**Chatbot using Google Dialogflow**

**Submitted by:**

**Nitesh Srivastava**

**Small demo URL: https://youtu.be/WevwphMIfzk**

**Presentation URL:** [**https://youtu.be/daeWaXsgZ14**](https://youtu.be/daeWaXsgZ14)

**Table of content**

1. [Objective](#_Objective:)
2. [Safe harbor statement](#_Safe_harbor_statement:)
3. [Functionality of chatbot](#Functionalityofchatbot)
4. [Tools and Technologies used](#_4._Tools_and)
5. [About Google Dialogflow](#_5._About_Google)
6. [Set up for COVID-19 chatbot](#_6._Set_up)
7. [Python code for the Bot](#_Python_code_for)
8. [Integration with Telegram](#_Integration_with_Telegram:)
9. [Integration with Facebook Messenger](#_Integration_with_Facebook)
10. [Integration with Slack](#_10._Integration_with)
11. [Saving the record in MongoDB](#_Saving_transactions_in)
12. [Conclusion](#_Conclusion:)
13. [Reference](#_Reference:)

# Objective:

This chatbot is created as a part of chatbot challenge thrown by [ineuron](https://ineuron.ai/home/). The chatbot is developed with the help of Google Dialogflow. Objective of this document is to showcase steps, how to configure and integrate google dialogflow chatbot.

# Safe harbor statement:

All the tools and technologies used to create this, are free and publicly available to use. I have taken reference of images and codes available in public domain, so given their due credit in detailed documentation. This is made only for educational and learning purpose. There will be no production or commercial use of this.

# Functionalities of chatbot:

There are few specific requirements to be included as apart of challenge. The asked requirements are as below. It should respond user queries and:

a. Greet the users

b. Ask and capture name of user.

c. Ask and capture email of user.

d. Ask and capture phone number of users.

e. Show COVID-19 cases across world.

f. Show the COVID-19 cases across India.

g. Show COVID-19 cases for specific states in India.

h. Show Dos and Don’ts.

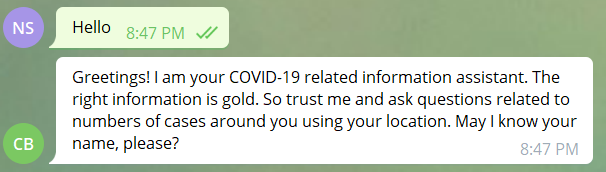
g. Email the information.

h. Save the transactions in MongoDB.

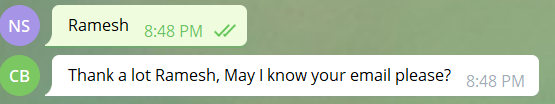
**Functionalities in Telegram (Desktop App)**

a. Greet the users:

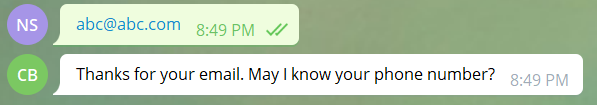
When users starts interacting with chatbot, it greets to users as below:



b. Ask and capture name of user



c. Ask and capture email of user

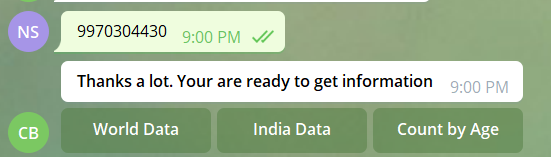


d. Ask and capture phone number of users

validation phone number



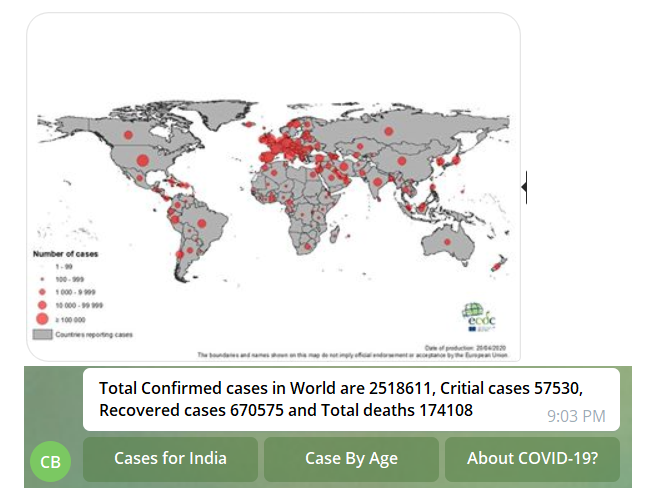
When phone number given is in correct format:



It validates for correct phone number and then gives options to choose. User can even write their input query if they don’t want to use given options.

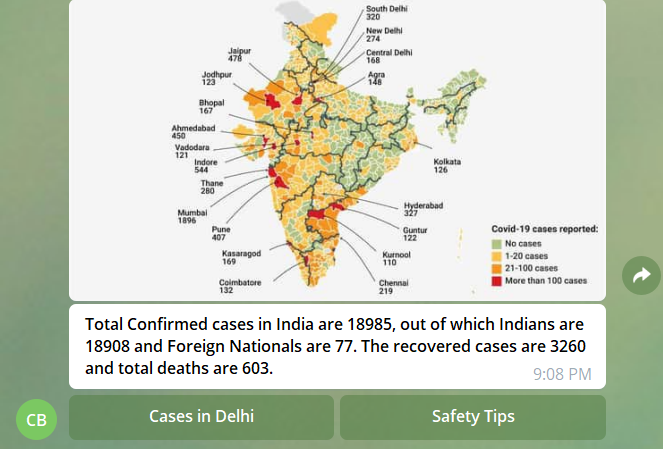
**1. Total Word Data:**

Click on which or type “World Data” will give distribution of COVID-19 cases across the world with the message in text also as below:



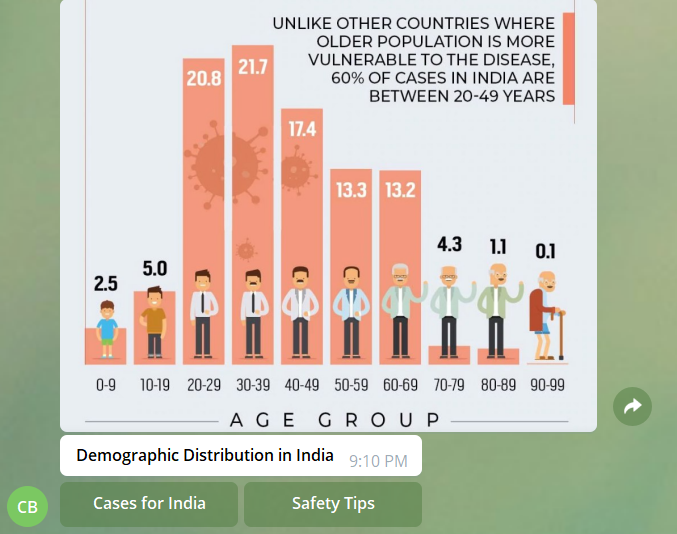
**2. India Data:**

When click on India Data or typed as “India Data” it will fetch the distribution of COVID-19 cases with image and text as below



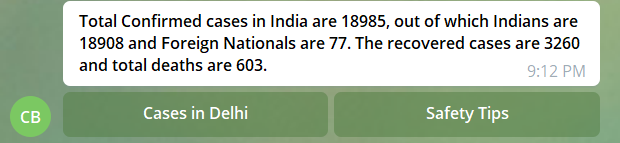
**3. Count by Age:**

When clicked on Count by Age or typed as “Count By Age” it will show the demographic image scraped from India Today as below:

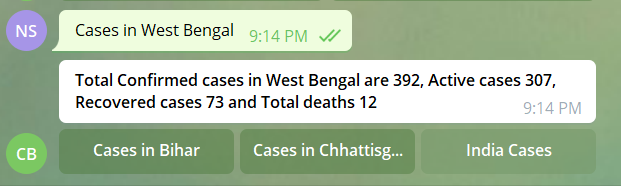


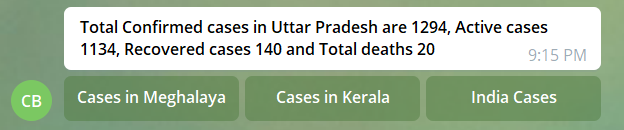
**4. Cases in Delhi:**

Click on the button or Type Cases in Delhi (Or any other state name), it will fetch the COVID-19 Cases for that particular state as below:



If users type any other state name or ISO code, he Total cases for that state will be shown as below:

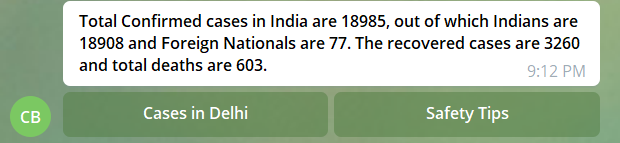




Below each state’s information, bot will give user another tow state’s option so that users can avoid writing wrong names of states. Wrong entry of data may lead to no information.

**5. Safety Tips:**

When users click of Total India data button or type so, with the result they will get a Safety Button as below:

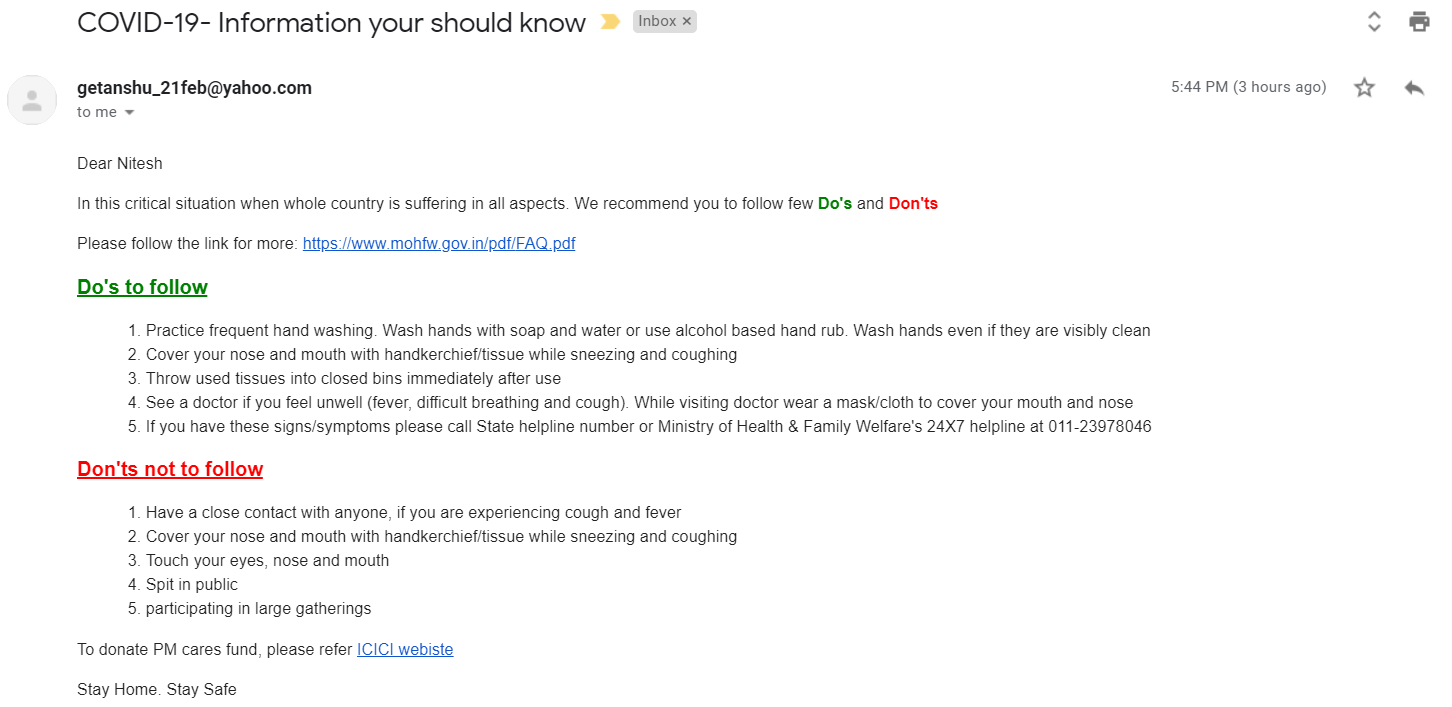


Click on which or type so, users will get the dos and Don’ts list as below:

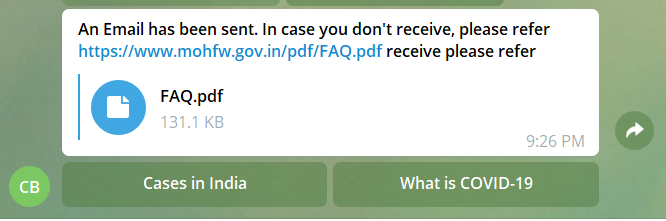


**6. Email to users:**

If the users have given correct email address in start of conversation, users will find button as above to email the information and tips to themselves. In case if the email has been mentioned wrong, users have option to request email on correct Email Address with query as “Email to <right email>”. The email will be sent to users in below format:



As result of the response of Email query the following will be shown to users:

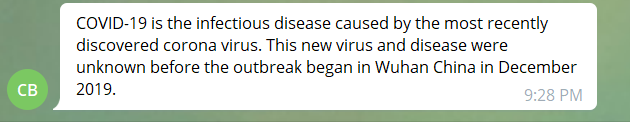


Link will be given users to follow in case they have not received the email due any reasons.

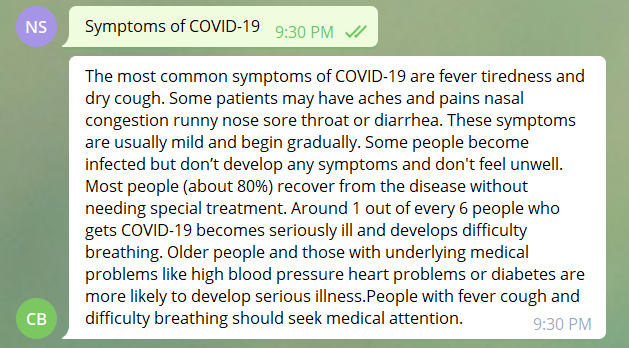
**7. FAQs:**

There are many FAQs has been configured as below:

When the user clicks on What is COVID-19 below response comes:

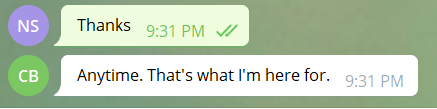


When user queried about symptoms of COVID-19, the answer came as below:

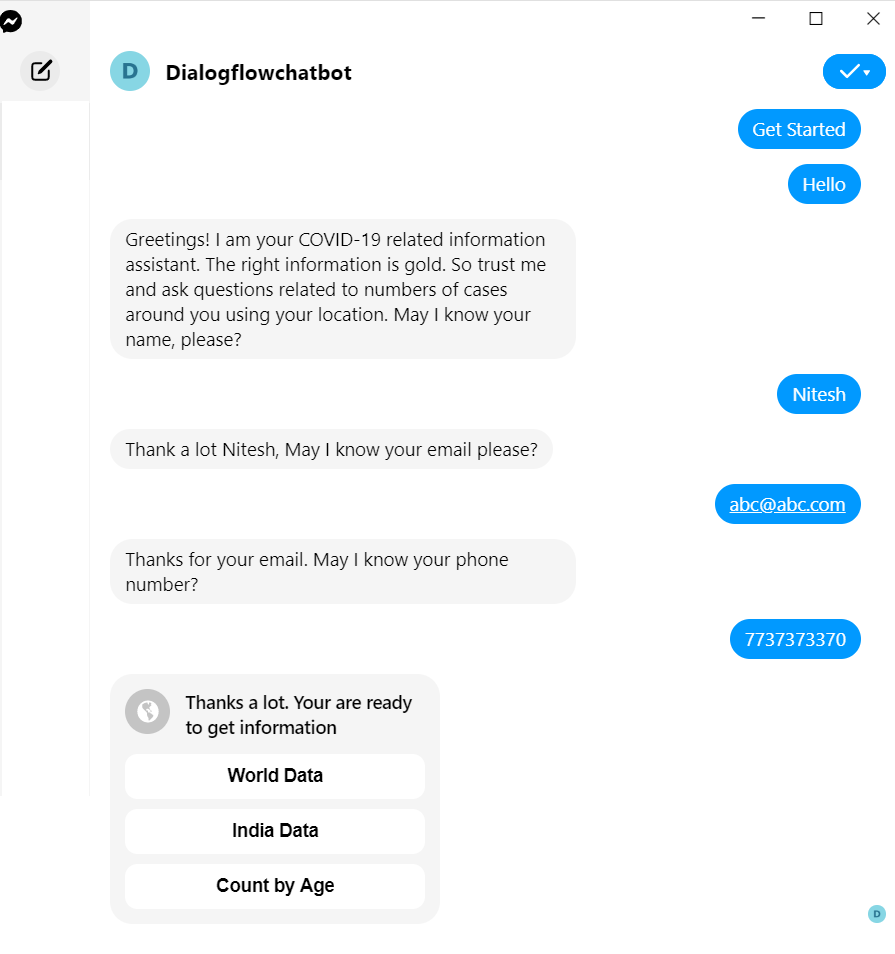


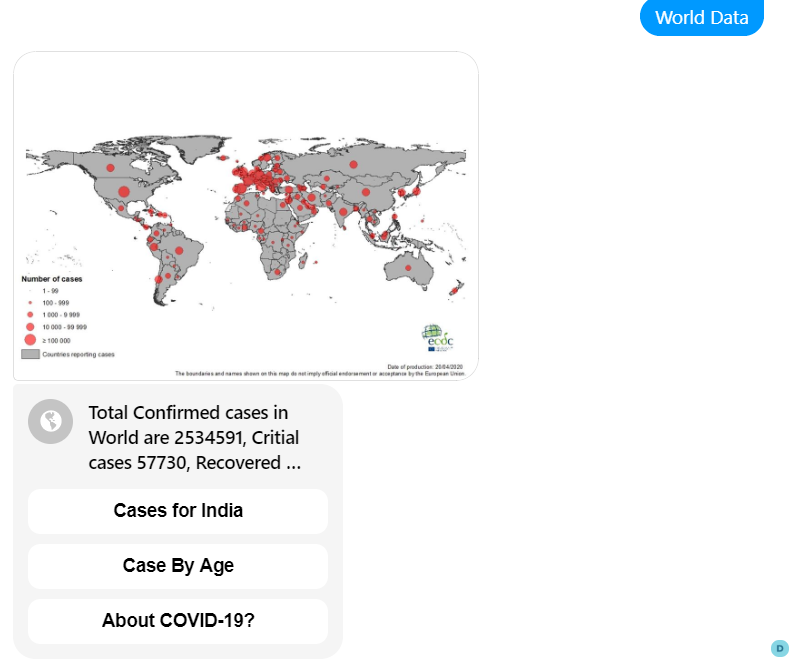
**8. Thank you note:**

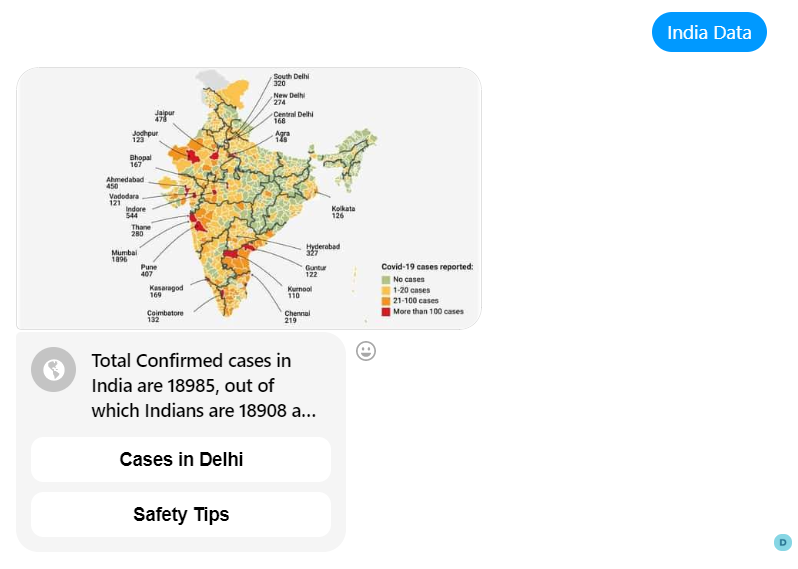
When user types Thanks or any other thanks text like this, response comes below:



**Functionalities for Facebook messenger Desktop App:**

****





**And the functionality is similar as Telegram chatbot.**

# 4. Tools and Technologies used:

1. Google dialogflow
2. Python and related dependencies
3. Yahoo mail as email server
4. Beautiful soup for web scrapping
5. Rapid APIs for Covid-19 related data
6. ngrok to expose webservices publicly
7. Images from India Today and MoHFW website
8. MongoDB

# 5. About Google Dialogflow:

Dialogflow is natural language understanding platform owned by Google. The company named API.ai was initial company to own this product and this was acquired by google in September 2016. Later in October 2017 it has been renamed as Google Dialogflow and November 2017, it became part of Google cloud platform. Data Sourced from [Wiki](https://en.wikipedia.org/wiki/Dialogflow). To start with chatbot developer must have access to <https://dialogflow.cloud.google.com/> and register.

# 6. Set up for COVID-19 chatbot:

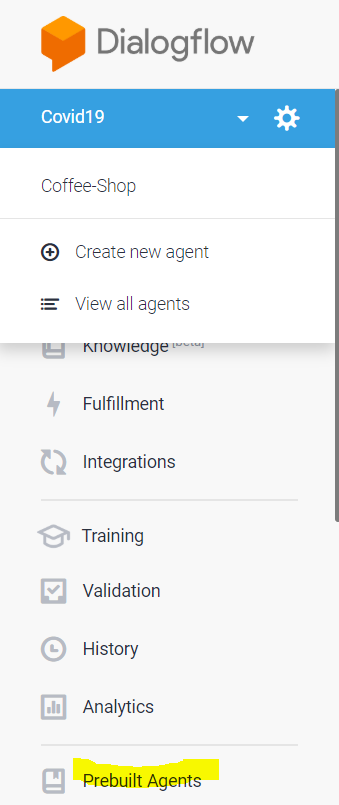
Google dialogflow is built with various components. These components and their configurations have been mentioned as below:

**1. Agents:**

Agents are the first ever thing that is created. Agents can be seen equivalent to the person sitting other side of users who is going to respond. On the left hand top corner of the window after sign in Agent section can be found as below:



You can see all the agents which are part of your google dialogflow account. You can create a new agent from here. If you want to refer any other pre built agent, you can find the prebuilt agent section downmost left hand section as below:



**2. Entity:**

After agent creation the second step is to create Entities. Entities are pre-defined or pre trend worlds and their synonyms which can be used by agent when user dictates any query. For example in this bot, the Indian states are configured in such way that if user enter right name, ISO code, or small spelling mistakes, it will match with entities and return the right data. Details about entities can be found [here](https://cloud.google.com/dialogflow/docs/entities-options).

Entities are of two types:

1. System Entities: There are system pre defined and pre trained. Like number, countries, given name, first name. The details about system entities can be found [here](https://cloud.google.com/dialogflow/docs/entities-system).

2. Custom Entities: These are the dialects which are defined by developer to match with user queries. For example if users gives a query like “Total cases in Delhi”, so we can create two entities one for Total which will match anything like “All”, ”Active”, ”Confirmed”, ”Deaths” like these types od case data available. Developer can pass any value and if it is matched in entities it will pick and send for data retrieval. More details about custom entities can be found [here](https://cloud.google.com/dialogflow/docs/entities-custom).

For this chat bot the below entities are configured:

|  |  |  |
| --- | --- | --- |
| Name | Usage | Example |
| case\_type\_entity | This entity is configured to match types of cases like Total, Active, Recovered and Death |  |
| covid\_name | To identify various user input like Corona, COVID, COVID19, COVID-19 |  |
| state\_name | To identify input for various states in India, India as word and World as word |  |

**3. Intents:**

Intents are the most important part of chatbot. It Captures what users has asked for. If the right intent is not matched with the utterance the result will not be given back or a default fallback result like “Sorry, Can you please repeat?” will be given. Details about intent can be found [here](https://cloud.google.com/dialogflow/docs/intents-overview). Intents as few parameters as below:

a. Intent Name: Name of the intent. This will useful to identify which intent has been matched based on user utterances.

b. Training Phrases: with the use of training phrases we train out bot to match the utterances. Like if I train bot with “Give me total details in Rajasthan”, if users put same phrase and subphrase of this, the intent will be matched. Remember the entities, if the entities are mapped then user can enter any state name, any kind of cases and intent will be matched. Developers need not to train the bot for each case type and states. More about [training phrases](https://cloud.google.com/dialogflow/docs/intents-training-phrases).

c. Contexts: Contexts are used to take data (input from users) from one intent to other intent. Two types of contexts are mentioned. Input Context, contexts which are coming from previous intents. Output contexts, Contexts which are to be sent to coming intents. More about [contexts](https://cloud.google.com/dialogflow/docs/contexts-overview).

d. Action and Parameters: This section is used to interact with different APIs. In google dialogflow two technologies can be used to implement custom process. One is Node.js and second is Webhooks, that calls the external webservices. The scope of this document covers webhooks and webservices developed in Python and Flask.

Parameters are set in parameters section which will passed to webservices using post method. These parameters will be used to get data from external system. For example if user queries for “Get me details for Delhi”, the state ad type of query (Total, confirmed or recovered) can be passed as parameters.



e. Response:

Response section is part of intent where developer will define what to show to end users. This supported response by google dialogflow are:

1. Text Response: Plain text with data fetched from external system.

2. Image Response: To show users and static image with its title.

3. Card Response: To show users a card which can contain a text, Image, and button for users to click.

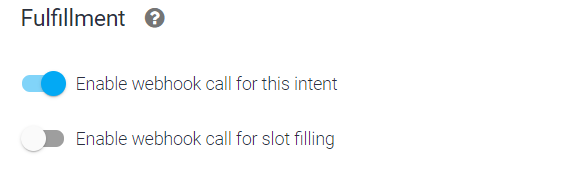
4. Quick Replies: To show users text and then suggestion to go forward.

5. Custom Payload: A custom defined text in Google Flow accepted can be set by APIs and user will see the text, images, cards, quick replies. Format of custom message can be as below:



**Fulfillment:**

Fulfillment is used to tell intent that if it is matched, the webservices need to be called with given action name and parameters.



List of Intents Created for this chatbot:

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Default Welcome Intent | To respond for hello, hi kind of text | Text Response |
| Save\_name | To get the name and save it and asking user for email | Text |
| save\_email | To get email,save it and ask for phone number | Text |
| save\_phone | To get phone number, validate, respond incase wrong phone number format, save and respond in case correct phone number.  Action: checkfoneformat  Parameters: phone-number | Custom Response |
| covid\_case\_intent | To get covid cases across world, India and states of India.  Action: getCovidCases  Parameters: case\_type\_entity  state\_name | Custom Response |
| Demographic\_distribution\_intent | To show demographic distribution of COVID cases  Action: showDemographicDistribution  Parameters: | Custom Response |
| DosDonts\_intent | To show Dos and Don’ts to the users  Action: getDosDonts | Custom Response |
| email\_info\_intent | To send emails to users  Action: sendEmailInfo | Custom Response |

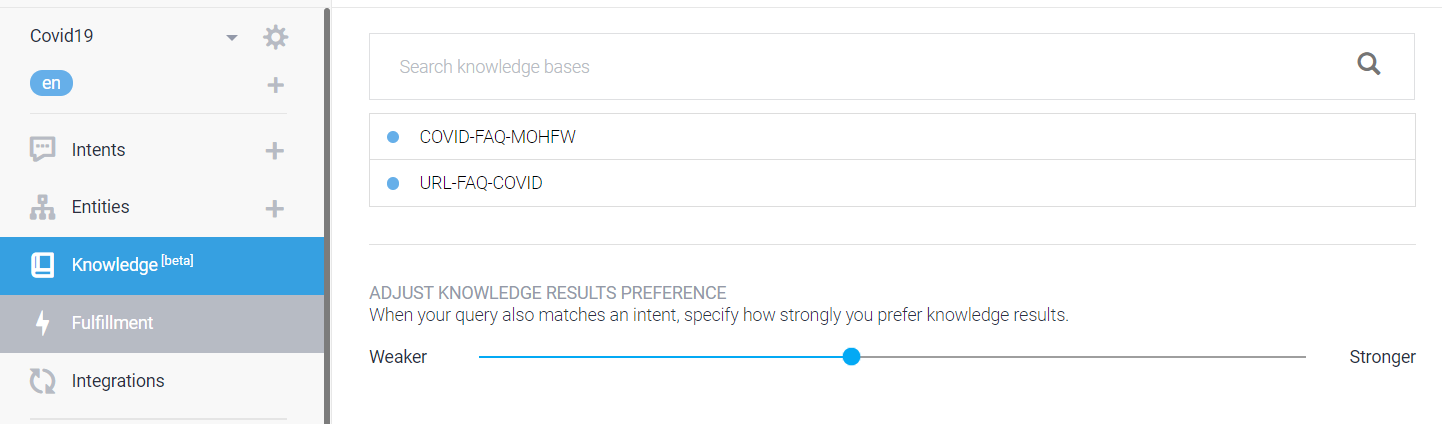
**4. FAQs:**

There are FAQs which has been configured for this chat bot. FAQs are configured in the format of csv files and URLs. Things to take care:

1. CSV must has two column only.

2. Data must not have any “,” in content else this will keep the next content in next column.

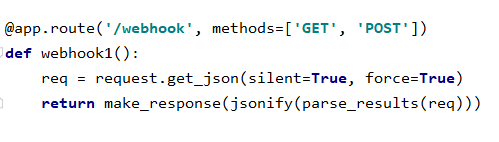
We can set the matching frequency of FAQs and Intents.



# Python code for the Bot:

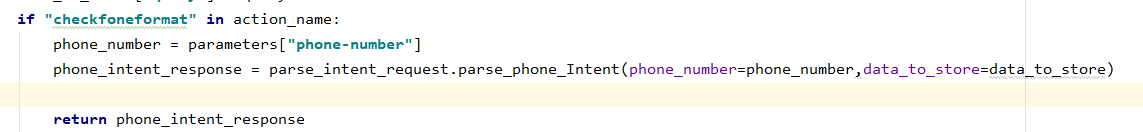
As it is mentioned earlier in document that this chatbot is developed using python assumes that Flask and it’s dependencies are already installed. Chatbot interacts with python code using webhooks i.e. webservices.

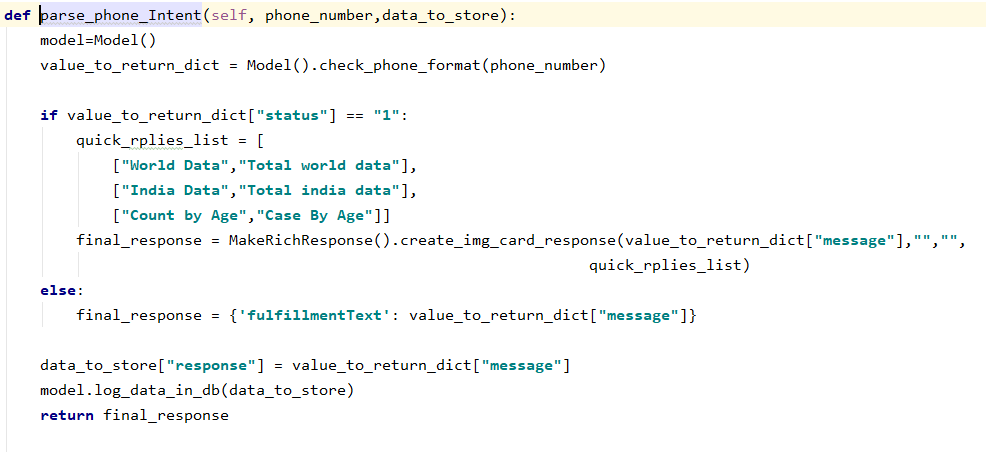
Webhook configured name: webhook



Action Names and Python code:

1. “checkfoneformat”: this webhook code is configured to confirm he phone number format validity. The input is phone number given by users and output will text to users.

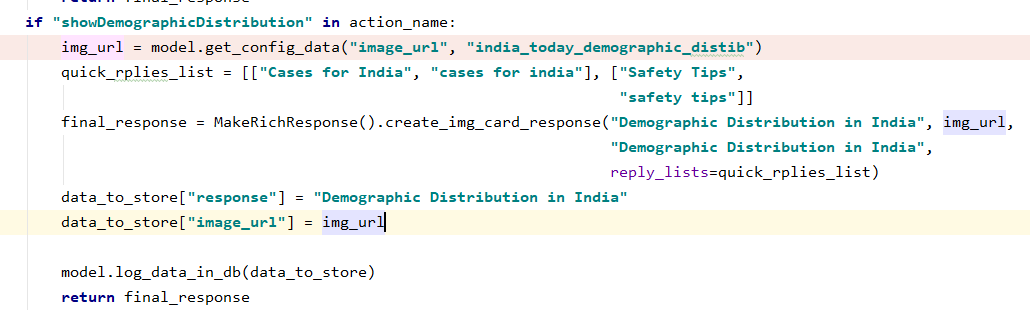




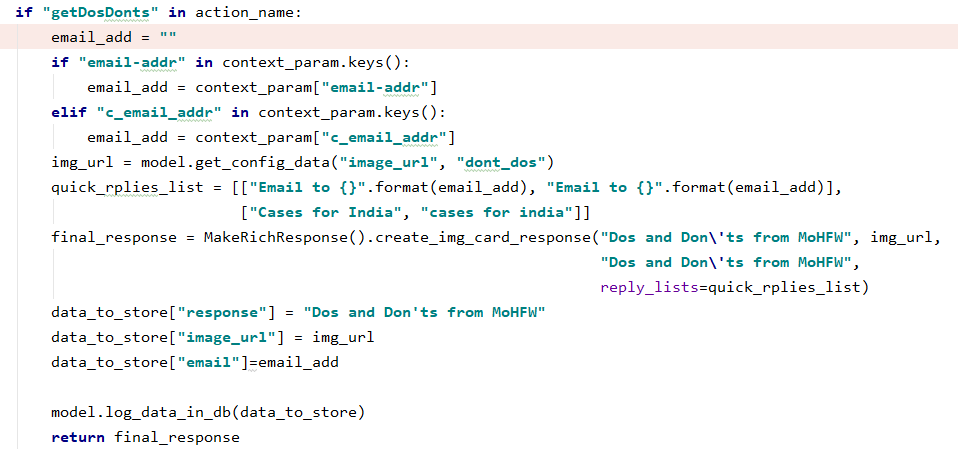
2. “getCovidCases”: Action configured to get COVID-19 related data from RAPID APIs and image sources. Input parameter will be state name (“World”, “India” or Indian State name) and case type (“Total”,”Active”,”Confirmed”,”Recovered”,”Deaths”). The return message will either text, image or mix or these in the forma of cards.



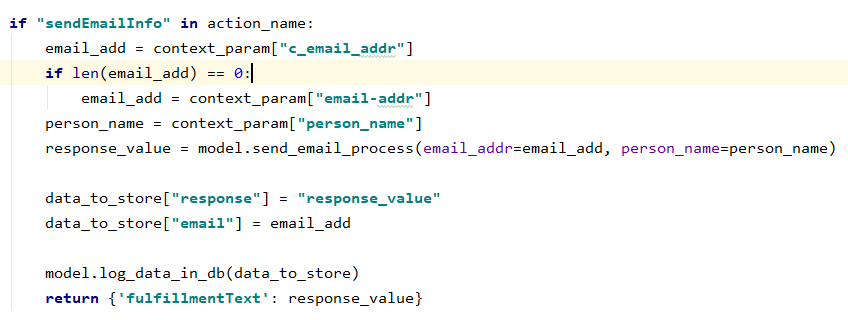
3. “showDemographicDistribution”: Action to get demographic distribution. There will be no input and output will image from India Today.



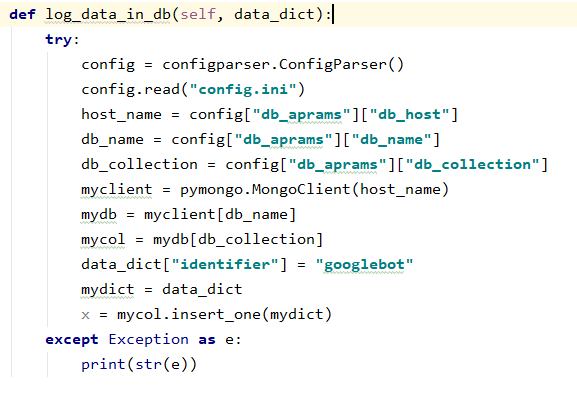
4. “getDosDonts”: Action to configure Safety tips and precautionary measures and show it to users when users are asking for this. No input and output will be a card response.



5. “sendEmailInfo”: Action to send email information to users when it is asked for. Input will be email address and name of the person given by user in earlier conversation. The out will be a text and link to PDF file.



6. “log\_data\_in\_db”: This function is configured to store the required transactions in MongoDB. The input will be in dictionary format for text to store.



7. “Cofig.ini”: the ini file to store the configuration parameters which are changed frequently. Below is the snapshot of this file:

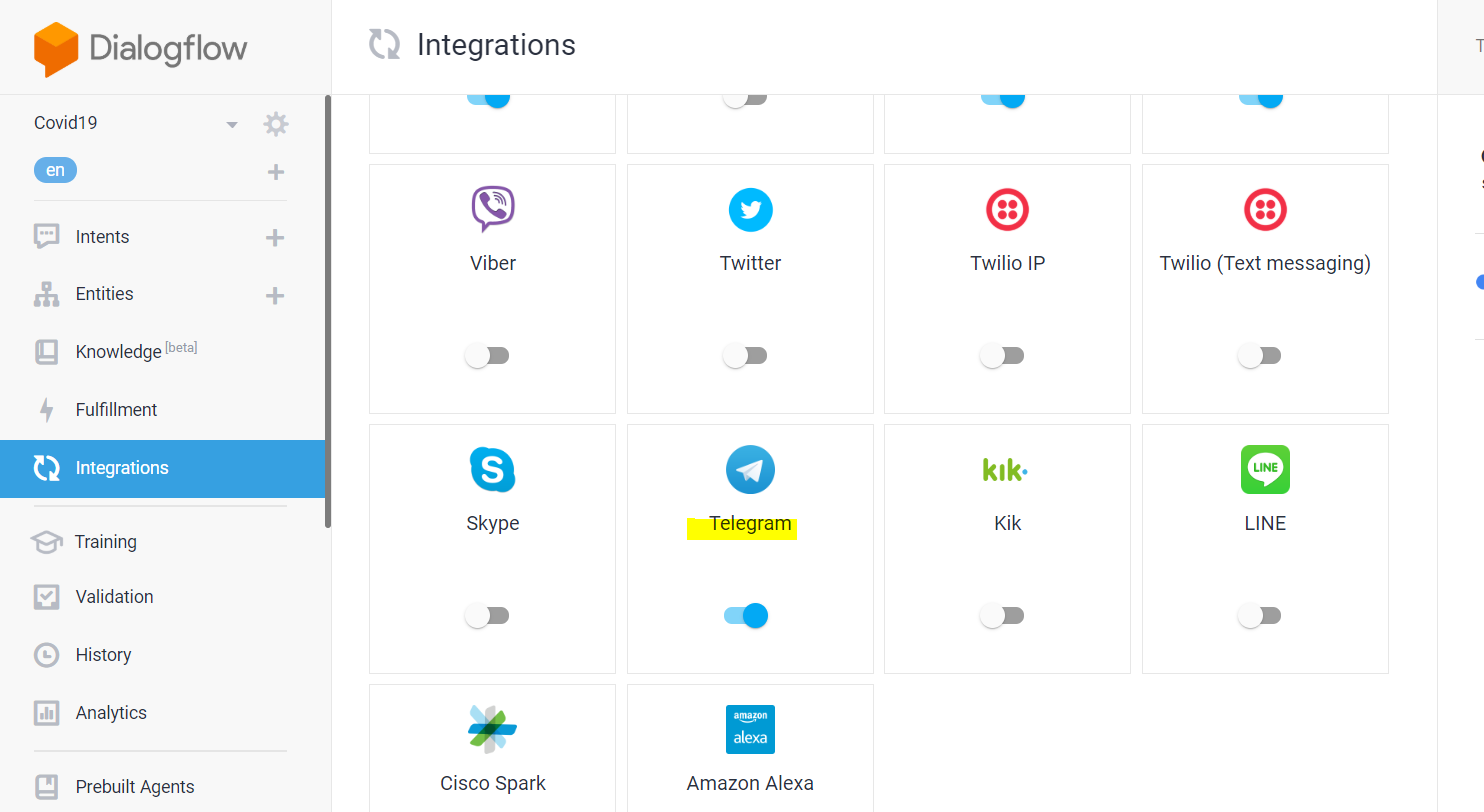


# Integration with Telegram:

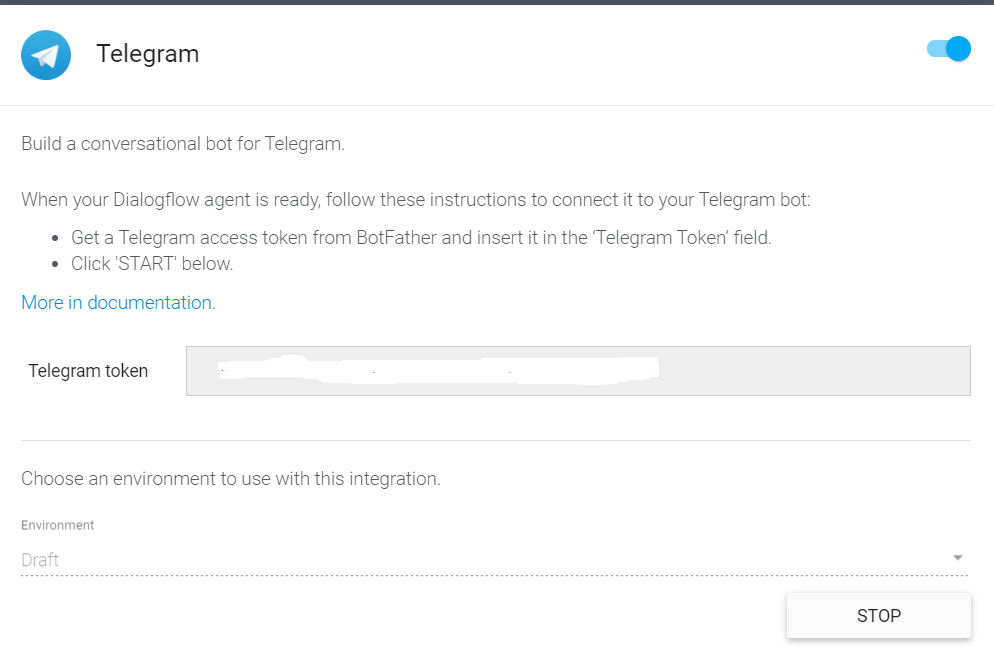
Once the code and dialogflow set up is done and tested using dialogflow, its time to integrate the same with Telegram App. The integration is set up in two different steps:

a. Configuration in Dialogflow:

1. Open the integration section as below and switch on the Telegram button:

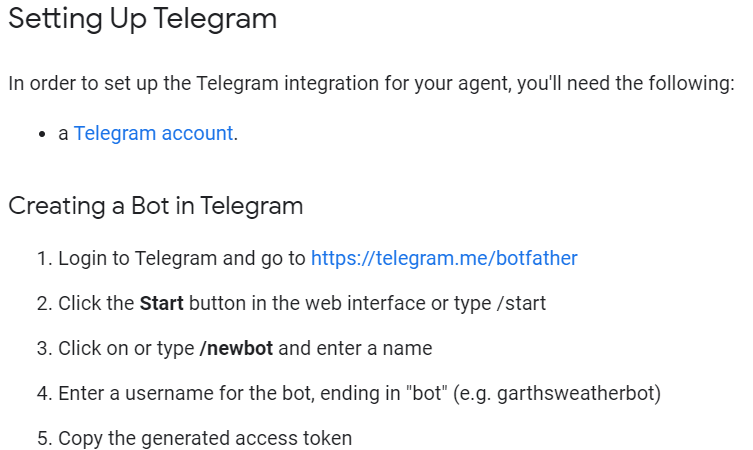


2. A page will be open to show below:



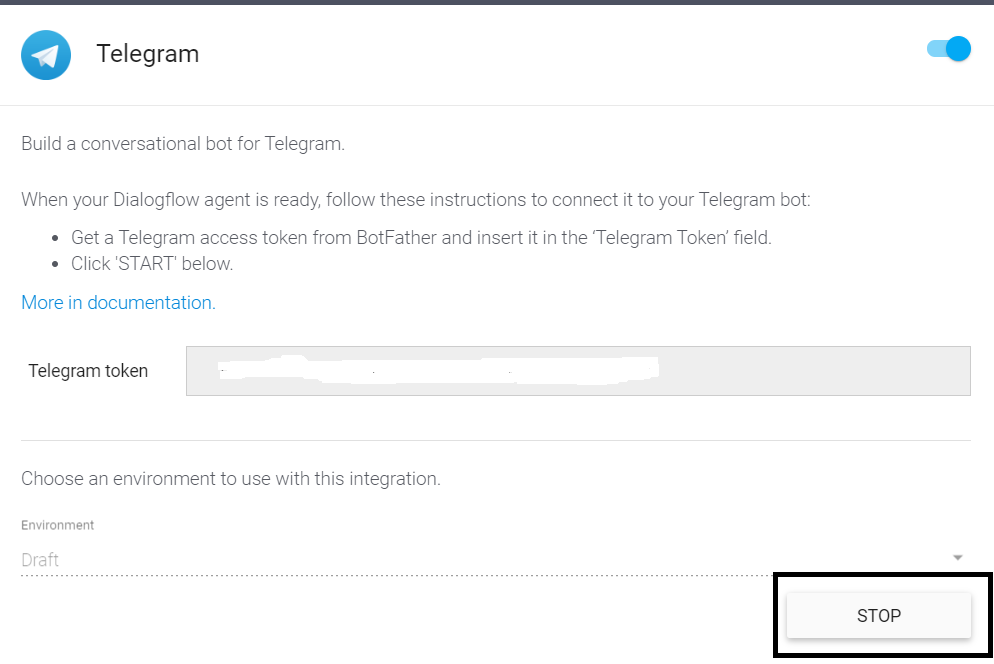
We need to get Telegram token from botFather in Telegram. Refer to More in [documentation](https://cloud.google.com/dialogflow/docs/integrations/telegram) to configure further.

2. Configure Telegram botFather to integrate:



The same access token will be copied and pasted in dialogflow window. Click on start once copied. You are good to go.

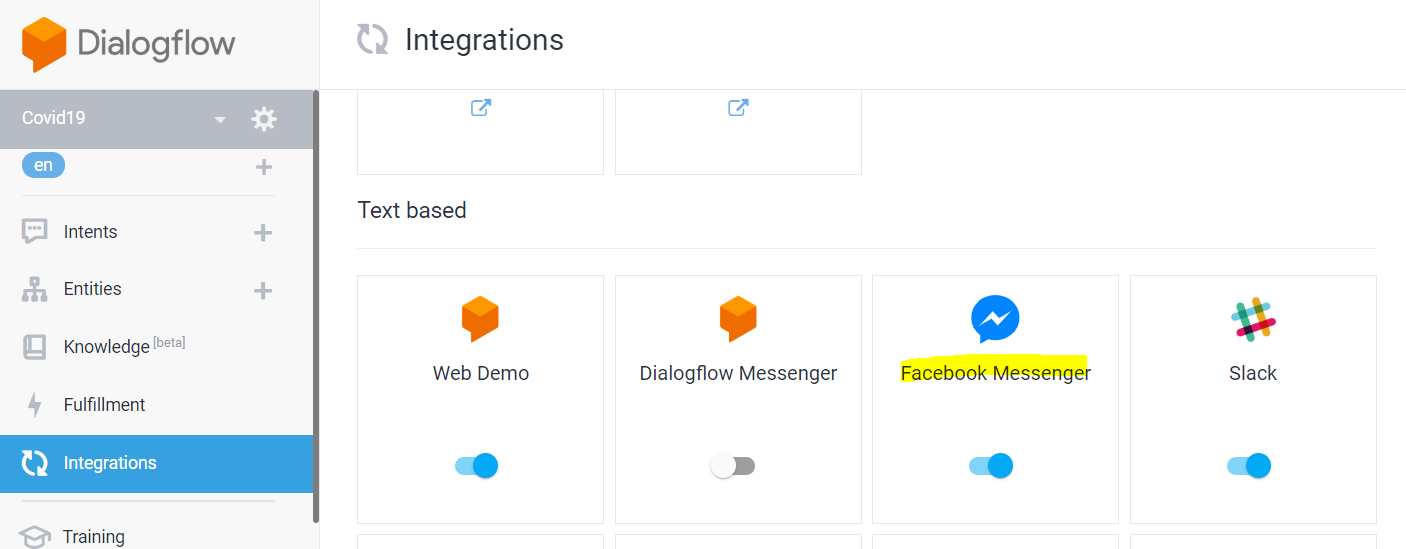
These services can be always start and stop from google dialogflow using the below highlighted button:



# Integration with Facebook messenger:

a. Dialogflow configuration:

1. Integration section click on FB messenger icon:



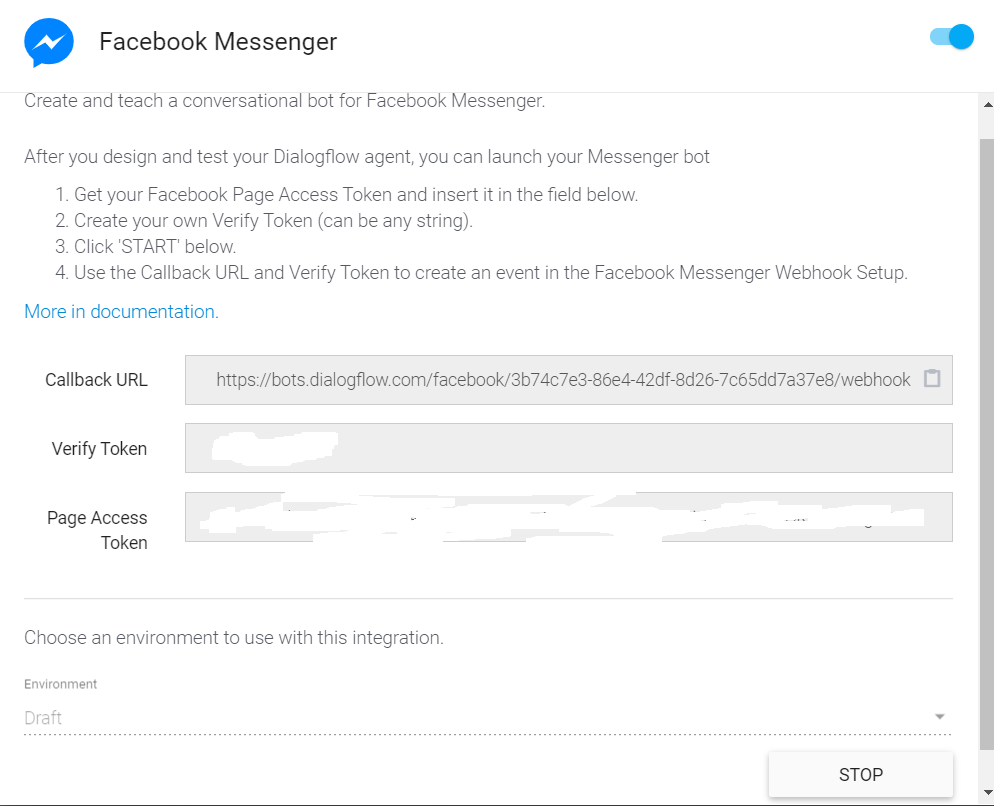
2. It will open a window like below:

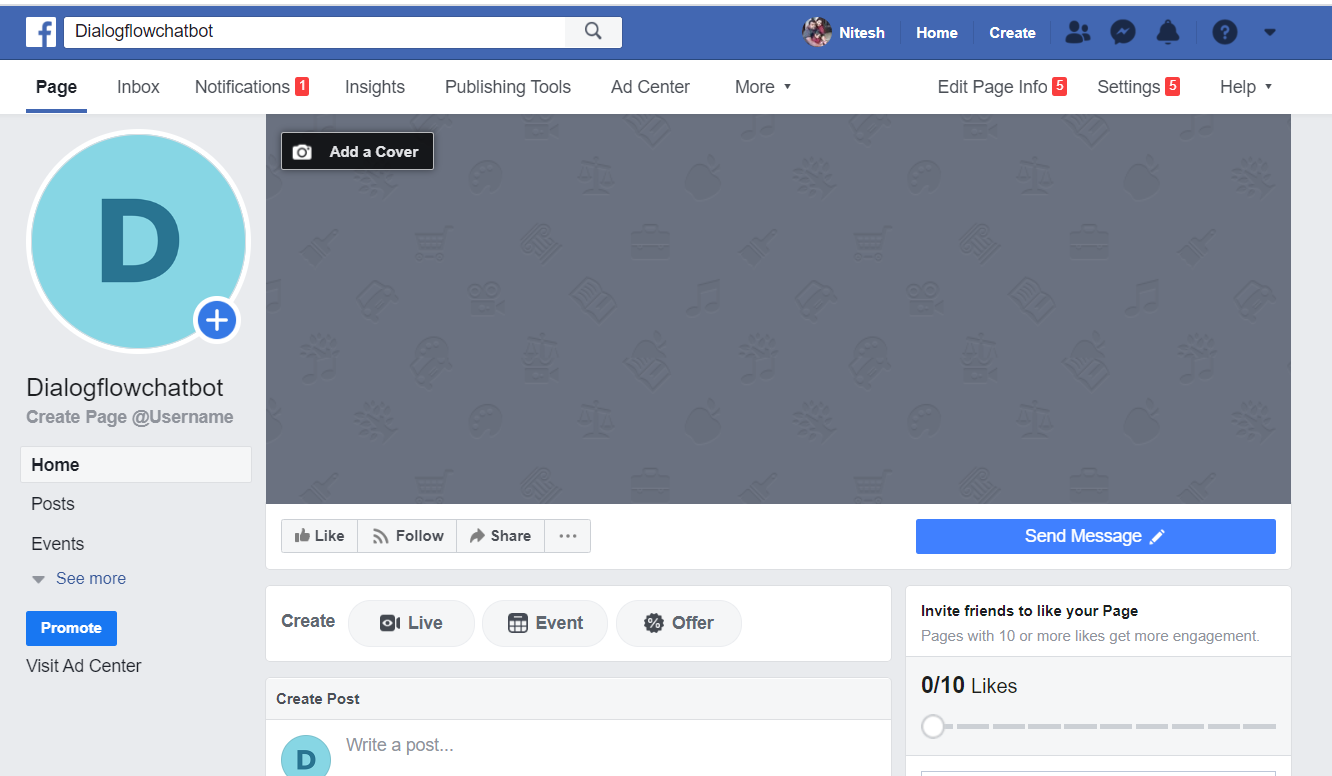
1. callback URL will be automatically filled and will be used while configuring Facebook.
2. Start and stop services using start and stop button in below.
3. Verify token is a kind of password. This will be given while configuring the Facebook.
4. Page Access Token will be coming after we configure Facebook for our bot.

[More documentation](https://cloud.google.com/dialogflow/docs/integrations/facebook) can be found here.

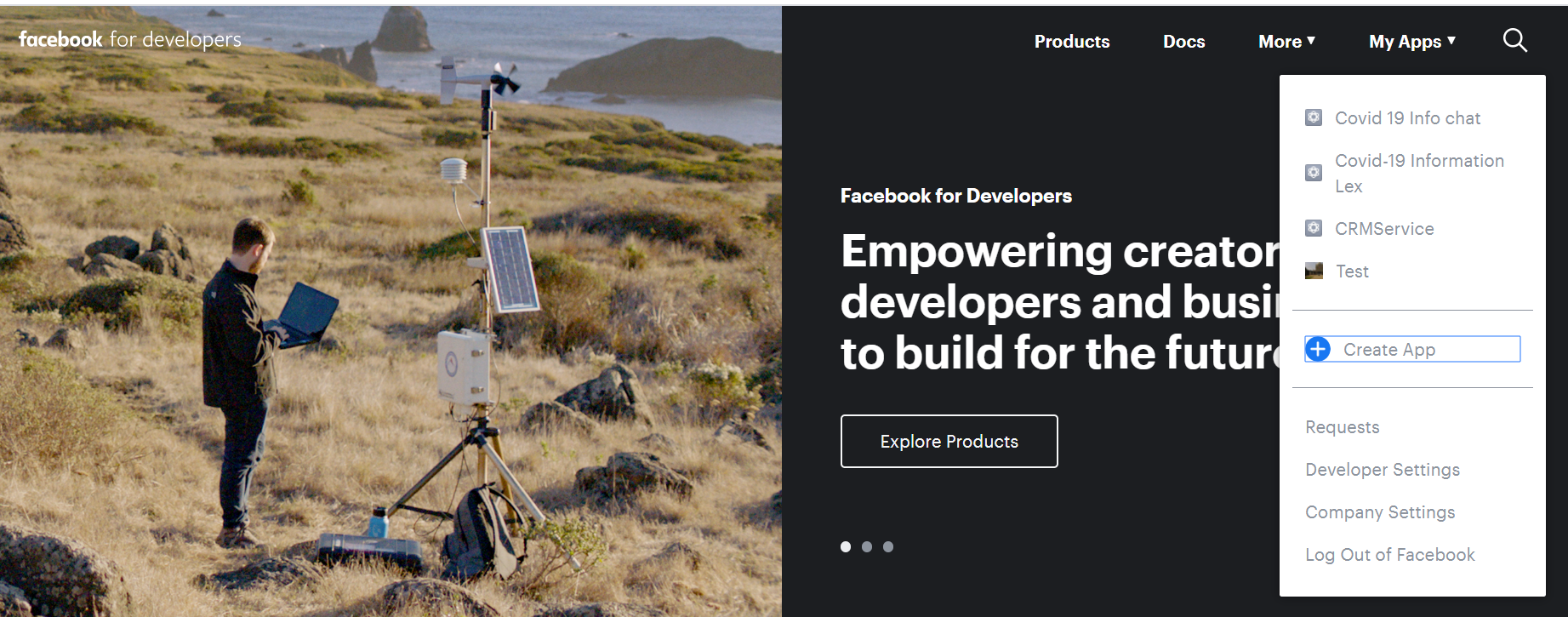
2. Facebook Messenger configuration:

a. Create a Public Page in Facebook and configure the “Send button” to show the “send message”





2. Open developers.facebook.com and register as a developer if not. This login is different from usual Facebook social media login.

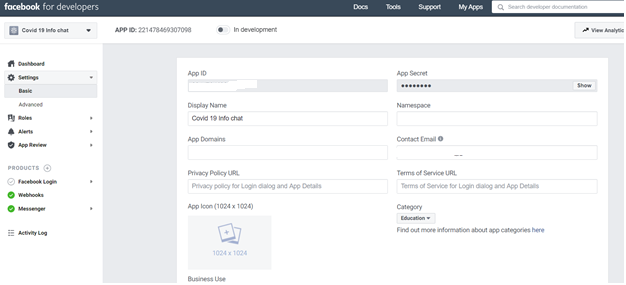


3. Create an app from section on right hand top corner.

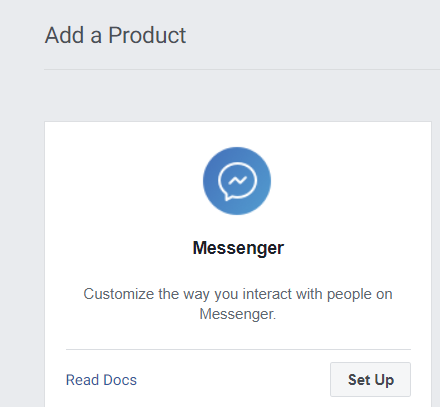
4. A page will be opened as below:



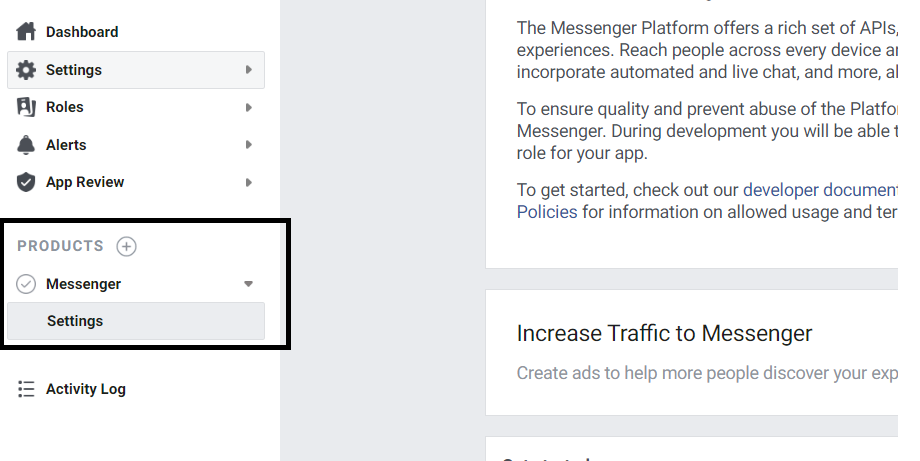
After filling data, the below window will be opened.



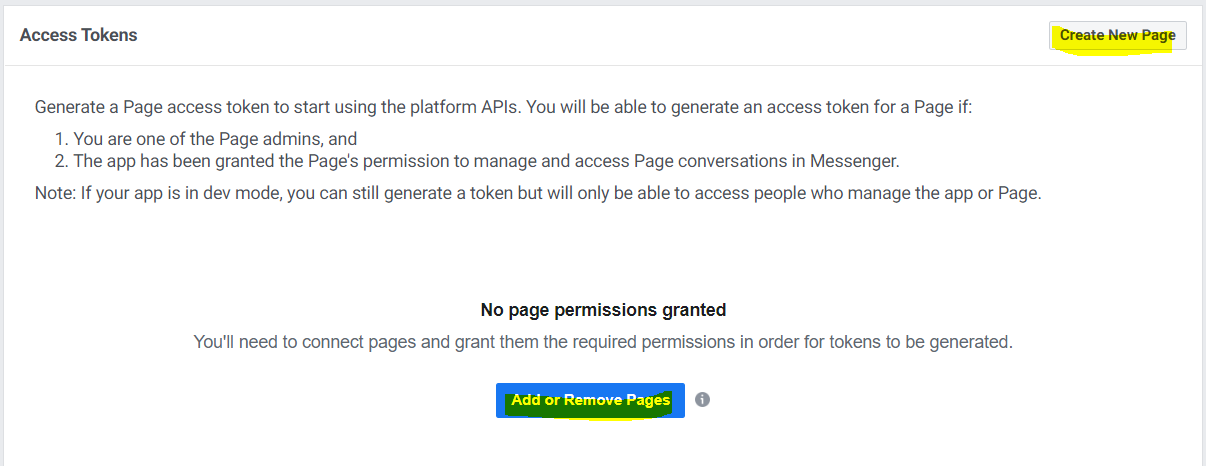
Click on messenger set up as below:



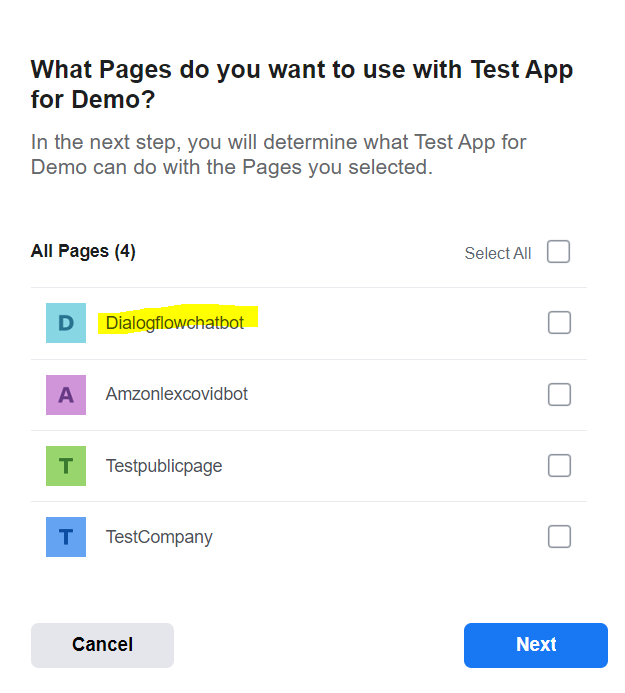
It will open setting section under Products->Messenger section as below:



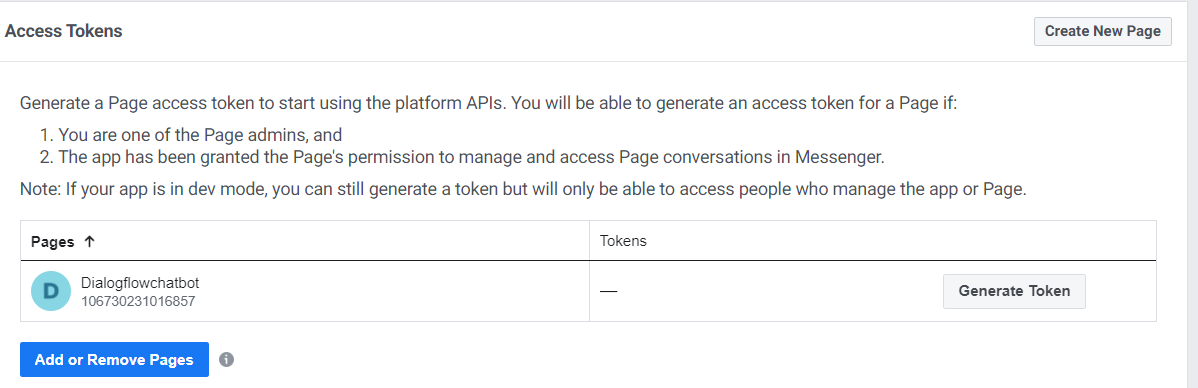
Create or add/remove public pages from Facebook as below. Since we have created public page already, we need to click on add/remove page.



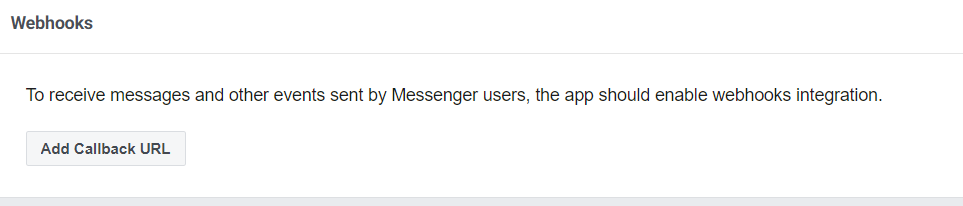
It opens to select the page that we have created in Facebook as below:



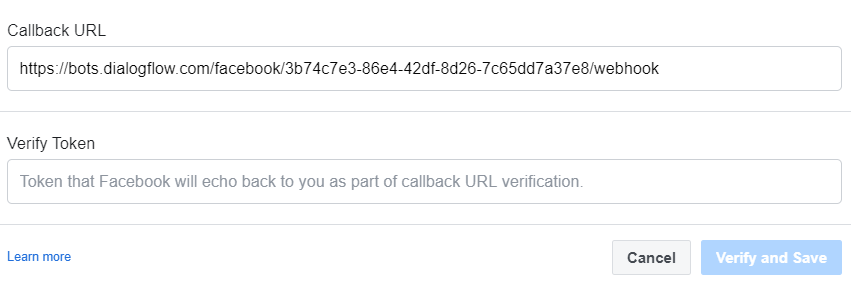
This will keep the below section as below:



Configure the webhook URL:

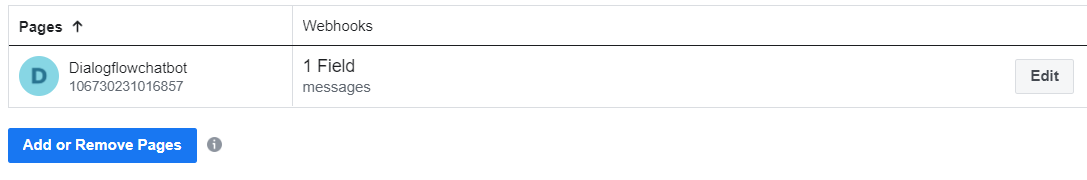


Click on Add Callback URL to configure the Dialogflow details

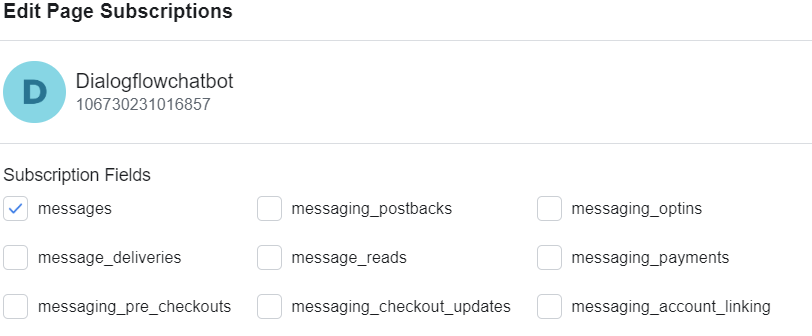


Callback URL and verify token from dialogflow.

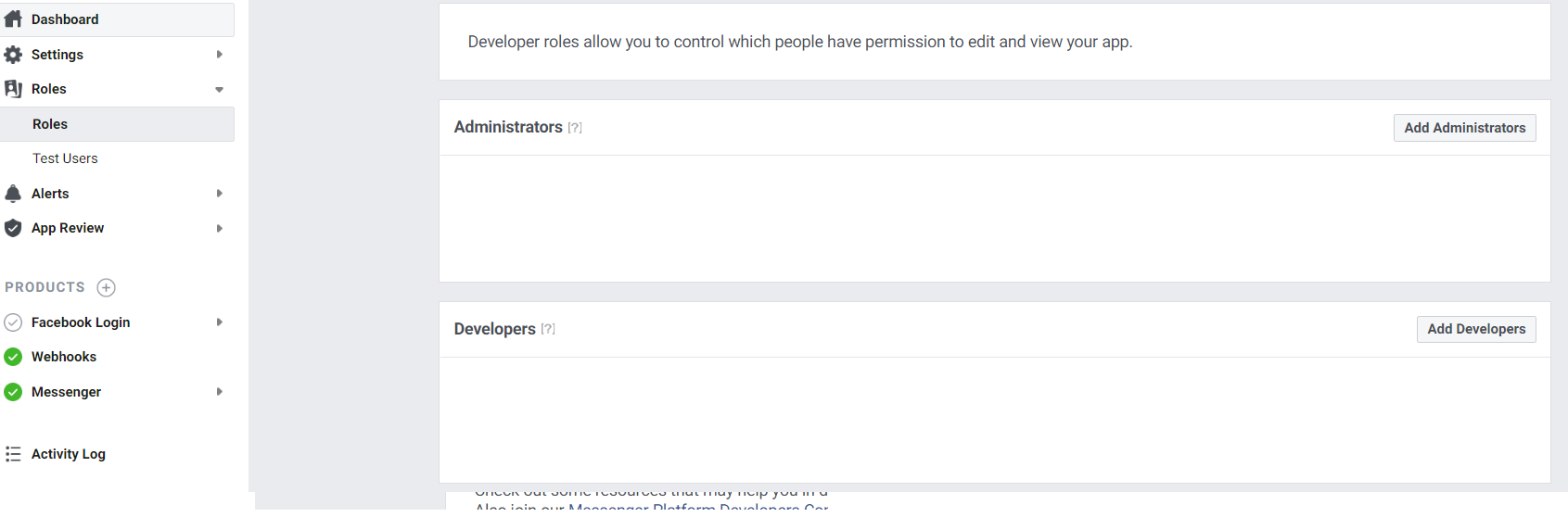
Integrate the page and webhook as below:



Click on message in window that will be opened after clicking on edit:



Define roles to test the application. Add the users, Facebook friends if you want to allow them to configure or test application based on the permissions given as below:

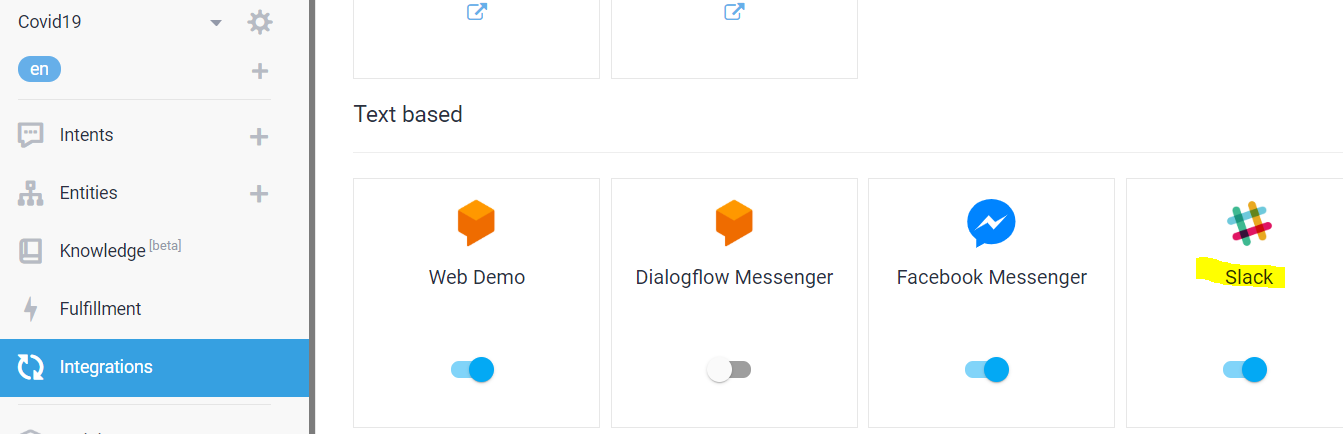


Finally, you are good to go.

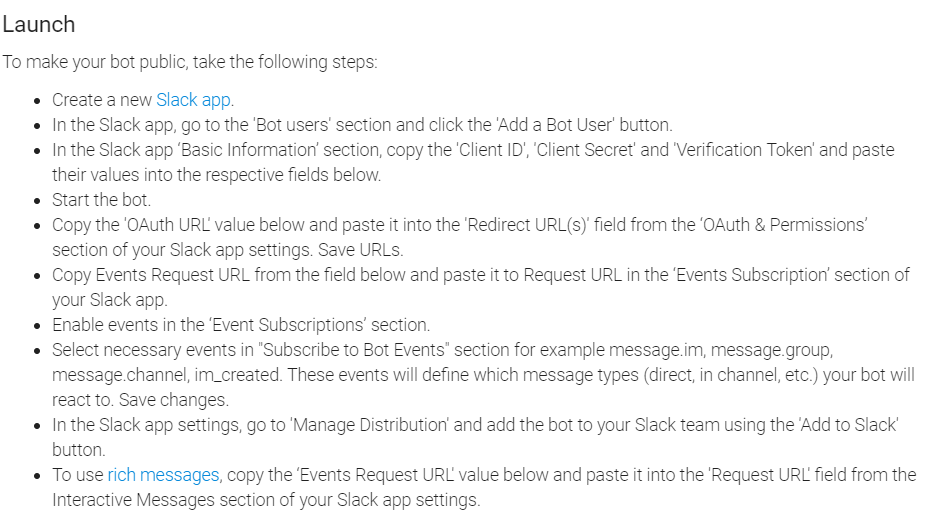
# 10. Integration with Slack

The Google Dialogflow support the chat interaction with Slack also. To configure two steps to be taken:

1. Open the integration section and switch on the slack button:



The below steps mentioned below are detailed to configure the Slack and Dialogflow:

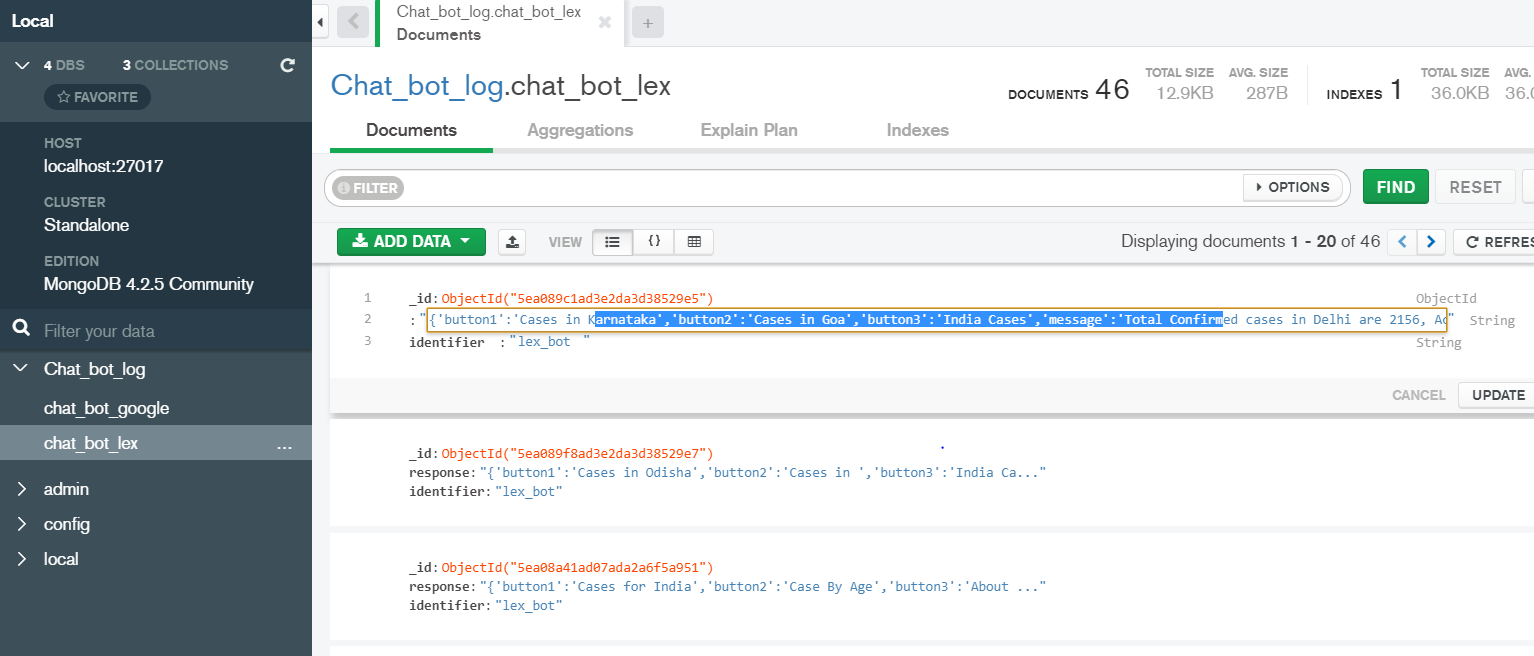


# Saving transactions in MongoDB:

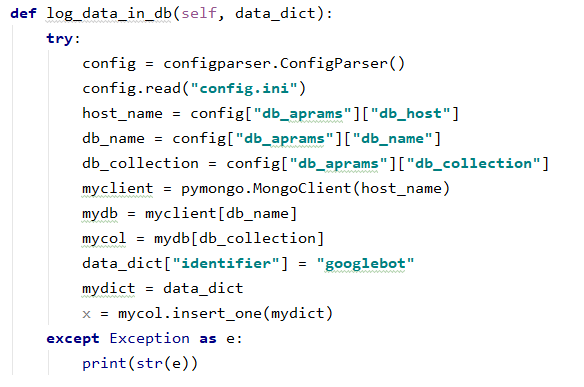
Every query and its response have to be recorded in any Database. For this demo purpose, MongoDB with NoSQL is used. This document assumes that MongoDB is setup.

The details for this chatbot are as below:

* db\_host=mongodb://<servername>:27017/ \*\*\* localhost incase it is running on your machine
* db\_name=Chat\_bot\_log \*\*\* Database name
* db\_collection=chat\_bot\_google \*\* Collection name
* This will be seen in MongoDB Compass as below:



Code to insert a data in MongoDB using python is as below:



The input to save record in DB will be in dictionary format like below. I have added an identifier select and do delete operation to clean database in required.

{

‘query’: ’Total cases in up,

‘response’: ’ Total Confirmed cases in Uttar Pradesh are 1449, Active cases 1255, Recovered cases 173 and Total deaths 21’,

‘identifier’: ‘googlebot’

}

# Conclusion:

The following observation are listed as part of learning outcomes from the above exercise:

1. Configuring webhooks using python and Flask make this integration easy, maintainable and flexible to scale.

2. Slack may not support the buttons coming from Card response. Slack will be more transacted through text commands only.

# Reference:

* https://www.livemint.com/news/india/mapped-the-spread-of-coronavirus-across-india-s-districts-11587179250870.html
* 2.https://www.ecdc.europa.eu/sites/default/files/styles/is\_full/public/images/novel-coronavirus-COVID-19-geographical-distribution-world-<date>.jpg
* <https://www.indiatoday.in/diu/story/coronavirus-india-young-patients-age-groups-covid19-1662698-2020-04-03>
* <https://www.mohfw.gov.in/>
* <https://github.com/pragnakalp/dialogflow-webhook-response-libary-in-python>