

Step 2: Creating your Bot

1. Create Amazon Lex Bot

Open the Amazon Lex console at <https://console.aws.amazon.com/lex/>.

On the Bots page, choose Create .

Select custom bot and provide the following information, then choose Create.

Provide The Following Information:

1. Bot Name: ScheduleAdjustment
2. Choose an output voice - None. This is only a text based application.
3. Set Session Timeout - 5 mins
4. Add AMazon lex basic role to your Bot app - AWSServiceRoleForLexBots

Bot name	ScheduleAdjustment
Description	<input type="text" value="No Description provided"/>
Language	English (US)
Output voice	<input type="text" value="None. This is only a text based application."/> ▼
Session timeout	<input type="text" value="5"/> <input type="text" value="min"/> ▼ ⓘ
IAM Role	AWSServiceRoleForLexBots ⓘ Automatically created on your behalf

Step 3: Creating your Bot Conversations


2. Create Slots

In this section you create the three slots type for ChatBot.

Create your first slot

First slot type is for to input the user you want to check the schedule.

Slot Type Name : Users

Add slot type 

Slot type name

Configure your first slot

Add the member name as values.

Also add the following, chatbot can specify all members at once.

- all
- all members
- everyone

When input is completed, save your slot.

Create your second slot

Second slot type is for to input the title of the schedule.

Slot Type Name : Title

Add slot type ×

Slot type name

Configure your second slot

Add the types of schedule as values.

The following is an example.

Edit slot type ×

Title Latest ▾

Value ⓘ

+

×

×

×

×

Cancel Save slot type Add slot to intent

When input is completed, save your second slot.

Create your third slot

Third slot type is for to input the meeting place.

Slot Type Name : Location

Add slot type

×

Slot type name

Configure your third slot

Add the name of the meeting room as values.

When input is completed, save your third slot.

3. Create Intent

This bot requires creation of two intents.

Create your First intent

First intent is for to check the member's schedule.

Intent Name : Reference

Create intent

Give a unique name for the new intent

Reference

Previous

Add

Configure your first Intent

Sample Utterances:

- I would like to check the schedule.
- check the schedule
- I would like to check the schedule on {Date}
- {member} schedule

Lambda initialization and validation:

▼ Lambda initialization and validation ⓘ

☒ Initialization and validation code hook

ScheduleAjustmentBotlamb... ▼

slots:

1.Member

Slot type:Users

Prompt:Who do you want to check the schedule?

2.Date

Slot type:AMAZON.DATE

Prompt:What day do you want to check the schedule?

▼ Slots ⓘ

Priority	Required	Name	Slot type		Prompt	
		<input type="text" value="e.g. Location"/>	<input type="text" value="e.g. AMAZO..."/>		<input type="text" value="e.g. What city?"/>	
1.	<input checked="" type="checkbox"/>	<input type="text" value="Member"/>	<input type="text" value="Users"/>	1	<input type="text" value="Who do you want to check ti"/>	
2.	<input checked="" type="checkbox"/>	<input type="text" value="Date"/>	<input type="text" value="AMAZON.DA..."/>	Built-in	<input type="text" value="What day do you want to ch"/>	

▼ Confirmation prompt ⓘ

☒ Confirmation prompt

Confirm

Cancel (if the user says "no")

▼ Fulfillment ⓘ

☒ AWS Lambda function ☐ Return parameters to client

☐ Goodbye message ☐ Follow-up message ☒ None

When you have configured, save your first intent.

Create your second intent

Second intent is for to add the schedule.

Intent Name : Add

Create intent

×

Give a unique name for the new intent

Add

Previous

Add

Configure your second Intent

Sample Utterances:

- I want to add the schedule.

Lambda initialization and validation:

▼

Lambda initialization and validation

?

☒

Initialization and validation code hook

ScheduleAjustmentBotlamb...

▼

slots:

1.Member

Slot type:Users

Prompt:Whose schedule would you like to add?

2.Date

Slot type:AMAZON.DATE

Prompt:Please tell me the schedule date.

3.StartTime

Slot type:AMAZON.TIME

Prompt:What time is the start?

4.EndTime

Slot type:AMAZON.TIME

Prompt:What time is the closing time?

5.Location

Slot type:Location

Prompt:Where is the meeting place?

6.Title

Slot type:Title

Prompt:What is the title of the meeting?

▼ Slots ⓘ

Priority	Required	Name	Slot type		Prompt		
		<input type="text" value="e.g. Location"/>	<input type="text" value="e.g. AMAZO..."/>		<input type="text" value="e.g. What city?"/>	⚙️	+
3.	▼	<input checked="" type="checkbox"/>	<input type="text" value="Member"/>	<input type="text" value="Users"/>	1 ▼	<input type="text" value="Whose schedule would you"/>	⚙️ ✖️
4.	^ ▼	<input checked="" type="checkbox"/>	<input type="text" value="Date"/>	<input type="text" value="AMAZON.DA..."/>	Built-in ▼	<input type="text" value="Please tell me the schedule"/>	⚙️ ✖️
5.	^ ▼	<input checked="" type="checkbox"/>	<input type="text" value="StartTime"/>	<input type="text" value="AMAZON.TIME"/>	Built-in ▼	<input type="text" value="What time is the start?"/>	⚙️ ✖️
6.	^ ▼	<input checked="" type="checkbox"/>	<input type="text" value="EndTime"/>	<input type="text" value="AMAZON.TIME"/>	Built-in ▼	<input type="text" value="What is the closing time?"/>	⚙️ ✖️
7.	^ ▼	<input checked="" type="checkbox"/>	<input type="text" value="Location"/>	<input type="text" value="Location"/>	1 ▼	<input type="text" value="Where is the meeting place?"/>	⚙️ ✖️
8.	^	<input checked="" type="checkbox"/>	<input type="text" value="Title"/>	<input type="text" value="Title"/>	1 ▼	<input type="text" value="What is the title of the meeti"/>	⚙️ ✖️

▼ Confirmation prompt ⓘ

☒ Confirmation prompt

Confirm

Cancel (if the user says "no")

▼ Fulfillment ⓘ

☒ AWS Lambda function ☐ Return parameters to client

☒ Goodbye message ☐ Follow-up message ☐ None

When you have configured, save your second intent.

Error Handling

Provide The Following Information:

Clarification prompts:Sorry, can you please repeat that?

Maximum number of retries:5

Hang-up phrase:Sorry, I could not understand. Goodbye.

The screenshot shows a web interface for configuring a chatbot's error handling. At the top, there's a header bar with a back arrow and the text 'ScheduleAdjustment'. Below this is a tabbed interface with four tabs: 'Editor' (selected), 'Settings', 'Channels', and 'Monitoring'. On the left side, under the 'Editor' tab, there's a sidebar menu with the following items: 'Intents' (with a plus icon), 'Add', 'Reference', 'Slot types' (with a plus icon), 'Location', 'Title', 'Users', and 'Error Handling' (which is highlighted with an orange bar). The main content area is titled 'Error handling' and contains the following fields: 'Clarification prompts' (with a text input field containing 'e.g. Sorry, can you please repeat that?' and a plus icon to add more), 'Maximum number of retries' (with a text input field containing '5'), and 'Hang-up phrase' (with a text input field containing 'e.g. Sorry, I could not understand. Please contact customer support.' and a plus icon to add more). Below these fields is a 'Save' button.

Step 4. Build

When you have finished configuration , click on build to build your chatbot.

The screenshot shows a dialog box titled 'Build your bot' with a close button (X) in the top right corner. Below the title bar, there's a message: 'You can start testing the Bot once the build completes successfully.' At the bottom right of the dialog, there are two buttons: 'Cancel' and 'Build'.

Step 5. Test App

When the build is complete, test your chatbot.

Test Bot

Build: **Latest** | Status: READY

I would like to check the schedule.

Who do you want to check the schedule?

Clear

Type to your bot...

Step 6. Publish App

Once your Bot is build and tested the next step is to publish our Bot and make it avaialble on Slack.To publish your Bot you need to create an Alias.

Publish ScheduleAdjustment

Publishing is the last step before you can connect your bot to your mobile app or chatbot.

Choose an alias Choose an alias

Create a new alias

Cancel Publish