

ShopBot

for earphones and headphones

User Guide



Team Members

Shashank Nigam	A0198469A
Lin Wenqi	A0198435R
Ng Mei Ying	A0198546L
Edmund Leow Kwong Wei	A0198458H

Contents

Getting Started	1
Full installation and deployment	1
System Requirements and Dependencies	1
Pre-requisites	1
Importing ShopBot into DialogFlow	1
Connecting Kommunicate with DialogFlow	2
Starting Rasa NLU server	3
Starting Python Flask for Webhooks	3
Launching the Website	4
Troubleshooting	5
Test Scenarios	6
Welcome	6
Get Product Recommendations	6
Get Pricing and Buying information	7
Get answers to headphone-related stuff (FAQ)	8

Getting Started

Full installation and deployment

Follow this section if installing and deploying to your own DialogFlow account and local web server.

System Requirements and Dependencies

- Python or Anaconda, and Python libraries as specified in requirements.txt
- A modern web browser. Recommended Google Chrome version 76 and above.

Pre-requisites

- Google DialogFlow account. Sign up at <https://console.dialogflow.com/>
- Kommunicate account. Sign up at <https://www.kommunicate.io/>
- Download ngrok.exe or ngrok binary to your machine from <https://ngrok.com/download>
- Clone or download project source code from GitHub repository from <https://github.com/eleow/shopBot>, navigate to the root folder of the repo, and install python project dependencies

- In a new Anaconda environment (eg shopbot) with **Python 3.6**,

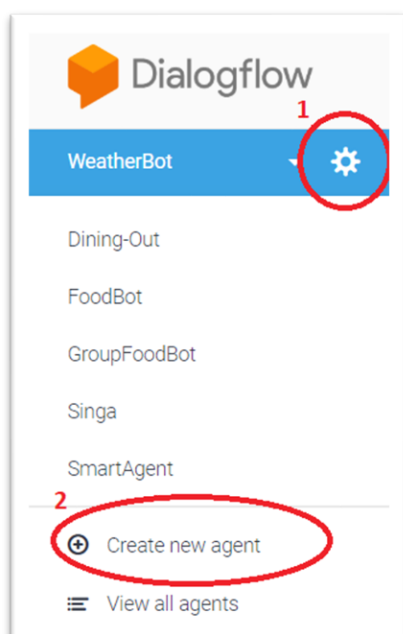
```
pip install -r requirements.txt
```

- Alternatively, use the script below. This will automatically create a new environment shopbot and install all dependencies (Note: Tested only on Windows 10 version 1909)

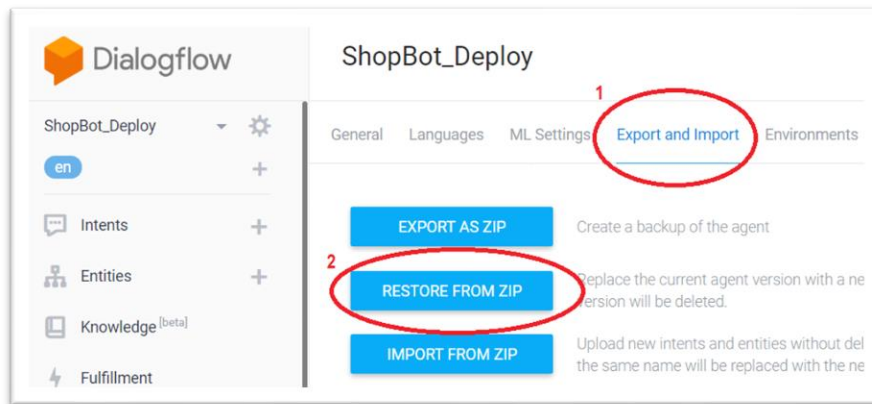
```
conda env create -f environment.yml
```

Importing ShopBot into DialogFlow

- Login to DialogFlow console at <https://console.dialogflow.com/>
- Create a new agent called "ShopBot" or whatever name you desire by
 - Clicking the cog button
 - Clicking the "Create new agent" button

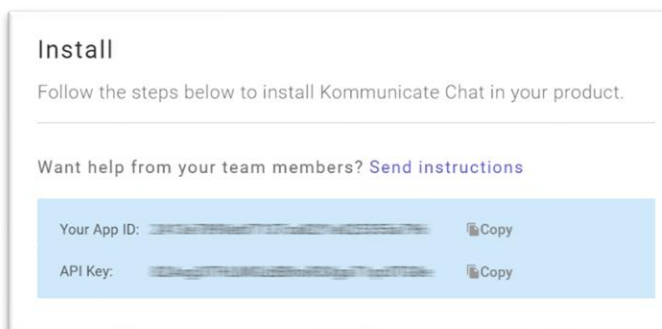


- Import the zip file (ShopBot_Deploy.zip) from the local copy of the GitHub repository at \SystemCode\DialogFlow by clicking the “Export and Import” button on DialogFlow, and then clicking “**RESTORE FROM ZIP**”

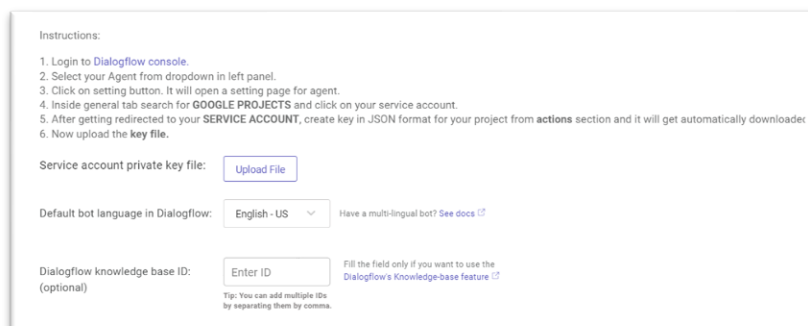


Connecting Kommunicate with DialogFlow

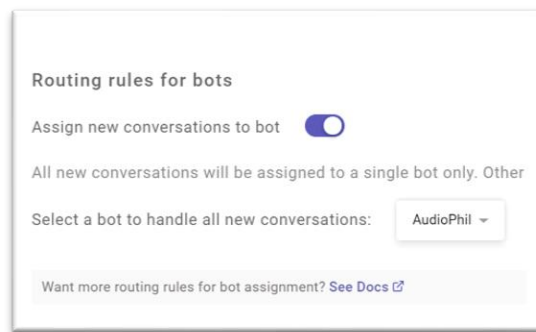
- Login to [Kommunicate](#)
- In [Install Page](#), note down your **App ID**. This will be used later



- Integrate Bot with DialogFlow in [Bot Integration Page](#), by following the instructions below



- **Important:** Go to [Conversation Rules page](#) (Settings → Conversation Rules → Routing rules for bots), and enable “Assign new conversations to bot. Select the bot that you have created.”



Starting Rasa NLU server

- Navigate to the root of your local copy of the GitHub repository eg D:\github\shopBot
- Run Rasa NLU server by the following script in console:
`rasa run --enable-api -m SystemCode/rasa/models/nlu_rasa.tar.gz --cors "*" -p 5015`
- It might take some time to start up. But finally, you should see the following in the console

```
(eba5004) D:\Users\Edmund\Documents\GitHub\shopbot>rasa run --enable-api -m SystemCode/rasa/m
odels/nlu_rasa.tar.gz --cors "*" -p 5015
No chat connector configured, falling back to the REST input channel. To connect your bot to
another channel, read the docs here: https://rasa.com/docs/rasa/user-guide/messaging-and-voic
e-channels
2020-03-22 16:15:28 INFO      root - Starting Rasa server on http://localhost:5015
2020-03-22 16:15:32 INFO      absl - Entry Point [tensor2tensor.envs.tic_tac_toe_env:TicTacTo
eEnv] registered with id [T2TEnv-TicTacToeEnv-v0]
```

Starting Python Flask for Webhooks

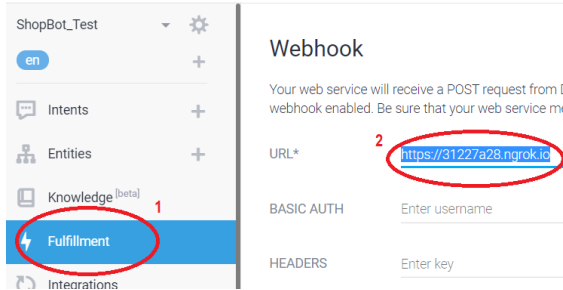
- Navigate to the root of your local copy of the GitHub repository eg D:\github\shopBot
- Run shopbot_main.py by the following script in console, using the **Kommunicate AppID** from earlier instead of <AppID>:
`python ./SystemCode/Fulfillment/shopbot_main.py -s False -n True -k <AppID>`

- It might take some time to start up. But finally, you should see the following in the console

```
(eba5004) D:\Users\Edmund\Documents\GitHub\shopbot>python ./SystemCode/Fulfillment/shopbot_main.py -s False -n True -k 1234567
-----
Flask is executed locally and ngrok will be started automatically
Remember to update DialogFlow fulfillment webhook with the PUBLIC_URL
Loading rasa library...
RASA will be used for Intent classification and entity detection
RASA server should be started manually and point to localhost:5015
In the root directory, use command: make rasa_run
-----
* PUBLIC URL: https://31227a28.ngrok.io
* RASA URL: http://localhost:5015


Starting app on port 5000
* Serving Flask app "shopbot_main" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```

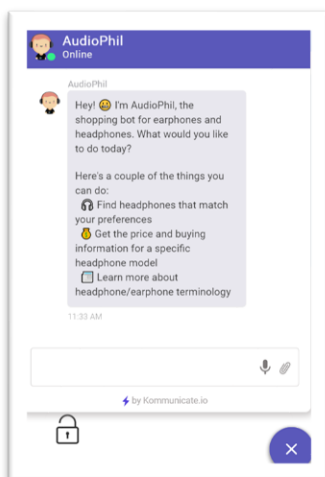
- Backend server is now running, and **ngrok has been automatically started as well**. Note down the public url for ngrok (Public url will change everytime you restart the backend)
- Now, we need to update DialogFlow Webhook URL with this. In DialogFlow, navigate to **Fulfillment** and configure the webhook to point to the URL that was captured earlier, and then **remember to click Save**.



- The ShopBot is now running on DialogFlow and is connected to the local Python Flask web server through ngrok, and Kommunicate is connected to DialogFlow.

Launching the Website

- Ensure that (i) ngrok, (ii) rasa NLU server, (iii) flask webhook are running from the steps above.
- Finally, you can launch the website using PUBLIC_URL or using <http://localhost:5000>
- Click on the AudioPhil icon  at the bottom right of the page (or any of the other items circled in red below), and start chatting!
- Note that the rest of the website will NOT work as this is just a proof-of-concept of how the chatbot could be integrated into an actual e-commerce website selling headphones.



Troubleshooting

[WinError 10061] No connection could be made because the target machine actively refused it

```
[2020-03-27 09:56:20,574] ERROR in app: Exception on / [POST]
Traceback (most recent call last):
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\connection.py", line 157, in _new_conn
    (self.dns_host, self.port), self.timeout, **extra_kw
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\util\connection.py", line 84, in create_connection
    raise err
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\util\connection.py", line 74, in create_connection
    sock.connect(sa)
ConnectionRefusedError: [WinError 10061] No connection could be made because the target machine actively refused it

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\connectionpool.py", line 672, in urlopen
    chunked=chunked,
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\connectionpool.py", line 387, in _make_request
    conn.request(method, url, **httplib_request_kw)
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\http\client.py", line 1254, in request
    self.send_request(method, url, body, headers, encode_chunked)
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\http\client.py", line 1300, in _send_request
    self.endheaders(body, encode_chunked=encode_chunked)
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\http\client.py", line 1249, in endheaders
    self.send_output(message_body, encode_chunked=encode_chunked)
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\http\client.py", line 1036, in _send_output
    self.send(msg)
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\http\client.py", line 974, in send
    self.connect()
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\connection.py", line 184, in connect
    conn = self._new_conn()
  File "D:\Users\Edmund\Anaconda3\envs\eba5004\lib\site-packages\urllib3\connection.py", line 169, in _new_conn
    self, "Failed to establish a new connection: %s" % e
urllib3.exceptions.NewConnectionError: <urllib3.connection.HTTPConnection object at 0x0000023624E991D0>: Failed to establish
ror 10061] No connection could be made because the target machine actively refused it
```

Looks like you might have forgotten to run Rasa NLU server! This error can occur if Rasa NLU server is not running or is running on a different port from what is configured. Refer to Starting Rasa NLU server

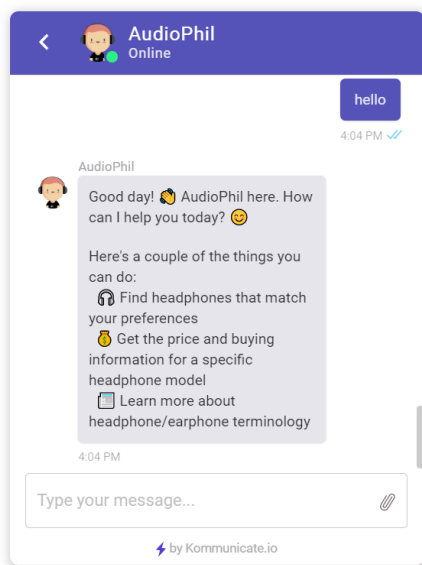
Test Scenarios

The following intents are available

- Welcome
- Get product recommendations
- Get pricing and buying information for a specific headphone model
- Get answers to headphone-related stuff (FAQ)

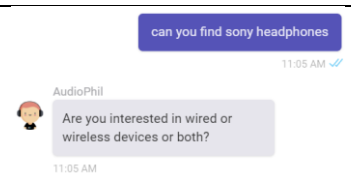
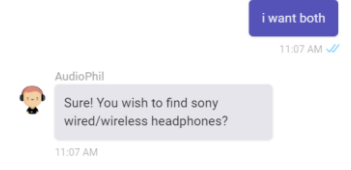
Welcome

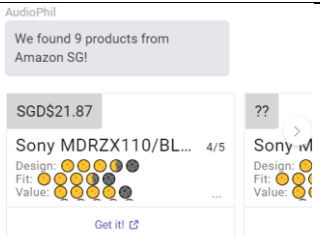
On launching AudioPhil, the chatbot will automatically welcome you.



Get Product Recommendations

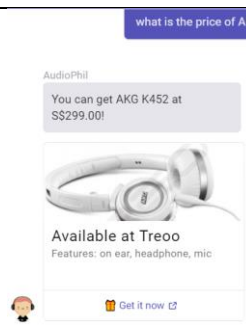
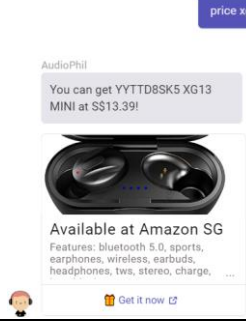
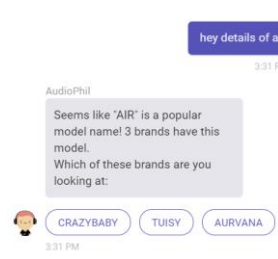
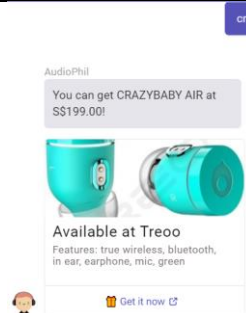
If the user asks for product recommendation, we will return products based on his preferences.

Sample User Query	Sample Response	Entities	Description
"can you find sony headphones?"		product_type=["headphones"] ent_brand=["sony"]	Product type and brand was entered by user and chatbot asked for connectivity type
"I want both"		wired="both"	Connectivity type was entered by user and chatbot asked for confirmation of selection

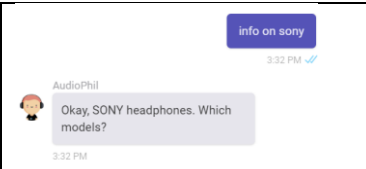
"that's right"			Results based on user's preferences were returned, as a carousel
----------------	---	--	--

Get Pricing and Buying information

If the user provides a specific headphone model, we will provide the product information (price, features and link to buy it), if it is found.

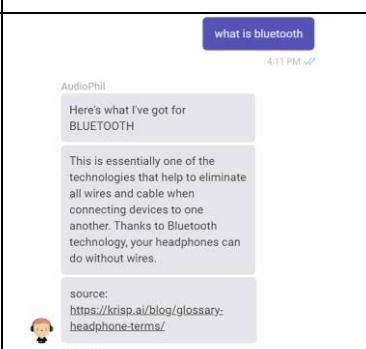
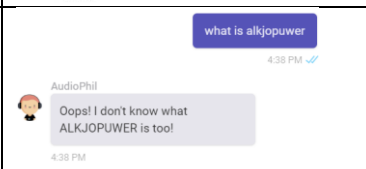
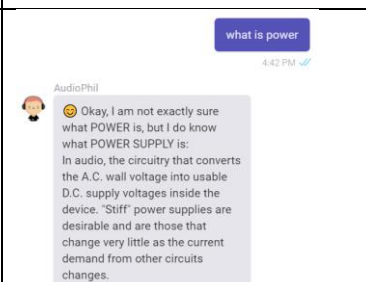
Sample User Query	Sample Response	Entities	Description
"what is the price of AKG 452?"		brand= "AKG" model= "452"	Product brand and model were provided by the user, and result was returned. Product was either found only in Treoo or was cheaper in Treoo, therefore, the product price and link were given from Treoo.
"price xg13 mini"		brand="" model= "xg13"	Same as above, but this time, only product model was provided. And product was found to be cheaper from Amazon SG
"hey details of air"		brand="" model= "air"	In this case, there were multiple brands that had this particular model. Therefore, we asked the user for the brand. Since the possible brands are not many (<5), we showed suggestion chips.
		brand= "crazybaby" model= "air"	Then the user selected "crazybaby", and results were returned

Master of Technology (Intelligent Systems)

"info on sony"	 <p>AudioPhil 3:32 PM</p> <p>info on sony</p> <p>3:32 PM</p> <p>Okay, SONY headphones. Which models?</p>		In this case, user only entered a brand. Since possible models are too many (>5), we did not show any suggestion chips
----------------	---	--	--

Get answers to headphone-related stuff (FAQ)

You can ask the chatbot specific terms related to headphones. We will provide an explanation of the term, as well as the source of the information.

Sample User Query	Sample Response	Entities	Description
"what is Bluetooth"	 <p>AudioPhil 4:11 PM</p> <p>what is bluetooth</p> <p>4:11 PM</p> <p>Here's what I've got for BLUETOOTH</p> <p>This is essentially one of the technologies that help to eliminate all wires and cable when connecting devices to one another. Thanks to Bluetooth technology, your headphones can do without wires.</p> <p>source: https://krisp.ai/blog/glossary-headphone-terms/</p>	query="bluetooth"	Search term was found in our list of glossary. Description and source of information is provided to the user
"what is alkjopower"	 <p>AudioPhil 4:38 PM</p> <p>what is alkjopower</p> <p>4:38 PM</p> <p>Oops! I don't know what ALKJOPUWER is too!</p>	query="alkjopower"	In this case, search term was not found in our list. Therefore, we tell the user that we do not know what it is.
"what is power"	 <p>AudioPhil 4:42 PM</p> <p>what is power</p> <p>4:42 PM</p> <p>😊 Okay, I am not exactly sure what POWER is, but I do know what POWER SUPPLY is: In audio, the circuitry that converts the A.C. wall voltage into usable D.C. supply voltages inside the device. "Stiff" power supplies are desirable and are those that change very little as the current demand from other circuits changes.</p>	query="power"	In this case, while an exact match was not found, we managed to match it to the closest term "POWER SUPPLY", by comparing word vector similarity.