RideRequest bot created using AWS Lex

To create the RideRequest chatbot using AWS Lex, I followed the following steps:

1. I accessed the AWS account or AWS Academy account and logged in.

2. I navigated to the AWS Lex service in the AWS Management Console.

3. In the Lex console, I clicked on "Create" to start creating a new chatbot.

4. I selected "Custom bot" as the bot type.

5. I provided a name for the bot, such as "RideRequest".

6. I configured the intents and slots according to the requirements for the Taxi and Car rental service.

a. For the "Self-drive" intent, I created sample utterances like "I want to request a self-drive ride."

b. I defined slots for capturing information like name, the type of ride (self-drive or taxi), customer address, pickup date and time, arrival time for self-drive, vehicle type (SUV, Sedan, Minivan), and quantity.

7. I set up prompts to gather the required information from the user. For example, I created prompts like "When are you coming to get your vehicle?" and "What do you want today (SUV, Sedan, Minivan)?"

8. I configured the fulfillment for the intent, where I set up responses to confirm the user's request and provide relevant details.

a. For example, I configured the fulfillment to respond with "You have requested for 1 SUV, and you will be arriving at 12:00 pm" when all the necessary information is collected.

b. I also set up a response to acknowledge the successful placement of the request, such as "Your request has been placed successfully."

9. I customized the chatbot further by adding clarification prompts, confirmation prompts, and error handling.

10. For the Taxi booking, I created an another Intent, with giving inputs for sample utterances as I want to book a taxi, etc..,

11. Then I provided slots and slot type as name, city, date, time. With custom prompts for each slots. In this section I have used aws provided slot type such as AMAZON.city, AMAZON.date, AMAZON.time.

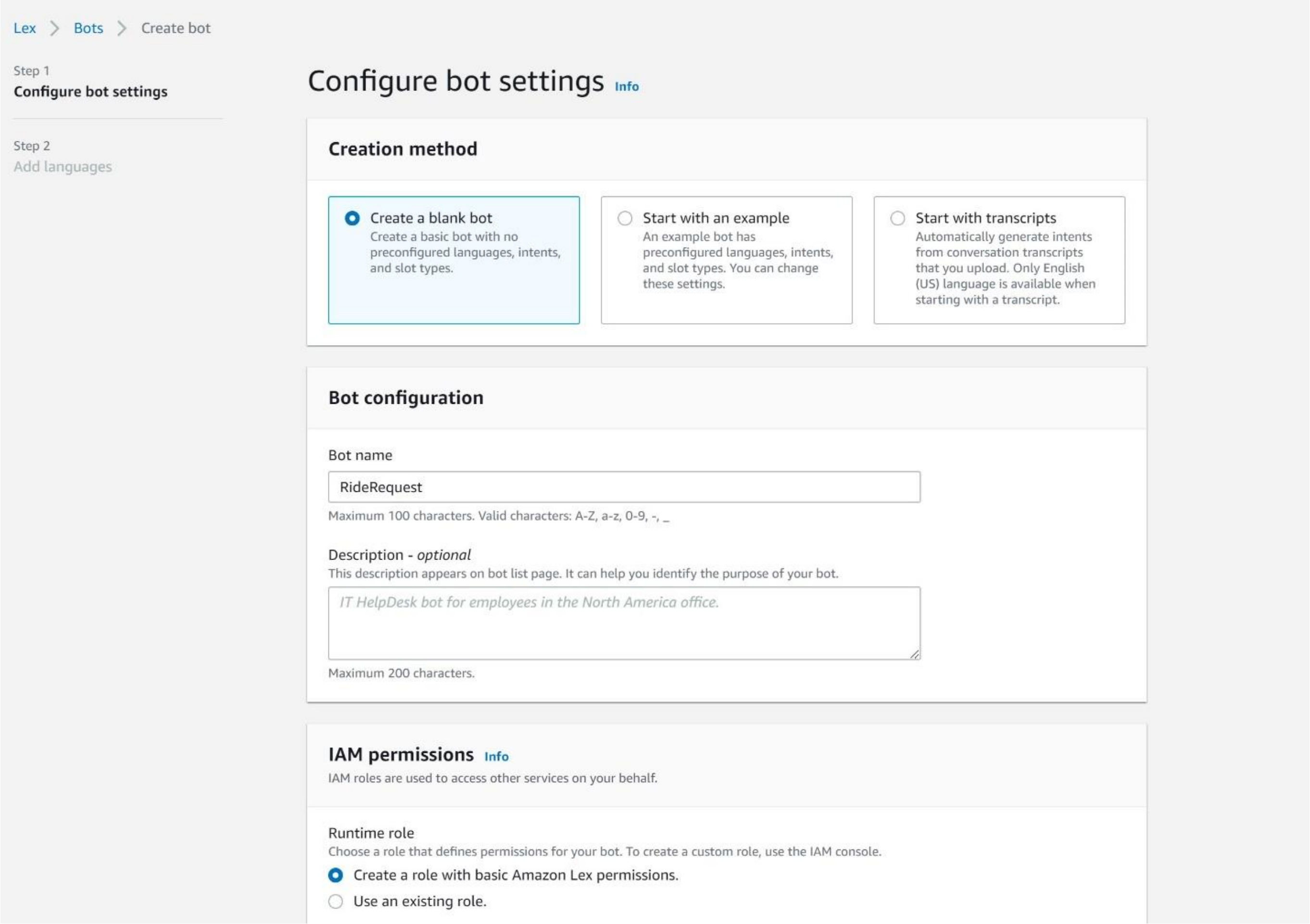
12. After adding slots, I provided Confirmation message and Fulfillments to the Taxi Intent.

13. Once the chatbot was configured, I tested it using the Lex test bot interface to ensure it was functioning correctly.

14. I took screenshots at each step of the process, including the creation of the bot, customization, and testing.

RideRequest bot created using AWS Lex

Steps for Creating a Chat bot using AWS Lex:



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

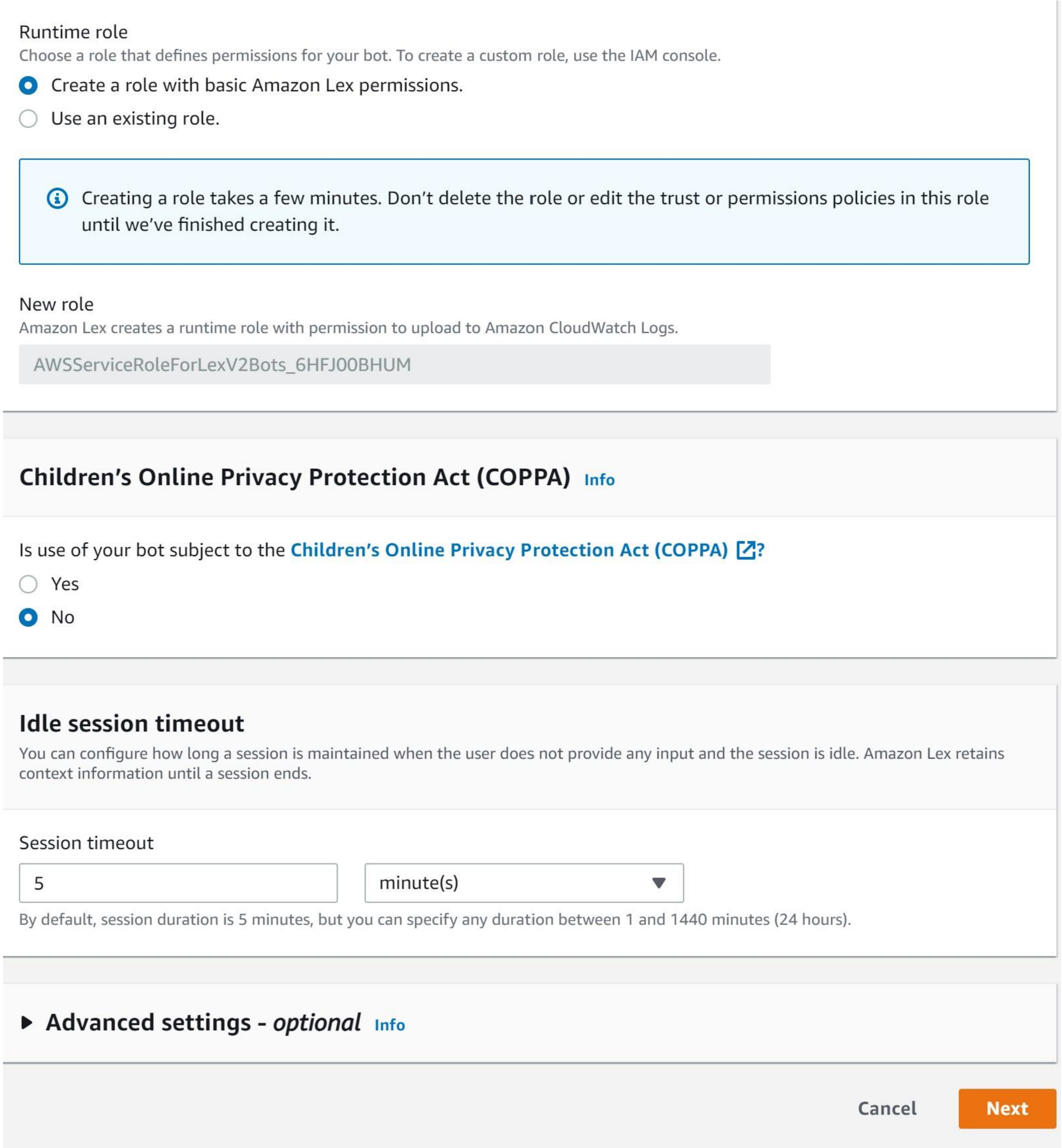
Description - *optional*
This description appears on bot list page. It can help you identify the purpose of your bot.
IT HelpDesk bot for employees in the North America office.
Maximum 200 characters.

IAM permissions Info
IAM roles 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.

Fig 1: Custom Bot creation config with RideRequest name

RideRequest bot created using AWS Lex



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.

AWSServiceRoleForLexV2Bots_6HFJ00BHUM

Children's Online Privacy Protection Act (COPPA) Info

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

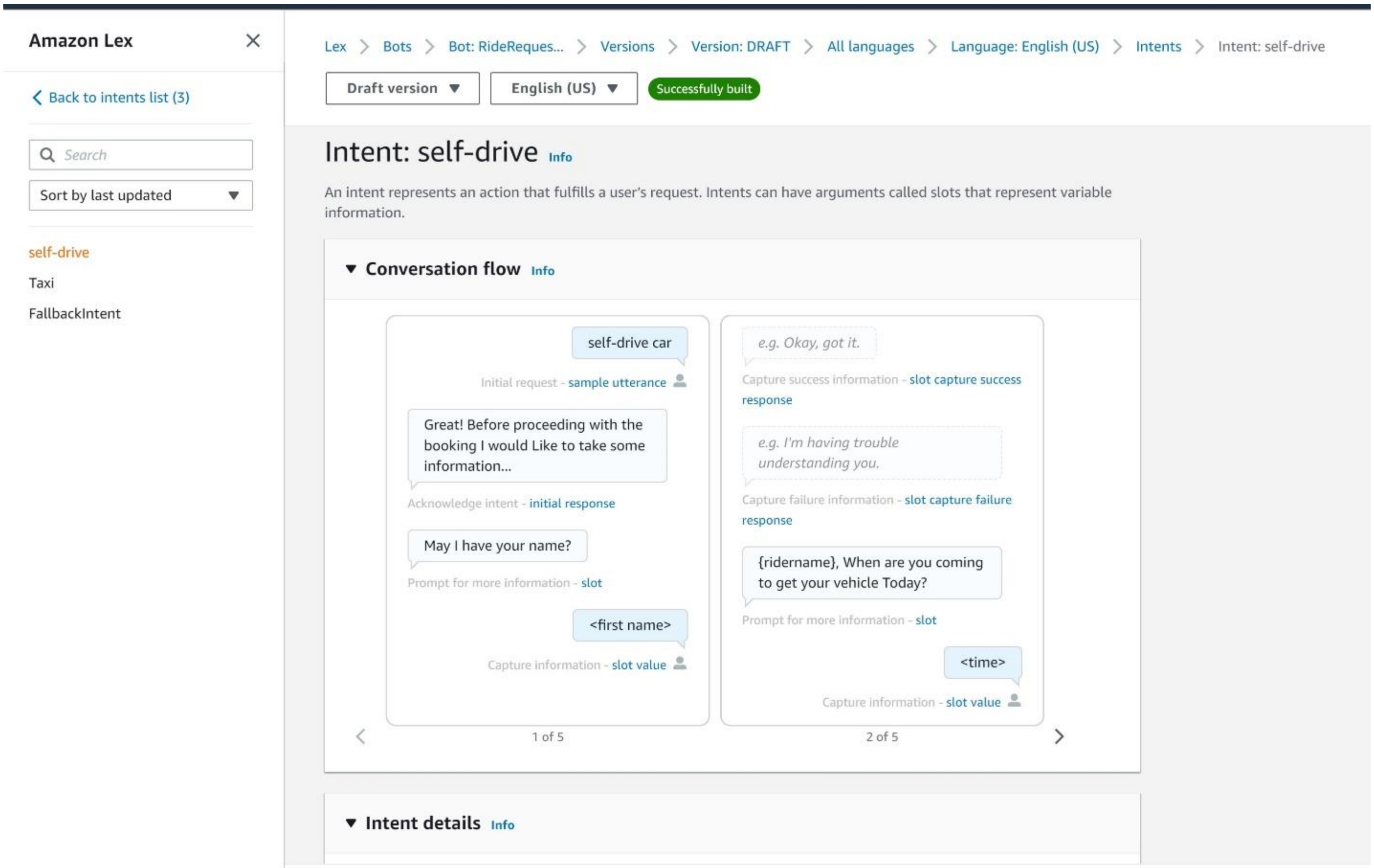
5 | minute(s) ▼

By default, session duration is 5 minutes, but you can specify any duration between 1 and 1440 minutes (24 hours).

▶ **Advanced settings - *optional*** Info

Fig 2: Bot creation configuration part

RideRequest bot created using AWS Lex



Amazon Lex ✕

‹ Back to intents list (3)

Search

Sort by last updated ▼

self-drive

Taxi

FallbackIntent

Lex > Bots > Bot: RideReques... > Versions > Version: DRAFT > All languages > Language: English (US) > Intents > Intent: self-drive

Draft version ▼ English (US) ▼ Successfully built

Intent: self-drive Info

An intent represents an action that fulfills a user's request. Intents can have arguments called slots that represent variable information.

▼ Conversation flow Info

self-drive car

Initial request - sample utterance

Great! Before proceeding with the booking I would Like to take some information...

Acknowledge intent - initial response

May I have your name?

Prompt for more information - slot

<first name>

Capture information - slot value

1 of 5

e.g. Okay, got it.

Capture success information - slot capture success response

e.g. I'm having trouble understanding you.

Capture failure information - slot capture failure response

{ridername}, When are you coming to get your vehicle Today?

Prompt for more information - slot

<time>

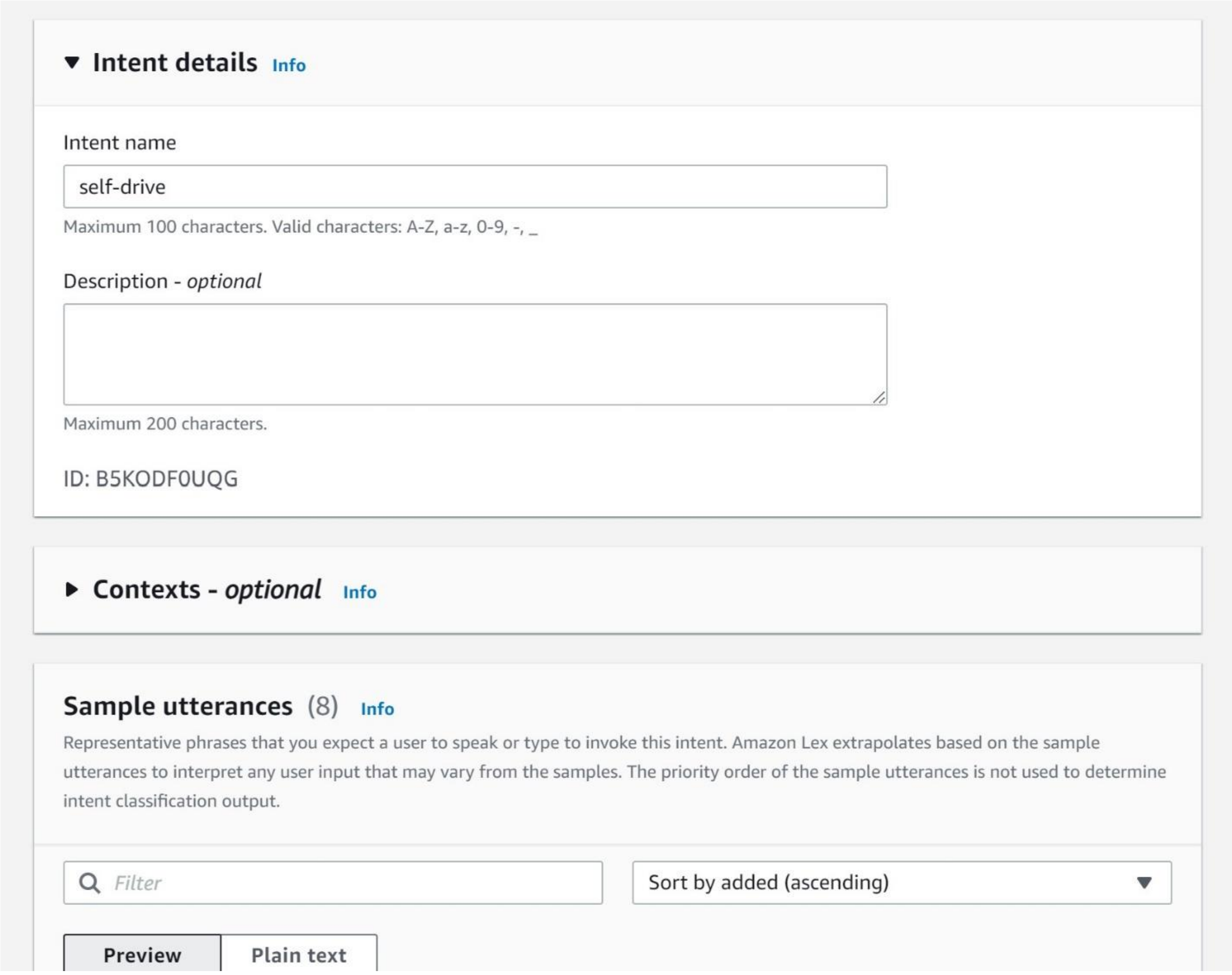
Capture information - slot value

2 of 5

▼ Intent details Info

Fig 3: Intent for Self-drive

RideRequest bot created using AWS Lex



▼ Intent details

Intent name

self-drive

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

Description - *optional*

Maximum 200 characters.

ID: B5KODF0UQG

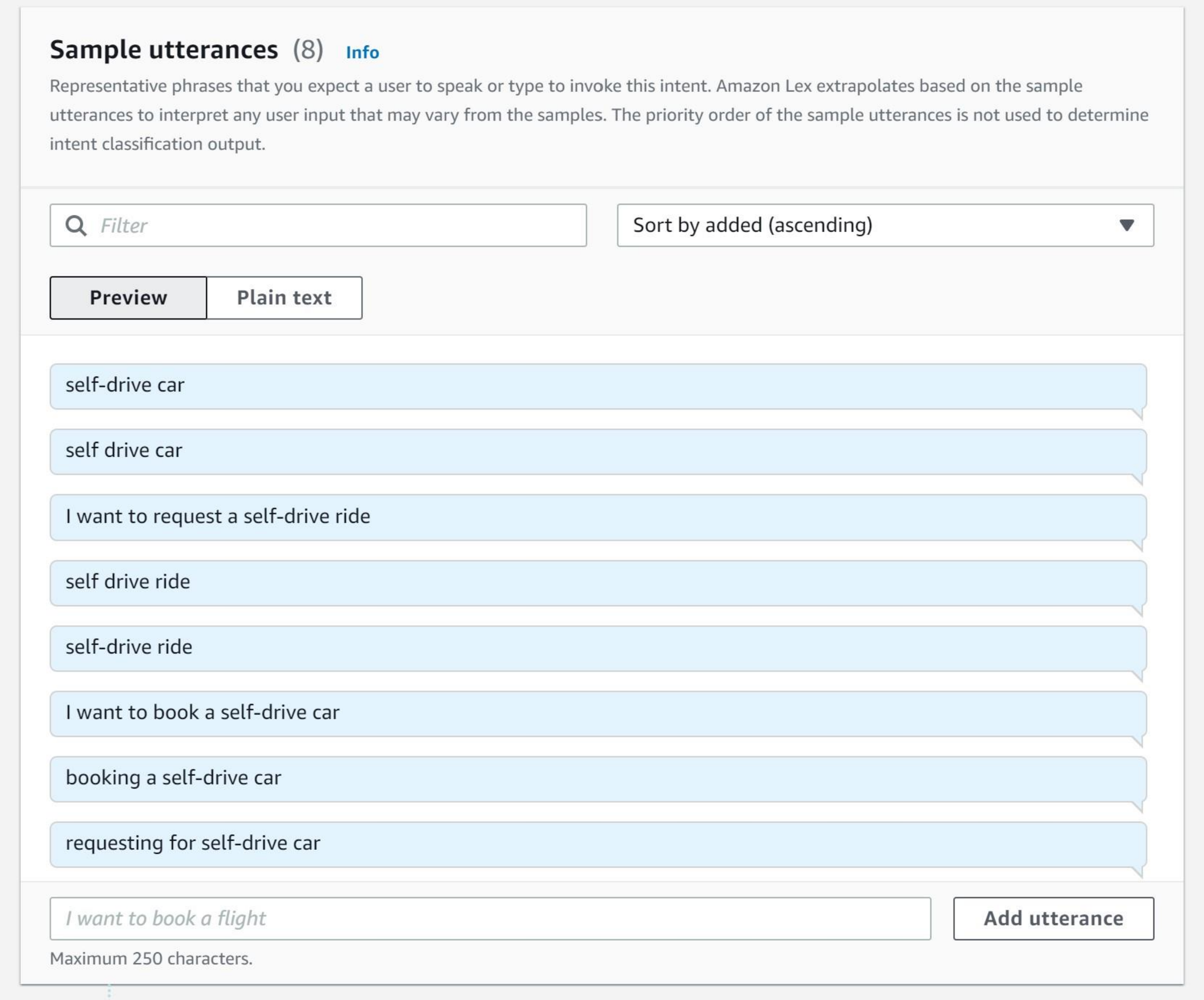
▶ Contexts - *optional*

Sample utterances (8)

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.

Fig 4: Intent section details for Self-Drive Intent

RideRequest bot created using AWS Lex



Sample utterances (8) Info

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.

self-drive car

self drive car

I want to request a self-drive ride

self drive ride

self-drive ride

I want to book a self-drive car

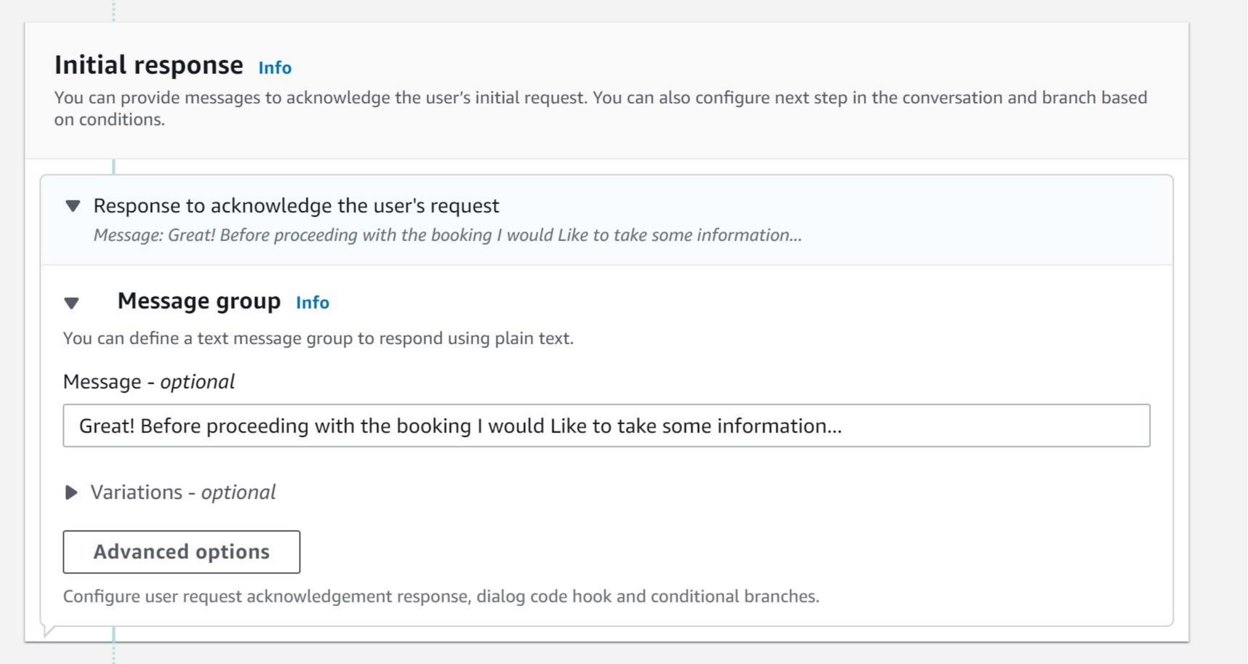
booking a self-drive car

requesting for self-drive car

Maximum 250 characters.

Fig 5: Sample utterances added for self-drive intent

RideRequest bot created using AWS Lex



Initial response Info

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: Great! Before proceeding with the booking I would Like to take some information...

▼ **Message group** Info

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

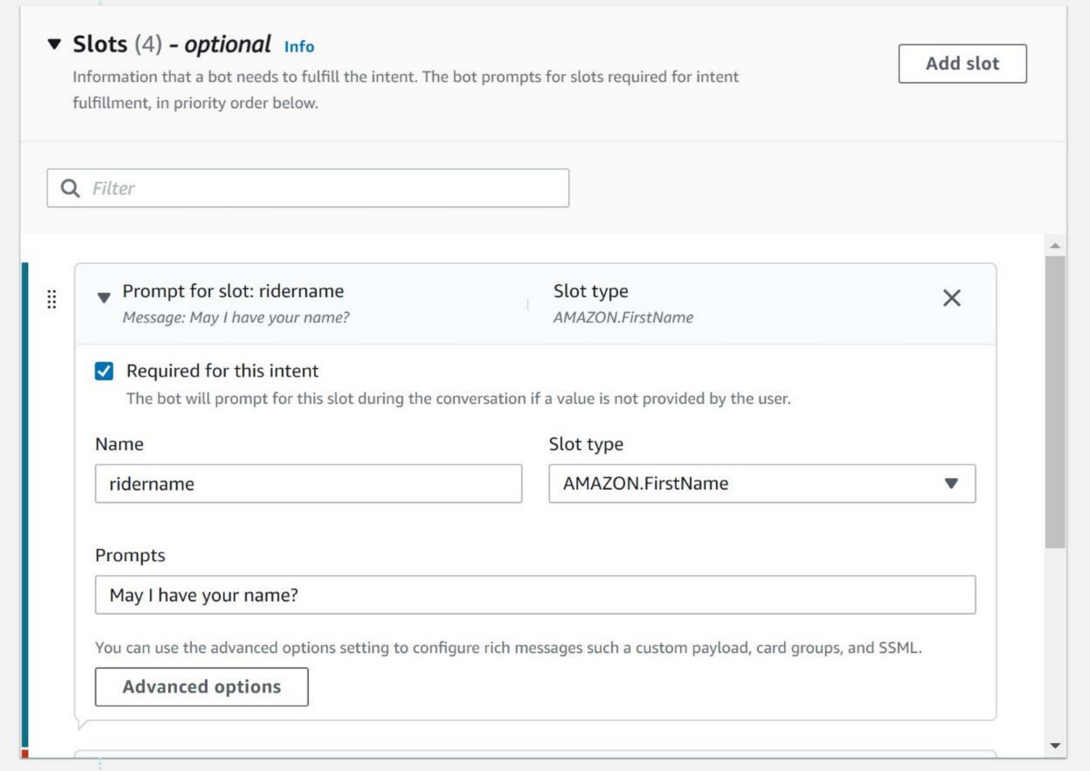
Message - *optional*

Great! Before proceeding with the booking I would Like to take some information...

▶ Variations - *optional*

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

Fig 6: Initial Response provided of Self-drive Intent



▼ **Slots** (4) - ***optional*** Info

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

Filter

▼ Prompt for slot: ridername

Message: May I have your name?

Slot type

AMAZON.FirstName

☑ Required for this intent

The bot will prompt for this slot during the conversation if a value is not provided by the user.

Name: ridername

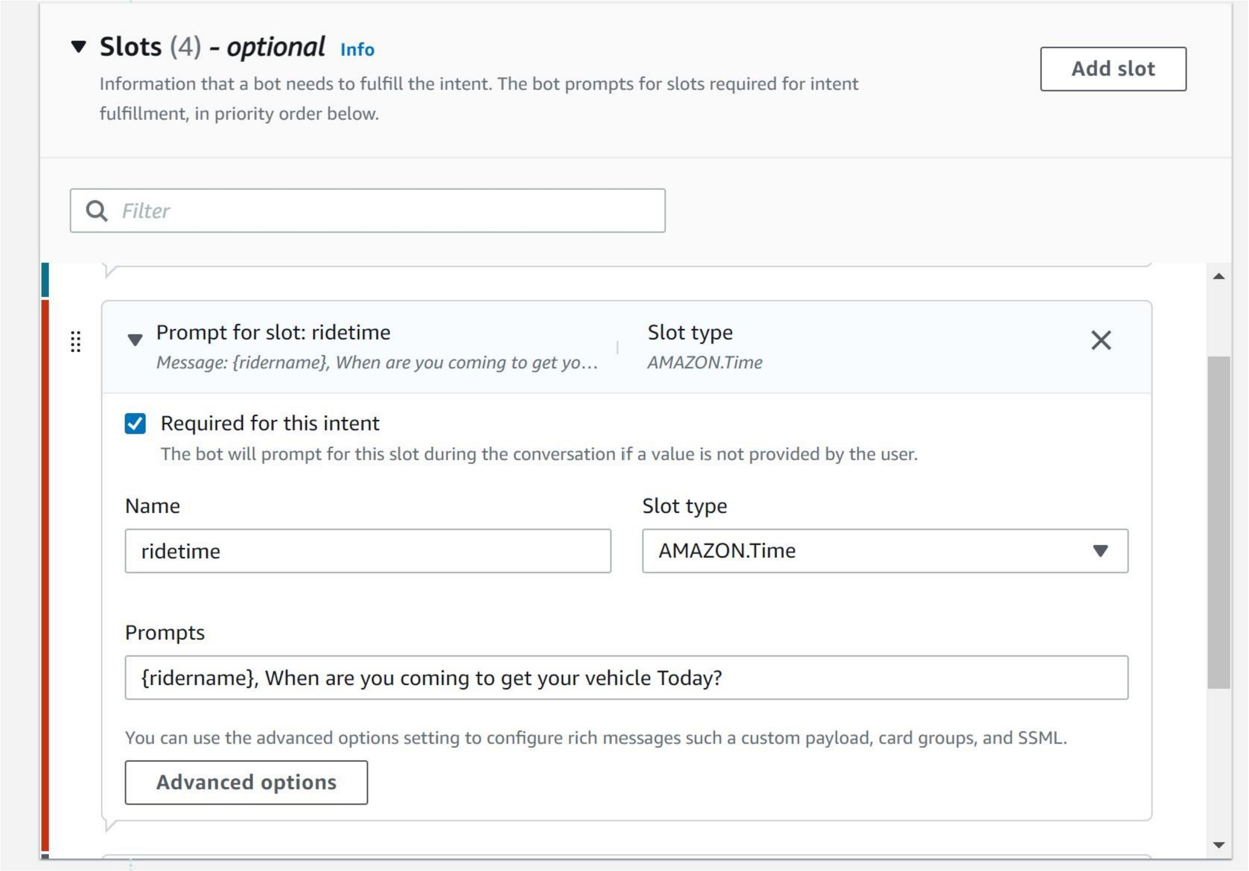
Slot type: AMAZON.FirstName

Prompts: May I have your name?

You can use the advanced options setting to configure rich messages such a custom payload, card groups, and SSML.

Fig 7: Creating Slots for self-drive intent

RideRequest bot created using AWS Lex



Slots (4) - *optional* Info

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

Prompt for slot: ridetime

Message: {ridername}, When are you coming to get yo…

Slot type

AMAZON.Time

☑ Required for this intent

The bot will prompt for this slot during the conversation if a value is not provided by the user.

Name: ridetime

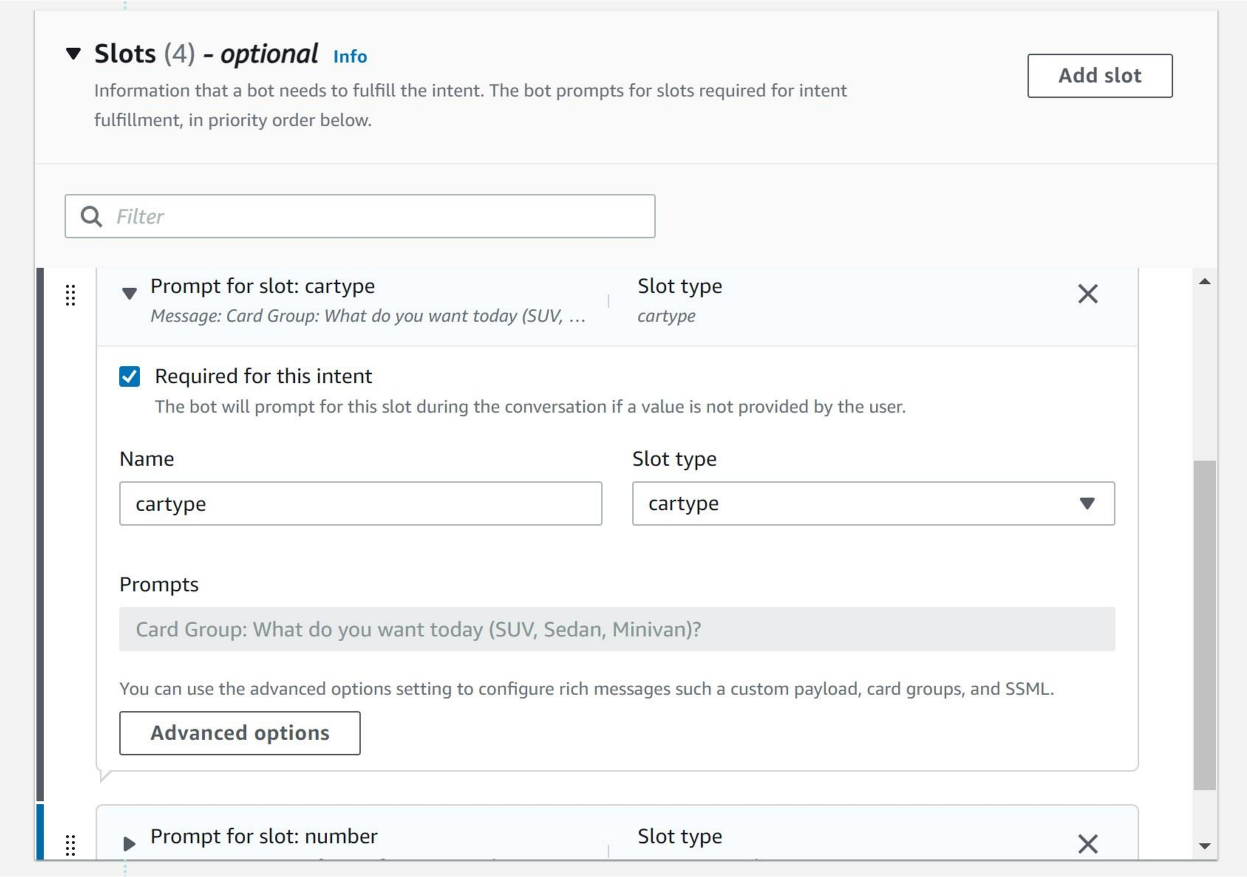
Slot type: AMAZON.Time

Prompts: {ridername}, When are you coming to get your vehicle Today?

You can use the advanced options setting to configure rich messages such a custom payload, card groups, and SSML.

Fig 8: Slots for self-drive Intent

RideRequest bot created using AWS Lex



▼ **Slots** (4) - *optional* Info

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

▼ Prompt for slot: cartype | Slot type
Message: Card Group: What do you want today (SUV, … | *cartype*

☑ Required for this intent

The bot will prompt for this slot during the conversation if a value is not provided by the user.

Name: cartype

Slot type: cartype

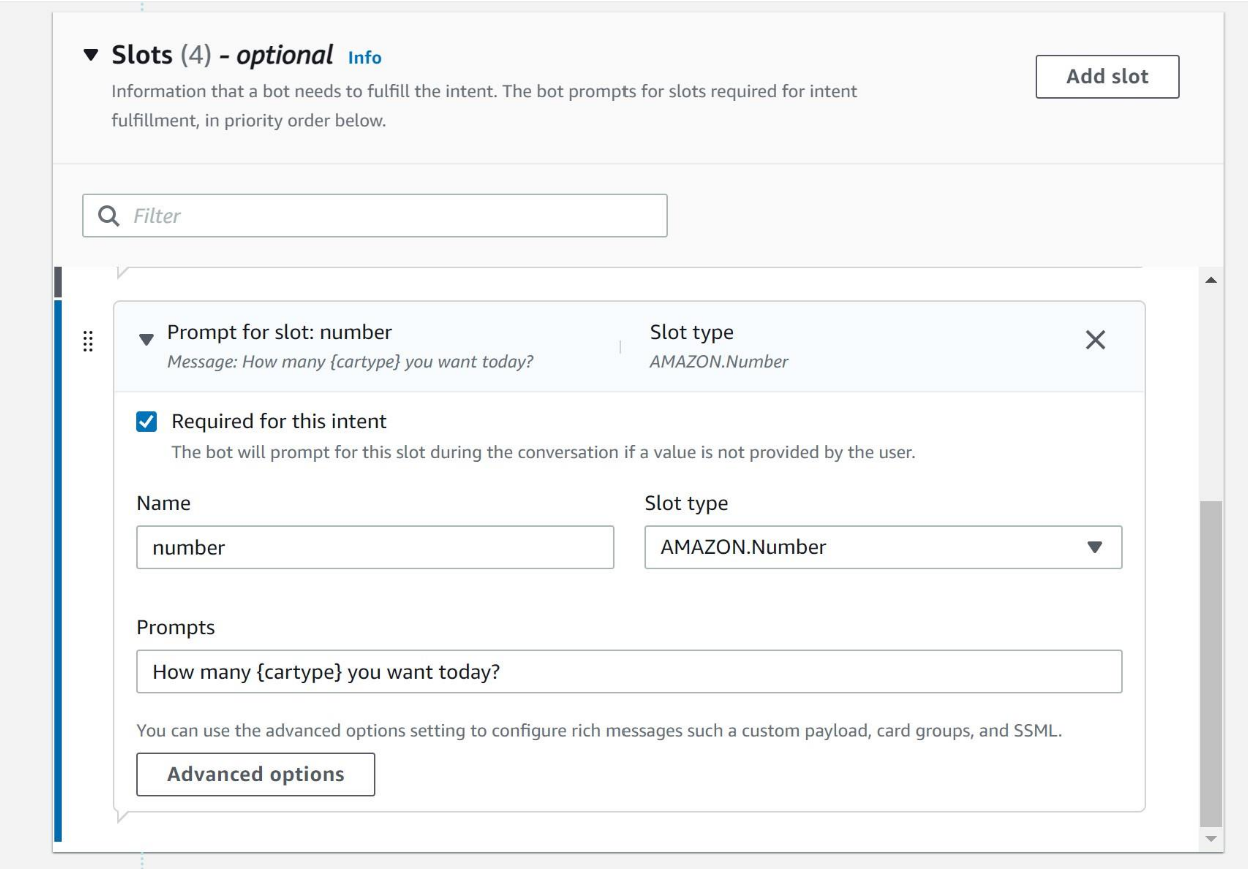
Prompts: Card Group: What do you want today (SUV, Sedan, Minivan)?

You can use the advanced options setting to configure rich messages such a custom payload, card groups, and SSML.

▶ Prompt for slot: number | Slot type

Fig 9: Slots for self-drive Intent

RideRequest bot created using AWS Lex



Slots (4) - *optional* Info

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

Add slot

Filter

Prompt for slot: number
Message: How many {cartype} you want today?

Slot type
AMAZON.Number

☑ Required for this intent

The bot will prompt for this slot during the conversation if a value is not provided by the user.

Name: number

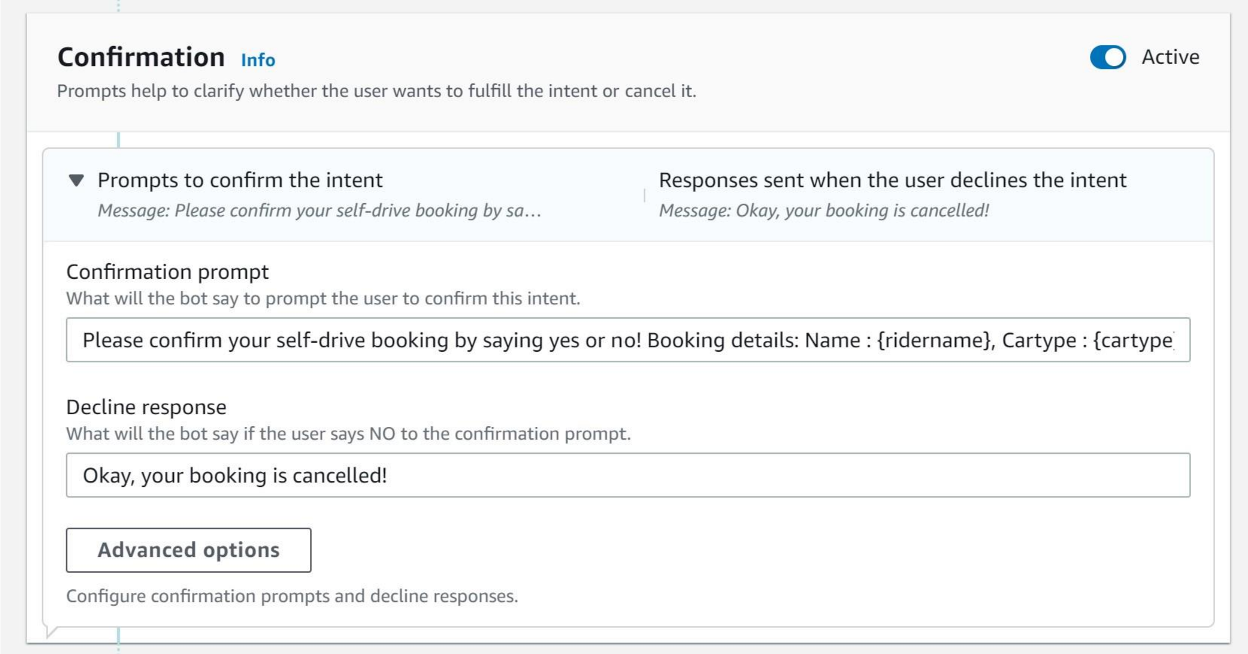
Slot type: AMAZON.Number

Prompts: How many {cartype} you want today?

You can use the advanced options setting to configure rich messages such a custom payload, card groups, and SSML.

Advanced options

Fig 10: Slots for self-drive Intent



Confirmation Info

Active

Prompts help to clarify whether the user wants to fulfill the intent or cancel it.

Prompts to confirm the intent
Message: Please confirm your self-drive booking by sa…

Responses sent when the user declines the intent
Message: Okay, your booking is cancelled!

Confirmation prompt
What will the bot say to prompt the user to confirm this intent.

Please confirm your self-drive booking by saying yes or no! Booking details: Name : {ridername}, Cartype : {cartype

Decline response
What will the bot say if the user says NO to the confirmation prompt.

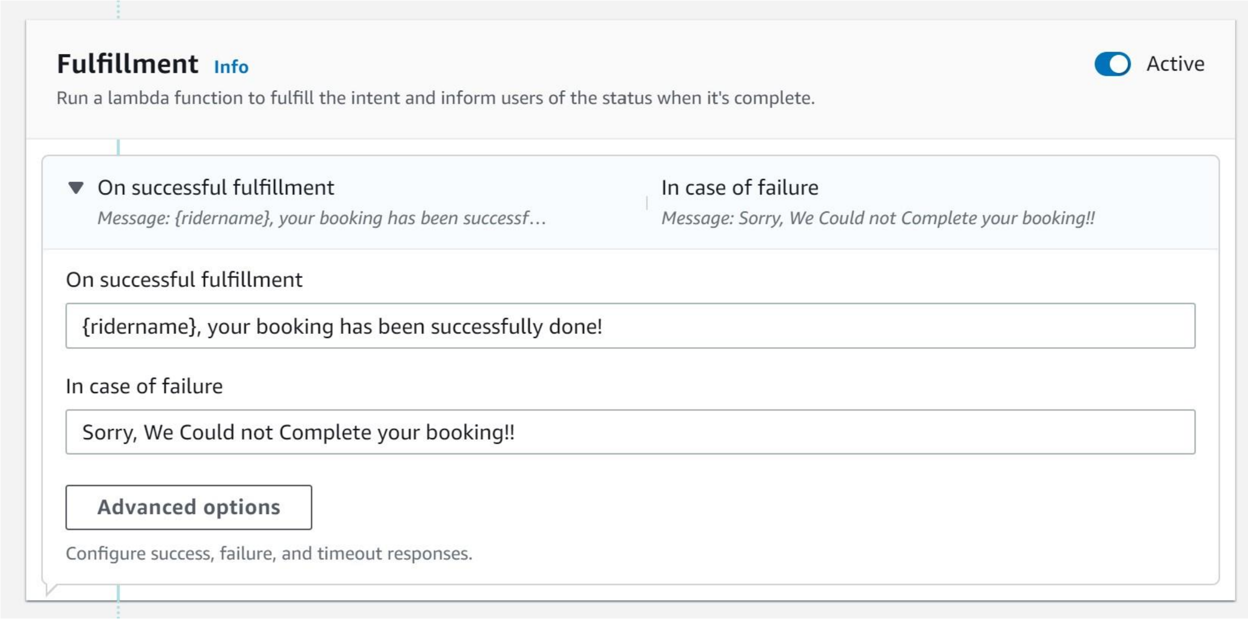
Okay, your booking is cancelled!

Advanced options

Configure confirmation prompts and decline responses.

Fig 11: Confirmation Section for self -drive intent with Confirmation Prompt and Decline response

RideRequest bot created using AWS Lex



Fulfillment Info

Active

Run a lambda function to fulfill the intent and inform users of the status when it's complete.

▼ On successful fulfillment

Message: {ridername}, your booking has been successf…

In case of failure

Message: Sorry, We Could not Complete your booking!!

On successful fulfillment

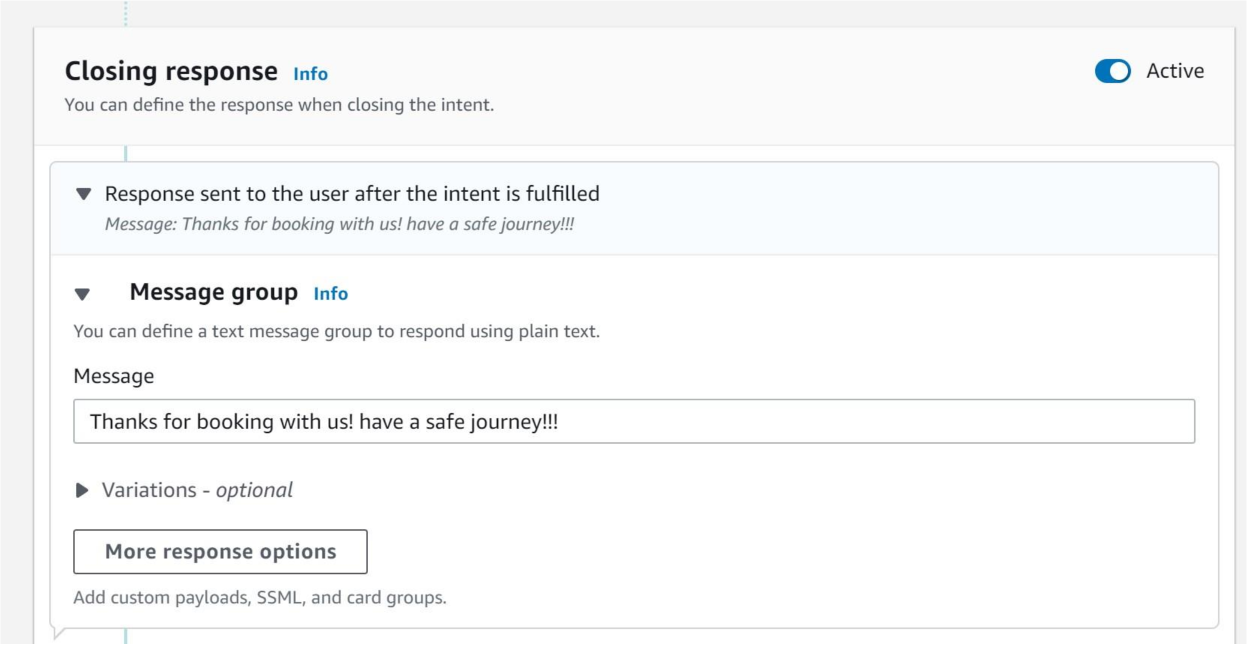
{ridername}, your booking has been successfully done!

In case of failure

Sorry, We Could not Complete your booking!!

Configure success, failure, and timeout responses.

Fig 12: Fulfillment(Successful or Failure) of Self-drive intent



Closing response Info

Active

You can define the response when closing the intent.

▼ Response sent to the user after the intent is fulfilled

Message: Thanks for booking with us! have a safe journey!!!

▼ **Message group** Info

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

Message

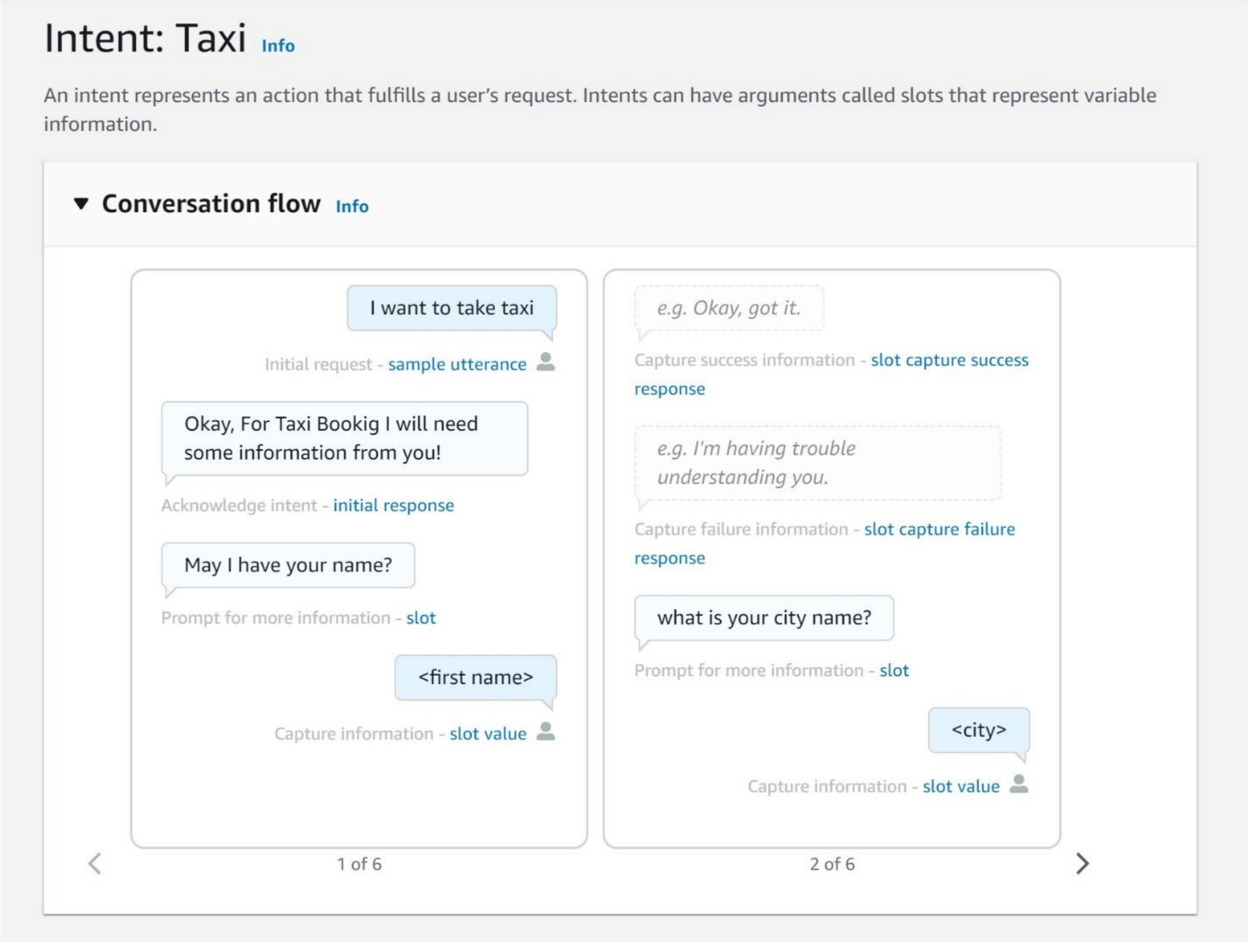
Thanks for booking with us! have a safe journey!!!

▶ Variations - *optional*

Add custom payloads, SSML, and card groups.

Fig 13: Adding Closing Response of Self- Drive intent

RideRequest bot created using AWS Lex



Intent: Taxi

An intent represents an action that fulfills a user's request. Intents can have arguments called slots that represent variable information.

▼ Conversation flow

I want to take taxi

Initial request - sample utterance

Okay, For Taxi Bookig I will need some information from you!

Acknowledge intent - initial response

May I have your name?

Prompt for more information - slot

<first name>

Capture information - slot value

e.g. Okay, got it.

Capture success information - slot capture success response

e.g. I'm having trouble understanding you.

Capture failure information - slot capture failure response

what is your city name?

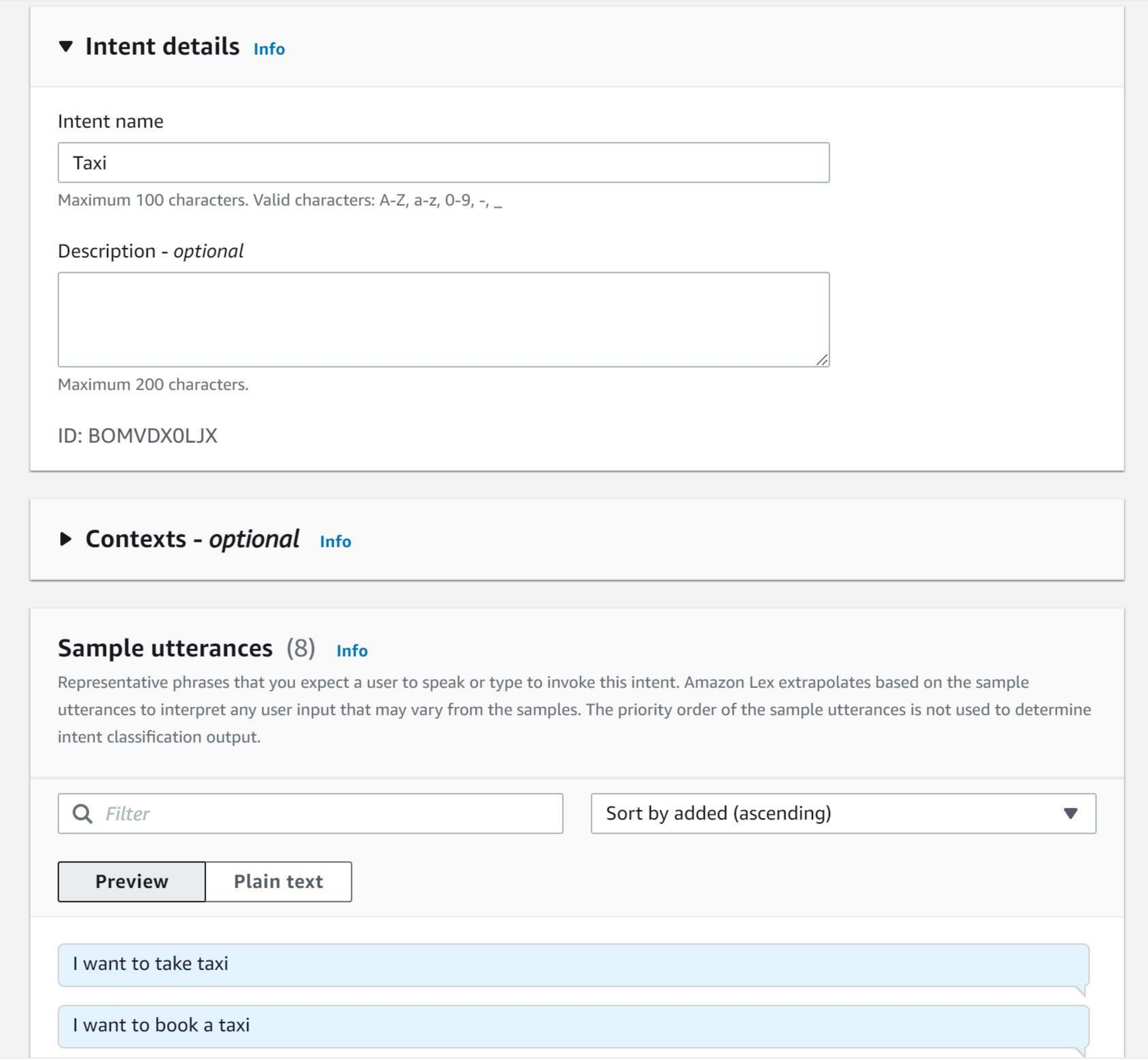
Prompt for more information - slot

<city>

Capture information - slot value

Fig 14 : Creating Intent for Taxi booking

RideRequest bot created using AWS Lex



Intent name

Taxi

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

Description - *optional*

Maximum 200 characters.

ID: BOMVDX0LJX

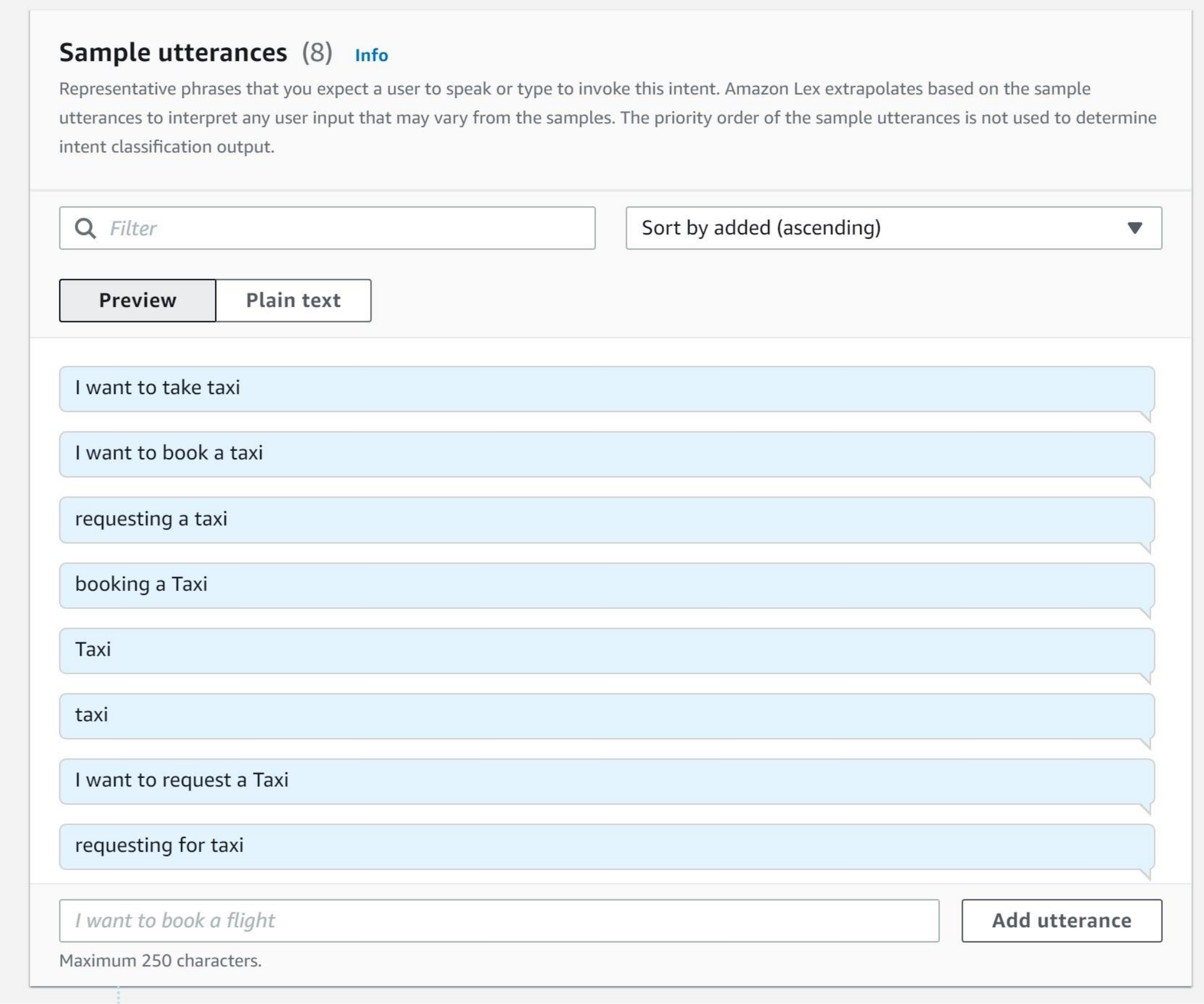
Representative phrases that you expect a user to speak or type to invoke this intent. Amazon Lex extrapolates based on the sample utterances to interpret any user input that may vary from the samples. The priority order of the sample utterances is not used to determine intent classification output.

I want to take taxi

I want to book a taxi

Fig 15: Intent configuration of Taxi booking

RideRequest bot created using AWS Lex



Sample utterances (8) Info

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.

I want to take taxi

I want to book a taxi

requesting a taxi

booking a Taxi

Taxi

taxi

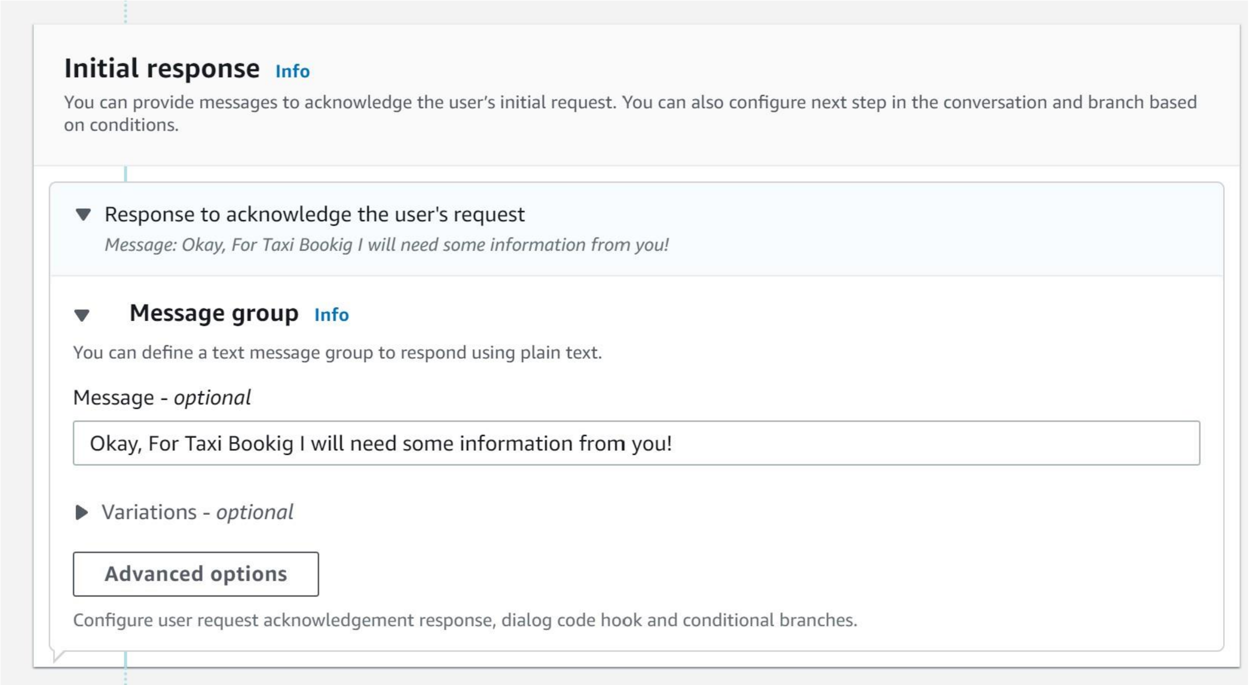
I want to request a Taxi

requesting for taxi

Maximum 250 characters.

Fig 16: Sample Utterances added for Taxi intent

RideRequest bot created using AWS Lex



Initial response Info

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, For Taxi Bookig I will need some information from you!

▼ **Message group** Info

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

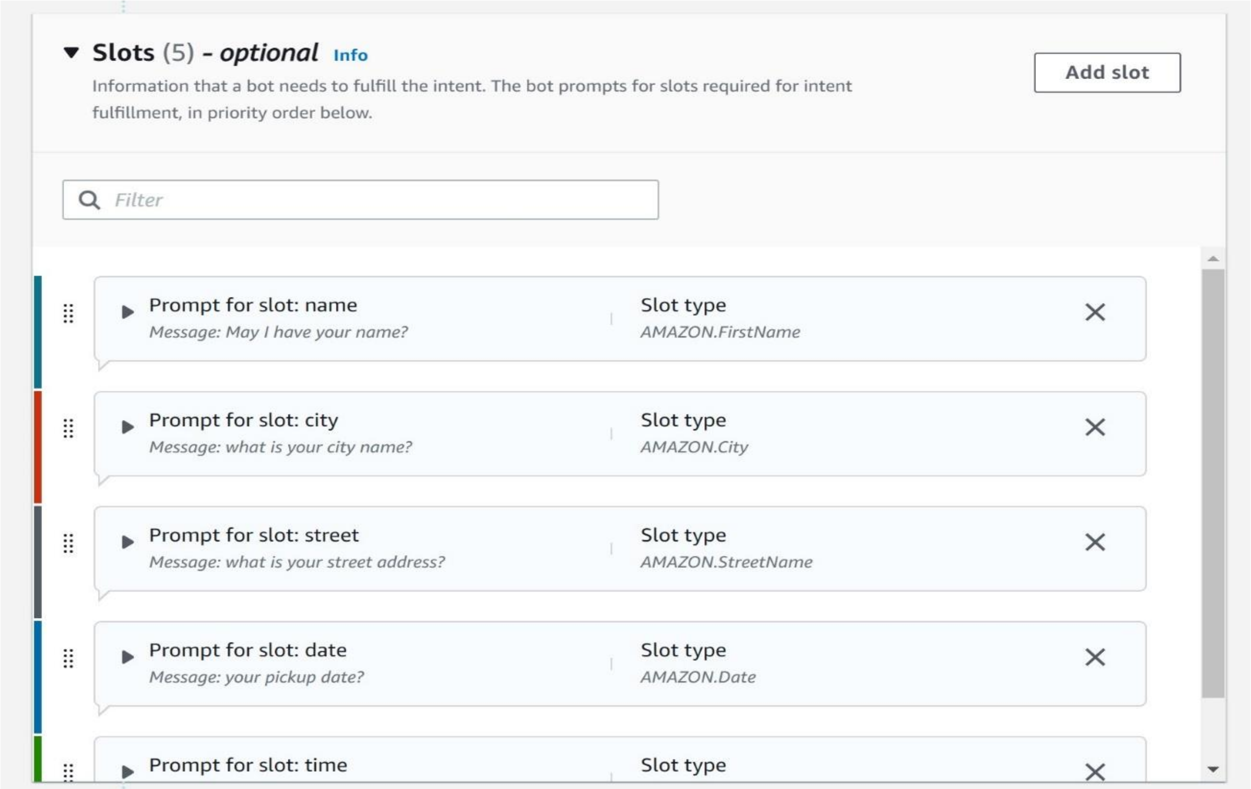
Message - *optional*

Okay, For Taxi Bookig I will need some information from you!

▶ Variations - *optional*

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

Fig 17: Initial Response with message for Taxi Intent has been added



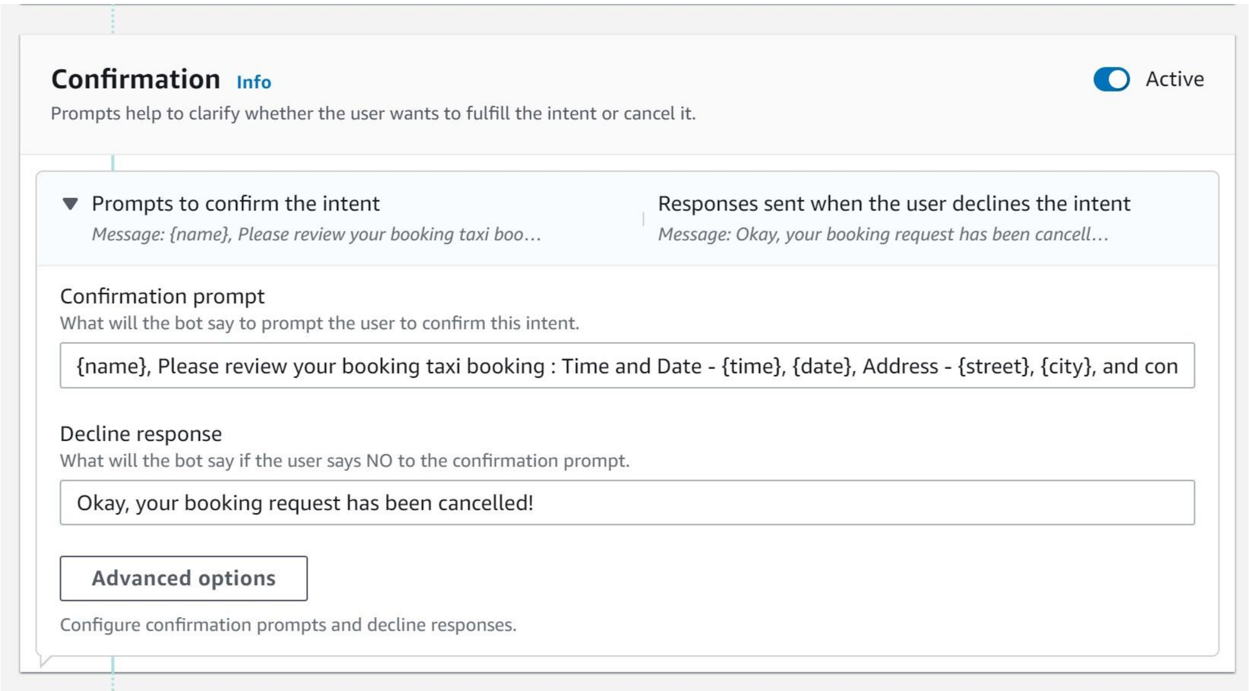
▼ **Slots** (5) - ***optional*** Info

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

▶ Prompt for slot: name *Message: May I have your name?*	Slot type *AMAZON.FirstName*
▶ Prompt for slot: city *Message: what is your city name?*	Slot type *AMAZON.City*
▶ Prompt for slot: street *Message: what is your street address?*	Slot type *AMAZON.StreetName*
▶ Prompt for slot: date *Message: your pickup date?*	Slot type *AMAZON.Date*
▶ Prompt for slot: time	Slot type

Fig 18: Adding Slots and slot types for Taxi intent

RideRequest bot created using AWS Lex



Confirmation Info

Active

Prompts help to clarify whether the user wants to fulfill the intent or cancel it.

▼ Prompts to confirm the intent

Message: {name}, Please review your booking taxi boo…

Responses sent when the user declines the intent

Message: Okay, your booking request has been cancell…

Confirmation prompt

What will the bot say to prompt the user to confirm this intent.

{name}, Please review your booking taxi booking : Time and Date - {time}, {date}, Address - {street}, {city}, and con

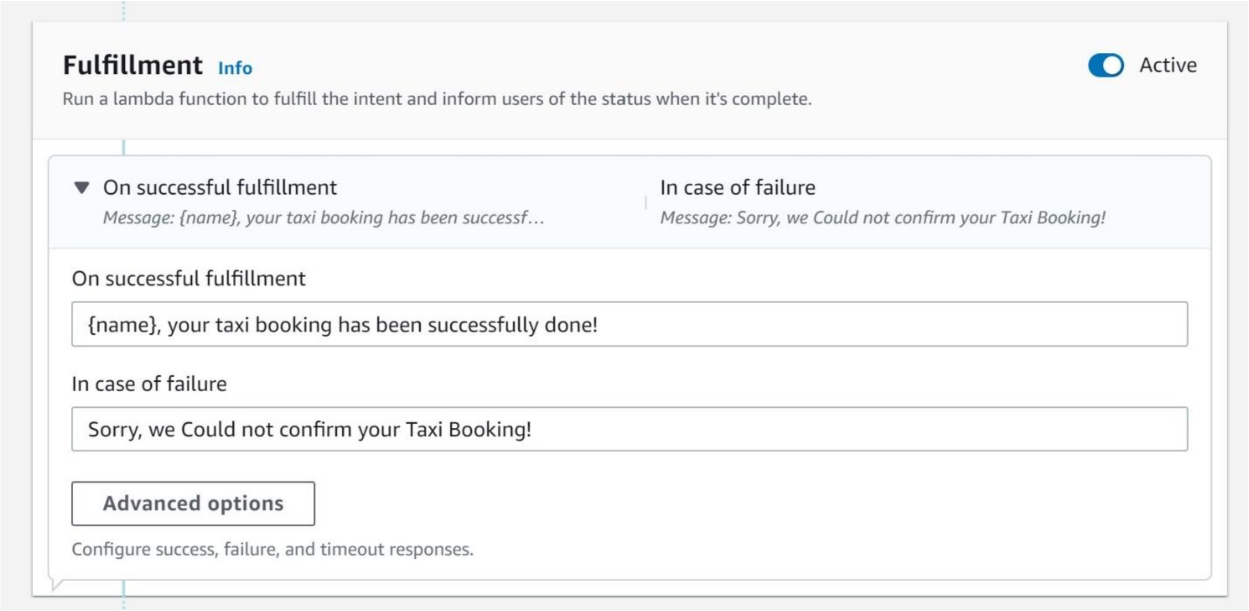
Decline response

What will the bot say if the user says NO to the confirmation prompt.

Okay, your booking request has been cancelled!

Configure confirmation prompts and decline responses.

Fig 19: Adding Confirmation Prompt and Decline response into Taxi Intent



Fulfillment Info

Active

Run a lambda function to fulfill the intent and inform users of the status when it's complete.

▼ On successful fulfillment

Message: {name}, your taxi booking has been successf…

In case of failure

Message: Sorry, we Could not confirm your Taxi Booking!

On successful fulfillment

{name}, your taxi booking has been successfully done!

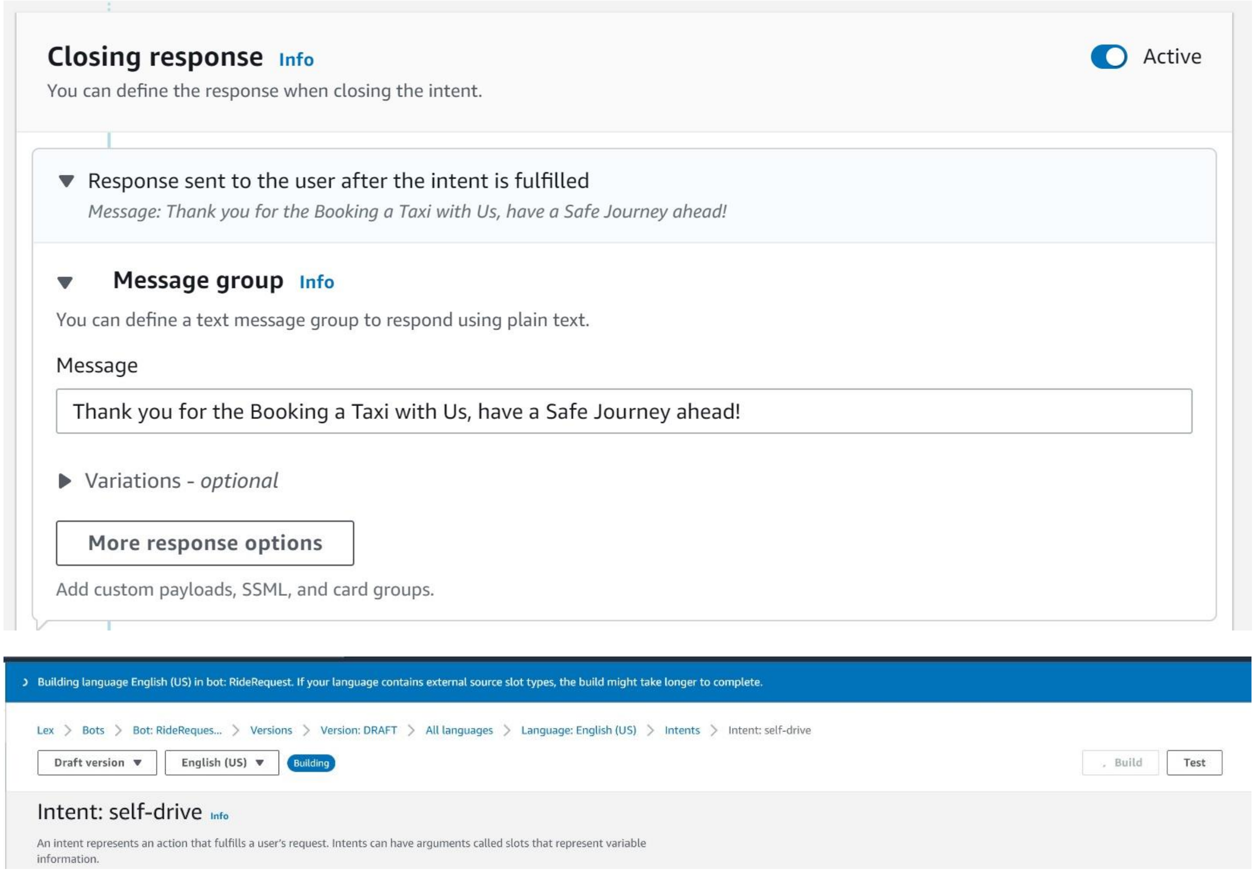
In case of failure

Sorry, we Could not confirm your Taxi Booking!

Configure success, failure, and timeout responses.

Fig 19: Fulfillment(Successful or Failure) for Taxi Intent

RideRequest bot created using AWS Lex



Closing response Info

Active

You can define the response when closing the intent.

▼ Response sent to the user after the intent is fulfilled

Message: Thank you for the Booking a Taxi with Us, have a Safe Journey ahead!

▼ **Message group** Info

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

Message

Thank you for the Booking a Taxi with Us, have a Safe Journey ahead!

▶ Variations - *optional*

More response options

Add custom payloads, SSML, and card groups.

› Building language English (US) in bot: RideRequest. If your language contains external source slot types, the build might take longer to complete.

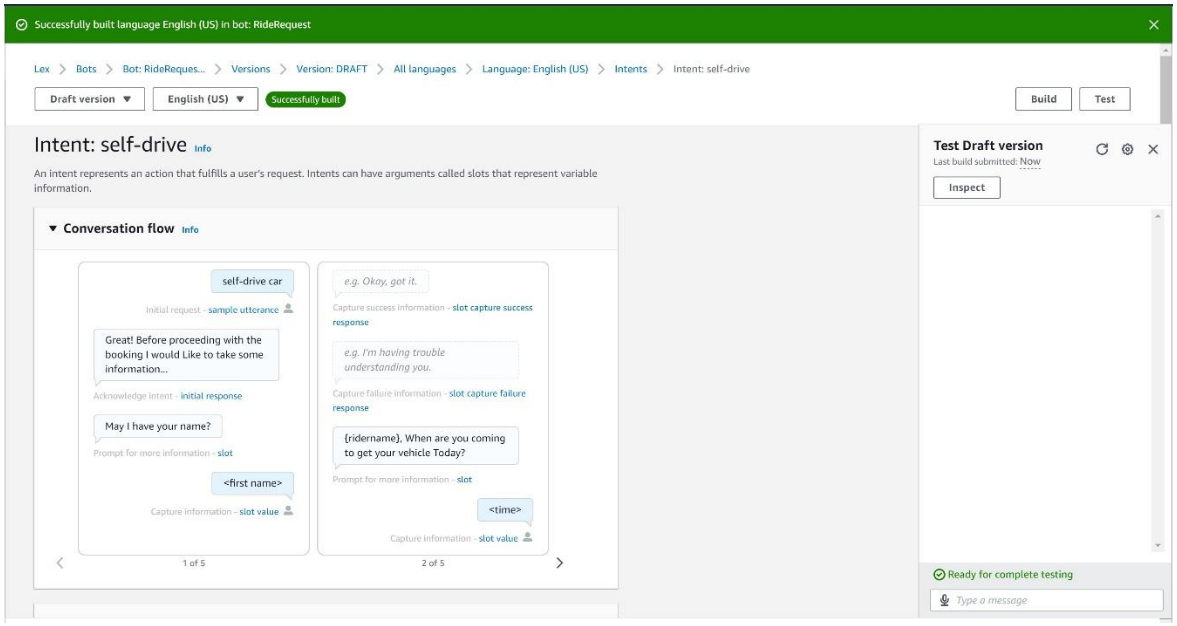
Lex > Bots > Bot: RideReques... > Versions > Version: DRAFT > All languages > Language: English (US) > Intents > Intent: self-drive

Draft version ▼ English (US) ▼ Building Build Test

Intent: self-drive Info

An intent represents an action that fulfills a user's request. Intents can have arguments called slots that represent variable information.

Fig 20 : Closing Response of Taxi Intent and then Building the Bot



Successfully built language English (US) in bot: RideRequest

Lex > Bots > Bot: RideReques... > Versions > Version: DRAFT > All languages > Language: English (US) > Intents > Intent: self-drive

Draft version ▼ English (US) ▼ Successfully built Build Test

Intent: self-drive Info

An intent represents an action that fulfills a user's request. Intents can have arguments called slots that represent variable information.

▼ **Conversation flow** Info

self-drive car

Initial request - sample utterance

Great! Before proceeding with the booking I would Like to take some information...

Acknowledge intent - initial response

May I have your name?

Prompt for more information - slot

<first name>

Capture information - slot value

1 of 5

e.g. Okay, got it.

Capture success information - slot capture success response

e.g. I'm having trouble understanding you.

Capture failure information - slot capture failure response

{ridername}, When are you coming to get your vehicle Today?

Prompt for more information - slot

<time>

Capture information - slot value

2 of 5

Test Draft version

Last build submitted: Now

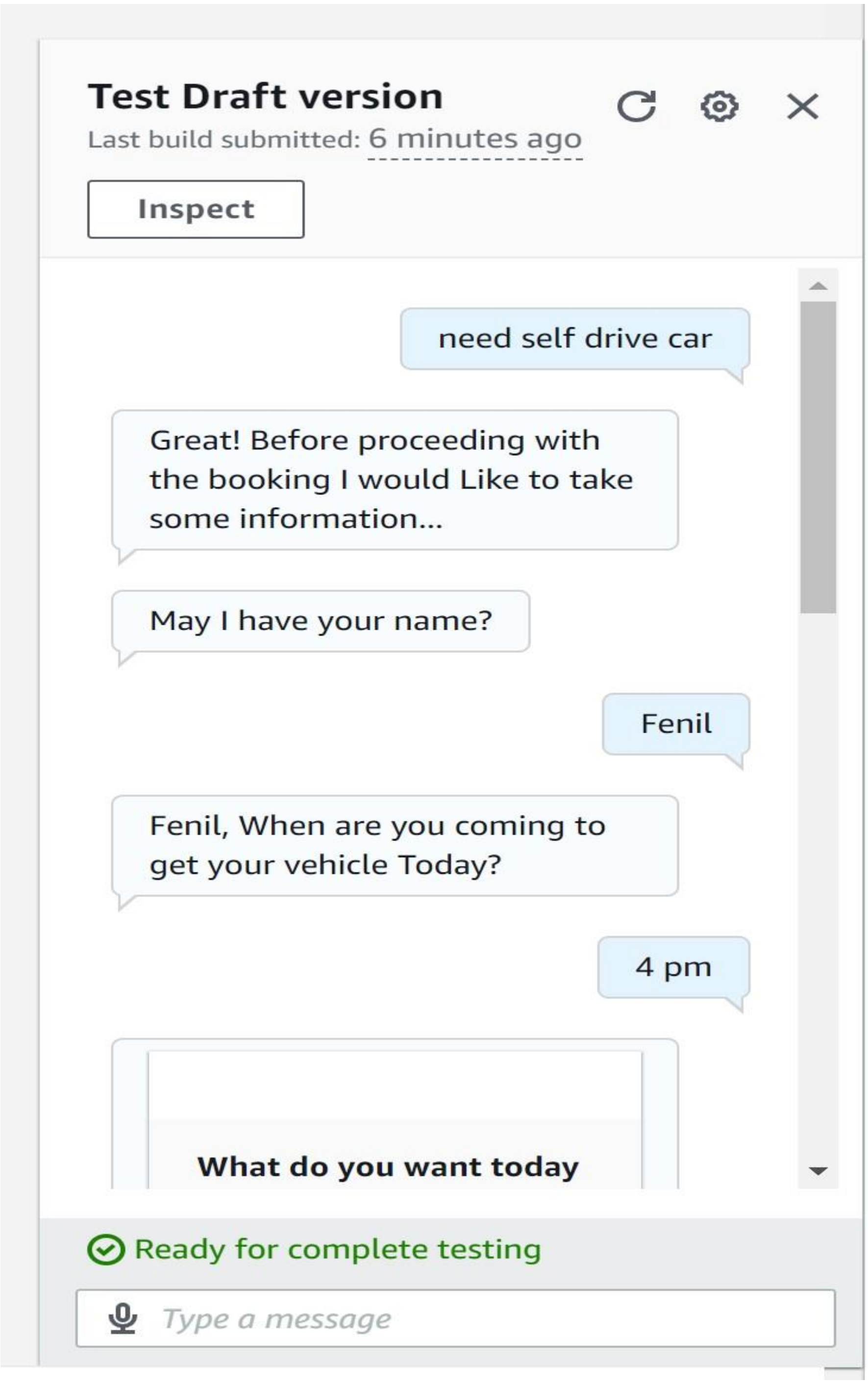
Inspect

Ready for complete testing

Type a message

Fig 21: Bot building done successfully

RideRequest bot created using AWS Lex



Test Draft version
Last build submitted: 6 minutes ago
Inspect

need self drive car

Great! Before proceeding with the booking I would Like to take some information...

May I have your name?

Fenil

Fenil, When are you coming to get your vehicle Today?

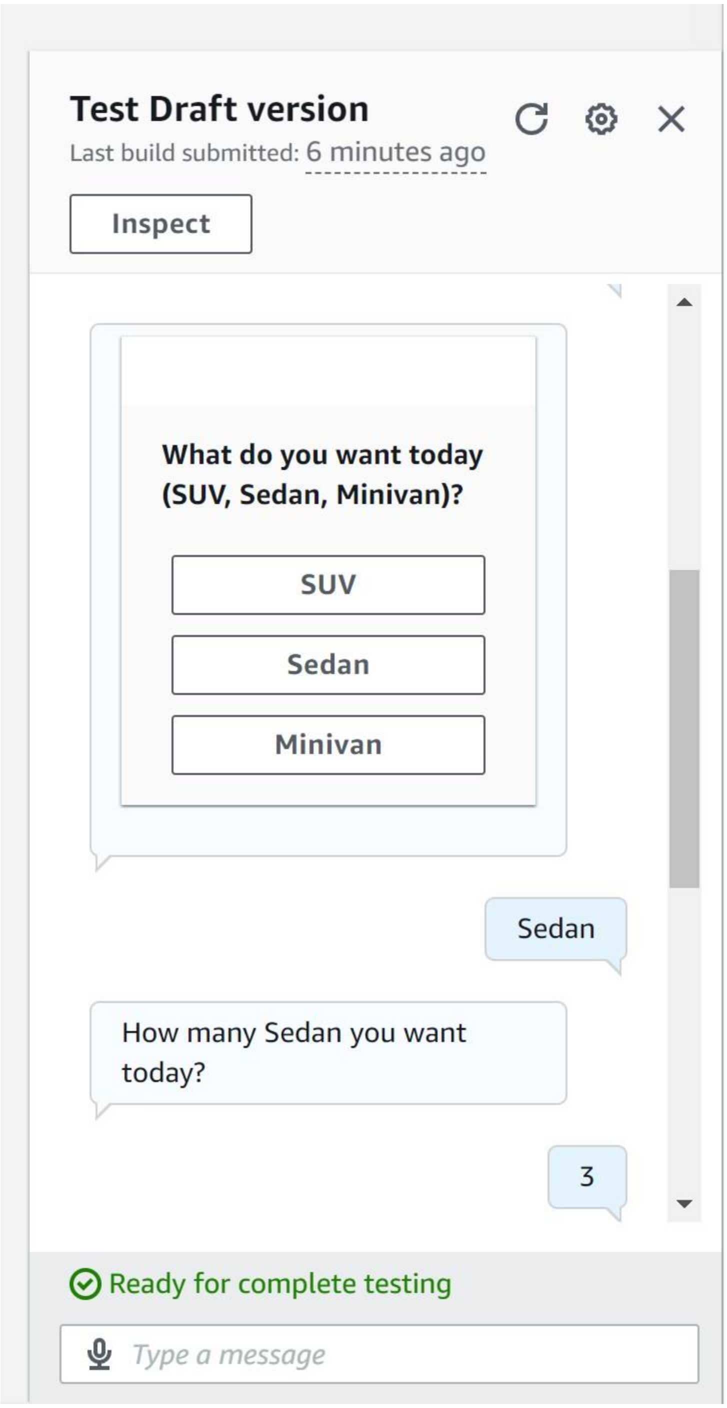
4 pm

What do you want today

Ready for complete testing
Type a message

Fig 22: Testing of Bot for Self Drive Car Booking

RideRequest bot created using AWS Lex



Test Draft version

Last build submitted: 6 minutes ago

What do you want today (SUV, Sedan, Minivan)?

Sedan

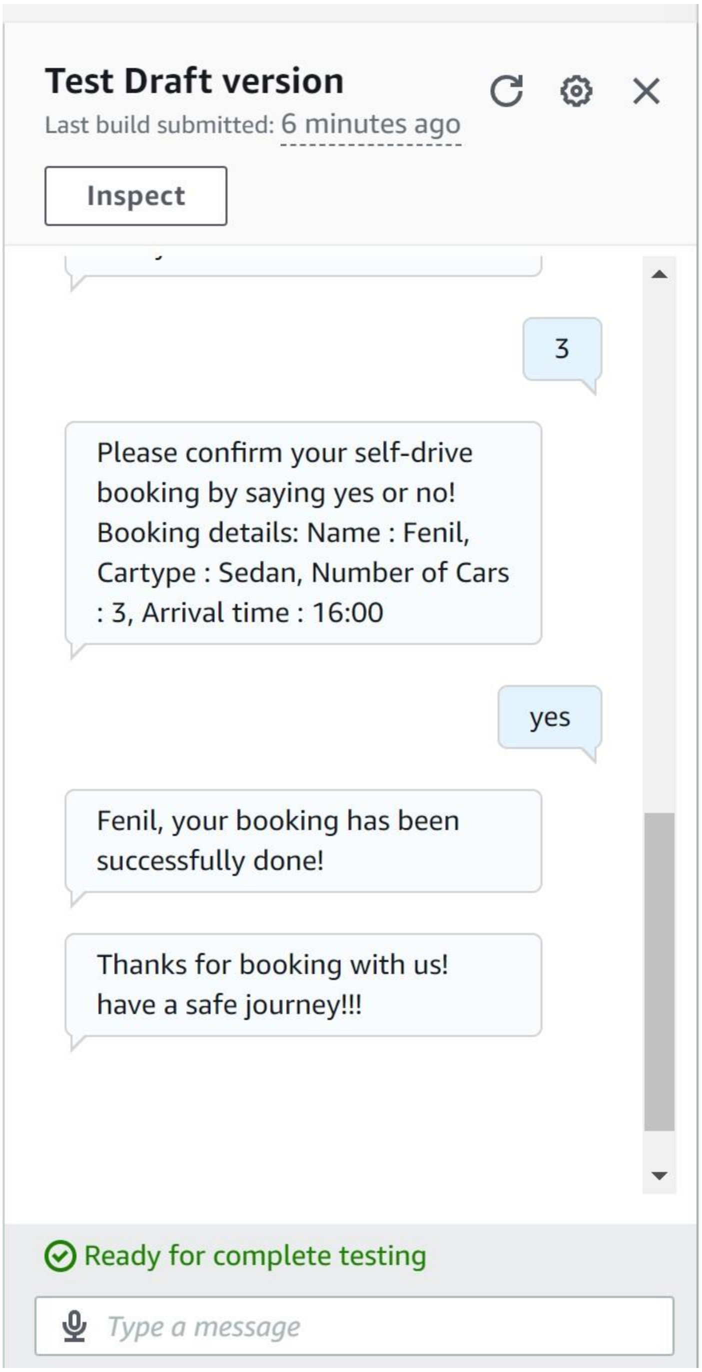
How many Sedan you want today?

3

Ready for complete testing

Fig 23: Testing of Bot

RideRequest bot created using AWS Lex



Test Draft version

Last build submitted: 6 minutes ago

Inspect

3

Please confirm your self-drive booking by saying yes or no! Booking details: Name : Fenil, Cartype : Sedan, Number of Cars : 3, Arrival time : 16:00

yes

Fenil, your booking has been successfully done!

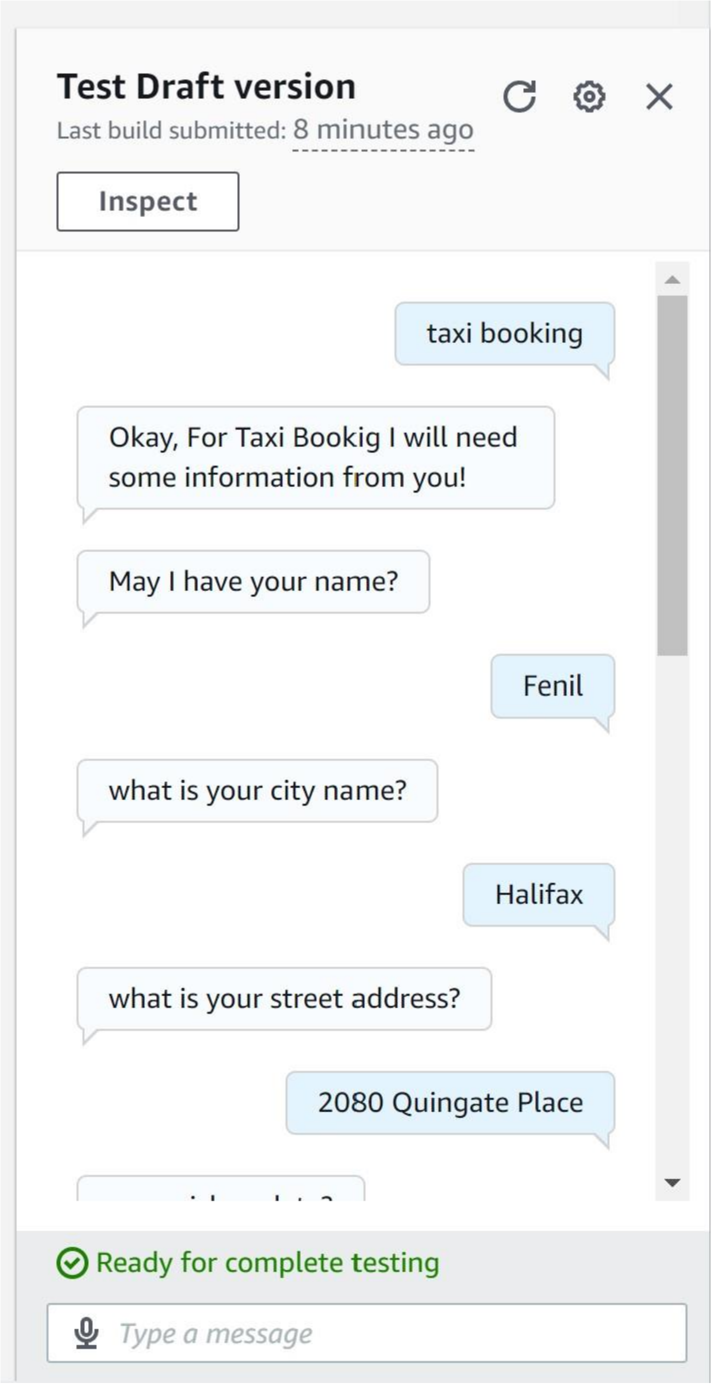
Thanks for booking with us! have a safe journey!!!

Ready for complete testing

Type a message

Fig 24: Testing of Bot

RideRequest bot created using AWS Lex



Test Draft version
Last build submitted: 8 minutes ago
Inspect

taxi booking

Okay, For Taxi Bookig I will need some information from you!

May I have your name?

Fenil

what is your city name?

Halifax

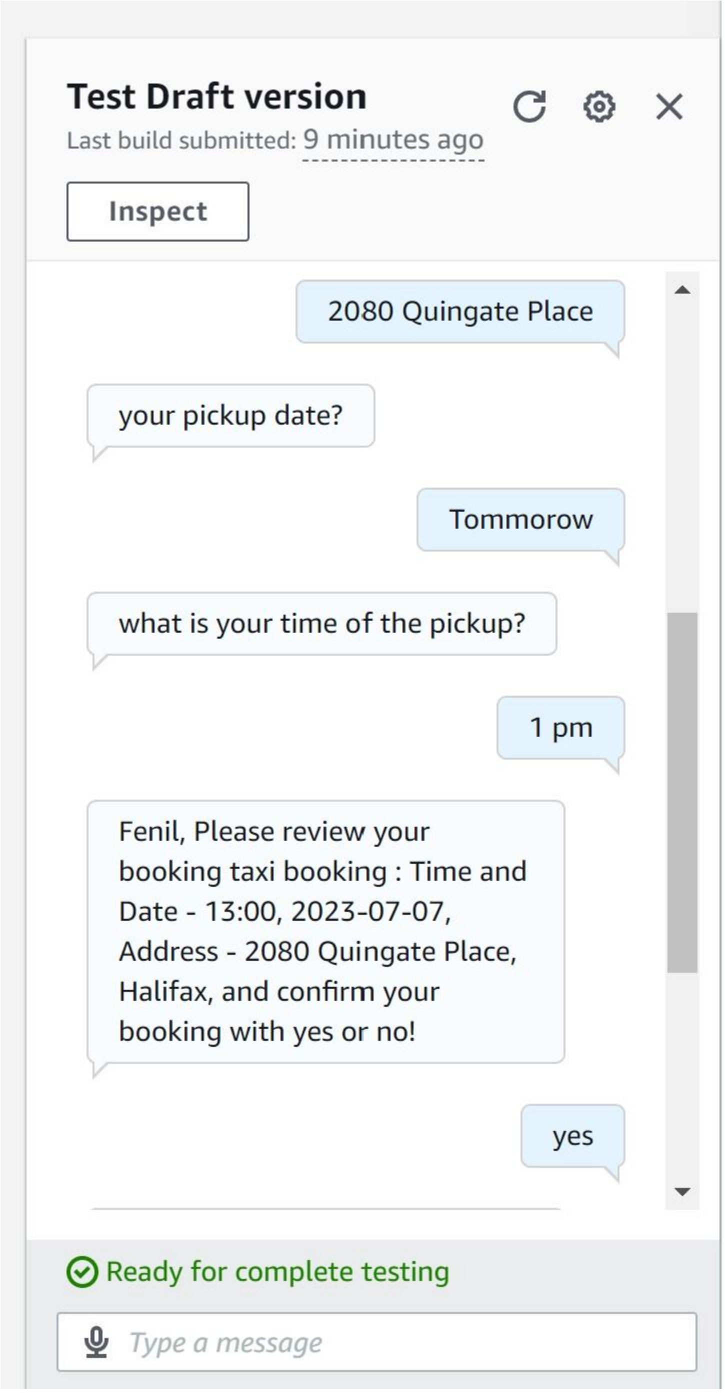
what is your street address?

2080 Quingate Place

Ready for complete testing
Type a message

Fig 25: Testing of Bot for Taxi Booking system

RideRequest bot created using AWS Lex



Test Draft version
Last build submitted: 9 minutes ago
Inspect

2080 Quingate Place

your pickup date?

Tommorow

what is your time of the pickup?

1 pm

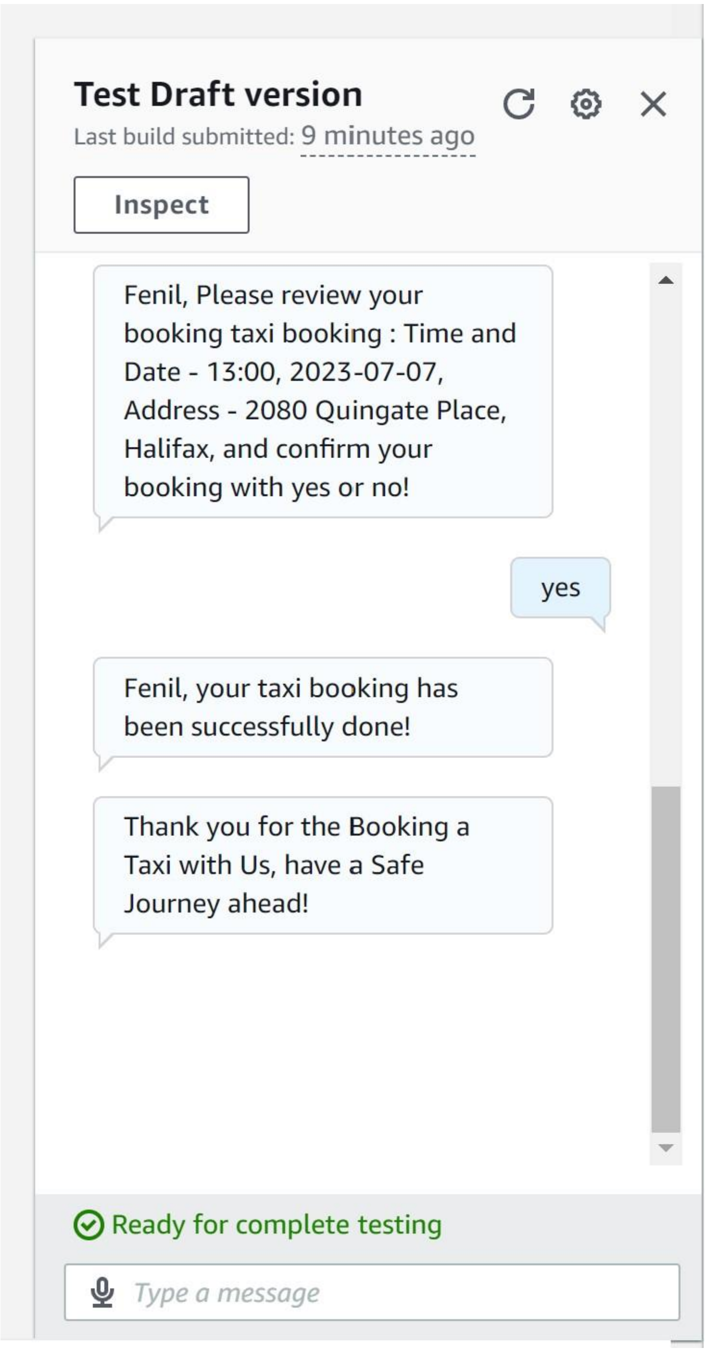
Fenil, Please review your booking taxi booking : Time and Date - 13:00, 2023-07-07, Address - 2080 Quingate Place, Halifax, and confirm your booking with yes or no!

yes

Ready for complete testing
Type a message

Fig 26: Testing of Bot

RideRequest bot created using AWS Lex



Test Draft version

Last build submitted: 9 minutes ago

Fenil, Please review your booking taxi booking : Time and Date - 13:00, 2023-07-07, Address - 2080 Quingate Place, Halifax, and confirm your booking with yes or no!

yes

Fenil, your taxi booking has been successfully done!

Thank you for the Booking a Taxi with Us, have a Safe Journey ahead!

Ready for complete testing

Fig 27: Testing of Bot

References:

How To Develop A Chat Bot Using Amazon Lex?
https://medium.com/edureka/how-to-develop-a-chat-bot-using-amazon-lex-a570beac969e