SET UP ANDROID STUDIO

PATH FOR APK: Secretary\Sarah\app\build\outputs\apk\debug

Open the android studio project saved under the name “Sarah” and click on the run button. Make sure that you have the latest version of android studio installed and set up on your desktop.

Compatible devices: All Android Devices

The application can also be run using genymotion or android studio’s emulator.

Some of the features such as calendar and contacts will not open on the Samsung device due to the security reasons as the app is not downloaded from the play store, disabling them would allow the application to function well.

Make sure that Google contacts, calendar and Gmail are installed on your device.

NOTE MAKER

Pre-requisites

- Software
  - PyCharm: https://www.jetbrains.com/pycharm/download (**The Python script must run on Python version 3.7 or below as with 3.8 or above)**
  - Android Studio: https://developer.android.com/studio
- Firebase
  - Make an account on google firebase, and connect it to android studio as well as python with the help of pyrebase.
  - Firebase Storage needs to be connected.

Procedure

- Python – PyCharm
  - Install all these libraries by clicking on the bulb which pops up or typing them down manually in the PyCharm terminal
  - pip3 install pyrebase4

main.py

```
import importlib
import urllib
from sklearn.feature_extraction.text import TfidfVectorizer
import string
from nltk.corpus import stopwords
from nltk import sent_tokenize, word_tokenize, WordNetLemmatizer
import pandas as pd
import pyrebase
from warnings import simplefilter
import inputs
```

- Make sure to download the JSON file (and store in appropriate location) from firebase service account before setting up config.

```
config = {
    "apiKey": "AIzaSyB70srkaYwGkf8cI23JcnuhVG2KVhPGk4s",
    "authDomain": "virtual-secretary-fec6f.firebaseapp.com",
    "databaseURL": "https://virtual-secretary-fec6f.firebaseio.com",
    "projectId": "virtual-secretary-fec6f",
    "storageBucket": "virtual-secretary-fec6f.appspot.com",
    "messagingSenderId": "118347623643",
    "appId": "1:118347623643:web:f74b35f786bd3ebc30b0fe",
    "measurementId": "G-NPW2T0EWJ1",
    "serviceAccount": "/Users/faizarahman/Desktop/virtual-secretary.json"
}
```

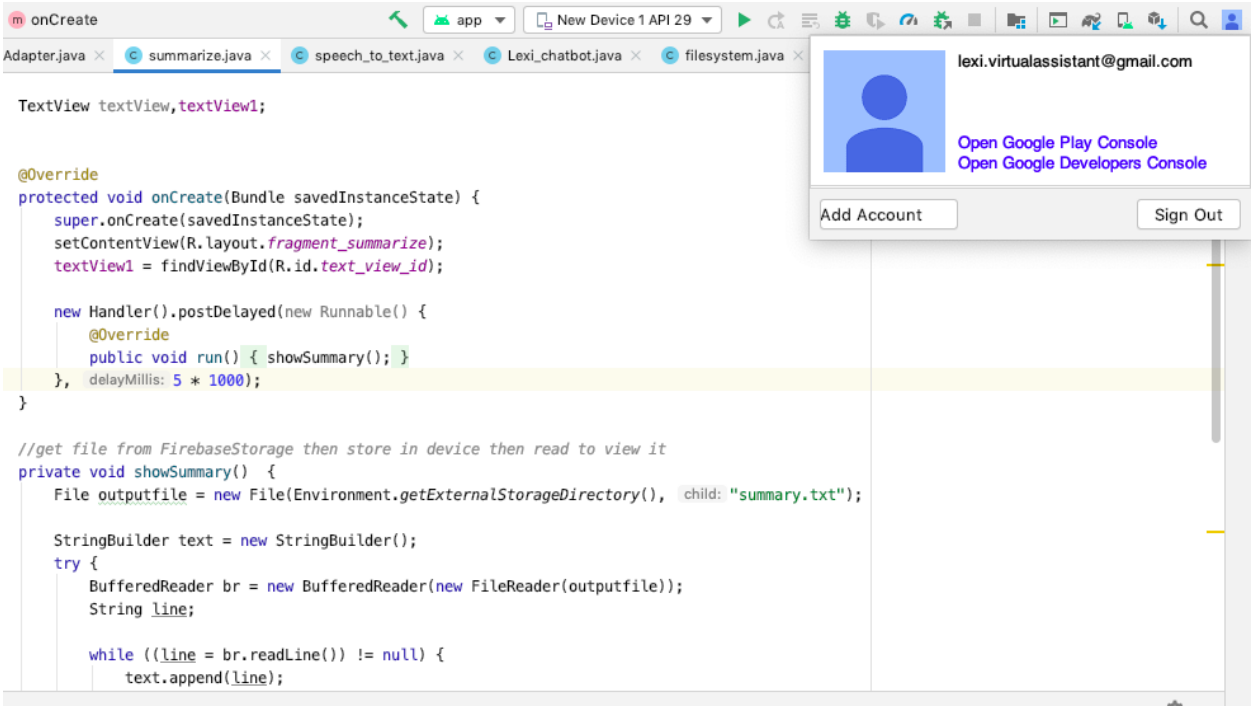
- When a new file is created on your device, make sure to provide its path in order to read its content.

```
file1 = open("MyFile.txt", "w+")
file1.write(summary)
file1.read()
file1.close()
with open('MyFile.txt', 'r') as f2:
    data = f2.read()

fixresult = "/Users/faizarahman/Desktop/MyFile.txt"
cloudfilename = "Summarized Text"
storage.child(cloudfilename).put(fixresult)
print(storage.child(cloudfilename).get_url(None))
```

NOTE: The python script has to be run manually when the summarize button is clicked, to update the summary file for the recording requested, on firebase storage.

- Android Studio
    - Make sure to log into the same firebase account.



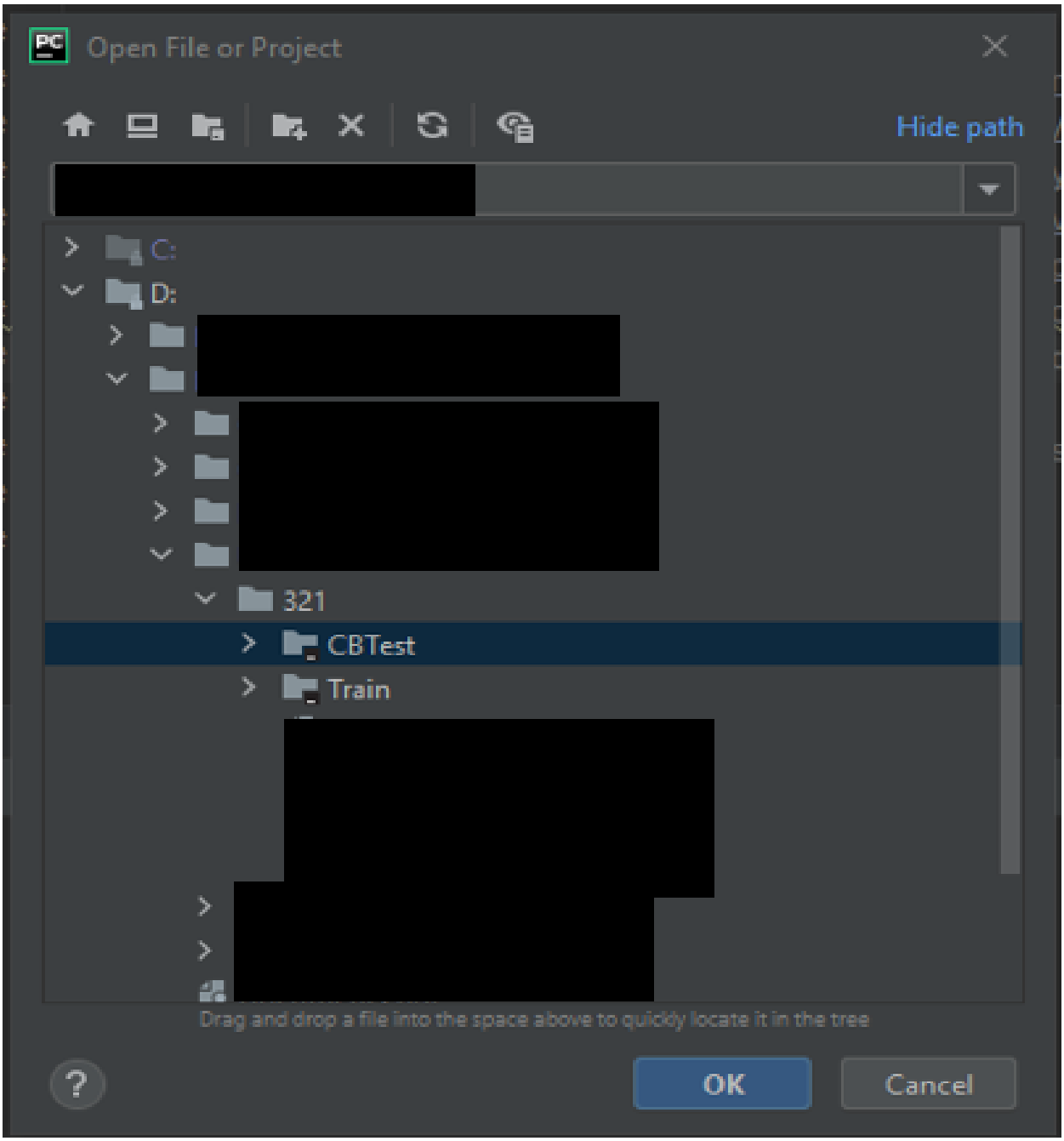
VOICE ASSISTANT

Pre-requisites

- Software
  - Anaconda: https://www.anaconda.com/products/individual (**The Python script must run on Python version 3.7 or below as with 3.8 or above, some libraries such as tensor flow used for training models are not supported**)
  - PyCharm: https://www.jetbrains.com/pycharm/download
  - Android Studio: https://developer.android.com/studio
  - NGROK: https://ngrok.com/download **(On Windows, NGROK must be downloaded and installed in the System32 directory)**
  - WinRAR: https://www.win-rar.com/start.html?&L=0
- Minimum Hardware Specifications
  - Dedicated GPU: 8GB **(NVIDIA preferred as the chat bot training method utilizes NVIDIA CUDA to use the dedicated GPU for training and predicting)**
  - RAM: 4GB

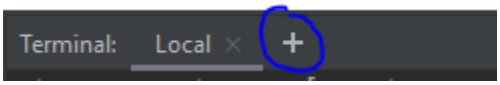
Procedure

- Extract the folder with the name “VoiceAssistant.zip” into desired directory.
- Open PyCharm (**Note: Jupyter Notebook will not work as it does not support terminal functionality. Other IDE’s were not tested but they must have a terminal to be able to execute commands for the CMD. Hence, PyCharm is recommended**)
- Open file or project and locate “CBTest” and then, press “OK”

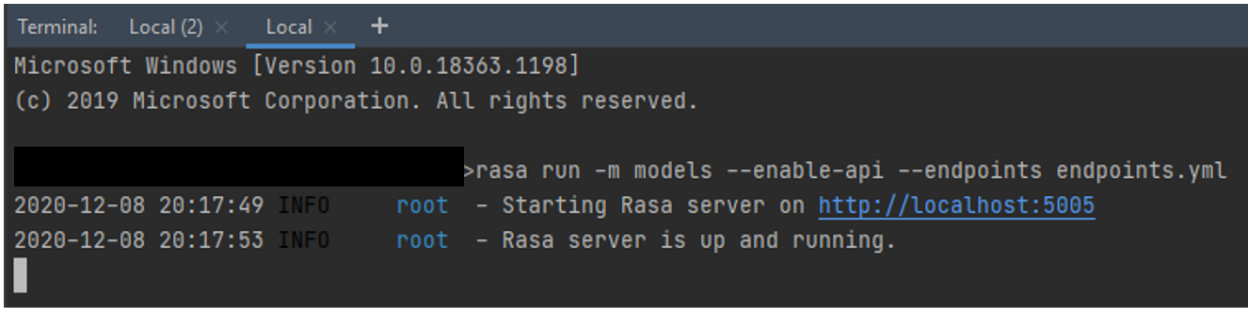


- Open two terminals at the bottom left of the IDE and run the following commands (might take a while):

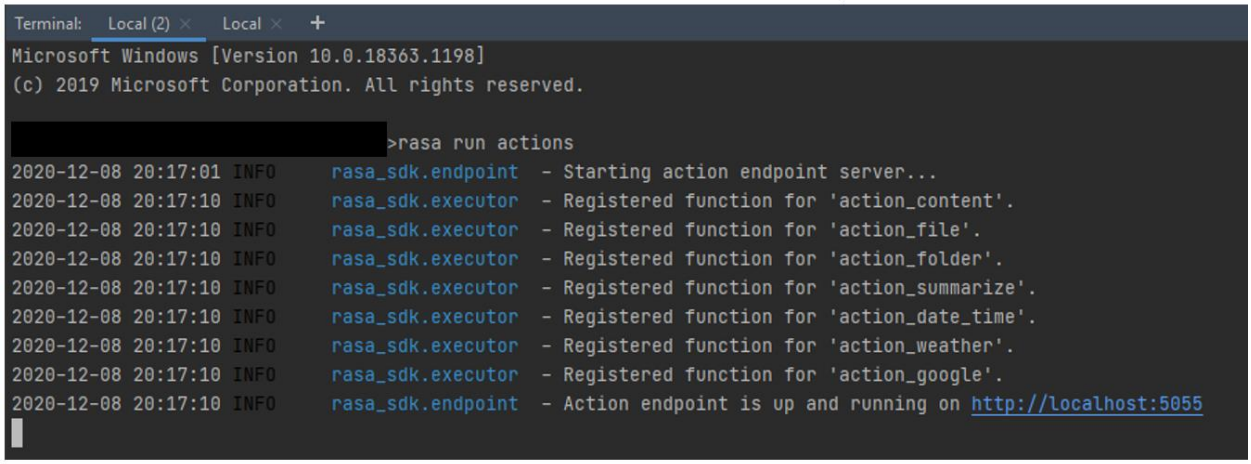
After clicking on the terminal open another terminal:



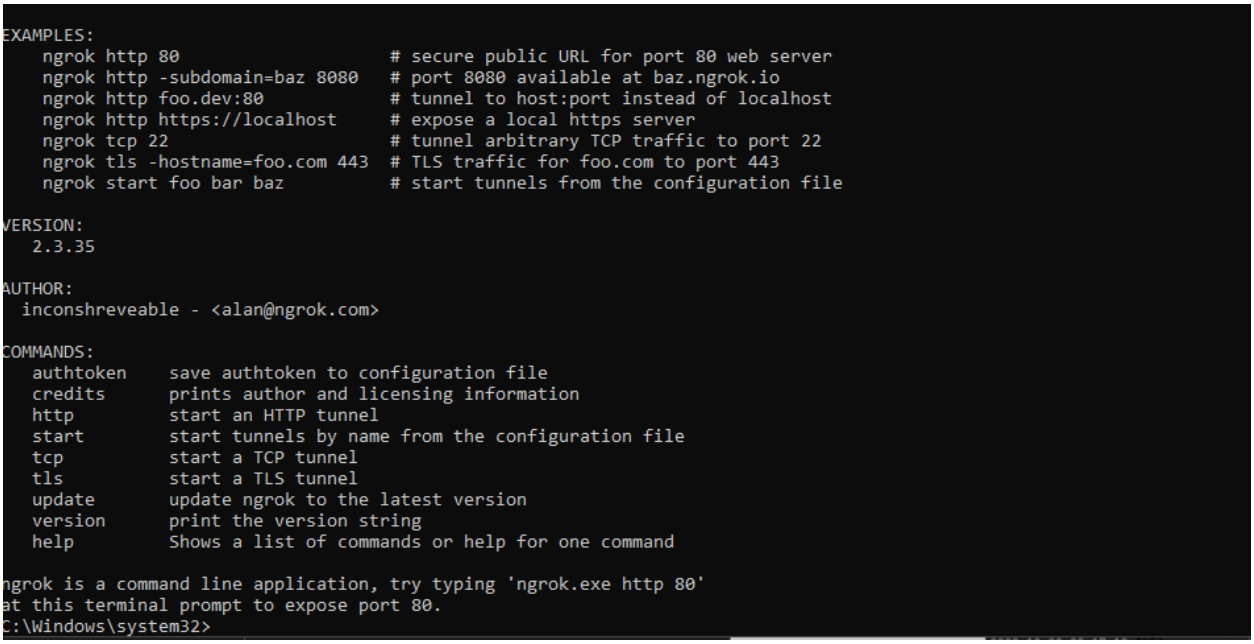
On the first terminal type: $rasa\ run - m\ models - -enable - api - -endpoints\ endpoints.yml$ and press enter.



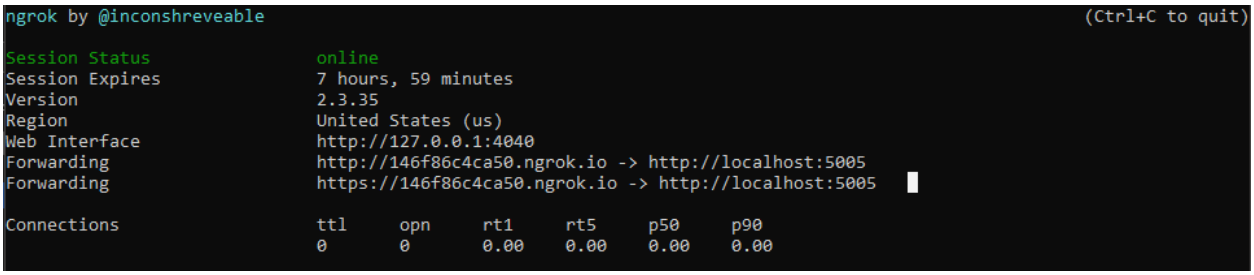
On the second terminal type: $rasa\ run\ actions$ and press enter.



- Open NGROK



- Type on NGROK: $ngrok\ http\ 5005$ (depending on which port is being used to host the bot locally. By default, its 5005) and press enter:



Copy any link either $http$ or $https$.

- Open Android studio project.
  - Navigate to “Lexi_chatbot.java” class.
  - Under the “sendMessage()” method, replace the link copied from NGROK with the previous link as shown in the image below:

```
public void sendMessage(View view) {
    UserMessage userMessage = null;
    MessageSender messageSender;
    String msg = editText.getText().toString();
    OkHttpClient okHttpClient = new OkHttpClient();
    Retrofit retrofit = new Retrofit.Builder()
            .baseUrl("https://ff5ec107e35e.ngrok.io/webhooks/rest/")
            .client(okHttpClient)
            .addConverterFactory(GsonConverterFactory.create())
            .build();
```

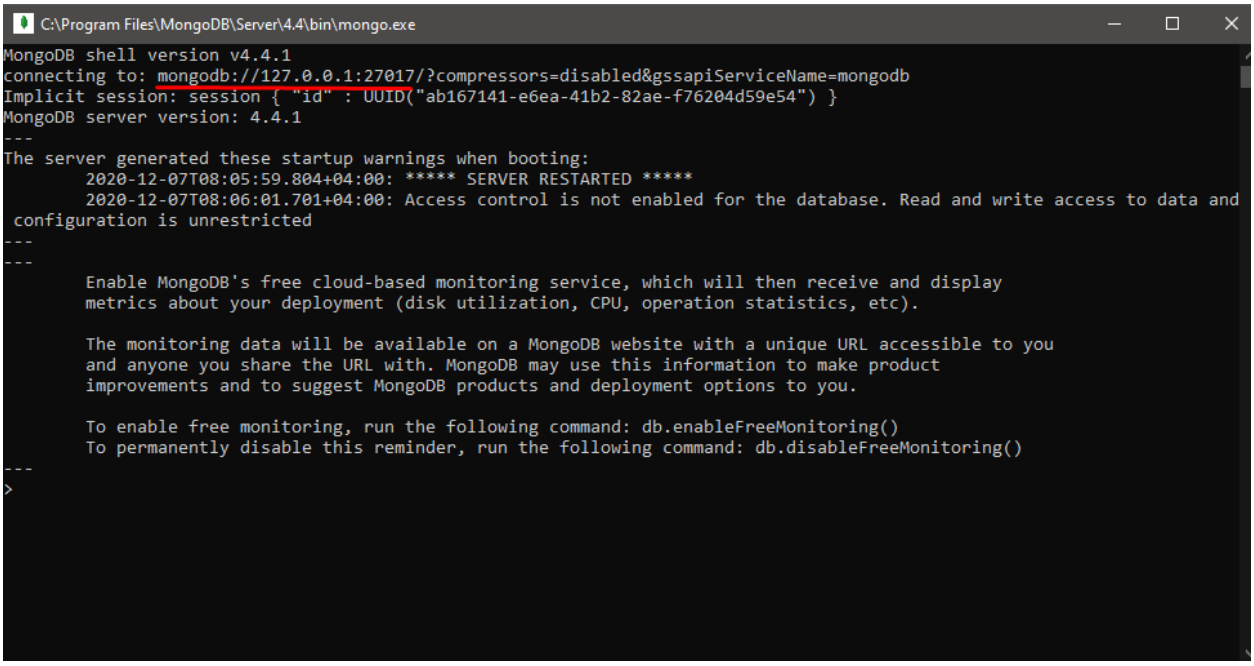
Note: After every 8 hours, NGROK will shut down and must restart and hence, from opening the NGROK step until the end, it must be done again as the server is being restarted every 8 hours. Alternatively, other methods could be used to host the bot permanently without having to change the link every 8 hours. However, they are paid.

FILE SYSTEM

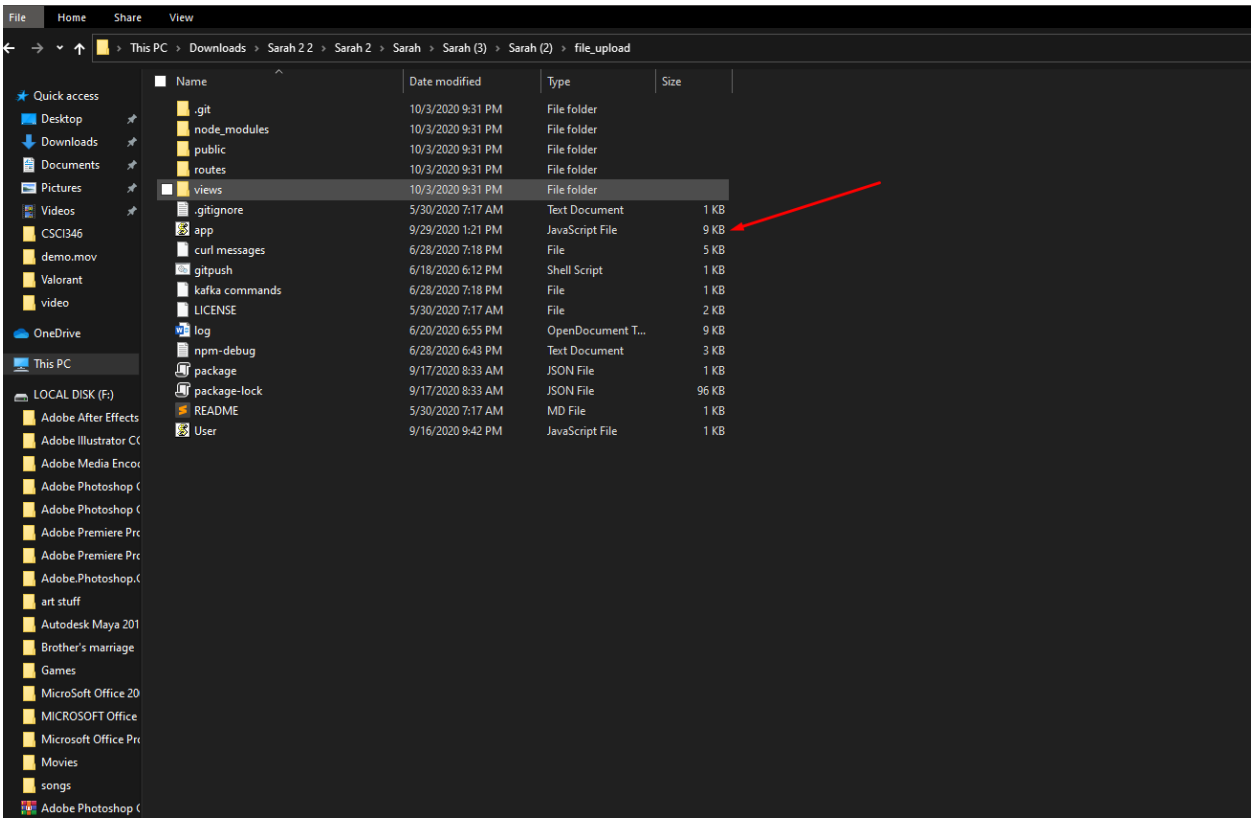
First you would need to download and install mongoDB on your machine if not already installed. Please follow this link to install it:

[How to Install MongoDB on Windows 10 - YouTube]

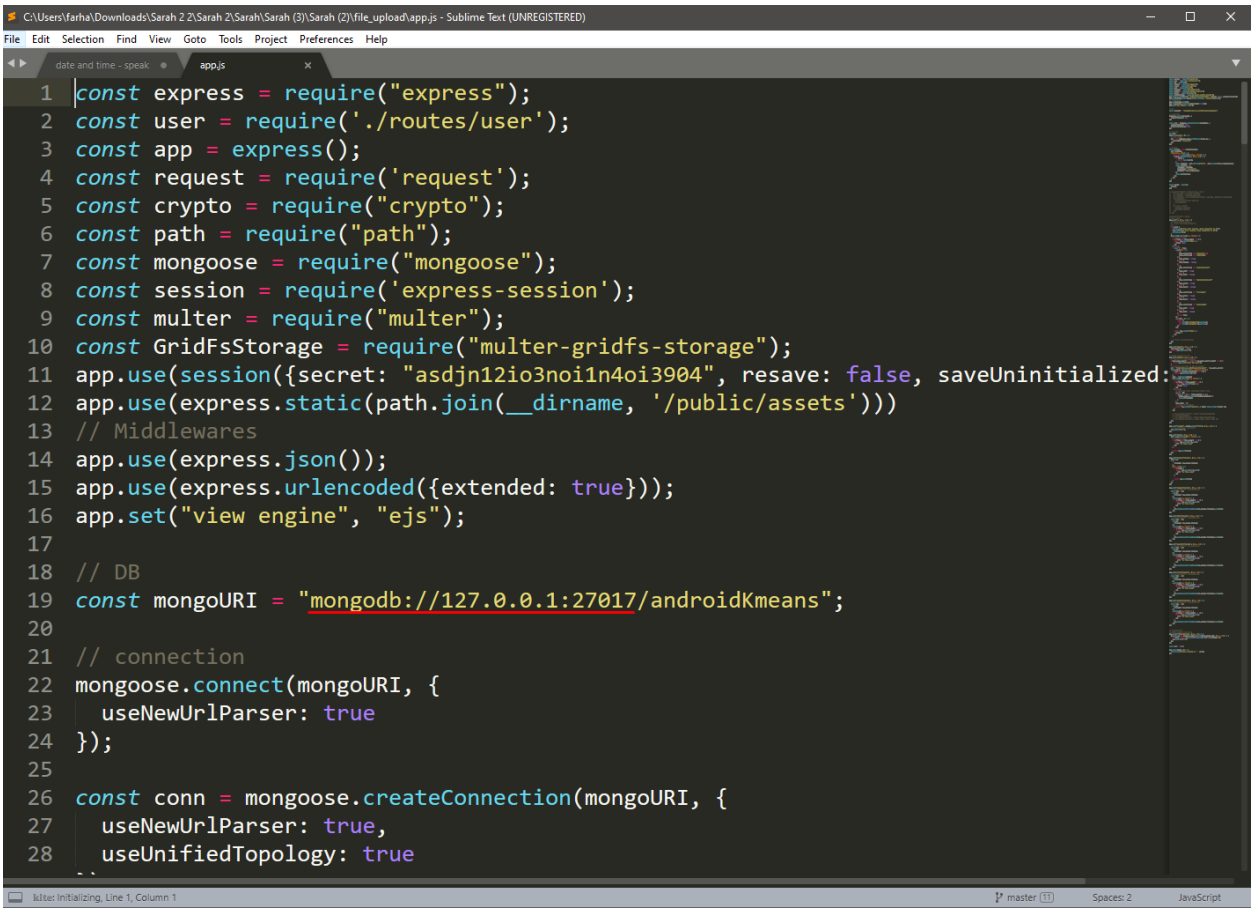
After which you then open the mongo shell and look for this link below:



You then take that link and paste it in the app.js file which can be found under the file upload folder as seen below:



Once you are within the app.js file paste it as highlighted below on line number 19:



If the link is the same as the screenshots above then obviously no changes are required to be made to the app.js file. After which you then open the command line and install the dependencies required for the project like so:

npm install -y

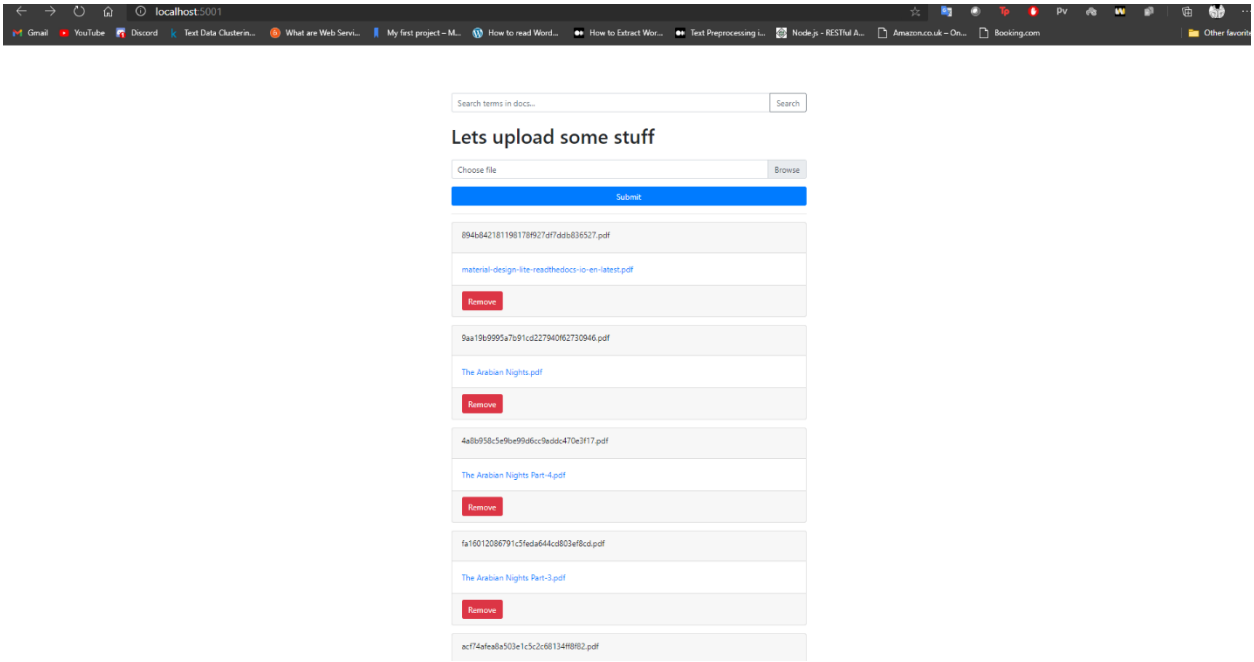
after which you then start the node.js server by typing the following commands:

npm start

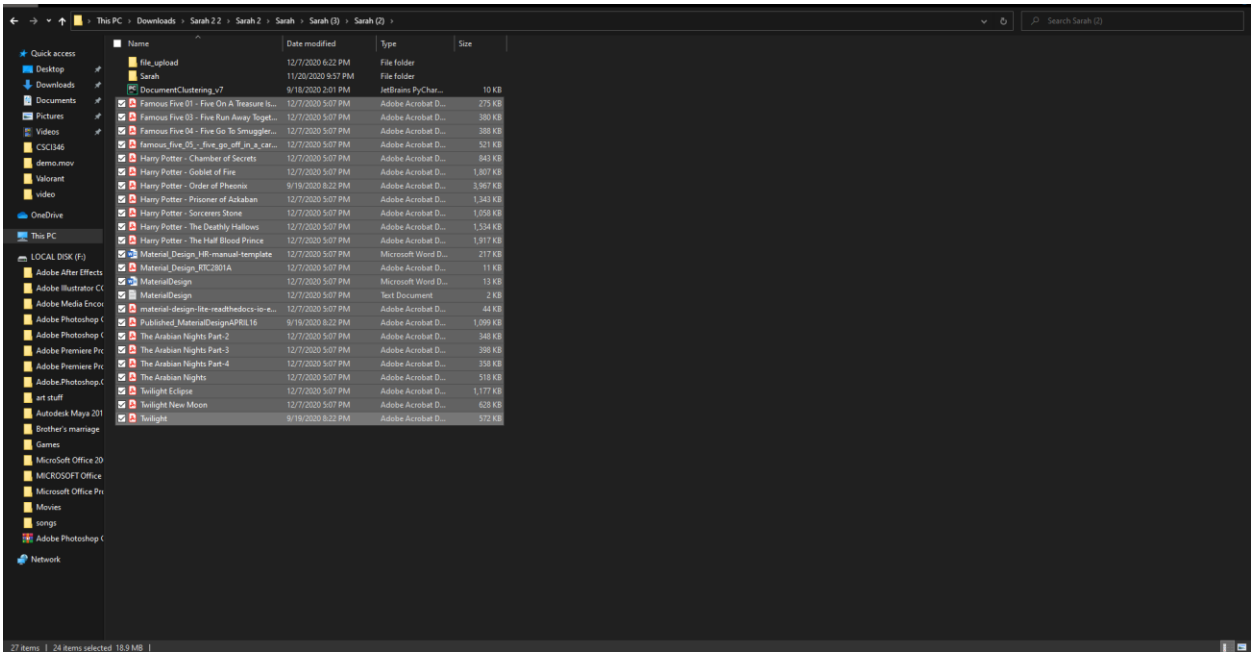
you can now open your browser and search:

http://localhost:5001/

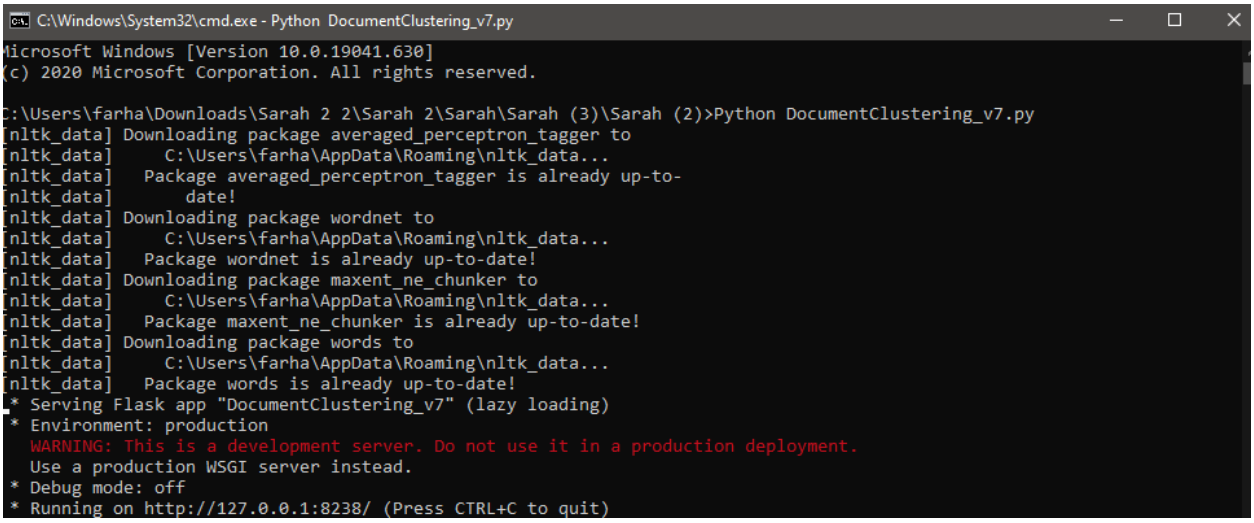
you will then be greeted with the following the interface:



You can now upload the highlighted documents present in this folder:



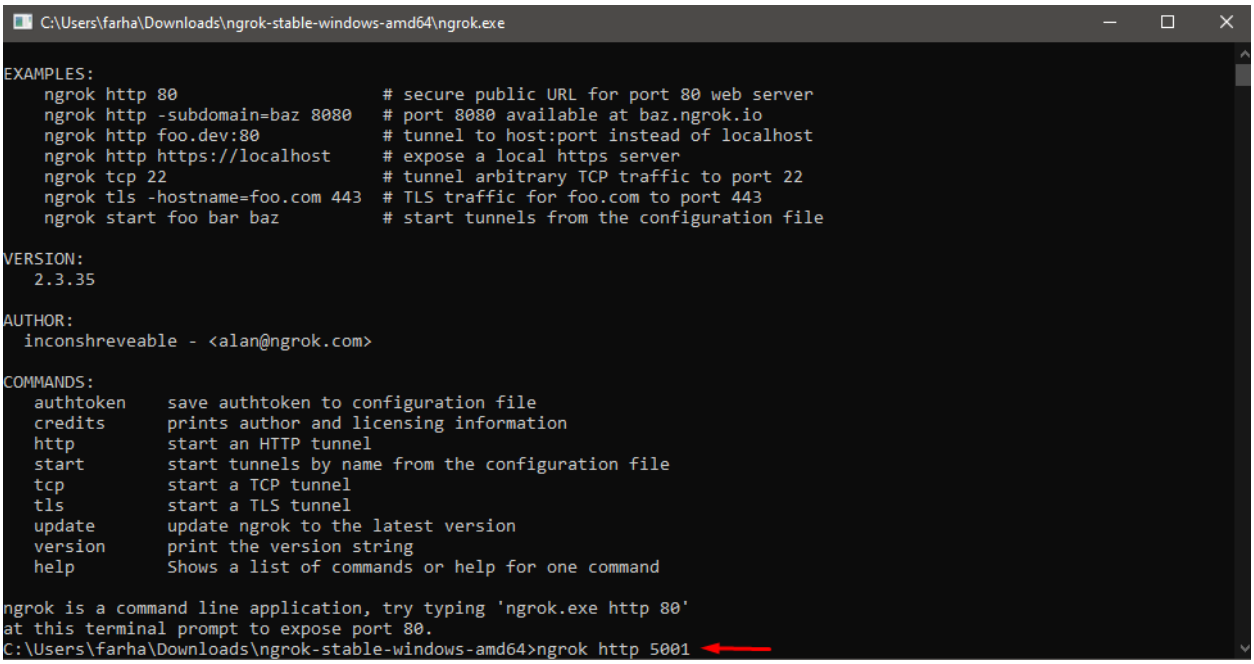
You can now run the document clustering code by opening the command line within the same directory and typing the following command and as a result after sometime your command line should look like this:

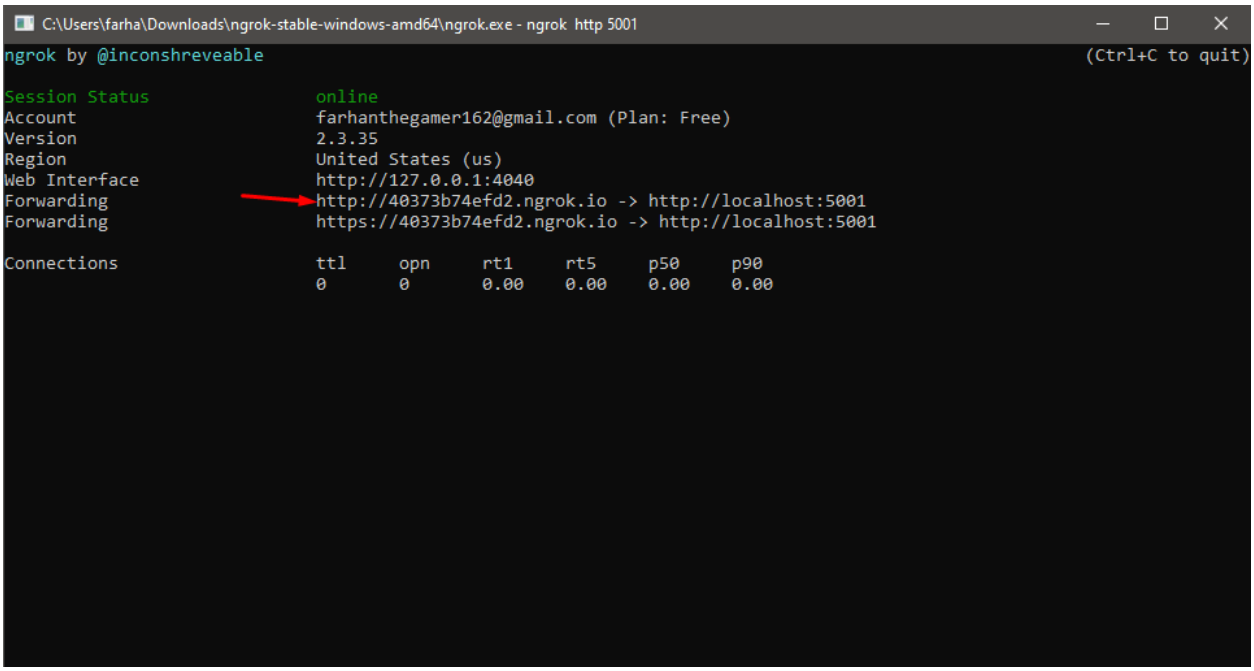


Once that is done setup ngrok using the following link:

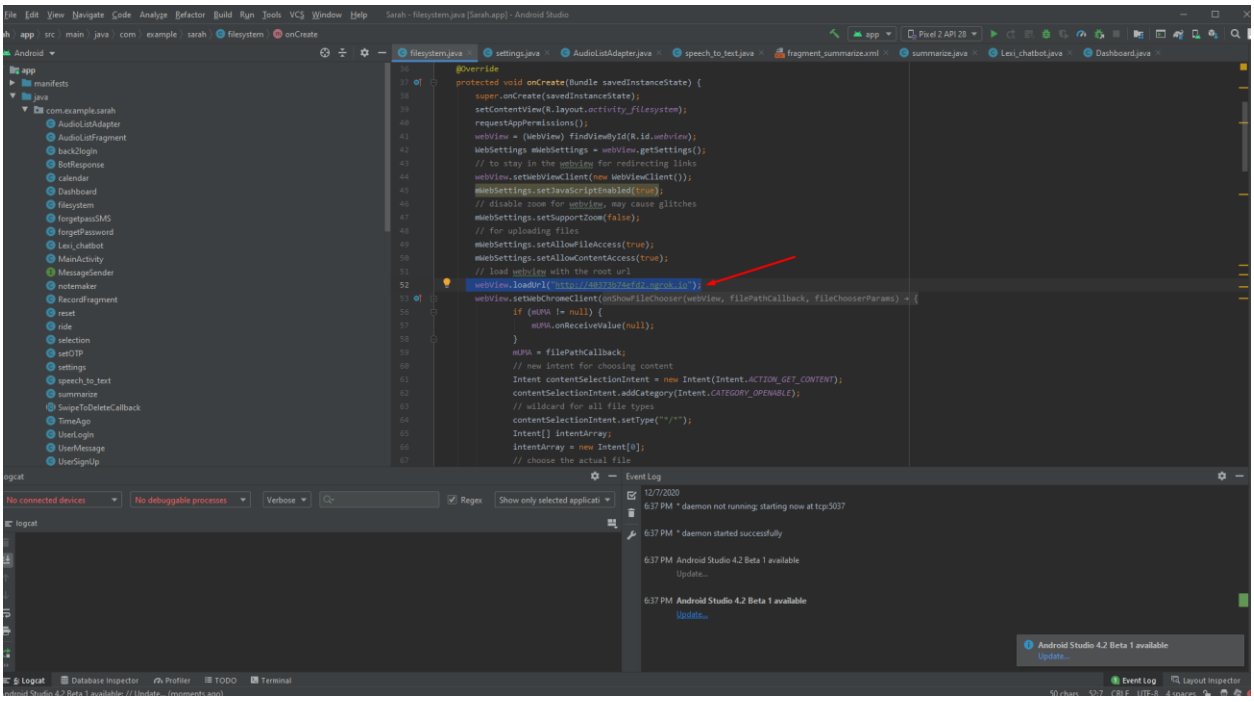
[Get a public URL to Localhost — The easy way | by Dilan Tharaka | Medium]

After which you can then open ngrok and type the following command like so:





After which you take the forwarding url shown in the above screenshot and then paste it on the android studio code under the filesystem.java file as shown below on line 52:



After setting everything up: you can then run the code on android studio.