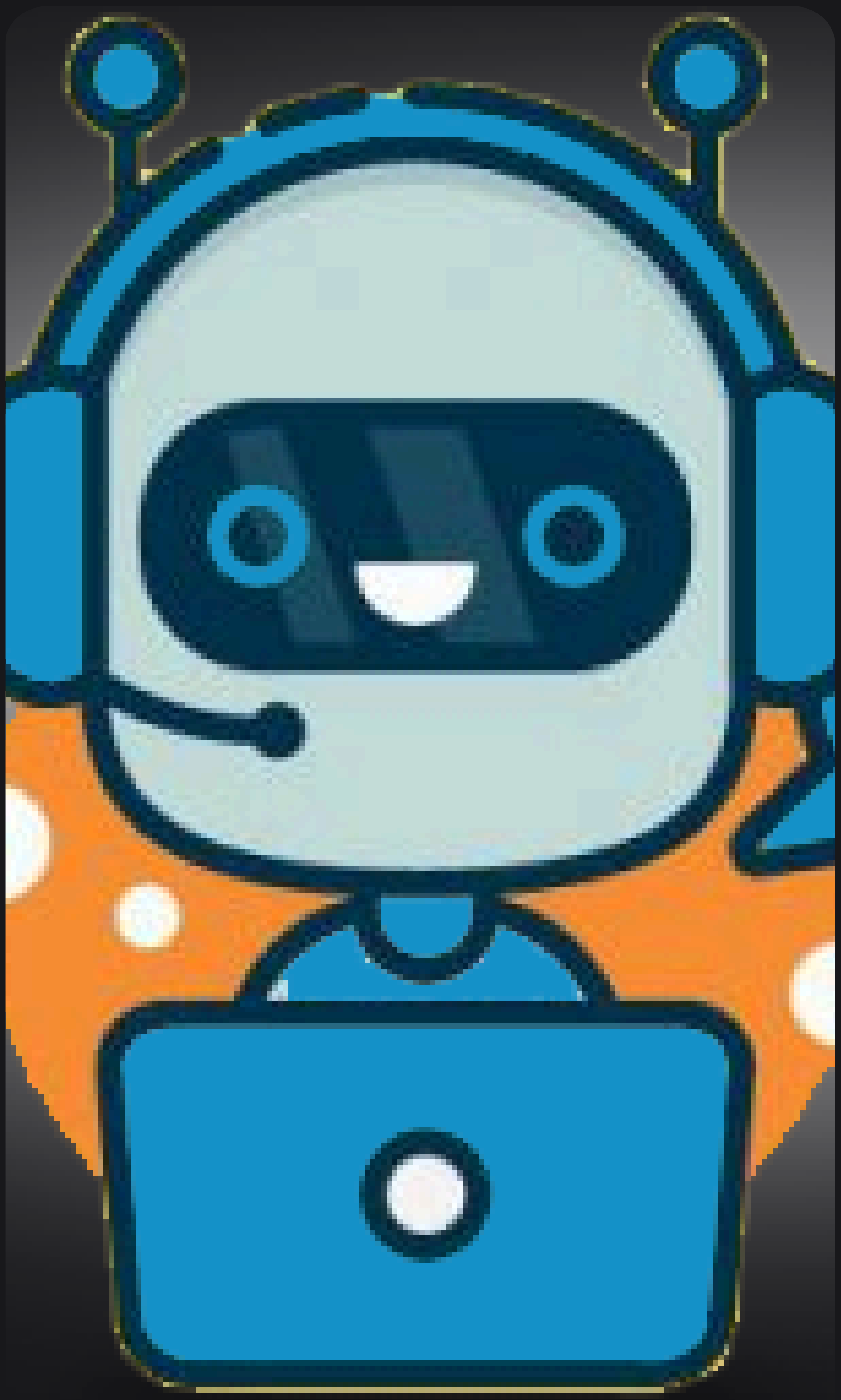
NEXTWORK

How I built a chatbot with Amazon Lex



Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

What is Amazon Lex?

What it does:

- Helps build chatbots, call center bots

Why it's useful:

- Automatic Speech Recognition to convert speech to text and Natural language understanding to recognize the intent of text, callers.

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

- In this project, I'm using Amazon Lex to create BankerBot, which can help your imaginary bank's customers check their account balance and transfer money between accounts!

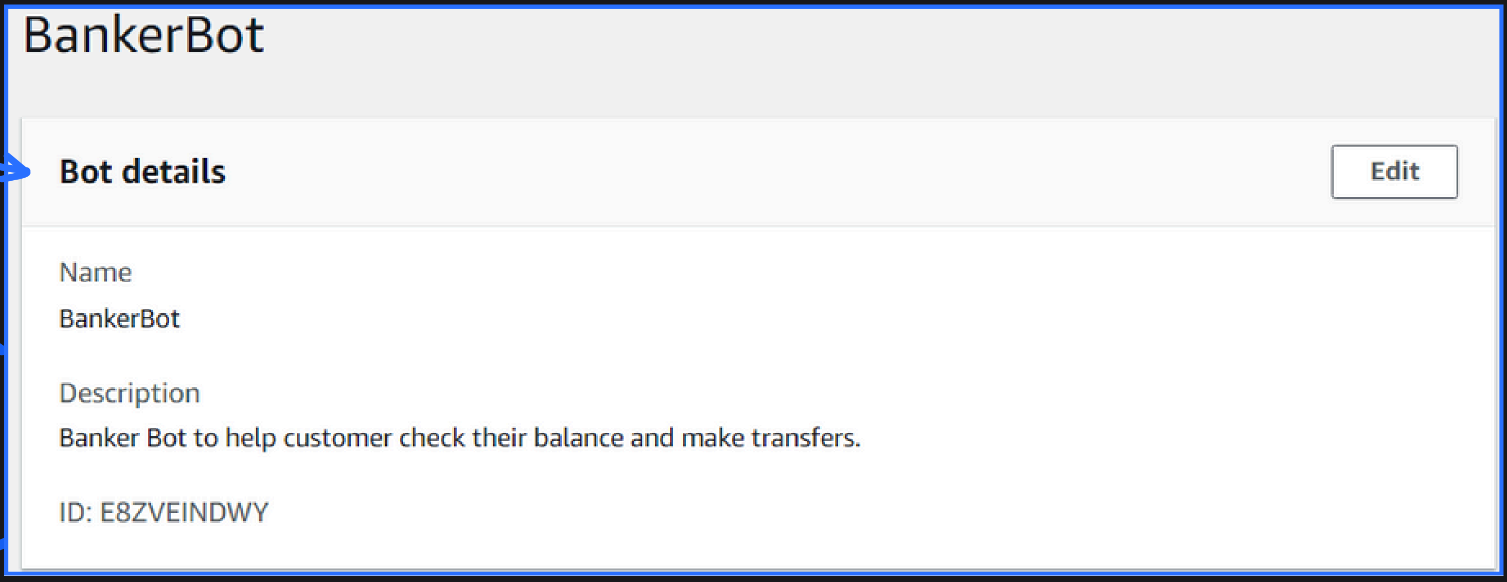
Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Set up a Lex chatbot

- I created BankerBot from scratch and used most default settings on Lex.

- In terms of the **intent classification confidence score,** I kept the default value of 0.40. What this means for my chatbot is it should at least be 40% confident about the intent/goal of the chatbox user to respond. In more technical terms, there should at least be a 40% match between the user's input and an intent I program for my Banker Bot to respond accordingly.

Setting up my Lex chatbot...



BankerBot

Bot details

Name
BankerBot

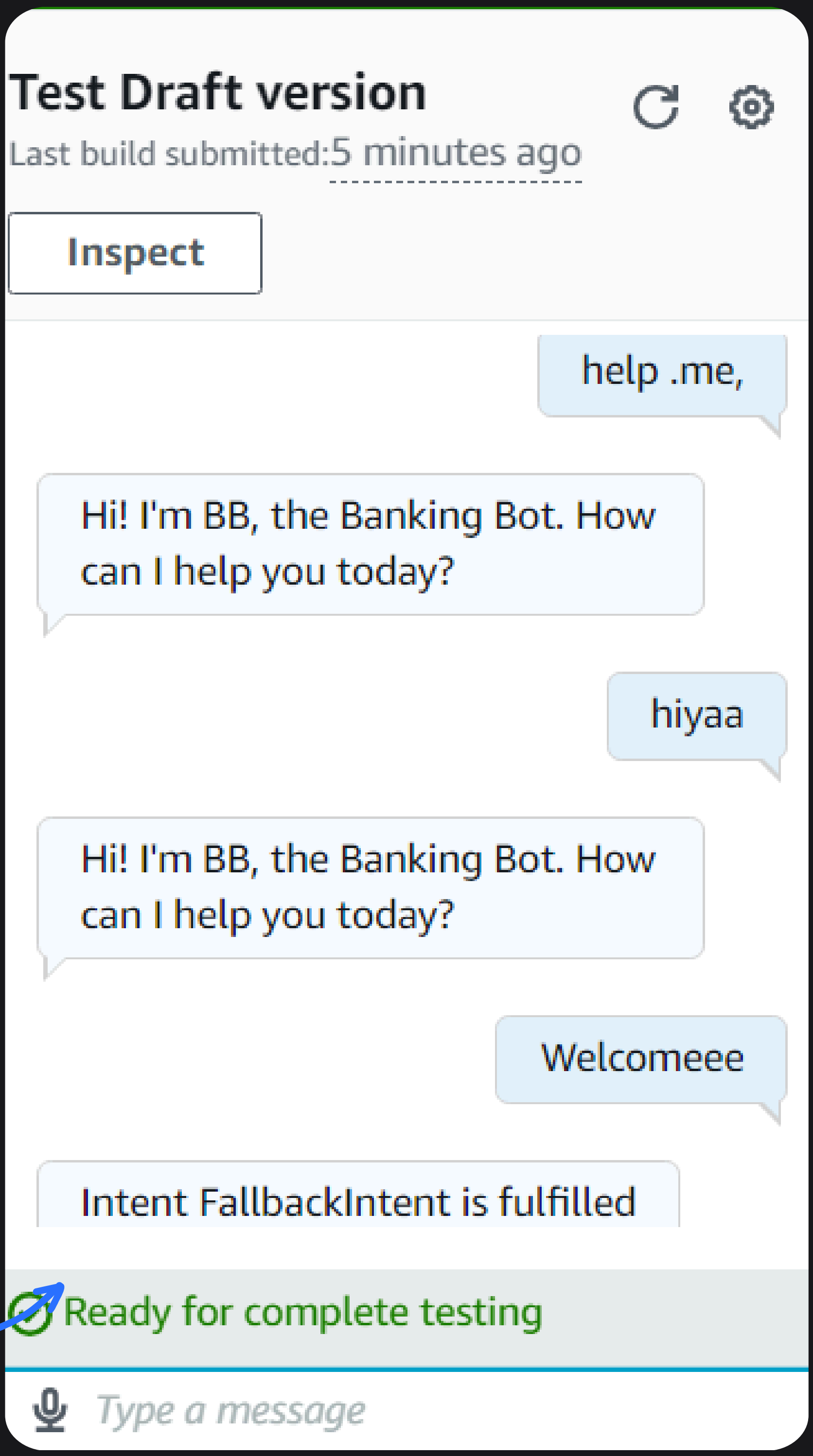
Description
Banker Bot to help customer check their balance and make transfers.

ID: E8ZVEINDWY

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Create an intent in Lex

- Intents represent the user's goals/purposes for using the chatbot. In Amazon Lex, a chatbot is defined by the intents that it supports.
- My first intent, WelcomeIntent, was created to greet the user when they say hello.
- To set up this intent, I created sample utterances(e.g. "Hi", "Hello", "I need help") and a closing response i.e. how the chatbot will respond.
- I launched and tested the chatbot, which could still respond if I enter similar utterances e.g."Hiyaa".
- However, the chatbot returned the error message "Intent FallbackIntent is fulfilled" when I entered "Welcomeee".
- This error message occured because my chatbot could not understand the intent of the phrase "welcomeee".


Test Draft version
Last build submitted:5 minutes ago

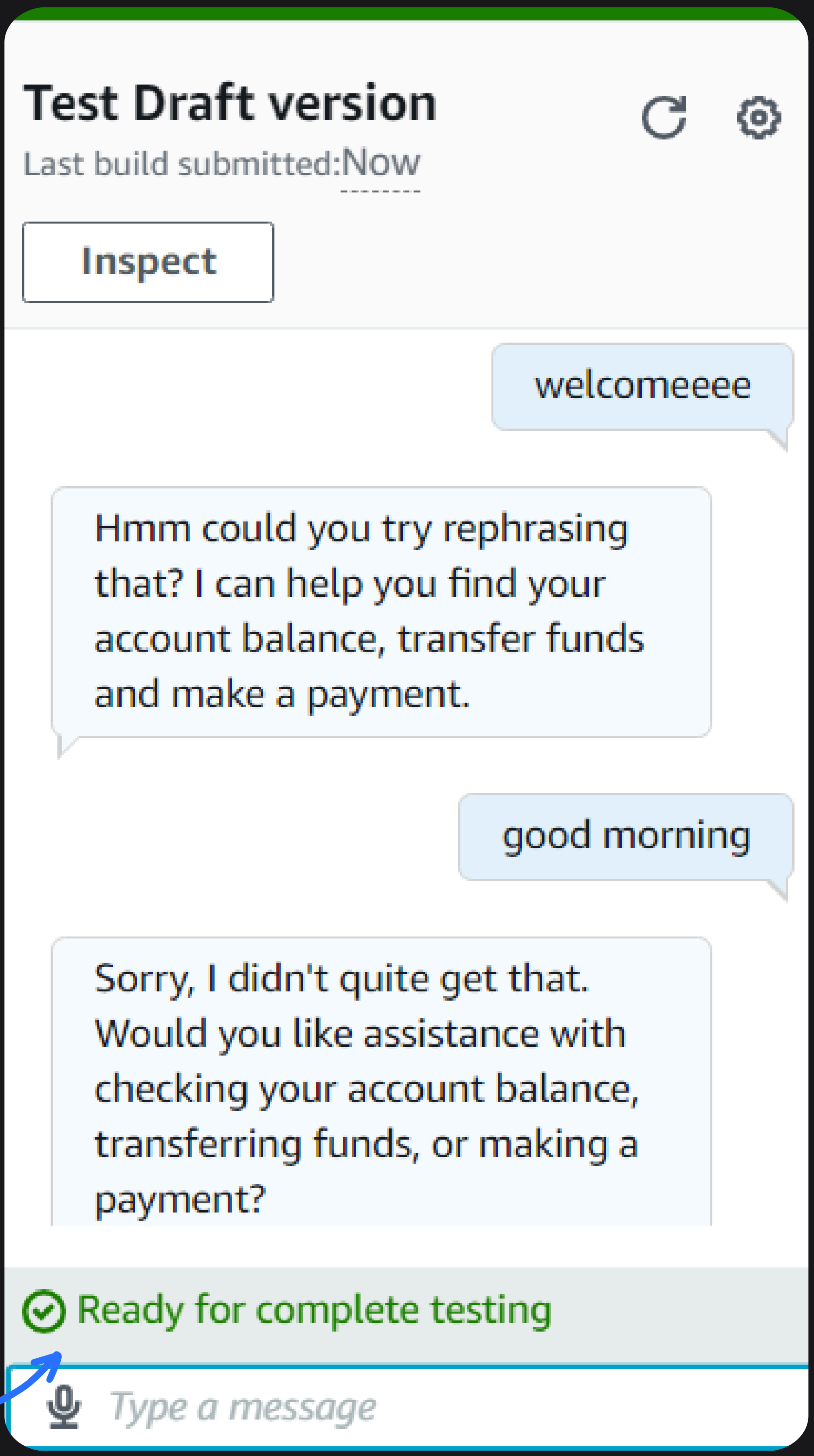
help .me,
Hi! I'm BB, the Banking Bot. How can I help you today?
hiyaa
Hi! I'm BB, the Banking Bot. How can I help you today?
Welcomeee
Intent FallbackIntent is fulfilled
Ready for complete testing

My first test of the chatbot

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Manage FallbackIntent

- FallbackIntent is a default intent in every chatbot that gets triggered when the chatbot does not recognize the user's goal/purpose.
- I wanted to configure FallbackIntent because the default closing response to the user is not easily understandable.
- To configure FallbackIntent, I had to create my own closing response in the intent's set up page. "Sorry I am having trouble understanding. Can you describe what you'd like to do in a few words? ..."
- I also added variations! What this means for an end user is they get to see different forms of my chatbot's closing response.



Test Draft version

Last build submitted:Now

Inspect

welcomeeee

Hmm could you try rephrasing that? I can help you find your account balance, transfer funds and make a payment.

good morning

Sorry, I didn't quite get that. Would you like assistance with checking your account balance, transferring funds, or making a payment?

Ready for complete testing

Type a message

Perfect! The error message is now much clearer, and there are variations too

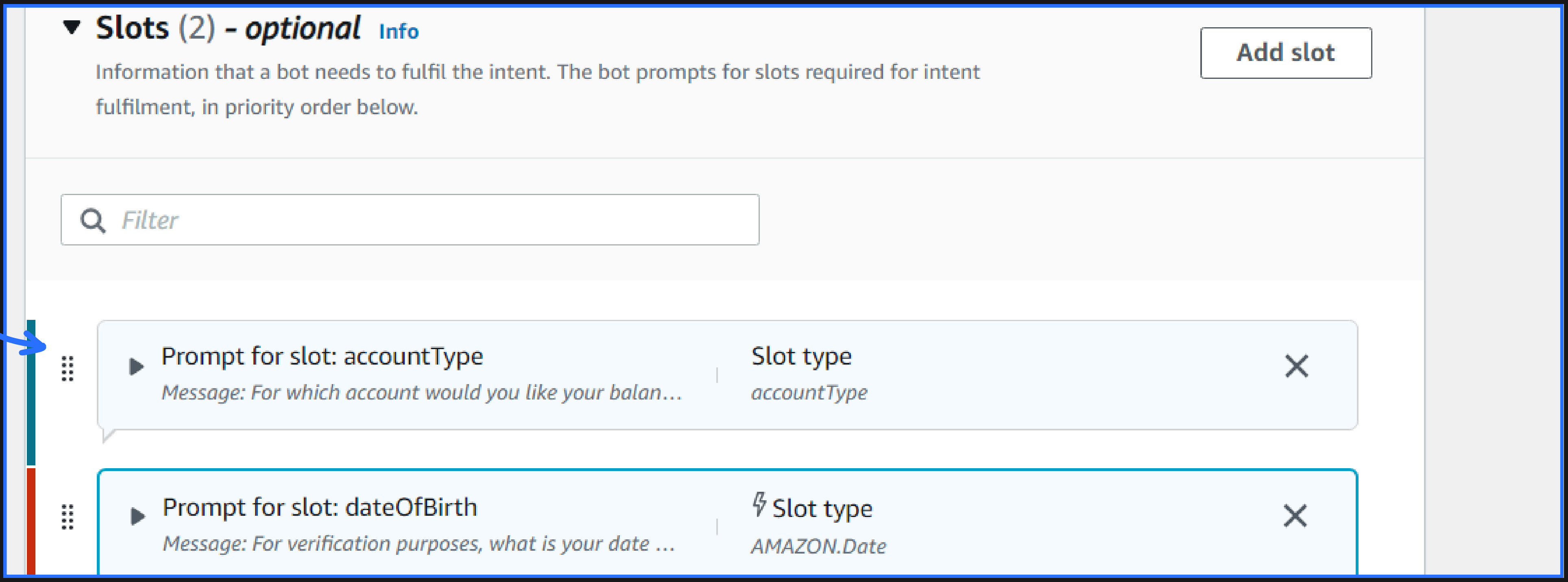
Vamsi Muppana

linkedin.com/in/vamsi-muppana-564373209

Create custom slots

- **Slots** are like placeholders / 'blanks' of information that my chatbox will seek to fill to fulfill an intent.
- In this project, I created a **custom slot type** to represent the different account types that a banking customer could have. A custom slot type was required as the default slot types did not accommodate for account types (e.g. Checking, Credit, Savings).
- I then associated the custom slot with a new intent, CheckBalance, which is a new intent I created that will help my bank's customers check its users account balances.

My custom slots



Information that a bot needs to fulfil the intent. The bot prompts for slots required for intent fulfilment, in priority order below.

Prompt for slot: accountType *Message: For which account would you like your balan…*	Slot type *accountType*
Prompt for slot: dateOfBirth *Message: For verification purposes, what is your date …*	Slot type *AMAZON.Date*

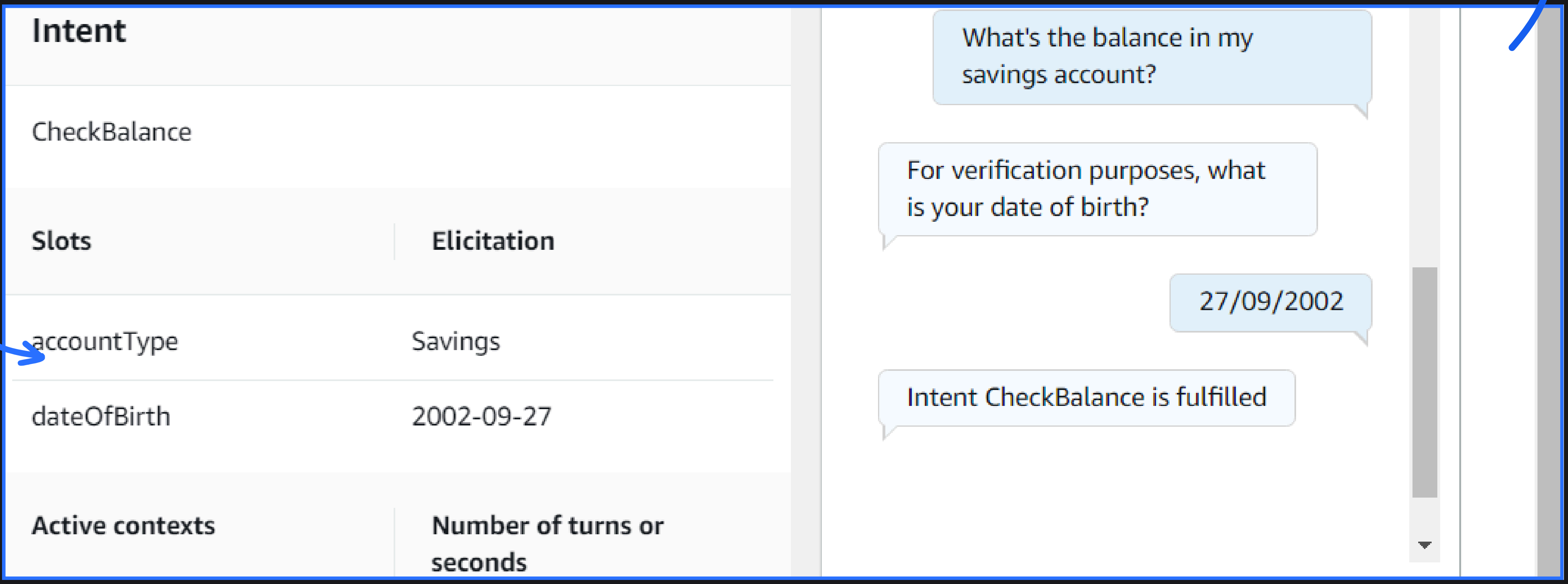
Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Simplifying the user experience

- I included slot values in some of the utterances (i.e. user inputs) for this intent too. For example, I defined the utterance **What's the balance in my {accountType} account?**
  - This is an example of an utterance that expects the slot accountType.

- Adding custom slots in the utterance enhances the user experience - the bot will automatically register which account type the user is trying to check, and will not ask for it again. This saves the user's time and makes the conversation much more efficient.

Slot values getting recognised in a conversation



Intent

CheckBalance

Slots	**Elicitation**
accountType	Savings
dateOfBirth	2002-09-27

Active contexts	**Number of turns or seconds**

What's the balance in my savings account?

For verification purposes, what is your date of birth?

27/09/2002

Intent CheckBalance is fulfilled

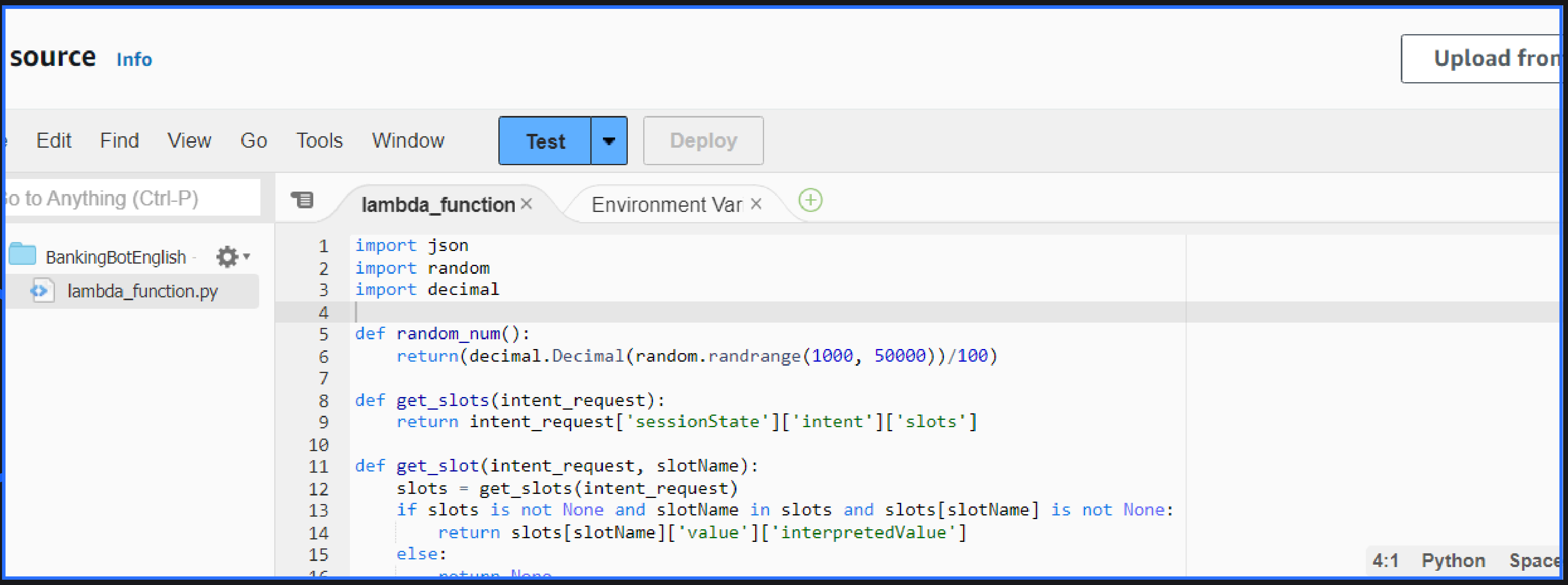
Vamsi Muppana

linkedin.com/in/vamsi-muppana-564373209

Using AWS Lamba

- AWS Lambda is an AWS service that helps you run code without managing 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 a 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!



```python
import json
import random
import decimal

def random_num():
    return(decimal.Decimal(random.randrange(1000, 50000))/100)

def get_slots(intent_request):
    return intent_request['sessionState']['intent']['slots']

def get_slot(intent_request, slotName):
    slots = get_slots(intent_request)
    if slots is not None and slotName in slots and slots[slotName] is not None:
        return slots[slotName]['value']['interpretedValue']
    else:
```

Vamsi Muppana

linkedin.com/in/vamsi-muppana-564373209

Connecting Lambda with Lex

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

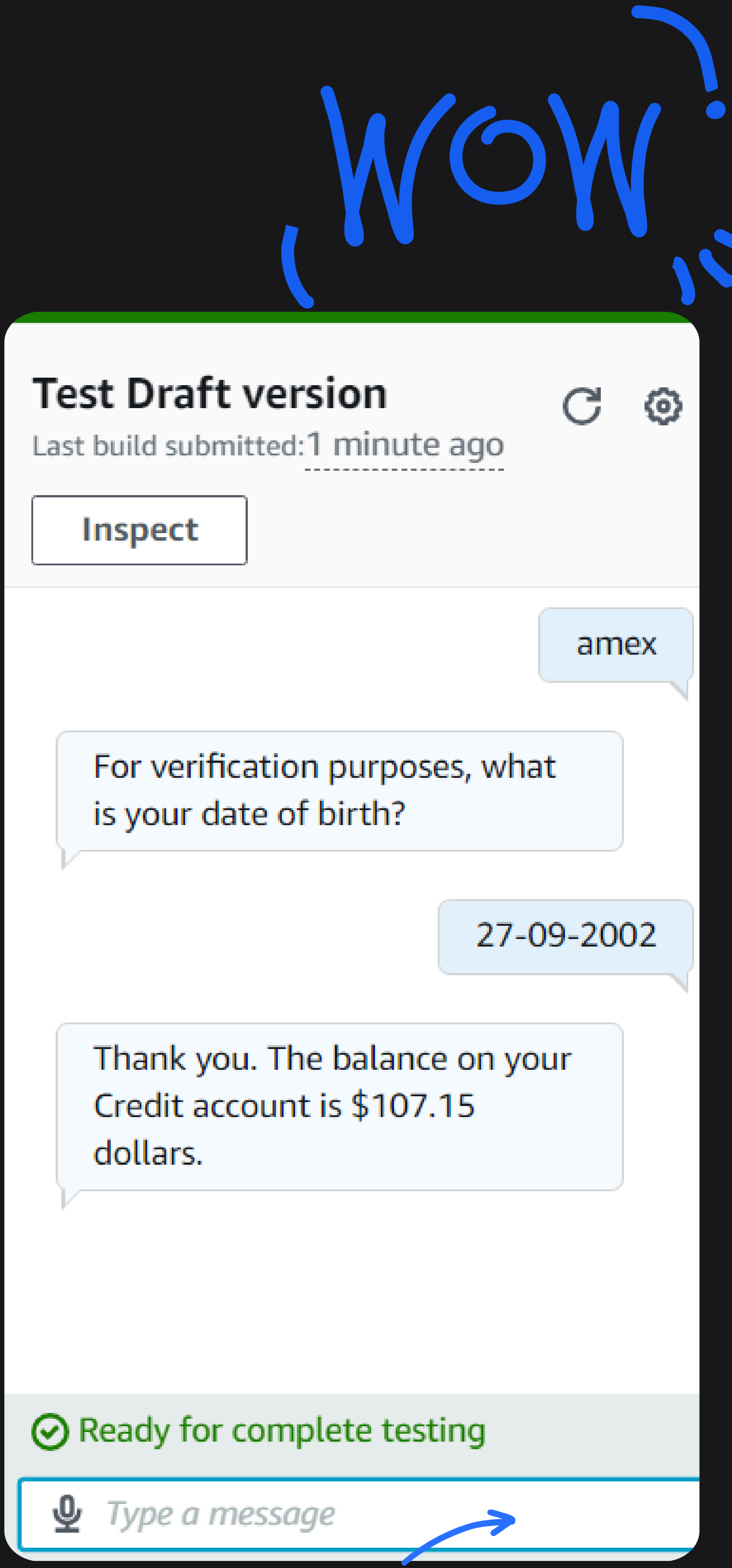
Step 1

- 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.


WOW
Test Draft version
Last build submitted: 1 minute ago
Inspect
Ready for complete testing
Type a message

amex

For verification purposes, what is your date of birth?

27-09-2002

Thank you. The balance on your Credit account is $107.15 dollars.

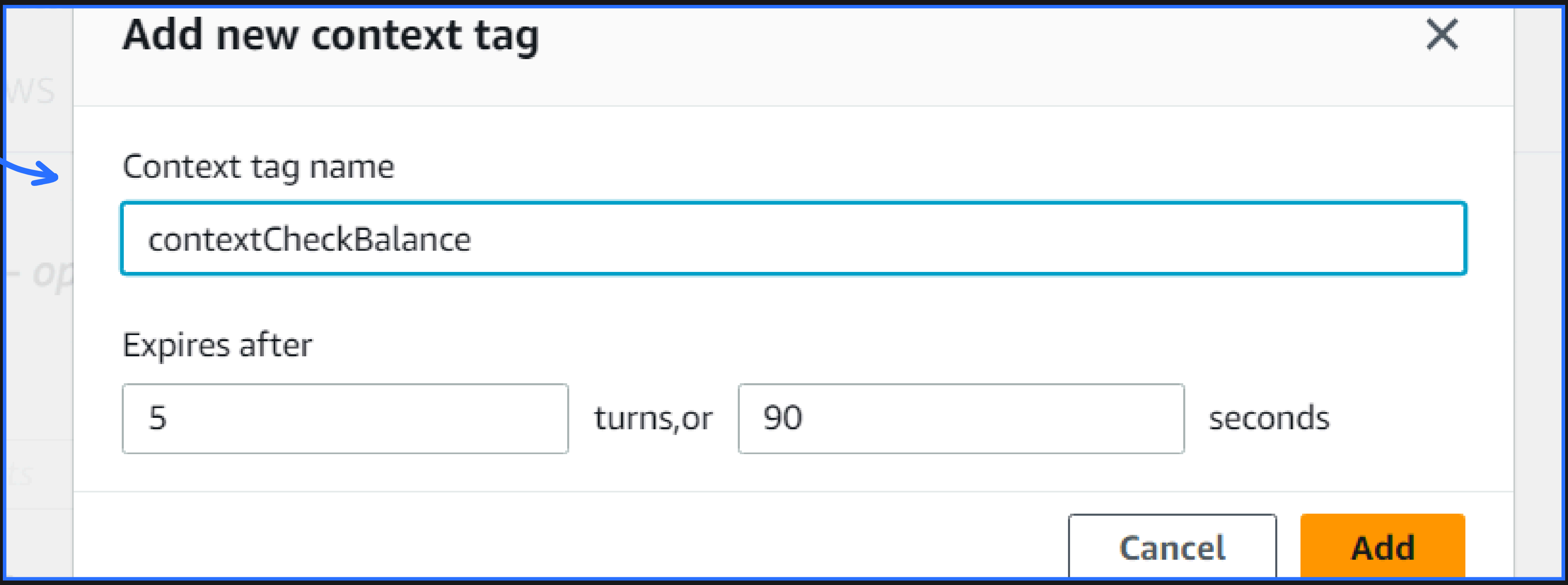
My chatbot now returns a bank balance number thanks to Lambda!

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Context Tags

- Context tags are tools for Amazon Lex to remember specific pieces of information gathered from a conversation, and reuse that information throughout the session with its user.
- There are two types of context tags: output context tags and input context tags.
- I created an output context tag called contextCheckBalance, and I created this in the intent CheckBalance.

A look at output contexts


Add new context tag

Context tag name

contextCheckBalance

Expires after

5 turns,or 90 seconds

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

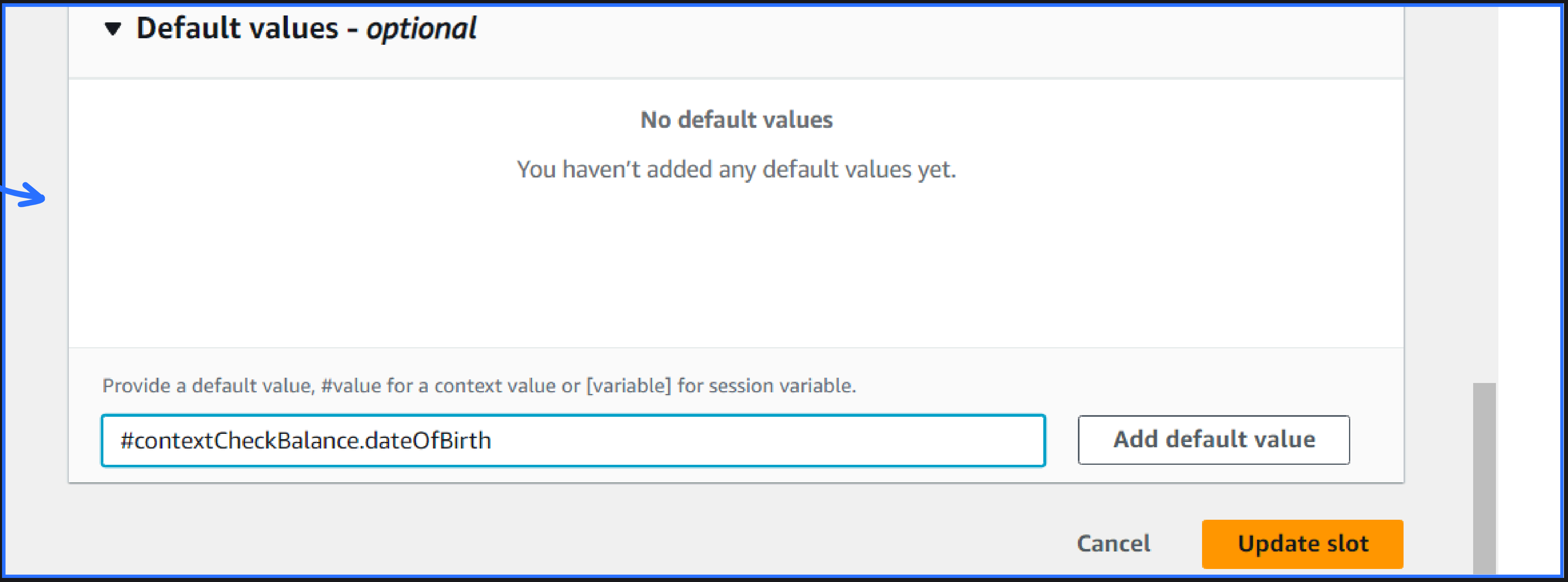
A Follow-Up Intent

- I created a new intent called **FollowupCheckBalance.** The purpose of this intent is to let the user check another account's balance without having to provide their date of birth again.

- This intent is related to the previous intent I made, **CheckBalance,** because FollowUpCheckBalance will only get triggered after the user has checked their balance once already (i.e., triggered CheckBalance).

- I created an input context, **contextCheckBalance**, that uses the exact tag as the output tag I've set up in the CheckBalance intent. What this means is, the input information we are looking for in this intent (FollowUpCheckBalance) can now be retrieved from the CheckBalance intent through this tag.

A look at input contexts


▼ Default values - *optional*
No default values
You haven't added any default values yet.
Add default value
Cancel
Update slot

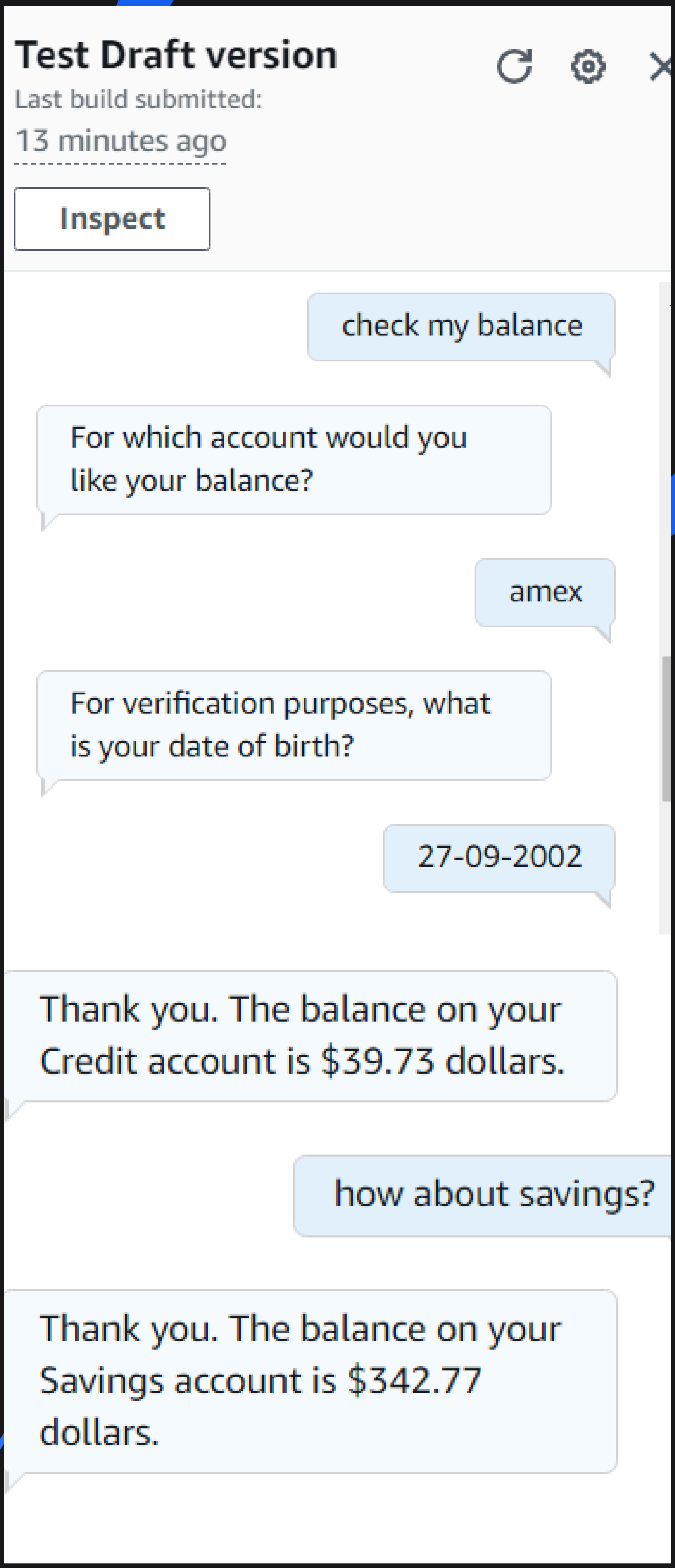
Provide a default value, #value for a context value or [variable] for session variable.

#contextCheckBalance.dateOfBirth

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Context Tags in Action

- Conversation time! I built and tested my bot after creating the context tags and new intent.
- To see the context tags and the follow-up in intent in action, I first triggered the CheckBalance intent, followed up with the utterance "**What about savings**" to trigger FollowUpCheckBalance.
- If I had tried to trigger FollowUpCheckBalance without setting up any context, my chatbot would not have the context needed to fulfill the conversation. As a result, it will return the FallbackIntent i.e. let the user know it doesn't understand the request.


Test Draft version
Last build submitted:
13 minutes ago

check my balance

For which account would you like your balance?

amex

For verification purposes, what is your date of birth?

27-09-2002

Thank you. The balance on your Credit account is $39.73 dollars.

how about savings?

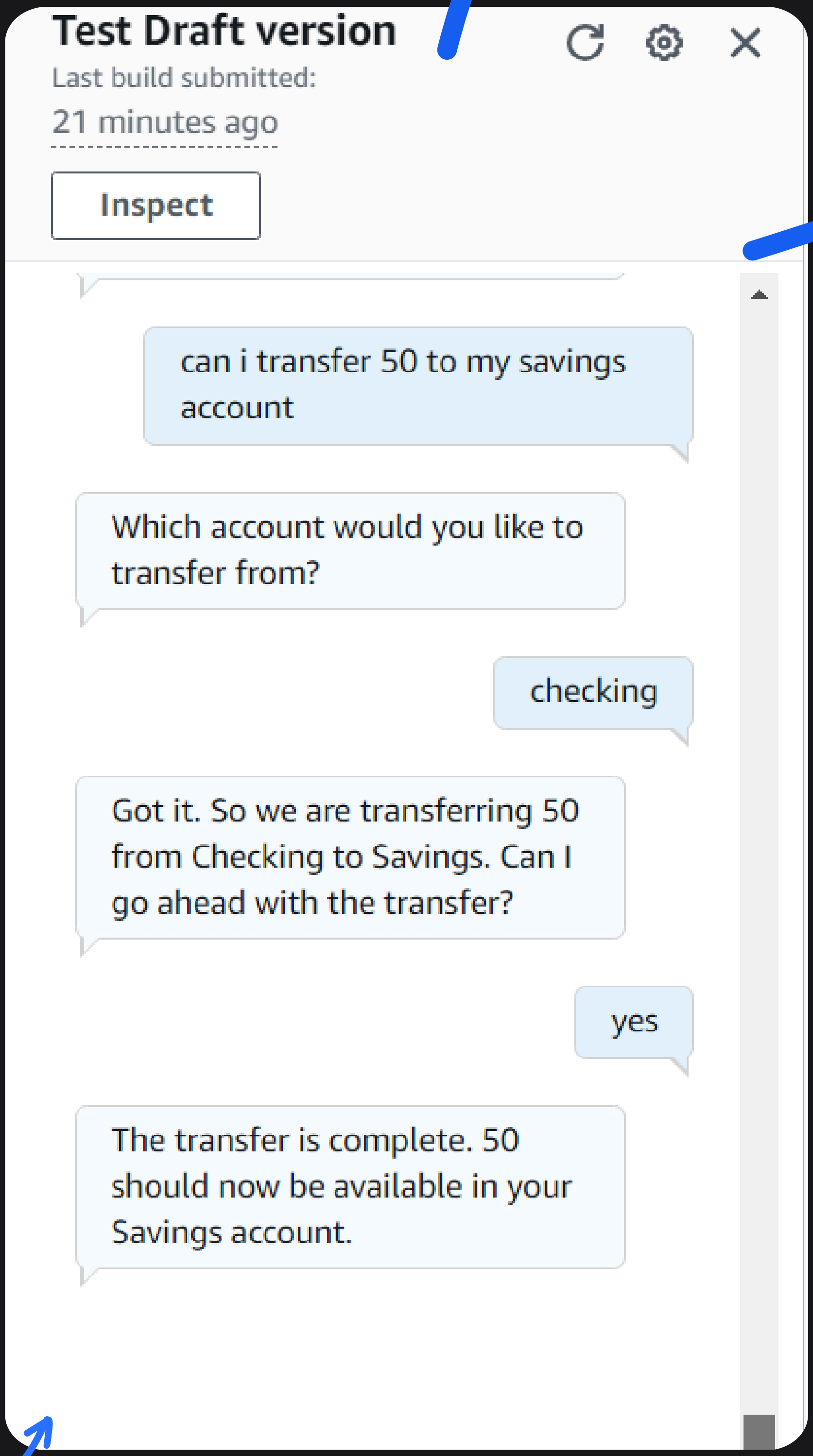
Thank you. The balance on your Savings account is $342.77 dollars.

My chatbot now carries over the user's date of birth to the next intent!

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

More slots!

- Slots are pieces of information that my chatbot needs to fulfill an intent.
- The final intent for my chatbot was TransferFunds, which will help the user transfer money between bank accounts.
- For this intent, I had to use the same slot type twice. This is because the TransferFunds intent involved two different accounts- the source account (i.e., the account that we are transferring money from) and the target account (i.e., the account that the money will land in).
- I also learned how to create confirmation prompts, which are prompts designed for the chatbot to confirm the user's intention to carry out the intent. In this project, a confirmation prompt was used for the chatbot to confirm that the user is wanting to transfer a specific amount of money between two of their bank accounts.



can i transfer 50 to my savings account

Which account would you like to transfer from?

checking

Got it. So we are transferring 50 from Checking to Savings. Can I go ahead with the transfer?

yes

The transfer is complete. 50 should now be available in your Savings account.

A conversation demonstrating the two slots and the confirmation prompts in action!

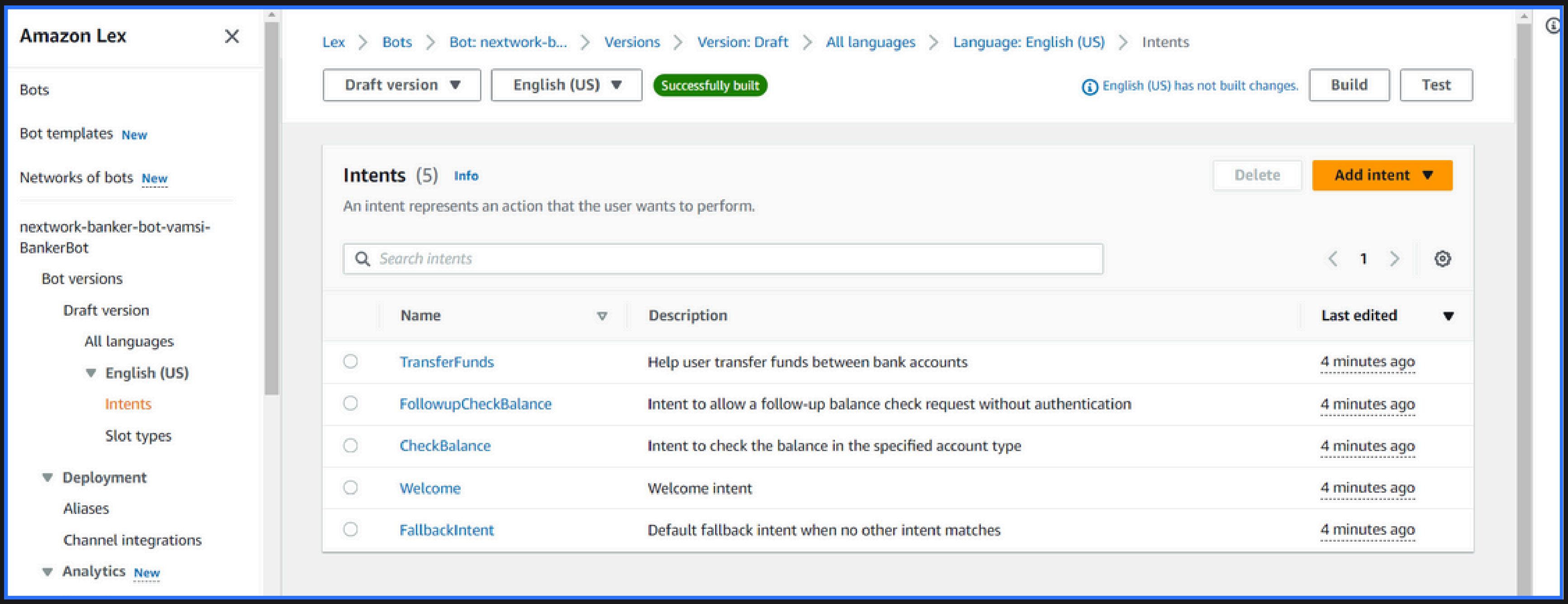
Vamsi Muppana

linkedin.com/in/vamsi-muppana-564373209

Deploying with CloudFormation

- AWS CloudFormation is a service that helps users deploy AWS resources in seconds, by defining the resources and their characteristics in a code file (called a YAML file).
- As an extension to this project, I learned how to deploy the entire BankerBot using a single CloudFormation stack.
- Doing this took me 2 minutes to set up the CloudFormation stack. 3-4 minutes to wait for deployment to complete.
- Something, I learned from deploying with CloudFormation was that you can deploy resources from all different AWS services in the same YAML file.

CloudFront deployed this for me!



Intents (5) Info

An intent represents an action that the user wants to perform.

Name	Description	Last edited
TransferFunds	Help user transfer funds between bank accounts	4 minutes ago
FollowupCheckBalance	Intent to allow a follow-up balance check request without authentication	4 minutes ago
CheckBalance	Intent to check the balance in the specified account type	4 minutes ago
Welcome	Welcome intent	4 minutes ago
FallbackIntent	Default fallback intent when no other intent matches	4 minutes ago

Vamsi Muppana

linkedin.com/in/vamsi-muppana-564373209

My Key Learnings

01 Created a chatbot from scratch using Amazon Lex and configured its responses to basic user inputs.

02 I've set up the new intents that help you check your account balance!

03 I've set up a connection between Amazon Lex bot and AWS Lambda, integrating Amazon Lambda for real-time data processing.

04 Design fallback intents to manage unclear user requests efficiently.

05 I've learned how to create a functional chatbot, BankerBot, designed to assist customers of an imaginary bank with checking their account balance and transferring money between accounts.

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Final thoughts...

- This project took me around 3 hours to complete, and documentation took me about 2 hours.
- Delete EVERYTHING at the end! Let’s keep this project free :)
- One thing I didn’t expect was how much time-saving and less effort it took to deploy the bot in CloudFormation Stack.

Vamsi Muppana
linkedin.com/in/vamsi-muppana-564373209

Find this helpful?

yes!

Like this post

Leave a comment

Save for later

Let's connect!

Vamsi Muppana

in linkedin.com/in/vamsi-muppana-564373209

NEXTWORK