

Google DialogFlow ES Integration with Webex Contact Center

Name

Dialogflow ES Virtual Agent

Labels

Intermediate, Voice, Inbound, Virtual Agent

Description

This template demonstrates the data flow between Google DialogFlow ES and Webex Contact Center, focusing on how to pass data to and from both platforms during an interaction. It provides a foundational flow where data is exchanged with DialogFlow ES for natural language processing and automated agent fulfillment.

Details

This flow showcases how data is passed between Webex Contact Center and DialogFlow ES for processing customer interactions. The integration with DialogFlow enables the bot to understand customer intents and take appropriate actions based on the conversation.

Additionally, the flow includes error handling to ensure a smooth customer experience, even when unexpected conditions arise.

Pre-requisites

To use this flow, ensure the following are set up:

- A Google DialogFlow ES agent with relevant intents for the conversation.
- Entry Point, Queue, Teams, and Entry Point Mapping configured in the Webex Contact Center Management Portal.
- Enable Webhook Fulfillment in DialogFlow ES and use the sample node.js code in the inline editor.
- Cisco Text-to-Speech (TTS) is enabled for generating custom messages dynamically.
- Upload static audio files if you are not using Cisco's default audio.

Integration Breakdown

1. **Customer initiates contact:** The call is received by Webex Contact Center.
2. **Data is passed to DialogFlow ES:** A custom greeting, which includes customer details such as name and reason for calling, is sent to the DialogFlow ES bot for processing.
3. **Bot interaction with DialogFlow:** DialogFlow processes the input and responds based on configured intents.
4. **Music in queue:** While the bot processes the request, the customer is placed in a queue with hold music.
5. **Disconnect:** The interaction ends once the dialog is completed.

Activities Used in the Flow

Start

- This activity marks the beginning of the flow. It is triggered when a new call is received.

Set Language

- The flow uses a Set Variable activity to configure the language code (en-US) for the entire interaction. This ensures all voice interactions align with the caller's language preference.

Custom Greeting

- This activity passes customer details such as name, email, and reason for calling to the DialogFlow ES bot. The greeting is dynamically generated using Cisco Text-to-Speech (TTS). Example data passed:
 - customerName: Jane Doe
 - customerEmail: customer@email.com
 - customerReason: Bookings

Queue to Agent

- If the interaction requires escalation, the customer is placed in a queue, and hold music is played using the defaultmusic_on_hold.wav file.

Play Music

- Music is played while the caller waits in the queue. The flow uses Cisco's default hold music but can be customized by uploading different music files.

Disconnect

- This activity disconnects the call once the flow is complete, ensuring a seamless end to the interaction.

Flow Specifics

The flow JSON used in this example contains variables and activities essential for interaction handling, error processing, and communication between Webex Contact Center and DialogFlow. The key variables used include:

- Global_FeedbackSurveyOptIn: Tracks whether the customer opts in for a post-call survey.
- customerName: Captures the customer's name for personalization.
- customerEmail: Captures the customer's email.
- customerReason: Records the reason for the customer's call.
- Global_Language: Configures the default language (en-US).
- Global_VoiceName: Determines the voice used for text-to-speech.

Additional Resources

- For a deeper dive into the integration, check out the [Working with Data on Google DialogFlow ES with Webex Contact Center](#) video.
- Refer to [Webex Contact Center Developer Documentation](#) and [DialogFlow ES Documentation](#) for further guidance.

Developer Support

For any support regarding this integration, open a ticket with the Webex Contact Center Developer Support team via the [Webex Developer Portal](#).

For further discussions, visit the [Webex Contact Center APIs Developer Community](#).