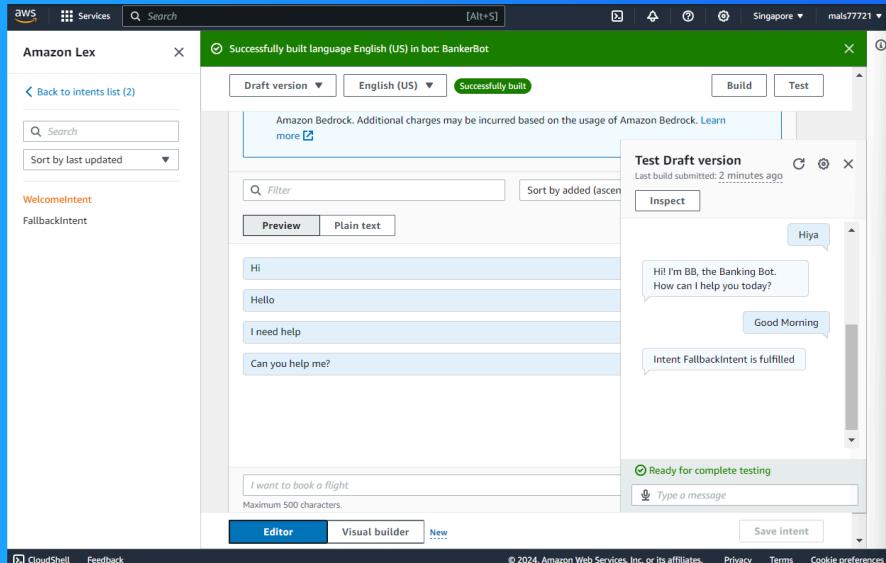




Build a Chatbot with Amazon Lex



bgef3021@gmail.com



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service that helps you create chatbots and virtual assistants. It uses natural language understanding and speech recognition to make interactions feel more human-like.

How I used Amazon Lex in this project

Today we built an interactive banking chat bot using Amazon Lex with the intent to make the chatbot sound more personable and less robotic while talking to the users.

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

After we got familiar with the service , the rest of the steps were easy to follow.

This project took me...

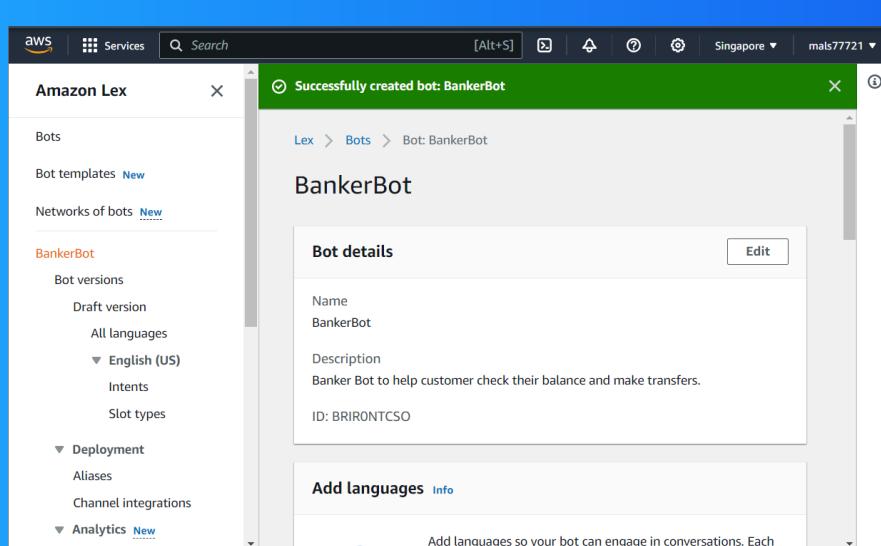
Since we sought some assistance from other sources to learn about this service while simultaneously building the chatbot, it took us nearly 3 hours to complete the project.

Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me 5-10 mins.

While creating my chatbot, I also created a role with basic permissions because this is necessary to call other AWS services on your behalf.

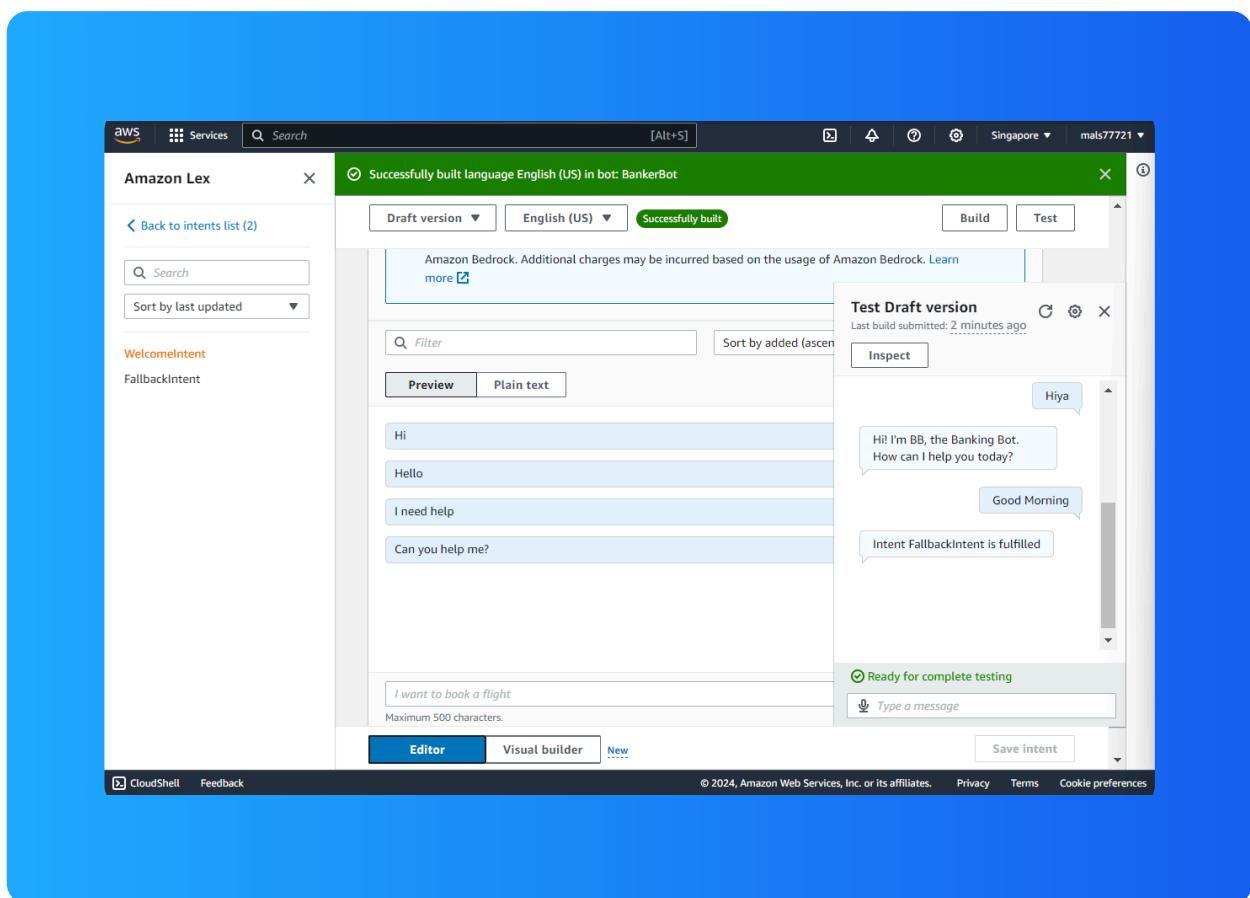
In terms of the intent classification confidence score, I kept the default value of 0.40. What this means for my chatbot is it should at least be 40% confident about the intent/goal of the chatbot user to respond.



Intents

An intent is what the user is trying to achieve in their conversation with the chatbot. So you build your chatbot by defining and categorizing different intents.

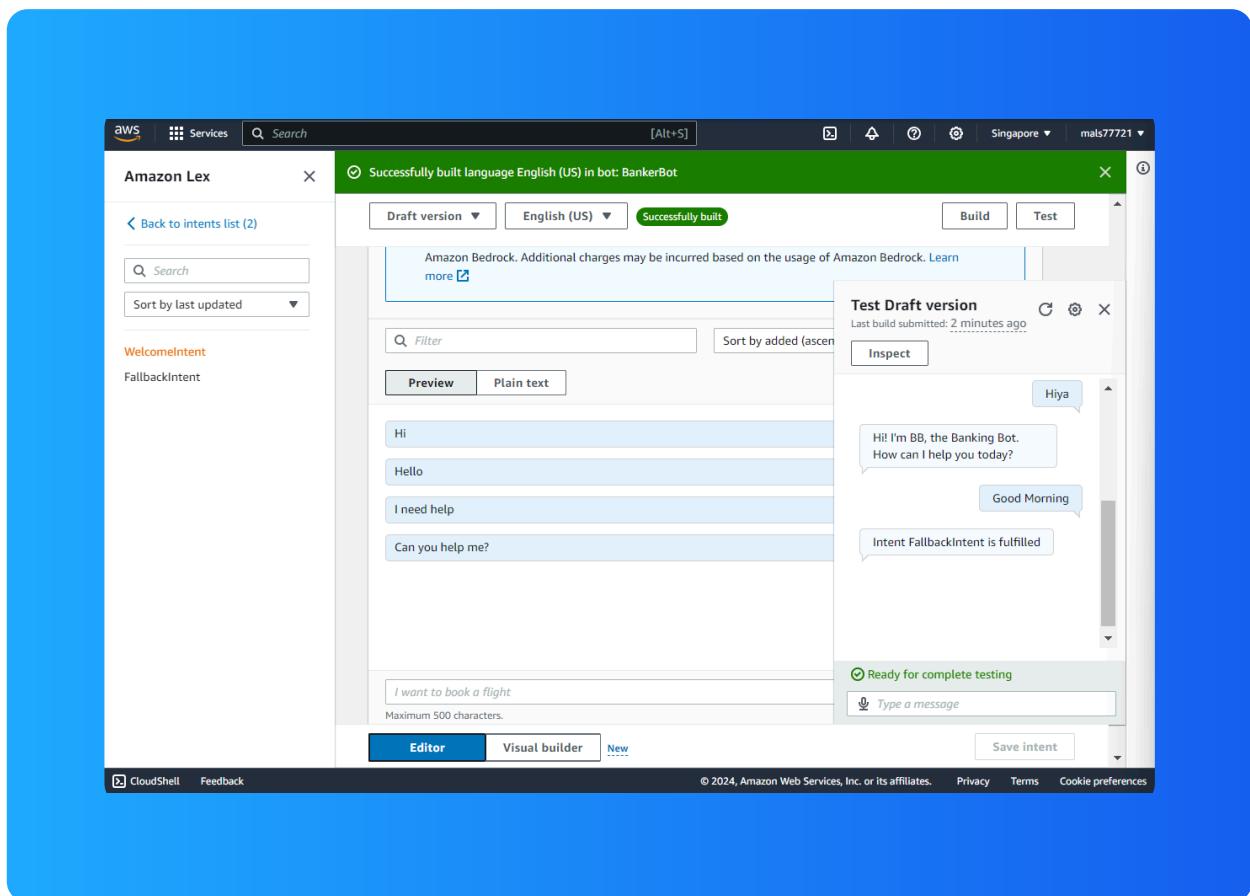
I created my first intent, WelcomeIntent, to welcome a user when they say hello.



FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter "Hello", "Hello!", "Help me", "Help m,e", "Hiya"

'Intent FallbackIntent is fulfilled` when I entered "Good Morning". This error message occurred because Amazon Lex doesn't quite recognize your utterance.



Configuring FallbackIntent

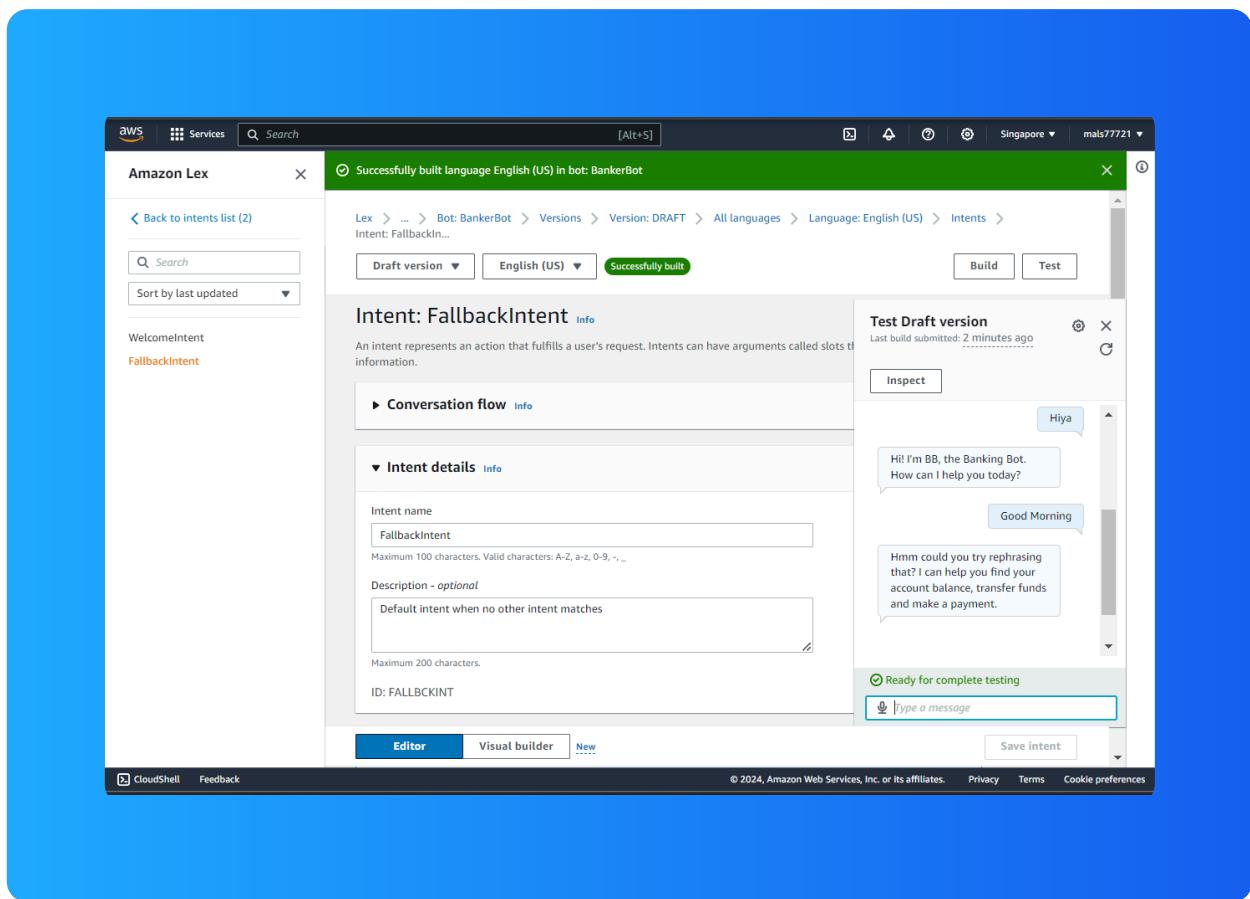
FallbackIntent is a default intent in every chatbot that gets triggered when the chat bot does not recognize the user's goal/purpose.

I wanted to configure FallbackIntent because the default closing response to the user is not easily understandable.

Variations

To configure FallbackIntent, I had to create my own closing response in the intent's set up page. "Hmm could you try rephrasing that? I can help you find your account balance, transfer funds and make a payment."

I also added variations! What this means for an end user is they get to see different forms of my chatbot's closing response.





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

