



nextwork.org

Build a Chatbot with Multiple Slots



Nchindo Boris

The screenshot shows the AWS Management Console interface for Amazon Lex. The top navigation bar includes tabs for 'aws', 'Search', and 'AWS Lambda'. The main content area displays a confirmation message: 'Successfully built language English (US) in bot: BankerBot'. Below this, there are sections for 'Variations - optional' (with a 'More response options' button), 'Set values' (with 'Next step in conversation' and 'End conversation' buttons), and 'Code hooks - optional' (with a checkbox for 'Use a Lambda function for initialisation and validation'). On the right side, a 'Test Draft version' window is open, showing a conversation transcript. The user asks '4000' and the bot responds with 'Got it. So we are transferring 4000 from Checking to Savings. Can I go ahead with the transfer?'. The user replies 'yes' and the bot confirms 'The transfer is complete. 4000 should now be available in your Savings account.' At the bottom right of the test window, there is a 'Ready for complete testing' button and a text input field with placeholder 'Type a message'. The bottom of the screen shows the Windows taskbar with various pinned icons and system status information.

Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for building chatbots and voice assistants using natural language understanding and speech recognition. It's useful because it simplifies bot development, handles scaling, integrates with AWS, and allows seamless interactions with users in real-time.

How I used Amazon Lex in this project

In this project, I used Amazon Lex to build or create an intent that lets me transfer money between accounts and learn about AWS CloudFormation

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

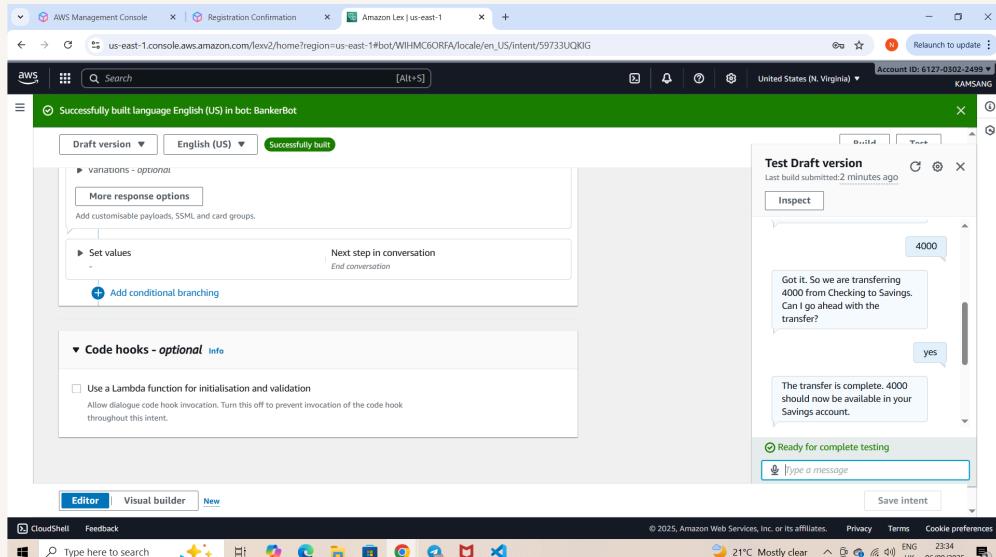
One thing I didn't expect in this project was how intuitive the visual builder in Amazon Lex was. It made designing and testing the conversation flow much simpler, allowing me to create and adjust the bot's logic without writing complex code.

This project took me...

The project took me about a 2 hours to complete. Setting up the intents, integrating Lambda, and testing the bot took the most time, but the visual builder and pre-configured tools in Amazon Lex helped speed up the process significantly.

TransferFunds

An intent I created for my chatbot was TransferFunds, which helps users transfer funds between bank accounts. The bot prompts for details like the amount and destination account, then processes the request and confirms the successful transfer.

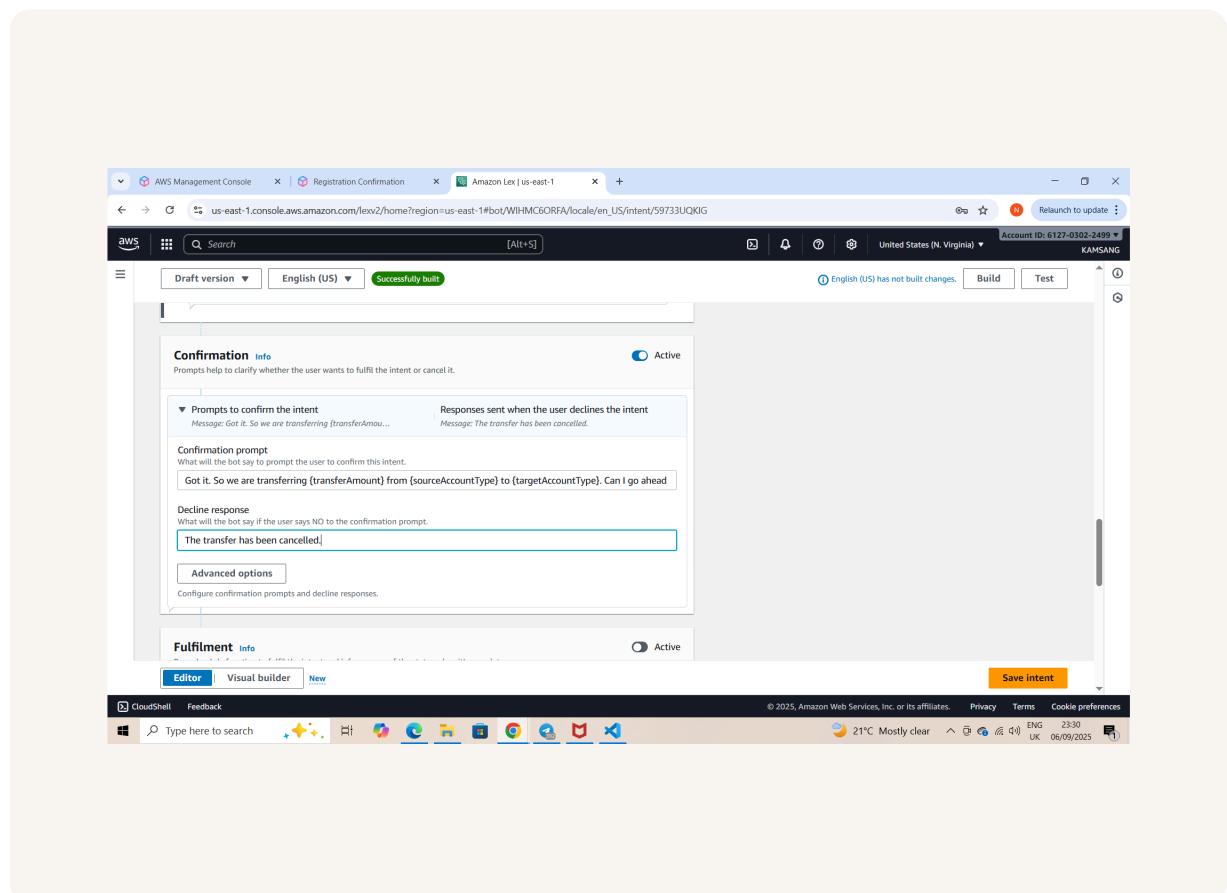




Using multiple slots

For this intent, I had to use the same slot type twice. This is because I needed to collect the same kind of information more than once, like two different account numbers or amounts, to process the user's request properly.

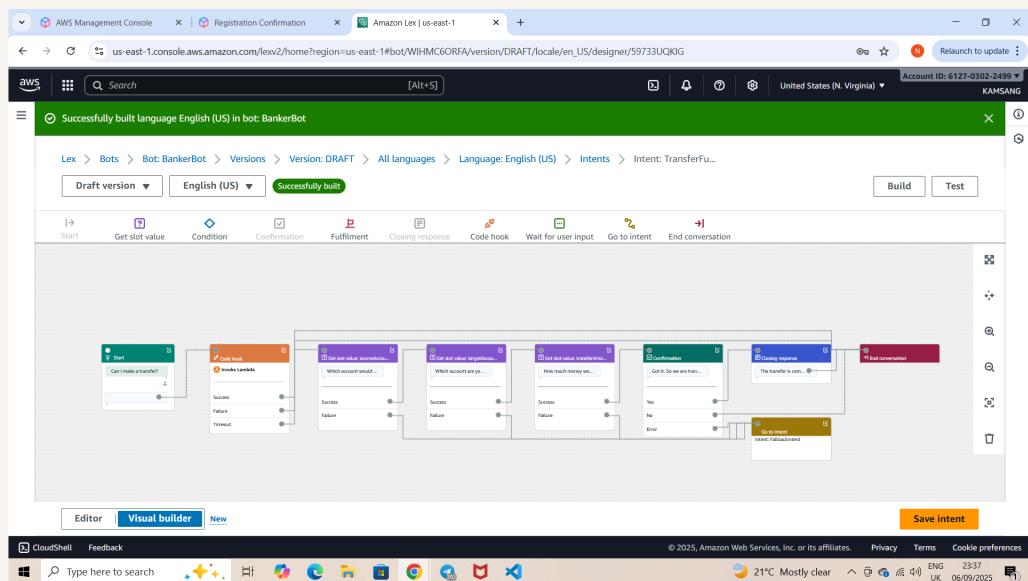
I also learned how to create confirmation prompts, which are messages that ask users to confirm their input before proceeding. They help ensure accuracy by allowing users to review and verify details, reducing errors in actions like transactions or settings changes.



Exploring Lex features

Lex also has a conversation flow feature that lets you control how the interaction progresses. It helps guide through different steps, making sure the conversation stays on track by managing intents and slots based on what the user says.

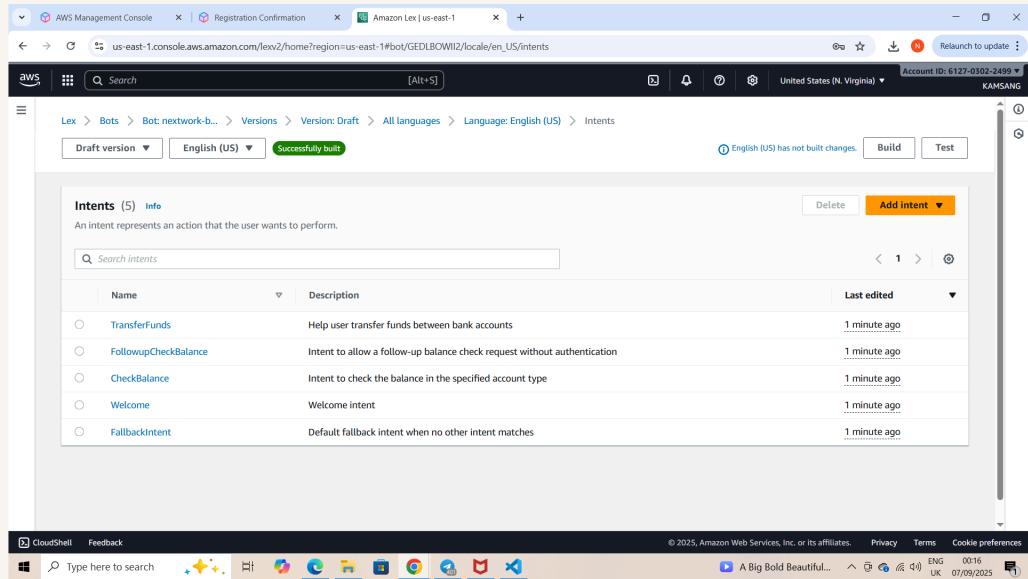
You could also set up your intent using a visual builder! A visual builder lets you design and organize your chatbot's conversation flow with a drag-and-drop interface, making it easier to create and manage intents, slots, and responses.



AWS CloudFormation

AWS CloudFormation is a service that lets you set up and manage AWS resources using code. It simplifies creating and configuring things like EC2 instances or S3 buckets, making it easier to automate and replicate infrastructure across different environments.

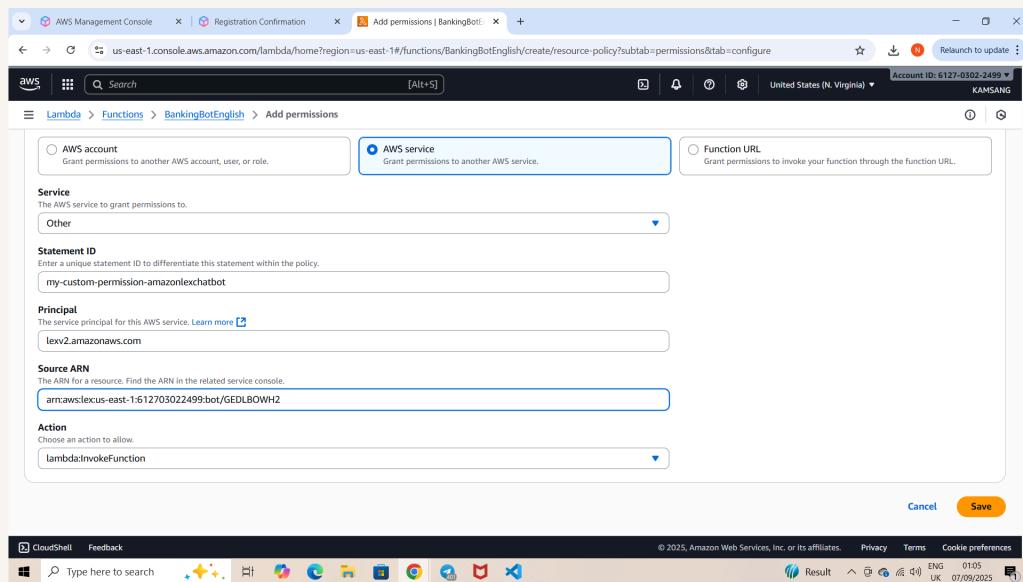
I used CloudFormation to automate the setup of AWS resources needed for creating the bot, Amazon Lex and Lambda configurations. This made it easier to deploy and manage the bot's infrastructure consistently without doing everything manually.



The final result!

Re-building my bot with CloudFormation took me 10 minutes which is very much quicker than setting up manually

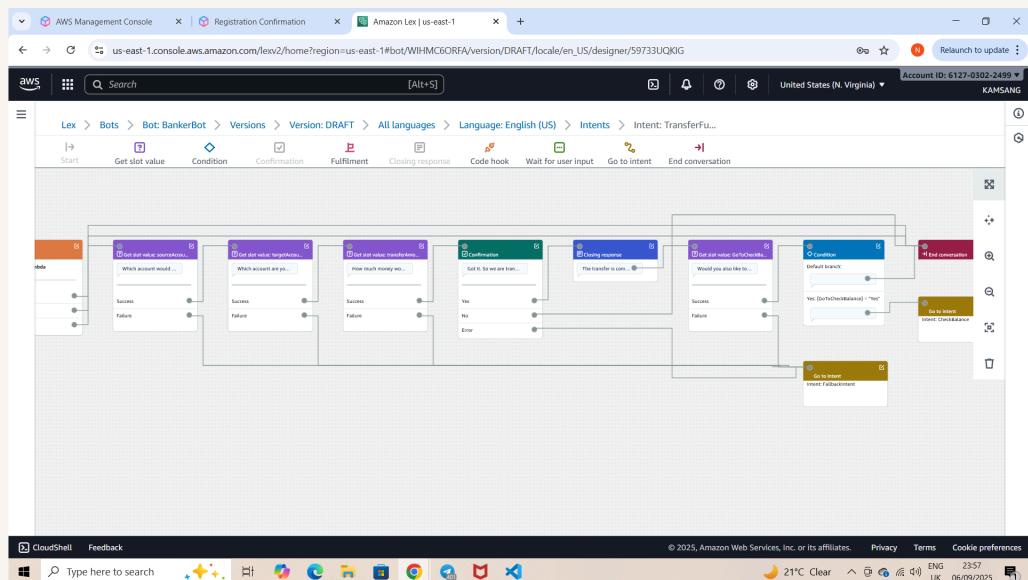
There was an error after I deployed my bot! The error was because CheckBalance intent wasn't working. I fixed this by reviewing the Lambda function's configuration and ensuring the correct permissions and references were set.



Using the visual builder

Using the visual builder, I added Get slot value card, Condition card, Go to intent card. What this means for an end user is the possibility to check account balance or not

The new Get slot value card will trigger when transfer is completed. This card will try to capture success or failure of the transfer. It also checks whether the user wants to check account balance.





nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

