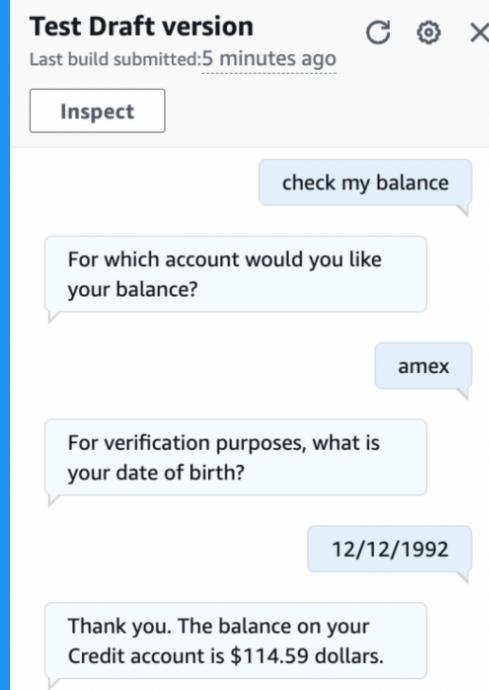




Connect a Chatbot with Lambda



khushinewgrad@gmail.com



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for building conversational interfaces using voice and text. It enables developers to create chatbots that can handle user interactions naturally, powered by automatic speech recognition and natural language understanding!

How I used Amazon Lex in this project

Today, I defined an AWS Lambda function that gets a user's balance, connected my AWS Lambda function with my Amazon Lex chatbot and connected my CheckBalance intent with my Lambda function.

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

The role of coding in connecting chatbot services

This project took me...

It took about 35 mins to complete this project!

AWS Lambda Functions

AWS Lambda is a service that lets you run code without provisioning or managing servers.

In this project, I created a Lambda function to get a user's balance, connect my AWS Lambda function with my Amazon Lex chatbot and connect my CheckBalance intent with my Lambda function.

The screenshot shows the AWS Lambda console interface. On the left, the Lambda function 'BANKINGBOTENGLISH' is selected. The main area displays the code for 'lambda_function.py':

```
1  """
2  How does AWS Lambda cheer up Amazon Lex? By saying, "Don't worry, I've got
3  you covered!" This Lambda function generates a random number between 1000 and 50000
4  - NextWork :)
5  """
6  import json
7  import random
8  import decimal
9
10 def random_num():
11     return decimal.Decimal(random.randrange(1000, 50000))/100
12
13 def get_slots(intent_request):
14     return intent_request['sessionState']['intent']['slots']
15
16 def get_slot(intent_request, slotName):
17     slots = get_slots(intent_request)
18     if slotName is not None and slotName in slots and slots[slotName] is not None:
19         return slots[slotName]['value']['interpretedValue']
20     else:
21         return None
22
23 def get_session_attributes(intent_request):
24     sessionState = intent_request['sessionState']
25     if 'sessionAttributes' in sessionState:
26         return sessionState['sessionAttributes']
27
28     return {}
29
30 def elicit_intent(intent="AMAZON.CancelIntent", message=""):
31     return {
32         "dialogAction": {
33             "type": "ElicitIntent"
34         }
35     }
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
```

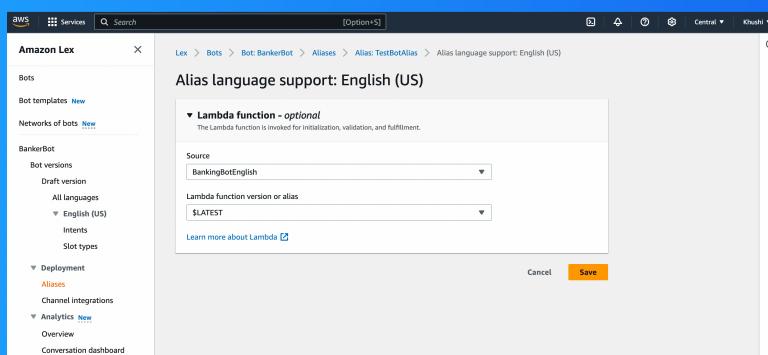
The right side of the interface features a 'Tutorials' sidebar titled 'Create a simple web app'. It includes a brief description and a 'Start tutorial' button.

Chatbot Alias

An alias in Amazon Lex is a pointer for a specific version of my bot. So when I connect Lex with other AWS services or my custom applications, those external resources will connect to an alias, which will point to the specific version of my bot.

TestBotAlias is a default version of my bot that's made for testing or development.

To connect Lambda with my BankerBot, I visited my bot's TestBotAlias and chose the language as English followed by choosing BankingBotEnglish as the source and \$LATEST as the lambda function version or alias.

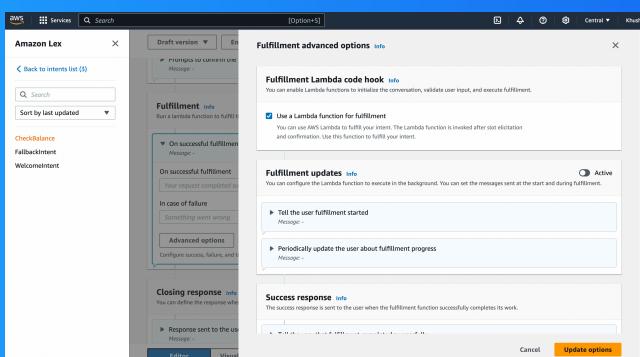


Code Hooks

A Code hook helps in connecting my chatbot to custom Lambda functions for doing specific tasks during a conversation.

Even though I already connected my Lambda function with my chatbot's alias, I had to use code hooks because they're used to handle more complex actions that the basic chatbot setup can't do on its own, like checking data from a database.

I could find code hooks at advanced option in Fulfilment called "Fulfilment Lambda Code Hook" panel.



KH

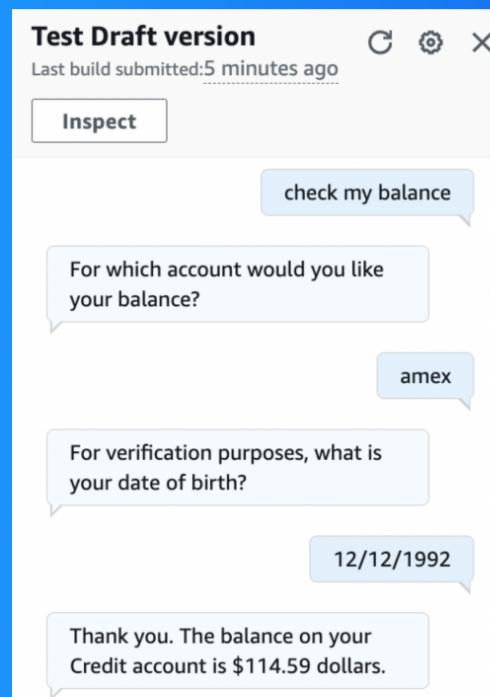
khushinewgrad@gmail.c...

NextWork Student

NextWork.org

The final result!

I've set up my chatbot to trigger Lambda and return a random dollar figure when a date of birth is added by the user.





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

