

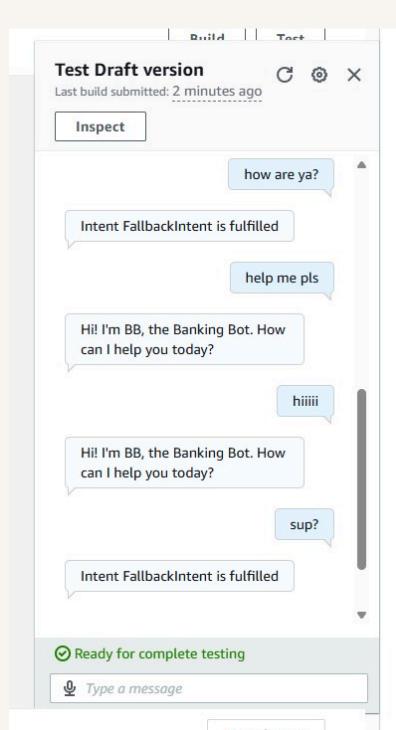


[nextwork.org](https://nextwork.org)

# Build a Chatbot with Amazon Lex



Gursimran Singh



A circular profile picture of a man with a beard and a dark turban, wearing a dark jacket.

**Gursimran Singh**  
NextWork Student

[nextwork.org](http://nextwork.org)

# Introducing Today's Project!

## What is Amazon Lex?

Amazon Lex is a service by AWS that lets you build chatbots and voice assistants. It uses natural language understanding to help your bot understand and respond to user input in a human-like way through text or speech.

## How I used Amazon Lex in this project

Services I used were Amazon Lex and IAM Roles. Key concepts I learnt include creating intents, configuring FallbackIntent, handling user input, and making the bot respond naturally using variations and confidence scores.

## One thing I didn't expect was...

One thing I didn't expect in this project was how quickly I could build a working chatbot using Amazon Lex. The interface was simple, and setting up intents and responses felt much easier than I thought it would be.



**Gursimran Singh**  
NextWork Student

[nextwork.org](http://nextwork.org)

## This project took me...

This project took me approximately 1 hour. The most challenging part was fine-tuning the intents to handle varied user inputs. It was most rewarding to see the bot respond accurately and feel like a real conversation.



# Setting up a Lex chatbot

I created my chatbot from scratch with Amazon Lex. Setting it up took me around 15 minutes — the interface was super user-friendly, and the steps were pretty straightforward to follow.

While creating my chatbot, I also created a role with basic permissions because Amazon Lex needs to invoke AWS Lambda functions to process user input and generate dynamic responses. This helps make the bot more interactive and intelligent.

In terms of the intent classification confidence score, I kept the default value of 0.40. This means Lex will only accept a predicted intent if it's at least 40% confident, helping avoid wrong responses when the user input is unclear.

**Gursimran Singh**  
NextWork Student

[nextwork.org](http://nextwork.org)

The screenshot shows the 'Add language to bot' configuration page in the AWS Lex console. The top navigation bar includes links for Home, IAM, EC2, S3, Route 53, Lambda, VPC, AWS Organizations, and CloudFormation. Below the navigation, the breadcrumb trail shows Lex > Bots > Create bot. The main form is titled 'Add language to bot' with an 'Info' link. It is divided into two steps: Step 1 (Configure bot settings) and Step 2 (Add languages). Step 2 is currently active, showing the 'Add languages' section. Under 'Language: English (US)', the 'Select language' dropdown is set to 'English (US)'. There is an optional 'Description' field with a note about character limits. The 'Voice interaction' section allows selecting a text-to-speech voice, with 'Stephen' chosen. A 'Voice sample' field contains the text 'Hello, my name is Stephen. Let me know how I can assist you.' and a 'Play' button. An 'Intent classification confidence score threshold' input field is set to '0.40'. At the bottom right are 'Cancel', 'Add another language', and a prominent orange 'Done' button.

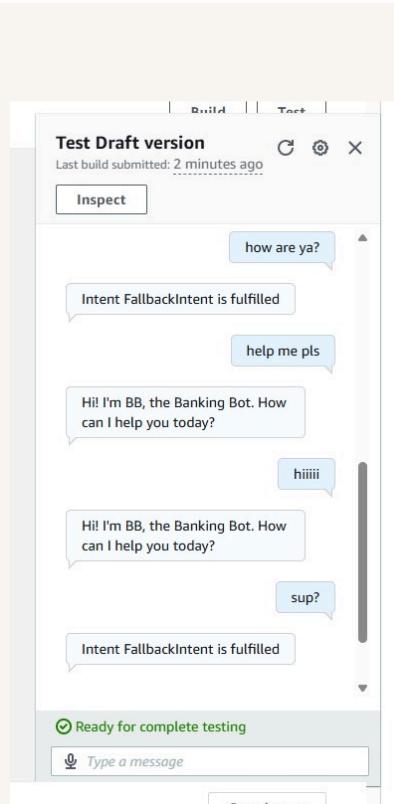
**Gursimran Singh**  
NextWork Student

[nextwork.org](http://nextwork.org)

# Intents

Intents are the goals or purposes behind what a user says. For example, if someone types "Check my balance," the intent is to get account info. Intents help the bot understand and respond to user needs accurately.

I created my first intent, WelcomeIntent, to make BankerBot greet users when they say hi or hello. It helps start the conversation on a friendly note and lets users know the bot is ready to help.



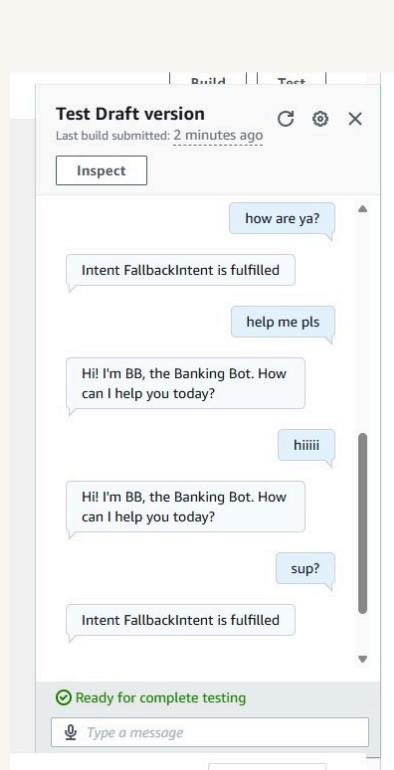
**Gursimran Singh**  
NextWork Student

[nextwork.org](http://nextwork.org)

## FallbackIntent

I launched and tested my chatbot, which could respond successfully if I enter greetings like "hiii," "hiiii," or "help me pls." These matched the WelcomeIntent, while other casual phrases triggered the fallback.

My chatbot returned the error message 'Intent FallbackIntent is fulfilled' when I entered casual phrases like "yo," "sup?," or "good morning." This error message occurred because those inputs didn't match any defined intent in the bot.





# Configuring FallbackIntent

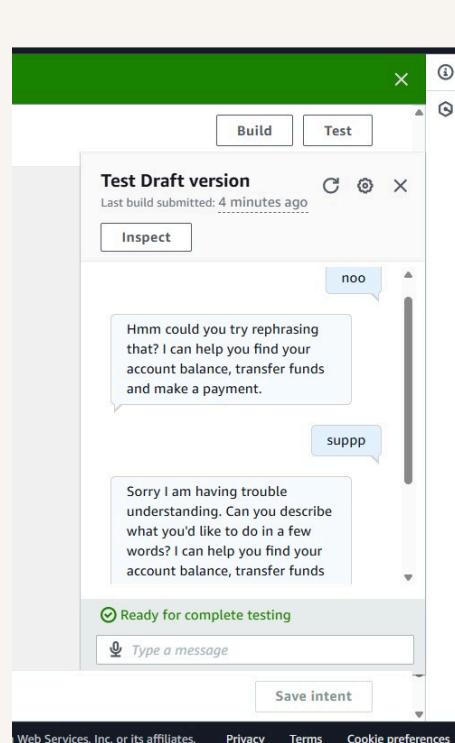
FallbackIntent is a default intent in every chatbot that gets triggered when the user's input doesn't match any of the defined intents with enough confidence, helping the bot handle unexpected or unclear messages.

I wanted to configure FallbackIntent because it allows my chatbot to reply with a friendly message when it doesn't understand the user, instead of just showing a default error. It helps keep the conversation smooth and clear.

## Variations

To configure FallbackIntent, I opened the FallbackIntent settings in Amazon Lex and replaced the default message with a clearer, more helpful response to guide users when their input isn't recognized.

I also added variations! What this means for an end user is that the chatbot can understand different ways of saying the same thing, making it feel more natural and responsive to how real people talk.





[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

