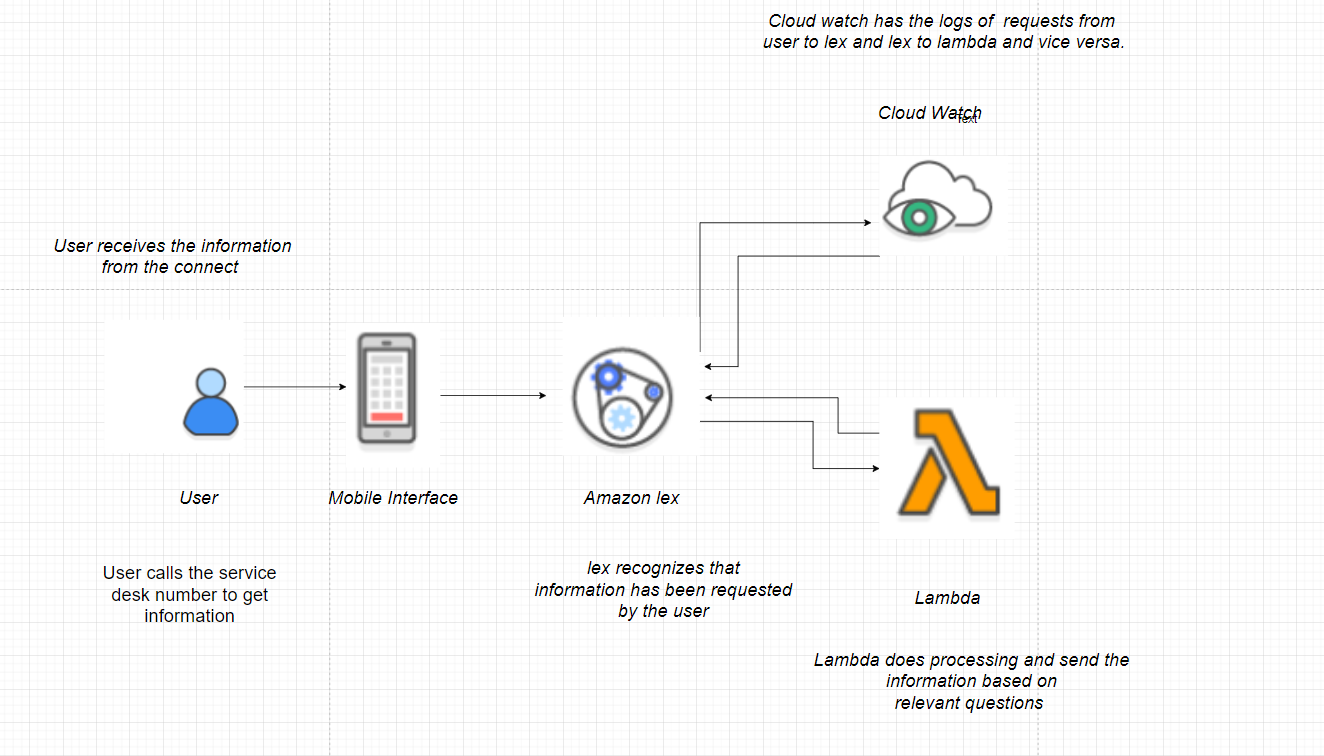
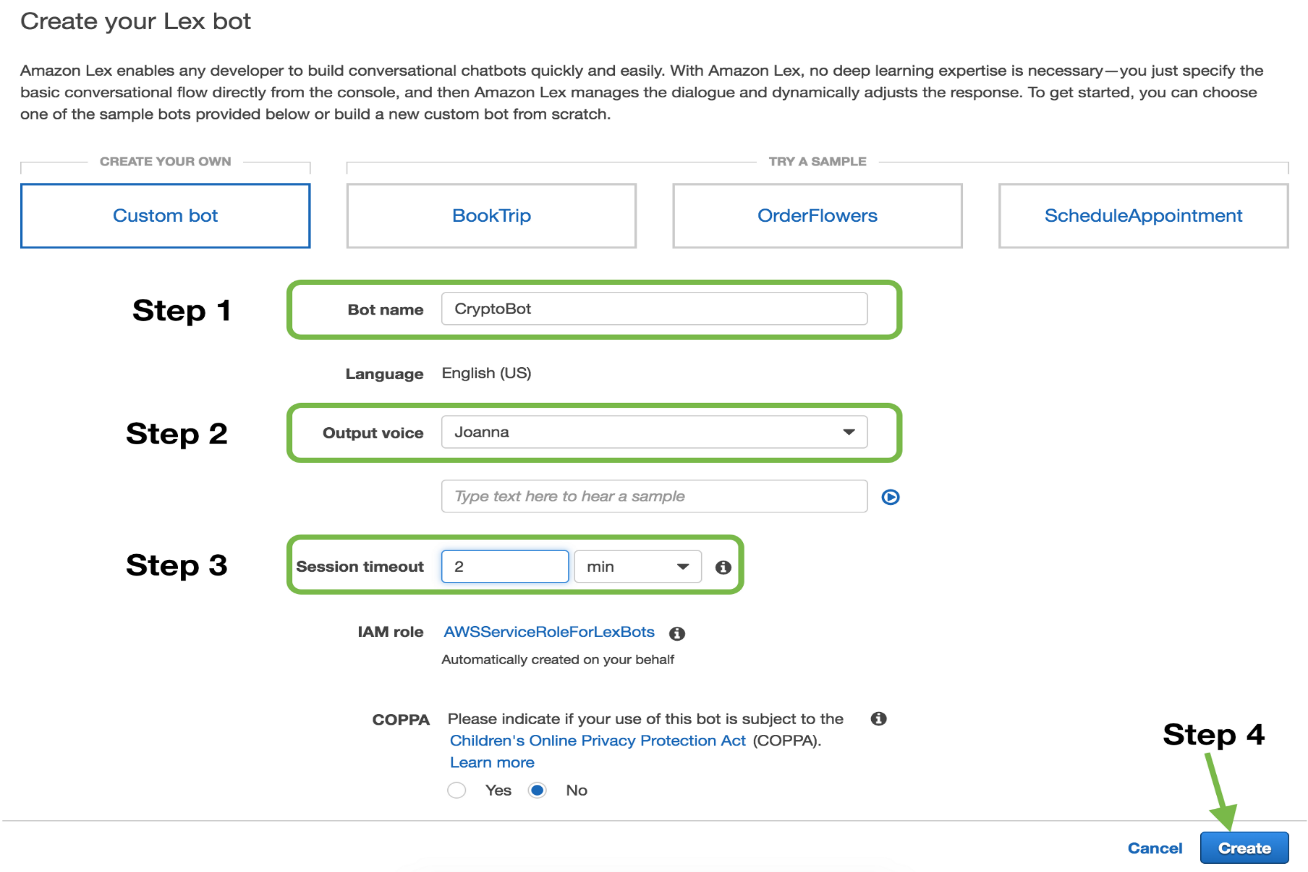
**How to create chat bot using AMZON LEX and LAMBDA functions**



**Step 1: Create a custom lex bot**

We’ll start off by creating a custom bot.



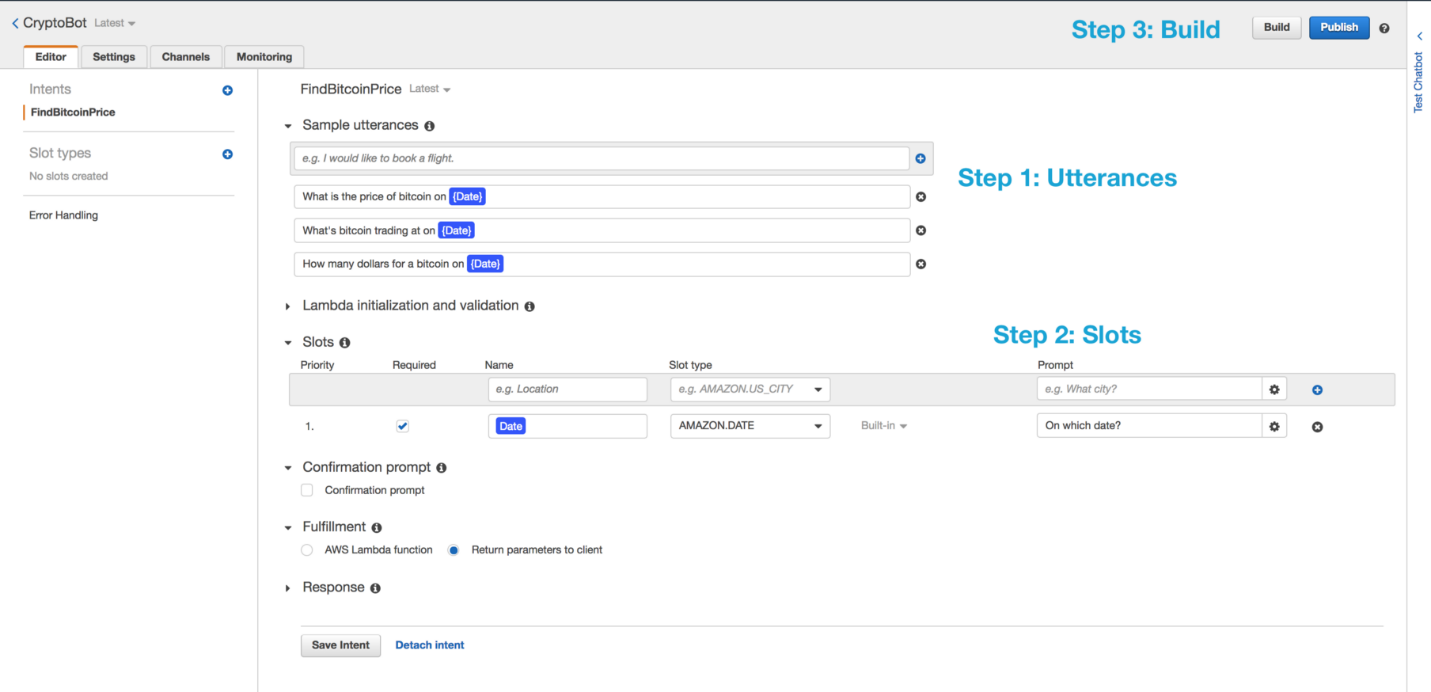
Create a custom lex bot

**Step 2: Building the language model**

Here’s where we give our bot the ability to understand conversations. Some terms used here:

1. ***Intent***: Intent is a skill the bot has. Our bot has one skill for now i.e. *FindBitcoinPrice*.
2. ***Utterances***: Sentences to invoke an intent. For find bitcoin price intent, an example utterance would be *“what is the price of bitcoin on {Date}?”*
3. ***Slots***: Values user must supply to an intent. In this example, it is the *“{Date}”* on which the user wants bitcoin’s value. Amazon provides a lot of [built-in slot types](https://developer.amazon.com/docs/custom-skills/slot-type-reference.html#list-slot-types), we can use `AMAZON.DATE` for this slot.

Let’s fill these up and build our bot.



Lex bot model

Build will take a couple of seconds. You can now test your bot by asking it a question in ***Test Chatbot***section.

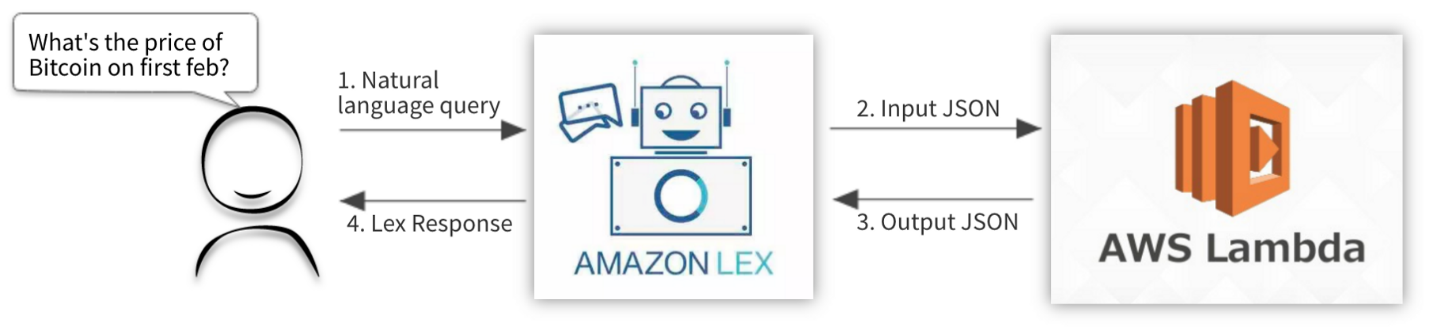
Try asking *“what’s the price of bitcoin on first feb?”.*

**Step 3: Adding the logic**

Now our bot is able to understand questions and get inputs, we need to add “the logic to perform the skill” i.e. getting the dollar value of bitcoin. We’ll be adding this logic through [**AWS Lambda**](https://aws.amazon.com/lambda/?hp=tile&so-exp=below).

*AWS Lambda lets us run code without having to create & host a server. Read more about “serverless” computing.*

***Workflow with lambda:***

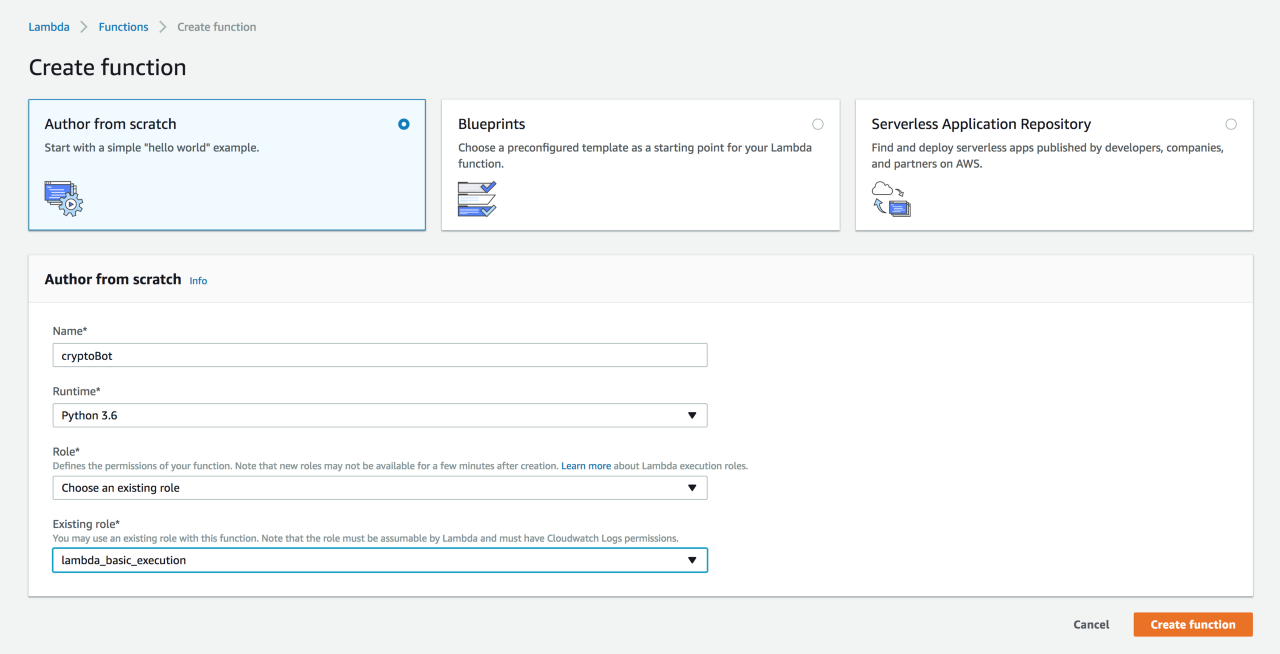


Workflow with lambda

Once lex understands user’s sentence, it produces an input JSON. This input JSON goes to the code performing logic. Lex expects an output JSON in return. *Both the input and output JSON have specific formats described in detail* [*here*](https://docs.aws.amazon.com/lex/latest/dg/lambda-input-response-format.html). First let’s create a lambda to add our logic.

**Step 3.a Create a Python 3.6 lambda**

On your AWS console, create a lambda with python 3.6 runtime. Choose a role that has access to Cloudwatch logs (useful for lambda function logs.)



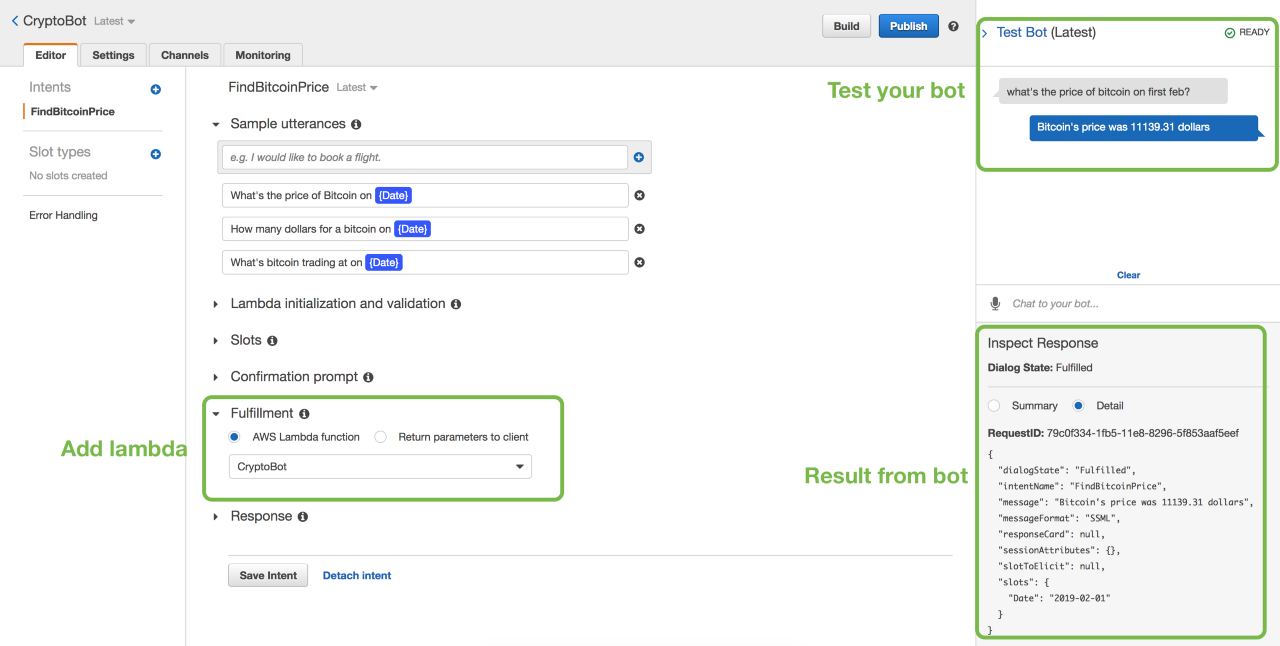
Create Python 3.6 lambda function

**Step 3.b Write python code for processing request**

This is the input JSON we get from lex on asking the price of bitcoin.

**Step 3.c Connect lambda to lex bot**

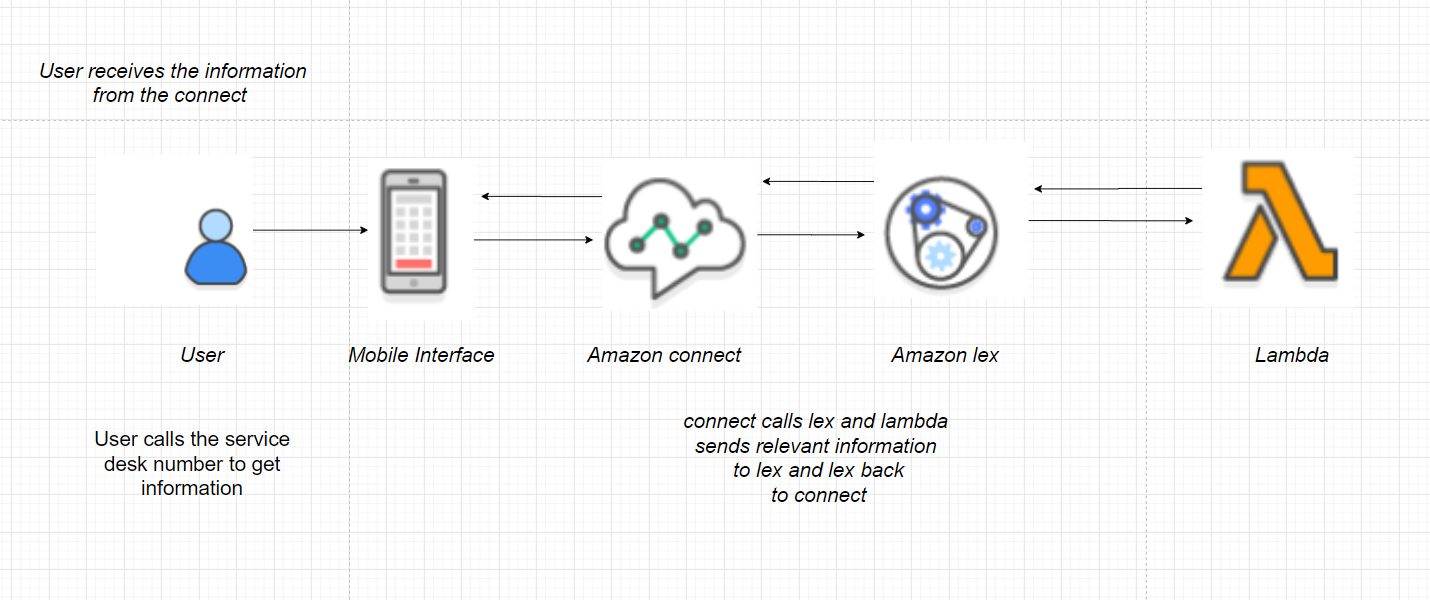
On the lex console, add the above lambda in Fulfillment section.



Connecting lambda to lex bot

Save and build the bot.

**How the Amazon connect is used to connect user to call center:**



## Add the Amazon Lex Bot to an Amazon Connect Instance

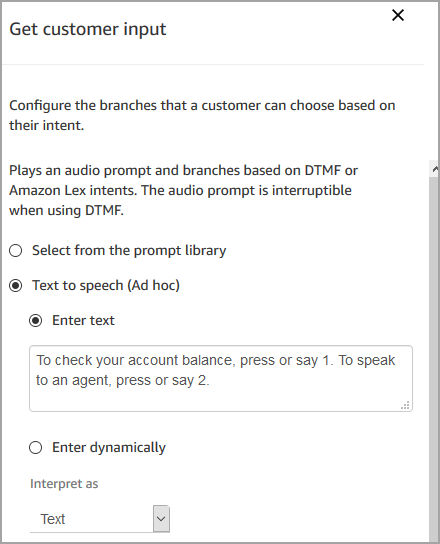
Before you can use a bot in your contact flow you need to add it to your Amazon Connect instance. You can only add bots created under the same AWS account.

If you add Amazon Lex bots created in a different Region from your instance, performance may be affected.

1. Open the [Amazon Connect console.](https://console.aws.amazon.com/connect/)
2. Select the **Instance Alias** of the instance to which to add the bot.
3. Choose **Contact flows**.
4. Under **Amazon Lex**, use the drop-down to choose a name for your bot and then choose **+ Add Lex Bot**.
5. Select the **AccountBalance** bot and choose **Save Lex Bots**. If the name of your bot doesn't appear in the list, reload the page to get it to show up.

## Create a Contact Flow and Add Your Amazon Lex Bot

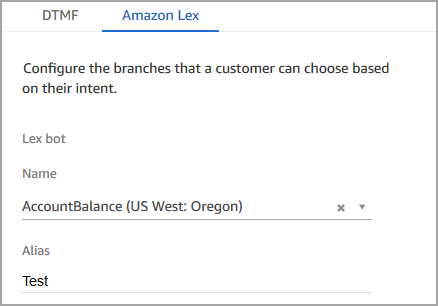
Next, create a new contact flow that uses your Amazon Lex bot. When you create the contact flow, you configure the message played to callers.

1. Log in to your Amazon Connect instance with an account that has permissions for contact flows and Amazon Lex bots.
2. Choose **Routing, Contact flows, Create contact flow**, and type a name for the flow.
3. Under **Interact**, drag a **Get customer input** block onto the designer, and connect it to the **Entry point block**.
4. Click the **Get customer input** block to open it. Choose **Text to speech (Ad hoc), Enter text**.
5. Type a message that provides callers with information about what they can do. For example, use a message that matches the intents used in the bot, such as “To check your account balance, press or say 1. To speak to an agent, press or say 2.” 

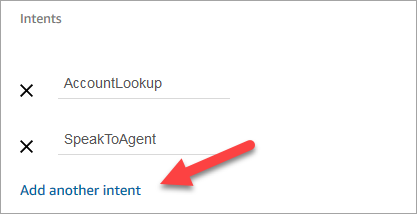
### Add the Amazon Lex Bot to Your Contact Flow

In this step you'll specify the bot as the method of getting customer input.

1. In the **Get customer input** block select **Amazon Lex**.
2. For **Name**, use **AccountBalance**. For **Alias**, use **Test**.



1. Under **Intents**, choose **Add an intent**.
2. Type **AccountLookup** and choose **Add another intent**.



1. Type **SpeakToAgent** and choose **Save**.

### Finish the Contact Flow

After the caller interacts with the bot, finish the contact flow to complete the call for the customer.

1. If the caller presses 1 to get their account balance, use a **Prompt** block to play a message and disconnect the call.
2. If the caller presses 2 to speak to an agent, use a **Set queue** block to set the queue and transfer the caller to the queue, which ends the contact flow.

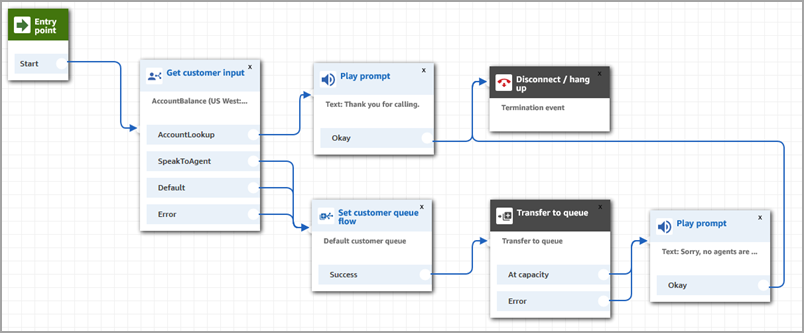
To complete the **AccountLookup** intent:

1. Under **Interact**, drag a **Play prompt block** to the designer, and connect the **AccountLookup** node of the **Get customer input** block to it. After the customer gets their account balance from the Amazon Lex bot, the message in the **Play prompt** block plays.
2. Under **Terminate/Transfer**, drag a **Disconnect/hang up** block to the designer, and connect the **Play prompt** block to it. After the prompt message plays, the call is disconnected.

To complete the **SpeakToAgent** intent:

1. Add a **Set customer queue** block and connect it to the **SpeakToAgent** node of the **Get customer input** block.
2. Add a **Transfer to queue** block.
3. Connect the Success node of the **Set customer queue** block to the **Transfer queue**.
4. Choose **Save**, then **Publish**.

Your finished contact flow will look something like the following one:



## Assign the Contact Flow to a Phone Number

When callers call in to your contact center, the contact flow to which they are sent is the one assigned to the telephone number that they dialed. To make the new contact flow active, assign it to a phone number for your instance.

1. Open the Amazon Connect Dashboard.
2. Choose **View phone numbers**.
3. Select the phone number to which to assign the contact flow.
4. Add a description.
5. In the **Contact flow/IVR** menu, choose the contact flow that you just created.
6. Choose **Save**.