

Name: Krishna Anish Garg

Email: gargkrishna730@gmail.com

Phone No.: 9152099209

Project Name: Amazon Web Service
Major Project

Project Deadline: 06th February 2023

Problem Statement:

Create a Chatbot using Amazon Lex Tool.

The intent of the chatbot: Book Hotel

1. All the information must be conveyed to the user after booking the room and must be informed to the user of the price of the hotel room and the day of stay.
2. Using this chatbot user must aware of the types of Available rooms (Classic, Duplex, etc.) - Choose your own Category as well.
3. All events must be in flow for the fulfillment of the intent.

Amazon Lex:

1. Amazon Lex is a cloud-based service provided by Amazon Web Services (AWS) that allows developers to build conversational interfaces using voice and text.
2. It uses natural language processing (NLP) to understand and respond to user inputs, making it easier for developers to create chatbots and other conversational applications.
3. Amazon Lex also integrates with other AWS services, such as Lambda and DynamoDB, to provide a complete solution for building and deploying conversational applications.

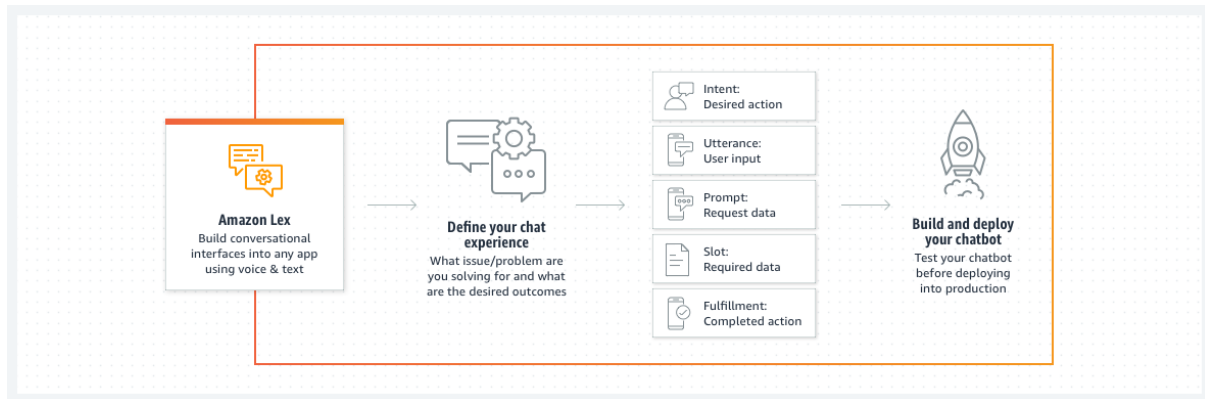
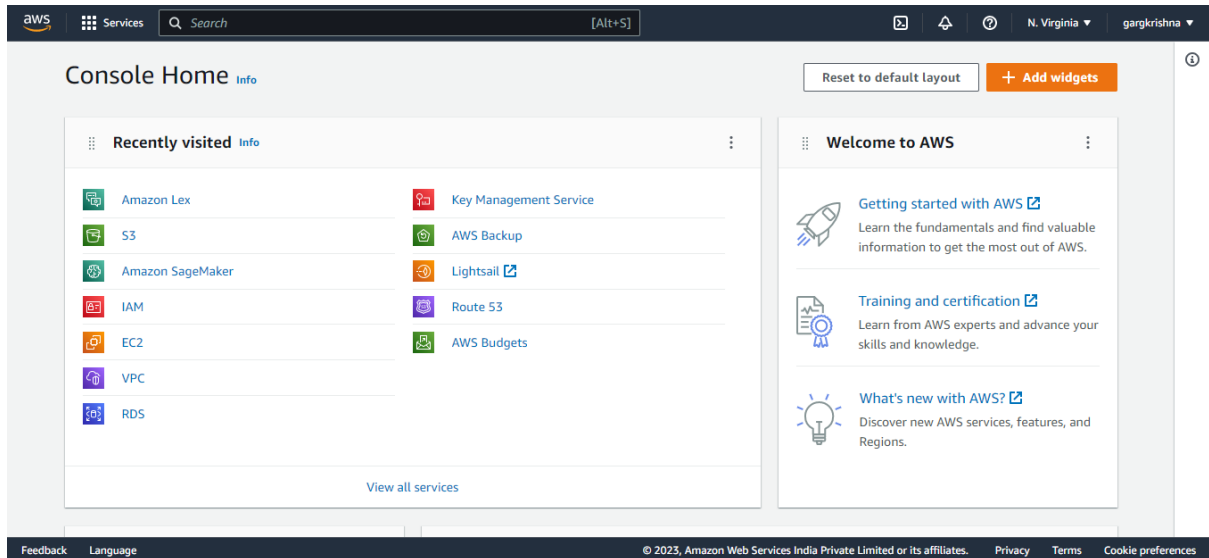


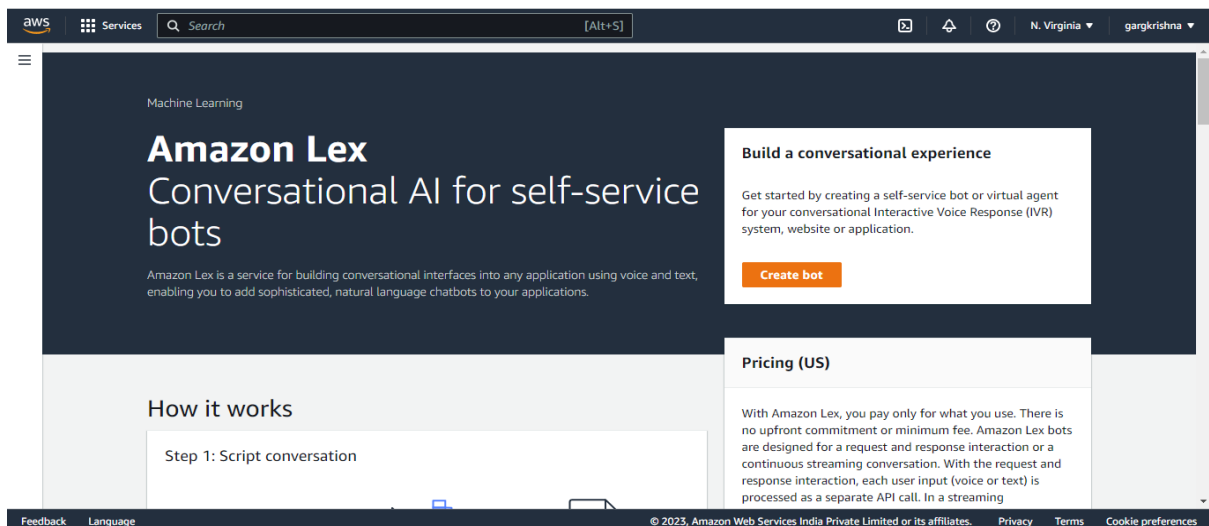
Figure 1: Amazon Lex

Solution:

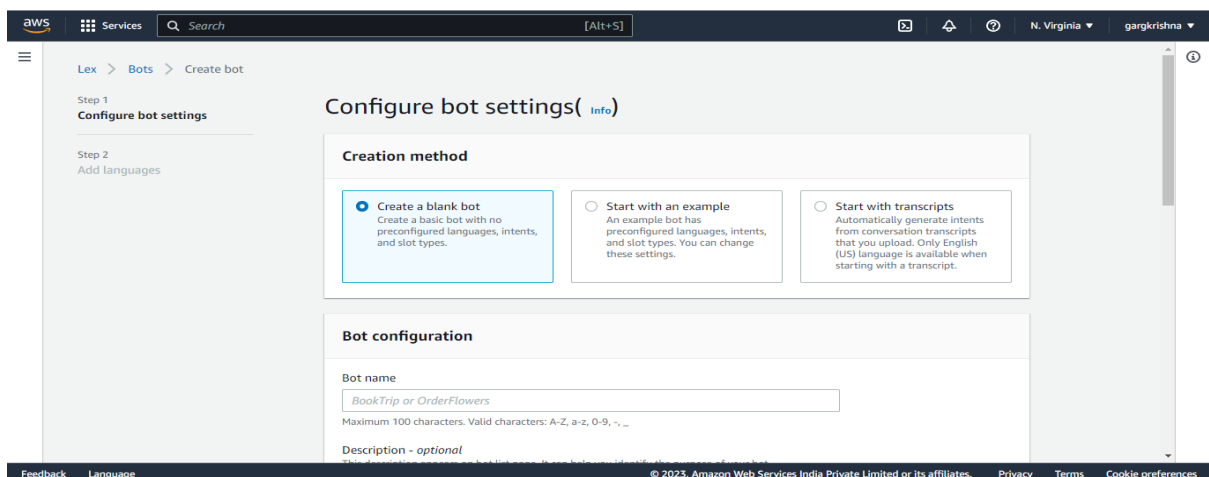
Step 1: Log in to AWS Console:



Step 2: Search for Amazon Lex & go to Amazon Lex Homepage:



Step 3: Click on Create Bot:



Step 4: Select “Create a Blank Bot”:

aws Services Search [Alt+S] N. Virginia gargkrishna

Lex > Bots > Create bot

Step 1
Configure bot settings

Step 2
Add languages

Configure bot settings(info)

Creation method

- ☒ **Create a blank bot**
Create a basic bot with no preconfigured languages, intents, and slot types.
- ☐ **Start with an example**
An example bot has preconfigured languages, intents, and slot types. You can change these settings.
- ☐ **Start with transcripts**
Automatically generate intents from conversation transcripts that you upload. Only English (US) language is available when starting with a transcript.

Bot configuration

Bot name

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

Description - optional

Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 5: Configure a Bot:

Give a name to the Bot and a Description.

aws Services Search [Alt+S] N. Virginia gargkrishna

Lex > Bots > Create bot

Step 1
Configure bot settings

Step 2
Add languages

Configure bot settings(info)

Creation method

- ☒ **Create a blank bot**
Create a basic bot with no preconfigured languages, intents, and slot types.
- ☐ **Start with an example**
An example bot has preconfigured languages, intents, and slot types. You can change these settings.
- ☐ **Start with transcripts**
Automatically generate intents from conversation transcripts that you upload. Only English (US) language is available when starting with a transcript.

Bot configuration

Bot name

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

Description - optional

Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 6:

On IAM permissions, configure the runtime role

Create a role with basic Amazon Lex permissions.

The screenshot shows the AWS IAM console 'IAM permissions' page. The 'Runtime role' section is active, showing the option 'Create a role with basic Amazon Lex permissions' selected. A new role is being created with the name 'AWSServiceRoleForLexV2Bots_NGPAKOGMFM'. The 'Children's Online Privacy Protection Act (COPPA)' section shows 'No' selected for 'Is use of your bot subject to the Children's Online Privacy Protection Act (COPPA)'.

Step 7: Select “No” on Children’s Online Privacy Protection Act (COPPA) as we must specify whether your use of Amazon Lex is related to a website, program, or other application that is directed or targeted, in whole or in part, to children under age 13 and subject to the Children's Online Privacy Protection Act (COPPA).

The screenshot shows the AWS IAM console 'Children's Online Privacy Protection Act (COPPA)' page. The 'Is use of your bot subject to the Children's Online Privacy Protection Act (COPPA)' section shows 'No' selected. The 'Idle session timeout' section shows a session timeout of 5 minutes. The 'Advanced settings - optional' section is expanded, showing 'Tags - bot'.

Step 8: Configure the Idle Session Timeout.

We 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.

The screenshot shows the Amazon Lex console interface. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and user information. The main content area is titled 'Children's Online Privacy Protection Act (COPPA)' with an 'Info' link. Below this, there's a section for 'Idle session timeout' with a description: '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.' The 'Session timeout' is set to '5' minutes. A note states: 'By default, session duration is 5 minutes, but you can specify any duration between 1 and 1440 minutes (24 hours).' At the bottom of the configuration area, there's a link for 'Advanced settings - optional' with an 'Info' link. The bottom of the console shows 'Cancel' and 'Next' buttons.

Step 9: Click on the Next:

The screenshot shows the Amazon Lex console interface, specifically the 'Advanced settings - optional' section. It contains two main sections: 'Tags - bot' and 'Tags - testBotAlias'. The 'Tags - bot' section has a table with columns 'Key' and 'Value - required'. It shows a tag with 'Project' as the key and 'Hotel Booking' as the value. There's an 'Add new tag' button and a note: 'You can add 49 more tags.' The 'Tags - testBotAlias' section has a note: '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.' It also has an 'Add new tag' button and a note: 'You can add 50 more tags.' At the bottom of the configuration area, there's a 'Cancel' button and a 'Next' button.

Step 10: Add Language to the Bot:

The screenshot shows the 'Add language to bot' page in the AWS Lex console. The breadcrumb navigation is 'Lex > Bots > Create bot'. On the left, 'Step 1 Configure bot settings' is active, and 'Step 2 Add languages' is the next step. The main form is titled 'Add language to bot' with an 'Info' link. It contains the following fields:

- Language:** English (US) (selected from a dropdown)
- Description - optional:** A text area with a character count of 0/200.
- Voice interaction:** The text-to-speech voice that your bot uses to interact with users. Selected: Ivy.
- Voice sample:** A text input with the sample text 'Hello, my name is Ivy. Let me know how I can assist you.' and a 'Play' button.
- Intent classification confidence score threshold:** A text input with the value '0.40' and a note 'Min: 0.00, max: 1.00'.

At the bottom of the form are three buttons: 'Cancel', 'Add another language', and 'Done'.

Step 11: Click on Done and a Bot has been created:

This screenshot shows the 'Add language to bot' page with 'Hindi (IN)' selected. The fields are populated with Hindi-specific information:

- Language:** Hindi (IN)
- Description - optional:** Hindi Language for Indian Users
- Voice interaction:** Selected: Kajal
- Voice sample:** Sample text in Hindi: 'नमस्ते, मेरा नाम Kajal है। मुझे बताएं कि मैं आपकी कैसे सहायता कर सकती हूँ।' with a 'Play' button.
- Intent classification confidence score threshold:** 0.40

The 'Done' button is now highlighted in orange, indicating it is the primary action.

Step 12: A bot named “HotelBooking” has been created successfully:

The screenshot shows the 'Bots' page in the AWS Lex console. A green banner at the top reads 'Successfully created bot: HotelBooking'. The left sidebar shows 'Amazon Lex' and 'Bots'.

The main content area shows a table of bots:

Name	Description	Status	Latest Version	Last updated
HotelBooking	-	Available	-	1 minute ago

Below the table is the 'Import/export history' section, which is currently empty with the message 'No import/export records found'.

Step 13: Click on “HotelBooking” and configure the intent details:

The screenshot shows the Amazon Lex console interface. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and user information. The main header indicates 'Amazon Lex' and 'Draft version'. Below this, there's a sidebar with a 'Back to intents list (2)' link and a search bar. The main content area is titled 'Intent details' and shows the 'HotelBooking' intent. The 'Intent name' field is set to 'HotelBooking'. The 'Description - optional' field is empty. The 'ID' is 'TVHKUBVDNJ'. There are buttons for 'Editor', 'Visual builder', and 'Save intent'. The footer contains 'Feedback', 'Language', and copyright information.

Amazon Lex

Back to intents list (2)

Search

Sort by last updated

NewIntent Unsaved

FallbackIntent

Intent details

Intent name

HotelBooking

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

Description - optional

Maximum 200 characters.

ID: TVHKUBVDNJ

Contexts - optional

Editor Visual builder New

Save intent

Feedback Language

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 14: Add Sample Utterances related to Hotel Booking:

The screenshot shows the Amazon Lex console interface. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and user information. The main header indicates 'Amazon Lex' and 'Draft version'. Below this, there's a sidebar with a 'Back to intents list (2)' link and a search bar. The main content area is titled 'Sample utterances (15)' and shows a list of sample utterances. The 'Filter' field is empty. The 'Sort by added (ascending)' dropdown is selected. The list of utterances includes: 'Reserve a room for two people', 'Book a room for next weekend', 'Need a hotel for my trip', 'Find a hotel near me', and 'Find a hotel in New York'. There are buttons for 'Preview', 'Plain text', 'Editor', 'Visual builder', and 'Save intent'. The footer contains 'Feedback', 'Language', and copyright information.

Amazon Lex

Back to intents list (2)

Search

Sort by last updated

NewIntent Unsaved

FallbackIntent

Sample utterances (15)

Representative phrases that you expect a user to speak or type to invoke this intent. Amazon Lex extrapolates based on the sample utterances to interpret any user input that may vary from the samples. The priority order of the sample utterances is not used to determine intent classification output.

Filter

Sort by added (ascending)

Preview Plain text

Reserve a room for two people

Book a room for next weekend

Need a hotel for my trip

Find a hotel near me

Find a hotel in New York

Editor Visual builder New

Save intent

Feedback Language

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 15: Configure Initial Response:

The screenshot shows the Amazon Lex console interface. At the top, there's a navigation bar with the AWS logo, 'Services', a search bar, and user information. The main header indicates 'Amazon Lex' and 'Draft version'. Below this, there's a sidebar with a 'Back to intents list (2)' link and a search bar. The main content area is titled 'Initial response' and shows the configuration for the initial response. The 'Response to acknowledge the user's request' is set to 'Message: Okay, I can help you with that'. The 'Message group' is set to 'Message - optional' with the message 'Okay, I can help you with that'. There are buttons for 'Advanced options', 'Editor', 'Visual builder', and 'Save intent'. The footer contains 'Feedback', 'Language', and copyright information.

Amazon Lex

Back to intents list (2)

Search

Sort by last updated

NewIntent Unsaved

FallbackIntent

Initial response

You can provide messages to acknowledge the user's initial request. You can also configure next step in the conversation and branch based on conditions.

Response to acknowledge the user's request

Message: Okay, I can help you with that

Message group

You can define a text message group to respond using plain text.

Message - optional

Okay, I can help you with that

Variations - optional

Advanced options

Configure user request acknowledgement response, dialog code hook and conditional branches.

Editor Visual builder New

Save intent

Feedback Language

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Step 16: Configure Slots:

For this Project, there are 7 slot types of which 3 slot types are custom slot types.

The screenshot shows the Amazon Lex console interface. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information. The main content area is titled 'Successfully built language English (US) in bot: HotelBooking'. It features a 'Draft version' dropdown, 'English (US)' language selection, and 'Build' and 'Test' buttons. Below this, the 'Slots (8) - optional' section is visible, with an 'Add slot' button. A list of slots is shown, including 'Prompt for slot: Name' (Slot type: AMAZON.FirstName) and 'Prompt for slot: Location' (Slot type: location). The bottom of the console shows a footer with '© 2023, Amazon Web Services India Private Limited or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot also shows the 'Slot types (3)' section, which contains a table of custom slot types:

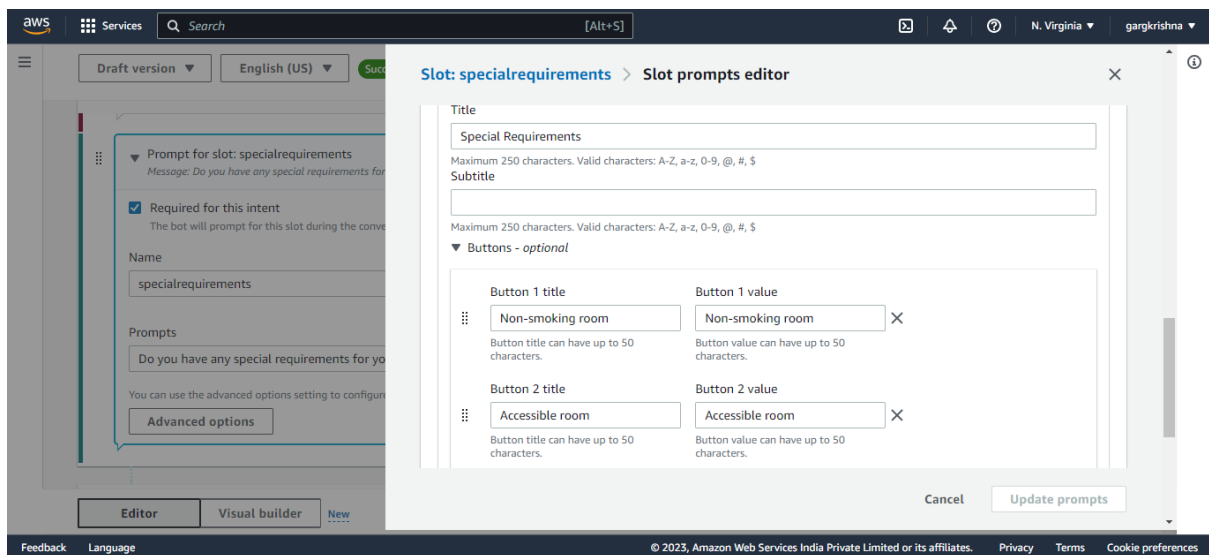
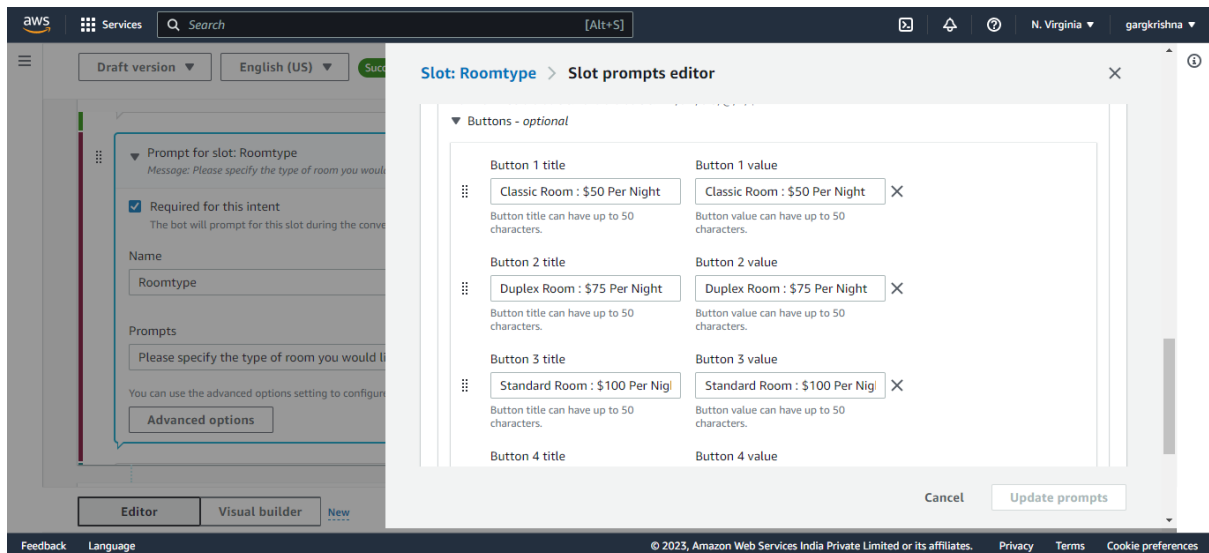
Name	Description	Type	Last edited
RoomType	-	Custom	2 hours ago
specialrequirements	-	Custom	2 hours ago
location	-	Custom	2 hours ago

Step 17: Adding Buttons in RoomType and specialrequirements slots:

The screenshot shows the Amazon Lex console interface. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information. The main content area is titled 'Slot: Roomtype > Slot prompts editor'. It features a 'Draft version' dropdown, 'English (US)' language selection, and 'Build' and 'Test' buttons. Below this, the 'Slot prompts editor' is visible, showing a 'Prompt for slot: Roomtype' with a message: 'Please specify the type of room you would like to book'. The 'Required for this intent' checkbox is checked. The 'Prompts' section shows the prompt: 'Please specify the type of room you would like to book'. The 'Advanced options' section is also visible. The bottom of the console shows a footer with '© 2023, Amazon Web Services India Private Limited or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot also shows the 'Card group' configuration for the 'Roomtype' slot. It includes a 'Card 1' configuration with the following details:

- Image URL: <http://www.example.com/image.png>
- Title: Which Room would you like to book {Name}?
- Subtitle: Maximum 250 characters. Valid characters: A-Z, a-z, 0-9, @, #, \$



Steps to add button to the slot:

1. Select the slot and click on Advanced Options.
2. Go to slot prompt and click on “*Bot elicits information*”
3. Click on More prompt options
4. Go to Slot Prompt & click on Add and then Click Add Card Group
5. Define the title to the Card Group.
6. There will be an option on the bottom which states buttons-optional. Click on button=optional and enter the button with suitable title and button value.
7. Click on Update Prompt.

Step 18: Configuring Confirmation:

The screenshot shows the Amazon Lex console interface for configuring the 'Confirmation' step of an intent. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information (N. Virginia, gargkrishna). The left sidebar shows the 'Amazon Lex' console with a search bar and a list of intents: 'HotelBooking' (selected) and 'FallbackIntent'. The main content area is titled 'Draft version' and 'English (US)', with a 'Successfully built' status. It features 'Build' and 'Test' buttons. The 'Prompts to confirm the intent' section includes a message: 'Can I go ahead with your request, {Name}?' and a 'Responses sent when the user declines the intent' section with a message: 'Okay. Your response will not be submitted.' Below this is a 'Confirmation prompt' section with a message: 'Can I go ahead with your request, {Name}?' and a 'Decline response' section with a message: 'Okay. Your response will not be submitted.' There is an 'Advanced options' button and a 'Fulfillment' section with a toggle switch set to 'Active'. The 'Fulfillment' section includes a description: 'Run a lambda function to fulfill the intent and inform users of the status when it's complete.' and buttons for 'Editor', 'Visual builder', and 'New'. A 'Save intent' button is at the bottom right. The footer contains 'Feedback', 'Language', and copyright information.

Step 19: Configuring Fulfillment:

The screenshot shows the Amazon Lex console interface for configuring the 'Fulfillment' step of an intent. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information (N. Virginia, gargkrishna). The left sidebar shows the 'Amazon Lex' console with a search bar and a list of intents: 'HotelBooking' (selected) and 'FallbackIntent'. The main content area is titled 'Draft version' and 'English (US)', with a 'Successfully built' status. It features 'Build' and 'Test' buttons. The 'Fulfillment' section is active, showing a description: 'Run a lambda function to fulfill the intent and inform users of the status when it's complete.' Below this is a section for 'On successful fulfillment' with a message: 'Your booking is confirmed! The details of your reservation are Name: {Name}, Location: {Location}, Check-in date: {Check-in date}.' and a section for 'In case of failure' with a message: 'Something went wrong'. There is an 'Advanced options' button and a 'Save intent' button at the bottom right. The footer contains 'Feedback', 'Language', and copyright information.

Step 20: Configuring Closing Response:

The screenshot shows the Amazon Lex console interface for configuring the 'Closing response' step of an intent. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information (N. Virginia, gargkrishna). The left sidebar shows the 'Amazon Lex' console with a search bar and a list of intents: 'HotelBooking' (selected) and 'FallbackIntent'. The main content area is titled 'Draft version' and 'English (US)', with a 'Successfully built' status. It features 'Build' and 'Test' buttons. The 'Closing response' section is active, showing a description: 'You can define the response when closing the intent.' Below this is a section for 'Response sent to the user after the intent is fulfilled' with a message: 'Thank you {Name} for choosing us! Your booking is confirmed. We are looking forward to welcoming you to our hotel. If you have any further requirements, please don't hesitate to reach out to us. Have a great day!'. There is a 'Message group' section with a message: 'Thank you {Name} for choosing us! Your booking is confirmed. We are looking forward to welcoming you to our ho'. There is a 'Variations - optional' section with a 'More response options' button and a 'Save intent' button at the bottom right. The footer contains 'Feedback', 'Language', and copyright information.

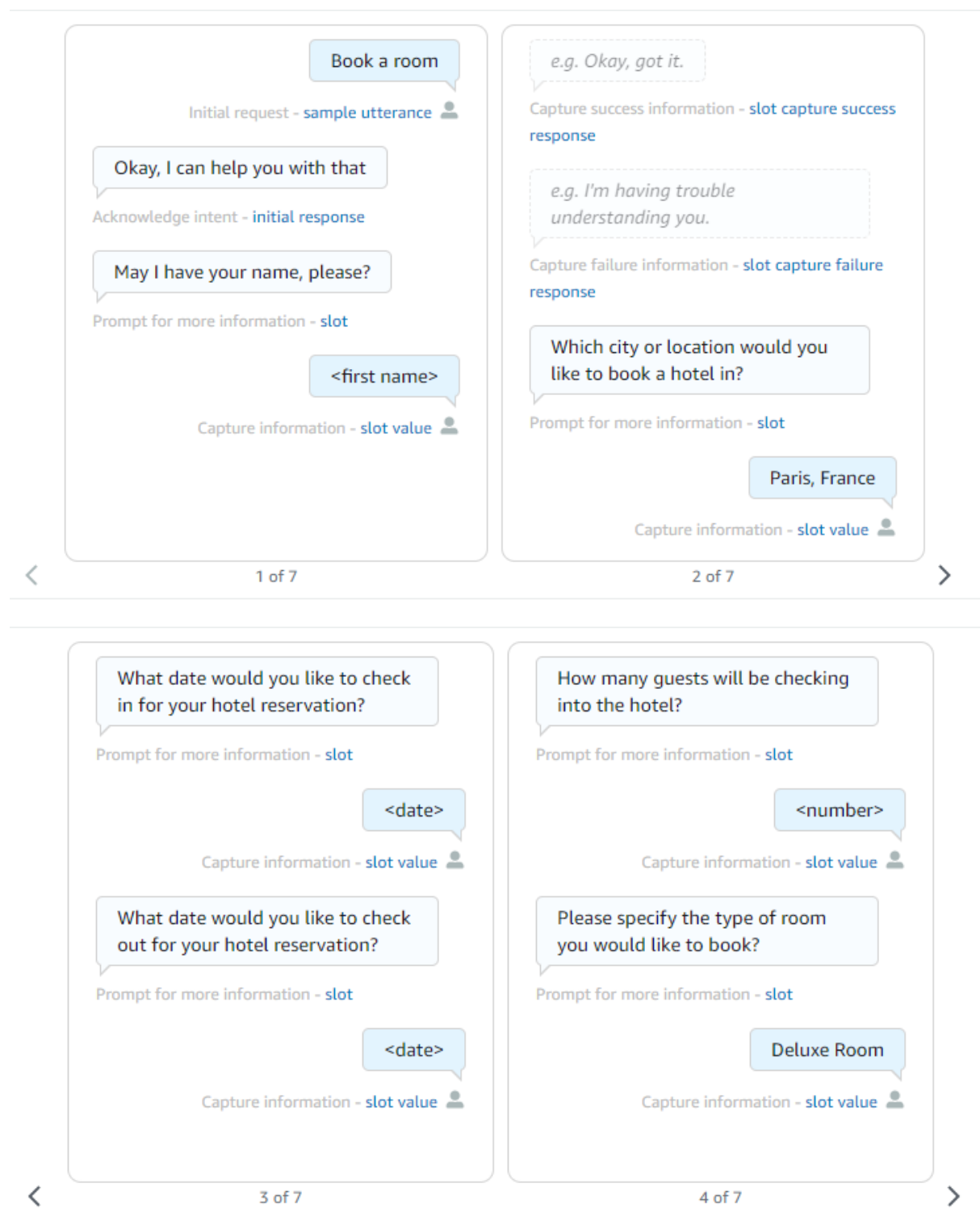
Step 21: Click on Build:

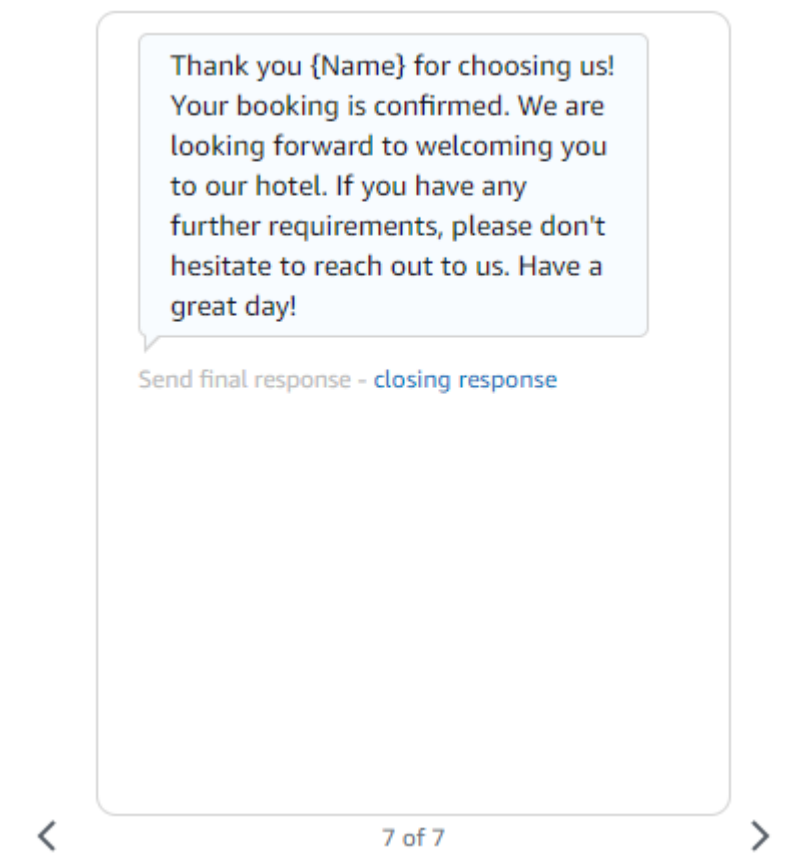
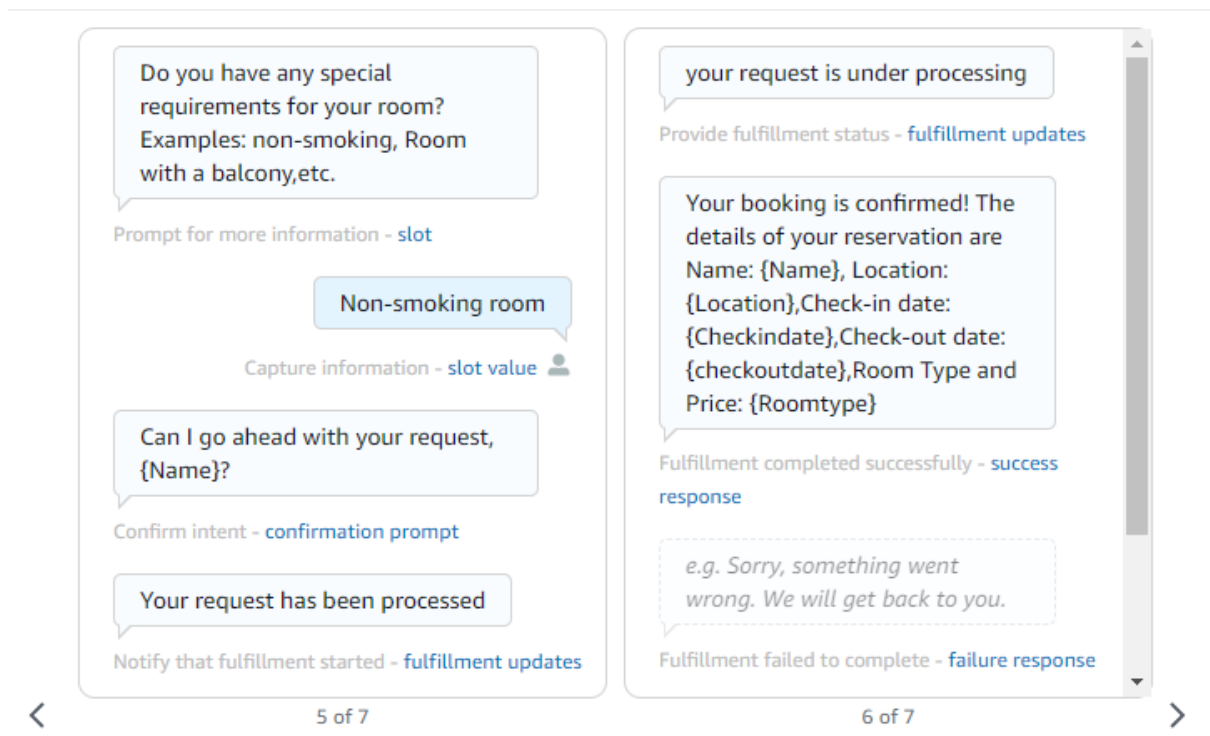
The screenshot shows the Amazon Lex console interface. On the left, there's a sidebar with 'Amazon Lex' and a list of intents: 'HotelBooking' (highlighted in orange) and 'FallbackIntent'. The main area displays the configuration for the 'HotelBooking' intent in English (US). At the top, a blue banner indicates 'Building language English (US) in bot: HotelBooking. If your language contains external source slot types, the build might take longer to complete.' Below this, there are buttons for 'Draft version', 'English (US)', and 'Building'. To the right of these are 'Build' and 'Test' buttons. The configuration area shows an initial response and a slot named 'Slots (7) - optional'. At the bottom, there are buttons for 'Editor', 'Visual builder', and 'New'. A footer bar contains 'Feedback', 'Language', and copyright information.

Step 22: Successfully Build a Bot:

The screenshot shows the Amazon Lex console interface after the bot has been successfully built. The main area now displays a green banner at the top stating 'Successfully built language English (US) in bot: HotelBooking'. The 'Building' button has been replaced by a green 'Successfully built' button. The 'Build' and 'Test' buttons remain. The configuration area for the 'HotelBooking' intent is still visible, showing the initial response and the 'Slots (7) - optional' slot. The footer bar is the same as in the previous screenshot.




Conversation flow for the Bot:





Bot Testing:

Name slot Executed Successfully.

Test Draft version   
Last build submitted: 5 minutes ago

Inspect


Book a room

Okay, I can help you with that




May I have your name, please?

Krishna Garg

Which city or location would you like to book a hotel in?

 Type a message

Location Slot executed Successfully.

Test Draft version   
Last build submitted: 7 minutes ago


Inspect

Krishna Garg

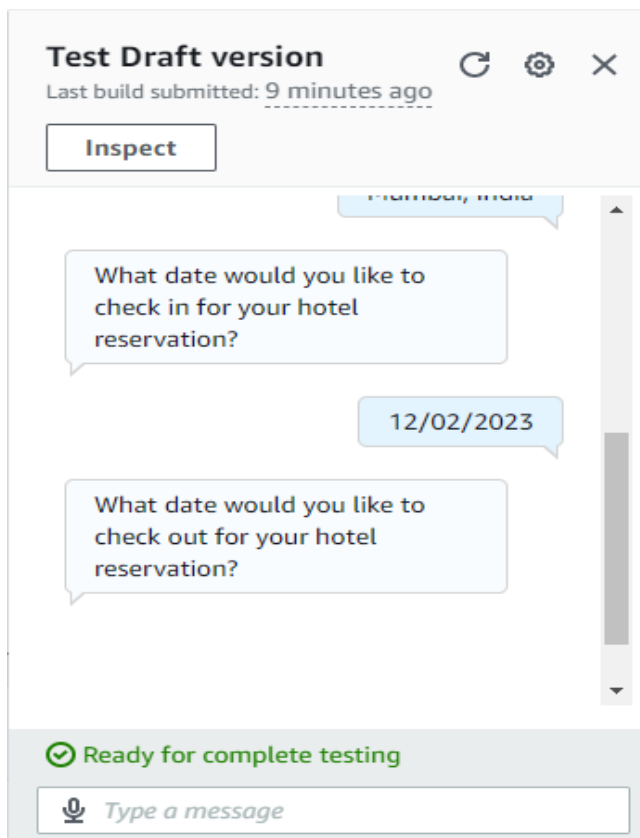
Which city or location would you like to book a hotel in?

Mumbai, India

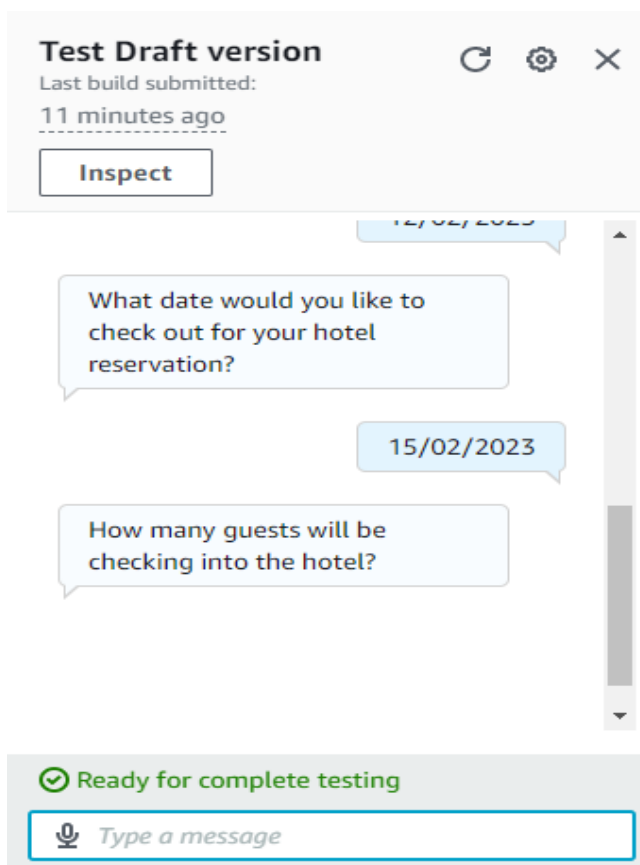
What date would you like to check in for your hotel reservation?

 Type a message

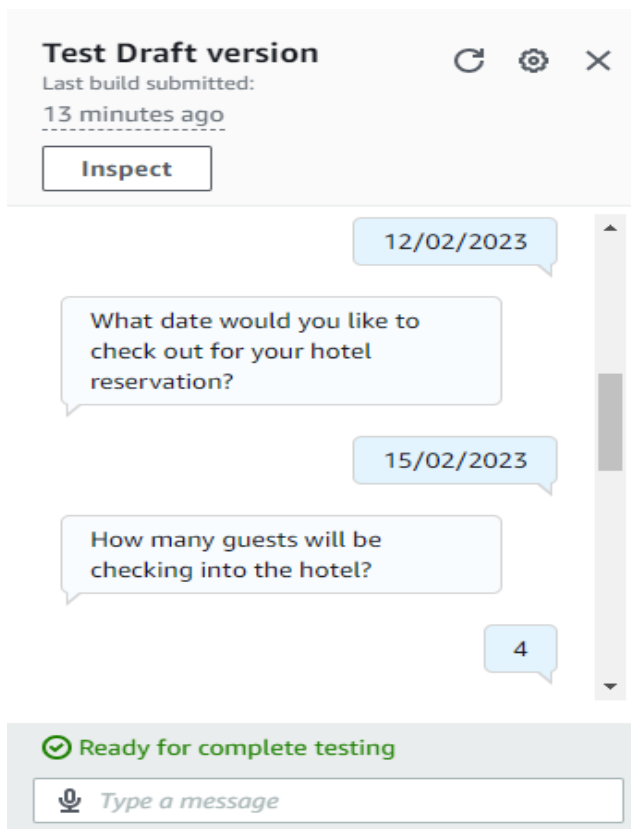
Checkindate Slot executed Successfully.



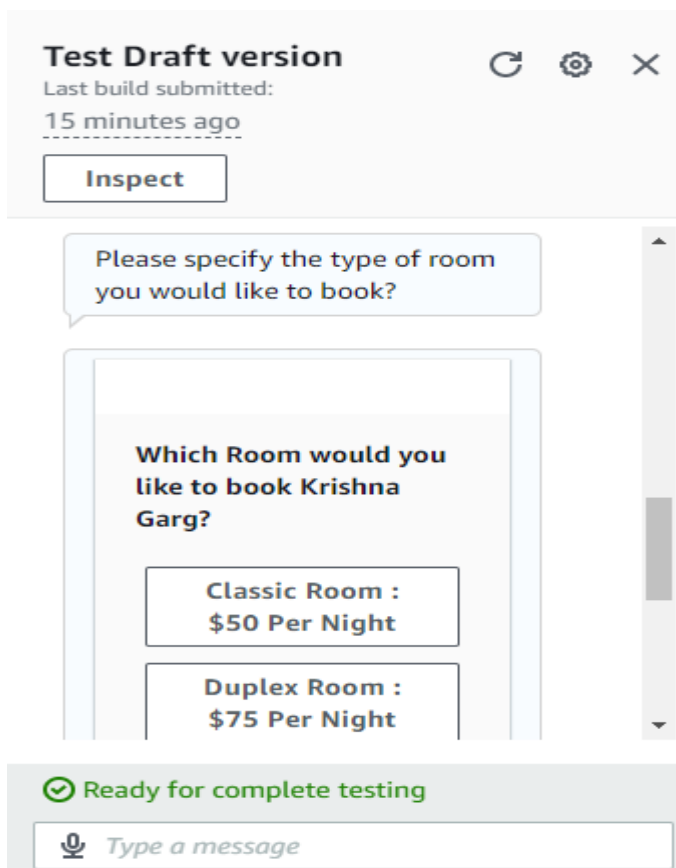
Checkoutdate Slot executed Successfully.



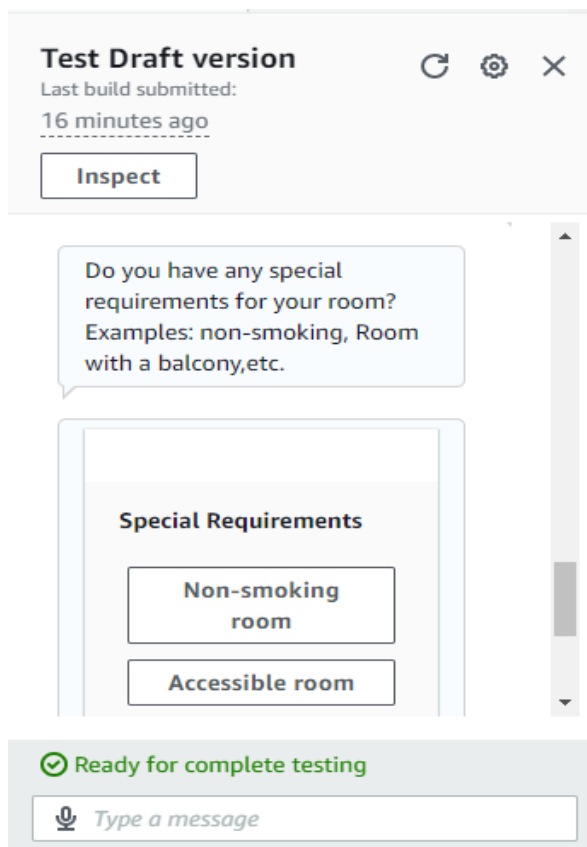
Noofguests Slot executed Successfully.



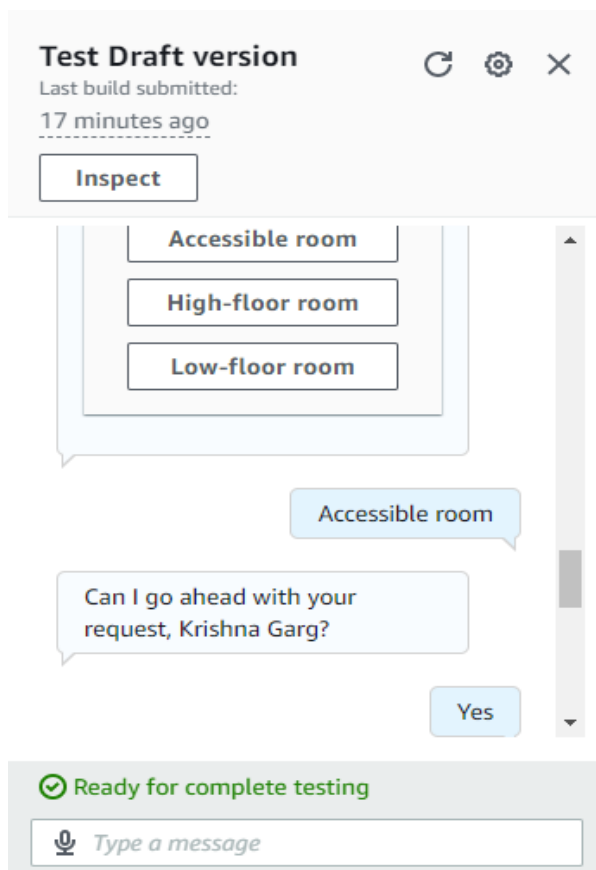
RoomType Slot with buttons executed Successfully.



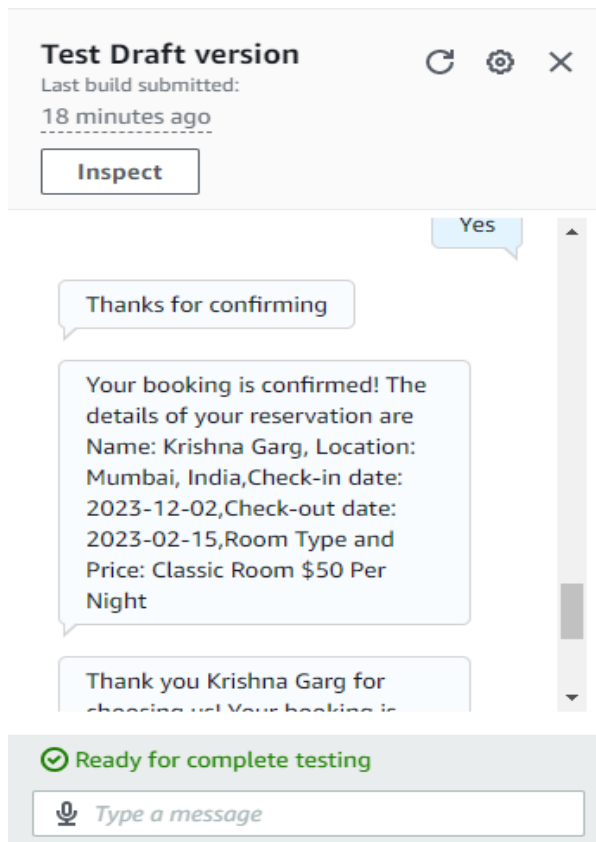
Specialrequirements Slot with buttons executed Successfully:



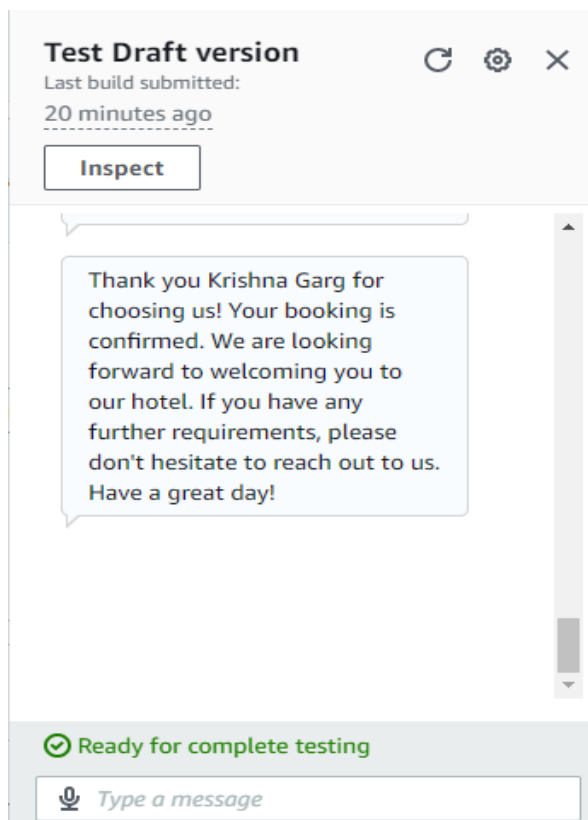
Confirmation Message is displayed.



Booking details with user input information such as Name, Location, Check-in Date, Check-out Date Room Type & Price has been displayed Successfully.



Closing Response has been displayed Successfully:



Conclusion:

In conclusion, Amazon Lex can be a powerful tool for creating a hotel booking chatbot that can assist users in finding and booking hotel rooms. The chatbot can be integrated with a variety of different systems, including databases and reservation systems, to ensure that all relevant information is easily accessible and up-to-date. To build a successful chatbot, it is important to carefully design and define the user flow, including prompts and slots, and implement a well-designed Lambda function for fulfillment. Additionally, regular testing and refinement of the chatbot's functionality will be necessary to ensure that it continues to meet the needs and expectations of users.

Testing Video of Chatbot:

<https://drive.google.com/file/d/1O2xiUfzQzQfryQyic6ai9qV64m-9i3A5/view?usp=sharing>