbot Challenge

Chatbots are changing how companies interface with their customers. With chatbots, you can easily fulfill the needs of your customers in an automated way using natural, human-like chat interfaces. Chatbots serve a variety of use cases, such as customer support, transaction fulfillment, data retrieval, or even DevOps functions (ChatOps).

However, building and running chatbots is a difficult task. First, most developers lack the deep learning expertise necessary to create bots that can intelligently interpret and respond to text. In addition, developers must also provision, manage, and scale the compute resources necessary to run the bot’s code.

What if you could build chatbots with sophisticated natural language processing and almost no operational overhead? [Amazon Lex](https://aws.amazon.com/lex/) is a fully managed service for building conversational interfaces into any application using voice and text. Lex is powered by the same deep learning technologies that power Amazon Alexa and lets you build natural language chatbots. Lex is integrated with [AWS Lambda](https://aws.amazon.com/lambda/), a service that lets you run code without provisioning or managing servers. Lambda enables you to write and run logic for your chatbot using serverless compute. Getting started with Amazon Lex and AWS Lambda is quick and easy.



# AWS ChatBot Design Concept:

How many hours in a day does a parent have to filter the digital content and ‘**Trust but Verify’** their children are communicating within acceptable guidelines. Would your ‘mamma’ let a stranger call the house and talk to you when you were a kid?

***MotherBot*** – Enable Parents or Parental guardians to manage the monitoring of the digital presence of those children under age 18, while providing an appropriate level of privacy and respect. The AWS Chat Bot allows various communication channels to be added to a child’s laptop or mobile devices with the primary purpose of monitoring of content, verify the identity of the individual connections and provide trending statistics to anticipate edge-cases of how these communications tools are used.

Workflows replicate several key components that a Parent might perform on a social media contact that their child may have, such as verify the name of the individual or address where they live. In the past, talking with other parents in the community provides a ‘*Herd mentality*’ approach to validating contacts as a protection function. ‘Googling’ the individuals of interest usually identify red-flags immediately if there are any issue to be concerned with about those individuals that are known to the parent.

***MotherBot*** creates this self-service administration platform for private networks to grant access, while leveraging the benefits of a high availability and accessibility infrastructure via AWS. It can provide a self-service verification process to those new contacts by describing the steps needed for contacts to be added and exactly what type of monitoring will be performed after it is completed. It can also be integrated into Public Search to verify the matching results. Once permissions are granted, it can continue to monitor communication transactions as a proxy and deliver notifications if context thresholds are exceeded for things like appropriateness or dialect.

Kids will be kids, but now they can be safe and enriched by providing the parents more time to be parents instead of data detectives.

# Eligibility

This competition is open to the following:

* Individuals, and teams of individuals, who have reached the age of majority in their eligible jurisdiction of residence at the time of entry
* Organizations (employ 50 or fewer people) - must have been duly organized or incorporated and validly exist in an eligible area at the time of entry
* Large Organizations (employ more than 50 people) in eligible areas will be eligible for a non-cash recognition prize

The competition welcomes submitters from most countries around the globe. However, individuals or organizations may be disqualified if they are based in a nation, state, province, or territory where U.S. or local law prohibits participating in the competition or receiving a prize. This includes individuals, who are residents of, and organizations domiciled in Brazil, Quebec, Cuba, Sudan, Iran, North Korea, Syria and any other country designated by the United States Treasury's Office of Foreign Assets Control.

# Requirements

## Main Requirement:

Build a conversational, natural language chatbot using Amazon Lex. Use Lex’s integration with AWS Lambda to execute logic on the backend, such as for fulfilling user intent or performing user data validation.

Your submission must be a new or existing bot (If existing, submitters must have updated their bot to run on Amazon Lex and AWS Lambda during the submission period).

## Submission Recommendations:

Solutions can, but are not required to

* Be deployed to Slack, Facebook Messenger, or Twilio
* Use other AWS services
* Record or retrieve data from sources like Salesforce, HubSpot, Marketo, Microsoft Dynamics, Zendesk, and QuickBooks using Lex’s pre-built enterprise connectors.
* Incorporate speech capabilities using a service like Amazon Polly
* Leverage third-party APIs, SDKs, and services

# How to enter

1. Register for the AWS Chatbot Challenge.
2. Create an account on [AWS](https://portal.aws.amazon.com/gp/aws/developer/registration/index.html).
3. Visit the [Resources page](https://awschatbot2017.devpost.com/details/resources) for links to documentation and resources.
4. Shoot your demo video that demonstrates your bot in action. Prepare a written summary of your bot and what it does.
5. Provide a way to access your bot for judging and testing, including a link to your repo hosting the bot code and all deployment files and testing instructions needed for testing your bot. (The Github or BitBucket code repository may be public or private. If the repository is private, share access with [testing@devpost.com](mailto:testing@devpost.com)).
6. Submit your bot on AWSChatbot2017.Devpost.com before July 18, 2017 at 5pm PDT and share access to your bot, its repo and its deployment files.

# Judging Criteria

* **Customer Value**   
  The extent to which the bot provides value to your users. Does your bot help solve a problem or painpoint for your users?
* **Bot Quality**  
  Creativity and originality of the bot. Is your bot differentiated and does it solve your users' problems in a unique way?
* **Implementation of the Bot**  
  How well the bot was built and executed by the developer. Does the bot function as intended and does it recognize and respond to the most common phrases asked of it?