



Build a Chatbot with Multiple Slots



sru anu

Test Draft version
Last build submitted: 1 minute ago

Inspect

Which account are you transferring to?

savings

How much money would you like to transfer?

5000

Got it. So we are transferring 5000 from Checking to Savings. Can I go ahead with the transfer?

yes

The transfer is complete. 5000 should now be available in your

Ready for complete testing

Type a message

A screenshot of a chatbot test interface. The interface has a light blue header with the title 'Test Draft version' and a timestamp 'Last build submitted: 1 minute ago'. Below the header is a button labeled 'Inspect'. The main area shows a conversation between a user and a bot. The user asks 'Which account are you transferring to?' and the bot replies 'savings'. The user then asks 'How much money would you like to transfer?' and the bot replies '5000'. The bot then sends a message stating 'Got it. So we are transferring 5000 from Checking to Savings. Can I go ahead with the transfer?'. The user replies 'yes'. Finally, the bot confirms the transfer with the message 'The transfer is complete. 5000 should now be available in your'. At the bottom of the interface, there is a green checkbox labeled 'Ready for complete testing' which is checked, and a text input field with the placeholder 'Type a message'.



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for building chatbots using voice and text. It's useful for creating conversational interfaces that understand natural language and integrate with AWS services.

How I used Amazon Lex in this project

Amazon Lex is a service for building chatbots using voice and text. It's useful for creating conversational interfaces that understand natural language and integrate with AWS services.

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

One thing I didn't expect was the complexity of managing context tags and their impact on conversation flow, which required more fine-tuning than anticipated.

This project took me...

One thing I didn't expect was the complexity of managing context tags and their impact on conversation flow, which required more fine-tuning than anticipated.

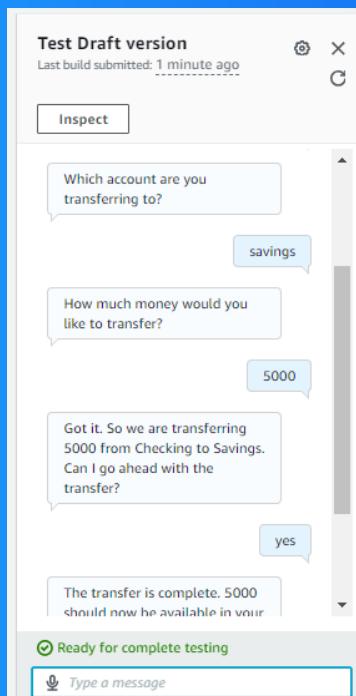


sru anu
NextWork Student

NextWork.org

TransferFunds

An intent I created for my chatbot was TransferFunds, which will handle user requests to transfer money between accounts. It processes details such as the source and destination accounts and the amount to be transferred.



Using multiple slots

For this intent, I had to use the same slot type twice. This is because the slot type was needed to capture both the source and destination account details, ensuring accurate transfer processing.

I also learned how to create confirmation prompts, which are used to verify user inputs before finalizing an action. They help ensure that the user's request is accurate and prevent mistakes by asking for confirmation.

Confirmation Info Active

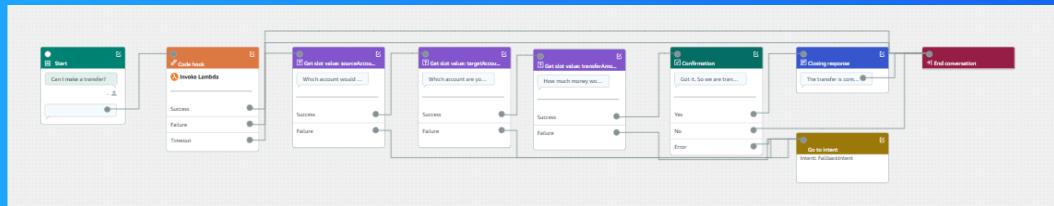
Prompts help to clarify whether the user wants to fulfill the intent or cancel it.

Prompts to confirm the intent <small>Message: Got it. So we are transferring {transferAmount} from {sourceAccountType} to {targetAccountType}. Can I go ahead?</small>	Responses sent when the user declines the intent <small>Message: The transfer has been cancelled.</small>
Confirmation prompt What will the bot say to prompt the user to confirm this intent. <div style="border: 1px solid #ccc; padding: 5px; width: 100%;">Got it. So we are transferring {transferAmount} from {sourceAccountType} to {targetAccountType}. Can I go ahead</div>	
Decline response What will the bot say if the user says NO to the confirmation prompt. <div style="border: 1px solid #ccc; padding: 5px; width: 100%;">The transfer has been cancelled.</div>	
Advanced options Configure confirmation prompts and decline responses.	

Exploring Lex features

Lex also has a special conversation flow feature that manages how interactions progress between the user and the chatbot. It controls the sequence of intents, contexts, and responses to ensure a smooth and logical conversation.

You could also set up your intent using a visual builder! A visual builder provides a graphical interface for designing and organizing intents, slots, and responses, making it easier to create and manage conversational flows.





AWS CloudFormation

AWS CloudFormation is a service that automates the creation and management of AWS resources. It uses templates to define and provision infrastructure, making it easier to deploy and manage complex environments consistently.

I used CloudFormation to automate the setup of AWS resources, including Lambda functions and Lex bots, ensuring consistent and efficient deployment through templates.

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



The final result!

Re-building my bot with CloudFormation took me around 30 minutes, including setting up the templates and deploying the necessary AWS resources.

There was an error after I deployed my bot! The error was due to incorrect resource configurations in the CloudFormation template. I fixed this by revising the template, correcting the configurations, and redeploying the stack.

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.

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

