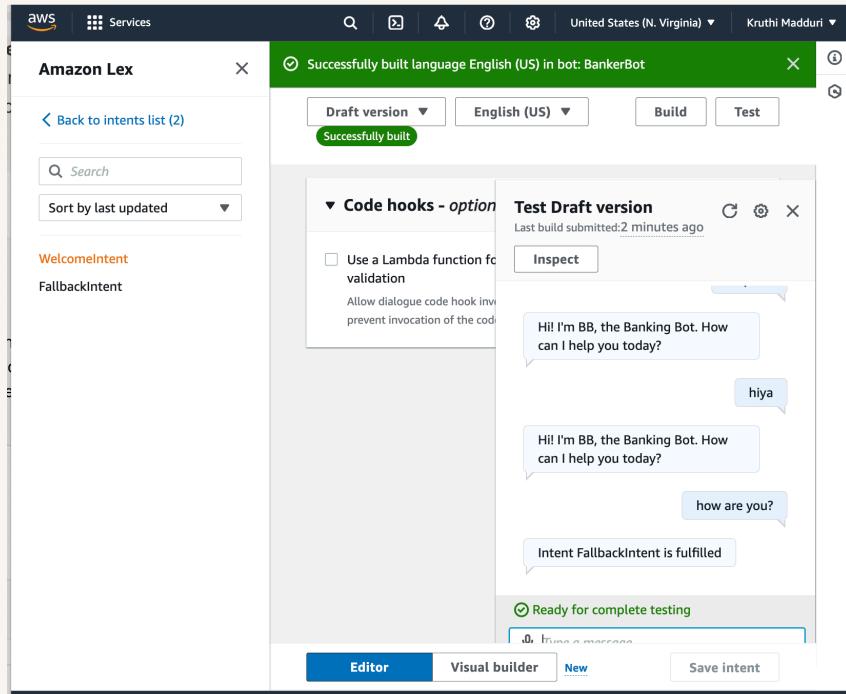




Build a Chatbot with Amazon Lex

MA

maddurishivakruthi@gmail.com



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a AWS service that provides platform to create an interface for various applications to chat or speak, enabling them to create chatbot using NLP that interacts with customers.

How I used Amazon Lex in this project

I have used Amazon Lex in today's project to create a chatbot for banking purposes, which assists users to help them with their queries in regards to banking. It answers their queries while understanding the intent otherwise throws an error.

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

One thing I didn't expect was for Amazon Lex is how it understands natural language. Lex is very useful to create chatbots and also makes it easy to create them without hassle while it also understands complex natural language.

This project took me...

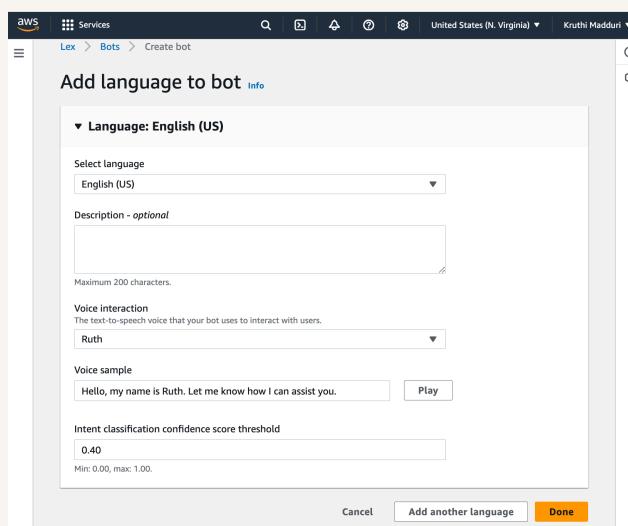
It took me maybe 30mins in total to complete this project.

Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me 5 mins.

While creating my chatbot, I also created a role with basic permissions to ensure controlled access to system resources. This approach helped manage security by granting only the necessary permissions required for the chatbot's functionality.

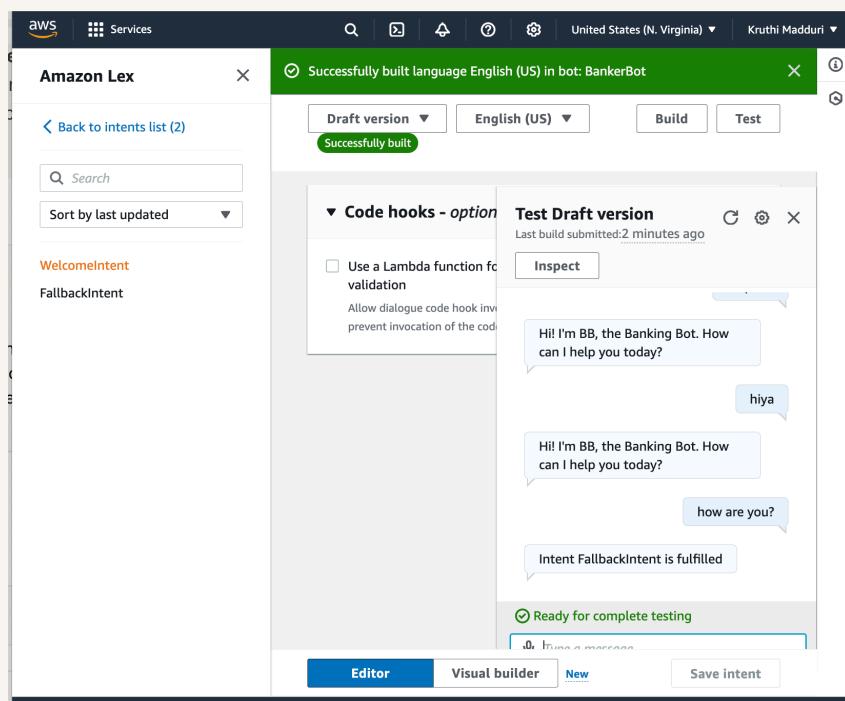
In terms of the intent classification confidence score, I kept the default value of 0.40. This means the chatbot will respond with an answer to the user's input only if it meets 0.40 confidence score. Otherwise, it throws an error message.



Intents

Intents are what the user is trying to communicate with chatbot for? Ex: To know the availability of store, to book flights tickets, to talk to an agent.

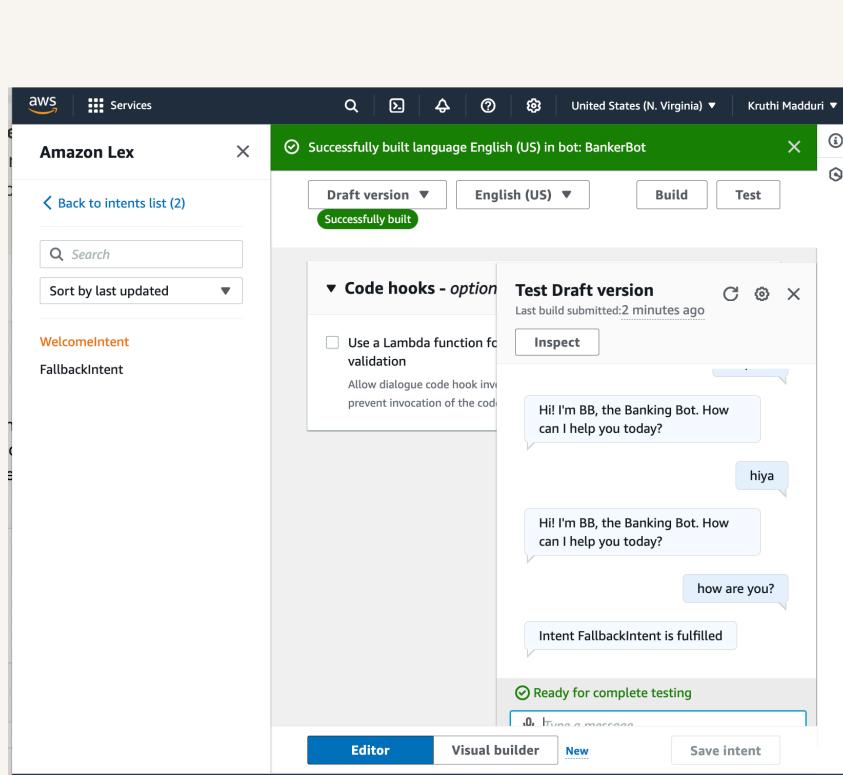
I created my first intent, WelcomeIntent, to greet the user in different ways.



FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter Hi, Hello, Hiya, Help me. But if I typed something different such as Hey, how are you? or good morning it would return an error saying 'Intent FallbackIntent is fulfilled'.

My chatbot returned the error message 'Intent FallbackIntent is fulfilled' when I entered different texts... This error message occurred because the text i entered do not fall into the same intent which i have mentioned under the welcomeintent.



Configuring FallbackIntent

FallbackIntent is a default intent in every chatbot that gets triggered when the user enters some data that chatbot is unable to understand.

I wanted to configure FallbackIntent because the error message is very confusing for the user therefore configuring it into simple terms as in 'Sorry I could not understand' would make it easier for the user to understand the error.

Variations

To configure FallbackIntent, I clicked on the FallbackIntent under Intents, scrolled down to Closing response and gave my different error scenarios responses, while adding few more under variations and then saved it, later I built and re-tested bot.

I also added variations! What this means for an end user is when he enters any data that the chatbot does not completely understand it throws the same error in different variations for the user to understand it.

The screenshot shows the AWS Lambda interface for configuring an intent. On the left, a sidebar lists intents: 'WelcomeIntent' and 'FallbackIntent'. The 'FallbackIntent' section is currently selected. The main panel displays the configuration for the 'FallbackIntent'. It includes fields for 'Message group' (set to 'Info'), 'Message' (containing 'Sorry I am having trouble understanding.'), 'Variations - optional' (containing 'Hmm could you try rephrasing that? I can' and 'Sorry I could not understand that, could you please elaborate.'), and 'More response options' (with a link to 'Add customisable payloads, SSML and card groups'). Below these are sections for 'Set values' and 'Add conditional branching'. To the right, a preview window titled 'Test Draft version' shows a conversation history: the user types 'hey', the bot responds with 'Sorry I could not understand that, could you please elaborate.', the user types 'what's good', and the bot responds with 'Hmm could you try rephrasing that? I can help you find your account balance, transfer funds and make a payment'. A green status bar at the bottom indicates 'Ready for complete testing'. Navigation tabs at the bottom include 'Editor' (which is active), 'Visual builder', and 'New'.



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

