**Question #1:**

I’ve implemented an incoming ringtone with Flex, we’ve built a small function that listen to reservation.created and then start an audio with js, i.e:

manager.workerClient.on(‘reservationCreated’, reservation => {

audio.play();

});

However when the agent picks up the call the ringtone still continues, how should I stop the ringtone?

**ANSWER:**

You need to add a check for when the reservation has been accepted then pause the audio.

Example: reservation.on(“accepted”, () => { audio.pause() });

**Question #2:**

We are getting Error - 81015 in the debugger in the Twilio console, we are not sure why, but it started happening after we’ve been playing around with the TaskRouter Workflows, we deleted some of them and that issue start happening.

**ANSWER:**

**Steps to troubleshoot:**

1. Identify the meaning of the error
   1. The error code 81015 means “Failed to Create Task” according to <https://www.twilio.com/docs/api/errors>. Possible causes include required values are missing from the widget, values were not correctly formatted (such as Attributes).
2. Look in the debugger log to see where the error is occurring and any other possible messages that would give a clue as to what is missing.
3. In this case, a Task was probably deleted, based on what was mentioned. Check what is missing and add a new Task to route to and see if this eliminates the error.
4. If the Task exists, check the parameters for any incorrectly formatted values.

**Question #3:**

We have setup SSO with Flex, when a user tries to login we are getting the error 502 from the network, can you please help us troubleshoot the issue?

**ANSWER:**

Error 502 refers to a Bad Gateway error indicating an issue connecting between online servers. Troubleshooting involves the following:

* First, have them close and re-open the browser and try to log in again. Clearing the cache and deleting cookies may also be required.
  + Trying another browser or the same user on another computer is also a good idea.
* If the issue is still occurring:
  + Determine the link they are trying to log into.
  + Determine the link they are using to reach the SSO Identity Provider.
  + Verify the links are accessible outside of their network such as from an offsite location or disconnected from the network/wireless on a smart phone.
    - If the links aren’t reachable then have them check with their local IT to check and possibly open the firewall to the location.
  + If the links are reachable, validation of the SSO setup may be required via the Identity Provider.

**Take home task:**

The customer wants to build a small contact center that sends out SMS messages, the receiver of the message then replies and the inbound message would have to go through a chatbot.

Please share the code in github and an abstract explanation of how you’ve built the above scenario.

**Github code:** <https://github.com/tasajara/twiliotest>

**Abstract explanation:**

There are a couple of possible approaches to this, but the main idea is to first deploy code that will take a set of phone numbers and messages that need to be sent, and then send further responses back to any messages from the receivers.

This was a generic client request, so one example solution might be to send an SMS for a dentist appointment reminder. The patient can reply with CONFIRM or RESCHEDULE. The SMS response back is routed to a chatbot via a Twilio webhook (chatbot can be code on a server, or via a Twilio workflow) that can then take the response and update their account and possibly route the request to a real person to handle a reschedule if required. Ideally, the chatbot might be able to suggest alternative times if they need to cancel (for a future iteration of the contact center)

For this code, a simple send code with example data is provided, and a simple chatbot server that is deployed in a Flask framework to receive and respond to Confirm/Reschedule messages back from the patients. This code is skeleton for client to work from showing how the SMS sending

and processing would work.