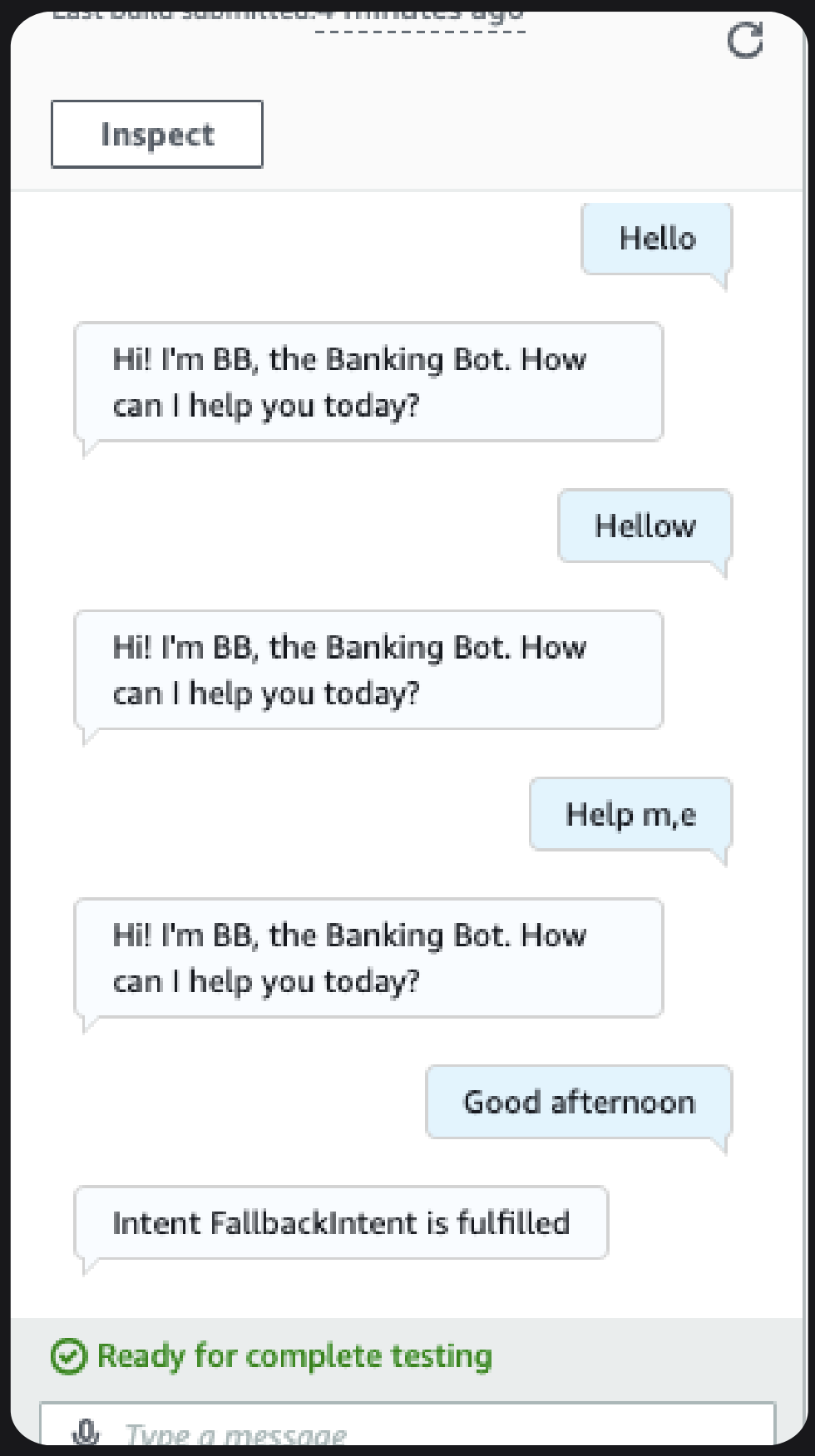
NEXTWORK

How I built a chatbot with Amazon Lex



Abdigani Awil Mohamed
@abdiganiawil10

What is Amazon Lex?

What it does:

- An AWS tool that allows you to build conversational interfaces for applications using voice and text.

Why it's useful:

- It uses AI/ML capabilities to classify user intents and understand intents that are beyond what 'I've programmed.

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

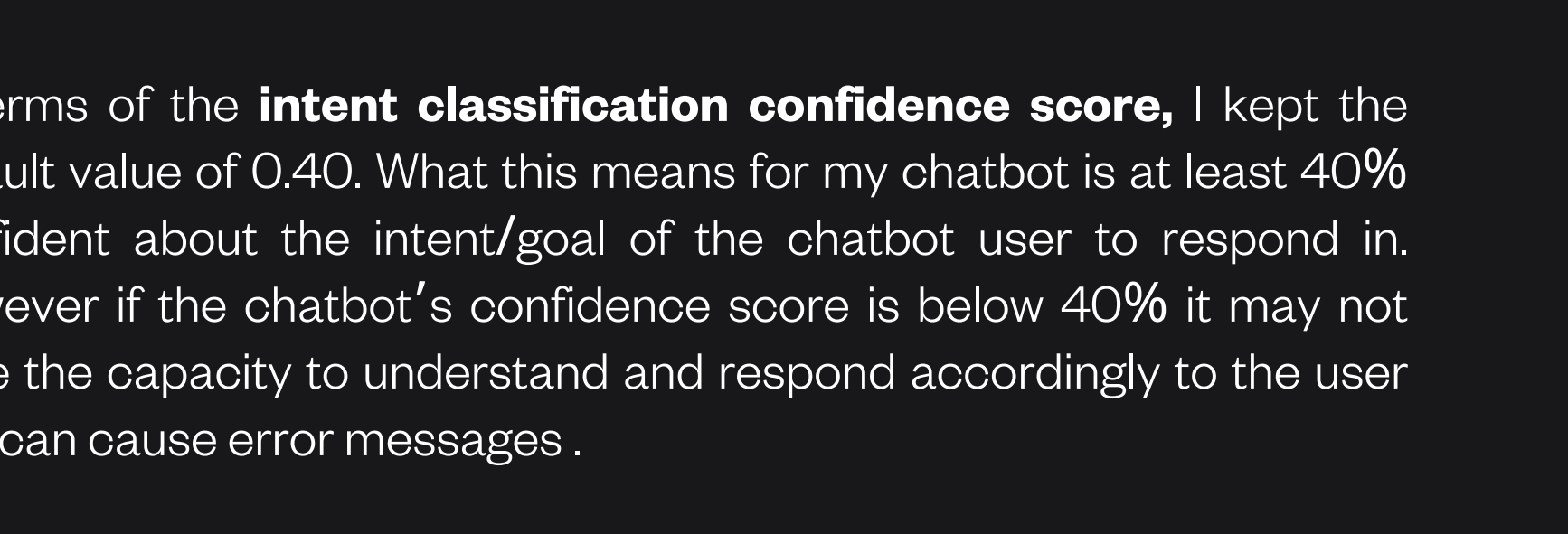
- In this project I'm using Amazon Lex to create BankerBot, a chatbot that can recognise greetings through messages or the voice of the user and also return error messages if it doesn't quite understand the users intent.

Abdigani Awil Mohamed
@abdiganiawil10

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 at least 40% confident about the intent/goal of the chatbot user to respond in. However if the chatbot's confidence score is below 40% it may not have the capacity to understand and respond accordingly to the user and can cause error messages .

Setting up my Lex chatbot...



Abdigani Awil Mohamed
@abdiganiawil10

Create an intent in Lex

- Intents represent users' goals/purposes for using the chatbot. In Amazon Lex, a chatbot is defined by the intents that it supports.

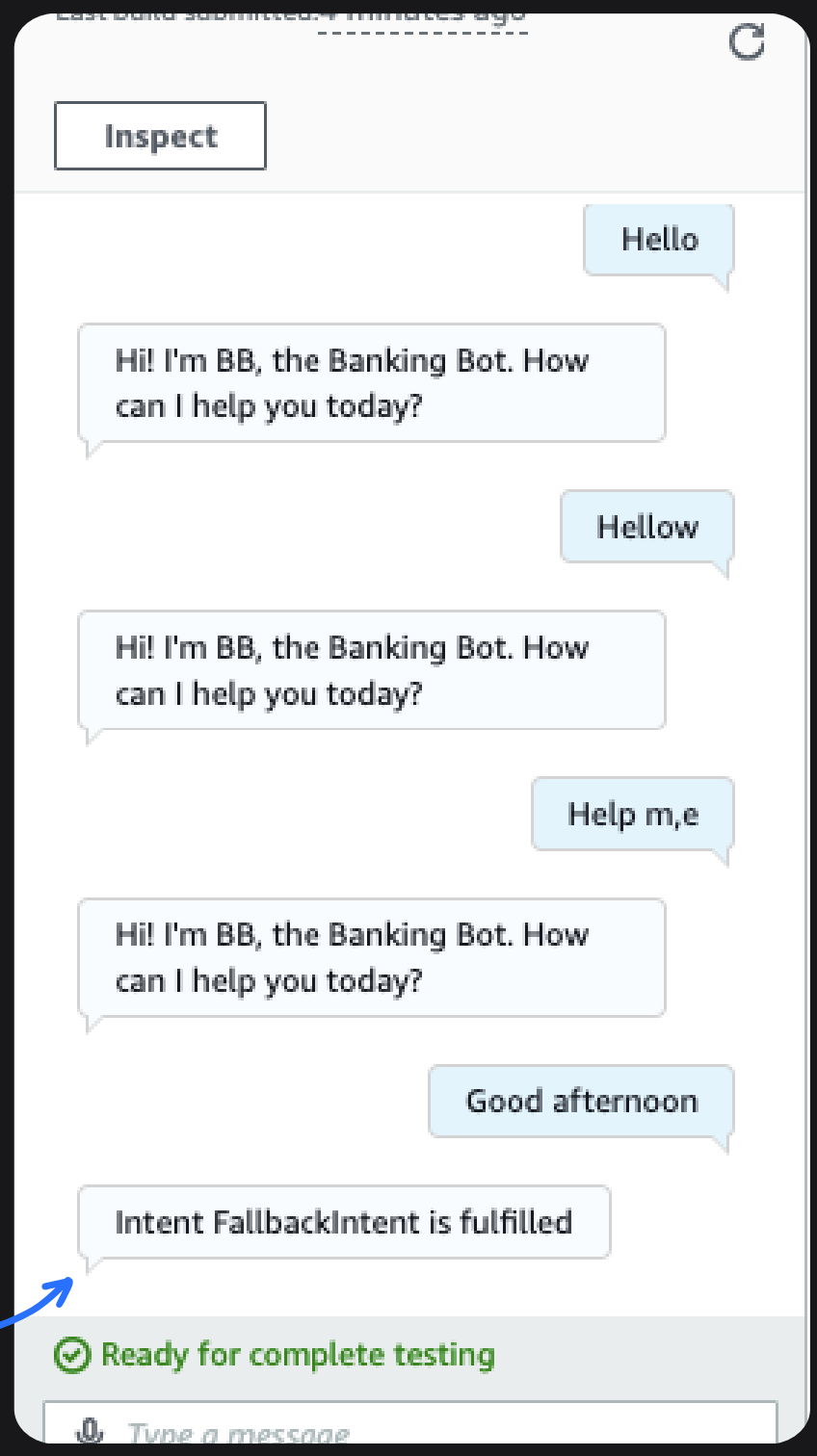
- My first intent, WelcomeIntent, was created to greet the user by saying Hello.

- To set up this intent, I created sample utterances such as "Hello" and "I need help" and how the chatbot should respond.

- I launched and tested the chatbot, which could still respond if I entered Hello incorrectly by spelling it as "Hellow" which was quite impressive as the chatbot's intent still understood it as a greeting.

- However, the chatbot returned the error message "Intent FallbackIntent is fulfilled" when I entered "Good morning".

- This error message occurred because my chatbot could not understand the intent phrase of "Good morning".



My first test of the chatbot

Abdigani Awil Mohamed
@abdiganiawil10

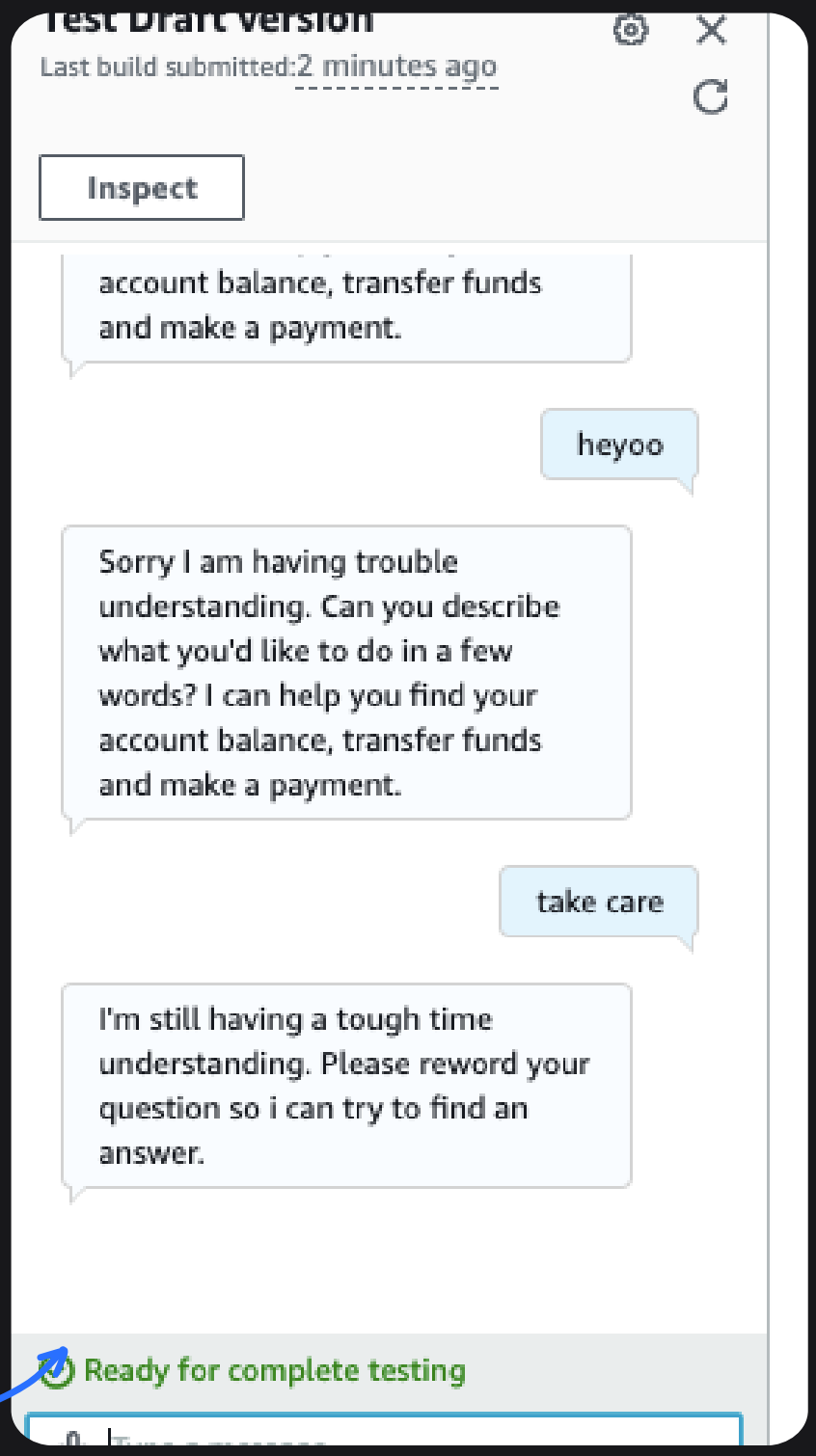
Manage FallbackIntent

- FallbackIntent is a default intent in every chatbot that gets triggered when the chatbot does not recognise 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 a closing response in the intent's set up. For example "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. E.g. "I'm still having a tough time understanding. Please reword your question so I can try to find an answer."



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

Abdigani Awil Mohamed
@abdiganiawil10

My Key Learnings

01 Amazon Lex is essentially is an AI service that allows you to build chatbots that understand and can respond to natural languages such as voice and messages.

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

03 AI/ML is used in Amazon Lex to understand what the user is saying and to help respond in the best possible way.

04 FallbackIntent is used for when the chatbot fails to understand what the users says.

Abdigani Awil Mohamed
@abdiganiawil10

Final thoughts...

- This part of the project took me around 30 minutes and writing up the documentation took me another 30 minutes.

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

- **What's next?** In the next phase of this project, I'll be adding a new flow that lets users check their account balances and verify their identity with their birthday. I'll be creating a custom slot type to handle the different bank account types. Excited to bring this feature to life and make our BankerBot smarter and more interactive! 🚀👀

Abdigani Awil Mohamed

@abdiganiawil10