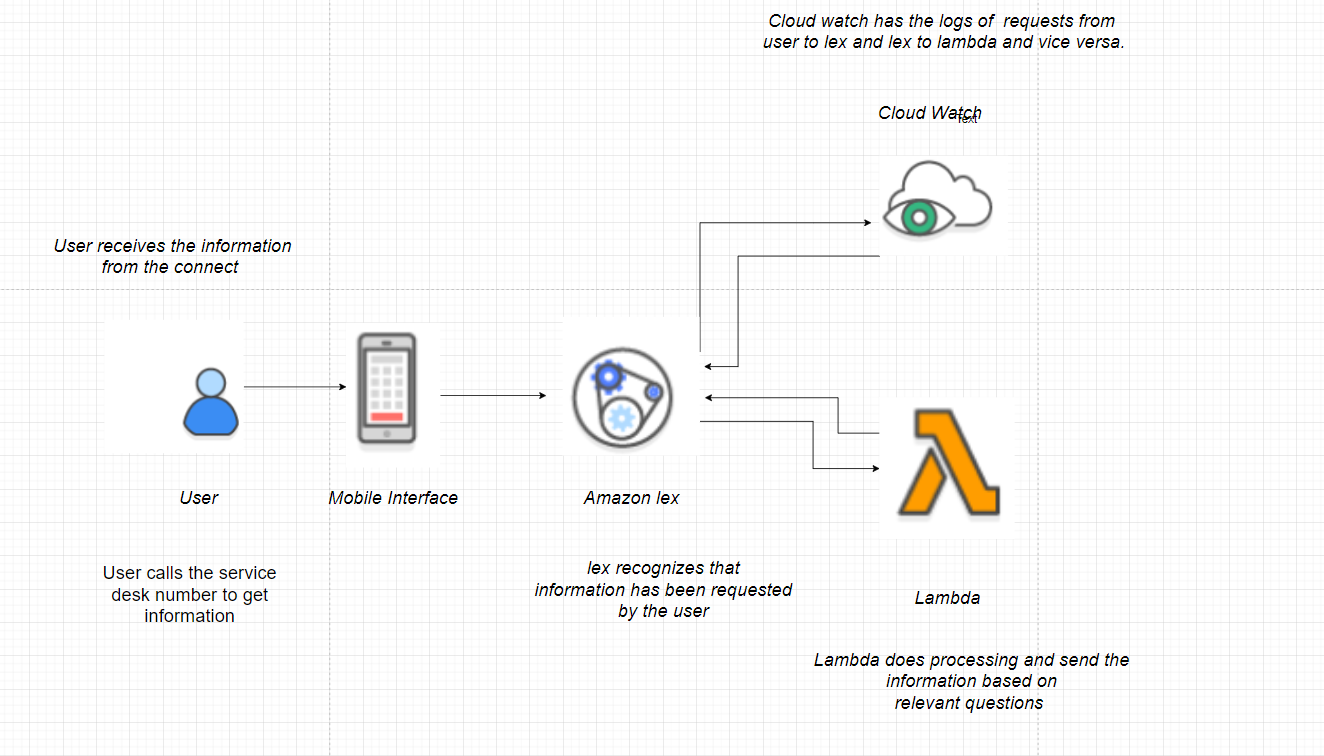
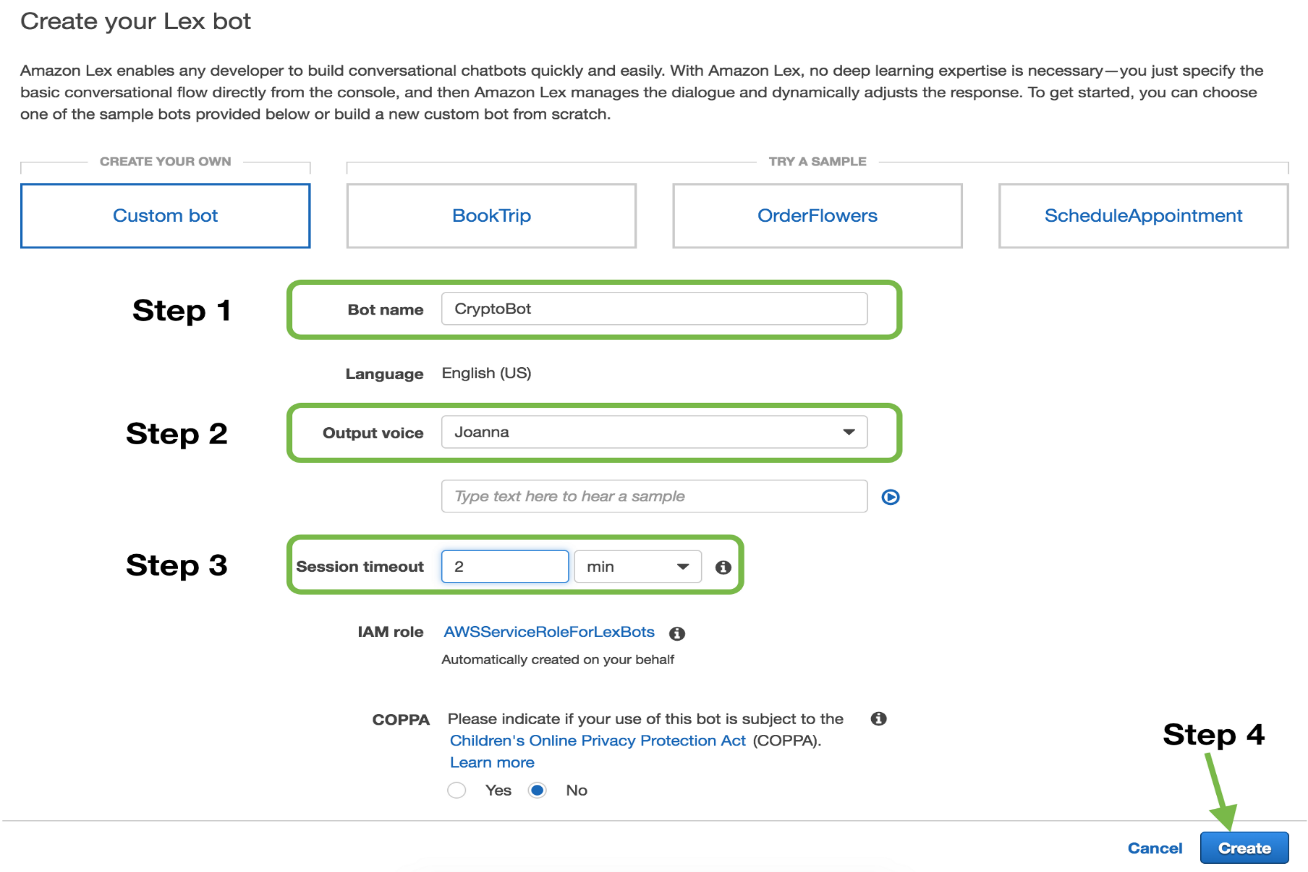
**How to create chat bot using AMZON LEX and LAMBDA functions**



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

We’ll start off by creating a custom bot.



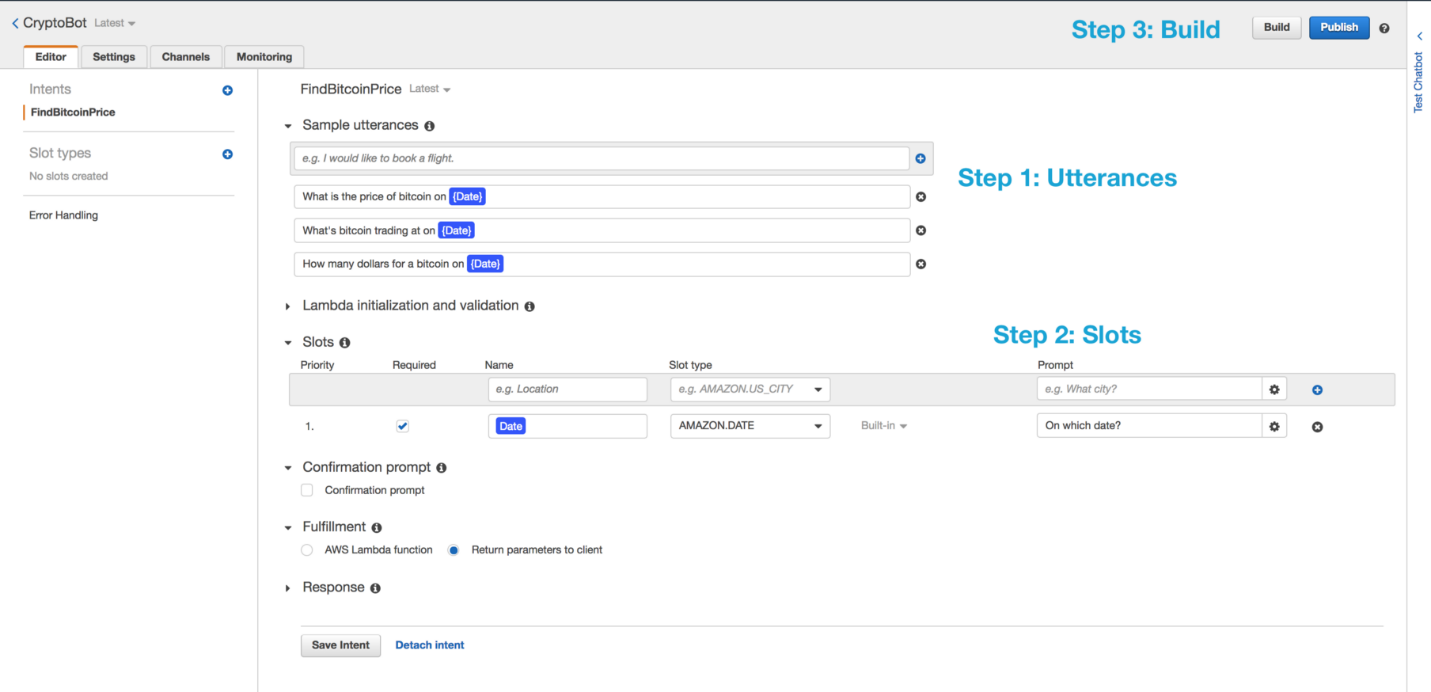
Create a custom lex bot

**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. **dueDateForMyBalance**.
2. ***Utterances***: Sentences to invoke an intent. For find bitcoin price intent, an example utterance would be *“*could I know the last date for my balance at CRA*”*
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.or we can create the new slots and slottypes.

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.

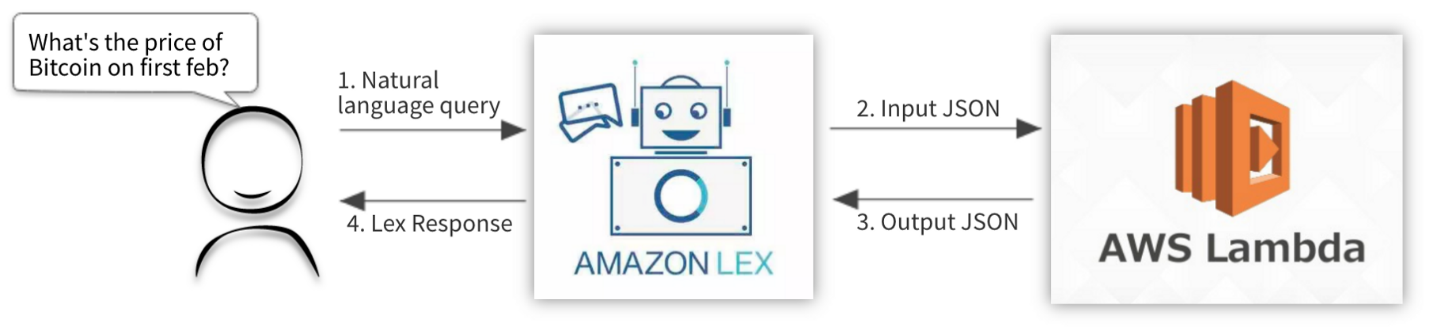
Try asking *“what is due date for my balance ?”.*

**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). here in this project we have used the nodeJS to define the logic.

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

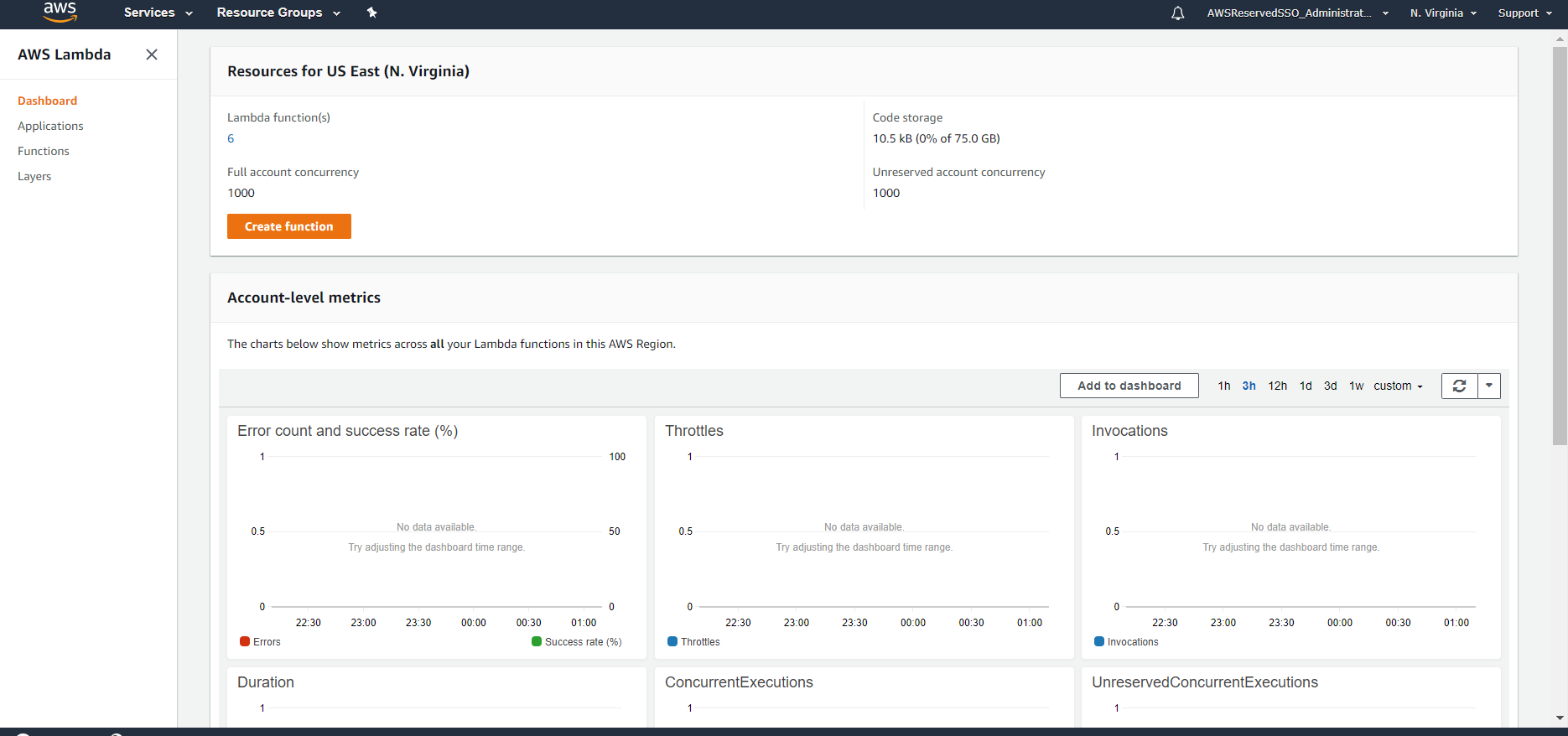
***Workflow with lambda:***



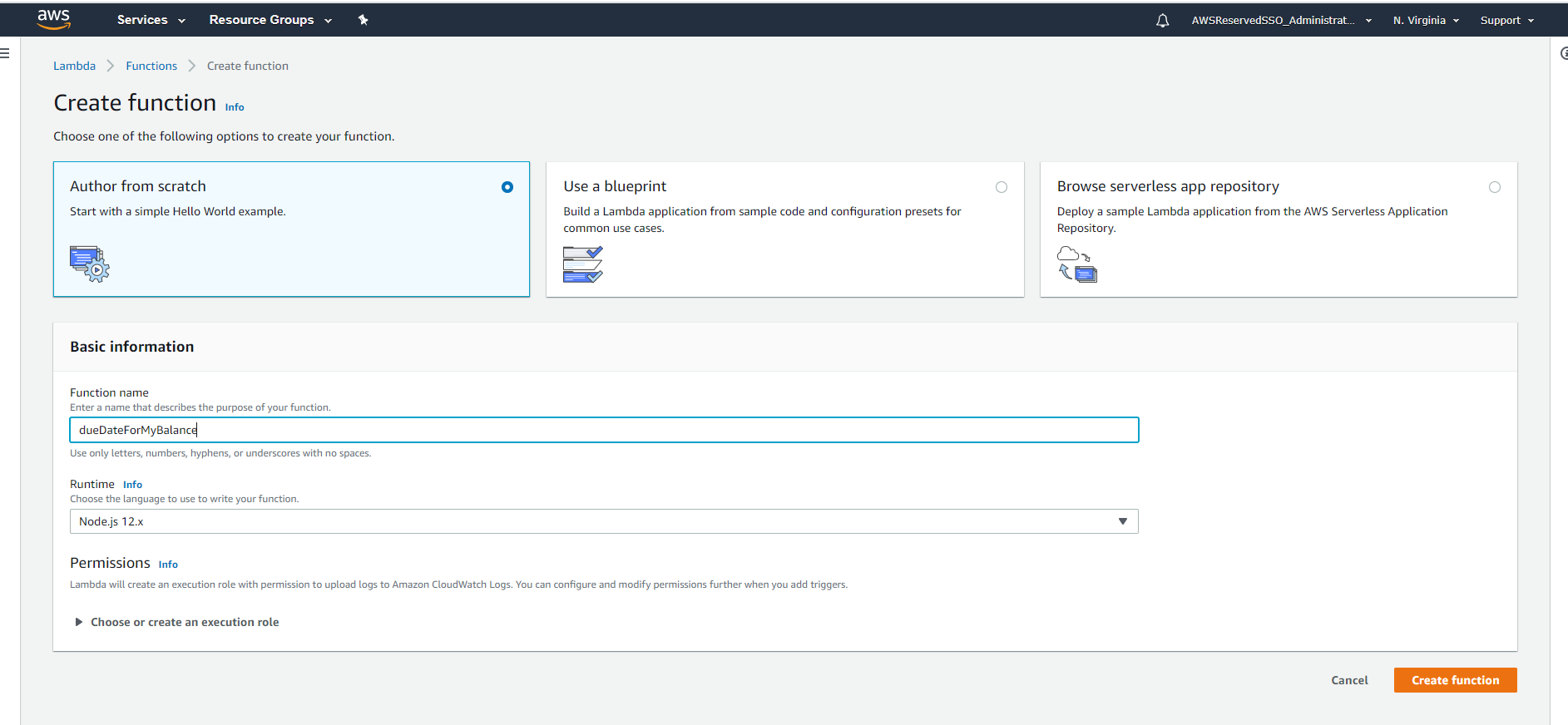
Workflow with lambda

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 Node JS lambda**



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



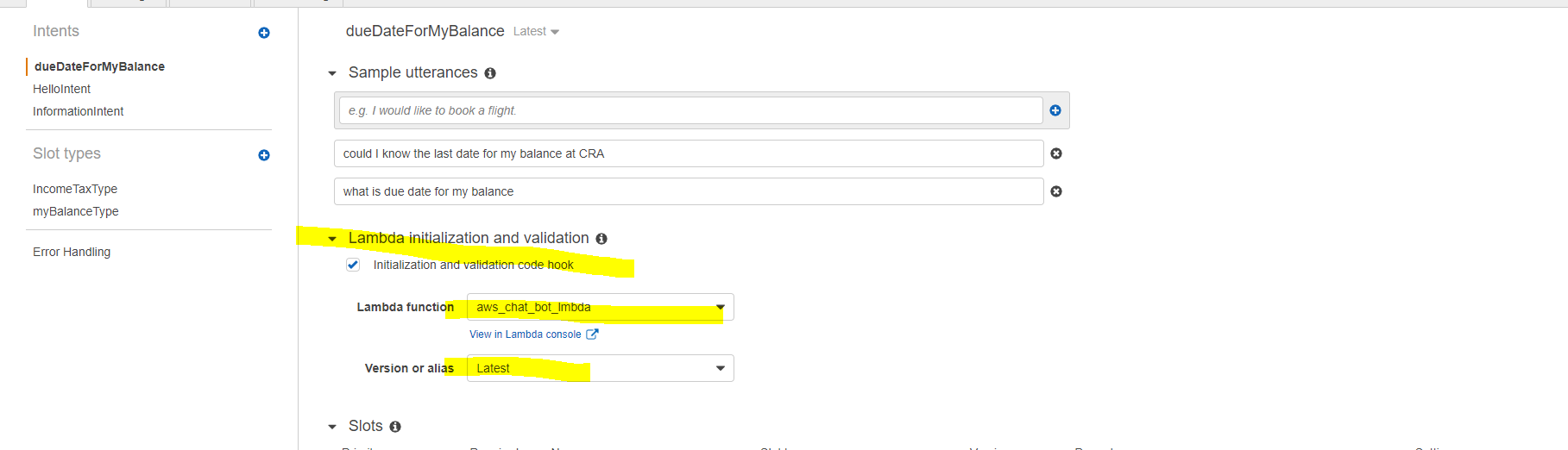
Create node.js 12x lambda function

**Step 3.b Write node js code for processing request**

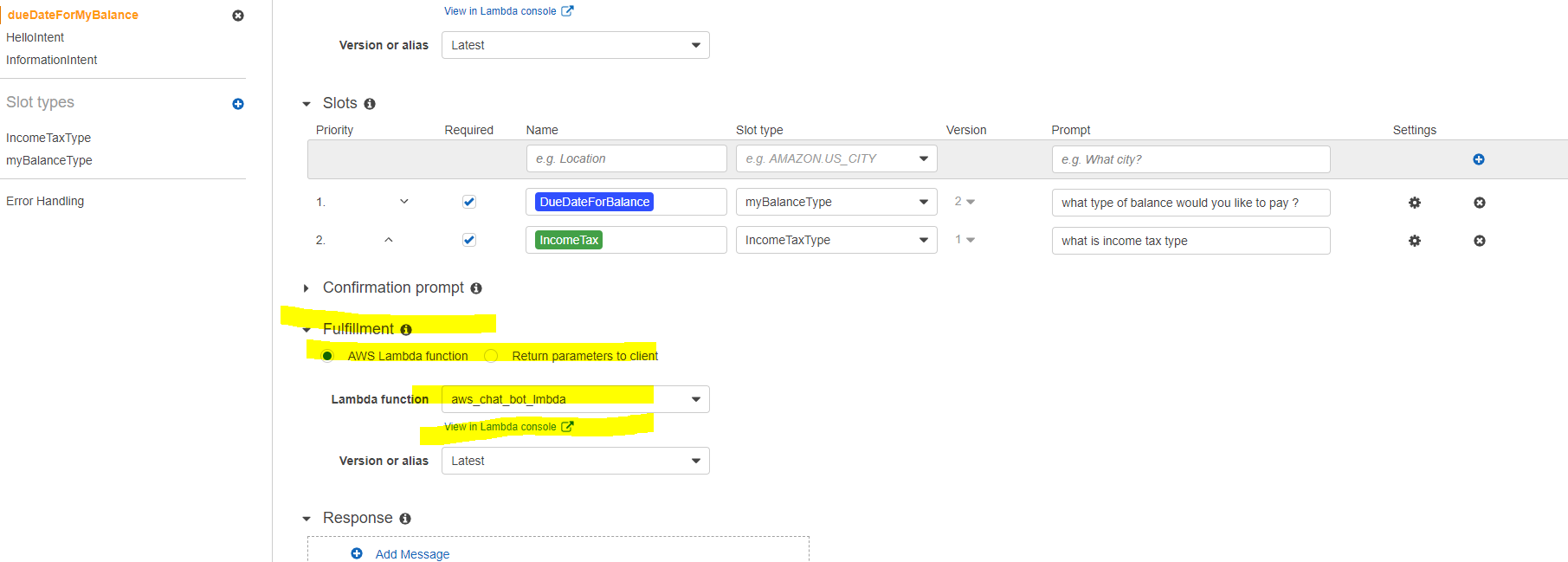
This is the input JSON we get from lex on what is due date for my balance?

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

For validation of slots we can also define a lambda function and use this in lex for validation.



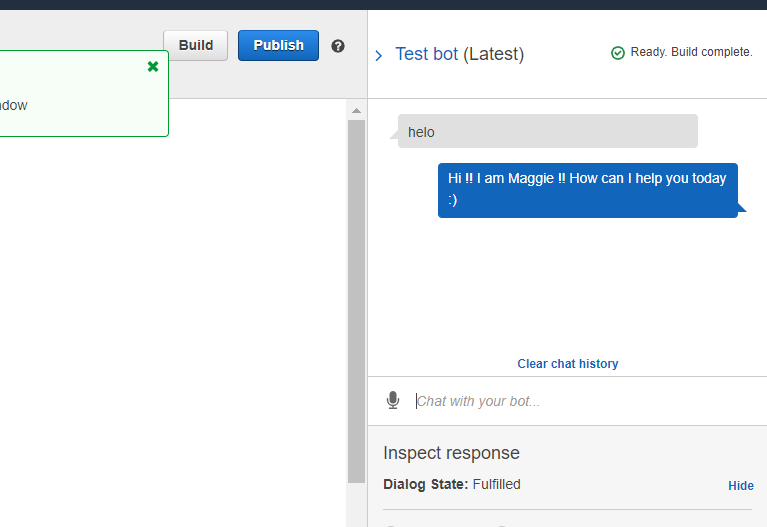
On the lex console, add the above lambda in Fulfillment section.



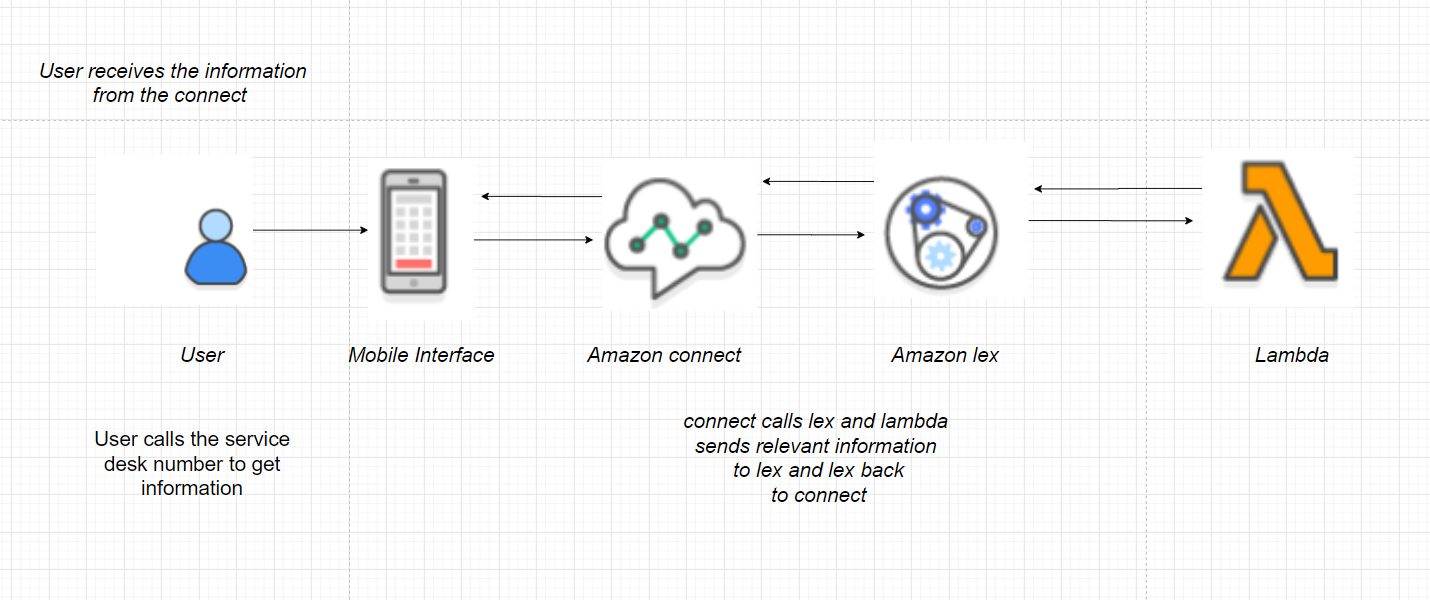
Connecting lambda to lex bot

Save and build the bot.

Test the chat bot by using testbot on right hand side of console .



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

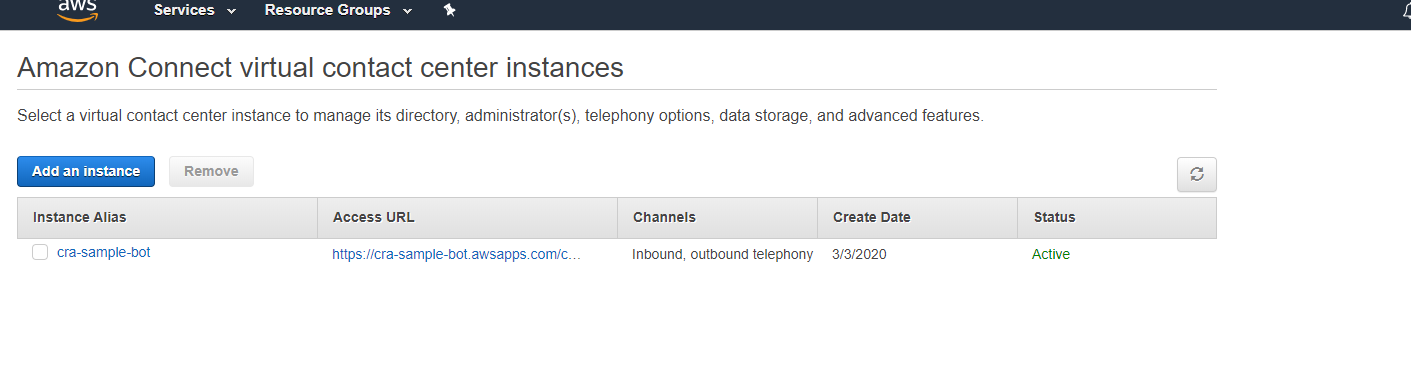


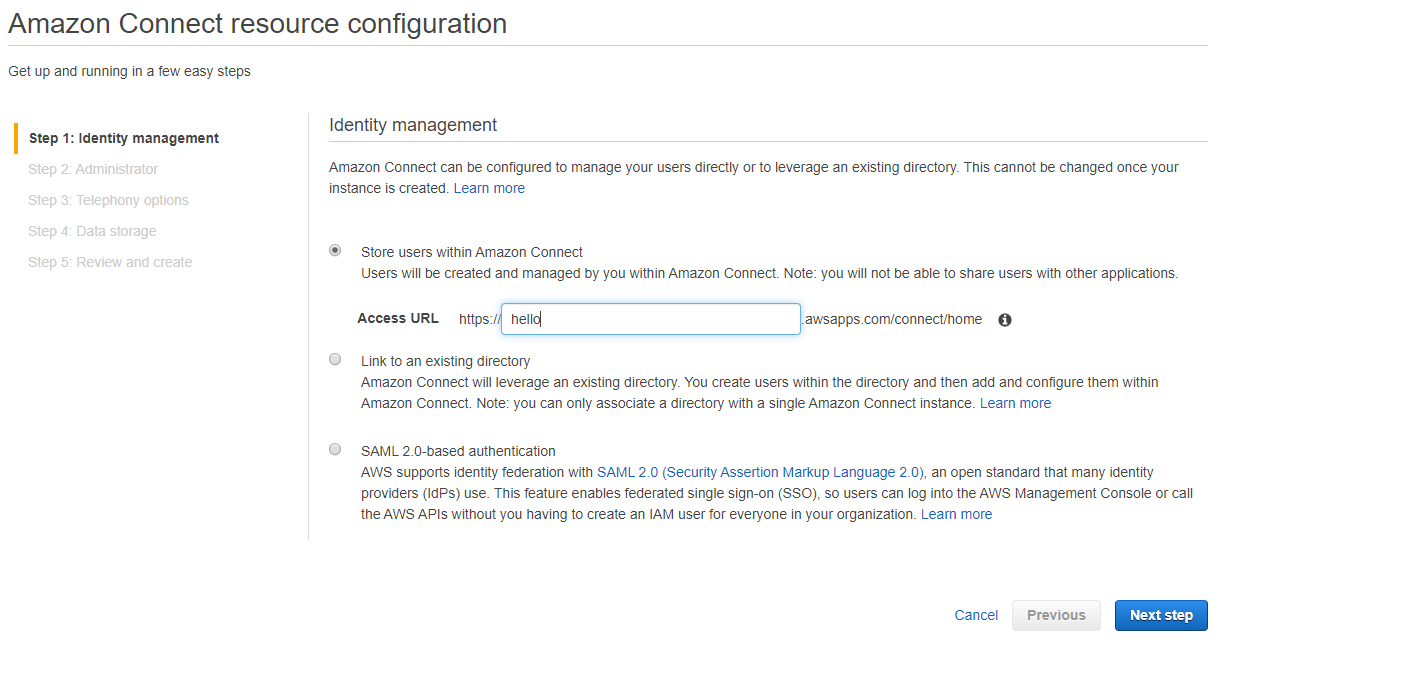
## Add the Amazon Lex Bot to an Amazon Connect Instance

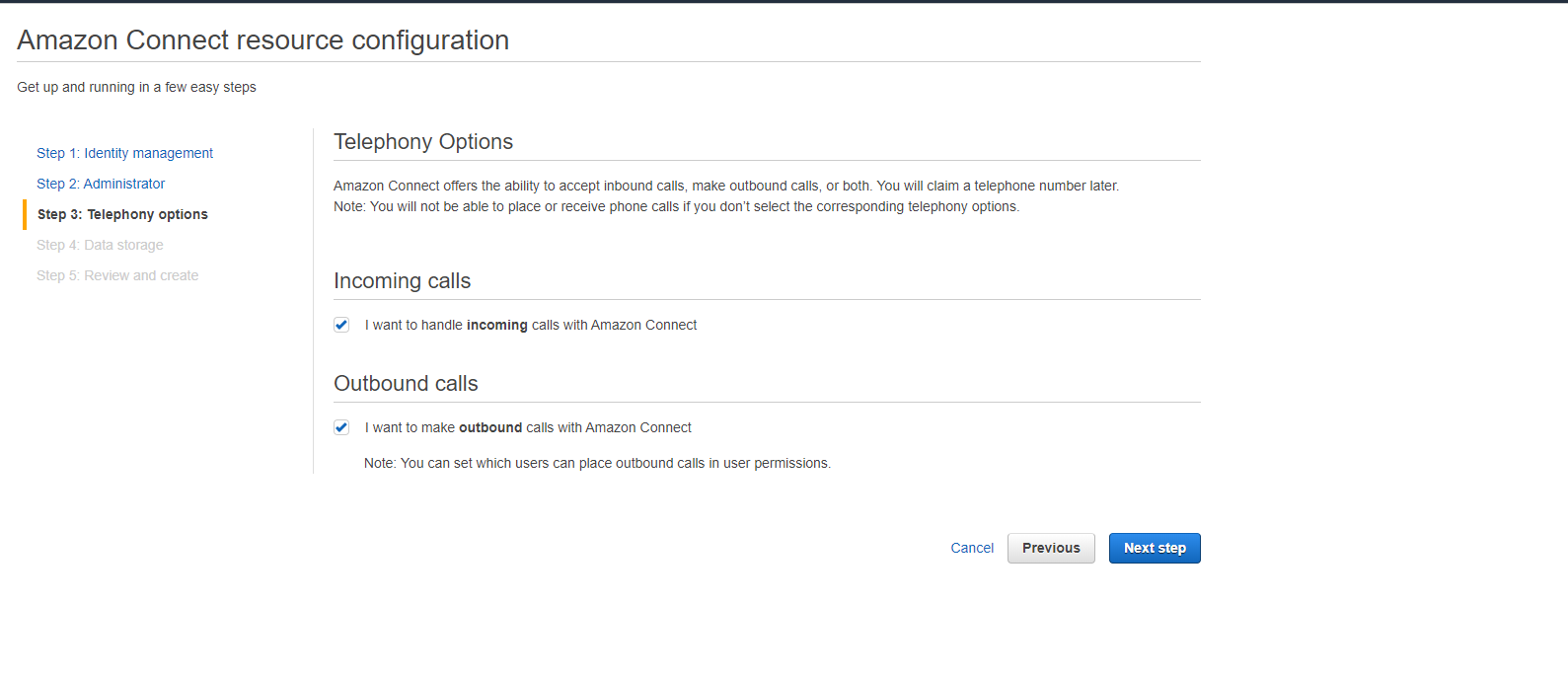
Before you can use a bot in your contact flow you need to add it to your Amazon Connect instance. You can only add bots created under the same AWS account.

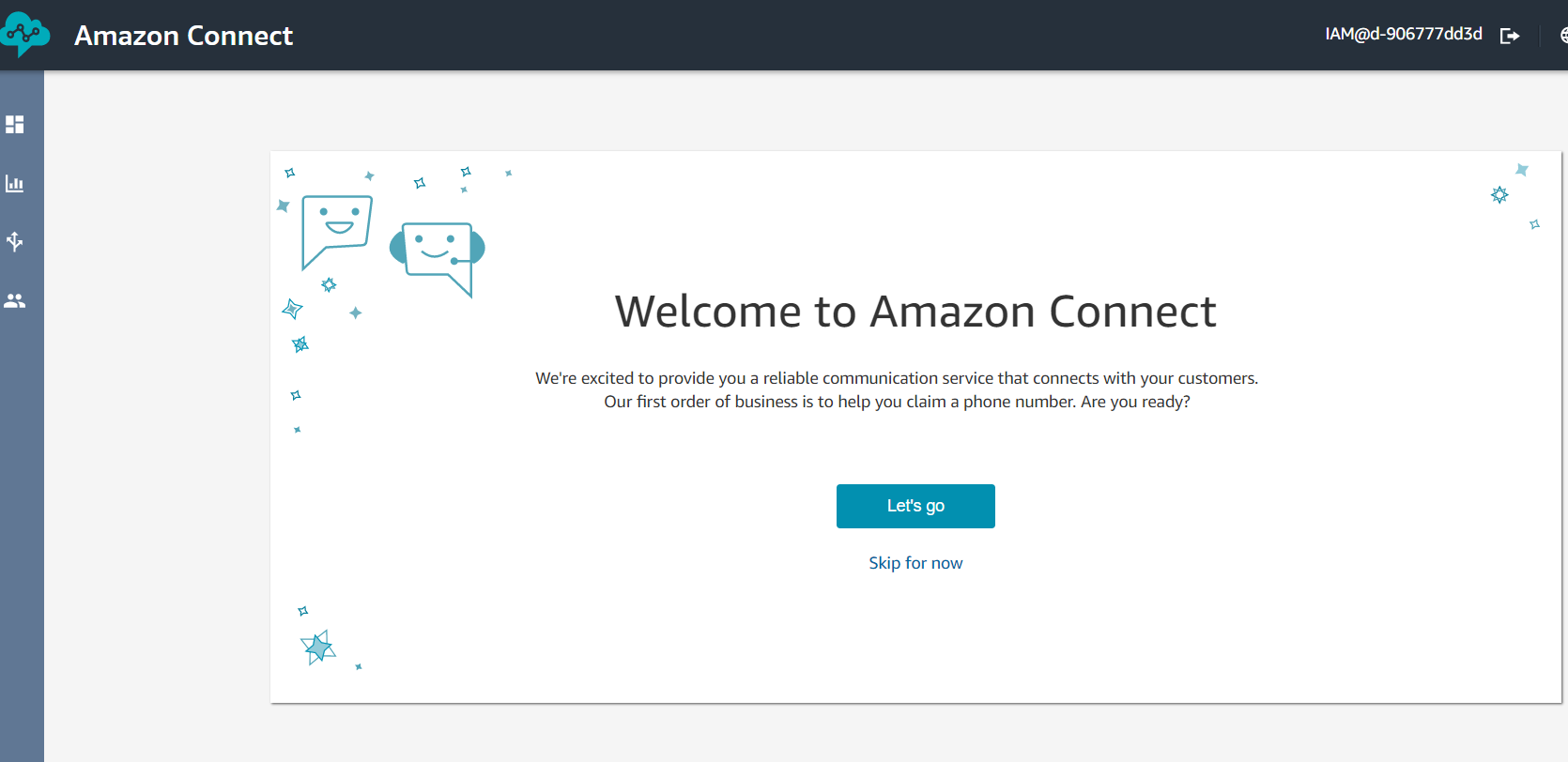
If you add Amazon Lex bots created in a different Region from your instance, performance may be affected.

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.

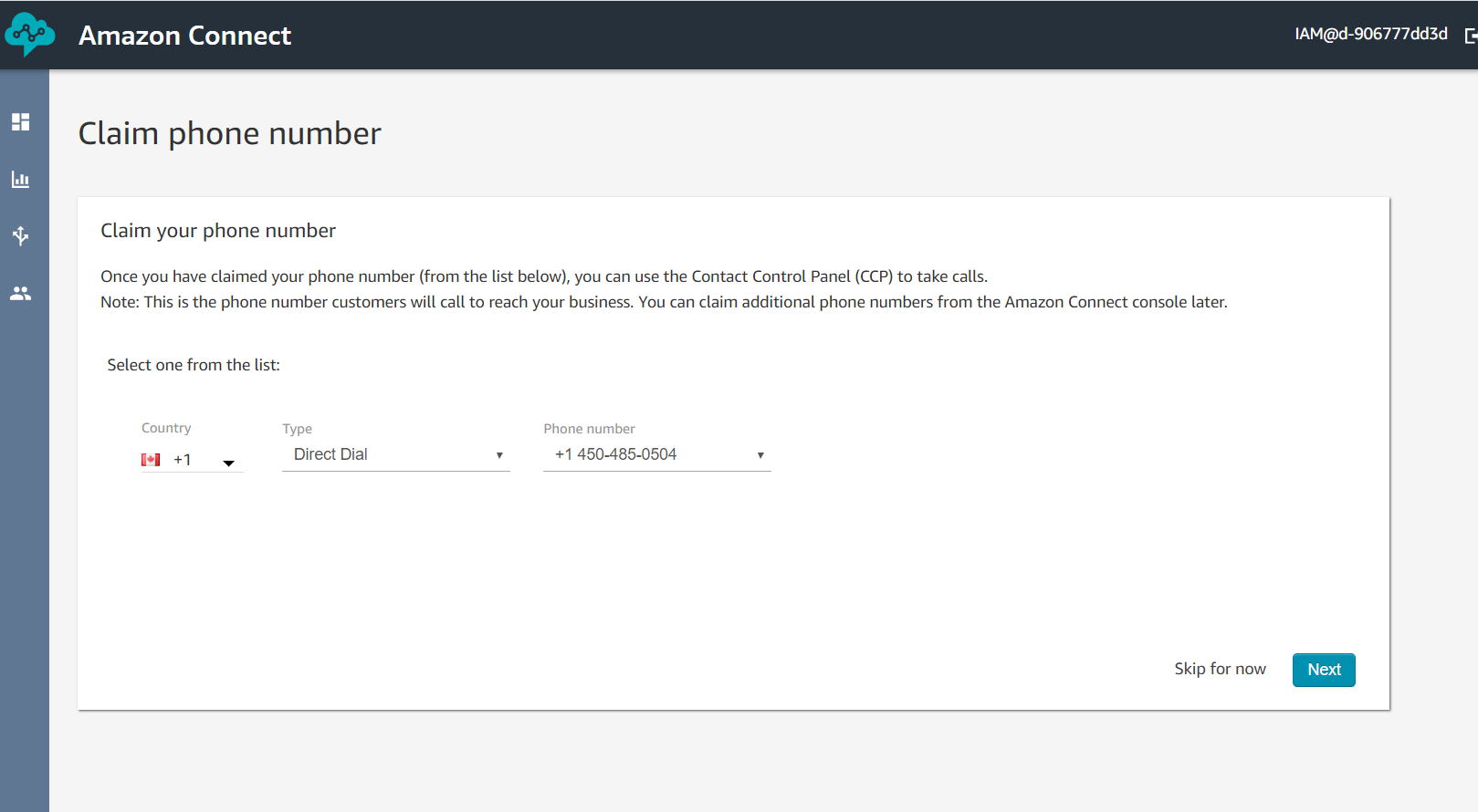




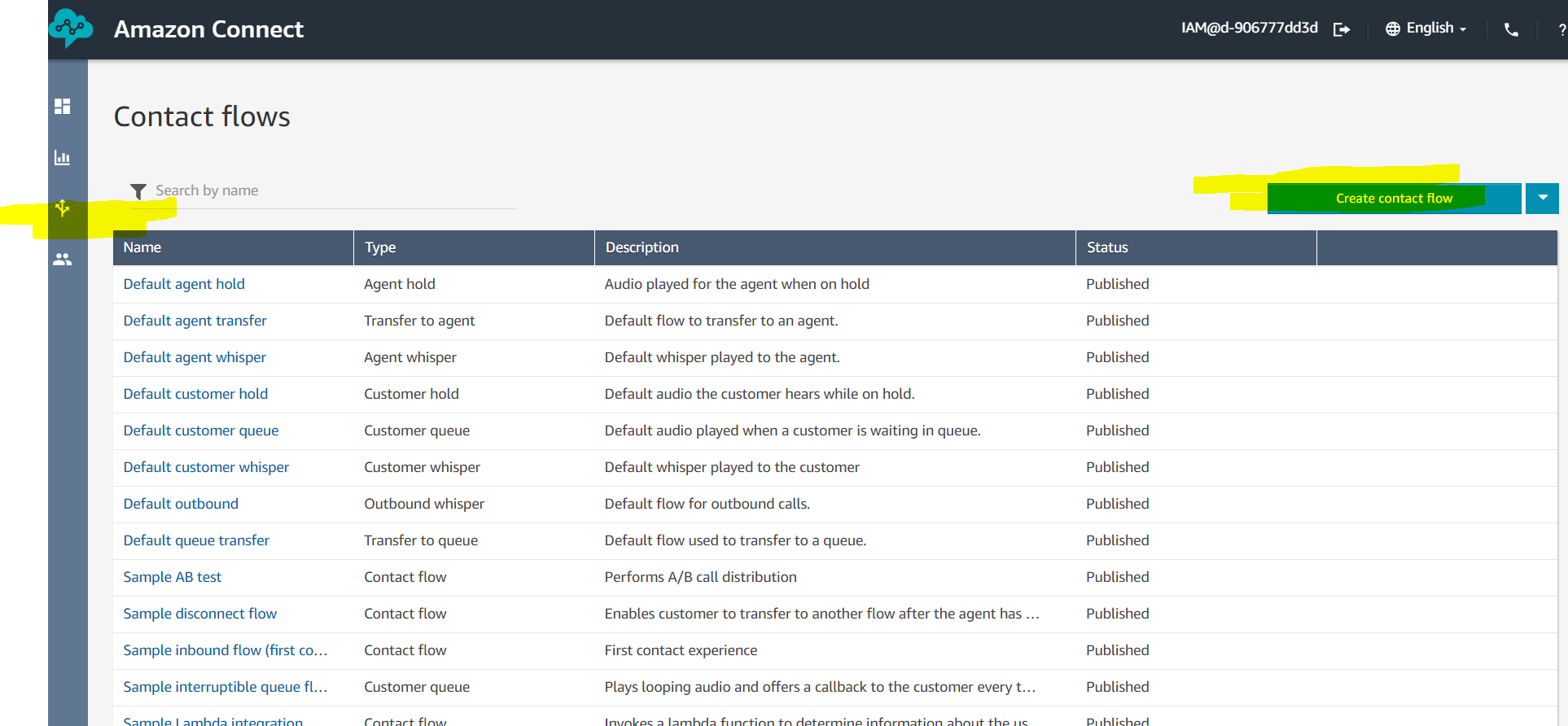




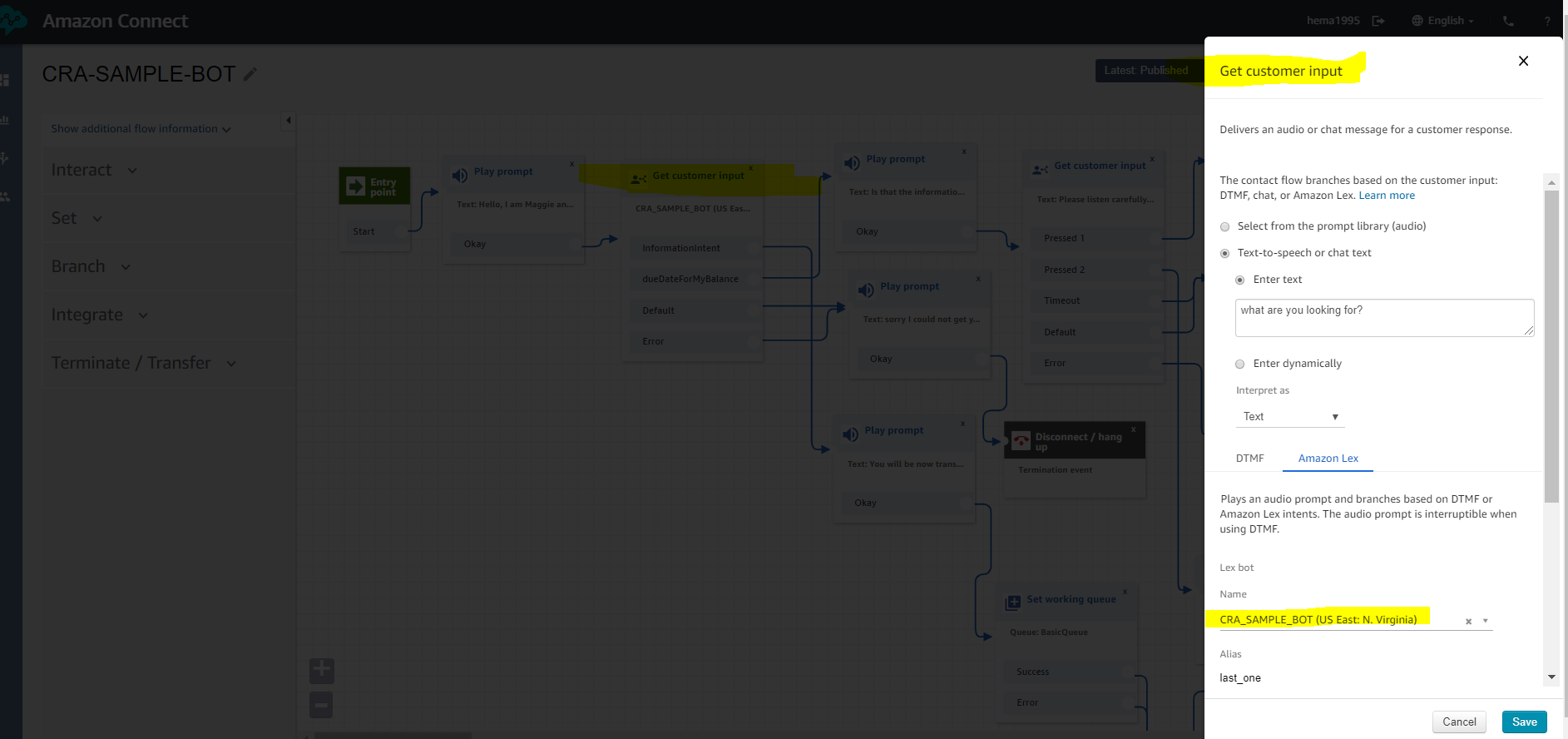
1. Claim the phone numbers for the agencts.



1. Choose **Contact flows.**



1. Under **Amazon Lex**, use the drop-down to choose a name for your bot and then choose **+ Add Lex Bot**.

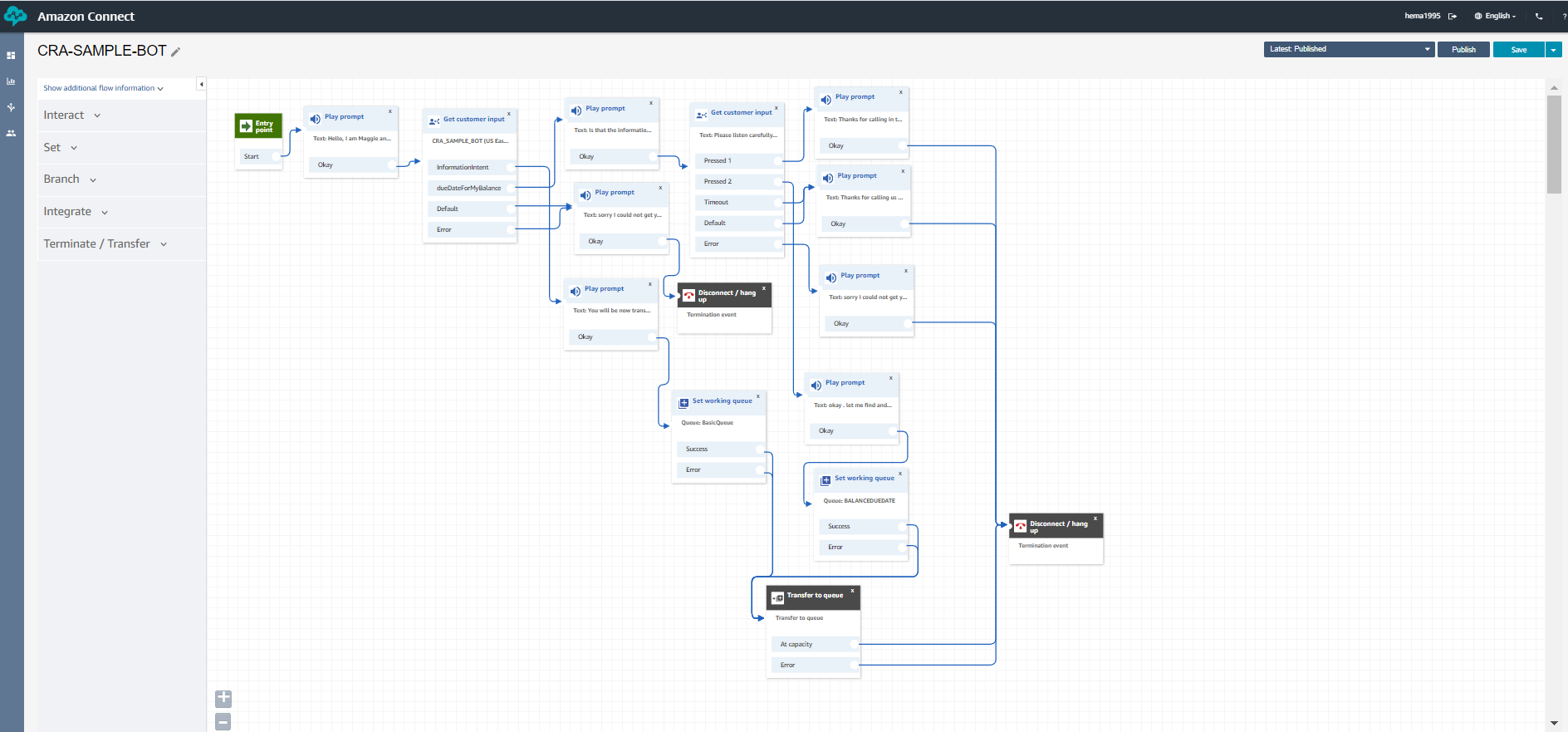


1. Select the **CRA\_SAMPLE\_BOT** bot and choose **Save Lex Bots**. If the name of your bot doesn't appear in the list, reload the page to get it to show up.

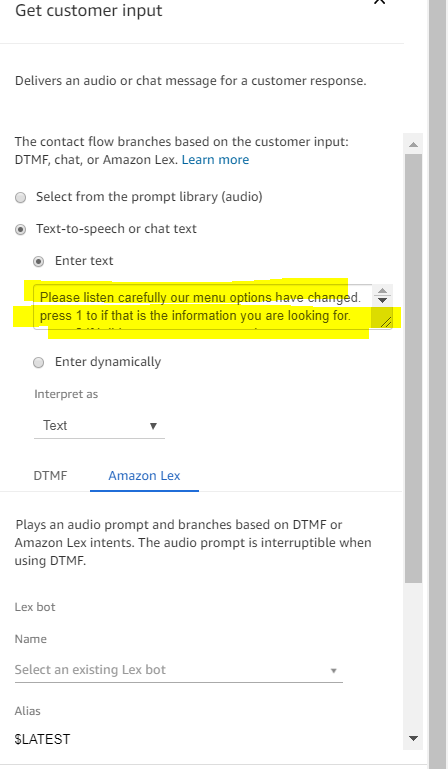
## Create a Contact Flow and Add Your Amazon Lex Bot

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.



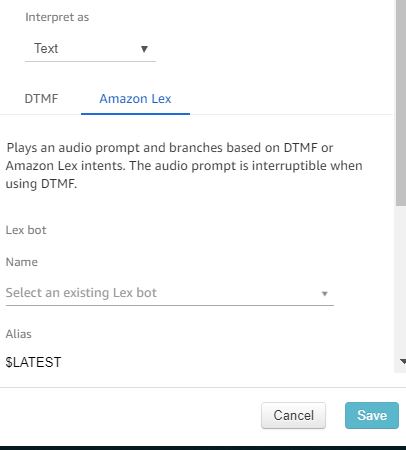
1. Under **Interact**, drag a **Get customer input** block onto the designer, and connect it to the **Entry point block**.
2. Click the **Get customer input** block to open it. Choose **Text to speech (Ad hoc), Enter text**.
3. Type a message that provides callers with information about what they can do. For example, use a message that matches the intents used in the bot, such as “Please listen our menu options has changed press 1 or else 2”



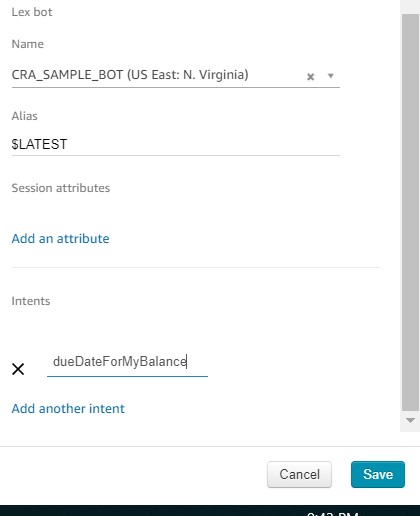
### Add the Amazon Lex Bot to Your Contact Flow

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 **CRA\_SAMPLE\_BOT**. For **Alias**, use **LATEST**.



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



1. choose **Save**.

### Finish the Contact Flow

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

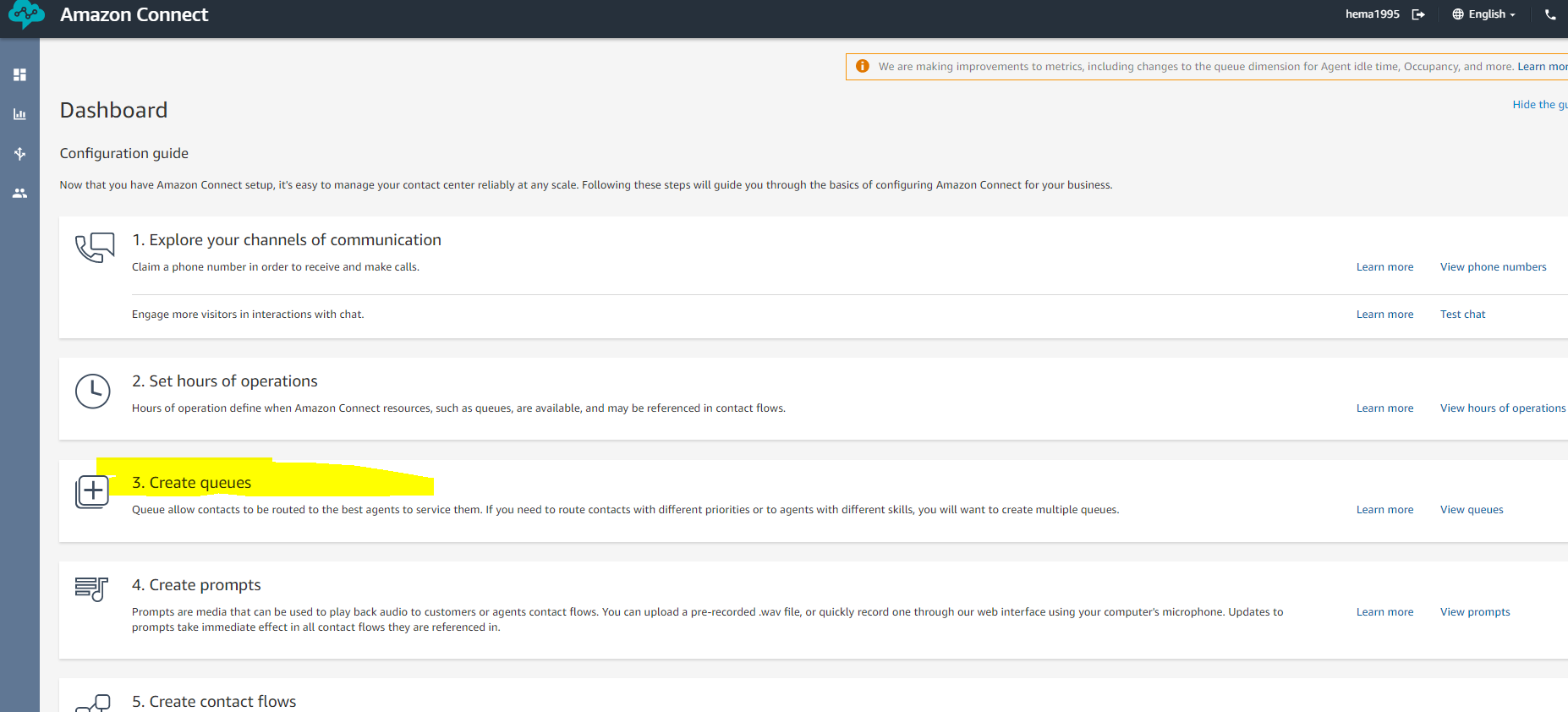
In our example the user calls the customer with the mobile we claimed first and then greets the bots like hi. bot plays prompt with greeting message.

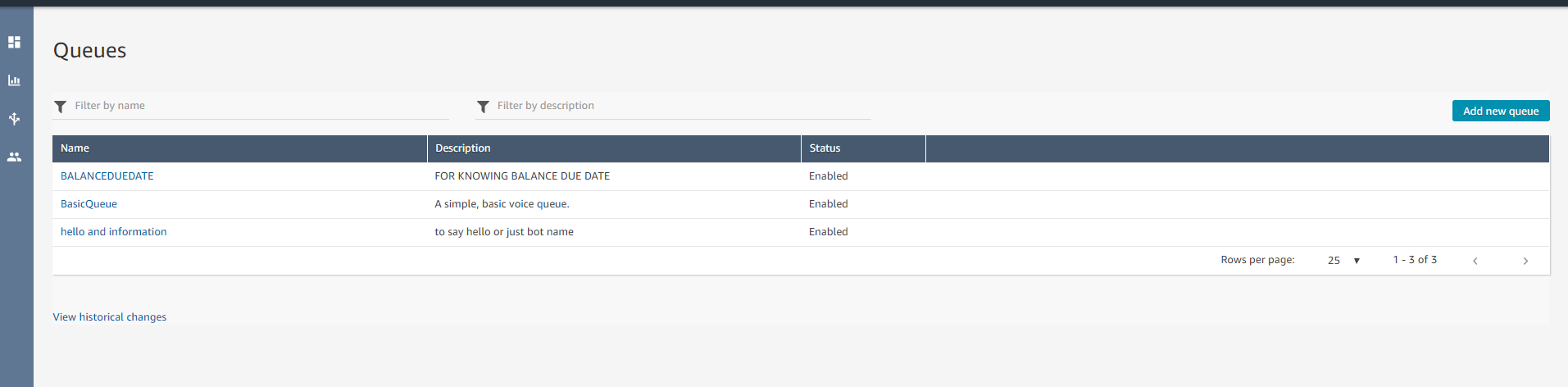
1.then user askes for the what is balance due date for my balance?

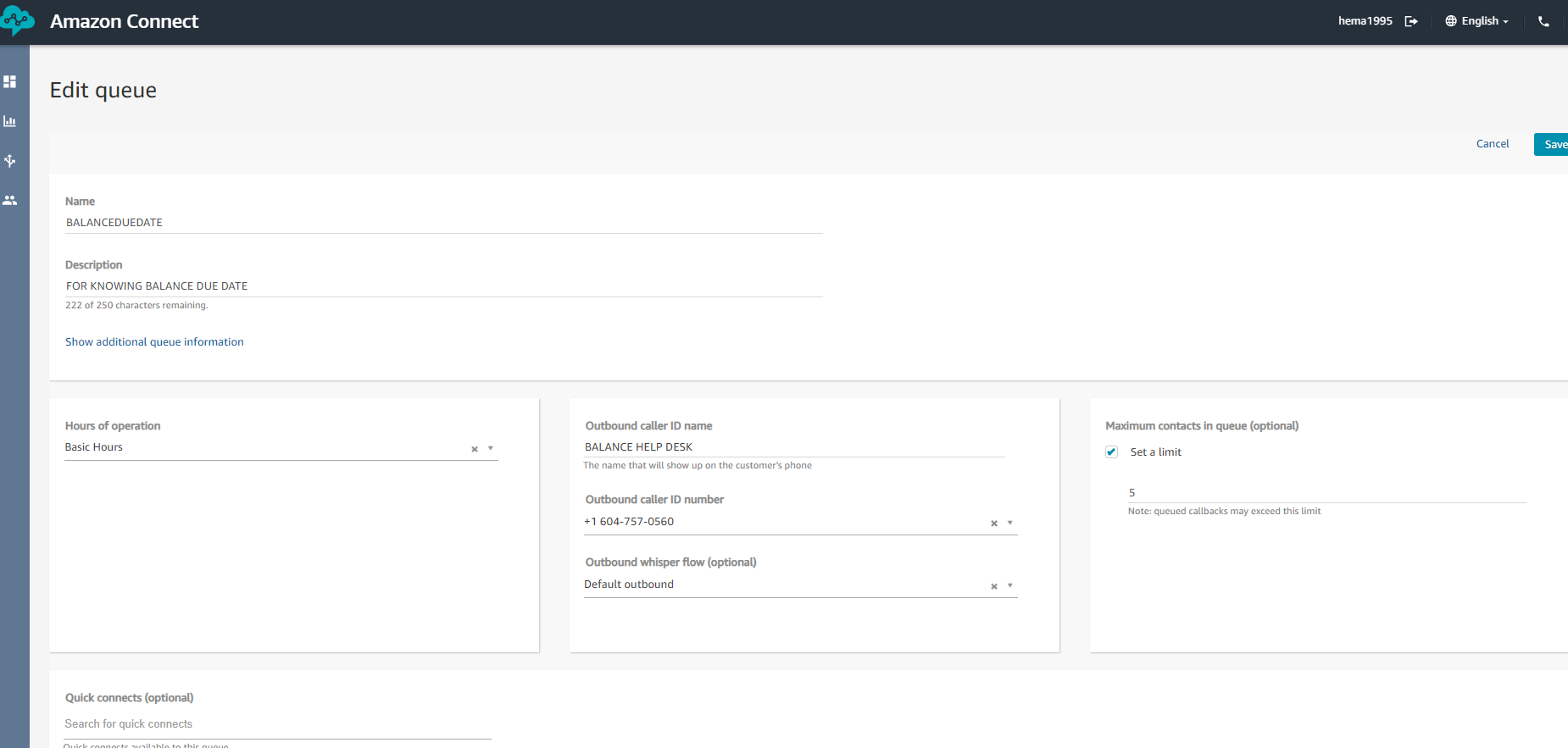
2.now the get customer input is triggered which behind triggers the dueDateForMyBalance intent from the lex and answers the person with corresponding questions in intent .once the user fulfills the intents questions then the lex is calls lambda to define the respective answer to the user and prompts it to him.

3.If the users is not satisfied to bot response he could opt to transfer to agent. We can do this by connecting the prompt message box to the transfer to queue and now customer will be placed in the call with agent by bot.

4. we can define different queues based on different agents we want to answer to specific type of question. In this case we goto amazon connect dashboard🡪 create queues







## 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🡪routing 🡪phone numbers
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**.