



NextWork.org

Build a Chatbot with Multiple Slots



ajv.singh3107@gmail.com



ajv.singh3107@gmail.com

NextWork Student

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

Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for building conversational interfaces with voice and text. It simplifies creating chatbots and virtual assistants by using natural language processing and integrating with other AWS services.

How I used Amazon Lex in this project

I created a banking query chatbot with Amazon Lex. It answers questions about account balance, transaction history, and loans, using AWS Lambda for backend processing, providing efficient, scalable customer support through natural language.

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

One thing I didn't expect was the complexity of integrating Amazon Lex with existing banking systems. It required careful mapping of intents and smooth communication between Lambda functions and the database.

This project took me...

This project took me about 2 hours. The bulk of the time was spent setting up intents, configuring Lambda functions, and testing the integration with banking

systems to ensure accurate and efficient responses to customer queries.



ajv.singh3107@gmail.com
NextWork Student

[NextWork.org](#)

TransferFunds

TransferFunds is an intent I created for my chatbot to handle money transfers. It collects details like the amount and recipient account, then triggers an AWS Lambda function to process the transfer.



ajv.singh3107@gmail.com
NextWork Student

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

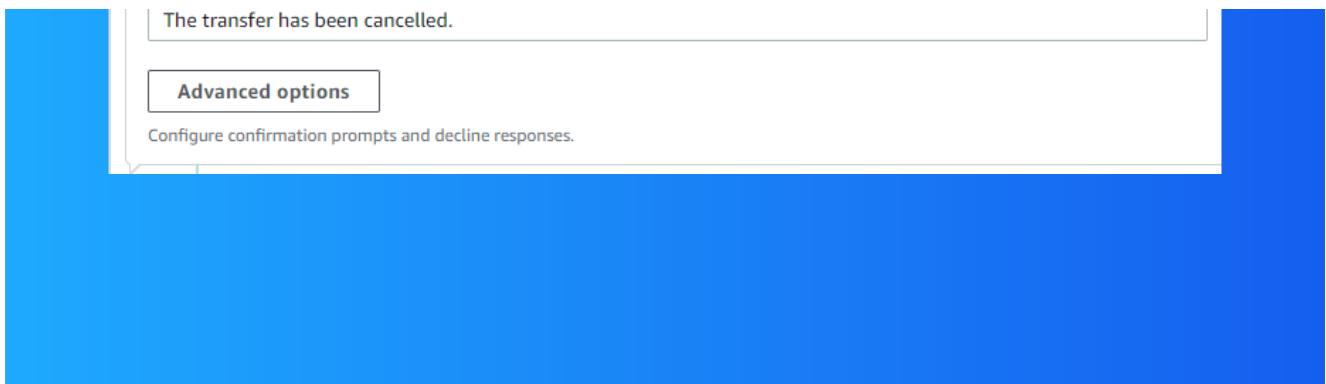
Using multiple slots

For this intent, I had to use the same slot type twice. This is because both the source and destination accounts required the same slot type (account number), ensuring that both account details could be captured and validated in one intent.

I also learned how to create confirmation prompts, which are used to confirm user inputs before proceeding. They help ensure accuracy by asking users to verify details like amounts or account numbers before the chatbot continues with the action.

The screenshot shows the configuration for a transfer intent. It includes sections for confirming the intent, sending decline responses, a confirmation prompt, and a decline response.

Prompts to confirm the intent Message: Got it. So we are transferring {transferAmount} from {sourceAccountType} to {targetAccountType}. Can I go ahead?	Responses sent when the user declines the intent Message: The transfer has been cancelled.
Confirmation prompt What will the bot say to prompt the user to confirm this intent. Got it. So we are transferring {transferAmount} from {sourceAccountType} to {targetAccountType}. Can I go ahead	
Decline response What will the bot say if the user says NO to the confirmation prompt.	



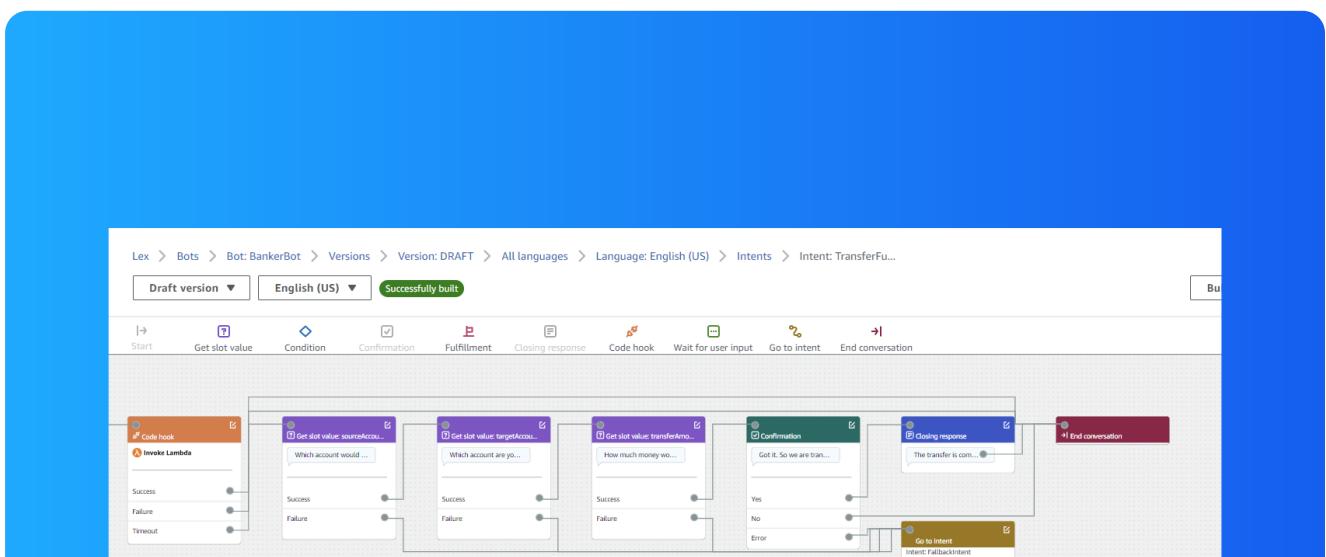
ajv.singh3107@gmail.com
NextWork Student

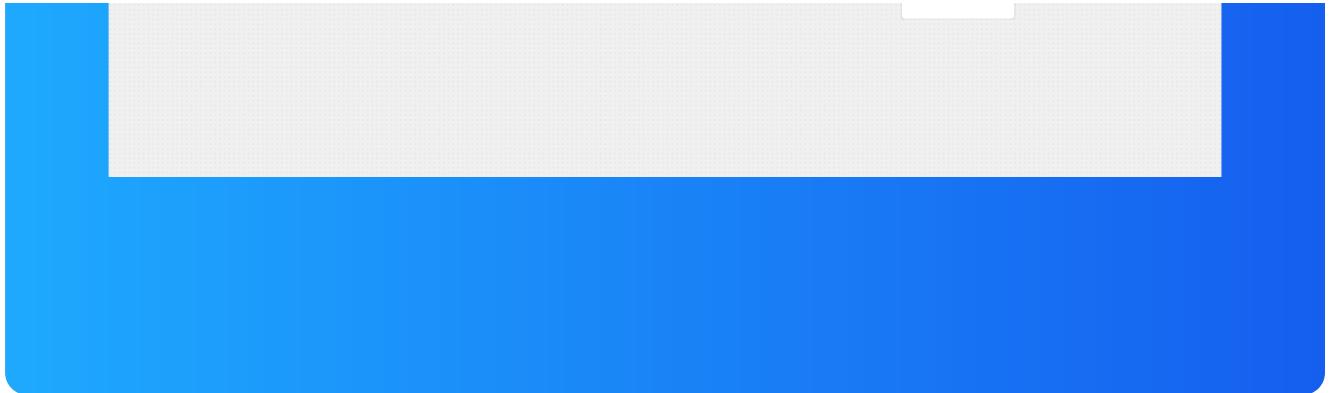
[NextWork.org](https://www.nextwork.org)

Exploring Lex features

Lex's conversation flow feature helps guide the chatbot's interaction, managing how conversations progress by collecting information, confirming details, and triggering specific intents based on user responses.

The visual builder offers a graphical interface to design and manage conversation flows, slots, prompts, and responses, making it easier to visualize and configure your chatbot's interactions in a step-by-step manner.





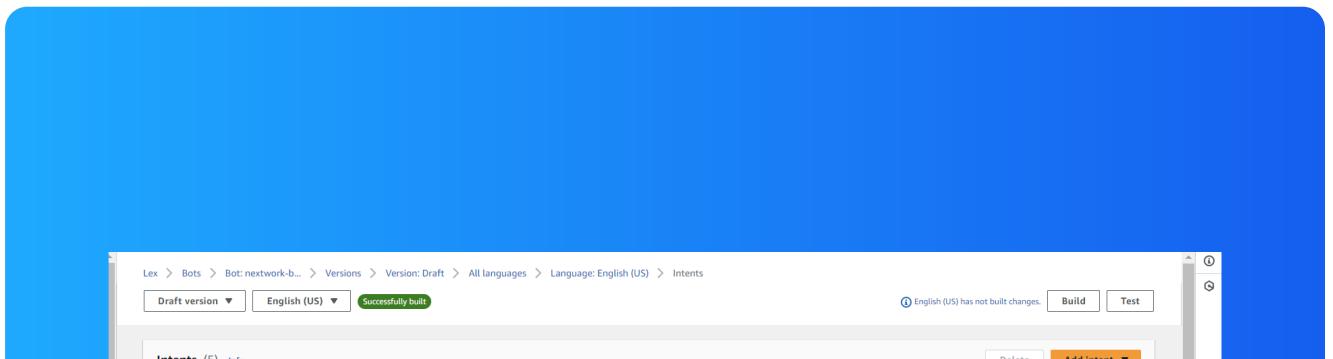
ajv.singh3107@gmail.com
NextWork Student

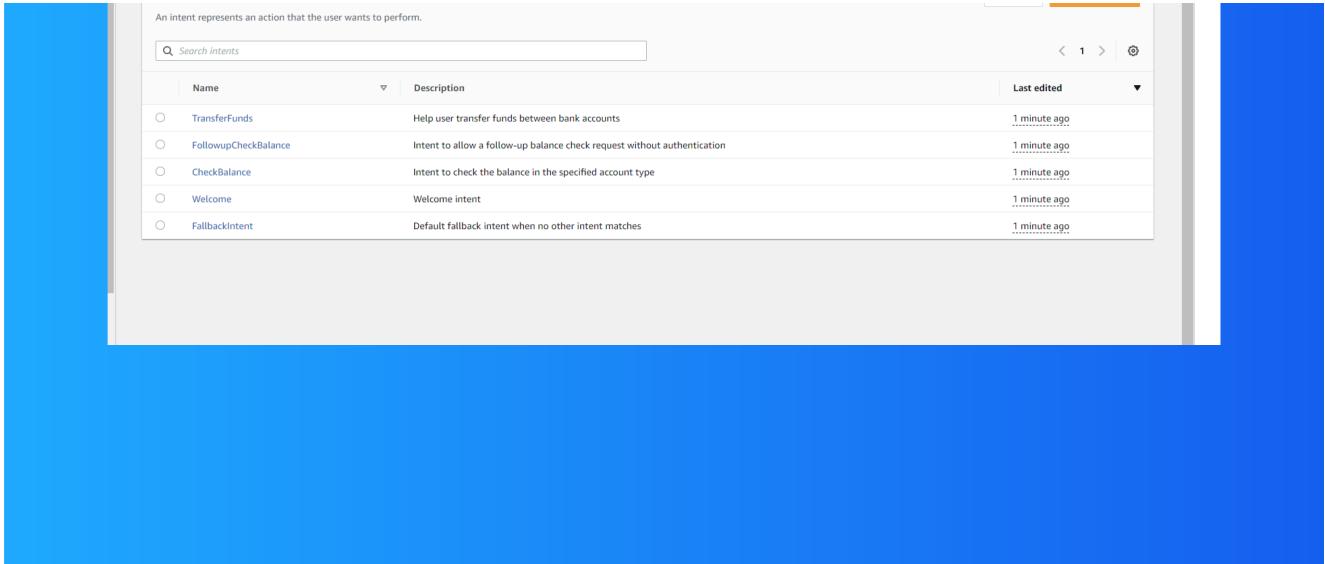
[NextWork.org](https://www.nextwork.org)

AWS CloudFormation

AWS CloudFormation is a service that allows you to define and provision AWS infrastructure using code. It enables automation of infrastructure deployment, ensuring consistency across environments by using templates written in JSON or YAML.

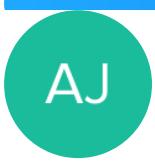
I used CloudFormation to automate the setup of AWS resources for my chatbot project. By creating templates, I defined and provisioned necessary services like Lambda functions and Lex bots, ensuring a repeatable and scalable infrastructure setup.





An intent represents an action that the user wants to perform.

Name	Description	Last edited
TransferFunds	Help user transfer funds between bank accounts	1 minute ago
FollowupCheckBalance	Intent to allow a follow-up balance check request without authentication	1 minute ago
CheckBalance	Intent to check the balance in the specified account type	1 minute ago
Welcome	Welcome intent	1 minute ago
FallbackIntent	Default fallback intent when no other intent matches	1 minute ago

 AJ

ajv.singh3107@gmail.com

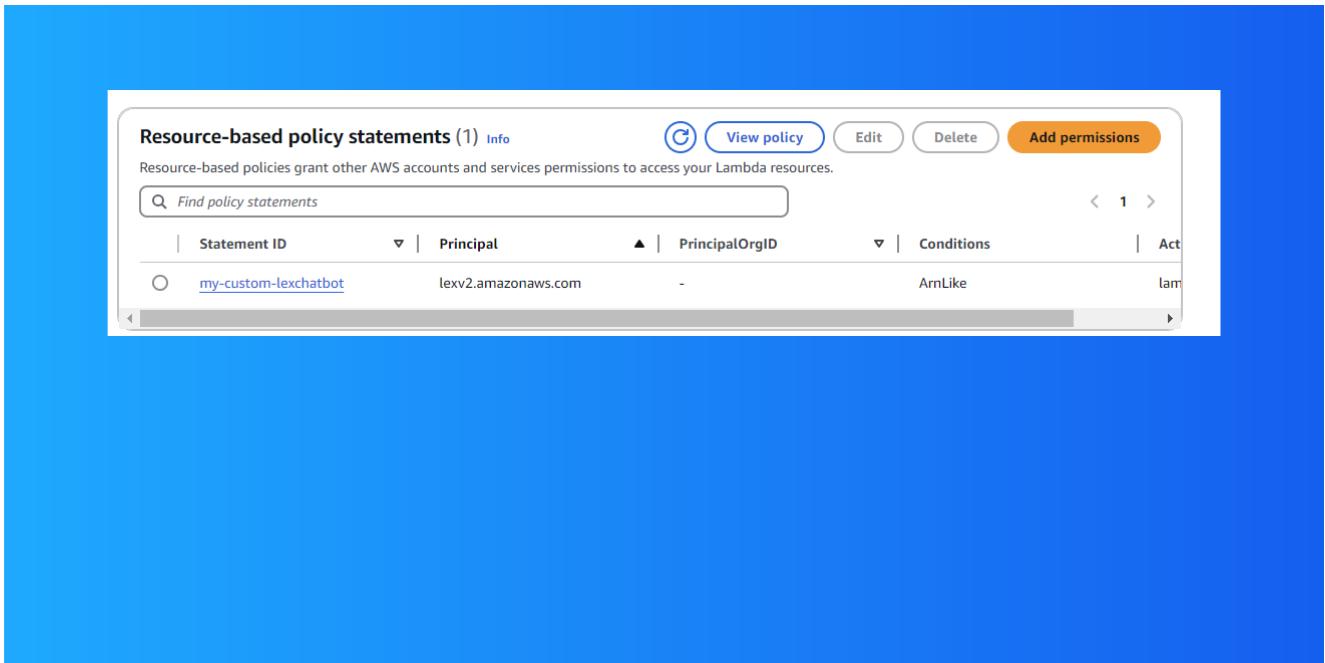
NextWork Student

NextWork.org

The final result!

Re-building my bot with CloudFormation took me around 10 minutes. This included writing the templates, deploying resources, and verifying the correct setup for services like Lambda and Lex, making the process efficient and repeatable for future use.

There was an error after I deployed my bot! The error was that the Lambda function was not attached to the intent, causing the bot to not respond correctly. I fixed this by linking the Lambda function to the appropriate intent and redeploying.



The screenshot shows the AWS Lambda Resource-based policy statements interface. At the top, there is a header with the title "Resource-based policy statements (1) Info", a "View policy" button, an "Edit" button, a "Delete" button, and an "Add permissions" button. Below the header, a message states: "Resource-based policies grant other AWS accounts and services permissions to access your Lambda resources." A search bar labeled "Find policy statements" is present. The main area is a table with the following data:

Statement ID	Principal	PrincipalOrgID	Conditions	Action
my-custom-lexchatbot	lexv2.amazonaws.com	-	ArnLike	lambda:InvokeFunction



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

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

Check out nextwork.org for
more projects

