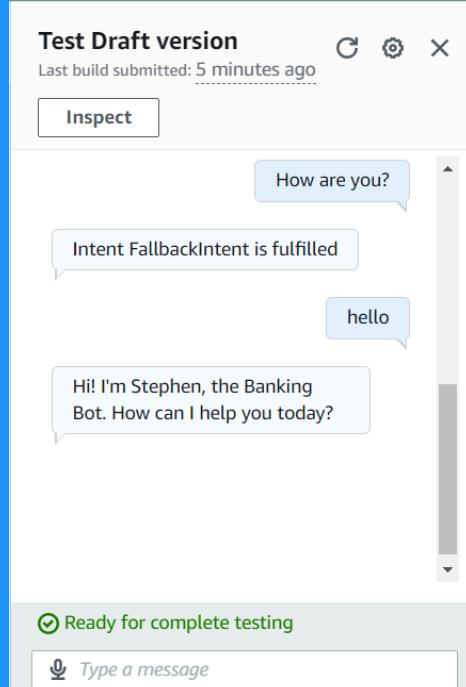




Build a Chatbot with Amazon Lex



aldricpinto1@gmail.com



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service provided by AWS that enables developers to build conversational interfaces using voice and text. It is developer friendly and provides ease in automating customer interactions without the need for large support teams.

How I used Amazon Lex in this project

In today's project I used Amazon Lex to configure a basic chatbot that was initialised to understand user intents from their inputs and provide meaningful responses.

One thing I didn't expect in this project was...

I was surprised at how easy it was to configure a chatbot in just a few minutes.

This project took me...

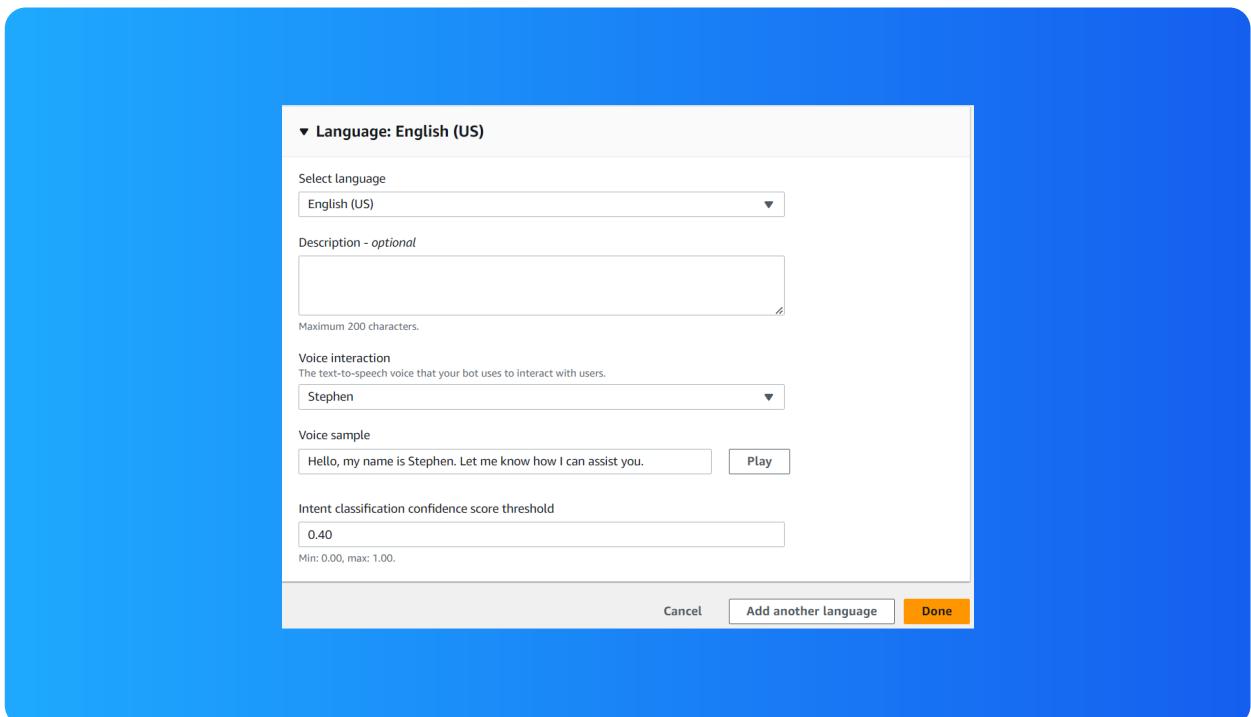
I completed this project in approximately 20-25 minutes.

Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me about 2-3 minutes.

While creating my chatbot, I also created a role with basic permissions because Amazon Lex needs to call other AWS services on my behalf.

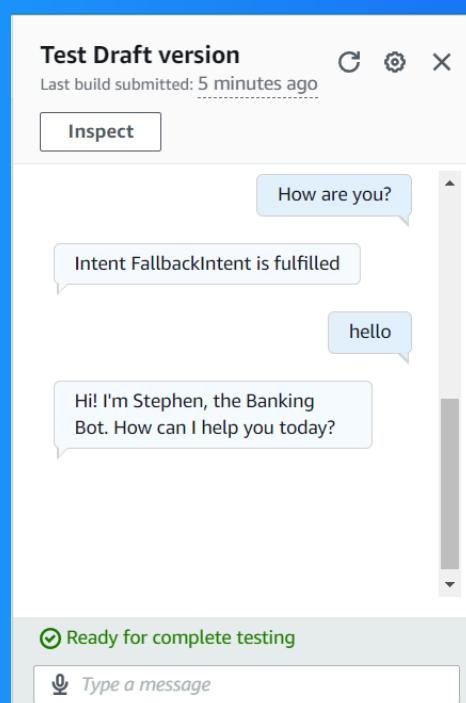
In terms of the intent classification confidence score, I kept the default value of 0.40. This means that the chatbot needs to be atleast 40% confident that it understands the user is asking for, to be able to give a response.



Intents

Intents are what the user is trying to achieve. In our case, an example of intent would be :checking bank account balance. In Amazon Lex we can define and categorise different intents, enabling the chatbot to manage requests that are related.

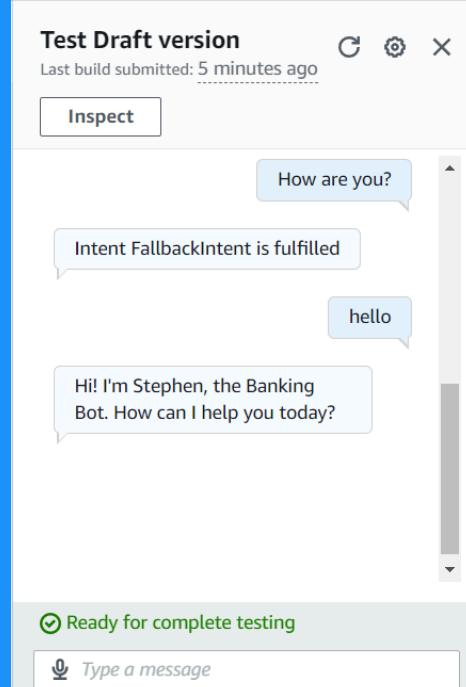
I created my first intent, WelcomeIntent, to welcome a user when they type or speak phrases like : "Hi", "Hey", "Hello", "Can you help me?"



FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter the phrases defined in the Utterances section.

My chatbot returned the error message 'Intent FallbackIntent is fulfilled' when I entered the phrase : "How are you?". This error message occurred because Lex was unable to understand the utterance.



Configuring FallbackIntent

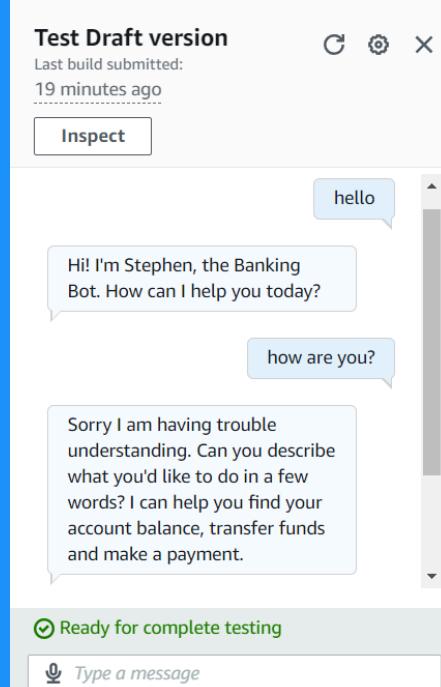
FallbackIntent is a default intent in every chatbot that gets triggered when it doesn't understand the user.

I wanted to configure FallbackIntent because, I wanted the chatbot to output a custom error message when it did not understand the user's input.

Variations

To configure FallbackIntent, I scrolled down to the closing response section and entered a custom message, that will be displayed when the chatbot does not understand the user input.

I also added variations! What this means for an end user is that, it incorporates dynamicism in responses, to user inputs that are ambiguous to the chatbot.





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

