BUILD A CHATBOT WITH AMAZON LEX

Building Better Bots

Code samples related to [Building Better Bots](https://aws.amazon.com/blogs/ai/building-better-bots-part-2/) published on the AWS ML Blog

SodaBot

SodaBot is a transactional chat bot that can help one order a soda (relies on AWS Amplify and Android).

Consider this conversation:

User: May I have a soda?  
SodaBot: What size? small, medium, large?  
User: small  
SodaBot: What kind of soda would you like?  
User: Coke  
SodaBot: You’d like me to order a small Coke. Is that right?  
User: Make it a large Coke  
SodaBot: You’d like me to order a large Coke. Is that right?  
User: yeah  
SodaBot: Great! Your Coke will be available for pickup soon. Thanks for using SodaBot!

To get started, we’ll first create the Amazon Lex bot. Then, we’ll add some Lambda Functions to bring it to life.

Amazon Lex bot

1. Create bot

1. From the Amazon Lex console, create a Custom bot with these settings (you can see these in the “Settings” tab later)
   * Bot name: SodaBot
     + To work independently in a shared environment, use your initials in the name (e.g., SodaBotXXX)
   * Output voice: Joanna
   * Session timeout: 5 min
   * Sentiment Analysis: (accept the default) No
   * IAM role: (accept the default) AWSServiceRoleForLexBots
   * COPPA: (our bot is not directed at children) No

2. Create Order Beverage Intent

From the left, click on the “+” sign next to the “Intents” section and add a new Intent called sodaOrderBeverageIntent with the following settings and click “Add” to save the Intent.

Sample Utterances: add these to the list of sample utterances so the bot recognizes similar phrases (each entry on a separate line). Lambda initialization and validation (leave unchecked)

* I would like a {BeverageSize} {BeverageType}
* Can I get a {BeverageType}
* May I have a {BeverageSize} {MixIn} {BeverageType}
* Can I get a {BeverageSize} {MixIn} {BeverageType}
* Let me get a {BeverageSize} {MixIn} {BeverageType}

Click “Save Intent” at the bottom to save the utterances.

3. Create Slot types

Add the following Slot types (each value should be a separate entry); remember to “Save slot type” as you go along. To work independently in a shared environment, use your initials in the names (e.g., sodaBeverageTypeXXX).  
Note: Although they are saved with the AWS Account, Slot Types will only show up in the list when they are associated in the next step.

| **Slot type name** | **Description** | **Values (each entry on a separate line)** |
| --- | --- | --- |
| sodaBeverageType | *Slot types are shared at the account level so text would help other developers determine if they can reuse this Slot type.* | Coke; Pepsi; Dr. Pepper; Sprite; Lemonaid  \*\* each entry on a separate line\* |
| sodaBeverageSize |  | small; medium; large |
| sodaMixIn |  | dirty; vanilla; cherry |

4. Lambda initialization and validation (leave unchecked)

5. Add Slots to the Intent

Add the following entries to the list of Slots, choosing the Slot Types created above. If you do not see the new slot types in the dropdown, refresh the browser. Click “Save Intent”.

| **Required** | **Name** | **Slot type** | **Prompt** |
| --- | --- | --- | --- |
| Yes | BeverageType | cafeBeverageType | What kind of beverage would you like? For example, mocha, chai. |
| Yes | BeverageSize | cafeBeverageSize | What size? small, medium, large? |
|  | MixIn | cafeMixInType | Would you like cherry, vanilla, or dirty? |

6. Add confirmation prompt

**Confirmation prompt:**  
“You’d like me to order a {BeverageSize} {BeverageType}. Is that right?” to confirm. “Okay. Nothing to order this time. See you next time!” to cancel

7. Choose Fulfillment option

**Fulfillment:**  
choose “Return parameters to client” for now

8. Add response messagee

**Response:**  
Select Add Message to add a message(s) to close the intent: Thank you. Your {BeverageType} has been ordered. Check the box for Wait for user reply type: OK. Thank you. Have a great day!

Click “Save Intent” to save all entries

9. Review Error Handling

Select Error Handling from the left

Clarification prompts: “Sorry, but I didn’t understand that. Would you try again, please?””

Maximum number of retries: 2

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

Click “Save”

10. Build and Test the Bot

Build the app by clicking the build button at the top right. To test the bot with some of the utterances, expand the Test Chatbot dialog at the top right corner of the Amazon Lex Console. For example, if you say May I have a Sprite? does Lex correctly map Sprite to the BeverageType slot? For example, if you say May I have a Sprite?, does Lex correctly map Sprite to the BeverageType slot?

Lambda Function

Go to AWS Management Console and click on “Lambda” to launch lambda console.

1. Create the sodaOrder function by saving sodaOrder\_lambda.js as a Node.js 12.x function.
   * To work independently in a shared environment, use your initials in the function name (e.g., cafeOrderCoffeeXXX)
   * Choose an IAM role that includes the AWSLambdaBasicExecutionRole Managed Policy. If no such role exists, you can create a new IAM Role using one of these approaches:
     + Choose “Create a new role with basic Lambda permissions”.
2. Overwrite the index.js function with the function source [here](https://ml-immersionday.workshop.aws/labs/chatbot/index.js).
   * (No need to set up a trigger; you can accept default values for most of the configuration)
3. Click on “Save” to save the function.

Test the bot

a. From the Lex Console, select the SodaBot bot and choose Latest from the version drop down to make changes  
b. Modify the sodaOrderBeverageIntent Intent

* Associate it with the new sodaOrder Lambda function (select “Lambda function” in the “Lambda initialization and validation” area). When prompted, allow Amazon Lex to call your new function
* Associate it with the new sodaOrder Lambda function for (select “Lambda function” in the “Fulfillment” area); remember to click “Save Intent”

c. Click on “Build” to build the bot  
d. Test using the Amazon Lex Console; do you see any responses when you ask May I have a Coke?