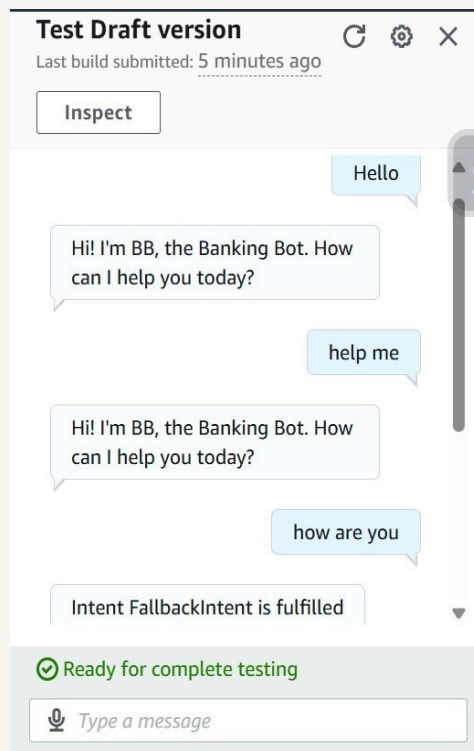


Build a Chatbot with Amazon Lex



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is an AWS service for building chatbots with voice and text using natural language processing. It integrates with AWS services for scalability and automation, enabling efficient conversational interfaces for customer support and sales.

How I used Amazon Lex in this project

In today's project, I used Amazon Lex to create a chatbot for voice and text interactions. I set up intents and responses, and integrated it with AWS Lambda for backend logic, enabling automated and scalable responses.

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

One thing I didn't expect in this project was how much fine-tuning was needed for handling unrecognized inputs. Setting up FallbackIntent and testing user variations took more time than expected, but it was essential for smooth interactions.

This project took me...

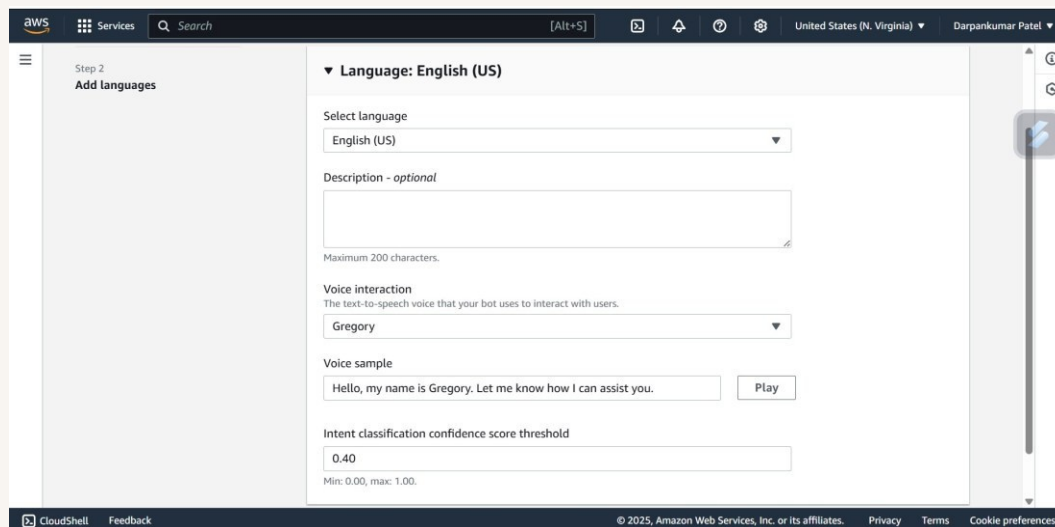
This project took me about two weeks to complete. It involved setting up Amazon Lex, defining intents, testing user inputs, and integrating with AWS Lambda. The process included fine-tuning the chatbot for accurate responses and smooth interactions.

Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me two weeks. I defined intents, set responses, integrated AWS Lambda, and debugged issues. After testing, seeing it run smoothly made all the effort worth it!

While creating my chatbot, I also created a role with basic permissions because it needed secure access to Amazon Lex, AWS Lambda, and session data. Keeping permissions minimal reduced risks while ensuring smooth functionality.

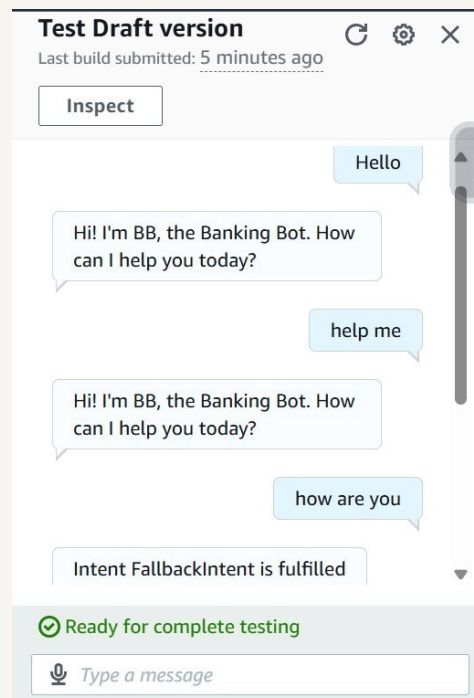
In terms of the intent classification confidence score, I kept the default value of 0.40. This means the bot considers an intent valid if the confidence is 40% or higher. If it's lower, the bot may not classify the intent, indicating uncertainty.



Intents

Intents are the goals or actions a user wants to achieve when interacting with a chatbot. They represent the purpose behind a user's input, like asking for information or making a request, and are linked to specific responses or actions.

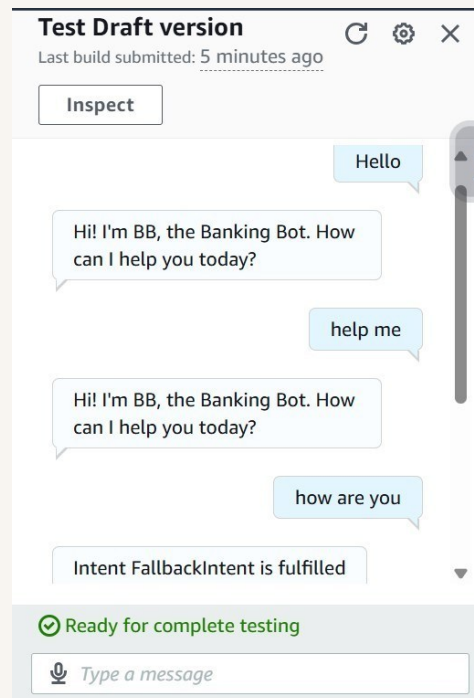
I created my first intent, `WelcomeIntent`, to greet users and initiate the conversation. It triggers when the user starts the interaction, providing a friendly message to engage them, like "Hello! How can I assist you today?"



FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter greetings like "Hello," "Hi," "I need help," or "Can you help me?." These trigger the WelcomeIntent, allowing the bot to greet users and start the conversation.

My chatbot returned the error message "Intent FallbackIntent is fulfilled" when I entered "How are you?" This error occurred because the input didn't match any defined intents, triggering the fallback intent to handle unrecognized queries.



Configuring FallbackIntent

FallbackIntent is a default intent in every chatbot that gets triggered when the user's input doesn't match any defined intents. It handles unclear queries by responding with a message like, "I'm sorry, I didn't understand that."

I wanted to configure FallbackIntent because it lets my chatbot handle unrecognized inputs. It responds gracefully when the bot doesn't understand, prompting users to rephrase or offering a helpful message.

Variations

To configure FallbackIntent, I created a default response that triggers when the bot doesn't recognize user input. I set up a simple message, like "I'm sorry, I didn't understand that," to prompt users to try again or rephrase their query.

I also added variations! What this means for an end user is that the chatbot can recognize different ways of asking the same question, like "What's the weather?" or "How's the weather?" This makes interactions feel more natural and flexible.

☑ Successfully built language English (US) in bot: BankerBot

Draft version ▾ English (US) ▾ Successfully built

▼ **Response sent to the user after the intent is fulfilled**
Message: *Sorry I am having trouble understanding. Can you describe what you'd like to do in a few words? I can help you find your account balance, transfer funds and make a payment.*

▼ **Message group** [Info](#)
You can define a text message group to respond using plain text.

Message

Sorry I am having trouble understanding. Can you describe what you'd like to do in a few words? I can help you find

▼ **Variations - optional**

Hmm could you try rephrasing that? I can help you find your account balance, transfer funds and make a payment.

More response options

Test Draft version Last build submitted: 4 minutes ago

Inspect

how are you

Hil I'm BB, the Banking Bot. How can I help you today?

what's good

Hmm could you try rephrasing that? I can help you find your account balance, transfer funds and make a payment.

Ready for complete testing

Type a message