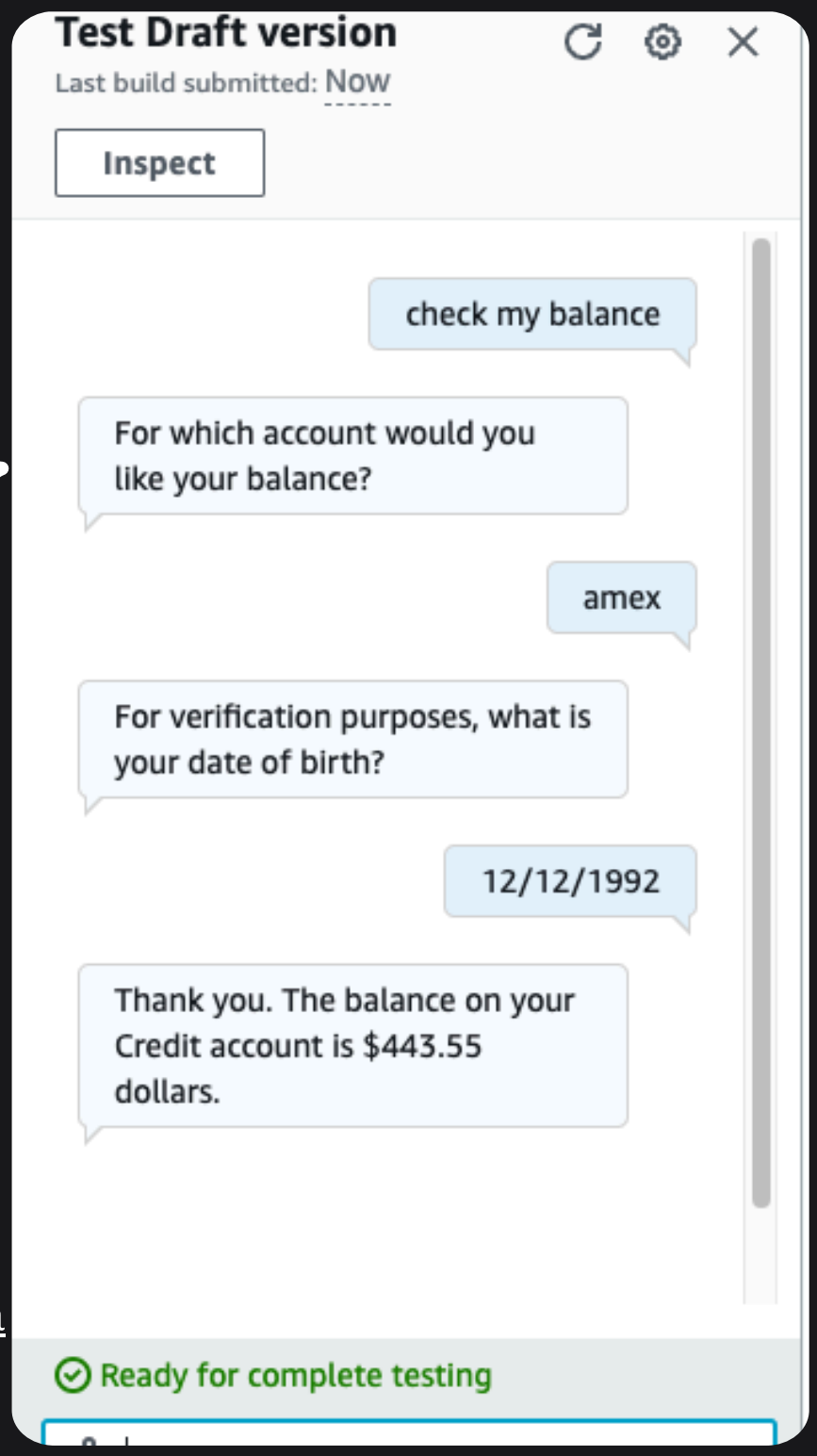
NEXTWORK

How I connected my chatbot with AWS Lambda!



Megha Naik

https://www.linkedin.com/in/naikmegha

What is Amazon Lex?

What it does:

- It helps you build Voice and Text Chatbots in minutes.

Why it's useful:

- It will save a lot of time for a business or a service and can be personalized

How I'm using it in today's project:

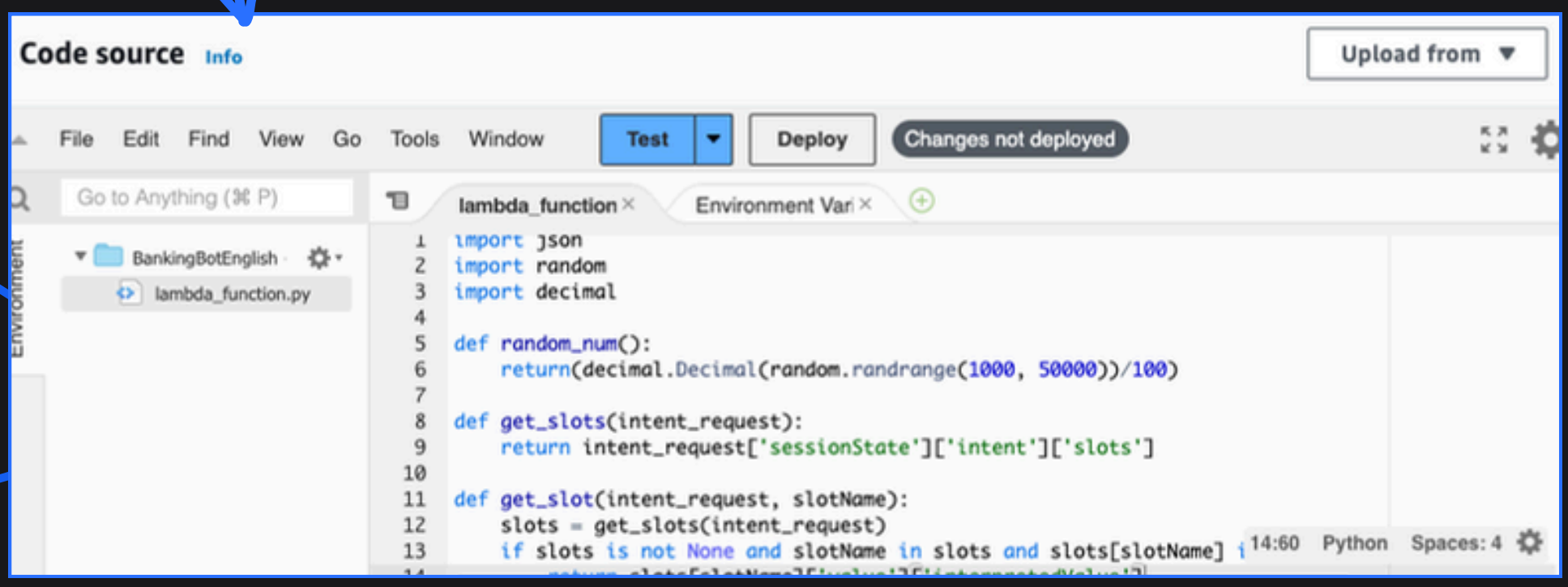
- In this project, I'm using Amazon Lex to create BankerBot. where I am going to connect Amazon Lex to AWS Lambda for end users to check their account balance.

Megha Naik
https://www.linkedin.com/in/naikmegha

Using AWS Lamba

- **AWS Lambda** is an AWS service that helps you run code without having to manage servers.
- In this project, a Lambda function was created to generate the user's bank balance. In this example, a random figure was generated, however, in the real world, the Lambda function can be used to extract the user's bank balance from the database. The Amazon Lex chatbot on its own would not be able to generate a bank balance. That's why this connection to AWS Lambda is crucial!

A peek into the Python code I uploaded into AWS Lambda!



Megha Naik

https://www.linkedin.com/in/naikmegha

Connecting Lambda with Lex

There were two steps to connecting the Lambda function with my chatbot:

Step 1

- To connect Lambda with my chatbot alias, I visited the Alias page of my chatbot and connected my TestBotAlias (my chatbot's default alias, made for development/testing) with the latest version of the AWS Lambda function defined.

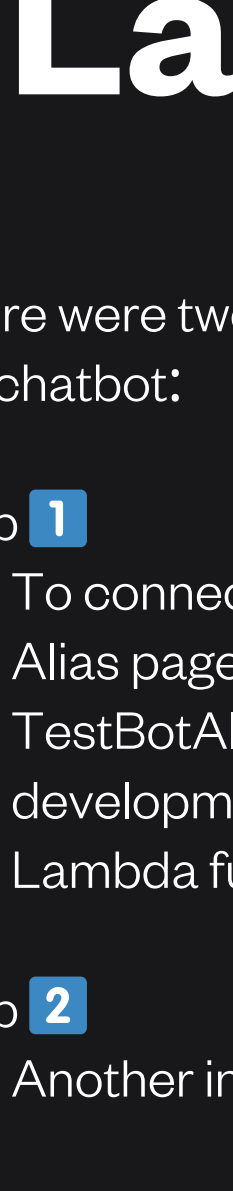
Step 2

- Another intent setting to configure is **code hooks**.

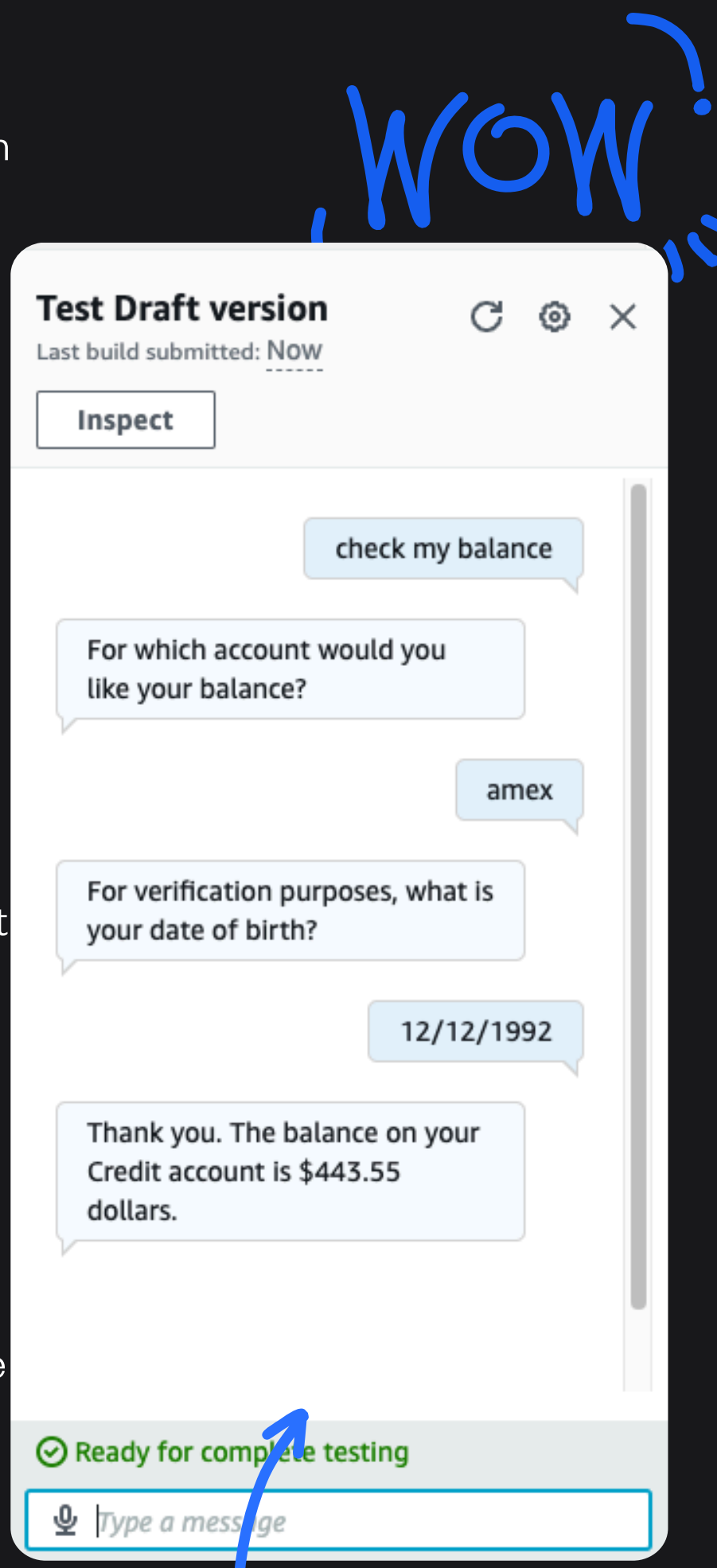
- A code hook is a piece of code that can be connected to my chatbot to perform functions/actions that my chatbot cannot do alone/by default

- In this project, I had to use code hooks because the chatbot is not able to calculate/return a bank balance figure on its own.

After connecting Lambda with my Lex bot, my chatbot could immediately start returning specific bank balance figures. The AWS Lambda function would generate a random number each time.



Megha Naik
https://www.linkedin.com/in/naikmegha



My chatbot now returns a bank balance number thanks to Lambda!

My Key Learnings

01

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

02

You connect Amazon Lex with AWS Lambda when you want to add custom logic or functionality to your chatbot

03

To connect Amazon Lex with AWS Lambda, I followed 2 steps:

- I linked my chatbot's "TestBotAlias" to the latest version of the AWS Lambda function on the Alias page.
- I configured code hooks, which are pieces of code connected to my chatbot to perform actions like calculating and returning a bank balance figure that the chatbot cannot do alone by default.

04

In this project, the Python script I used helps the chatbot quickly respond to users asking about their account balances. When someone asks about their balance, Lex tells Lambda to run the given Python code, which picks a random number to pretend it's the balance. Lambda then sends this number back to Lex, which shows it to the user through the chatbot.

Megha Naik

https://www.linkedin.com/in/naikmegha

Final thoughts...

- This project took me about 40 min. Writing documentation took me around 20 min.

- Delete EVERYTHING at the end! Let's keep this project free :)

- One thing I didn't expect was an "alias". I discovered the concept of using aliases as a middleman to connect the chatbot with the AWS Lambda function, which also improves security.

- In the next phase of this project, we're enhancing BankerBot's memory with context carryover! My BankerBot will remember key details like the user's birthday during a session for a smoother experience 🎂

Megha Naik
https://www.linkedin.com/in/naikmegha

Find this helpful?

Like this post

yes!

Leave a comment

Save for later

Let's connect!

pssst... if you want to get this free project guide and documentation template, **check out NextWork!**

Megha Naik

https://www.linkedin.com/in/naikmegha

Thanks NextWork for the free project guide!

NEXTWORK