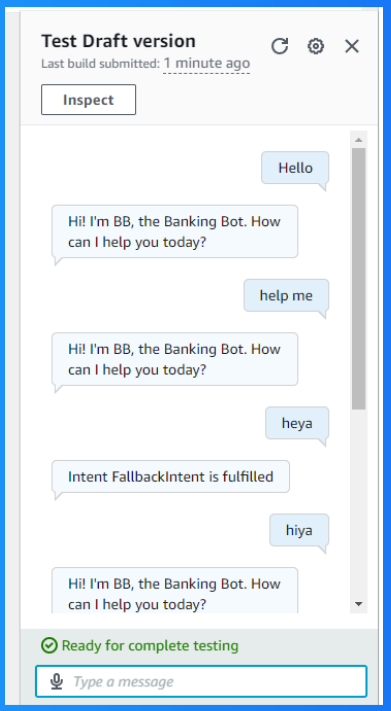
NextWork.org

Build a Chatbot with Amazon Lex

Sai Prasanth Reddy



Sai Prasanth Reddy
NextWork Student

Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for creating chatbots that understand natural language through voice and text. It's useful for building interactive applications, automating customer support, and providing personalized user experiences.

How I used Amazon Lex in this project

I used Amazon Lex in today's project to create a chatbot called BankerBot, define intents like WelcomeIntent, configure responses, and set up fallback handling to manage user interactions effectively.

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

One thing I didn't expect in this project was encountering the "Intent FallbackIntent is fulfilled" response when user inputs didn't match any defined intents, highlighting the need to refine intent settings and responses further.

This project took me...

This project took me about 45 minutes to complete, including setting up the chatbot with Amazon Lex, configuring intents, testing the bot, and handling unexpected fallback responses.

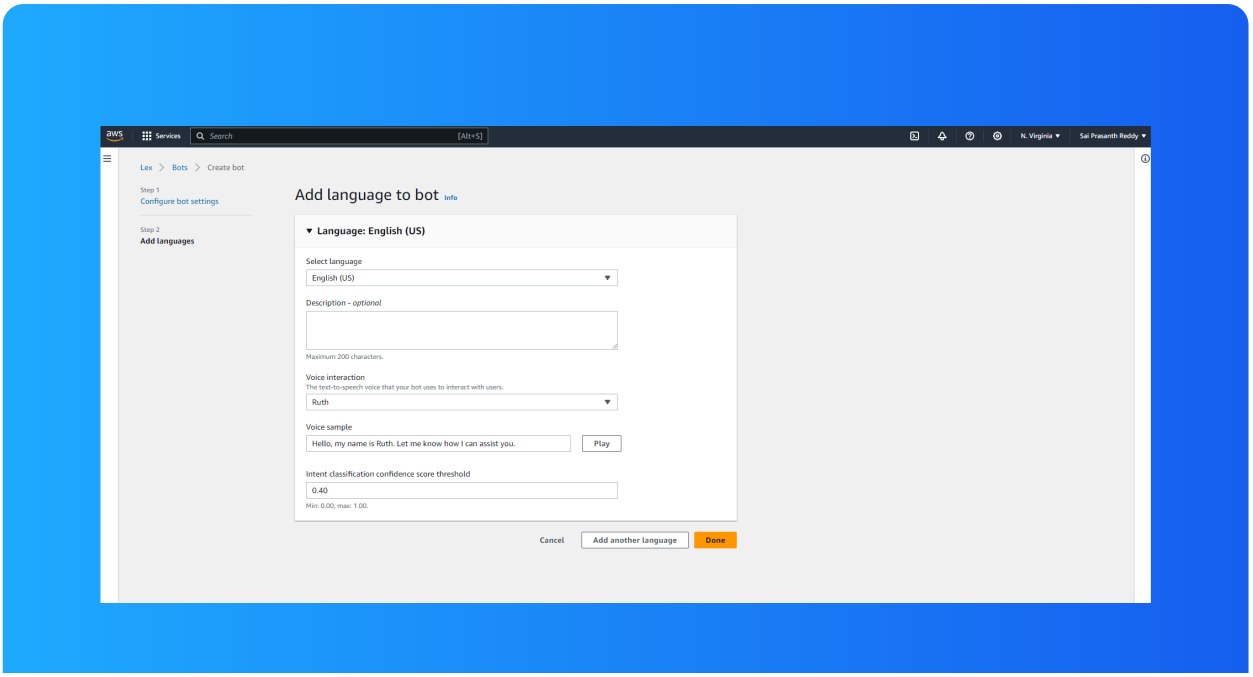
Sai Prasanth Reddy
NextWork Student

Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me about 10 minutes, including creating the bot and configuring basic settings.

While creating my chatbot, I also created a role with basic permissions because it allows Amazon Lex to access necessary AWS resources securely, such as logging interactions in CloudWatch and storing session data, ensuring proper functionality.

In terms of the intent classification confidence score, I kept the default value of 0.40. This means the chatbot will trigger an intent if it's at least 40% confident that the user input matches that intent, balancing accuracy and flexibility.

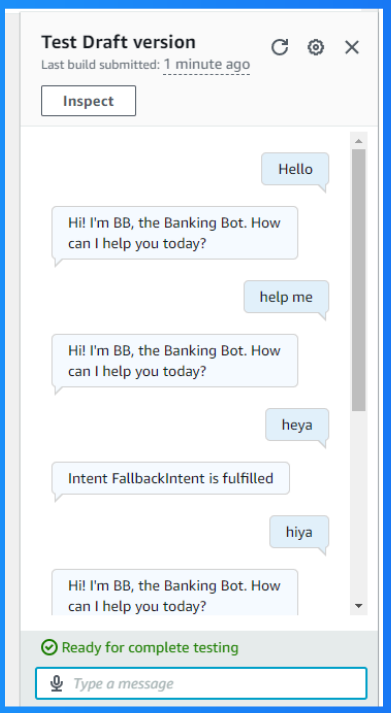


Sai Prasanth Reddy
NextWork Student

Intents

Intents are specific goals or actions that a user wants to achieve when interacting with a chatbot, such as booking a flight or checking the weather. They guide the chatbot on how to respond to user inputs effectively.

I created my first intent, WelcomeIntent, to greet users when they say hello, with a 40% confidence score threshold to ensure the chatbot responds appropriately even with varied user inputs.

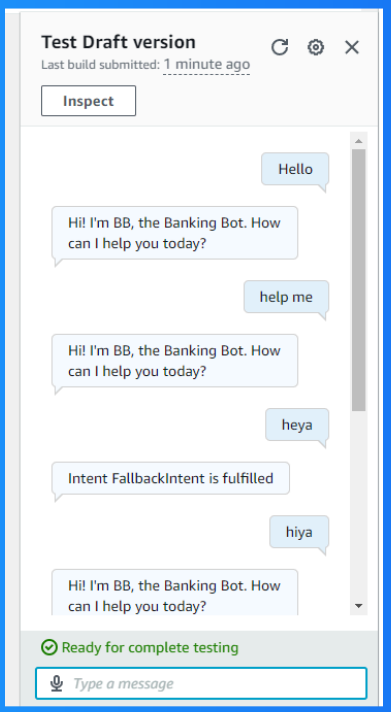


Sai Prasanth Reddy
NextWork Student

FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter "hello," "help me," or "hiya," showing it can recognize different greetings and start a conversation.

My chatbot returned the error message `Intent FallbackIntent is fulfilled` when I entered "heya," "how are you," or "good morning." This error message occurred because these inputs did not match any defined intents with sufficient confidence.



Sai Prasanth Reddy
NextWork Student

Configuring FallbackIntent

FallbackIntent is a default intent in every chatbot that gets triggered when the user's input doesn't match any defined intents with sufficient confidence, allowing the bot to handle unrecognized or ambiguous inputs gracefully.

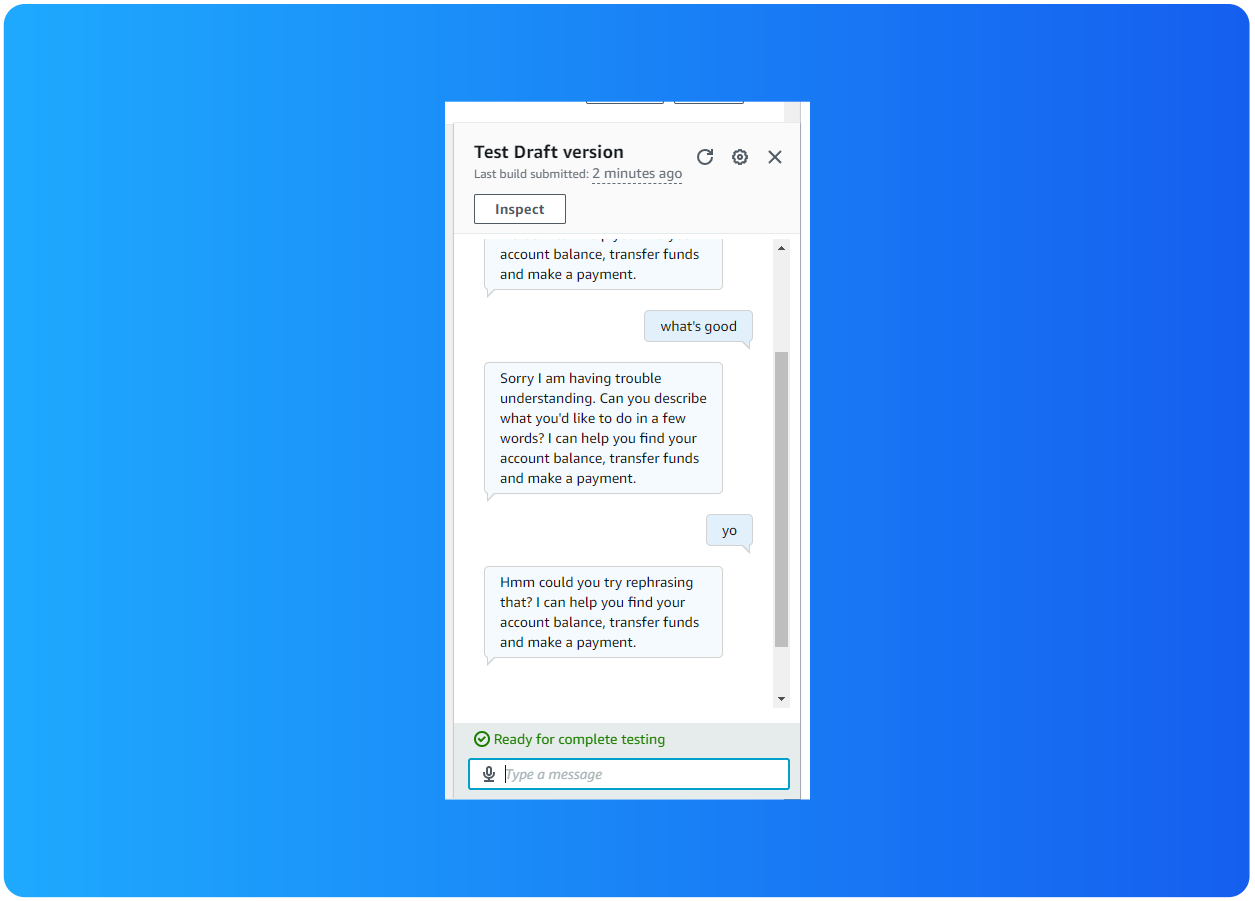
I wanted to configure FallbackIntent because it helps the chatbot manage unexpected or unclear user inputs, ensuring a smoother conversation flow by providing appropriate responses when no other intent is matched.

Sai Prasanth Reddy
NextWork Student

Variations

To configure FallbackIntent, I added a message in the closing response and included two more variations to ensure the chatbot provides helpful feedback when it can't understand the user's input.

I also added variations! What this means for an end user is that the chatbot can provide different responses for similar inputs, making the interaction feel more natural and dynamic by avoiding repetitive answers.



Everyone should be in a job they love.

Check out nextwork.org for more projects