**WxCC Prompt Management IVR**

**Author: Carl Newton** [**canewton@cisco.com**](mailto:canewton@cisco.com)

**Date: 20/12/22**

# Introduction

This guide details setting up a “Prompt management” IVR for Webex contact centre. Here is some sample text I sent to a customer to show them how to use it.

Hi Team, I have the greeting management up and working for you. We need to talk about the options as it may not be perfectly optimal at the moment, but its functional.

Dial +12792160010

Press 0 within 2 seconds of call establish to get to the "greeting management" script. You can also press 0 during the "Welcome to UC davis" main menu for the same effect. The PIN is 147258.

Options once verified with PIN

1. Play the current message

2. Change the greeting. Open, Custom1 and Custom2 greeetings play a message then pass to the main flow. Closed and emergency greetings play a message then disconnect the caller.

3. Change the greeting to "Text" mode. This will play the text to speech message. I am working on an interface for this so you can modify the text from a website so the message can be changed on the fly.

4. Change the greeting to "Audio" Mode. These are static greetings. Check the current audio files in WXCC And replace the relevant files to change the greetings.

Your API Endpoint is <https://REMOVED.mockapi.io/WXCC/CCPrompts> - Open the link to see the data!

# Requirements and initial prep

This guide requires a text to speech connector from google. The configuration guide is here, and there is a very generous free trial you can use for demos. https://www.cisco.com/c/en/us/support/docs/contact-center/webex-contact-center/217425-configure-google-text-to-speech-for-webe.html

You should also download the required files from <https://github.com/Delarado/WxCCPrompt>

# How to set up

Register for a free account at <https://mockapi.io/>

Create a new project and \*\*Critical step\*\* give it an API Prefix of /WXCC

Create a new resource and \*\*Critical step\*\* call it “CCPrompts” and save it as all the default values.

Graphical user interface, text, application, chat or text message

Description automatically generated

Click “Data” next to the resource you created and enter the following data:

Graphical user interface, text, application, chat or text message

Description automatically generated

[

{

"CCStatus": "Emergency",

"CCMode": "Text",

"Greetings": {

"Open": "The contact centre is currently Open.",

"Closed": "The contact centre is currently closed",

"Emergency": "It is currently an emergency.",

"Custom1": "This is a custom Greeting number 1",

"Custom2": "This is a Custom Greeting number 2"

},

"id": "1"

}

]

Graphical user interface, text, application

Description automatically generated

OPTIONAL: If you want to play around with the information contained in the DB with an external application, install the postman collection.

When you load the collection, you must update the request URL in each request to match your [MockAPI.io](http://MockAPI.io) endpoint. The following picture:

Graphical user interface, text, application

Description automatically generated

Should read as follows, where :endpoint is the name of your endpoint (CCPrompts)

Graphical user interface, text, application, email

Description automatically generated

Next, sign into control hub. Go to contact centre, Bulk Operations and create a new bulk operation. Change the configuration object to “Audio files” And upload the file “Greetings.zip”. These are the audio prompt files for each contact centre status.

Import the flow “GreetingMGMT\_V2.json” Into WxCC.

Open the flow and make the following changes:

Modify the APIURL variable to be something like “626aa1b27f8c1826c3b4.mockapi.io” or whatever your API URL Is. DO NOT INCLUDE ANYTHING ELSE IN THE VARIABLE SUCH AS SLASHES OR http:// etc.

Verify the PIN in the “CheckPIN” Condition step is the one your customer wants to use.

Reconfigure the Text To Speech & Audio prompt variable Messages (Marked with names beginning with TTS\_ ) with the desired connector, voice, and the text from the “Flow description” for all of the Play Message, EnterPIN and Menu steps. Note the “PlayAudioGreeting” Step has a multi part message so you need to split the greetings out in the activity description. You may also need to configure audio prompt variables instead of just text to speech messages.

Graphical user interface, text, application, chat or text message

Description automatically generated

Create a new entry point and mapping for your desired greeting management DDI. So this management flow can be called. Alternatively, configure some kind of “Secret menu” in an existing call flow and link it to the greetingMGMT call flow using a “Go to flow” step.

Test the greeting management call flow. You should be able to switch the greetings from Audio to text mode, and change the greeting between a few different statuses.

Once confirmed working, next import the Flow\_TOD\_Ex.json flow. As with the previous flow, Modify the APIURL variable to your [Mockapi.io](http://Mockapi.io) URL. Reconfigure the text to speech messages with the desired connector voice, and the text from the flow description on all nodes that’s names start with “TTS\_”

Now, either create “Go to flow” steps as desired to link to your main flow, or include this logic in your main flow. When sending to a main flow, you can adjust the flow variable mapping in the go-to step so that you can use the contact centre status variables in the main flow too. Using Go to pointed at an entry point instead of a flow automatically maps common variables. The greetings are designed to be used as follows if you exactly copy the Flow\_TOD Flow: Open, Custom 1 and Custom 2: Plays a message and then goes to the main flow Closed & Emergency: Plays a message and disconnects the caller