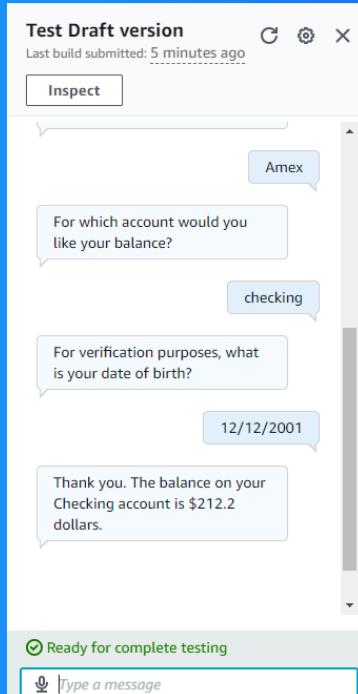




Connect a Chatbot with Lambda



Emmanuel Enalpe III





Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for building conversational interfaces using voice and text. It powers chatbots and virtual assistants with automatic speech recognition and natural language understanding, making it easier to automate customer interactions.

How I used Amazon Lex in this project

I used Amazon Lex in today's project to set up a chatbot with various intents and custom slots to check account balances and transfer money between accounts. I also connected the check balance intent to a Lambda function.

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

The only thing I didn't expect from this project is how easy it is to set up a chatbot with Amazon Lex due to its pre-built features that can meet your needs depending on how you use them.

This project took me...

Completing this project took me about 40–50 minutes to set up and configure the options for the chatbot I created.



AWS Lambda Functions

AWS Lambda is a serverless service that automatically runs your code in response to events without managing servers. It scales as needed, handling from a few requests to thousands per second. You simply provide your code in supported languages.

In this project, I created a Lambda function to help my chatbot give users quick answers about their account balances.

The screenshot shows the AWS Lambda Function Editor interface. The title bar says "lambda_function". The menu bar includes "File", "Edit", "Find", "View", "Go", "Tools", "Window", "Test", and "Deploy". A toolbar with "Upload from" and other icons is at the top right. The main area has tabs for "BankingBotEnglish" and "lambda_function". The "lambda_function" tab is active, showing the following Python code:

```
1 import json
2 import random
3 import decimal
4
5 def random_num():
6     return decimal.Decimal(random.randrange(1000, 50000))/100
7
8 def get_slots(intent_request):
9     return intent_request['sessionState']['intent']['slots']
10
11 def get_slot(intent_request, slotName):
12     slots = get_slots(intent_request)
13     if slotName in slots and slots[slotName] is not None:
14         return slots[slotName]['value']['interpretedValue']
15     else:
16         return None
```

At the bottom right, it says "39:74 Python Spaces: 4".



Chatbot Alias

In Amazon Lex, an alias acts as a pointer to a specific bot version. When integrating Lex with other AWS services or custom applications, external resources connect to the alias, which directs them to the desired bot version for use.

TestBotAlias is the default version of your bot used for testing and development. It serves as the sandbox version where you can verify that everything functions correctly before implementing changes in the live environment.

To connect Lambda with my BankerBot, I visited my bot's TestBotAlias and clicked the English language under the Languages panel. After clicking it, I selected BankingBotEnglish as the source, leaving the lambda function version as is.

Alias language support: English (US)

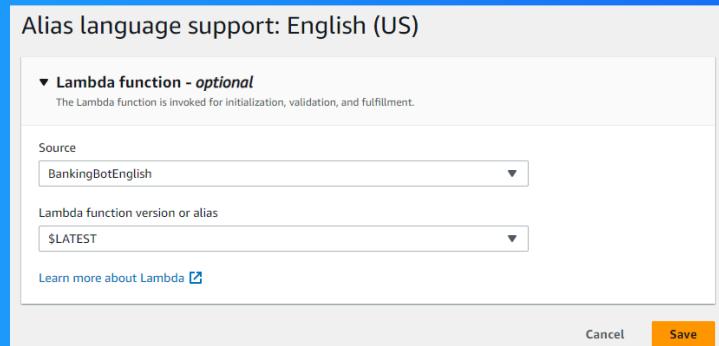
▼ Lambda function - optional
The Lambda function is invoked for initialization, validation, and fulfillment.

Source: BankingBotEnglish

Lambda function version or alias: \$LATEST

[Learn more about Lambda](#)

Cancel **Save**



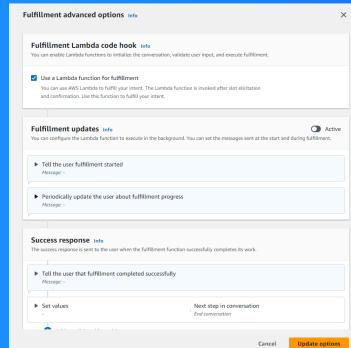


Code Hooks

Code hooks let you link your chatbot to custom Lambda functions for specific tasks during a conversation. They handle complex actions like querying databases or making decisions based on past interactions, extending the chatbot's basic capabilities.

Even though I already connected my Lambda function with my chatbot's alias, I had to use code hooks because code hooks make your chatbot smarter and more useful by allowing it to perform these extra steps seamlessly during chats.

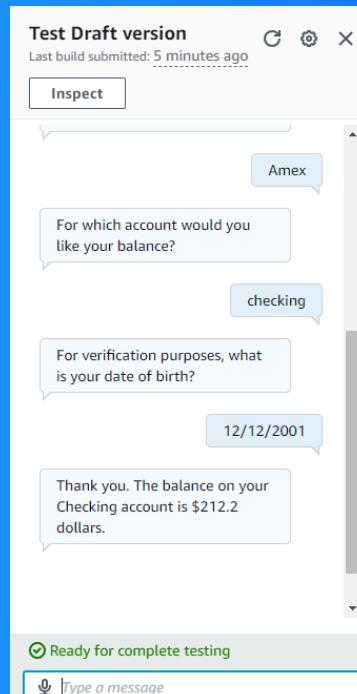
I could find code hooks in your chatbot's configuration settings. They're located within the dialog management or fulfillment sections, where you define triggers for specific intents and customize the chatbot's behavior using Lambda functions.





The final result!

I've set up my chatbot to trigger a Lambda function that returns a random dollar figure when a user inputs a birthdate for verification. After the input, the Lambda function is triggered.





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

