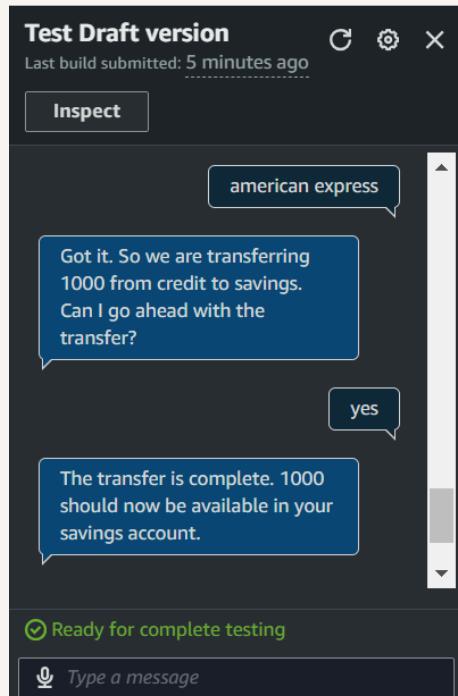


[nextwork.org](http://nextwork.org)

# Build a Chatbot with Multiple Slots

SI

sidhesh2508@gmail.com



# Introducing Today's Project!

## What is Amazon Lex?

Amazon Lex is a service for building conversational AI chatbots using voice and text. It is useful because it automates customer interactions, integrates with AWS services, and scales easily for businesses.

## How I used Amazon Lex in this project

I used Amazon Lex in today's project by deploying it via CloudFormation, automating the creation of the bot, intents, slots, and Lambda integration. This ensured a structured, repeatable, and scalable setup.

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

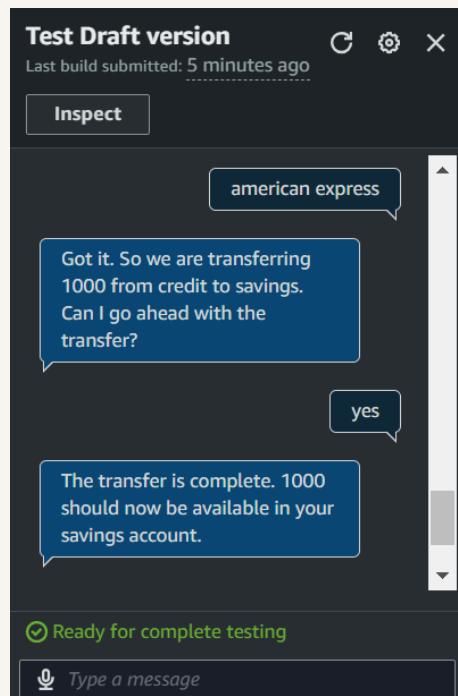
One thing I didn't expect in this project was the permission denied error after deployment. I had to troubleshoot and update the Lambda resource policy to allow LexV2 to invoke the function properly.

## This project took me...

This project took me a few hours to set up the CloudFormation template, deploy the Lex bot, and integrate it with Lambda. Troubleshooting the permission denied error added extra time.

# TransferFunds

An intent I created for my chatbot was TransferFunds, which allows users to transfer money between accounts by specifying the amount and account type, making transactions quick and seamless.



# Using multiple slots

For this intent, I had to use the same slot type accountType twice. This is because the user needs to specify both the source account (where the money is deducted from) and the destination account (where the money is sent).

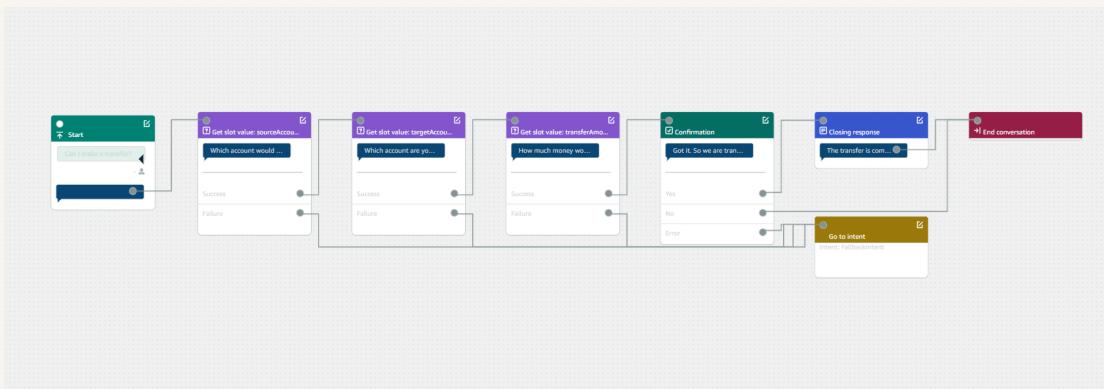
I also learned how to create confirmation prompts, which are messages that ask users to confirm their request before the chatbot proceeds with an action, helping to prevent errors and ensure accuracy.

The screenshot shows the configuration of a Confirmation intent. At the top, it says "Confirmation" with an "Info" link and a status indicator "Active". Below that, a note says "Prompts help to clarify whether the user wants to fulfill the intent or cancel it." The main area is divided into two sections: "Prompts to confirm the intent" and "Responses sent when the user declines the intent". Under "Prompts to confirm the intent", there is a "Message" field containing "Got it. So we are transferring {transferAmount} from {sourceAccountType} to {targetAccountType}. Can I go ahead with the transfer?". Under "Responses sent when the user declines the intent", there is a "Message" field containing "The transfer has been cancelled.". There are also sections for "Confirmation prompt" (with a note about what the bot says to prompt confirmation) and "Decline response" (with a note about what the bot says if the user says NO). At the bottom, there is an "Advanced options" section with a note about configuring prompts and responses.

# Exploring Lex features

Lex also has a special conversation flow feature that helps manage dialogue by guiding users through a structured interaction, handling slot filling, confirmations, and responses for a smooth user experience.

You could also set up your intent using a visual builder! A visual builder provides a drag-and-drop interface to design conversation flows, making it easier to define slots, prompts, and responses without coding.



# AWS CloudFormation

AWS CloudFormation is a service that enables automated provisioning and management of AWS infrastructure using JSON or YAML templates.

I used CloudFormation to deploy an Amazon Lex V2 chatbot, defining resources like the bot, intents, slots, and Lambda functions. This automated setup ensured consistency, repeatability, and reduced manual effort.

Intents (5) <a href="#">Info</a>		
An intent represents an action that the user wants to perform.		
<input type="text"/> <a href="#">Search intents</a>		
Name	Description	Last edited
TransferFunds	Help user transfer funds between bank accounts	18 minutes ago
FollowupCheckBalance	Intent to allow a follow-up balance check request without authentication	18 minutes ago
CheckBalance	Intent to check the balance in the specified account type	18 minutes ago
Welcome	Welcome intent	18 minutes ago
FallbackIntent	Default fallback intent when no other intent matches	18 minutes ago

# The final result!

Re-building my bot with CloudFormation took me a few hours, as I had to define the bot structure, intents, slots, and Lambda functions.

There was an error after I deployed my bot! The error was a permission denied issue, which prevented Lex from invoking the Lambda function. I fixed this by updating the Lambda resource policy to allow LexV2 to invoke it.

**Edit resource-based policy**

**Edit policy statement**

AWS account  
Grant permissions to another AWS account, user, or role.

AWS service  
Grant permissions to another AWS service.

Function URL  
Grant permissions to invoke your function through the function URL.

**Service**  
The AWS service to grant permissions to.  
 Other

**Statement ID**  
Enter a unique statement ID to differentiate this statement within the policy.

**Principal**  
The service principal for this AWS service. [Learn more](#)

**Source ARN**  
The ARN for a resource. Find the ARN in the related service console.

**Action**  
Choose an action to allow.



NextWork.org

# **Everyone should be in a job they love.**

Check out nextwork.org for  
more projects

