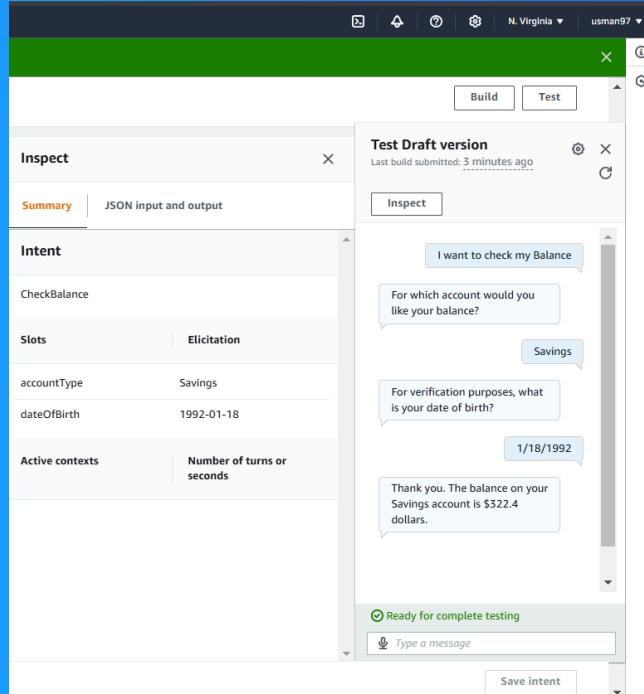




Connect a Chatbot with Lambda



mumer@hawk.iit.edu



Introducing Today's Project!

What is Amazon Lex?

Amazon Lex is a service for building interfaces using voice and text. It's useful because it leverages NLP techniques to create simple chatbots that understand and respond to basic queries for new users navigating the app or website.

How I used Amazon Lex in this project

I used Amazon Lex to build a simple chatbot to allow user to check their balance by talking to the chatbot

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

I didn't expect this tool to be so easy and simple to use, it is quite powerful and can make really good chatbots for company websites

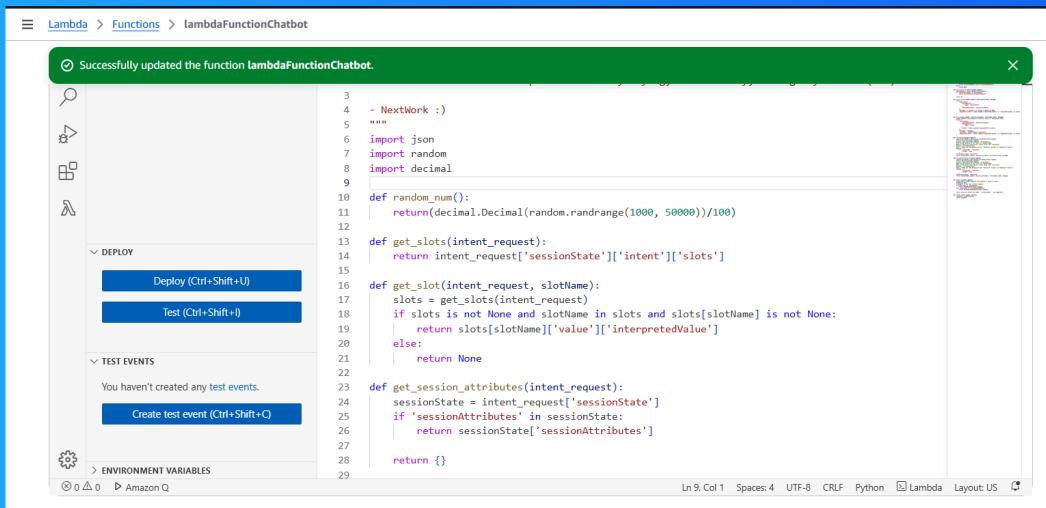
This project took me...

This project took me 30 mins to finish

AWS Lambda Functions

AWS Lambda is a service that lets you run code in the cloud without needing to manage any servers as Lambda takes care of it.

I created a Lambda function to generate a random number to get balance for the user when the prompt is I want to check my balance using the chatbot



The screenshot shows the AWS Lambda Function Editor interface. At the top, a green success message says "Successfully updated the function lambdaFunctionChatbot.". On the left, there's a sidebar with icons for search, deployment, and test events. Under "DEPLOY", there are two buttons: "Deploy (Ctrl+Shift+U)" and "Test (Ctrl+Shift+I)". Under "TEST EVENTS", it says "You haven't created any test events." and has a "Create test event (Ctrl+Shift+C)" button. At the bottom, there's a section for "ENVIRONMENT VARIABLES" with a "Create environment variable" button. The main area contains the Python code for the Lambda function:

```
3
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 slots 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
29 return {}
```

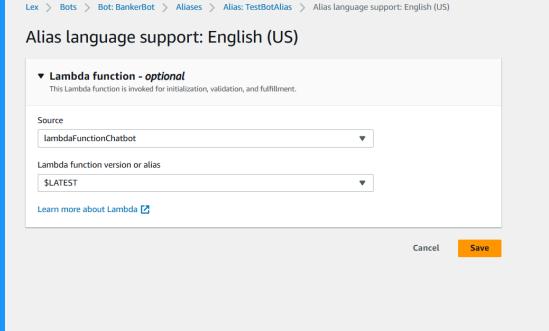
At the bottom right, there are status indicators: 0 errors, 0 warnings, and 0 info. It also shows "Amazon Q" and "Lambda". The bottom status bar includes "Ln 9, Col 1", "Spaces: 4", "UTF-8", "CRLF", "Python", "Lambda", "Layout: US", and a refresh icon.

Chatbot Alias

An Alias acts as a pointer for a specific version of your bot and connects to other services

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

To connect Lambda with my BankerBot, I visited my bot's TestBotAlias and selected English(US) from the languages panel and then added the source as the name of the lambda function and version was \$LATEST

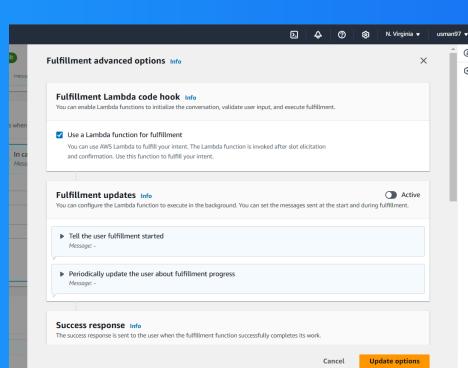


Code Hooks

Code hooks help you connect your chatbot to custom Lambda functions for doing specific tasks during a conversation.

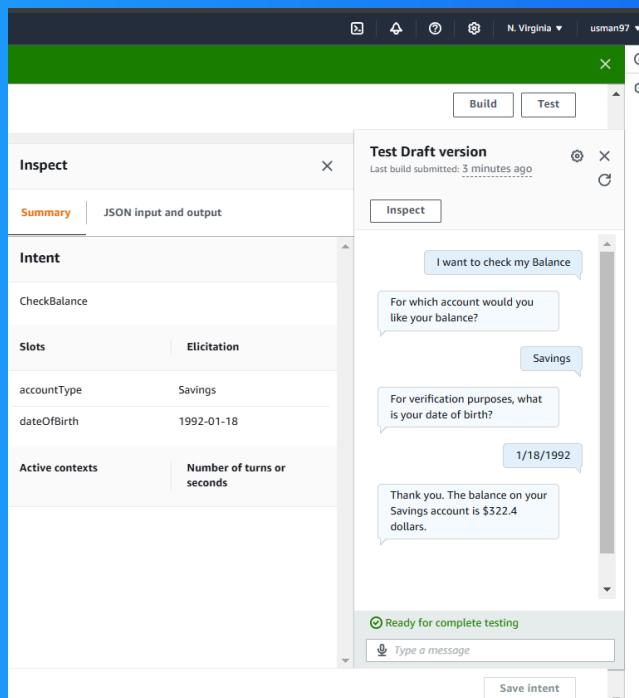
Even though I already connected my Lambda function with my chatbot's alias, I had to use code hooks because they allow me to add custom logic and validations during the interaction with the chatbot.

I could find code hooks at in the advanced options under the fulfillment panel



The final result!

I've set up my chatbot to trigger Lambda and return a random dollar figure when a successfull fulfillment occurs and chatbot has successfully got the type of account and the date of birth from user





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

