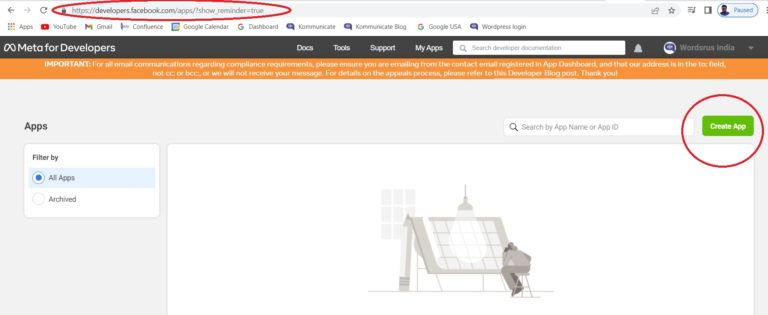
Whatsapp Integration Chatbot

# Step 1: Log in to the Facebook developer account

The first step to setting up the WhatsApp Cloud API is to visit developers.facebook.com and log in using your Facebook credentials.

You will then get an option to “Create App.” Click on it.



# Step 2: Select the app type

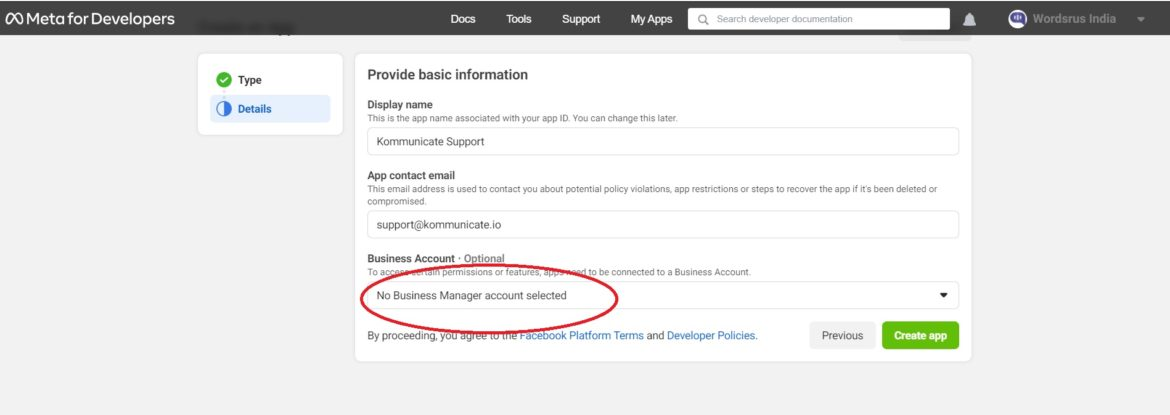
In the next step, select “Business” as your app type.

# 

# Step 3: Provide basic information

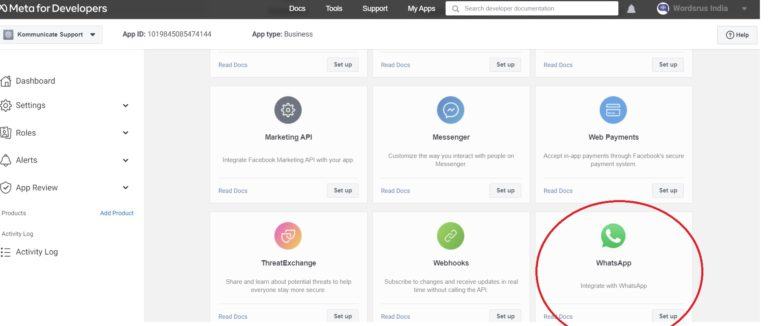
In this next step, you will have to provide basic information about your business. At this juncture, we would like to say that with the introduction of WhatsApp Cloud API, you no longer need a verified Facebook Business Manager account to get up and running, although we still recommend you to get a Facebook Business Manager account verified on the side.

For now, you can select “No Business Manager account selected” and click “Create App.”



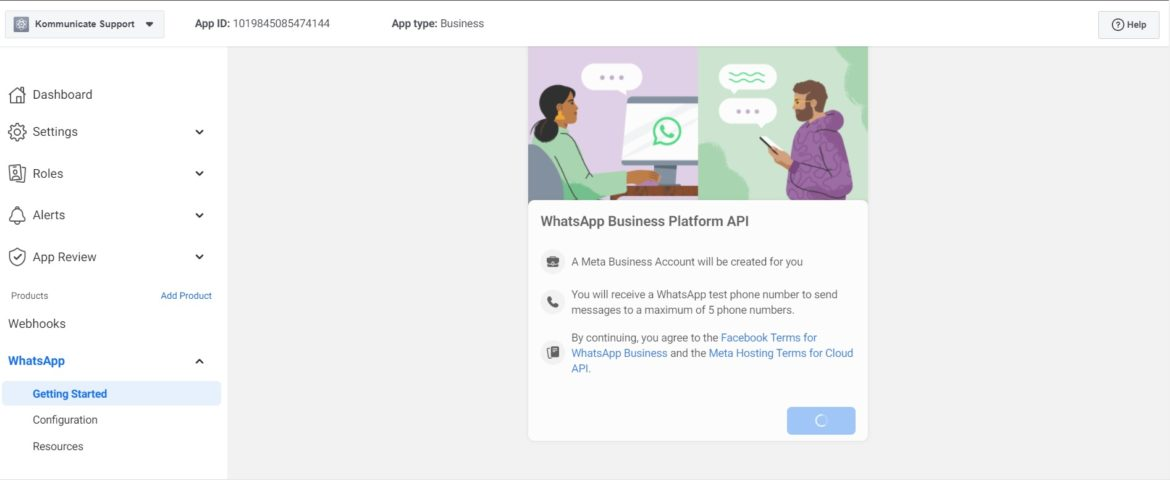
# Step 4: Set up WhatsApp from the Dashboard

You would now have successfully logged into your Meta dashboard. From here, scroll down to the WhatsApp section and click “Set up.”



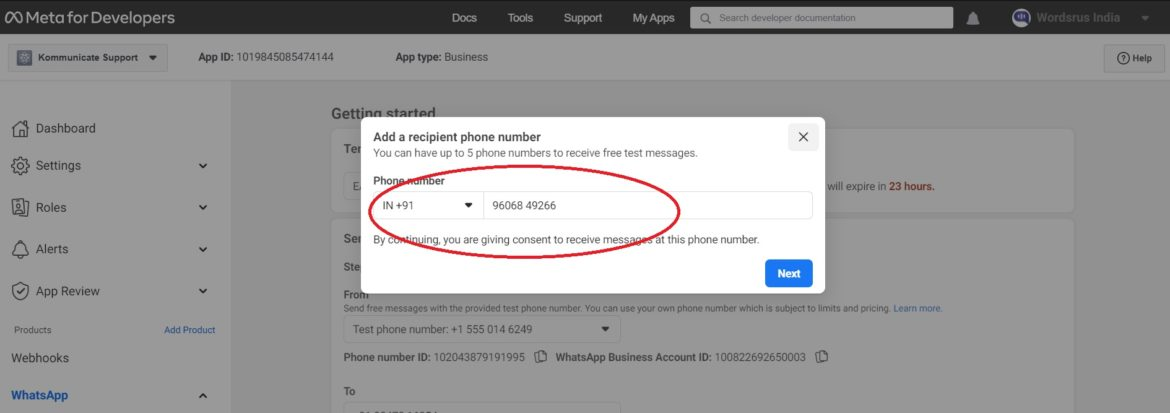
# Step 5: Accept WhatsApp Business Platform API conditions

The next screen shows you a list of the Terms and Conditions you must accept before using the WhatsApp Cloud API. Hit the “Continue” button, and you are good to go.



# Step 6: Send a test message to a WhatsApp-enabled number

We now come to our final screen. Here you are given a unique Phone number ID and a unique WhatsApp Business Account ID. The “From” number is a Test number that is assigned to you. The “To” number is the number you wish to send your first WhatsApp text to.



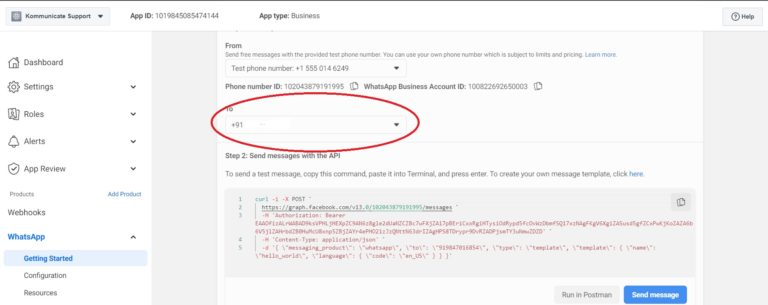
# Step 7: Enter the verification code you received

You will receive a verification code on your mobile phone from Facebook Business. Enter this number into the boxes provided to finish the verification.

# 

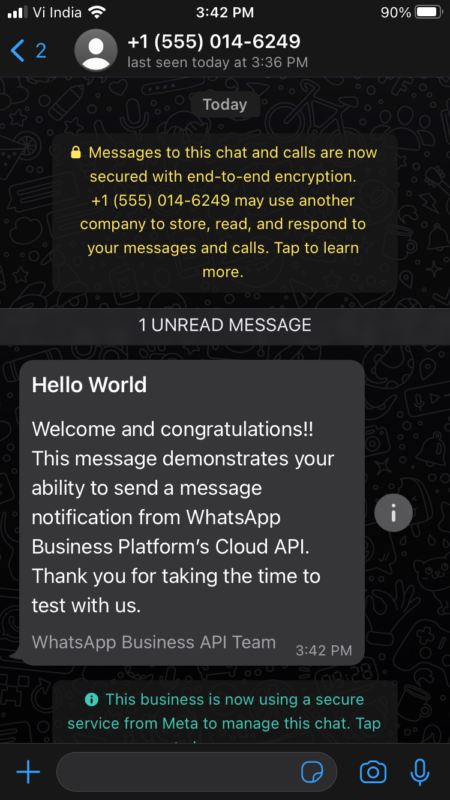
# Step 8: Send your first message over WhatsApp Cloud API

It is now time for you to test if your integration of the customer number is a success. Select the message that you have just verified from the drop-down box and hit “Send Message.”



# 

# The message will look something like this.



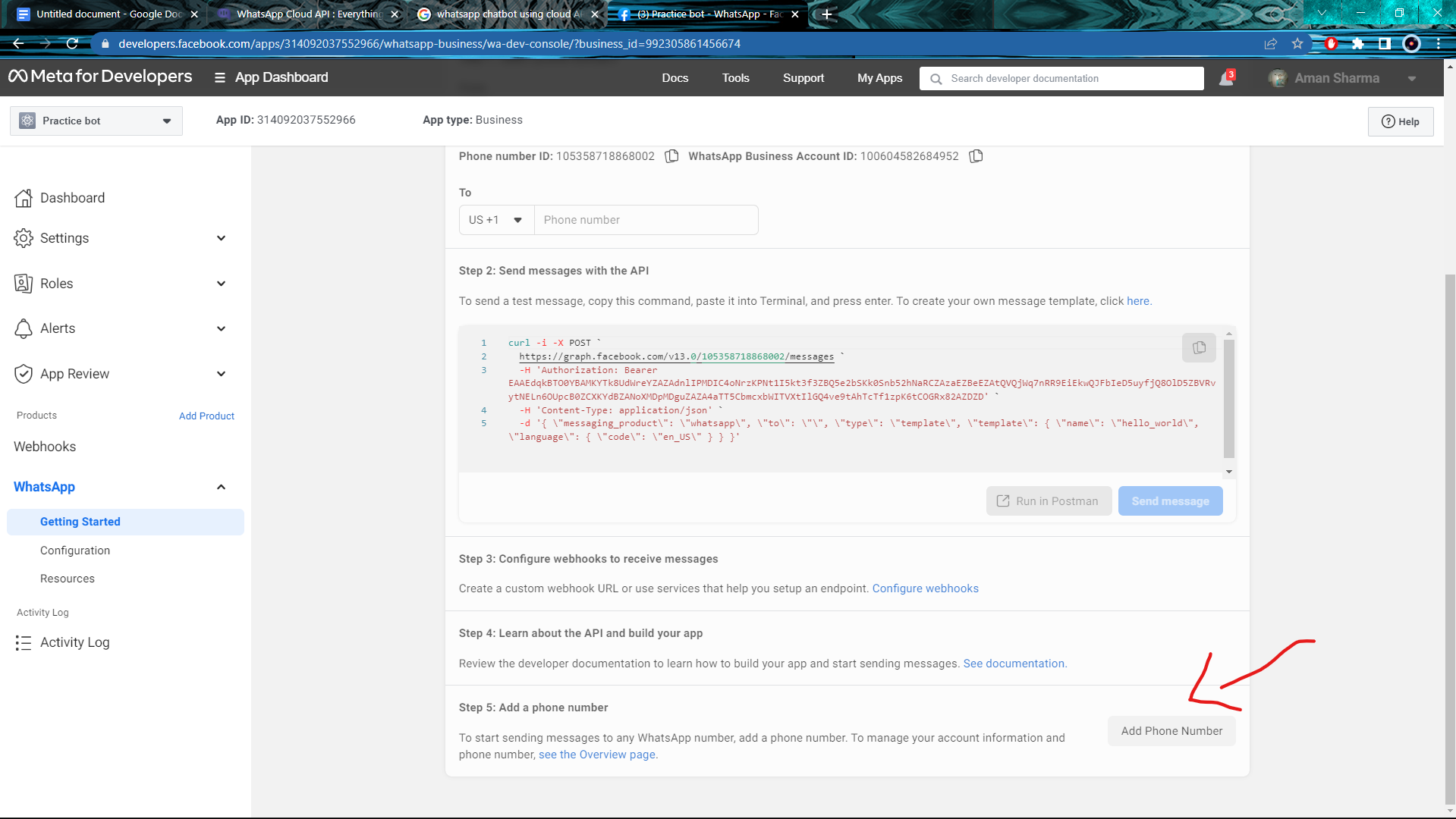
# Tip 1:

To Make your own templates click [here](https://developers.facebook.com/micro_site/url/?click_from_context_menu=true&country=IN&destination=https%3A%2F%2Fbusiness.facebook.com%2Fwa%2Fmanage%2Fmessage-templates%2F%3Fbusiness_id%3D992305861456674%26waba_id%3D100604582684952&event_type=click&last_nav_impression_id=01JocMwvfWtOD6N7w&max_percent_page_viewed=96&max_viewport_height_px=907&max_viewport_width_px=1740&orig_http_referrer=https%3A%2F%2Fdevelopers.facebook.com%2Fapps%2F314092037552966%2Fwhatsapp-business%2Fwa-dev-console%2F%3Fbusiness_id%3D992305861456674&orig_request_uri=https%3A%2F%2Fdevelopers.facebook.com%2Fapps%2F314092037552966%2Ftasks%2Fvisit_app_dashboard%2Fcomplete%2F&region=apac&scrolled=true&session_id=1CSTSde2QopoagZXL&site=developers)

–Templates are important Because without them you cannot start a conversation, The first message sent must be a template, and the receive must reply to it for you to be able to send other typ of messages.

# Step 9: Add a phone number

To start sending messages to any WhatsApp number,you need to add a phone number. If you have a whatsapp account on that number, then you will have to delete it, in order to use that number for chatbot(The number given by facebook, allows only conversation with 5 people)



THE CODE(Without RASA Integration)

import heyoo

from heyoo import WhatsApp

Temp\_access\_token='EAAEdqkBTO0YBALmjtqH25q8U5ByE1l7mjT23w1jlYOvwxd0qpzOIpVeazHsy2eZCx8ZCUWlEw9V2ZBW2FOZCoeSUX6LCrw0KP9GZARQ75DD1aRkRXxwJXUkZBG3QW4KaNZCpk4VNufxwEx0cLOYkcNhY24poMH4R8I4PVHUiUGrnYTwRPLFt7ZCBTnLurwDKnz9spanjifJOnQZDZD'

messenger = WhatsApp(Temp\_access\_token,phone\_number\_id='105358718868002')

#messenger.send\_template('friend\_update',919650998281)

import os

import json

from dotenv import load\_dotenv

from flask import Flask, request

app = Flask(\_\_name\_\_)

# Load .env file

load\_dotenv()

VERIFY\_TOKEN = "asanify"

@app.route("/webhook", methods=["GET", "POST"])

def hook():

if request.method == "GET":

if request.args.get("hub.verify\_token") == VERIFY\_TOKEN:

return request.args.get("hub.challenge")

return "Invalid verification token"

data = request.get\_json()

changed\_field = messenger.changed\_field(data)

if changed\_field == "messages":

new\_message = messenger.get\_mobile(data)

if new\_message:

mobile = messenger.get\_mobile(data)

message\_type = messenger.get\_message\_type(data)

print(message\_type)

print(data)

if message\_type == "text":

message = messenger.get\_message(data)

name = messenger.get\_name(data)

#print(data)

print(f"{name} with this {mobile} number sent {message}")

if message=='Y':

messenger.send\_template("friend", mobile)

elif message[0]=='S':

messenger.send\_message(message[1:], mobile)

messenger.send\_message("See, now your turn to tell me some thing", mobile)

elif message=='N':

messenger.send\_message(f"CRY NOISES!!!!!", mobile)

else:

messenger.send\_message(f"Hi {name}, Nice to connect with you, if you want to be my friend type 'Y' ", mobile)

#get Location

elif message\_type == "location":

print(data)

location=data['entry'][0]['changes'][0]['value']['messages'][0]['location']

address=list(location.values())

#print(address)

messenger.send\_message(f"{data}", mobile)

#get image

elif message\_type == "image":

print(data)

elif message\_type == "interactive":

message\_response = messenger.get\_interactive\_response(data)

print(message\_response)

else:

pass

else:

delivery = messenger.get\_delivery(data)

if delivery:

print(f"Message : {delivery}")

else:

print("No new message")

return "ok"

if \_\_name\_\_ == "\_\_main\_\_":

app.run(port=5002, debug=True)

Payloads

# Current location payload:

{'object': 'whatsapp\_business\_account', 'entry': [{'id': '100604582684952', 'changes': [{'value': {'messaging\_product': 'whatsapp', 'metadata': {'display\_phone\_number': '918178742757', 'phone\_number\_id': '105358718868002'}, 'contacts': [{'profile': {'name': 'Mamta'}, 'wa\_id': '917835077600'}], 'messages': [{'from': '917835077600', 'id': 'wamid.HBgMOTE3ODM1MDc3NjAwFQIAEhggRDdGMjY3NzZCMUJGQjIzNzkwODk0NTgzMTMzQTc1MDQA', 'timestamp': '1654530649', 'location': {'latitude': 28.5025544, 'longitude': 77.0230679}, 'type': 'location'}]}, 'field': 'messages'}]}]}

# Geo-specified location payload:

{'object': 'whatsapp\_business\_account', 'entry': [{'id': '100604582684952', 'changes': [{'value': {'messaging\_product': 'whatsapp', 'metadata': {'display\_phone\_number': '918178742757', 'phone\_number\_id': '105358718868002'}, 'contacts': [{'profile': {'name': 'Mamta'}, 'wa\_id': '917835077600'}], 'messages': [{'from': '917835077600', 'id': 'wamid.HBgMOTE3ODM1MDc3NjAwFQIAEhggQTRCODcwMDJGNUY1ODg4OTk1QUM2MDQ1MEFFQUM0QzAA', 'timestamp': '1654530667', 'location': {'address': 'Ground Floor, Ansal Plaza Mall, Gurgaon, Haryāna', 'latitude': 28.508034005337, 'longitude': 77.030052599386, 'name': 'Cafe Coffee Day (CCD)', 'url': 'https://foursquare.com/v/4e8830337ee60ccd06f7b181'}, 'type': 'location'}]}, 'field': 'messages'}]}]}

# Image payload:

{'object': 'whatsapp\_business\_account', 'entry': [{'id': '100604582684952', 'changes': [{'value': {'messaging\_product': 'whatsapp', 'metadata': {'display\_phone\_number': '918178742757', 'phone\_number\_id': '105358718868002'}, 'contacts': [{'profile': {'name': 'Aman Sharma'}, 'wa\_id': '919650998281'}], 'messages': [{'from': '919650998281', 'id': 'wamid.HBgMOTE5NjUwOTk4MjgxFQIAEhggMzhDODkyRTQyQzk5OTE3NzJBMjI2RTZBNzQxQ0JCOTEA', 'timestamp': '1654440973', 'type': 'image', 'image': {'mime\_type': 'image/jpeg', 'sha256': 'PcW1OREkSbz7dGdA10CSqPyrW15hsZGL+CySq1QTSKY=', 'id': '321625846835927'}}]}, 'field': 'messages'}]}]}

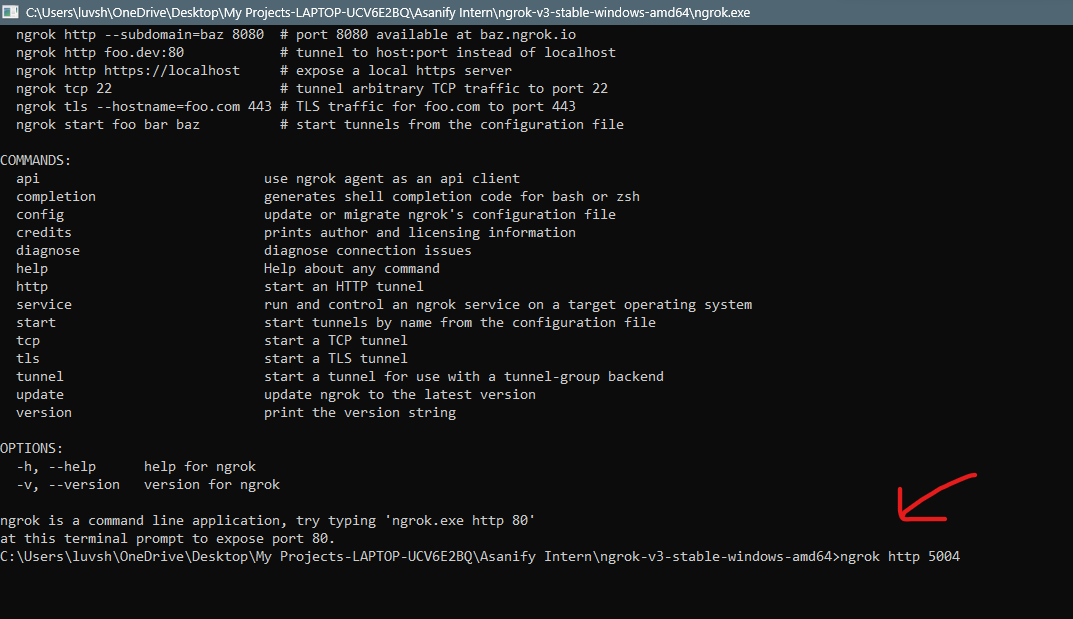
Webhooks:

# 

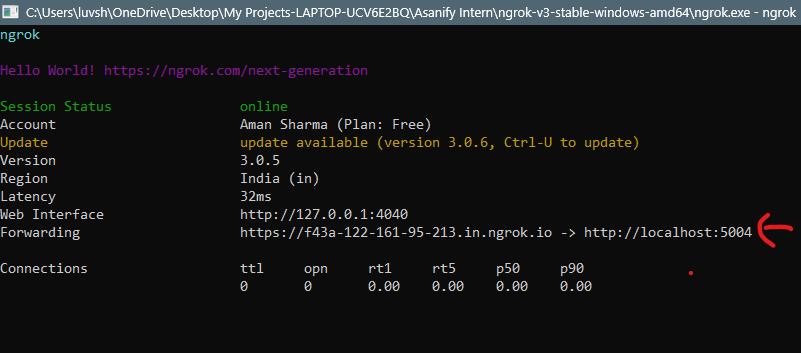
# NGROk

Install NGROK and create an account(necessary for a free subscription)

Open NGROk and Type → ngrok http <your flask app port number>



Now Copy the Forwarding url :

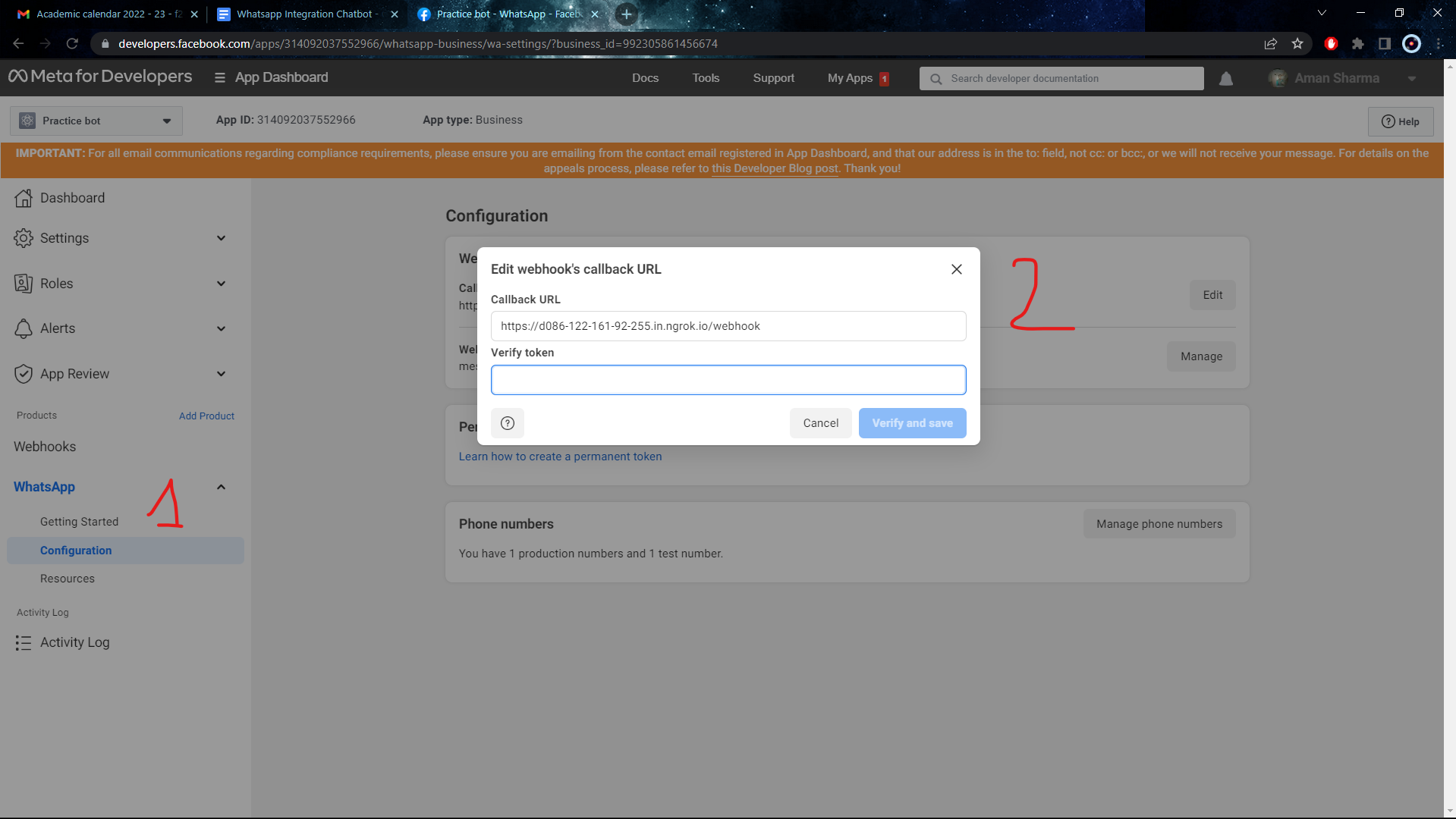


# Add callback URl:

Go to you app on facebook developer

Click on configurations

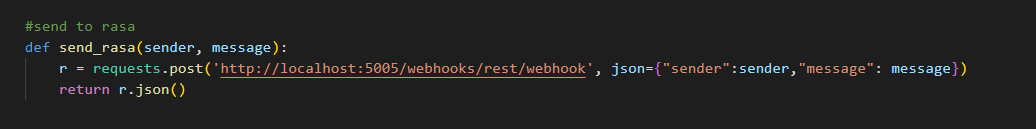
Add your callback url and verification token( as mentioned in your flask app)



Subscribe to messages in the webhook field.

Code (With RASA)

Using http get and post requests to connect flask app with RASA server



import requests

import heyoo

from heyoo import WhatsApp

import os

import json

from dotenv import load\_dotenv

from flask import Flask, request

import shutil

from inputimeout import inputimeout, TimeoutOccurred

Temp\_access\_token='EAAEdqkBTO0YBAEhf76daRpCwZAToVzVS4JG7r1ZA55btlvcZBmCjCNZB8vuKwmR4kCTuUj0ZBDCxMIIWmQd2PQlbM44fJlZBzx7h5mdEFwmMU3N04AhxMmGZBfAFYauBDNvhZA7wv5ZC8BEgfHf2hFZCCcmhpSzxLP8jTEtqxydMia93Ne4ep7ZAFLVAoAj4OObrvhTYzIj42Dn3QZDZD'

hed = {'Authorization': 'Bearer ' + Temp\_access\_token}

messenger = WhatsApp(Temp\_access\_token,phone\_number\_id='105358718868002')

print('Do you want to send a message')

#send to rasa

def send\_rasa(sender, message):

r = requests.post('http://localhost:5005/webhooks/rest/webhook', json={"sender":sender,"message": message})

return r.json()

#Send Message

def send(type,to,data=None,id=None):

if type == 'text':

if data is not None:

messenger.send\_message(data,to)

else:

data=input('Enter Message')

messenger.send\_message(data,to)

elif type == 'template':

if data is not None:

messenger.send\_template(data,to)

else:

data=input('Enter Template name')

messenger.send\_template(data,to)

elif type == 'image':

if id!=None:

messenger.send\_image(image=f'{id}', recipient\_id=f"{to}",link=False)

else:

data=input('Enter link')

messenger.send\_image(image=f'{data}', recipient\_id=f"{to}")

elif type == 'video':

if id!=None:

messenger.send\_video(video=f'{id}', recipient\_id=f"{to}",link=False)

else:

data=input('Enter link')

messenger.send\_video(video=f'{data}', recipient\_id=f"{to}")

elif type == 'audio':

if id!=None:

messenger.send\_audio(audio=f'{id}', recipient\_id=f"{to}",link=False)

else:

data=input('Enter link')

messenger.send\_audio(audio=f'{data}', recipient\_id=f"{to}")

elif type == 'document':

if id!=None:

messenger.send\_document(document=f'{id}', recipient\_id=f"{to}",link=False)

else:

data=input('Enter link')

messenger.send\_document(document=f'{data}', recipient\_id=f"{to}")

elif type == 'location':

if data!=None:

messenger.send\_location(lat=f"{data['latitude']}", long=f"{data['longitude']}",name='Your current location',address=None,recipient\_id=f"{to}")

else:

lat=input('enter lat')

long=input('enter lon')

name=input('enter name')

address='enter address'

messenger.send\_location(lat=f'{lat}',long='{long}',name=f'{name}',address=f'{address}', recipient\_id=f"{to}")

#Recieve message

def recieve(type,data):

if type == 'text':

message = messenger.get\_message(data)

print(message)

return message

elif type == 'template':

pass

elif type == 'image':

im\_id=data['entry'][0]['changes'][0]['value']['messages'][0][f'{type}']['id']

url='https://graph.facebook.com/v13.0/' + data['entry'][0]['changes'][0]['value']['messages'][0]['image']['id']

url='https://graph.facebook.com/v13.0/' + im\_id

response = requests.get(url, headers=hed)

return [im\_id,response.json(),response]

elif type == 'video':

ved\_id=data['entry'][0]['changes'][0]['value']['messages'][0][f'{type}']['id']

url='https://graph.facebook.com/v13.0/' + ved\_id

response = requests.get(url, headers=hed)

return [ved\_id,response.json(),response]

elif type == 'audio':

aud\_id=data['entry'][0]['changes'][0]['value']['messages'][0][f'{type}']['id']

url='https://graph.facebook.com/v13.0/' + aud\_id

response = requests.get(url, headers=hed)

return [aud\_id,response.json(),response]

elif type == 'document':

doc\_id=data['entry'][0]['changes'][0]['value']['messages'][0][f'{type}']['id']

url='https://graph.facebook.com/v13.0/' + doc\_id

response = requests.get(url, headers=hed)

return [doc\_id,response.json(),response]

elif type=='contacts':

contact\_info=data['entry'][0]['changes'][0]['value']['messages'][0]['contacts'][0]

return contact\_info

elif type == 'location':

address= data['entry'][0]['changes'][0]['value']['messages'][0]['location']

return address

#download message

def download(type,id,url):

file\_name='download'+'\_'+type+'\_'+id

if type == 'image':

read\_data = requests.get(url,headers=hed,stream=True)

read\_data.raw.decode=True

if read\_data.status\_code == 200:

with open(file\_name+'.jpg','wb') as f2:

shutil.copyfileobj(read\_data.raw, f2)

f2.close()

#print('Image sucessfully Downloaded: ',file\_name)

file\_path=os.path.abspath(file\_name)

return file\_path

else:

return 'image could not be downloaded'

elif type == 'video':

read\_data = requests.get(url,headers=hed,stream=True)

read\_data.raw.decode=True

if read\_data.status\_code == 200:

with open(file\_name+'.mp3','wb') as f2:

shutil.copyfileobj(read\_data.raw, f2)

f2.close()

#print('video sucessfully Downloaded: ',file\_name)

file\_path=os.path.abspath(file\_name)

return file\_path

else:

return 'image could not be downloaded'

elif type == 'audio':

read\_data = requests.get(url,headers=hed,stream=True)

read\_data.raw.decode=True

if read\_data.status\_code == 200:

with open(file\_name+'.mp3','wb') as f2:

shutil.copyfileobj(read\_data.raw, f2)

f2.close()

#print('audio sucessfully Downloaded: ',file\_name)

file\_path=os.path.abspath(file\_name)

return file\_path

else:

return 'image could not be downloaded'

elif type == 'document':

read\_data = requests.get(url,headers=hed,stream=True)

read\_data.raw.decode=True

if read\_data.status\_code == 200:

with open(file\_name+'.pdf','wb') as f2:

shutil.copyfileobj(read\_data.raw, f2)

f2.close()

#print('document sucessfully Downloaded: ',file\_name)

file\_path=os.path.abspath(file\_name)

return file\_path

else:

return 'image could not be downloaded'

app = Flask(\_\_name\_\_)

# Load .env file

load\_dotenv()

VERIFY\_TOKEN = "asanify"

@app.route("/webhook", methods=["GET", "POST"])

def hook():

if request.method == "GET":

if request.args.get("hub.verify\_token") == VERIFY\_TOKEN:

return request.args.get("hub.challenge")

return "Invalid verification token"

data = request.get\_json()

changed\_field = messenger.changed\_field(data)

if changed\_field == "messages":

new\_message = messenger.get\_mobile(data)

if new\_message:

mobile = messenger.get\_mobile(data)

name = messenger.get\_name(data)

message\_type = messenger.get\_message\_type(data)

#print(message\_type)

#print(data)

#FOR TEXT MESSAGES

if message\_type == "text":

#RECIEVE

got\_message=recieve(message\_type,data)

#print(got\_message,data)

print(f"{name} with this {mobile} number sent {got\_message}")

#Integrating with rasa

rasa\_message = send\_rasa(name,got\_message)

#SEND

for i in rasa\_message:

send('text',mobile,i['text'])

#get Location

elif message\_type == "location":

#RECIEVE

address =recieve(message\_type,data)

# Integrating with rasa

rasa\_message = send\_rasa(name,'location')

#print(rasa\_message)

#print(address)

#SEND

send('location',mobile,address)

for i in rasa\_message:

send('text',mobile,i['text'])

#get image

elif message\_type == "image":

#RECIEVE

image\_data=recieve(message\_type,data)[1]

im\_id=recieve(message\_type,data)[0]

print(recieve(message\_type,data)[2])

im\_url=image\_data['url']

#DOWNLOAD

download(message\_type,im\_id,im\_url)

print(download(message\_type,im\_id,im\_url))

# Integrating with rasa

rasa\_message = send\_rasa(name,message\_type)

#SEND

send(message\_type,mobile,None,im\_id)

send('text',mobile,'This is the image you sent')

for i in rasa\_message:

send('text',mobile,i['text'])

#get document

elif message\_type == "document":

#RECIEVE

doc\_data=recieve(message\_type,data)[1]

doc\_id=recieve(message\_type,data)[0]

print(recieve(message\_type,data)[2])

doc\_url=doc\_data['url']

#DOWNLOAD

download(message\_type,doc\_id,doc\_url)

print(download(message\_type,doc\_id,doc\_url))

#SEND

send(message\_type,mobile,None,doc\_id)

send('text',mobile,'This is the document you sent')

#get contact

elif message\_type == "contacts":

contact\_info=str(recieve(message\_type,data))

print(contact\_info)

send('text',mobile,contact\_info)

#get\_audio

elif message\_type == "audio":

#RECIEVE

aud\_data=recieve(message\_type,data)[1]

aud\_id=recieve(message\_type,data)[0]

print(recieve(message\_type,data)[2])

aud\_url=aud\_data['url']

#DOWNLOAD

download(message\_type,aud\_id,aud\_url)

print(download(message\_type,aud\_id,aud\_url))

#SEND

send(message\_type,mobile,None,aud\_id)

send('text',mobile,'This is the audio you sent')

#get\_vedio

elif message\_type == "video":

#RECIEVE

ved\_data=recieve(message\_type,data)[1]

ved\_id=recieve(message\_type,data)[0]

print(recieve(message\_type,data)[2])

ved\_url=ved\_data['url']

#DOWNLOAD

download(message\_type,ved\_id,ved\_url)

print(download(message\_type,ved\_id,ved\_url))

#SEND

send(message\_type,mobile,None,ved\_id)

send('text',mobile,'This is the video you sent')

elif message\_type == "interactive":

message\_response = messenger.get\_interactive\_response(data)

print(message\_response)

else:

pass

else:

delivery = messenger.get\_delivery(data)

if delivery:

print(f"Message : {delivery}")

else:

print("No new message")

return "ok"

if \_\_name\_\_ == "\_\_main\_\_":

#USER INPUT

try:

user\_input = inputimeout(prompt='You have 10 seconds to type "Y" or "N"---->', timeout=10)

except TimeoutOccurred:

user\_input = 'N'

print('Lets move ahead')

if user\_input=='Y':

message\_type = input('enter message type')

to=input('enter number preceeded by 91')

send(message\_type, to)

else:

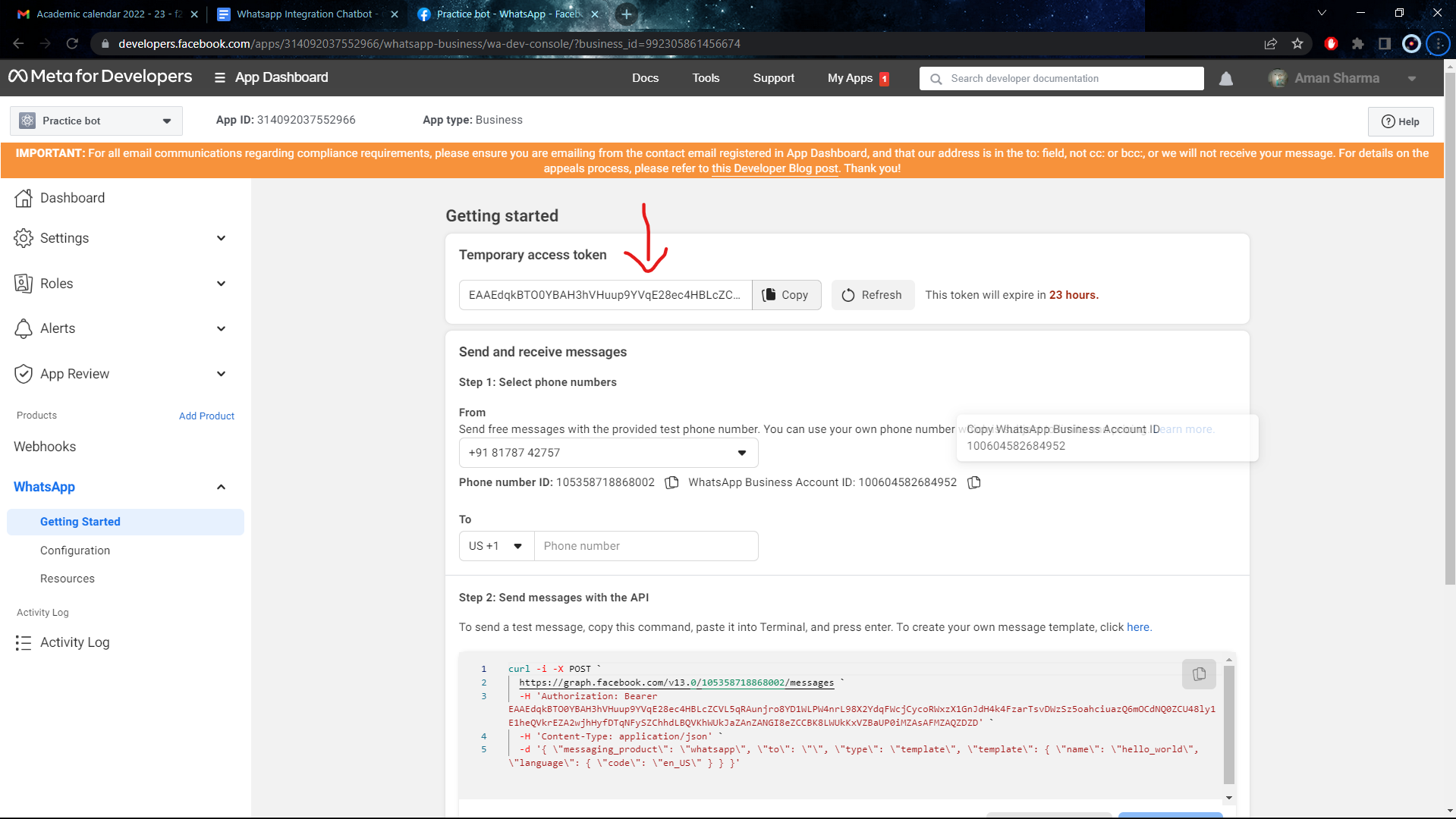
pass

#CLIENT INPUT

app.run(port=5002, debug=True)

Temporary access token:

Changes after every 24 hours, so need to be updated every day for the chatbot to work.



# TIP:

Create a permanent access token or use api requests to get the access token automatically.

