

Configure CVP Customer Virtual Assistant (CVA)

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background](#)

[Architecture](#)

[Cisco CVA Call Flows](#)

[Google Based IVR Logic \(Dialogflow\)](#)

[Premised Based Intent \(DialogflowIntent / DialogflowParam\)](#)

[Transcribe](#)

[Configure](#)

[Dialogflow Project / Virtual Agent](#)

[What is a Dialogflow?](#)

[CVVB Speech Server Configuration](#)

[CVP Call Studio Elements](#)

[Dialogflow](#)

[DialogflowIntent](#)

[DialogflowParam](#)

[Transcribe](#)

[CVP Call Studio Applications](#)

[Cloud Based Intent Processing - Google Based IVR Logic \(Dialogflow\)](#)

[Premise Based Intent Processing \(DialogflowIntent / DialogflowParam\)](#)

[Proxy Server Configuraion](#)

[Troubleshoot](#)

[Related Information](#)

[Cisco Documentation](#)

[Google Documentation](#)

Introduction

This document describes how to configure Customer Voice Portal (CVP) CVA feature.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco Unified Contact Center Enterprise (UCCE) Release 12.5
- Cisco Package Contact Center Enterprise (PCCE) Release 12.5
- CVP Release 12.5

- Cisco Virtualized Voice Browser (CVVB) 12.5
- Cisco Unified Border Element (CUBE) or Voice Gateway (GW)
- Google Dialogflow

Components Used

The information in this document is based on these software versions:

- Cisco Package Contact Center Enterprise (PCCE) Release 12.5
- CVP Release 12.5
- Cisco Virtualized Voice Browser (CVVB) 12.5
- Google Dialogflow

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Background

CVP 12.5 introduces the Customer Virtual Assistant (CVA) feature, in which you can use third-party vendors Text to Speech (TTS), Automatic Speech recognition (ASR) and Natural Language processing (NLP) services.

 **Note:** In this release only Google NLP is supported.

This feature supports human-like interactions that enable you to resolve issues quickly and more efficiently within the Interactive Voice Response (IVR) with Natural Language Processing.

Cisco CVA provides these modes of interaction:

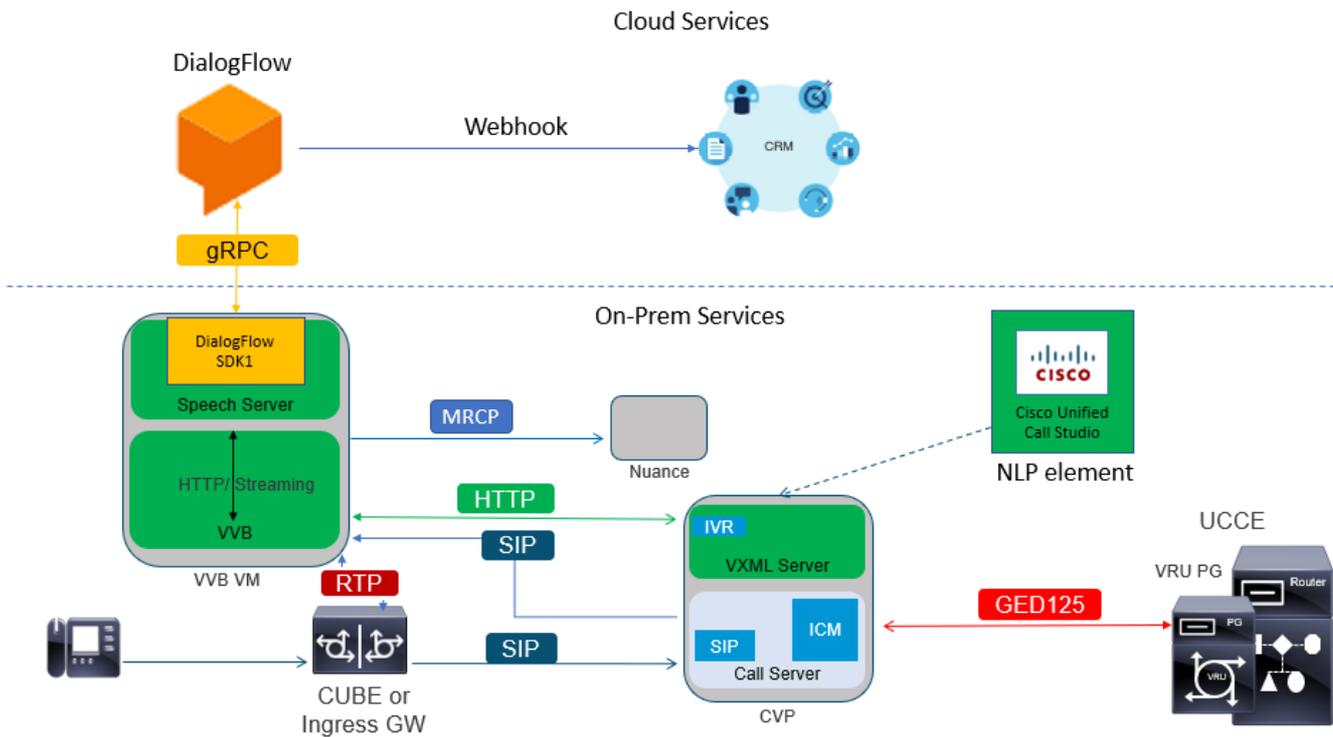
- **Local Interaction:** Prompts are played locally with the use of WAV files and user inputs are captured with the use of DTMF grammar.
- **MRCP-Based Interactions:** Prompts are played by the external on-prem based media server over Media Resource Control Protocol (MRCP) Synthesis command for TTS functionality. The prompts are recognized by the external media server based on pre-defined grammar by ASR.
- **Natural Language Understanding (NLU):** This feature enables a dialog to be initiated by the interaction with cloud-based Natural Language Processing (NLP) engines that are trained to understand natural language.

Architecture

In addition to the components required in a CVP Comprehensive call flow, CVA requires Cloud Services, Speech Services and specific CVP Call Studio elements to be implemented. This is the list of all components required in CVA:

- Ingress, Egress, CUBE Gateways
- Unified Customer Voice Portal (Unified CVP) Solution, including Call Studio
- Unified Contact Center Enterprise (Unified CCE)
- Cisco Virtualized Voice Browser (VVB) - Speech services

- Cloud Services (Google Dialogflow)



Cisco CVA Call Flows

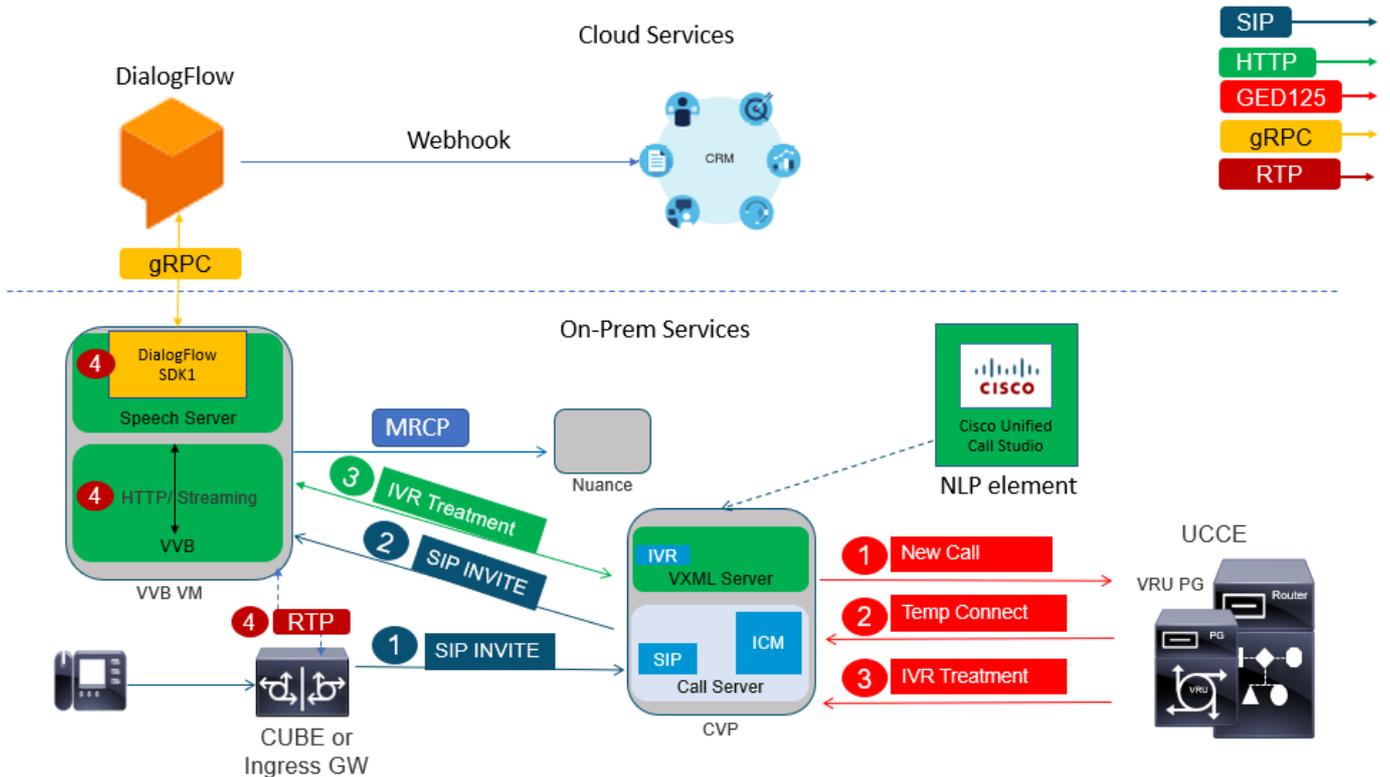
There are three main CVA call flows supported with Google Dialogflow.

- **Google Based IVR Logic (Dialogflow)**
- **Premise Based Intent (DialogflowIntent / DialogflowParam)**
- **Transcribe**

Google Based IVR Logic (Dialogflow)

Hosted IVR deployment is most suited to customers who plan to migrate their IVR infrastructure into the Cloud. Under Hosted IVR deployment, only IVR business logic resides in the Cloud whereas agents are registered to on-premises infrastructure.

Once hosted IVR is deployed, the core signal, and the media process happens in the Cloud; additionally, CVP and Cisco VVB solutions are in bridged mode for the media to be streamed to the Cloud. Once IVR is completed and an agent is required, call control is transferred back to CVP for the further process of the call and for queue treatment.



Here is an example of the call flow:

1. The call goes from CUBE or Ingress GW to the CVP Call Server. The call is then sent from CVP to Unified CCE/Package CCE.
2. CCE sends a temporary connect to CVP along with the instruction to set the VRU / IVR treatment with the Cisco VVB.
3. CCE instructs CVP to run a call studio application, which is deployed on the VXML server. CVP sends the call to Cisco VVB and the IVR treatment starts. The audio (RTP) is now established between Cisco VVB and CUBE or Ingress Gateway. Up to this point, the call flow steps are the same as any regular Comprehensive call flow. The next steps are unique to the CVA Dialogflow call flow.
4. The customer voice is streamed to Google Dialogflow by the use of Speech Server on the Cisco VVB.
 - a. Once the stream is received at Dialogflow, recognition takes place, and the NLU service is engaged to identify the intent.
 - b. The NLU service identifies the intents. The intent identification happens based on the virtual agent created in the Cloud.
 - c. Dialogflow returns the subsequent prompts to Cisco VVB in one of these ways (depend on the call studio application configuration):
 - Audio:** Dialogflow returns the audio payload in API response.
 - Text:** Dialogflow returns the text prompt in response, which must be synthesized by a TTS service.
 - d. Cisco VVB plays the prompt to the caller for additional information.
 - e. When the caller responds, Cisco VVB streams this response to Dialogflow.
 - f. Dialogflow performs the fulfilment and responds with the prompts again in one of these two ways:

Audio: Dialogflow returns the audio payload in API response with the fulfilment audio with the use of webhook.

Text: Dialogflow returns the text prompt with the fulfilment text in response with the use of webhook. This is synthesized by a TTS service.

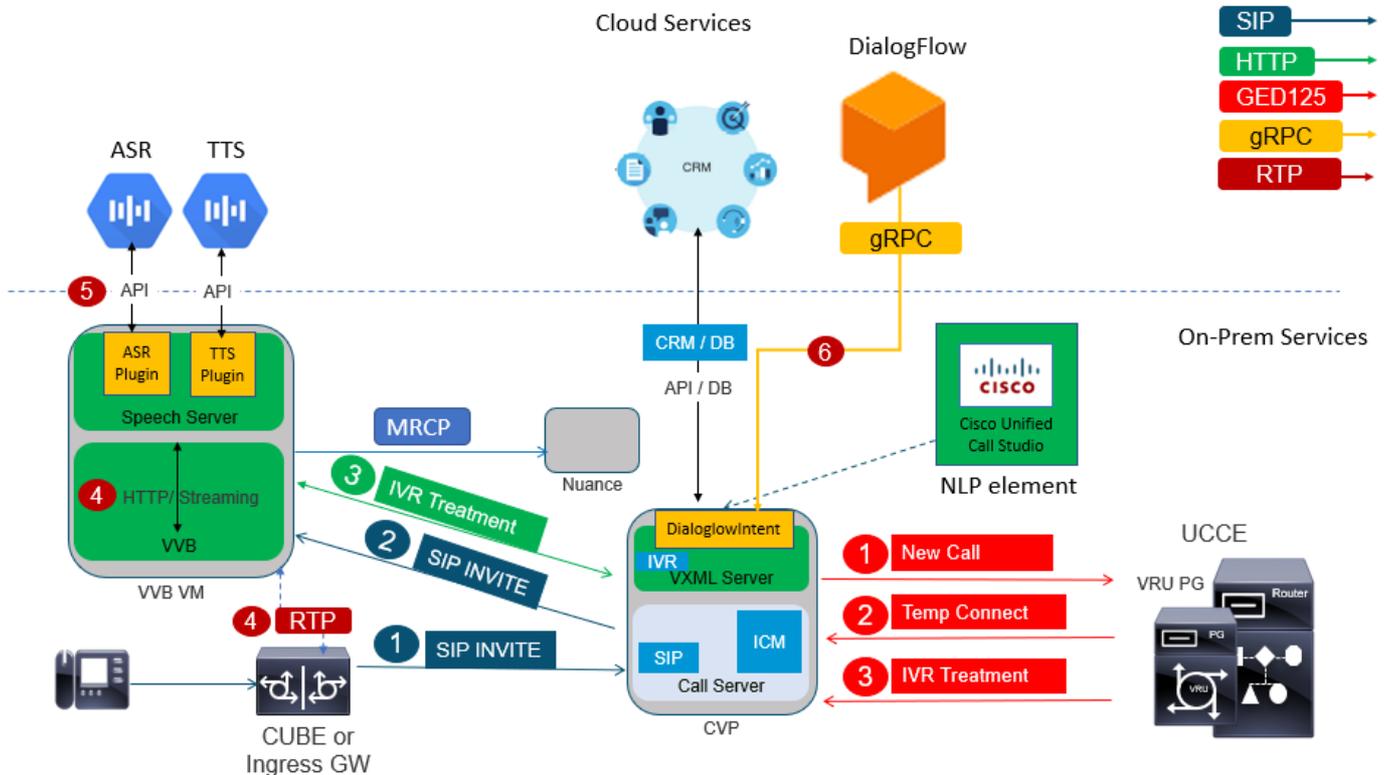
g. Dialogflow performs context management and session management for the entire conversation.

The flow control remains with Dialogflow unless customer requests an agent transfer or the call is disconnected.

Premised Based Intent (DialogflowIntent / DialogflowParam)

Premise based intent deployment are more suitable for customers who require the Personal Identity Information (PII) or any other sensitive data to be handled on their on-premises systems. Typically, in such deployments, Personal Identity Information (PII) is never sent to the Cloud to be processed; instead, it is collected in such a way that the information is always retained and processed on-premises. In this call flow most of the process control happens on the VXML server. This call flow allows:

- Local Parameter Prompt / Sequence
- Local DTMF Detection
- Extend current application
- Local fulfillment



This call flow uses the DialogflowIntent and DialogflowParamt elements of call studio. Steps 1 to 3 are the same as the previous Dialogflow callflow. Here are the subsequent steps

4. The customer voice is streamed to Google Dialogflow via the Speech Server on the Cisco VVB.
5. In this scenario the speech server passes the speech to the Cloud ASR
6. Once the stream is received at Google, recognition takes place and text is returned to VXML Server. VXML Server passes this text to Dialogflow and NLU service is engaged to identify the intent. NLU

identifies the intents already configured. The intent identification happens based on the virtual agent created in the Cloud.

a. Google Dialogflow returns the intent to the call studio application deployed in the VXML Server.

b. If the intent identified requires sensitive information to be processed, such as a credit card number or a PIN to be

entered, Cisco VVB can play the required prompt and collect Dual Tone Multy Frequency (DTMF)s from the end customer.

c. This sensitive information is collected by local business applications and sent to the Customer Relationship Management (CRM) database for authentication and to further

process.

d. Once the customer has been authenticated with their PIN, speech control can be passed back to the ASR service in the Cloud.

e. VXML Server via the call studio application performs context management and session management for the entire conversation.

Essentially, this call flow provides much more flexibility in terms of the definition of actions to be taken at each stage based on customer input and is driven entirely from on-prem applications. Cloud services are engaged primarily for recognition of speech and intent identification. Once the intent is identified, control is passed back to the CVP business application to process and decide what the next step should be.

Transcribe

This call flow provides customer input conversion from speech to a text sentence, basically ASR.

Configure

Dialogflow Project / Virtual Agent

Google Dialogflow needs to be configured and connected to Cisco Speech Server before you start CVA configuration. You require a Google service account, a Google project and a Dialogflow virtual agent. Then, you can teach this Dialogflow virtual agent the natural language so the agent can respond to the customer interaction with the use of Natural Language processing.

What is a Dialogflow?

Google Dialogflow, is a conversational User Experince (UX) platform which enables brand-unique, natural language interactions for devices, applications, and services. In other words, Dialogflow is a framework which provides NLP / NLU (Natural Language Understanding) services.Cisco integrates with Google Dialogflow for CVA.

What does this mean for you? Well, it means you can basically create a virtual agent on Dialogflow and then integrate it with Cisco Contact Center Enterprise.

Here are the steps to create a virtual agent or Dialogflow project:

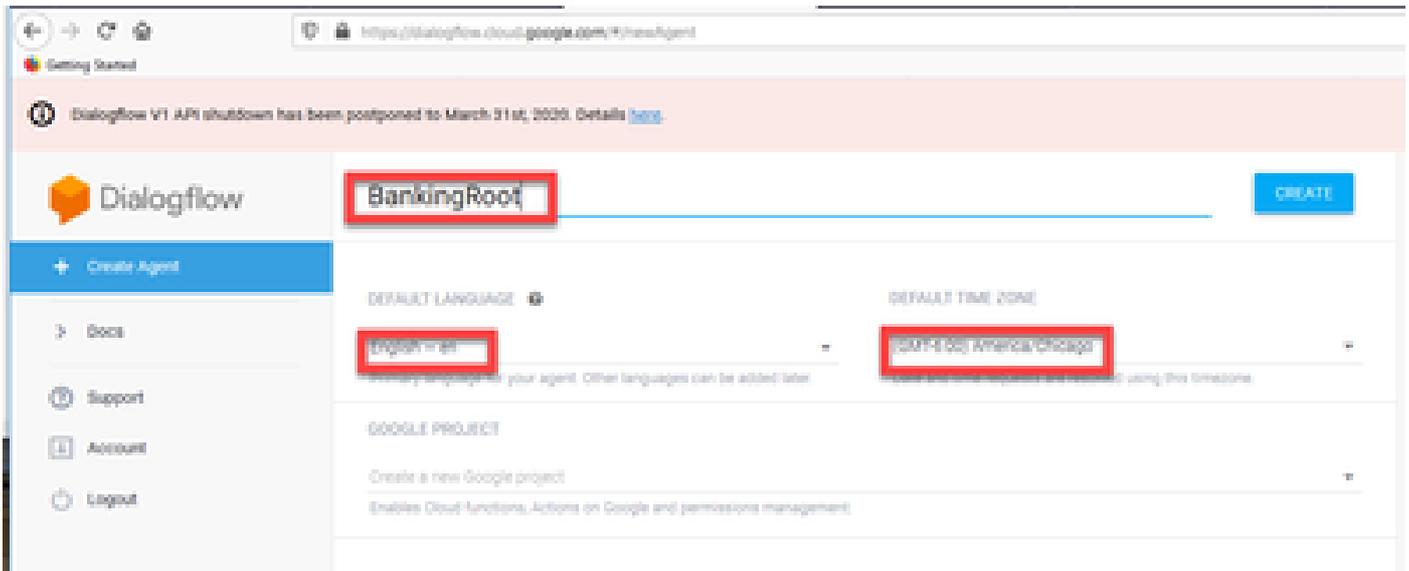
Step 1. Create a Google Account/ Project or Have a Google Project assigned to you from your Cisco

Partner.

Step 2. Log in to Dialogflow. Navigate to <https://dialogflow.com/>

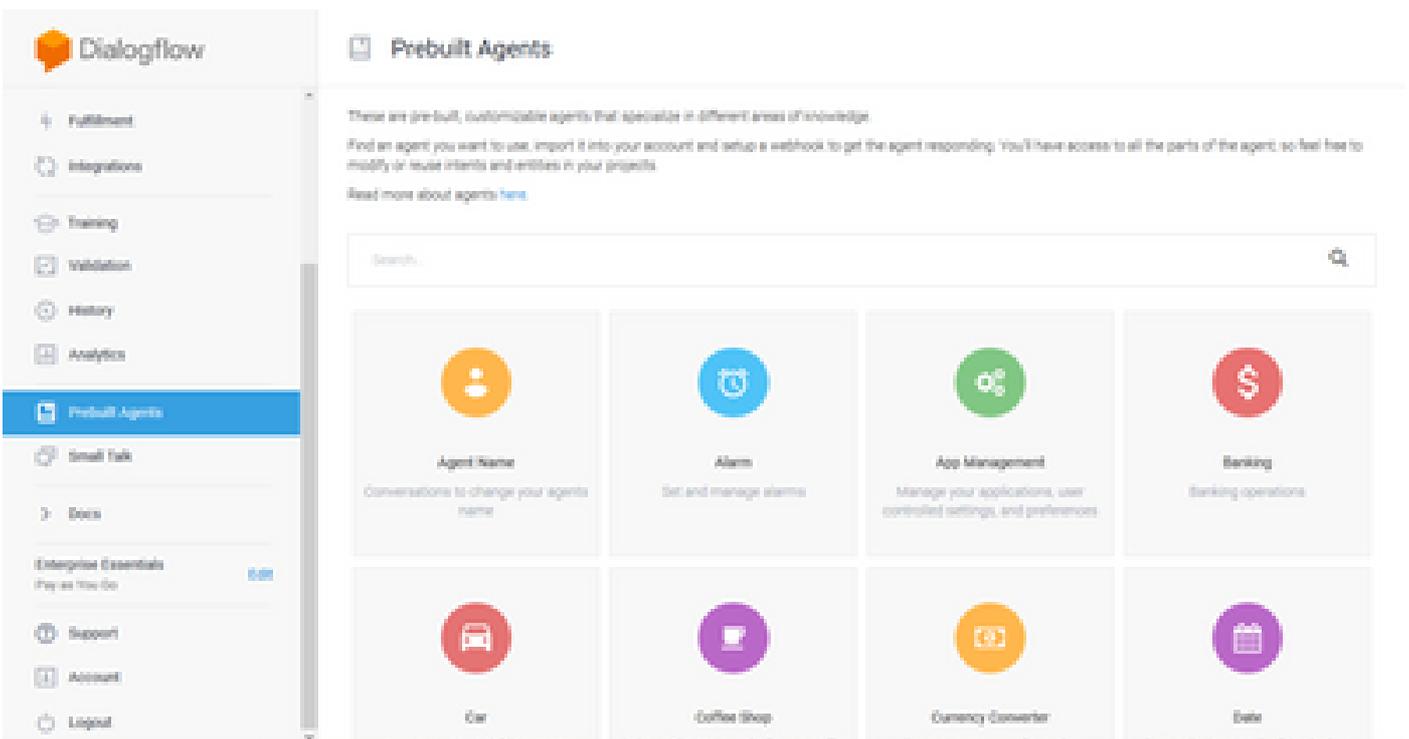
Step 3. Create a new agent. Choose a name for your new agent and the default time zone. Keep the language set to English. Click on **CREATE AGENT**.

In this example the virtual agent handles bank transactions, so the name of the agent for this lab is BankingRoot. The language is English and the Timezone is the default system time.



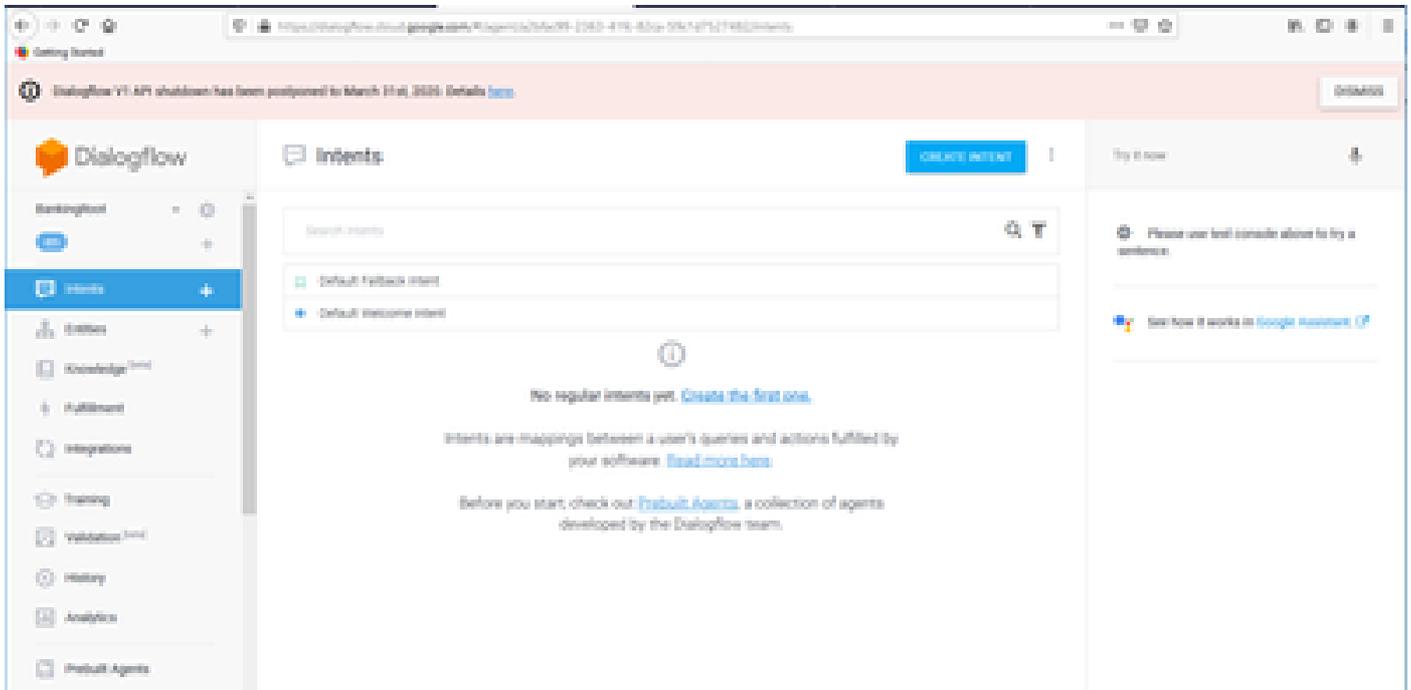
Step 4. Click on the **CREATE** tab.

Step 5. After the virtual agent is created, you can import pre-build Google virtual agents as shown in the image or you can teach the agent how to communicate with the caller.

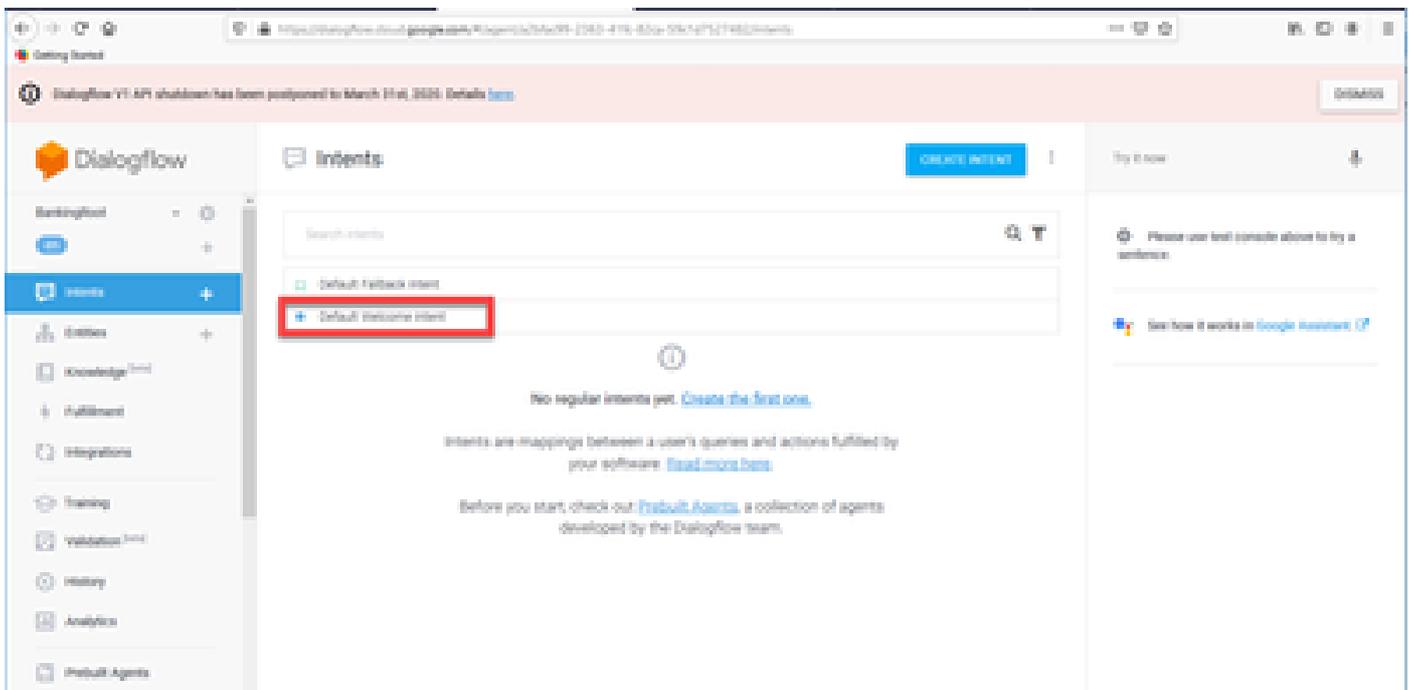


Step 6. At this point, the agent still doesn't know how to respond to any user input. The next step is to teach it how to behave. First, you model agent's personality and make it respond to a **hello** default welcome intent and present itself. After the agent is created you see this image.

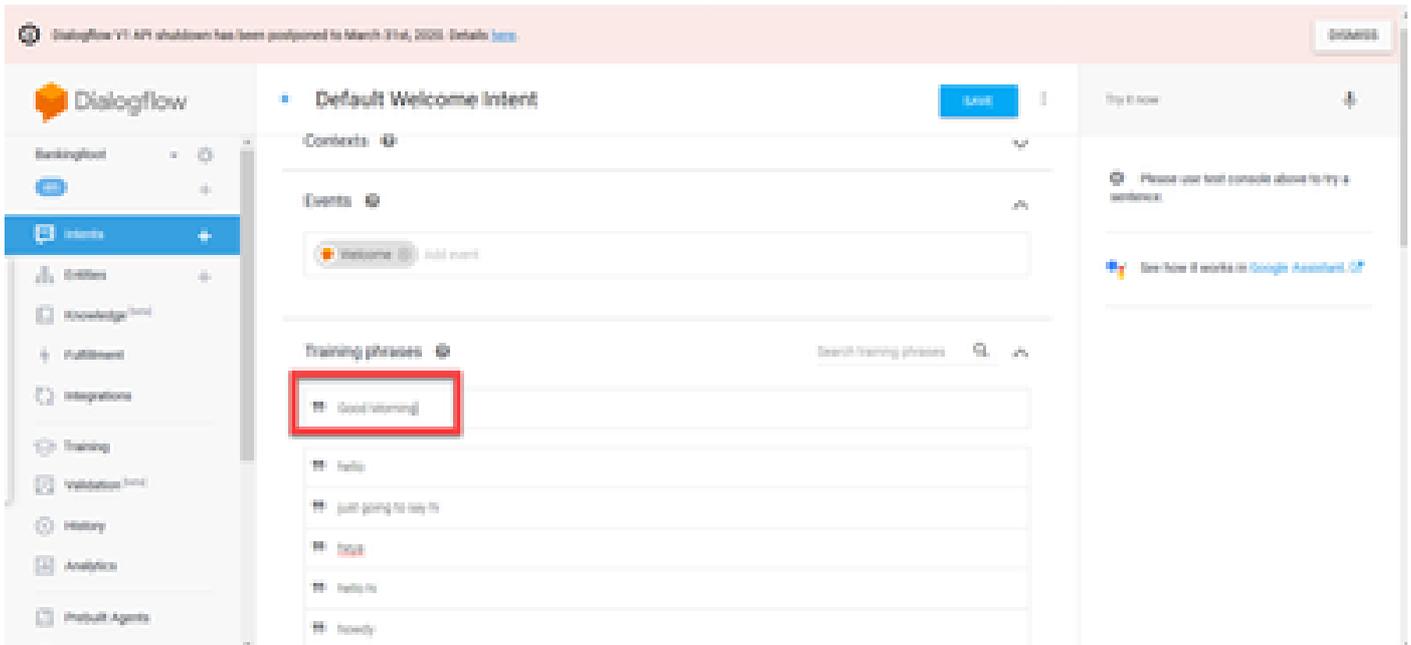
 **Note: hello** could be defined as the default welcome intent in the call studio application element **Dialogflow**.



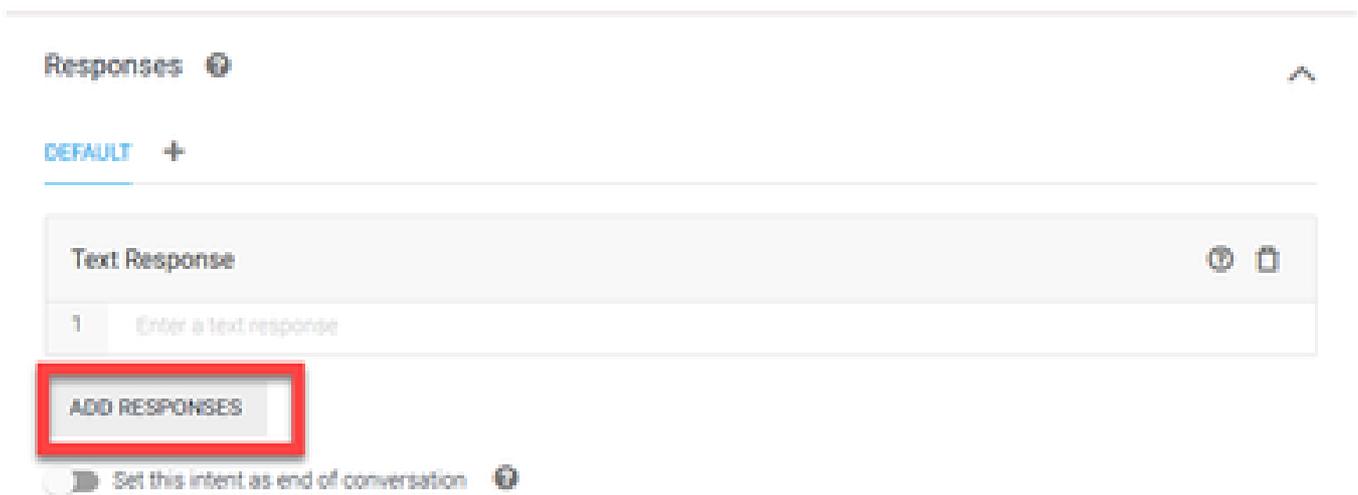
Step 7. Click on **Default Welcome Intent**.



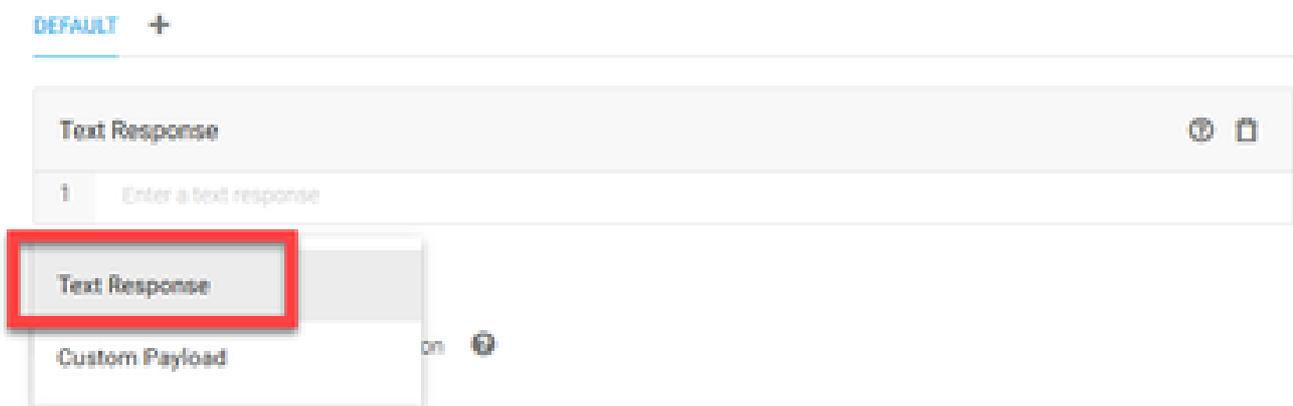
Step 8. Add **hello**, **Good Morning** and **Good Afternoon** to the **Training phrases**. Type them in the text form and press the enter key after each of them.



Step 9. Now scroll down to **Responses**, and click on **ADD RESPONSES**.

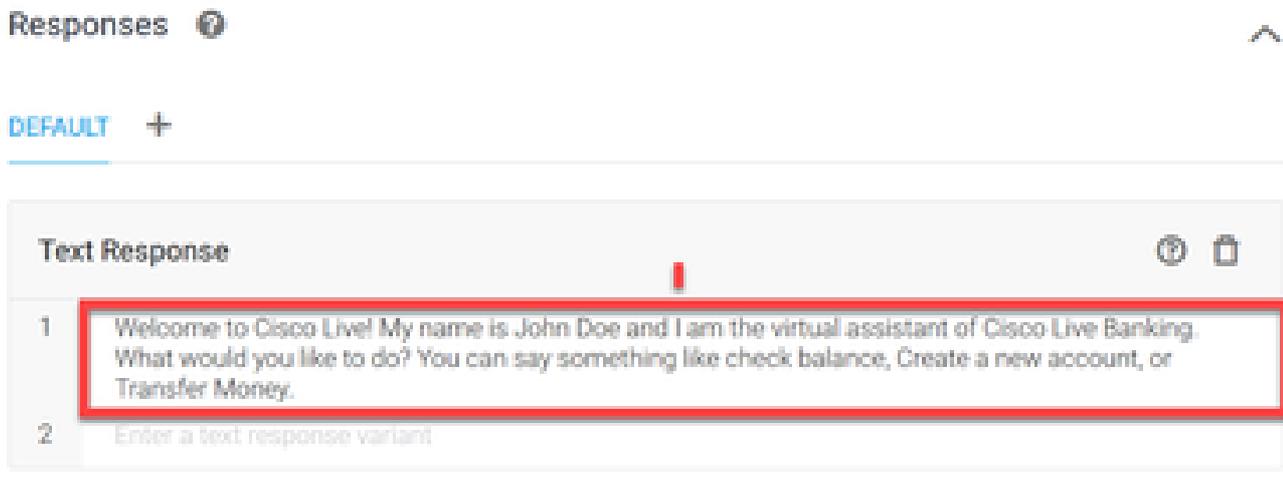


Step 10. Select Text Response.

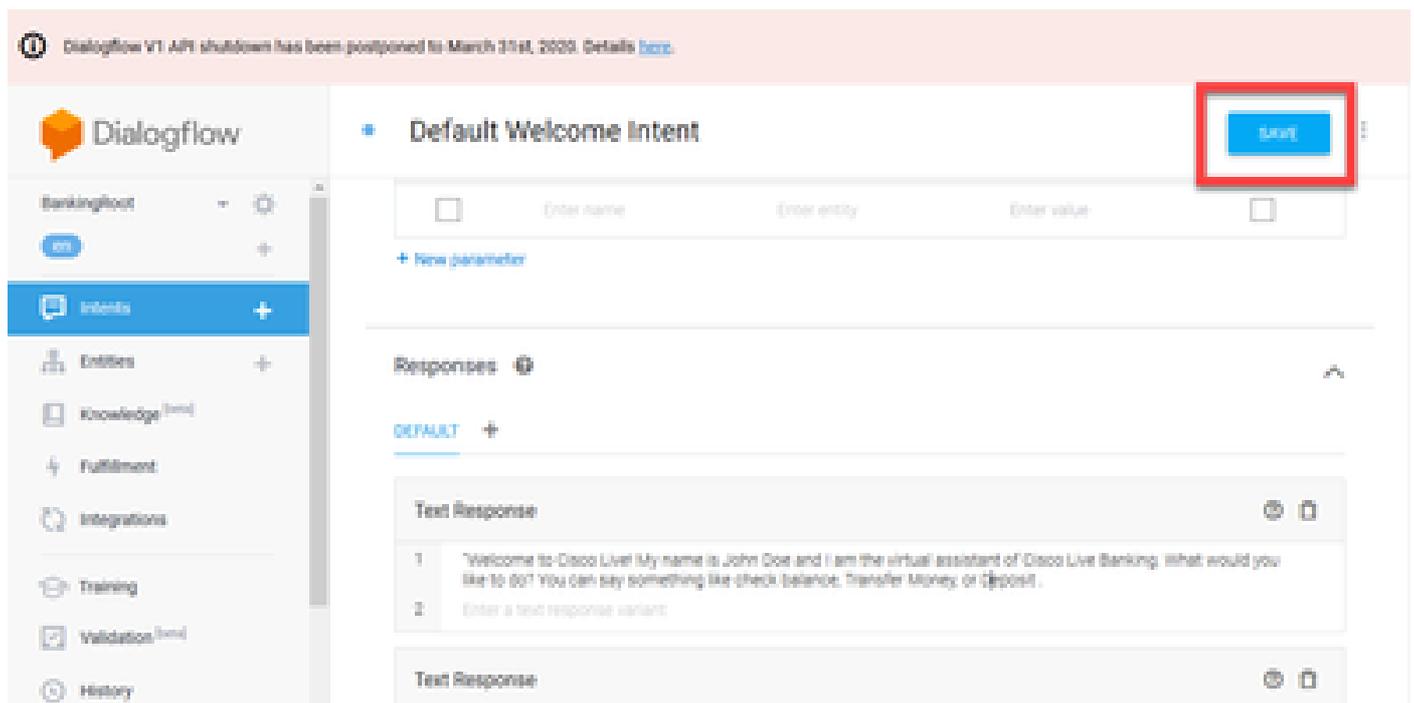


Just as you did with the training phrases, add a proper response. This is the first message the user receives from agent. In order to make your agent sound more natural and conversational, think of a normal

conversation and imagine what an agent would say. Still, it's a good practice to let the user know that the interaction is with an Artificially Intelligent (AI) agent. In this scenario a Cisco Live Banking application is used as an example, so you can add something like: Welcome to Cisco Live! My name is John Doe and I am the virtual assistant of Cisco Live Banking. What would you like to do? You can say something like Check Balance, Create a new account, or Transfer Money.

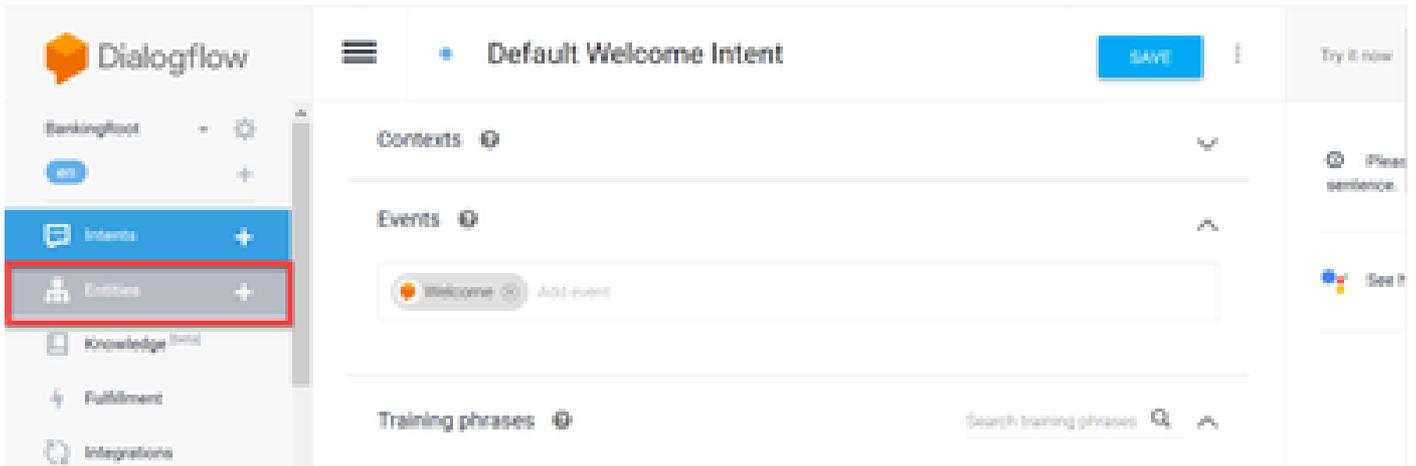


Step 11. Click **SAVE**.

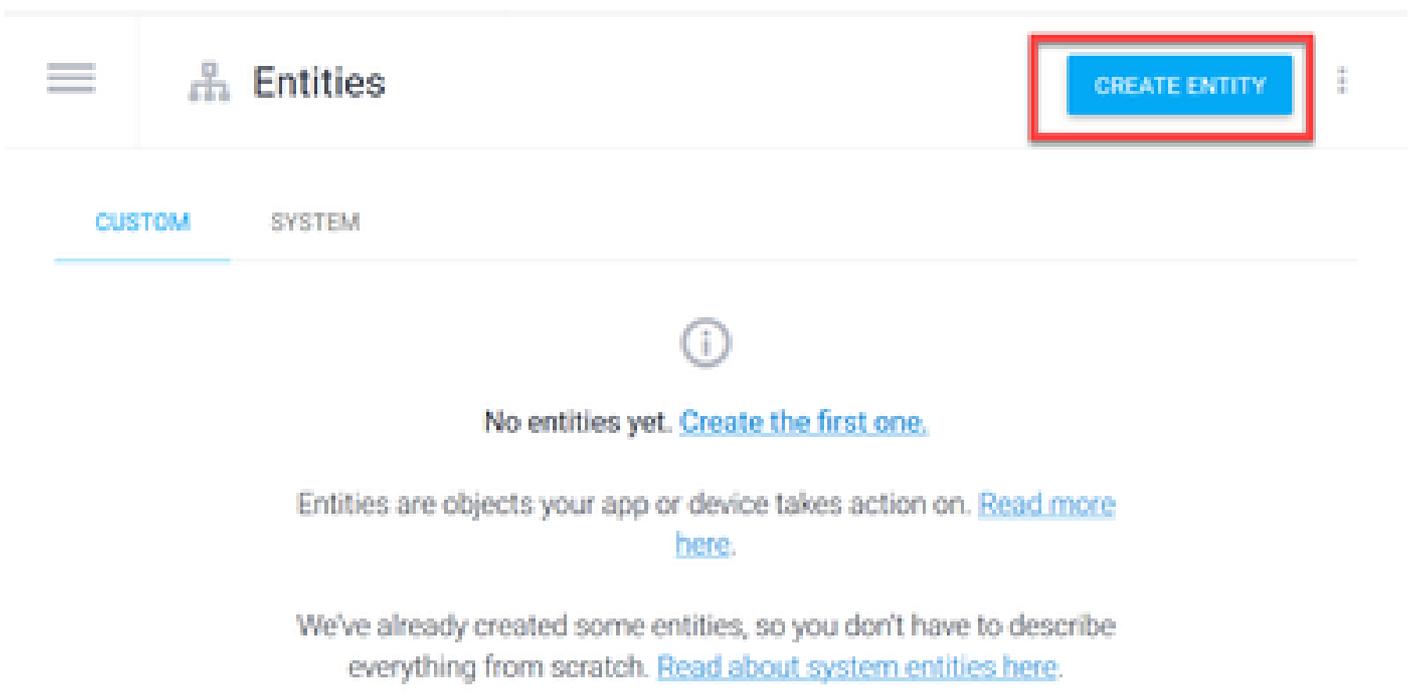


Before you create more intents, create the entities. An **Entity** is a property or a parameter which can be used by **Dialogflow** to answer the request from the user — the entity is usually a keyword within the intent such as an account type, date, location, etc. So before you add more intents, add the entities: Account Type, Deposit Type, and Transfer Type.

Step 12. On the Dialogflow Menu, click on **Entities**.



Step 13. On the Entities window, click on **CREATE ENTITY**.



Step 14. On the Entity name type AccountType. In the Define Synonyms field, type: Current, Loan and Savings and click SAVE.

AccountType

SAVE

Define synonyms Regex entity Allow automated expansion Fuzzy matching

Separate synonyms by pressing the enter, tab or ; key.

| | |
|---------|--------------------------|
| Current | Current |
| Loan | Loan |
| Savings | Savings Enter synonym |
| | Click here to edit entry |
| | Click here to edit entry |

Step 15. Navigate back to the Dialogflow menu and click on Entities once again. Then, in the Entities window, click **CREATE ENTITY**.

>

Entities

CREATE ENTITY

CUSTOM SYSTEM

Search entities

@ AccountType

Step 16. On the Entity name type: DepositType. On the Define synonyms field, type: Cash, and Check, and click on **SAVE**.

Entity: **DepositType** [SAVE]

Define synonyms Regexp entity Allow automated expansion Fuzzy matching

Separate synonyms by pressing the enter, tab or ; key.

| | |
|-------|--|
| Cash | Cash |
| Check | Check <input type="text" value="Enter synonym"/> |

Click here to edit entry  

Click here to edit entry

Step 17. You can create more entities such as: TransferType and on the Define synonyms field type: Cash, Pay Pal, PayTM and Wire Transfer, etc.

Entity: **TransferType** [SAVE]

Define synonyms Regexp entity Allow automated expansion Fuzzy matching

Separate synonyms by pressing the enter, tab or ; key.

| | |
|---------------|--|
| Cash | Cash |
| Pay Pal | Pay Pal |
| PayTM | PayTM |
| Wire Transfer | Wire Transfer <input type="text" value="Enter synonym"/> |

Step 18. Create the Account Holder entity. In the Entity name field type AccountHolder; in the Define Synonyms field.

AccountHolder
SAVE
⋮

Define synonyms ?
 Regexp entity ?
 Allow automated expansion

 Fuzzy matching ?

| | | |
|-------------|--|---------------|
| Amit | Amit | |
| X Bank | X Bank | |
| Y bank | Y bank | |
| Riyad | Riyad | |
| Dallas Bank | <input style="width: 100px;" type="text" value="Dallas Bank"/> | Enter synonym |
| RTP bank | RTP bank | |

Step 19. Now, continue the agent education with all the possible questions received in the banking system and the typical responses. Create the next intents: CheckBalance, TransferMoney. For CheckBalance intent you can add the training phrases shown in the image:

☰
CheckBalance

SAVE

⋮

Training phrases ?
Search training phrases 🔍 ^

🗨️ I want to check the balance of my savings account

🗨️ I wish to find out my current account balance

🗨️ I want to check my balance

You can also add this response:

The screenshot shows the Dialogflow console for the 'CheckBalance' intent. On the left is a navigation menu with options like 'Intents', 'Entities', 'Knowledge', 'Fulfillment', 'Integrations', 'Training', 'Validation', 'History', and 'Analytics'. The main area displays the 'Parameters' table:

| REQUIRED | PARAMETER NAME | ENTITY | VALUE | IS LIST |
|--------------------------|----------------|---------------|----------------|--------------------------|
| <input type="checkbox"/> | AccountTy | @Account Type | \$AccountT ype | <input type="checkbox"/> |
| <input type="checkbox"/> | Enter name | Enter entity | Enter value | <input type="checkbox"/> |

Below the parameters, the 'Responses' section is shown with a 'DEFAULT' response. The response text 'Your balance is 2000\$. What else I can help you with?' is highlighted in a red box.

Step 20. You can add the rest of the Intents(TransferMoney, CreateAccount and Exit), Training Phrases, parameters and responses.

Note: For more information about Google Dialogflow configuration navigate to: [DialogFlow Virtual Agent](#)

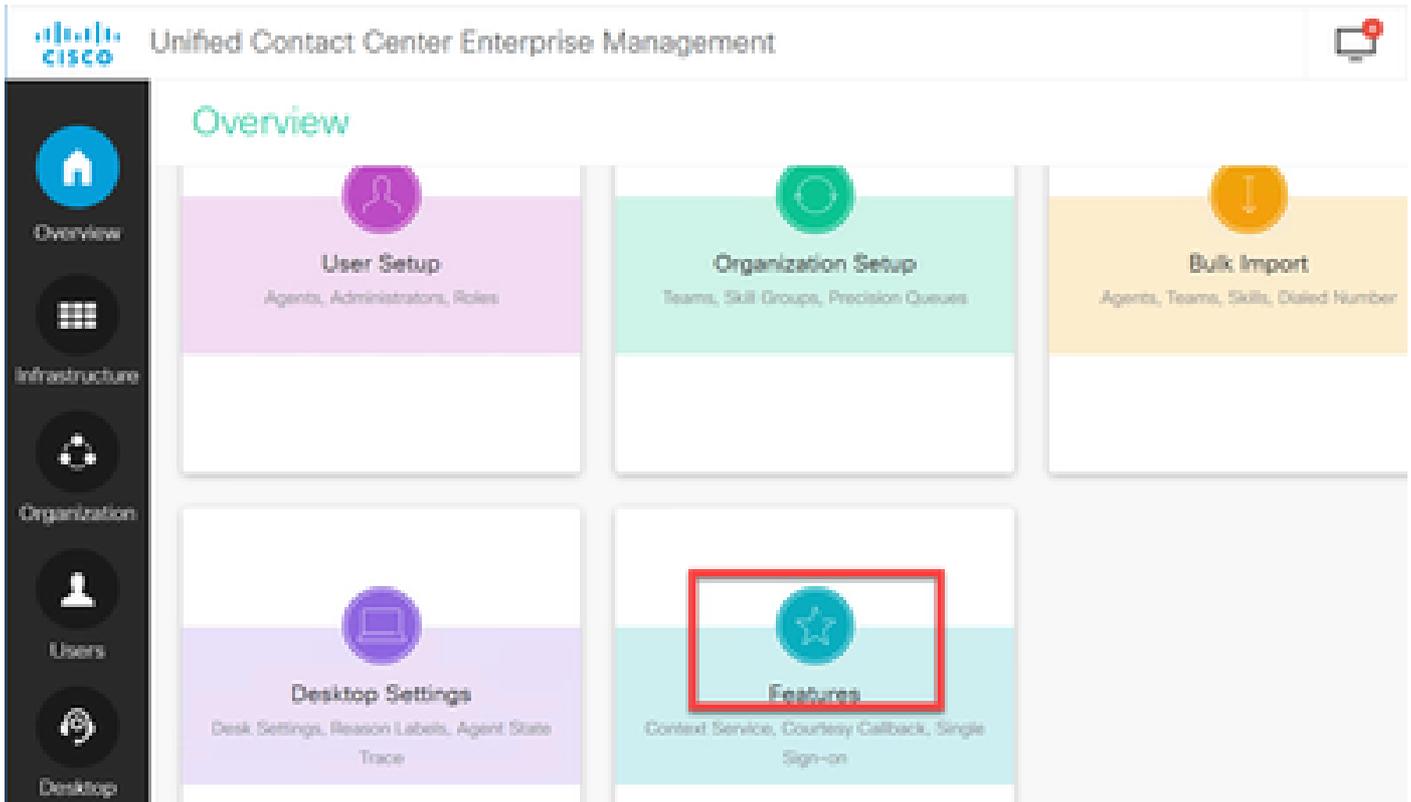
CVVB Speech Server Configuration

Speech Server is a new component integrated into the Cisco VVB. The Speech Server interacts with Google Dialog Flow through an open source Remote Procedure Call (gRPC) system initially developed by [Google](#)

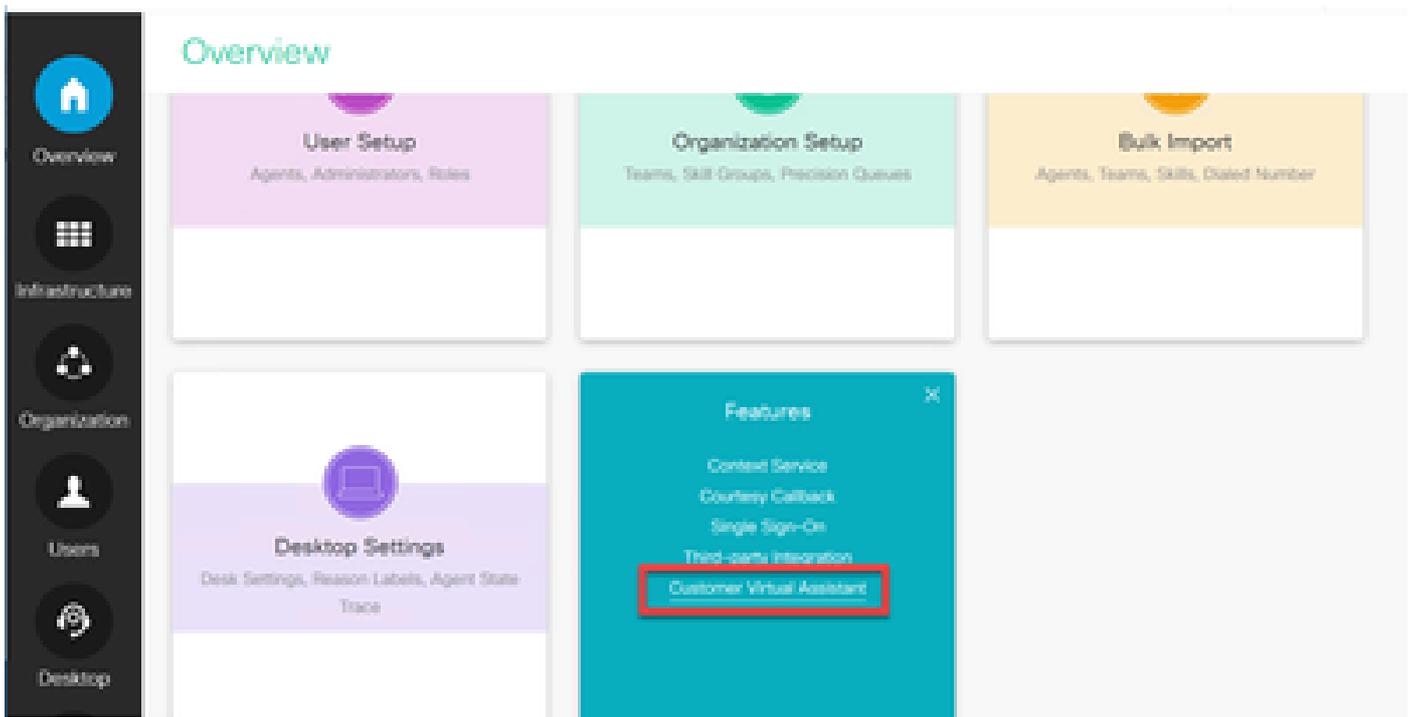
Step 1. Exchange certificates between PCCE Admin Workstation (AW), CVP and CVVB if you have not done so. If your deployment is on UCCE, exchange the certificates between CVP New Operations Manager server (NOAMP), CVP and CVVB.

Note: Please refer to these documents for PCCE certificate exchange: [Self-Signed Certificates in a PCCE Solutions](#) and [Manage PCCE Components Certificate for SPOG](#) . For UCCE please refer to [Self-Signed Certificate Exchanged on UCCE](#) .

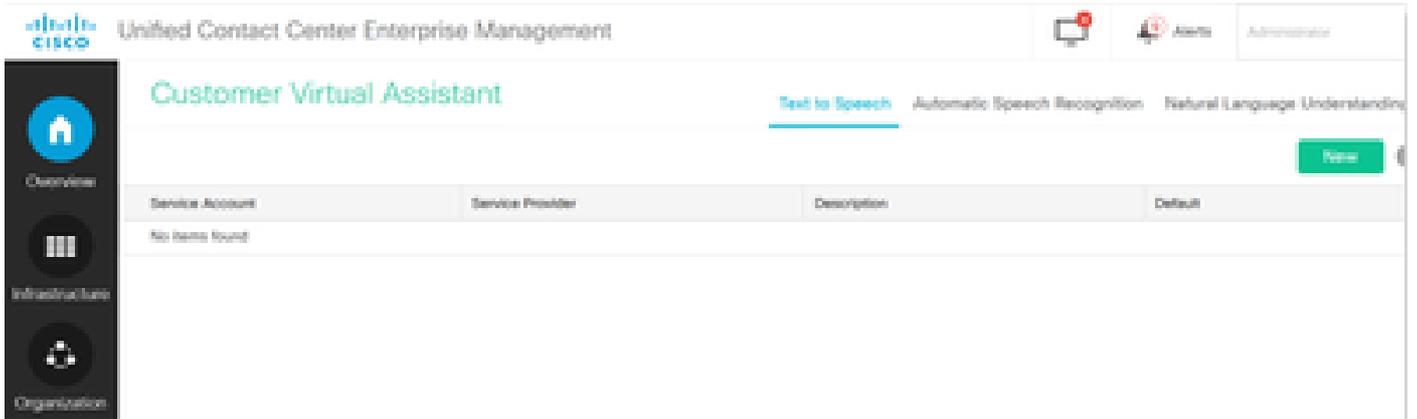
Step 2. On PCCE , Open the CCE Admin / Single Plane of Glass (SPOG) interface. if your depymnt is on UCCE, do these steps on the NOAMP server.



Step 3. Under Features, select **Customer Virtual Assistant**.



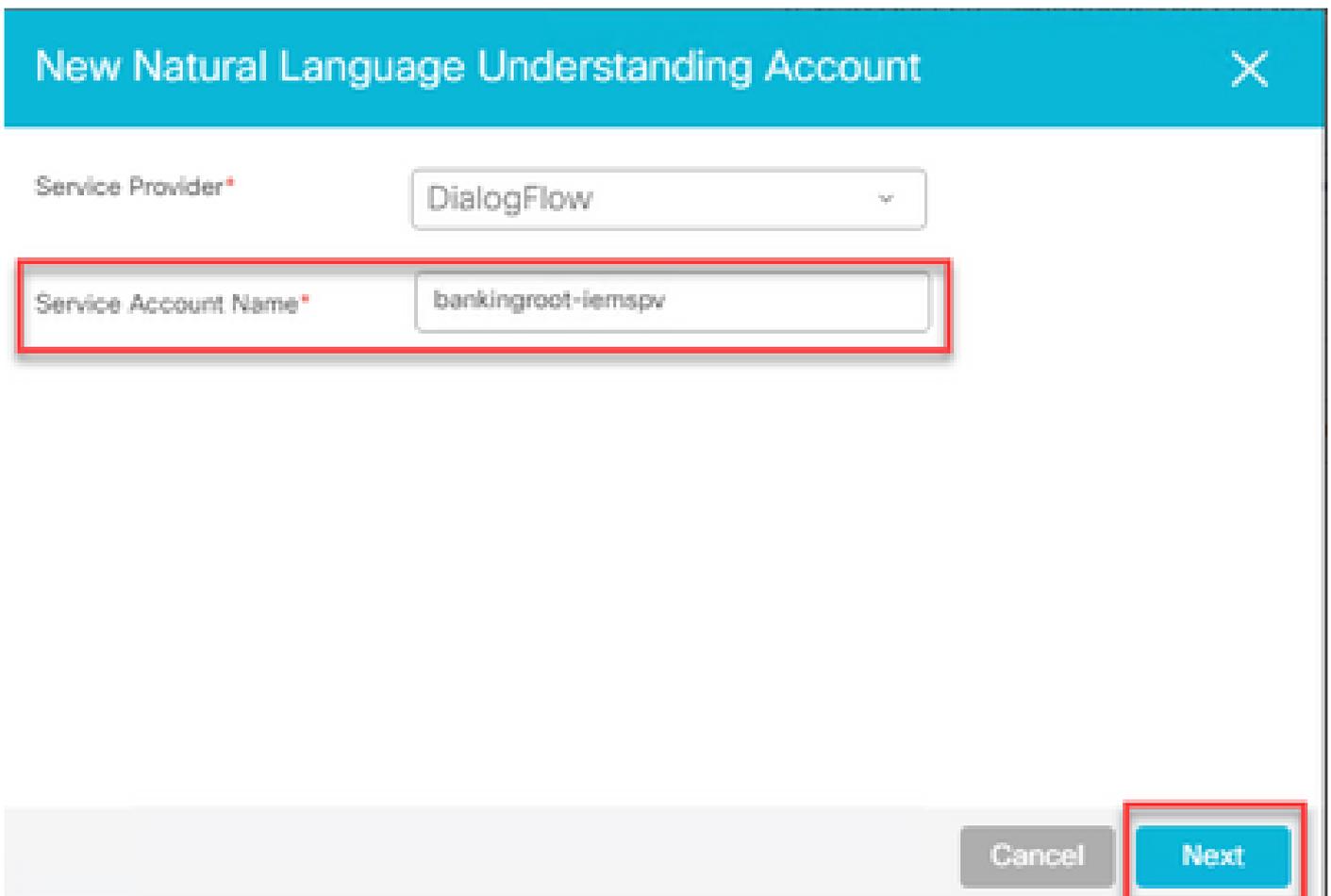
Step 4. Now you should see three tabs: Text to Speech, Automatic Speech Recognition and Natural Language Understanding.



Step 5. Click on the **Natural Language Understanding** and then click on **New**.

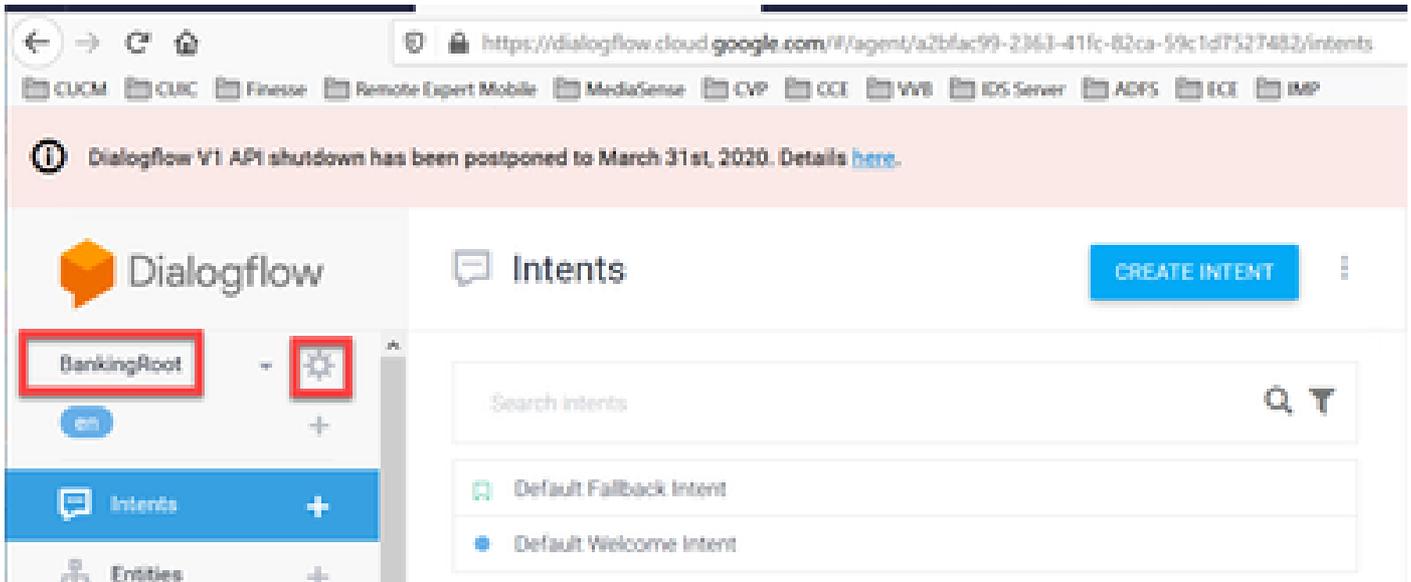
Step 6. In the New Natural Language Understanding Account window, select **Dialogflow** as the service provider.

Step 7. For the Service Account Name, you need to provide the Google Project related to the Virtual Agent you created in Google Dialogflow.

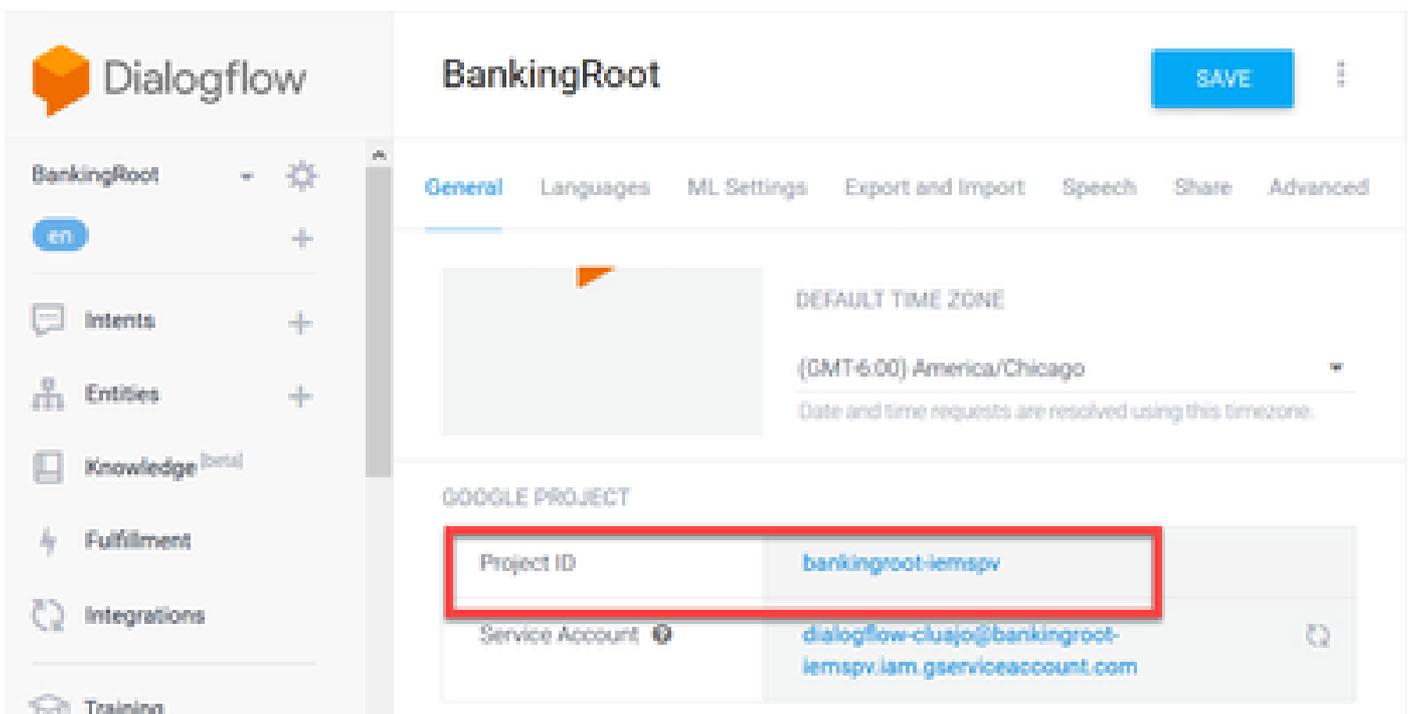


In order to identify the project related to the created virtual agent, follow this procedure:

a. Log in to your DialogFlow account (dialogflow.com), select the agent created and click on the settings icon.



b. Scroll down in the settings window on the right-hand side and you see the Service Account and the project ID. Copy the Project ID, which is the Service Account Name that you need to add in the Speech Server configuration.

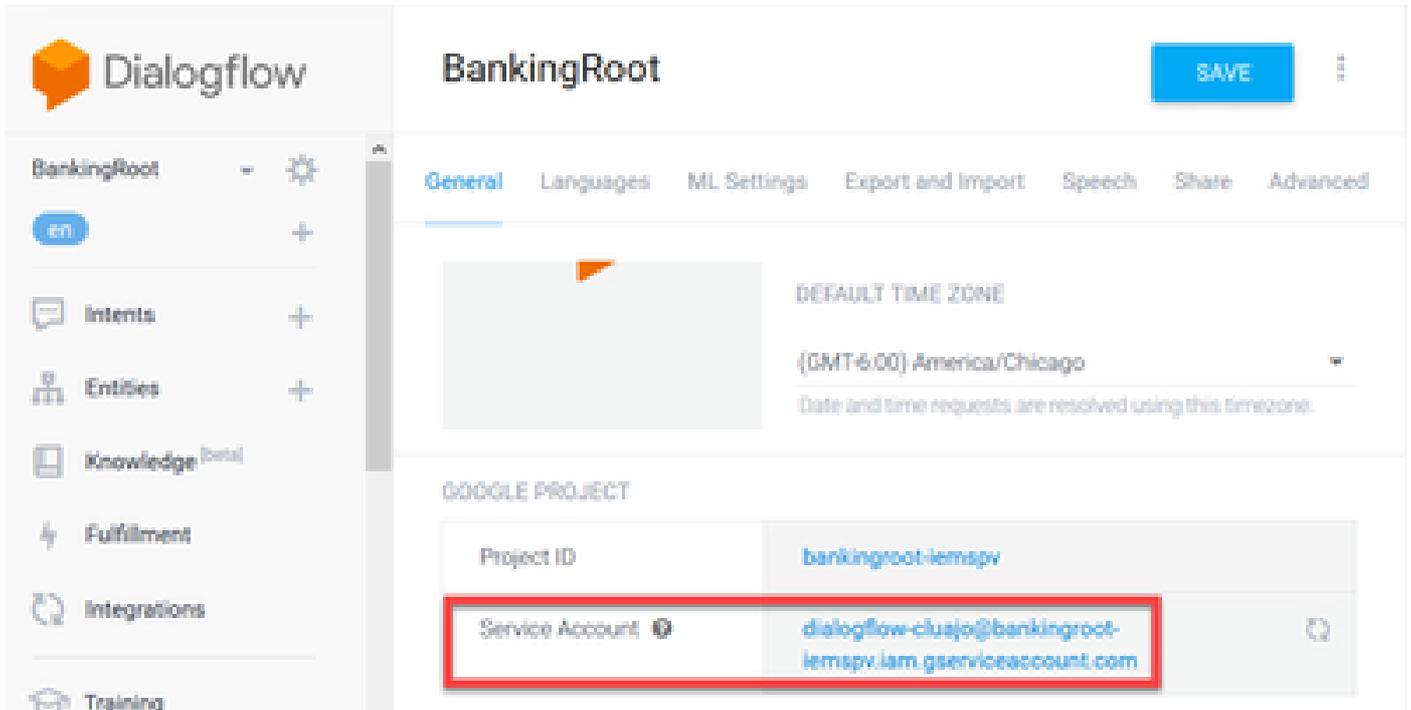


Step 8. In order to use the Google Dialog flow APIs required to identify and respond to customer intent, you need to obtain a private key associated with a virtual agent's service account.

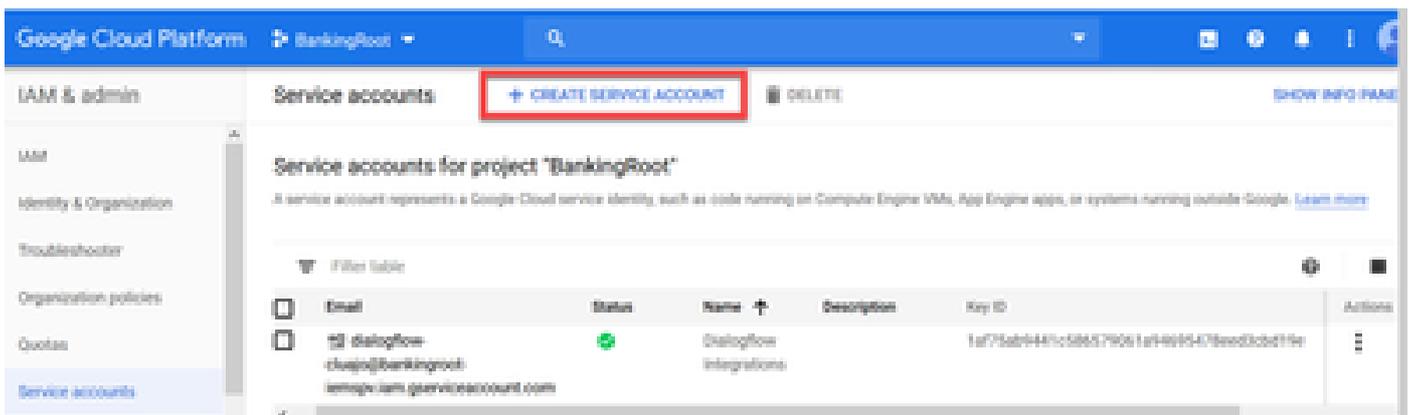
The private key is downloaded as a JSON file upon creation of the Service Account. Follow this procedure in order to obtain the virtual agent private key.

Note: It is mandatory to create a new service account instead of use any of the default Google service accounts associated with the project.

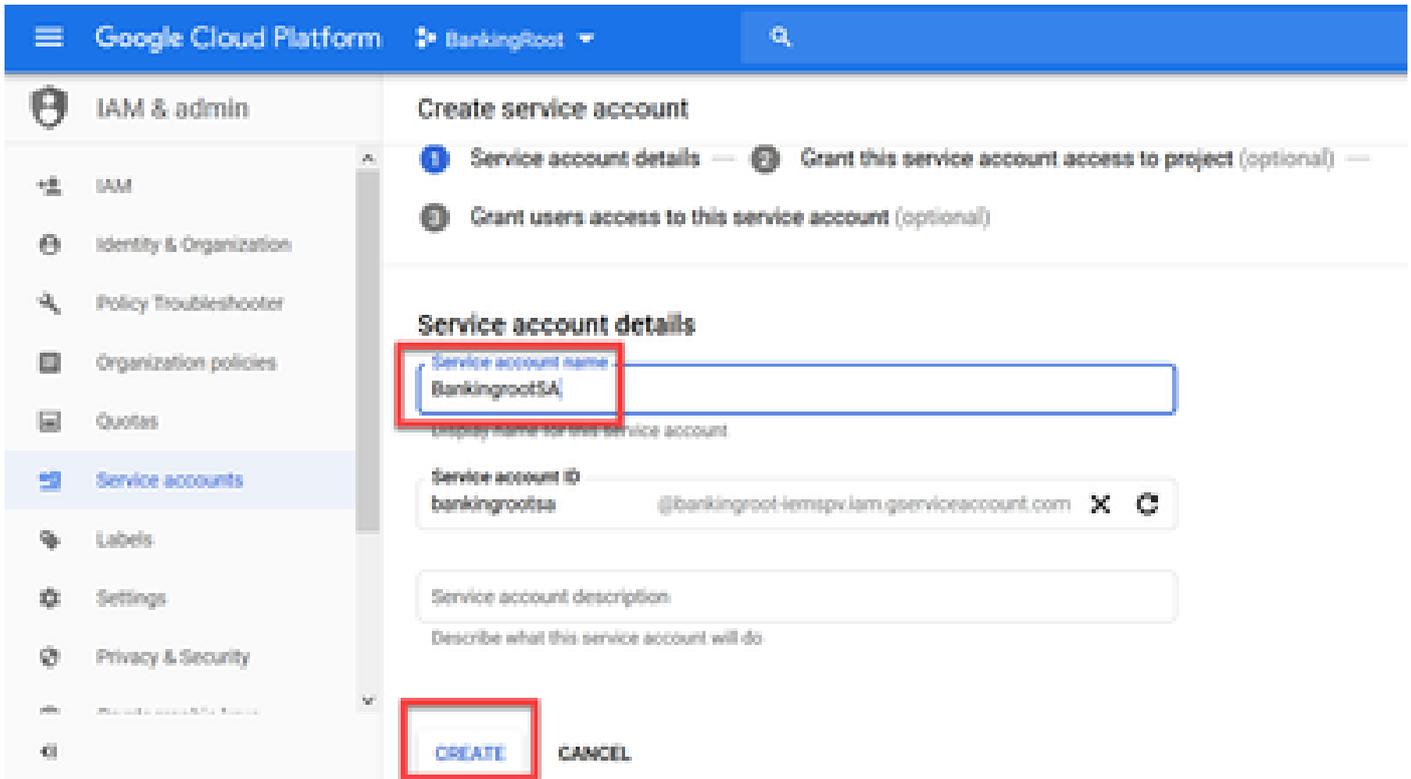
a. Under the Google Project section, click on the Service Account URL.



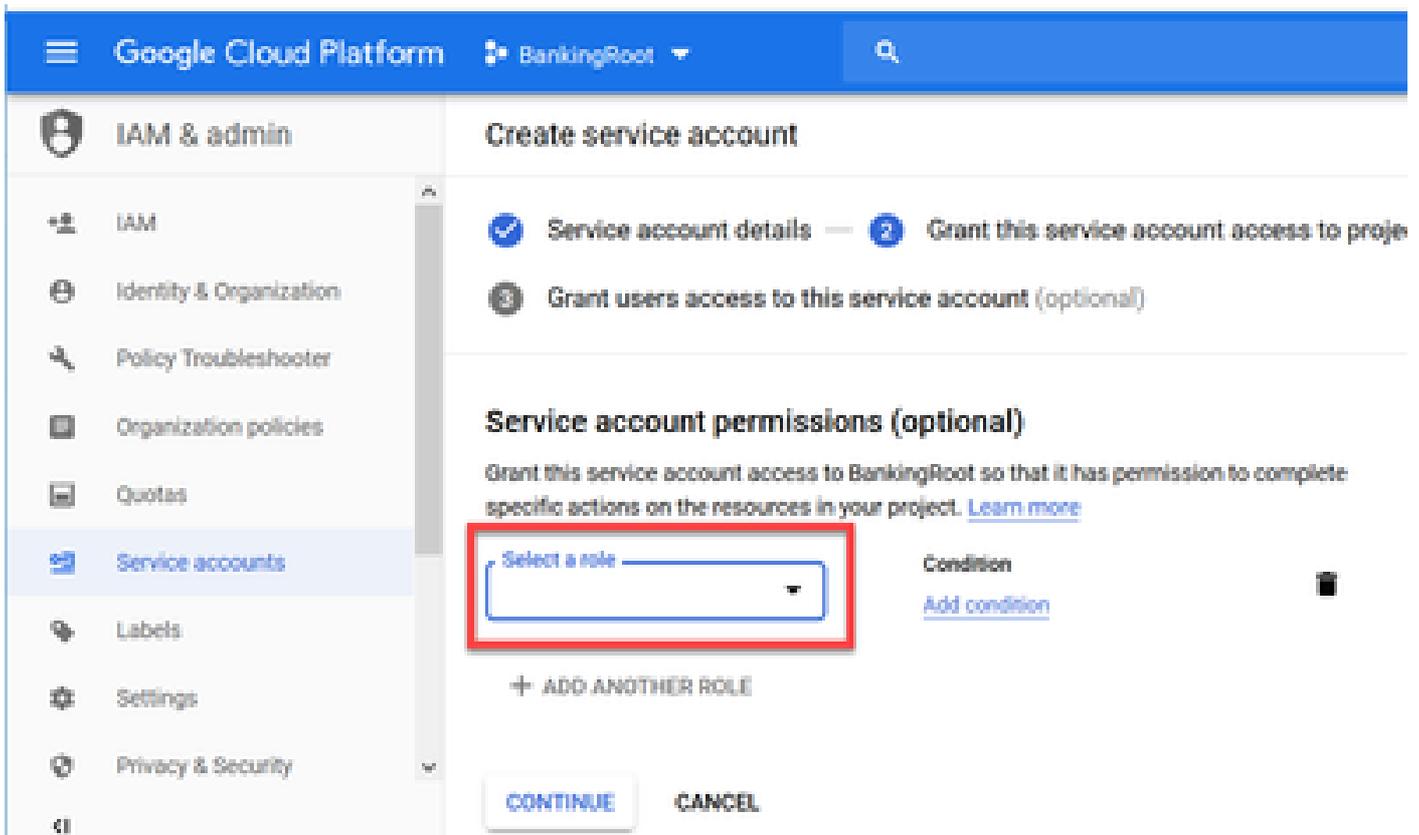
b. This takes you to the Google Cloud Platform Service Accounts page. Now, you first need to add roles to the Service Account. Click on the **Create Service Account** button at the top of the page.



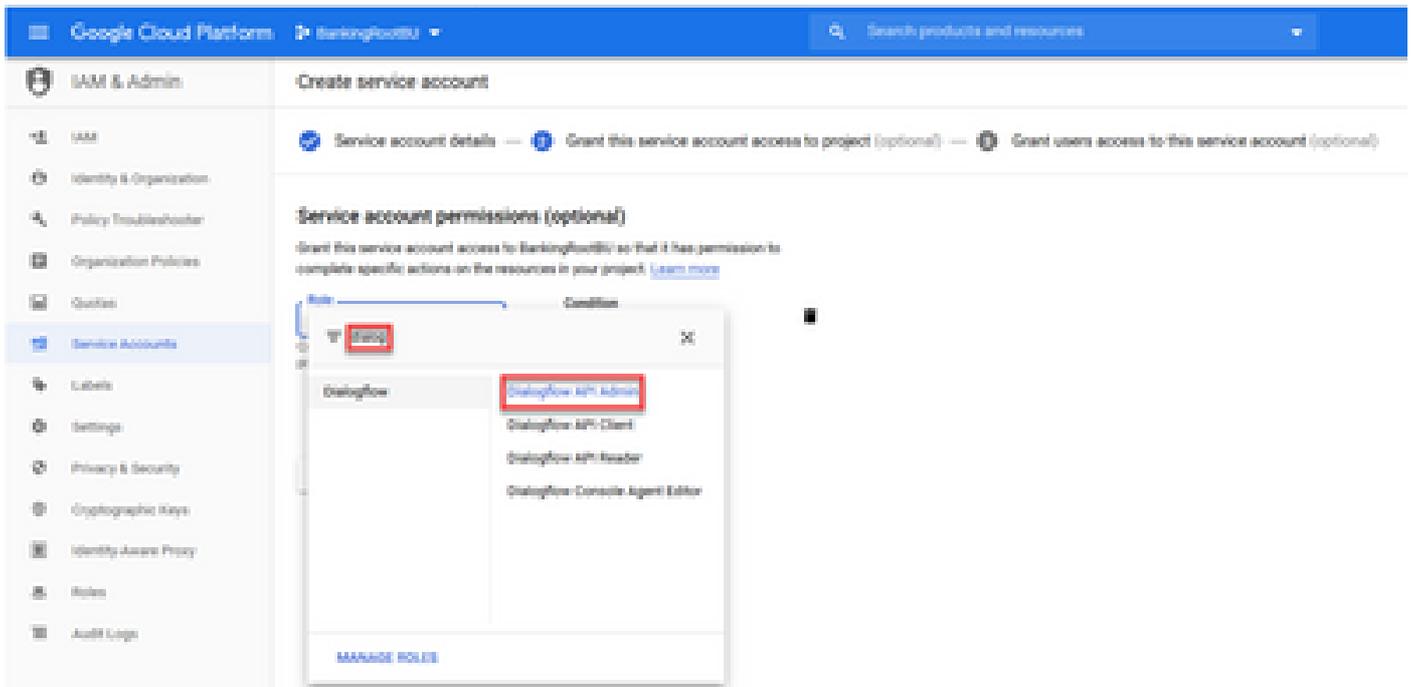
c. In the pop-up, enter a name for the service account. In this case enter BankingRootSA and click on **CREATE**.



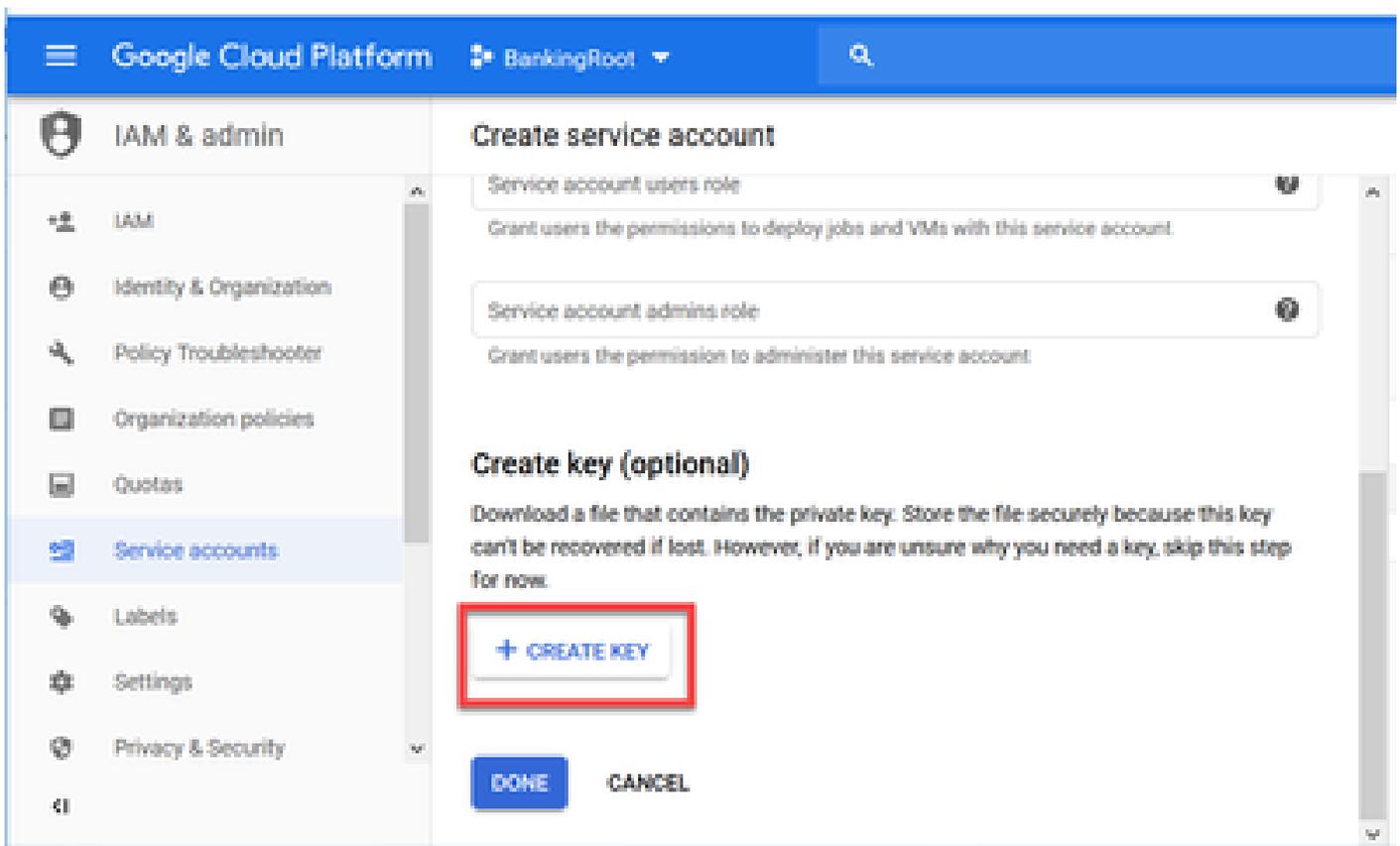
d. Click on **Select a role**.



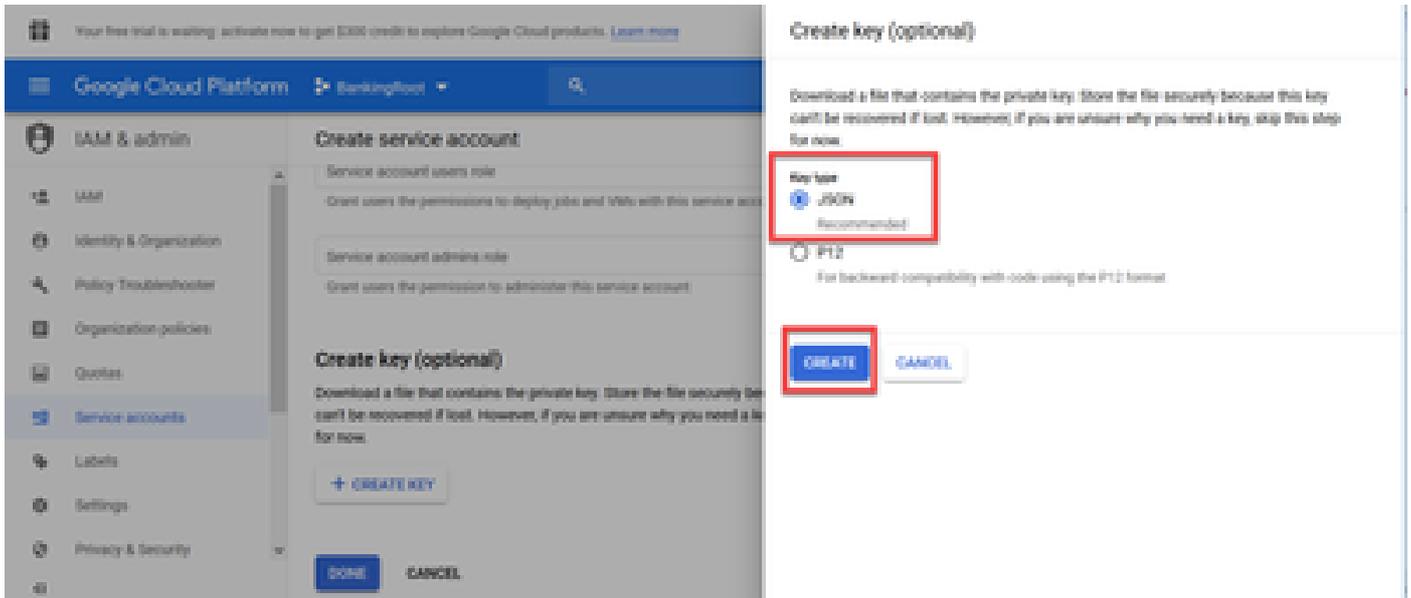
e. Under the **Dialogflow** category, select the desired role. Select Dialogflow API Admin and click Continue.



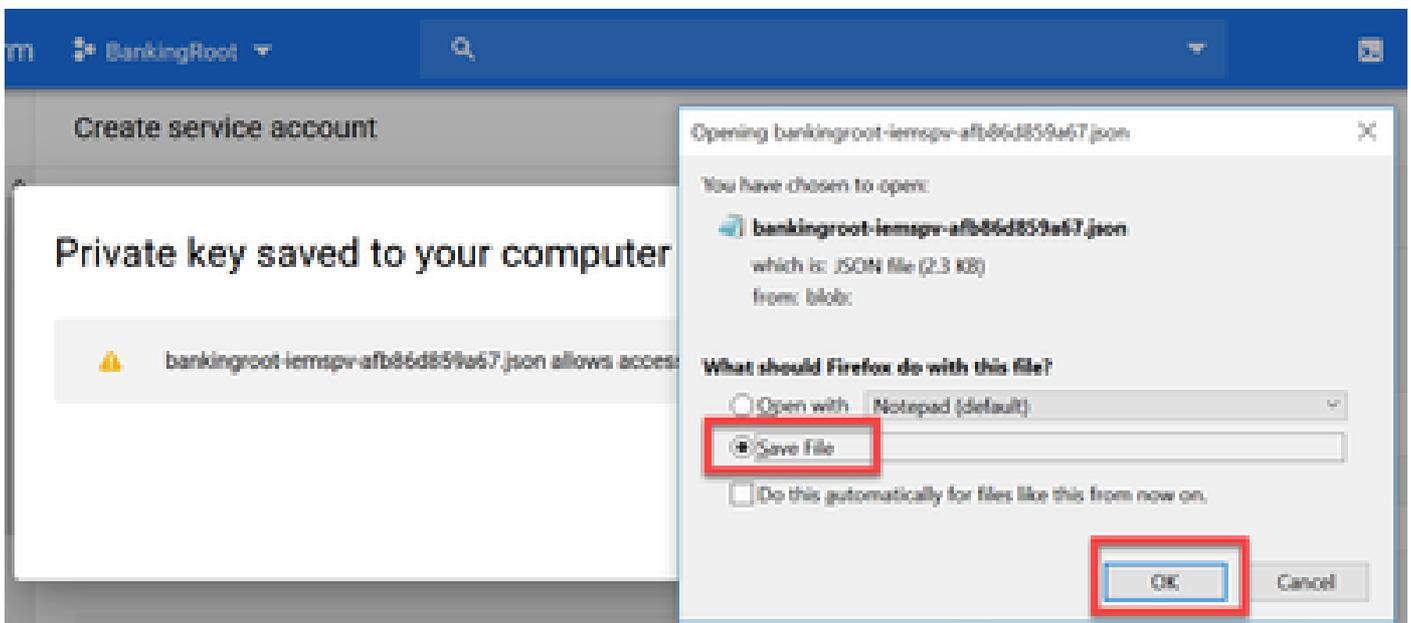
f. Scroll down and select **CREATE KEY**.



g. In the private key window, make sure **JSON** is selected for **Key type** and click the **CREATE**.



h. Download of the JSON file starts. Check the box **Save File**, choose a location to save it and confirm.



⚠ Caution: You can only download this JSON file once, so make sure to save the file and keep it somewhere safe. If you lose this key or it becomes compromised, you can use the same process to create a new key. The JSON file is saved to the C:\Download folder

i. Once complete, you see a pop-up with a confirmation message. Click **Close**.

Step 9. After you click **NEXT** on the NLU Account window, you need to provide the authentication key.

New Natural Language Understanding Account ✕

Service Provider

Service Account Name

Step 10. Add the description. Navigate to folder where you download the JSON file. Edit the file, select all the lines in the file and copy them to the Service Account Key field. Click **Save**.

Customer Virtual Assistant
Text to Speech Automatic Speech Recognition Natural Language Understanding

New Natural Language Understanding Account

Service Account Name

Service Provider

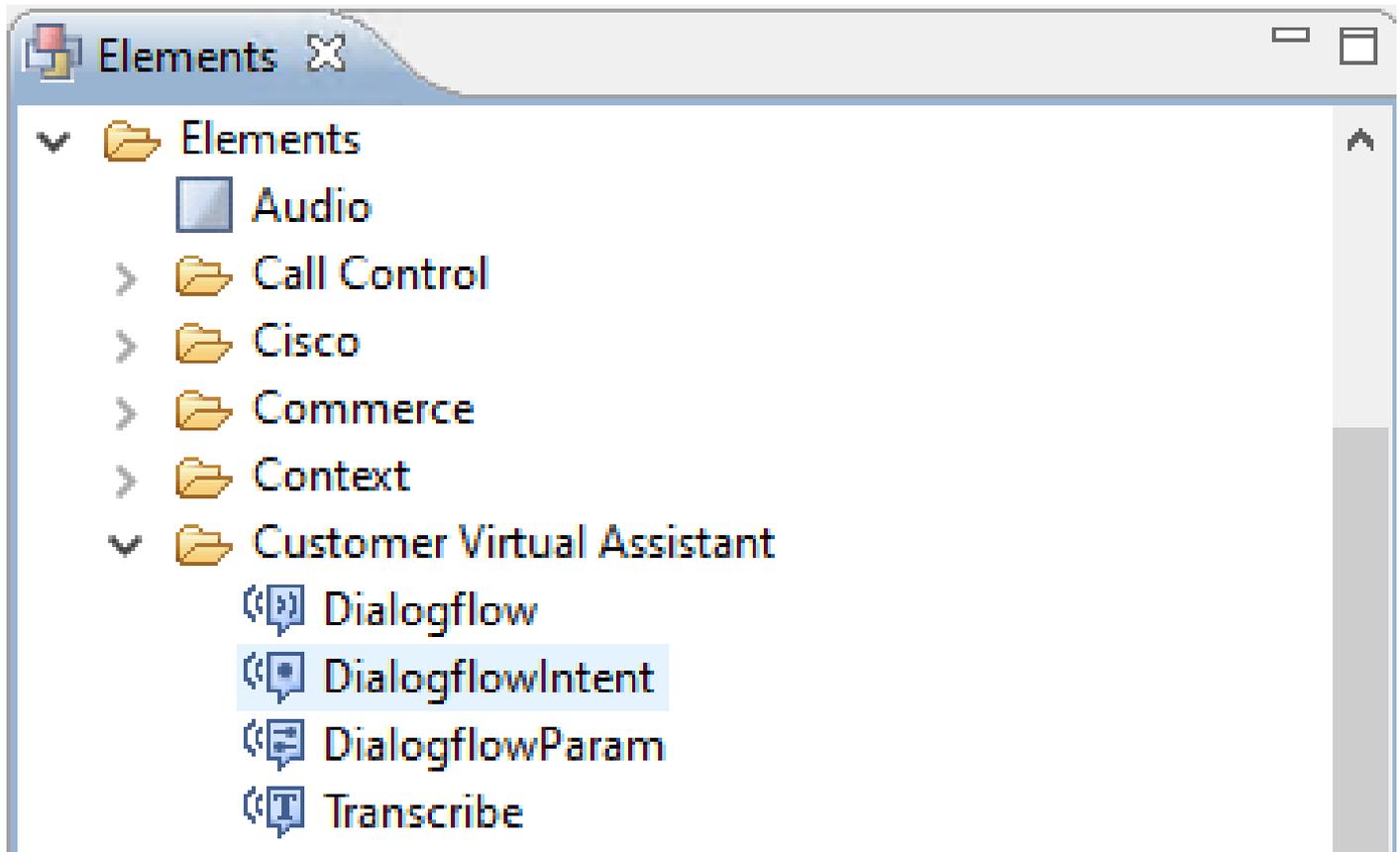
Description

Authentication Details

Service Account Key

CVP Call Studio Elements

CVP Call Studio release 12.5 has been enhanced and these four elements (as shown in the image) have been added in order to ease the configuration of the CVA feature.



Here is a brief description of each element:

Dialogflow

Dialogflow has been created in order to engage and manage the ASR, NLU, and TTS services from the Cloud. Dialogflow helps simulate a Hosted IVR deployment in which all speech services are engaged by Google Dialogflow and the entire business logic is controlled and driven from the Cloud.

DialogflowIntent

DialogflowIntent has been created for Cloud services to be engaged for recognition (ASR service) and Intent identification (NLU service). Once the intent has been identified and passed on to the CVP VXML server, the handle of the intent and any further actions can be performed at the CVP Call Studio script. Here flexibility has been provided for application developers to engage TTS services from the Cloud or from on-premises.

DialogflowParam

DialogflowParam works in conjunction with the DialogflowIntent element. In a typical on-premises based IVR deployment, when customer intent is identified and passed to the VXML server, parameter identification is required and should be driven by the CVP application. For example, a typical banking application could analyze missed inputs from customer speech and request remaining mandatory inputs before the entire transaction is processed. Under the above scenario, the DialogflowParam element works in conjunction with the DialogflowIntent element to process the intent that has been identified and add the required parameters.

Transcribe

Transcribe has been created to process customer speech and return text as an output. It basically performs the recognition function and provides text as an output. This element should be used when ASR functionality alone is required.

For more information on parameter setting under each of these elements, please refer to the [Element Specifications guide release 12.5](#).

CVP Call Studio Applications

Cloud Based Intent Processing - Google Based IVR Logic (Dialogflow)

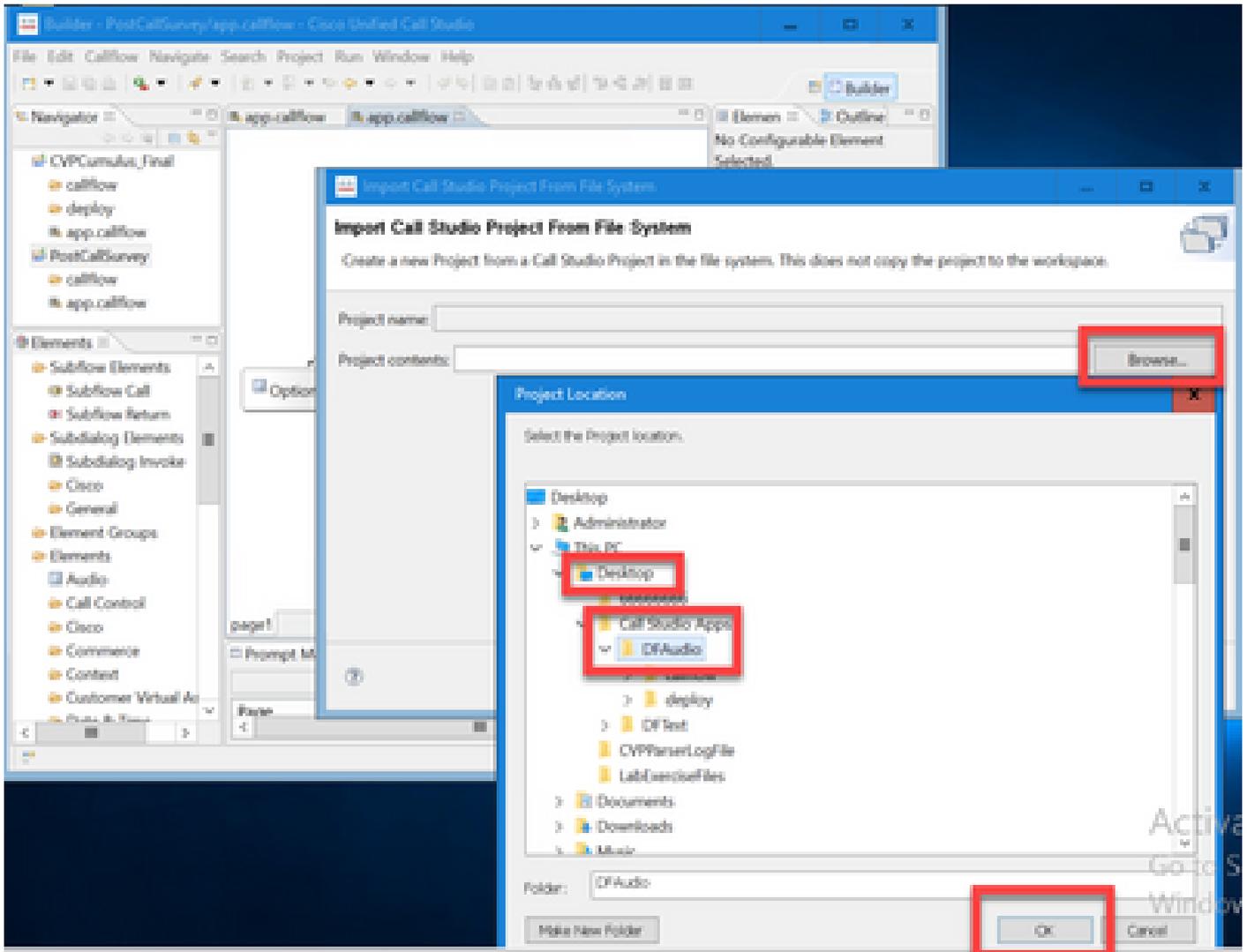
As a call hits a VXML application, the Dialogflow element takes over and starts to process the voice input.

The dialogue with the customer continues and as far as the Google virtual agent is able to identify intents and process them, media is relayed back via TTS services. For every ask from the customer, flow continues in a looped manner around the Dialogflow element and every matched intent is run against a decision box to determine whether the IVR treatment should continue, or whether customer needs to transfer the call to an agent.

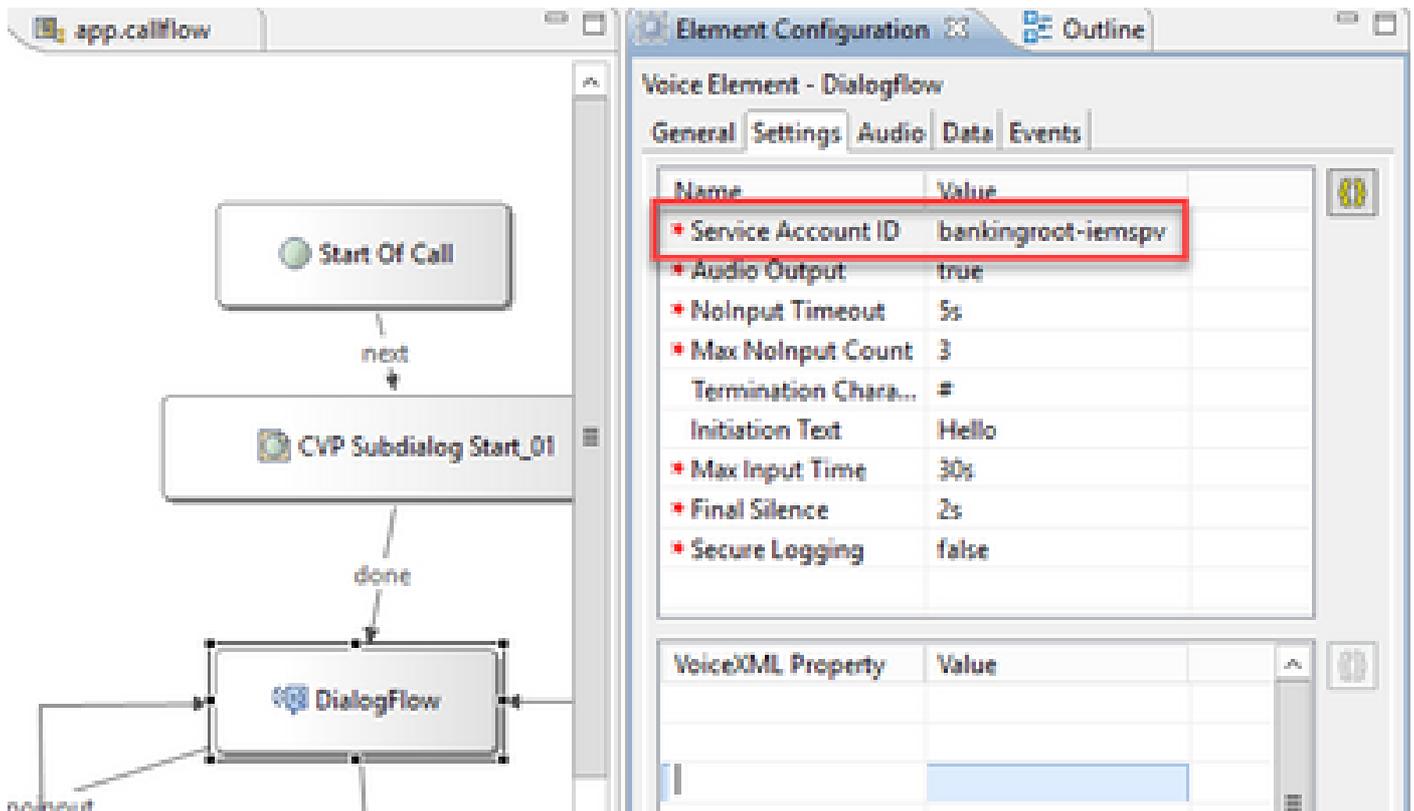
Once the agent transfer decision is triggered, the call is routed to CVP and control is handed over to put the the call in queue and then transfer the call to an Agent.

Here are the configuration steps for a sample call studio application:

Step 1. Import the application to Call studio or create a new one. In this example a call studio application called DFaudio has been imported from Cisco Devnet [Sample CVA Application-DFAudio](#).



Step 2. On the DFAudio application, select the Dialogflow element, and on the right-hand side select the Settings tab. Change the Service Account name to the Project ID which was previously added to the Speech Server, in this example is: **bankingroot-iemspv**



Step 3. Ensure that the Audio Output parameter is sent to true in order to send audio to the Dialogflow virtual agent instead of text.

Step 4. Validate, save and deploy the application to the VXML server.

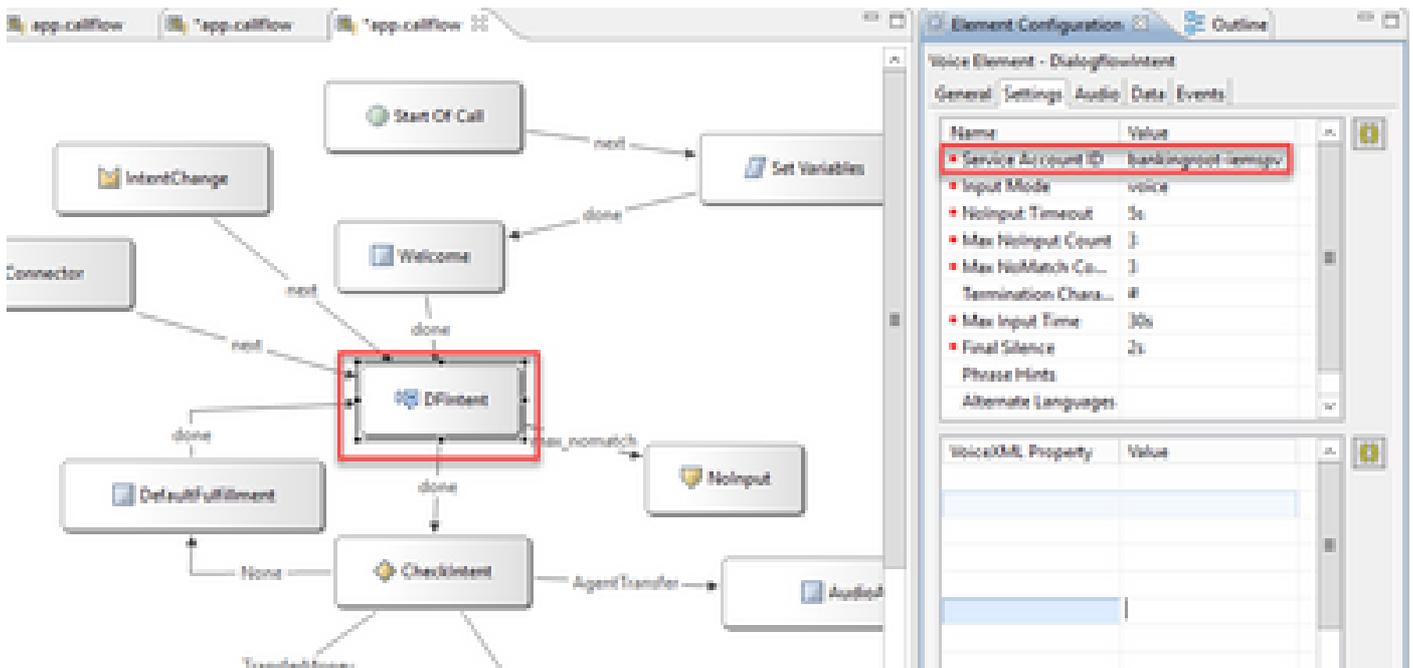
Step 5. Now deploy the application into the VXML server memory. On the CVP VXML server, open Windows Explorer, navigate to C:\Cisco\CVP\VXMLServer and click on **deployAllNewApps.bat**. If the application has been previously deployed to the VXML server, click on the **UpdateAllApps.bat** instead.

Premise Based Intent Processing (DialogflowIntent / DialogflowParam)

In this example, the call flow is related to a banking application in which customers can check their account balance and transfer a certain amount of money from a savings account to another account. The initial transcribe elements collect the identification data from the customer via speech and validate it with the ANI number. Once the end customer identification is validated, call control is handed over to the DialogFlowIntent element in order to identify the ask from the customer. Based on customer input (such as amount to be transferred), the CVP Call Studio application requests remaining parameters from the end customer to further process the intent. Once the money transfer transaction is over, the customer can choose to end the call or can request an agent transfer.

Step 1. Import the application to Call studio or create a new one. In this example a call studio application called DFRemote has been imported from Cisco Devnet [Sample CVA Application-DFRemote](#).

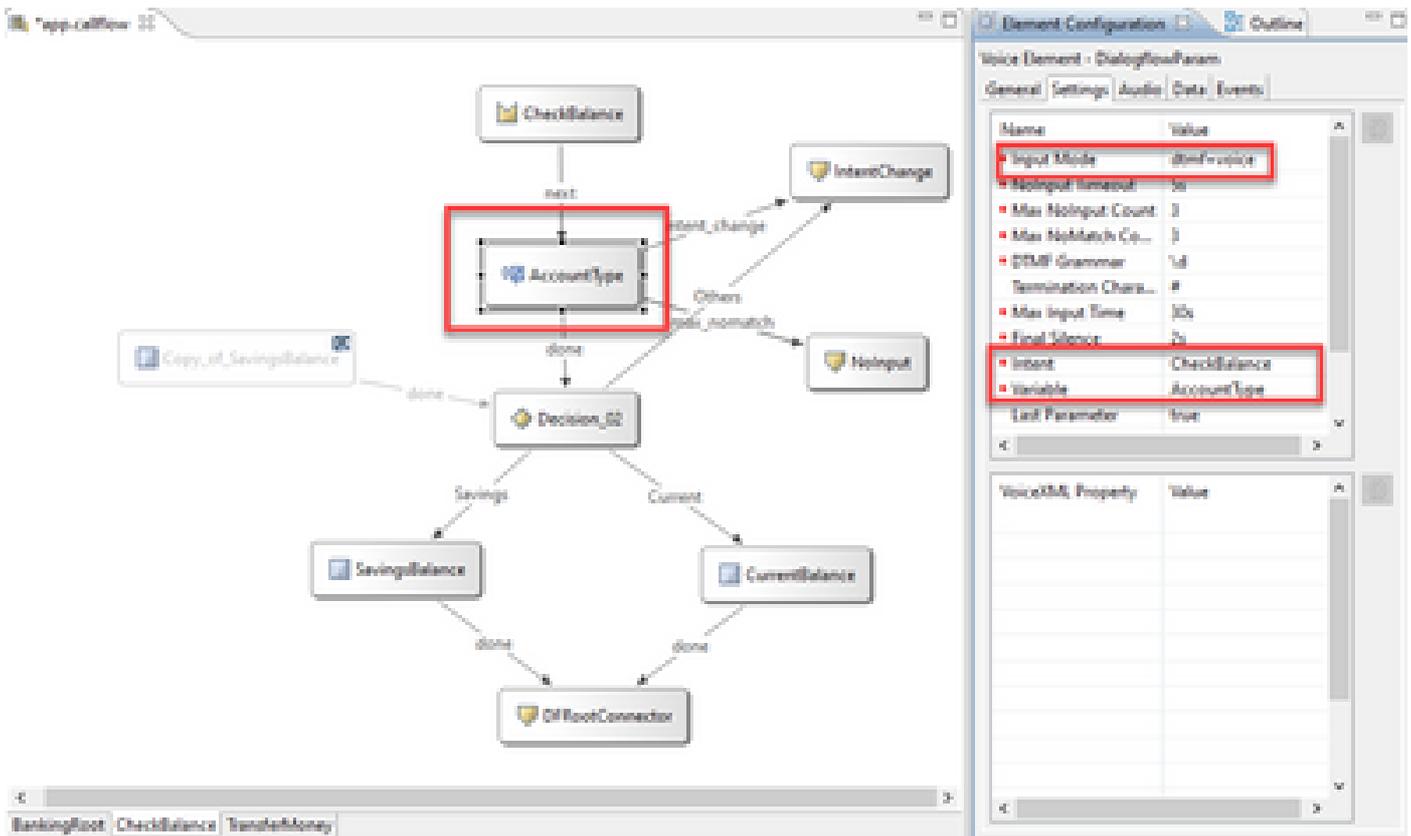
Step 2. In the DFRemote application, select the DialogflowIntent (DFIntent) element and on the right-hand side select the Settings tab. Change the Service Account name to the Project ID which was previously added to the Speech Server, in this example is: **bankingroot-iemspv**



Step 3. Ensure that the Input Mode parameter is set to voice. You can have set it to both, voice and DTMF, but for this element must set to voice because no parameters are collected. When you use the DialogflowParam you can set it to both. In this element is where you actually collect the input parameter from the caller.

Step 4. In this example, When a customer call to check the account balance, the application asks customer to provide the account type by DTMF or speech. This information is collected in the DialogflowParam element (AccountType as shown in the image). In order to collect the required parameters, change the DialogflowParam settings. in the **Input Mode** select dtmf+voice, so the caller can either enter or say the account type. In the **Intent** parameter, type the intent related, in this case **CheckBalance**. And in the **Variable** setting,

select the parameter of the intent, in this case **AccountType**. If this is the last parameter of the intent, set the **Last Parameter** variable to true. For more information on the DialogflowParam settings, please refer to the [Element Specifications guide release 12.5](#).



Step 5. Validate, save and deploy the application to the VXML server.

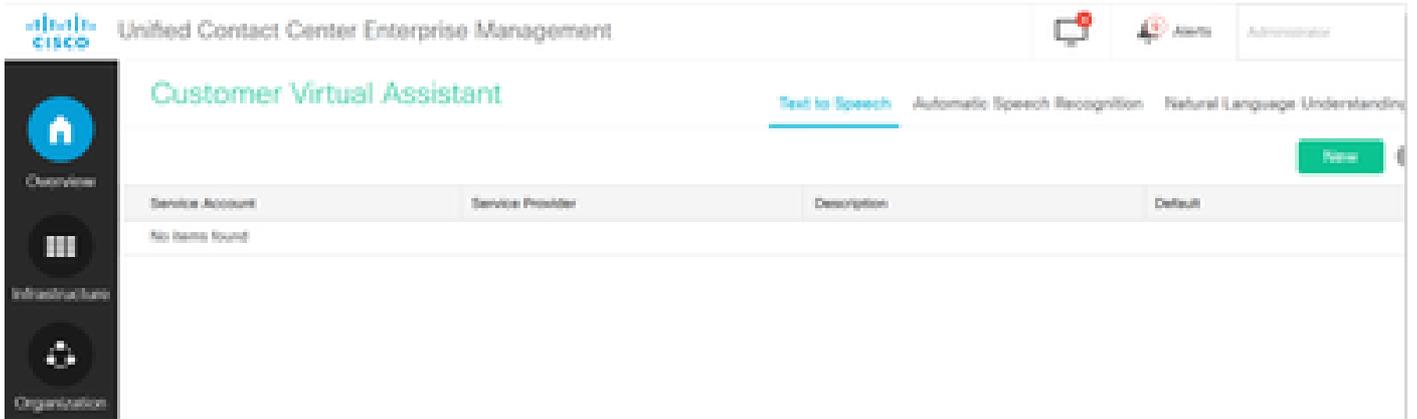
Step 6. Now deploy the application into the VXML server memory. On the CVP VXML server, open Windows Explorer, navigate to C:\Cisco\CVP\VXMLServer and click on **deployAllNewApps.bat**. If the application has been previously deployed to the VXML server, click on the **UpdateAllApps.bat** instead.

Step 7. Copy the JSON file previously downloaded to the C:\Cisco\CVP\Conf directory. The json file name must match the project name, in this case **bankingroot-iemspv.json**.

Step 8. Add the Google TTS and ASR services, if these services are needed, like in this example. if your deployment is on UCCE, add the TTS and ASR via the NOAMP server. On PCCE, open the CCE Admin / Single Plane of Glass (SPOG) interface.

Step 9. Under the Features card, select Customer Virtual Assistant.

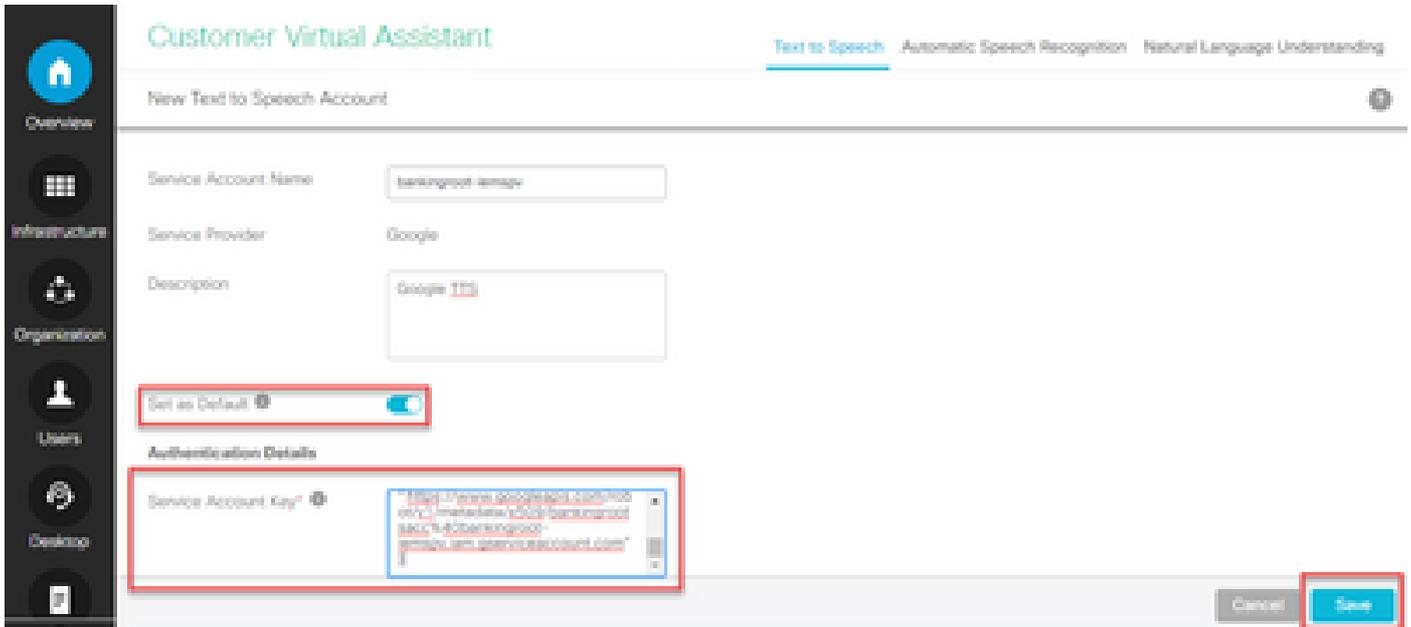
Step 10. Add first the TTS service, and then follow the same procedure to add the ASR service. Click on Text to Speech and then click on New.



Step 11. Select Google as the service provider and add the Service Account Name (same account name as the NLU account in previous steps). Click Next.

Step 12. Set this TTS service as the default, and copy the content of the NLU json file generated in previous steps as the ASR and TTS json key. Click on Save.

 **Note:** TTS and ASR service accounts do not require any role assigned. However, if you use the same NLU service account for ASR and TTS, you need to ensure that this service account has access to TTS and ASR APIs.



In general, this is the process flow when you use DialogflowIntent and DialogflowParam:

1. Call Studio / VXML application reads the JSON file from C:\Cisco\CVP\Conf\
2. DialogflowIntent audio prompt is played, either the audio file or the TTS in the audio setting is converted to audio.
3. Now, when the customer talks, the audio is streamed to the recognition engine Google ASR.
4. Google ASR converts the speech to text.
5. The text is sent to the Dialogflow from the VXML server.
6. Google Dialogflow returns the intent in form of text to the VXML application DialogflowIntent element.

Proxy Server Configuraion

Google Software Development Kit (SDK) in Cisco VVB uses gRPC protocol in order to interact with Google Dialogflow. gRPC uses HTTP/2 for transport.

As the underlying protocol is HTTP, you need to configure HTTP Proxy for end-to-end communication establishment if there is no direct communication between Cisco VVB and Google Dialogflow.

The proxy server should support HTTP 2.0 version. Cisco VVB exposes CLI command to configure Proxy Host and Port configuration.

Step 1. Configure httpsProxy Host.

```
set speechserver httpsProxy host <hostname>
```

Step 2. Configure httpsProxy Port.

```
set speechserver httpsProxy port <portNumber>
```

Step 3. Validate the configuration with the Show httpsProxy command.

```
show speechserver httpsProxy host
show speechserver httpsProxy port
```

Step 4. Restart Cisco Speech Server service after proxy configuration.

utils service restart Cisco Speech Server

In addition, if you have implemented the **Premise Based Intent Processing (DialogflowIntent / DialogflowParam)** deployment model, and If there is not a direct Internet connection between the CVP VXML Server and Google Dialogflow, you need to configure the proxy server on the CVP VXML Server.

Step 1. Log in to the CVP VXML Server.

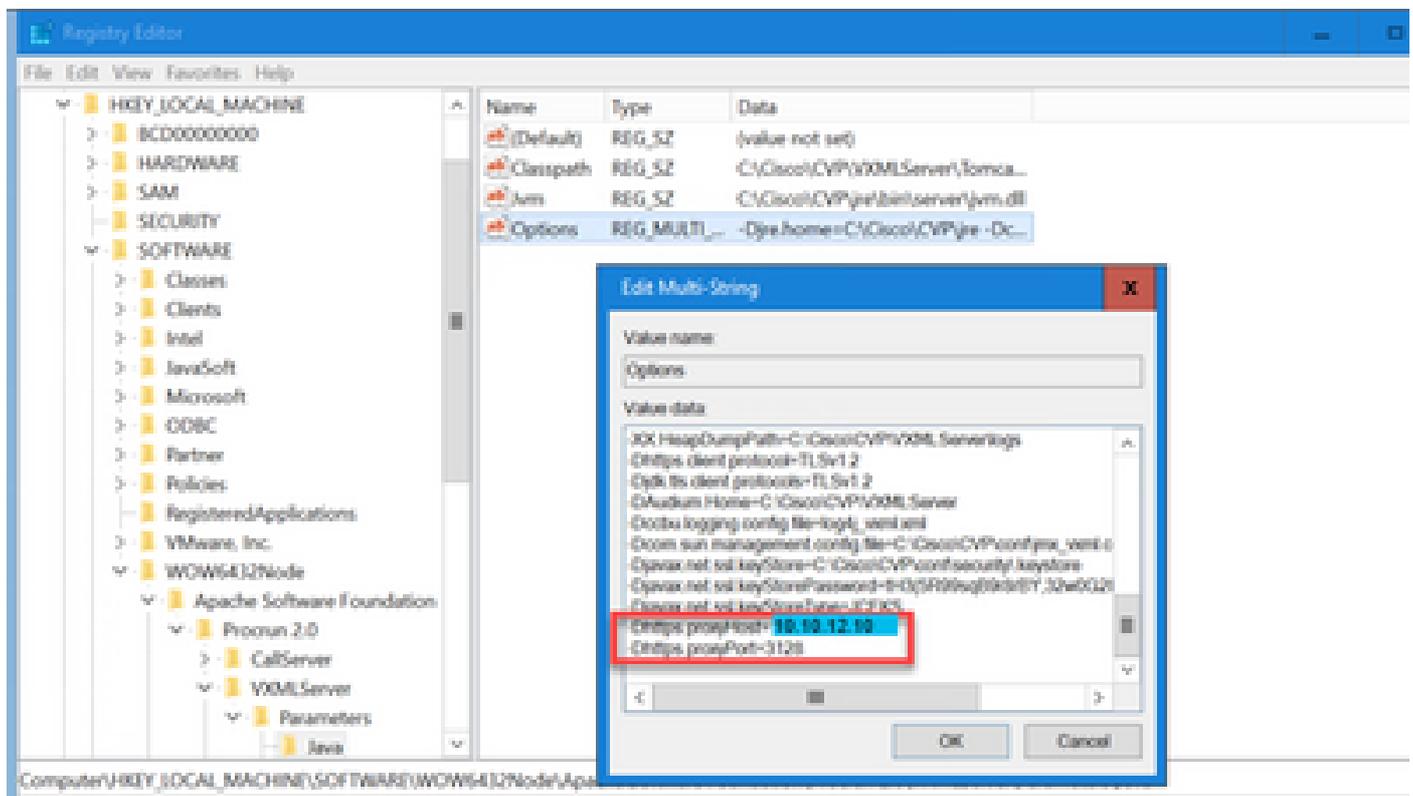
Step 2. Run the regedit command.

Step 3. Navigate to **HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Apache Software Foundation\Procrun 2.0\VXMLServer\Parameters\Java\Options.**

Step 4. Append these lines to the file.

-Dhttps.proxyHost=<Your proxy IP/Host>

-Dhttps.proxyPort=<Your proxy port number>



Step 5. Restart service Cisco CVP VXML Server.

Troubleshoot

If you need to troubleshoot CVA issues, please review the information in this document [Troubleshoot Cisco Customer Virtual Assistant](#) .

Related Information

Cisco Documentation

- [Sample Code](#)
- [CVA Design](#)
- [Configure CVA Services in UCCE with OAMP](#)
- [Configure CVA Services in PCCE](#)
- [Dialogflow Call Studio Element](#)
- [DialogflowIntent Call Studio Element](#)
- [DialogflowParam Call Studio Element](#)
- [Transcribe Call studio Element](#)

Google Documentation

- [Enable Dialogflow API](#)
- [Cloud Speech-to-Text API](#)
- [Cloud Text-to-Speech API](#)
- [Dialogflow Billing](#)
- [Google Enterprise Edition](#)
- [Enhanced Models](#)
- [Dialogflow Authentication Key](#)
- [Speech-to-Text Key](#)
- [Text-to-Speech Key](#)
- [Dialogflow Basics](#)
- [Setting up Dialogflow Agent](#)
- [Creating a Dialogflow Agent](#)

[Technical Support & Documentation - Cisco Systems](#)