



BUILDING A CHATBOT

CSCI 5410 – Serverless Data Processing

Assignment 2 –Part B

Dhrumil Amish Shah (B00857606)
dh416386@dal.ca

Amazon Lex [1] is a service provided by Amazon that helps users to rapidly and simply create chatbots with highly engaging user experiences with no prior learning expertise. To understand Amazon Lex, this assignment focuses on building a chatbot on **OrderFood**. OrderFood chatbot is a pizza place that accepts delivery and pickup of three regular pizzas namely veg, cheese, and pepperoni.

Figures 1, 2, 3, 4 and 5 display the screenshots of the steps of creating a chatbot **OrderFood** with the configuration settings for the chatbot.

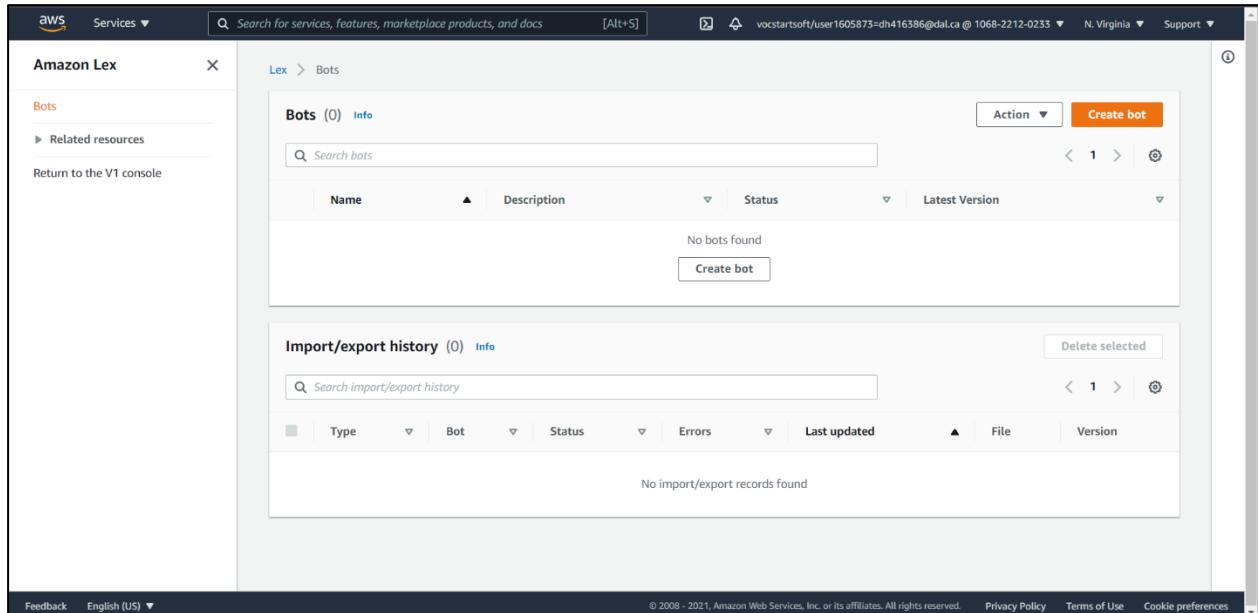


Figure 1 - Initial figure of Amazon Lex dashboard with no chatbots [1]

Configure bot settings

Creation method

- Create: Create a blank bot.
- Start with an example: An example bot has preconfigured languages, intents, and slot types. You can change these settings.

Bot configuration

Bot name: OrderFood
Maximum 100 characters. Valid characters: A-Z, a-z, 0-9, -, _

Description - optional:
This description appears on bot list page. It can help you identify the purpose of your bot.
OrderFood is a bot that serves three regular pizzas, namely veg, cheese, and pepperoni.

IAM permissions

IAM permissions are used to access other services on your behalf.

Runtime role: Choose a role that defines permissions for your bot. To create a custom role, use the IAM console.

- Create a role with basic Amazon Lex permissions.
- Use an existing role.

Creating a role takes a few minutes. Don't delete the role or edit the trust or permissions policies in this role until we've finished creating it.

New role: Amazon Lex creates a runtime role with permission to upload to Amazon CloudWatch Logs.
AWSLexServiceRoleForLexV2Bots_M8YE3740B

Children's Online Privacy Protection Act (COPPA)

Is use of your bot subject to the Children's Online Privacy Protection Act (COPPA)?
 Yes
 No

Idle session timeout

You can configure how long a session is maintained when the user does not provide any input and the session is idle. Amazon Lex retains context information until a session ends.

Session timeout: 30 minute(s)
By default, session duration is 5 minutes, but you can specify any duration between 1 and 1440 minutes (24 hours).

Advanced settings - optional

Tags - bot

You can tag the bot with a label. Tags can help you manage, identify, organize, search for, and filter resources.

Key	Value - required	Remove
company	OrderFood	Remove
food-type	Pizza	Remove

Add new tag
You can add 48 more tags.

Tags - testBotAlias

The test alias points to the draft version and intended for testing purposes. You can tag the test bot alias with a tag.

No tags associated with the resource.

Add new tag
You can add 50 more tags.

Next

Figure 2 - Setup of OrderFood chatbot [1]

The screenshot shows the 'Add language to bot' configuration page for Step 2: Add languages. The 'Language' dropdown is set to 'English (US)'. The 'Voice interaction' dropdown is set to 'Joey'. A voice sample preview shows the text 'Hello, my name is Joey. Let me know how I can assist you.' with a 'Play' button. The 'Intent classification confidence score threshold' input field is set to '0.40'. At the bottom right are 'Cancel', 'Add another language', and 'Done' buttons.

Figure 3 - Add language to the OrderFood bot (i.e., English US) [1]

The screenshot shows the 'Bots' list page. It displays one bot named 'OrderFood' with the status 'Available'. Below the main list is an 'Import/export history' section which is currently empty, stating 'No import/export records found'.

Figure 4 - Successful creation of OrderFood chatbot with status - Available [1]

The screenshot shows the Amazon Lex console interface for the 'OrderFood' bot. The left sidebar lists 'Bots' and 'OrderFood' (selected). The main content area is divided into several sections:

- Bot details:** Name: OrderFood, Description: OrderFood is a bot that serves three regular pizzas, namely veg, cheese, and pepperoni. ID: RQQQGZDBLP.
- Add languages:** Shows 1 language added. A button to 'View languages'.
- Create versions and aliases for deployment:** Shows 1 alias ('company'). Buttons to 'View aliases' and 'Add channel'.
- Analyze and improve your bot:** Shows Cloudwatch metrics.
- Tags:** 2 tags: company (Value: OrderFood) and food-type (Value: Pizza).
- Resource-based policy:** No policy yet.

Figure 5 - OrderFood chatbot dashboard with description [1]

Figure 6 displays the creation of three Intents related to the OrderFood chatbot namely DeliveryPizzaOrderIntent, PickupPizzaOrderIntent and FallbackIntent.

DeliveryPizzaOrderIntent is configured and responsible for handling the communication related to pizza delivery. For delivery, customer address, order delivery date and time are crucial along with the pizza.

PickupPizzaOrderIntent is configured and responsible for handling the communication related to pizza takeaway. It is assumed that the takeaway is on the same day and the only information required is the arrival time of the customer.

FallbackIntent is configured to handle any false requests by users.

The screenshot shows the AWS Lambda interface for creating a new Lambda function. The left sidebar lists 'Functions' and 'Actions'. The main area shows the configuration for a new function named 'HelloWorld'. The 'Handler' dropdown is set to 'lambda.lambda_handler'. The 'Memory Size' is set to 128 MB. The 'Timeout' is set to 3 seconds. The 'Role' is 'Lambda execution role'. The 'Code' section shows the code has been uploaded and is 'Successfully built'. The 'Environment' section shows the environment variables: 'AWS_LAMBDA_FUNCTION_NAME' (HelloWorld), 'AWS_LAMBDA_FUNCTION_MEMORY_SIZE' (128), 'AWS_LAMBDA_FUNCTION_TIMEOUT' (3), and 'AWS_LAMBDA_FUNCTION_HANDLER' (lambda.lambda_handler). The 'Configuration' section shows the runtime as 'Node.js 12.x' and the deployment package as 'HelloWorld.zip'. The 'Logs' section shows log events for the function's execution.

Figure 6 - Three intents namely DeliveryPizzaOrderIntent, PickupPizzaOrderIntent and FallbackIntent for pizza delivery, pizza pickup and false request handling [1]

Pizza Delivery Related Images of OrderFood Chatbot Intent - DeliveryPizzaOrderIntent

Figures 7, 8, 9, 10 and 11 display the configuration setup of the Intent DeliveryPizzaOrderIntent. The configuration steps consist of conversation flow, intent details, utterances, slots, confirmation prompts, fulfillment, and closing response.

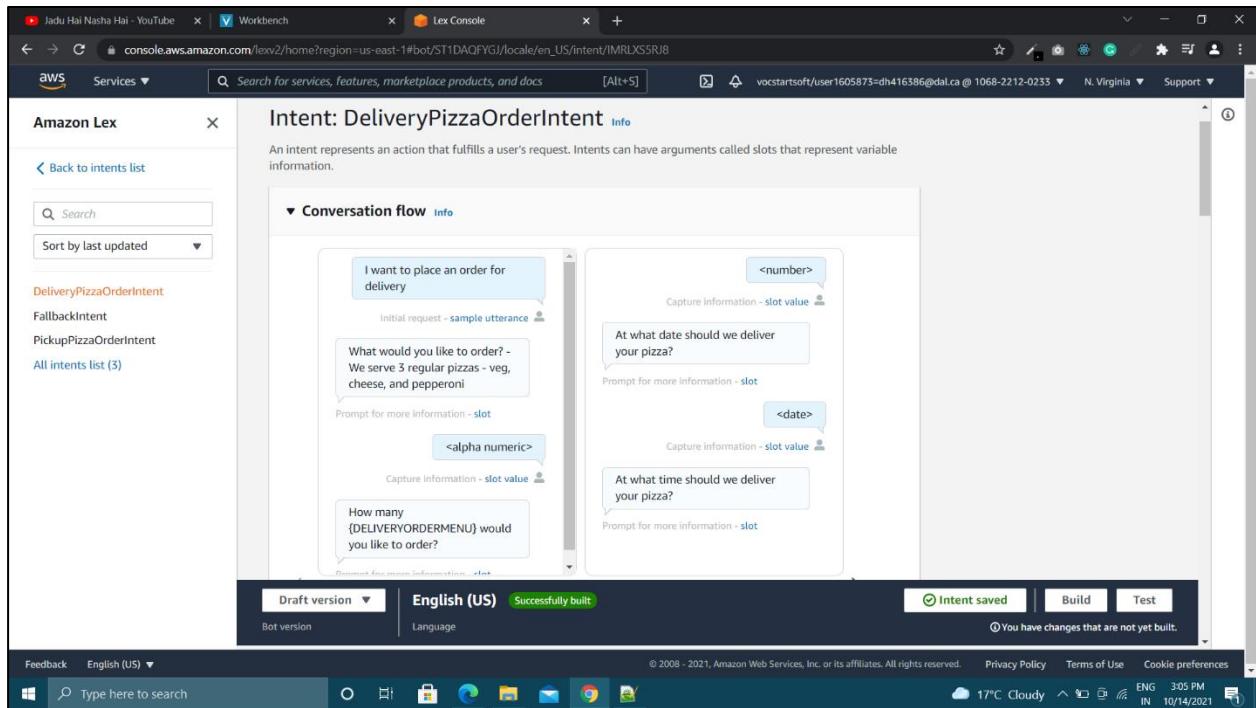


Figure 7 - Pizza delivery conversation flow [1]

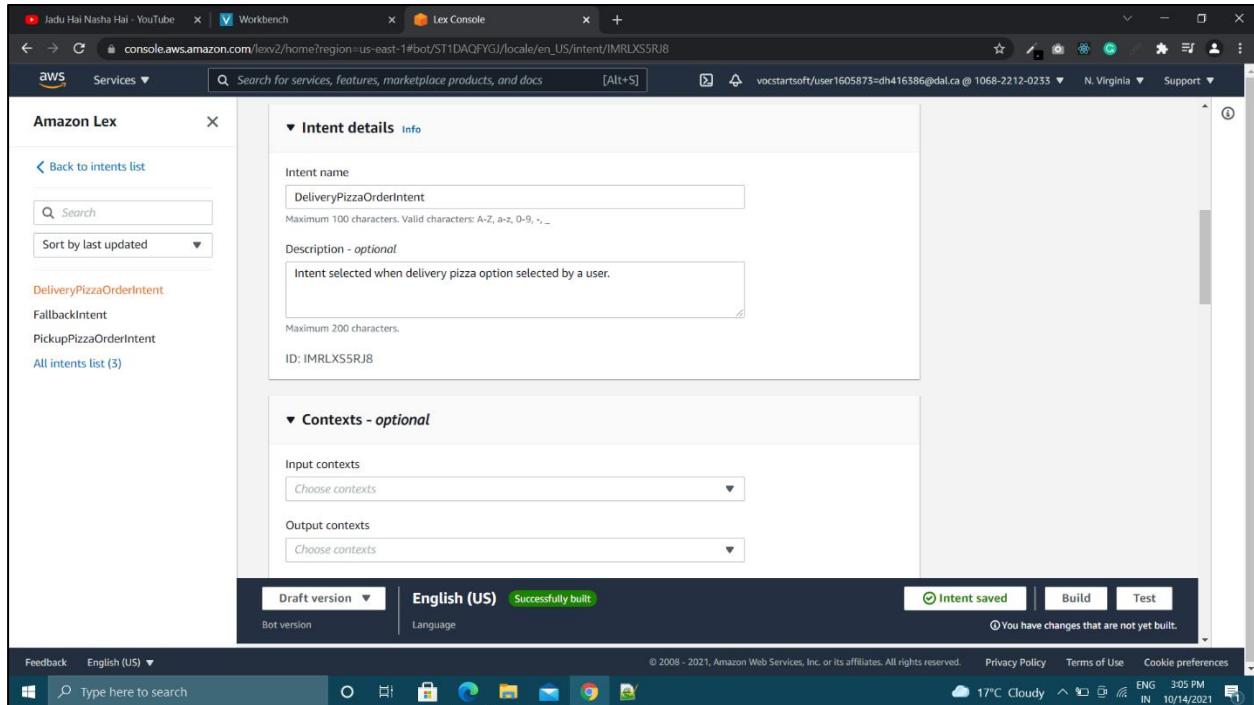


Figure 8 - Pizza delivery intent details that contain intent name and intent description [1]

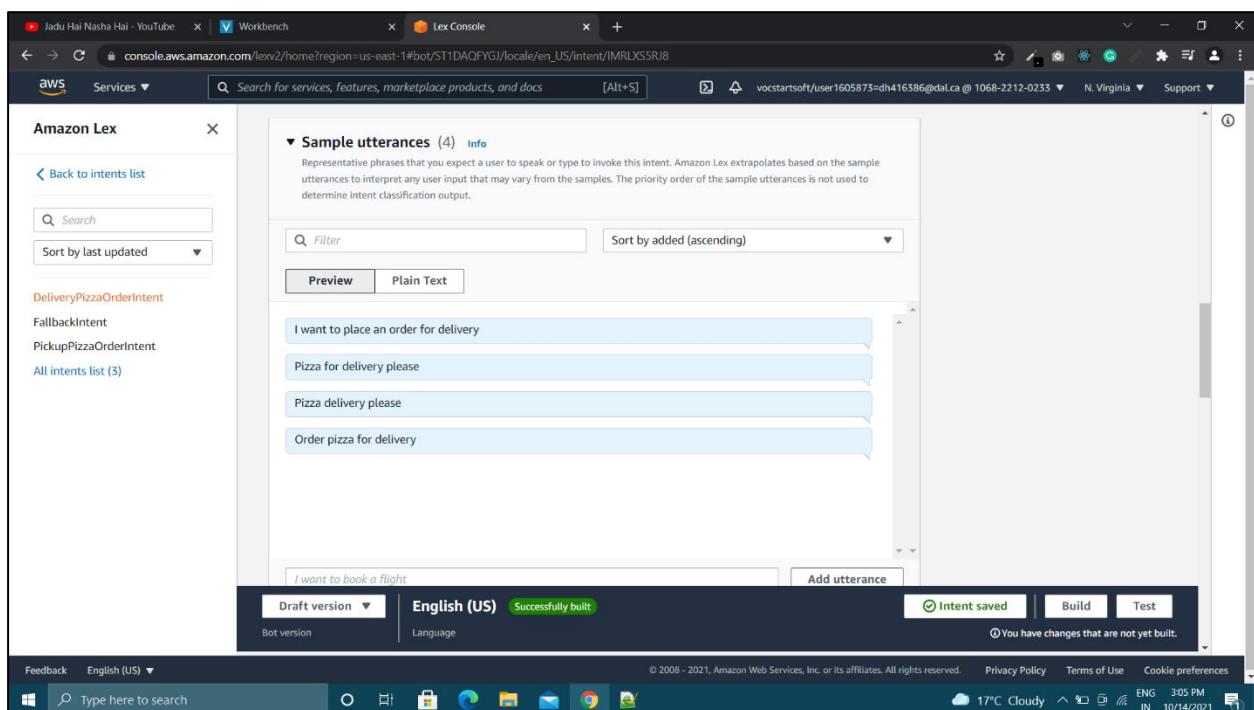


Figure 9 - Pizza delivery utterances that user enters to order pizza for delivery [1]

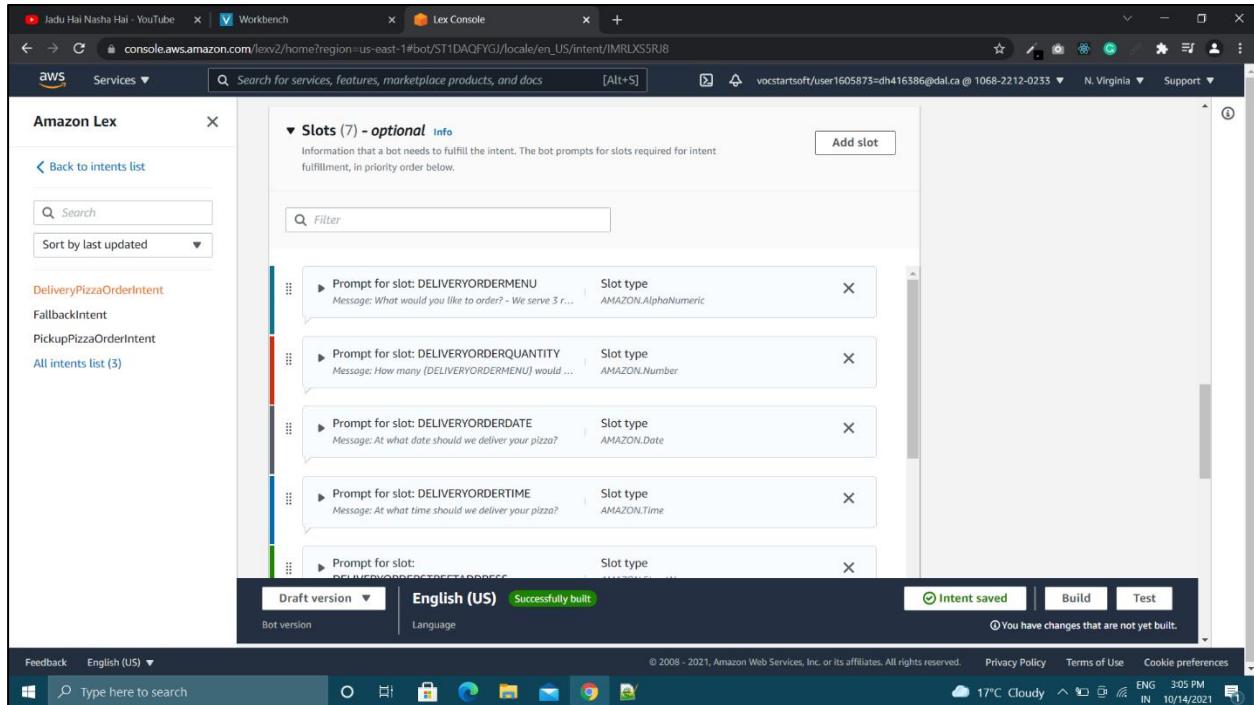


Figure 10 - Pizza delivery slots configured to accept pizzas, delivery address, delivery date and time [1]

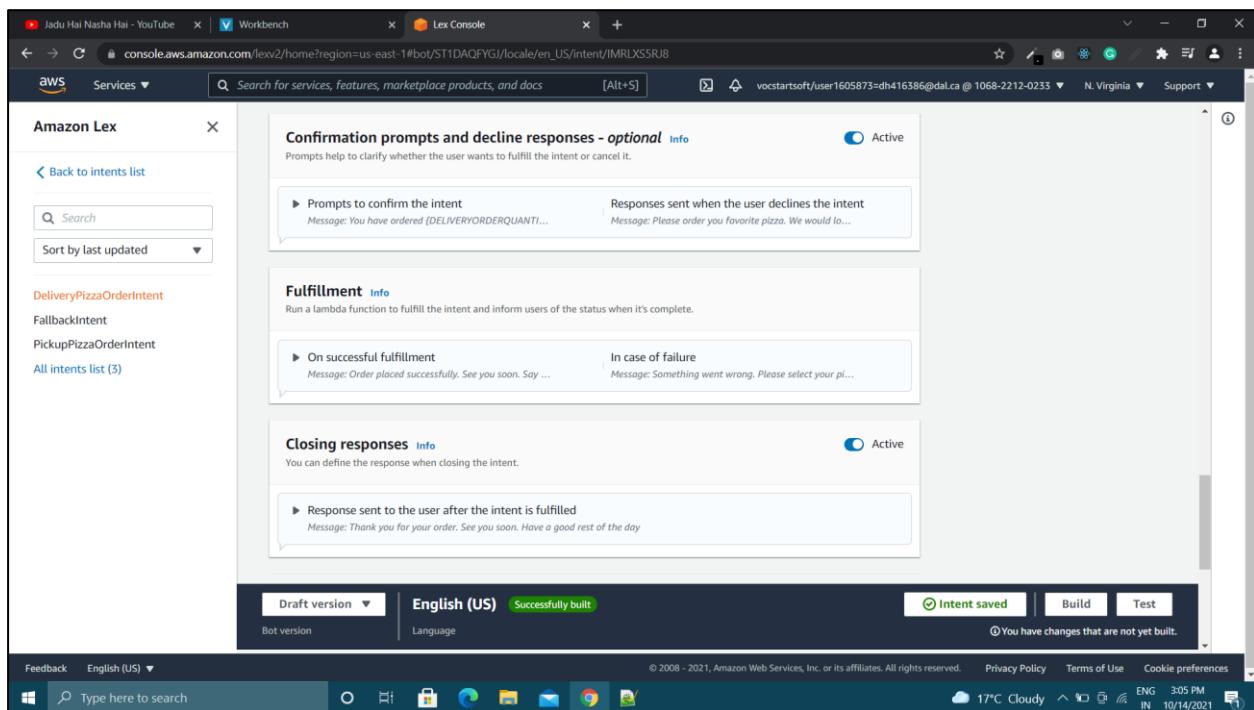


Figure 11 - Pizza delivery confirmation prompts, fulfillment and closing response [1]

Figures 12, 13, 14, 15, 16, 17, 18, 19, 20, and 21 display the OrderFood pizza delivery intent in action. (i.e., Conversation flow in working)

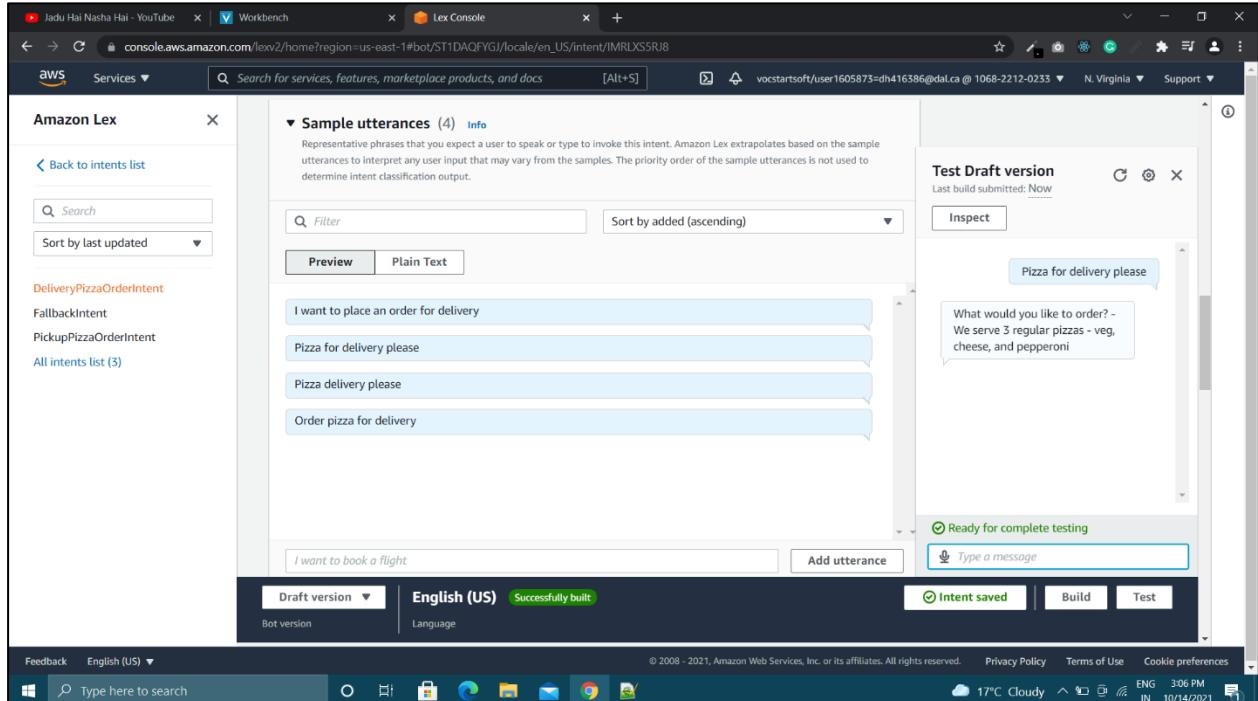


Figure 12 - Pizza delivery chatbot testing conversation in working [1]

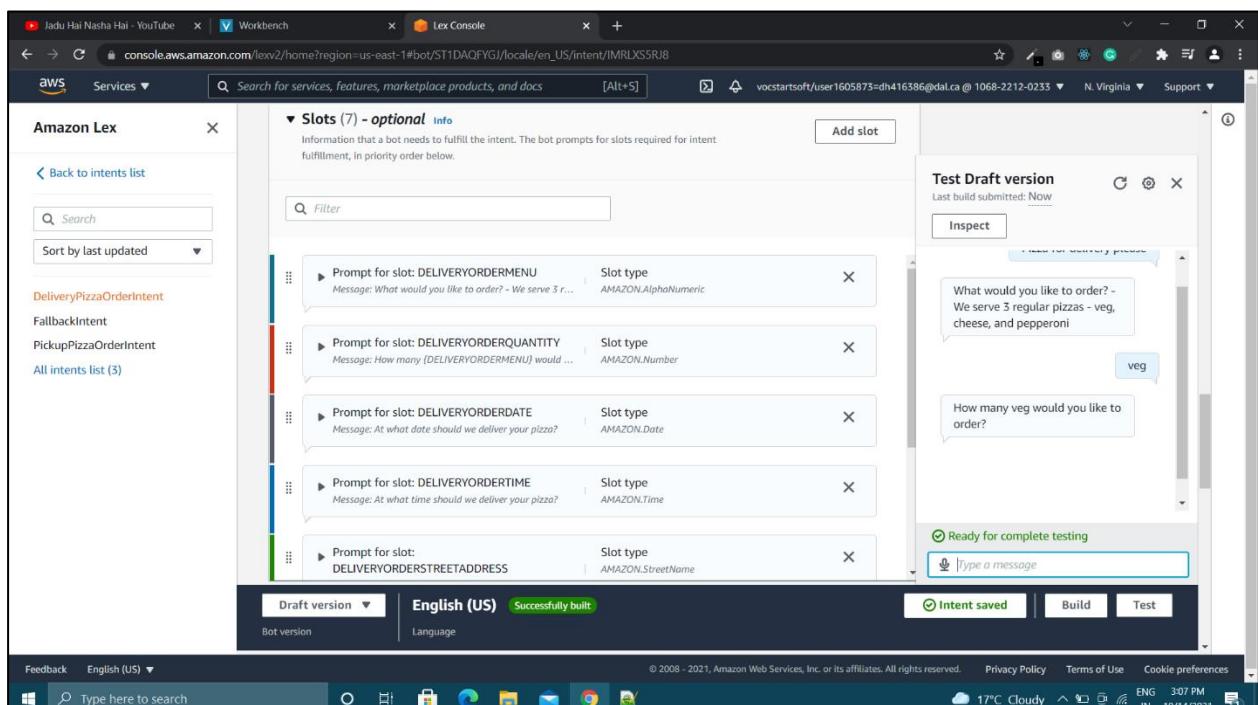


Figure 13 - Pizza delivery chatbot testing conversation in working (contd.) [1]

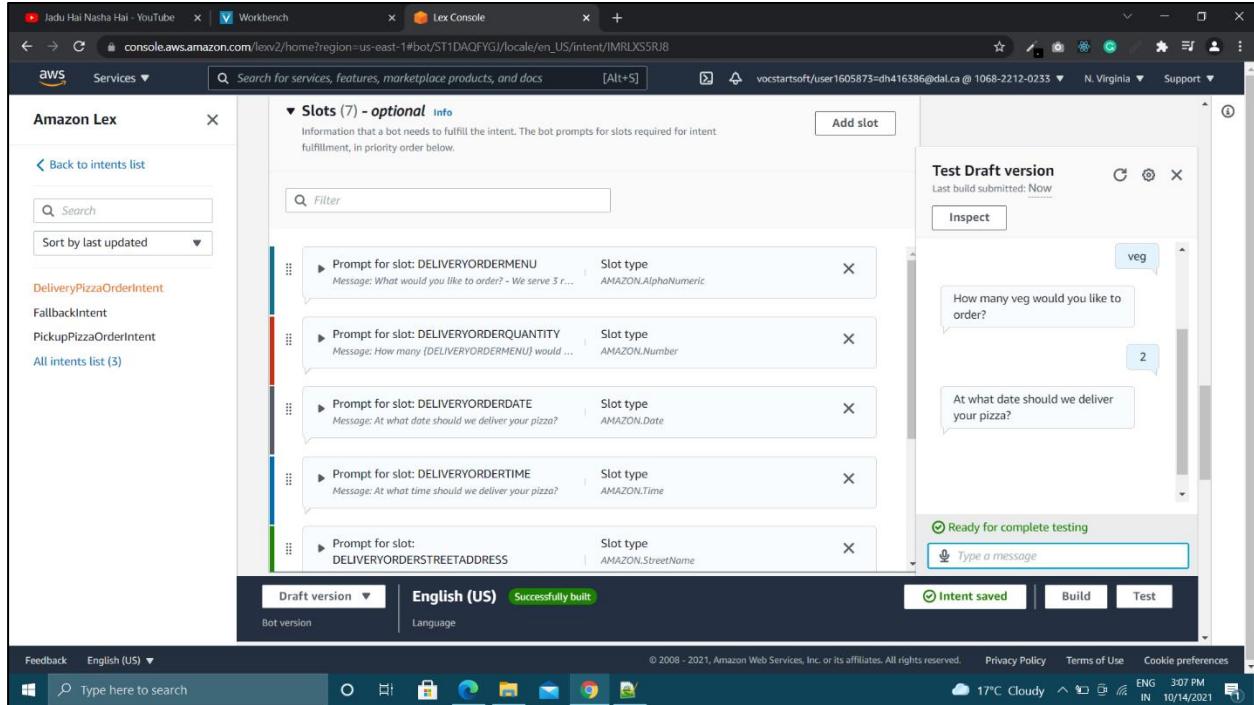


Figure 14 – Pizza delivery chatbot testing conversation in working (contd.) [1]

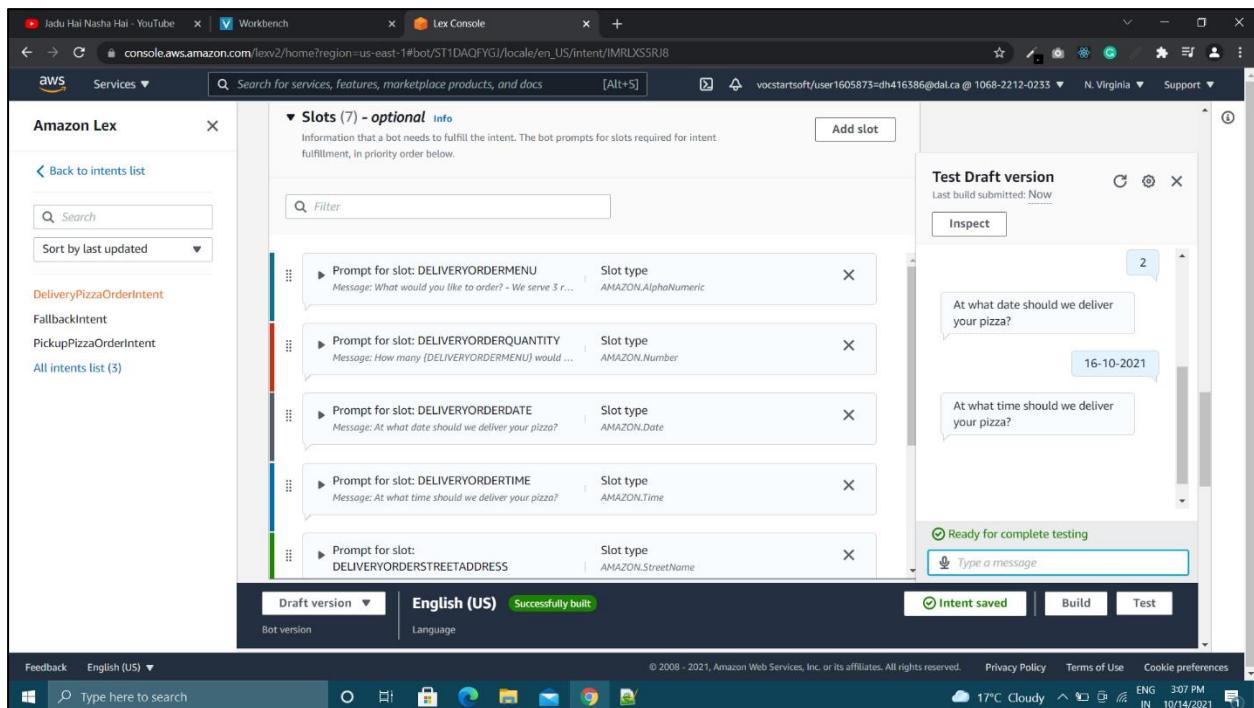


Figure 15 - Pizza delivery chatbot testing conversation in working (contd.) [1]

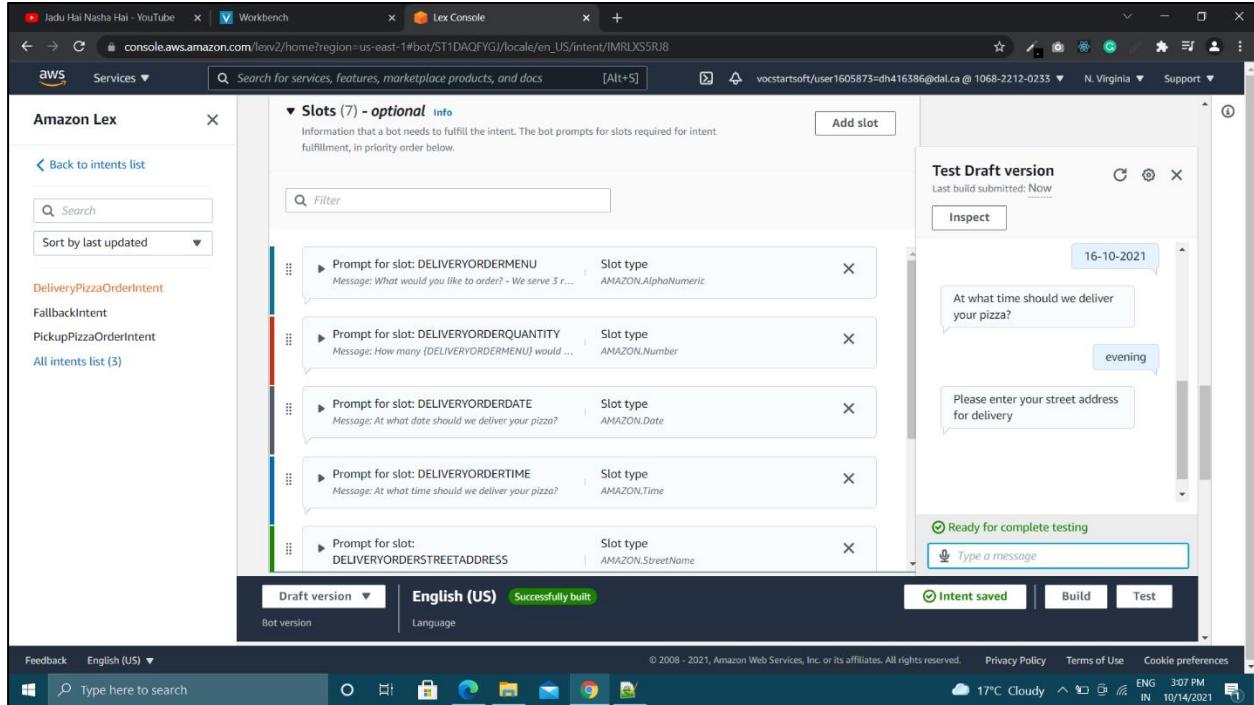


Figure 16 - Pizza delivery chatbot testing conversation in working (contd.) [1]

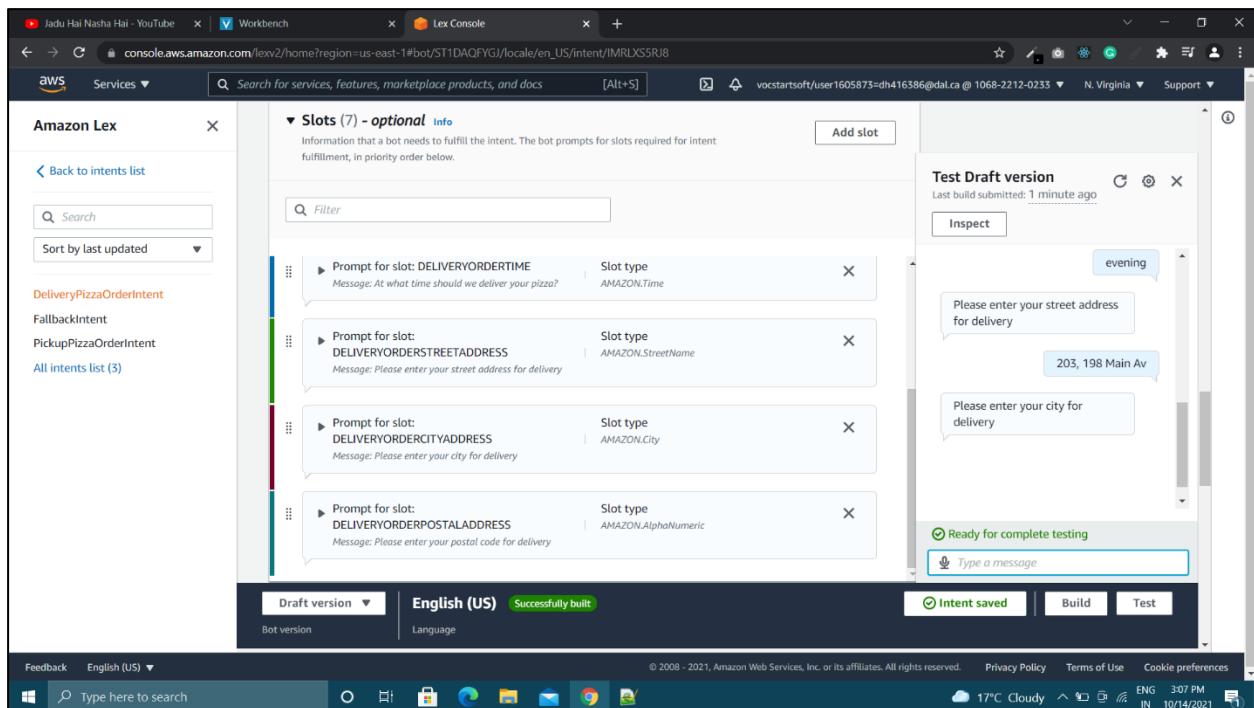


Figure 17 - Pizza delivery chatbot testing conversation in working (contd.) [1]

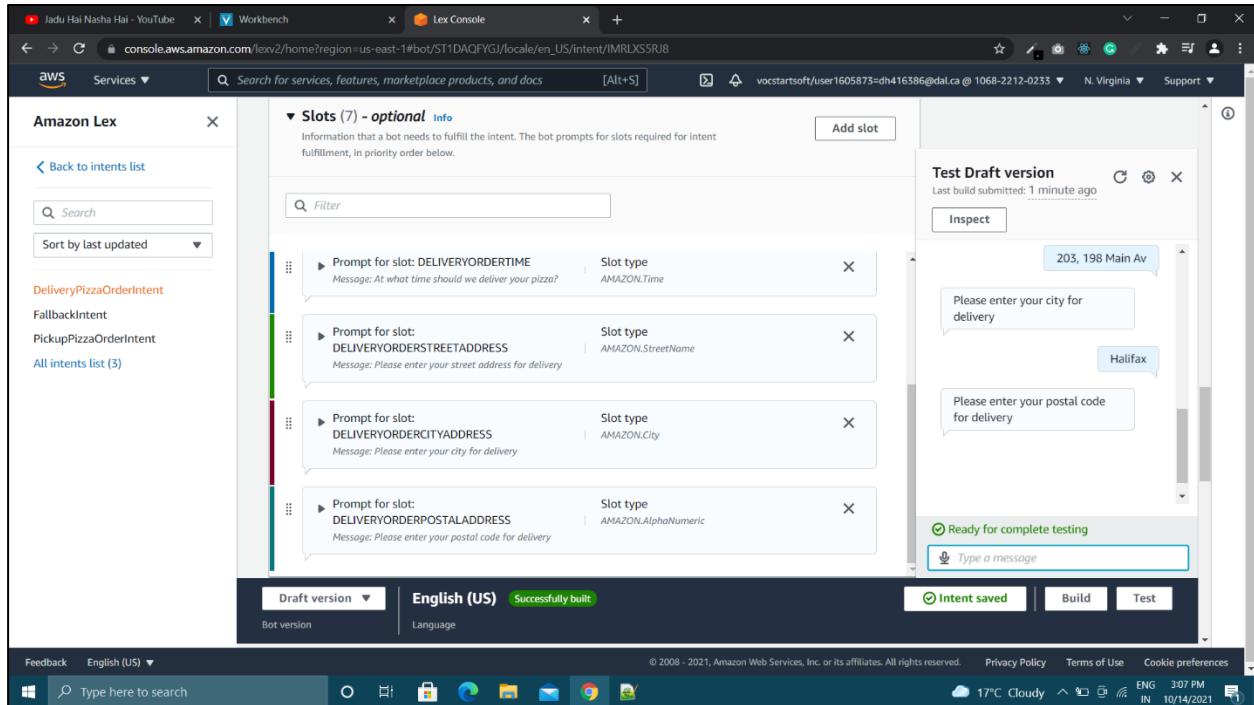


Figure 18 - Pizza delivery chatbot testing conversation in working (contd.) [1]

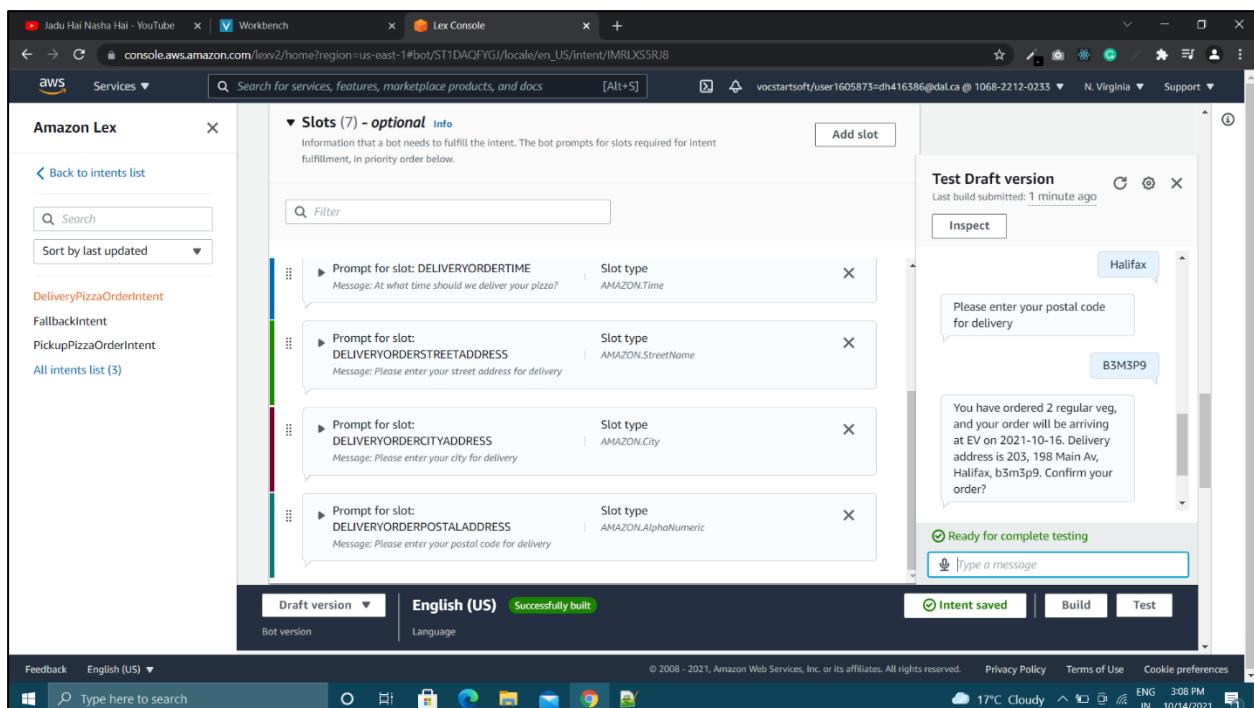


Figure 19 – Pizza delivery chatbot testing conversation in working (contd.) [1]

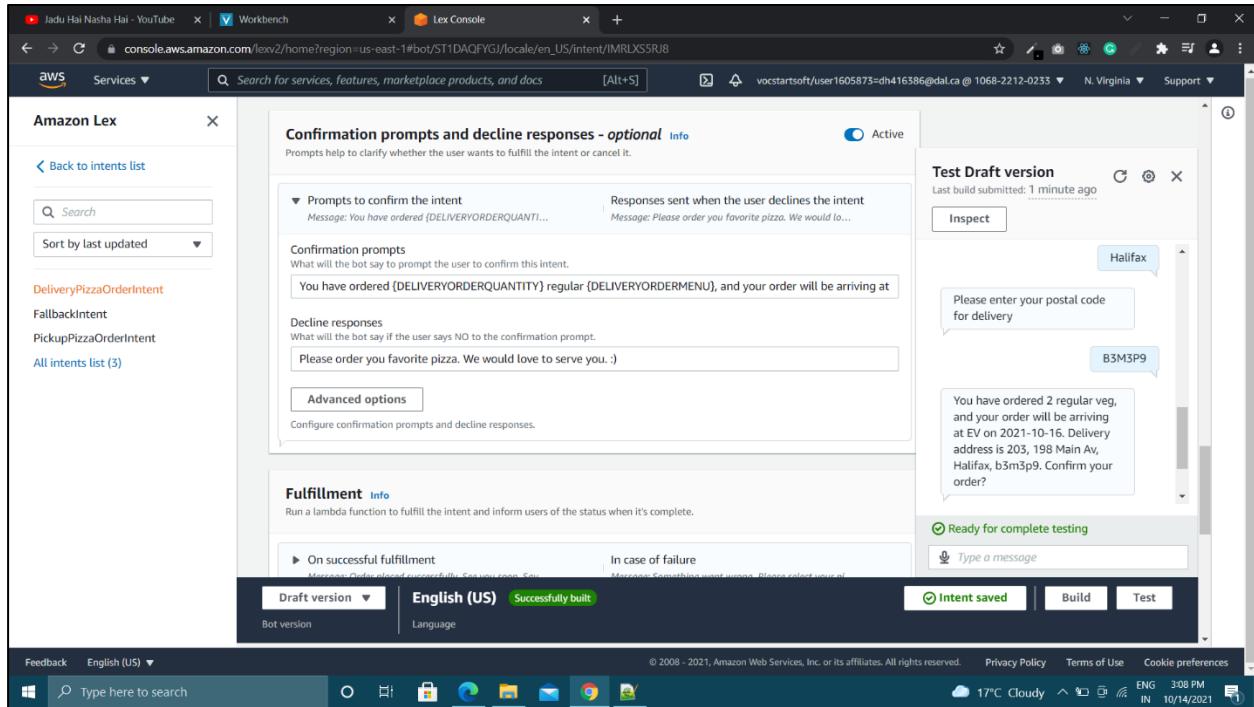


Figure 20 - Pizza delivery chatbot testing conversation in working (contd.) [1]

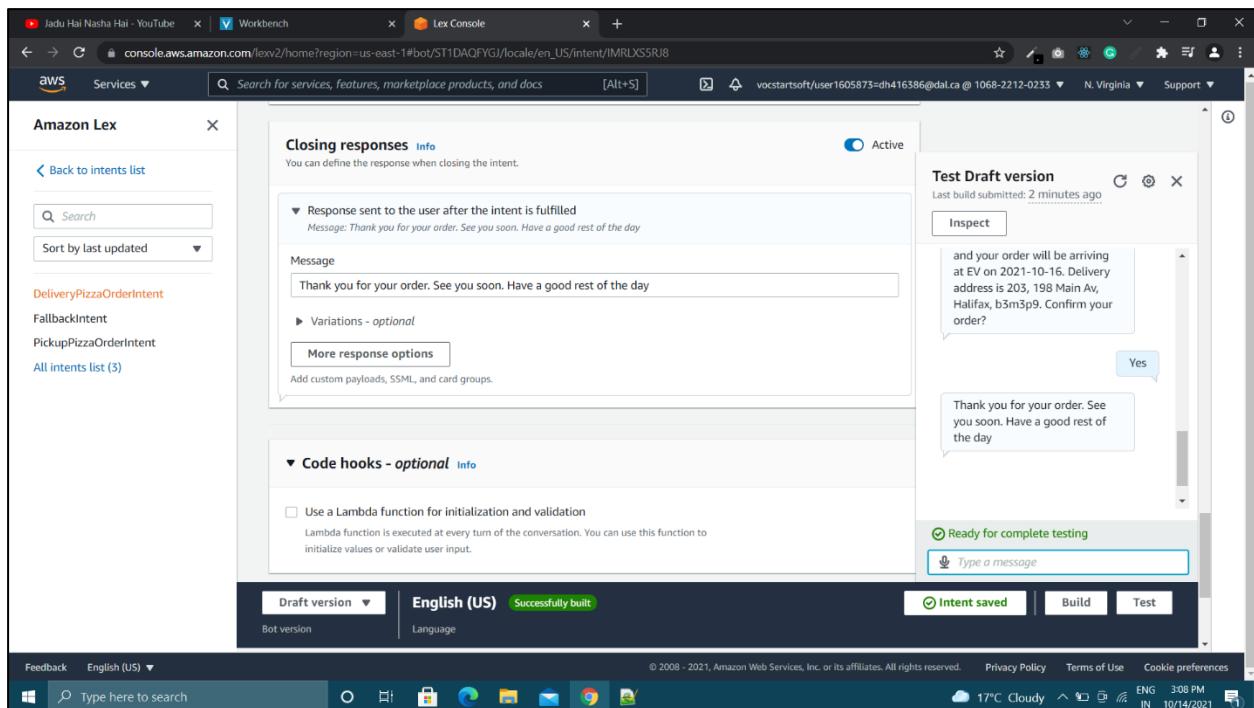


Figure 21 - Pizza delivery chatbot testing conversation in working (contd.) [1]

Pizza Pickup Related Images of OrderFood Chatbot Intent - PickupPizzaOrderIntent

Figures 22, 23, 24, 25, 26, 27, and 28 display the configuration setup of the Intent PickupPizzaOrderIntent. The configuration steps consist of conversation flow, intent details, utterances, slots, confirmation prompts, fulfillment, and closing response.

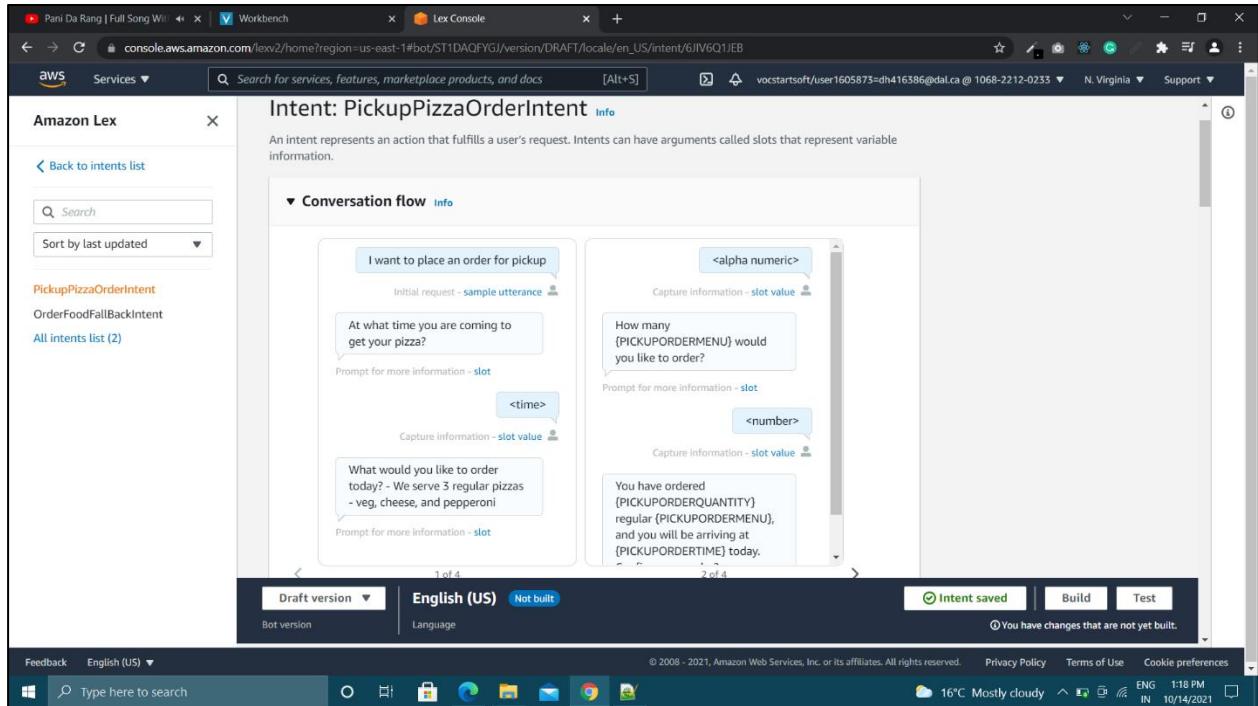


Figure 22 – Pizza pickup conversation flow [1]

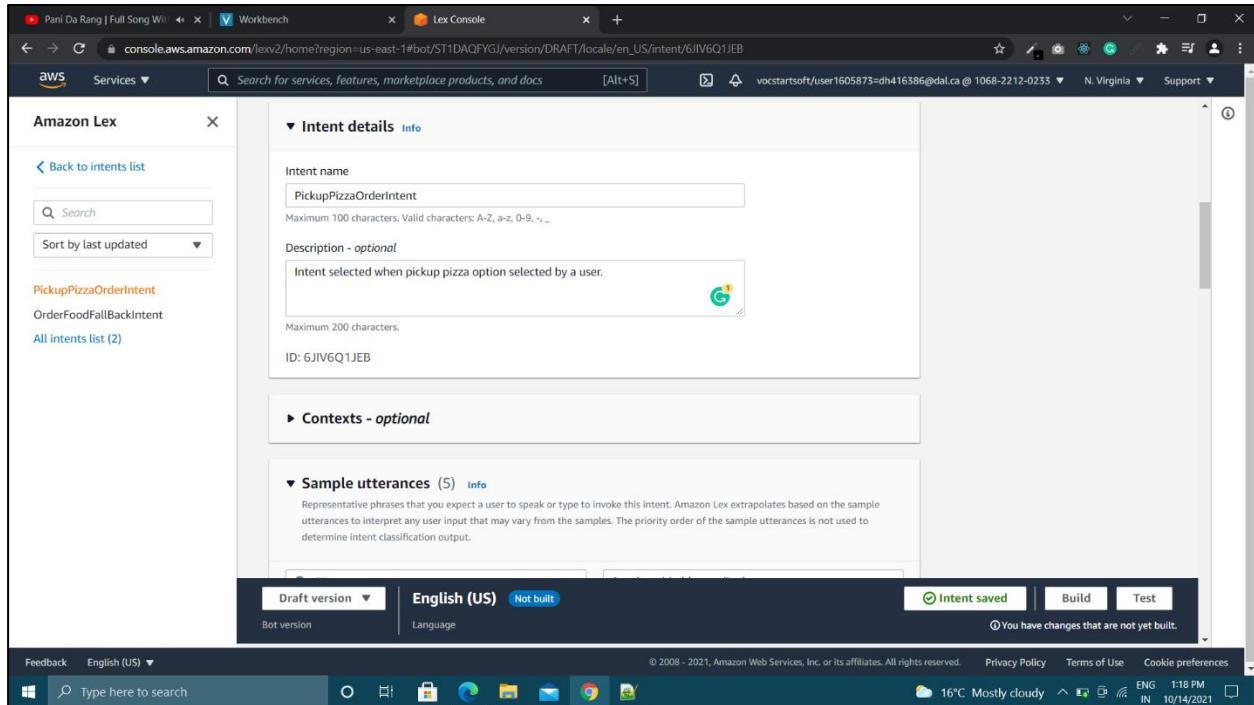


Figure 23 - Pizza pickup intent details that consist of intent name and description [1]

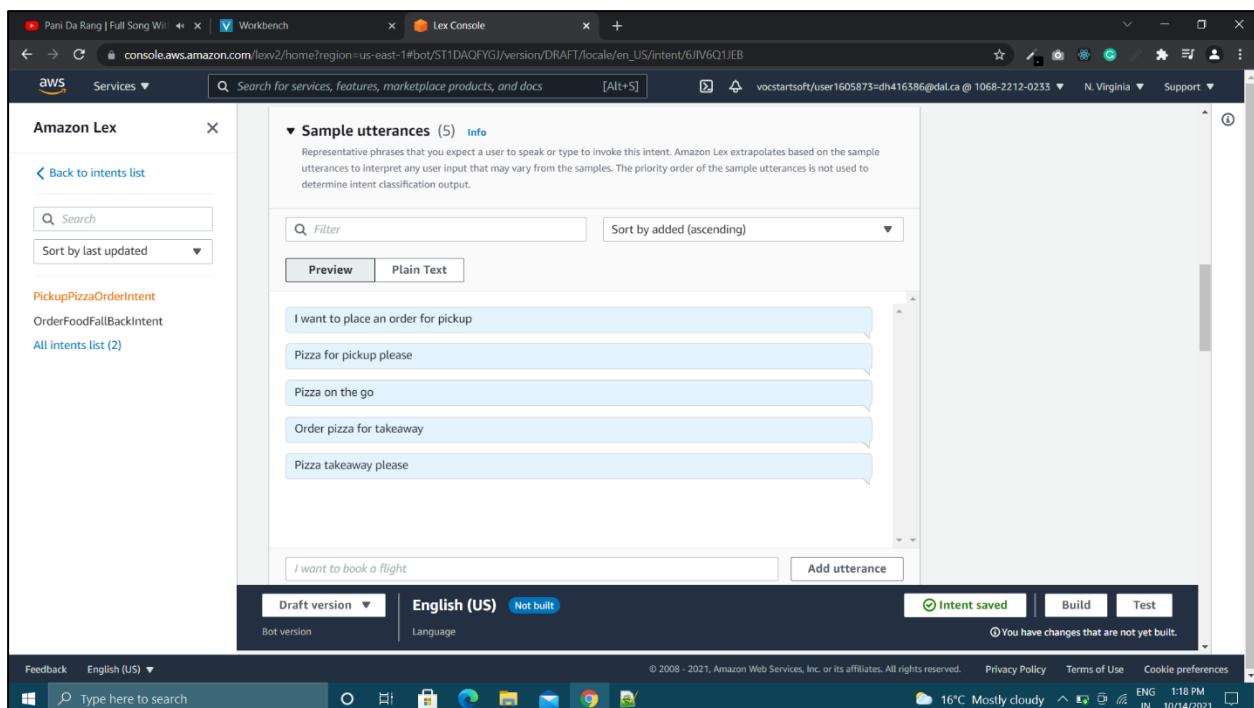


Figure 24 - Pizza pickup utterances that user enters to request pizza pickup [1]

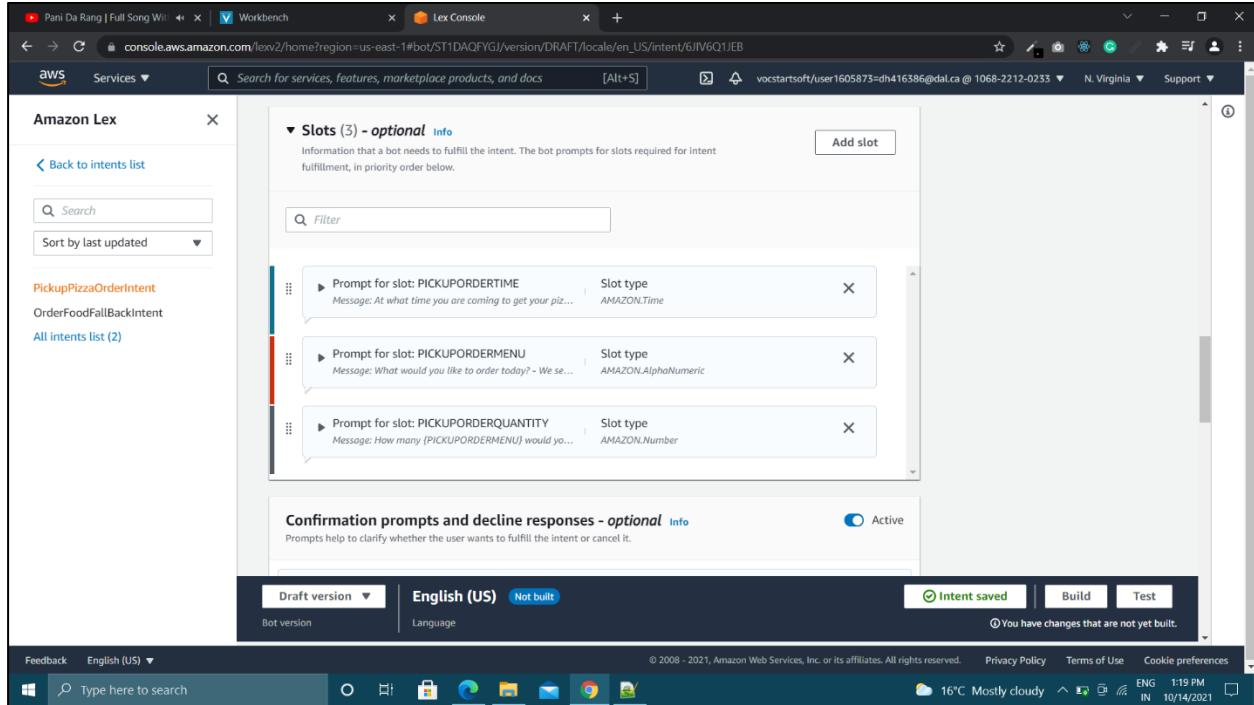


Figure 25 - Pizza pickup slots to accept pizza order, pickup time and quantity of pizza [1]

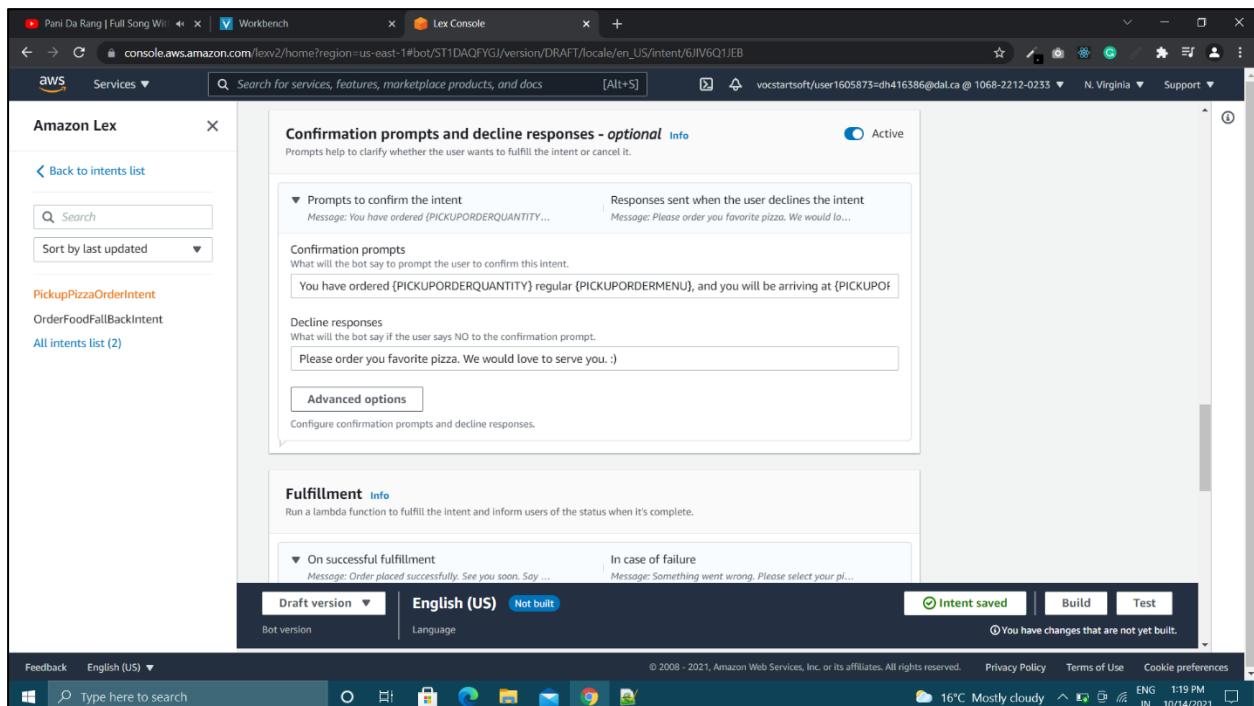


Figure 26 - Pizza pickup confirmation prompt and decline response message [1]

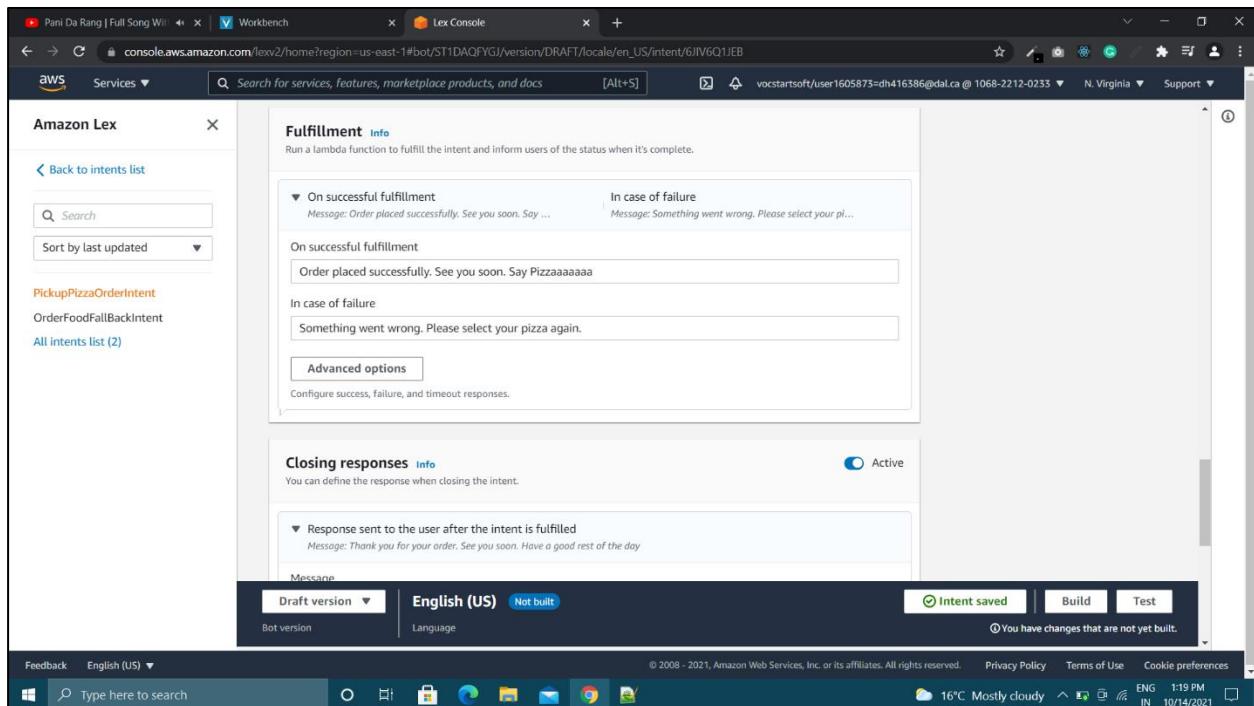


Figure 27 - Pizza pickup fulfillment message [1]

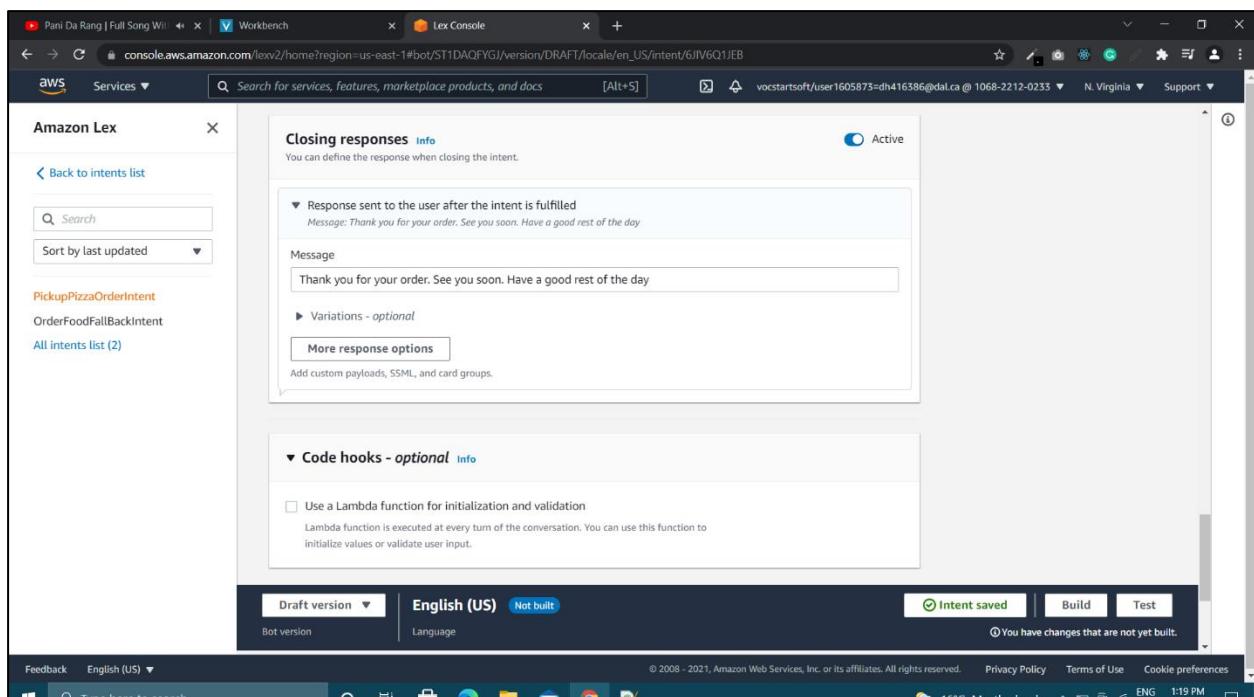


Figure 28 - Pizza pickup closing response [1]

Figures 29, 30, 31, 32, 33, and 34 display the OrderFood pizza pickup intent in action. (i.e., Conversation flow in working)

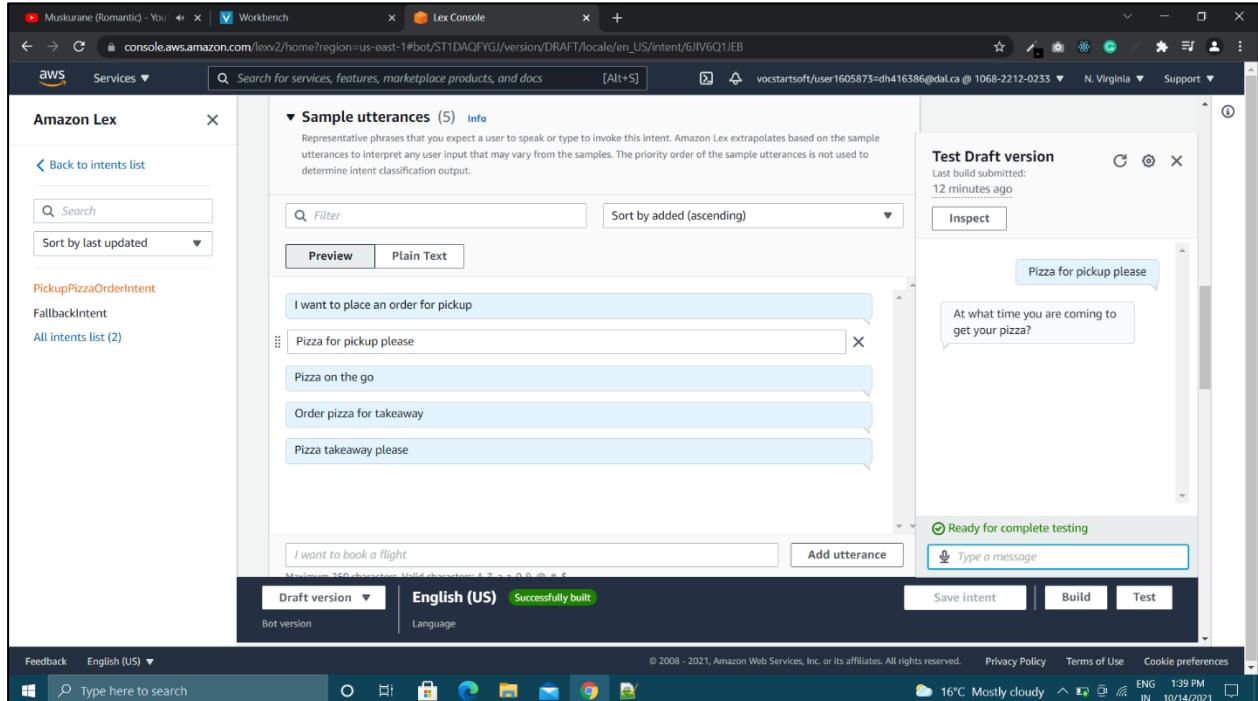


Figure 29 - Pizza pickup chatbot testing conversation in working [1]

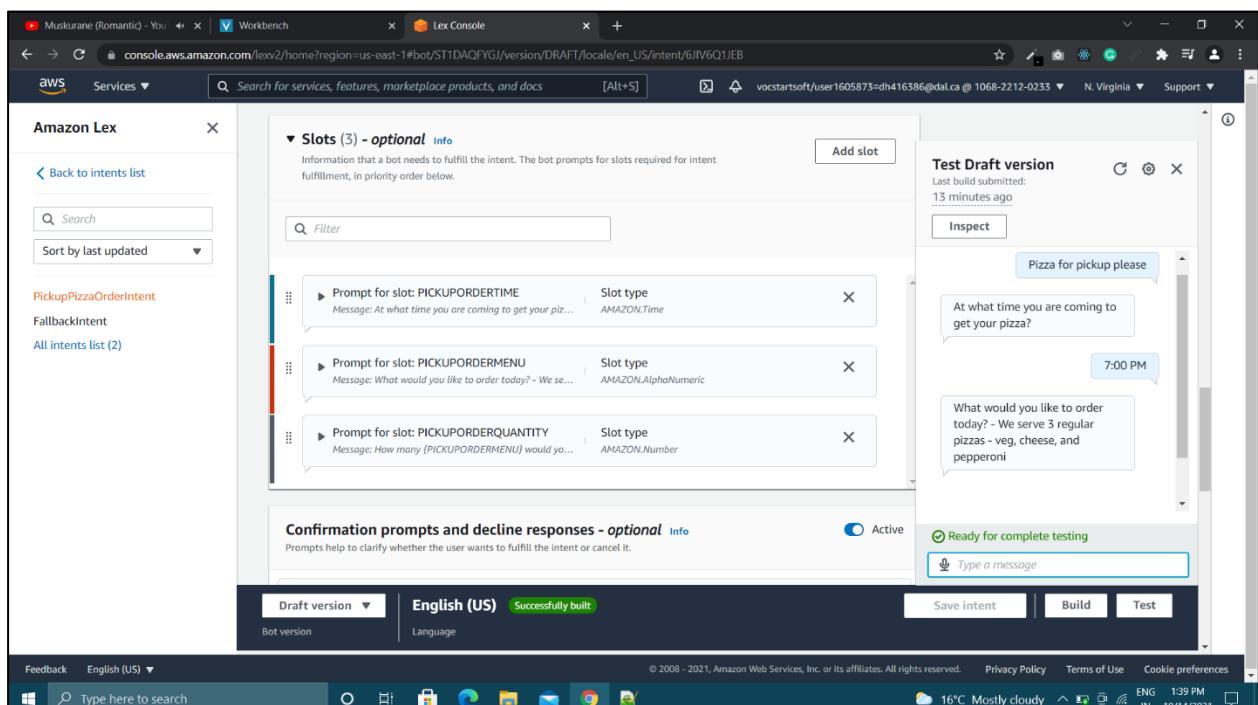


Figure 30 - Pizza pickup chatbot testing conversation in working (contd.) [1]

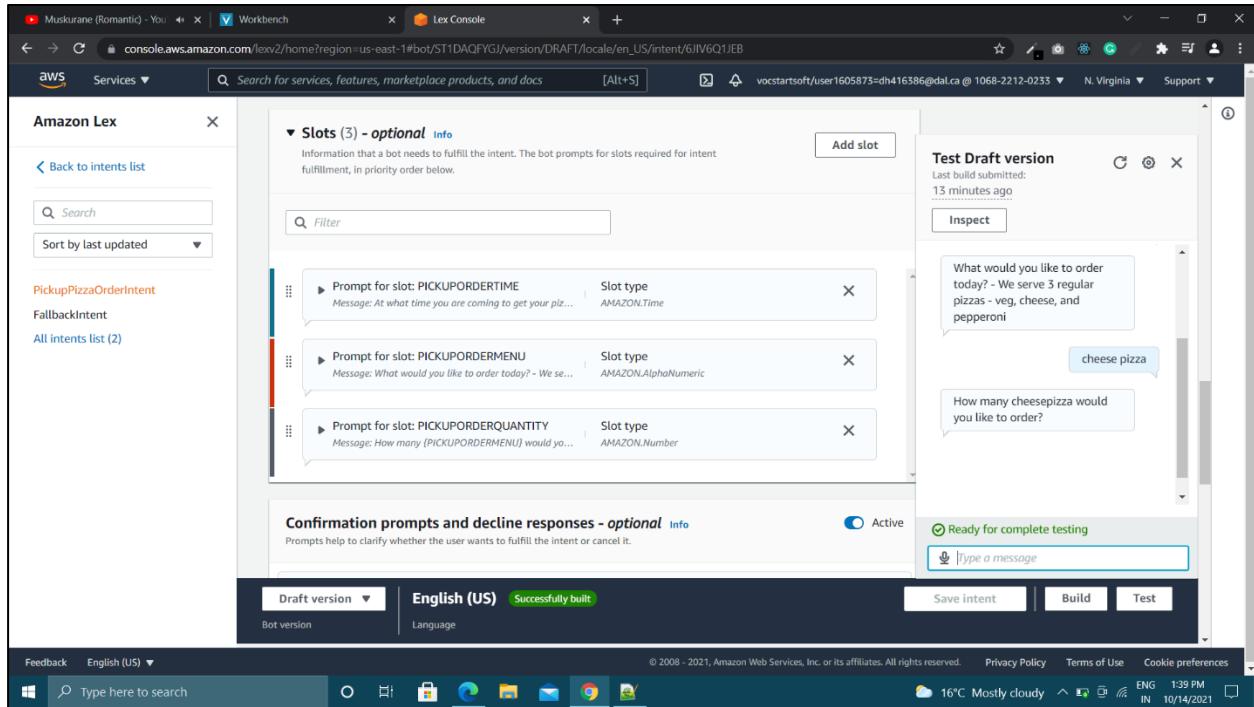


Figure 31 - Pizza pickup chatbot testing conversation in working (contd.) [1]

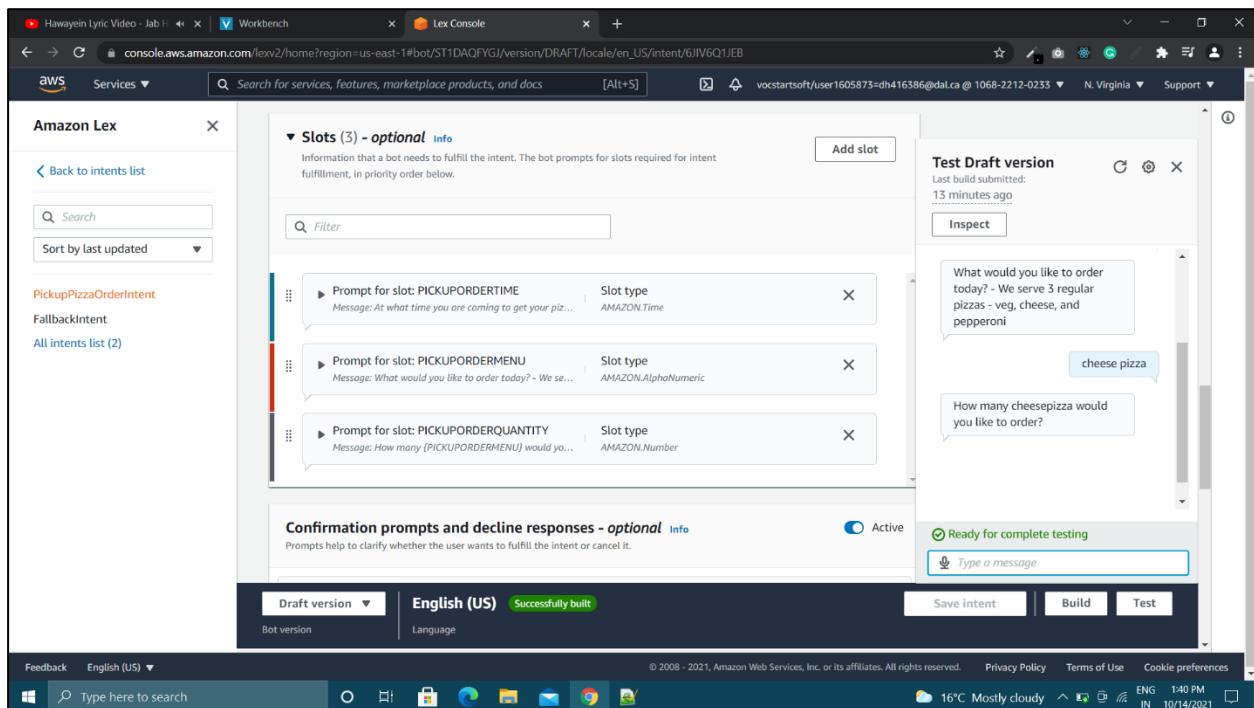


Figure 32 - Pizza pickup chatbot testing conversation in working (contd.) [1]

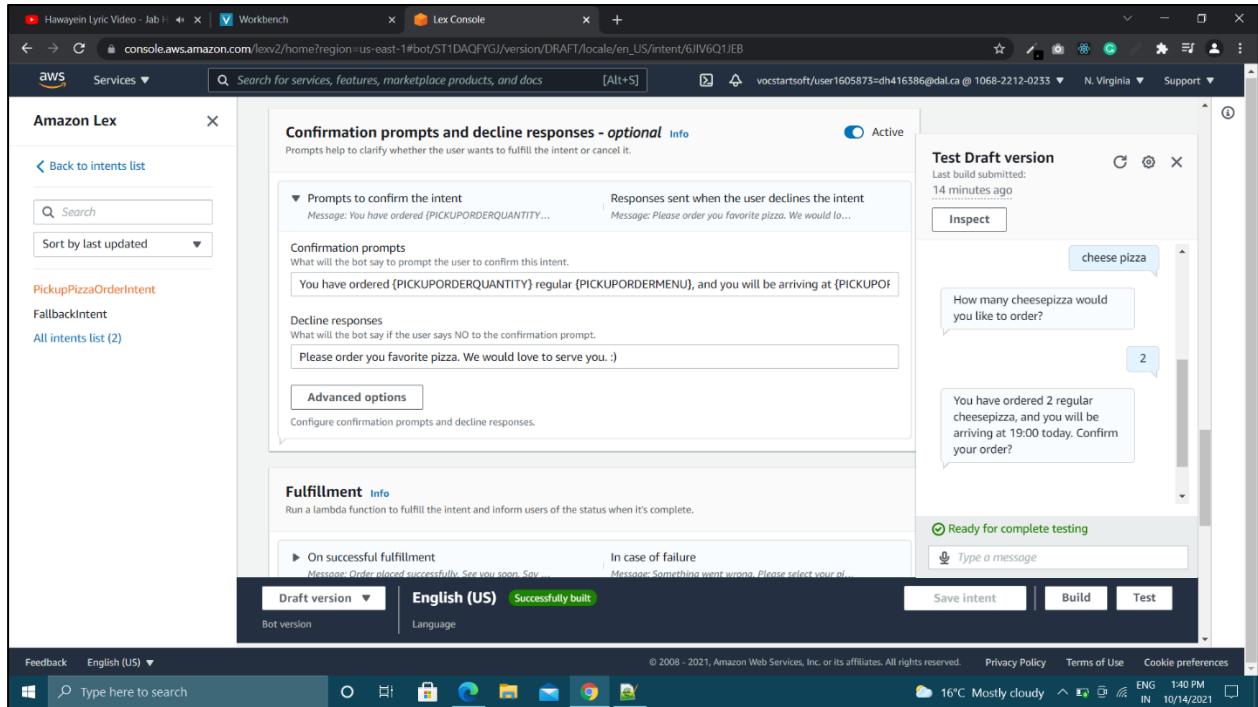


Figure 33 - Pizza pickup chatbot testing conversation in working (contd.) [1]

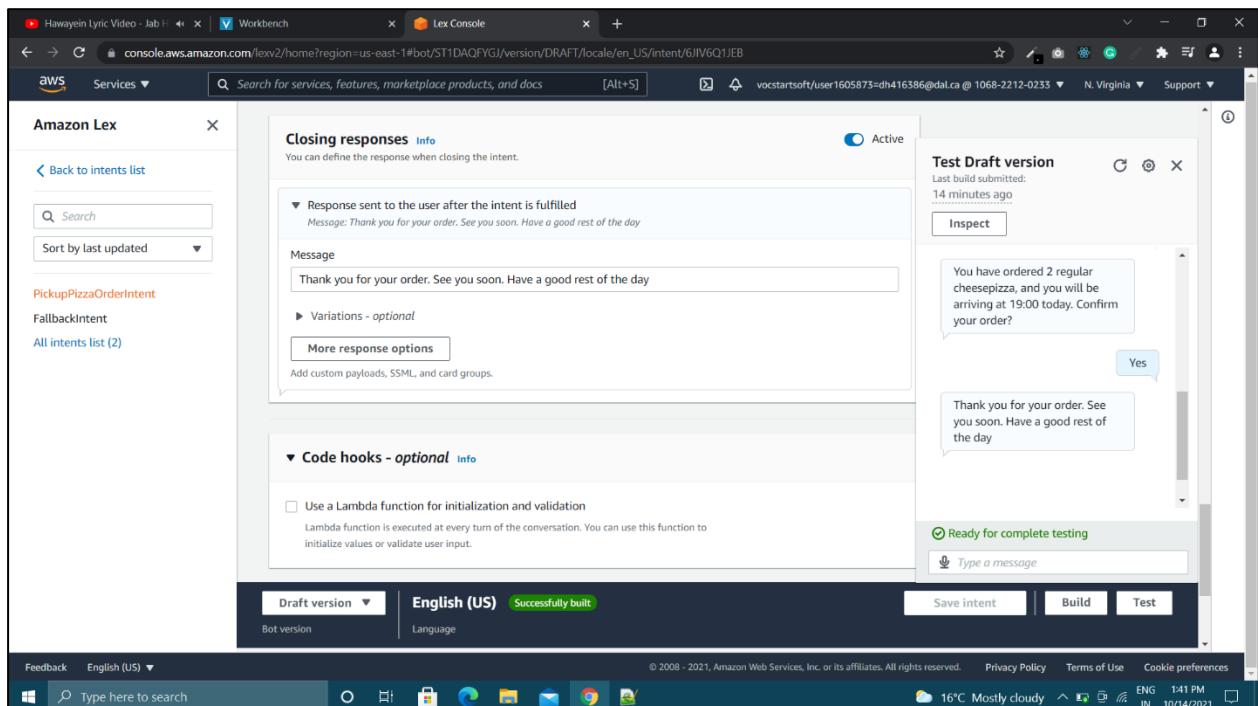


Figure 34 - Pizza pickup chatbot testing conversation in working (contd.) [1]

Fallback Intent Configuration Image

Intent – FallbackIntent

Figure 35 displays the configuration setup of fallback intent.

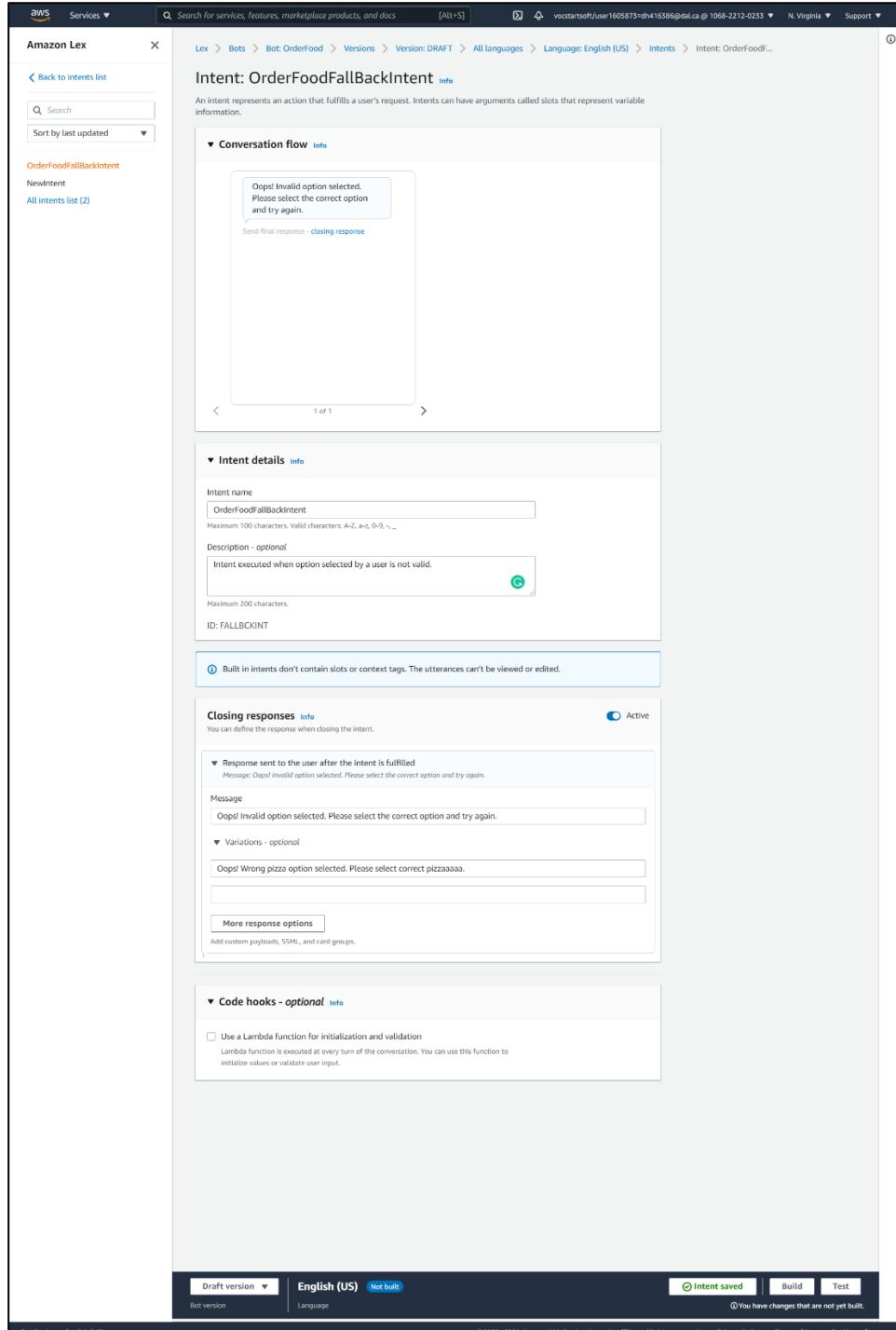


Figure 35 - Fallback intent configuration setup [1]

Summary of Operations Performed

The **OrderFood** chatbot is created using the Amazon Lex service of Amazon Web Services (AWS). This service is typically used to create chatbots that carry out a conversation in an organized manner. The chatbot is created by configuring details such as the name of the chatbot, a description of what it does, IAM permissions being assigned, the language of the bot being set. The chatbot setup involves the creation of intents. For the given exercise, I have created three intents namely DeliveryPizzaIntent, PickupPizzaIntent and FallbackIntent. The intent is used to fulfil tasks like picking up a pizza or getting it delivered in this case. An intent covers all the components utilized for having a conversation in the OrderFood chatbot. It involves various components such as utterances, slots, confirmation messages, fulfillment messages, and closing messages for the OrderFood chatbot. An utterance is the starting point of the conversation which is usually entered by the user to initiate the conversation. In order for the chatbot to know and understand the type of message that can be expected from the user, sample utterances are set for the chatbot. Slots are the responses or queries the chatbot will have for the user. Any query required for fulfilling the intent is covered by the chatbot. It is basically how and what the bot would communicate to user responses or in order to achieve what's intended. Confirmation messages are messages that are messages that the chatbot would display upon asking the user if he wants the intent to be fulfilled or not. Fulfillment message is generally not displayed unless set otherwise. To set a fulfillment message for an intent, certain parameters are set to specify the user of the details of fulfillment. The closing message is displayed after the intent is fulfilled. All these components are set and designed using the Amazon Lex console offered by AWS.

References

- [1] Amazon, "Amazon Lex," Amazon, [Online]. Available: <https://aws.amazon.com/lex/>. [Accessed 14 October 2021].