Major Project Report

On

Ai ChatBot for Polytechnic College Counselling

Submitted in Partial Fulfillment for the Award

Of

Diploma in Computer Science Engineering
(2021-22)

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALYA BHOPAL (M.P.)

Submitted

By

Abhishek Dwivedi

Under the Supervision

of

Mr. Mahendra Gupta

GOVERNMENT POLYTECHNIC COLLEGE ANUPPUR

MP-484224

Certificate

This is to certify that project entitled **"Ai ChatBot for Polytechnic College Counselling "** is being submitted by **Mr. Abhishek Dwevidi** to the **Dept. of Computer Science and Engineering, Govt Polytechnic College Anuppur, M.P.-484224, India**, in the partial fulfillment of the requirements for the award of the diploma in **"Computer Science and Engineering"**. This work is carried out by himself in the Dept. of Computer Science and Engineering under the supervision of **Mr. Mahendra Gupta.** The matter personified in the project report has not been submitted for the award of any other degree or diploma.

Mr. Raju Singh Paraste (**Principal**)	Mr. Utkarsh Agrawal (**HoD**)	Mr. Mahendra Gupta (**Supervisor**)

Contents

DECLARATION

We hereby declare that the work which is being presented in the project report fulfillment of the requirement of the "Diploma in Computer Science" branch is an Authentic record of our work carried out under the guidance of "Mr. Mahendra Gupta(lecturer)". The work has been carried out at Govt. Polytechnic College Anuppur(M.P).

PROJECT ASSOCIATES:-

→ ABHISHEK DWIVEDI

ACKNOWLEDGMENT

A project like one involves many people and it would be complete without the mention of all those people whose guidance and encouragement helped in the successful completion of this project.

We heartily thank our faculty member of the Department of Computer Science Govt. Polytechnic College Anuppur for their efforts towards our project.

We would like to thank our project in-charge Mr. Mahendra Gupta who has been a great source of inspiration for us and without whose humble guidance the project was never shaped.

We are also indebted to our guide Mr. Mahendra Gupta for the Encouragement Guidance and Support.

We are also thankful to all the many people whose timely help our the paucity of space is restricting us from their name.

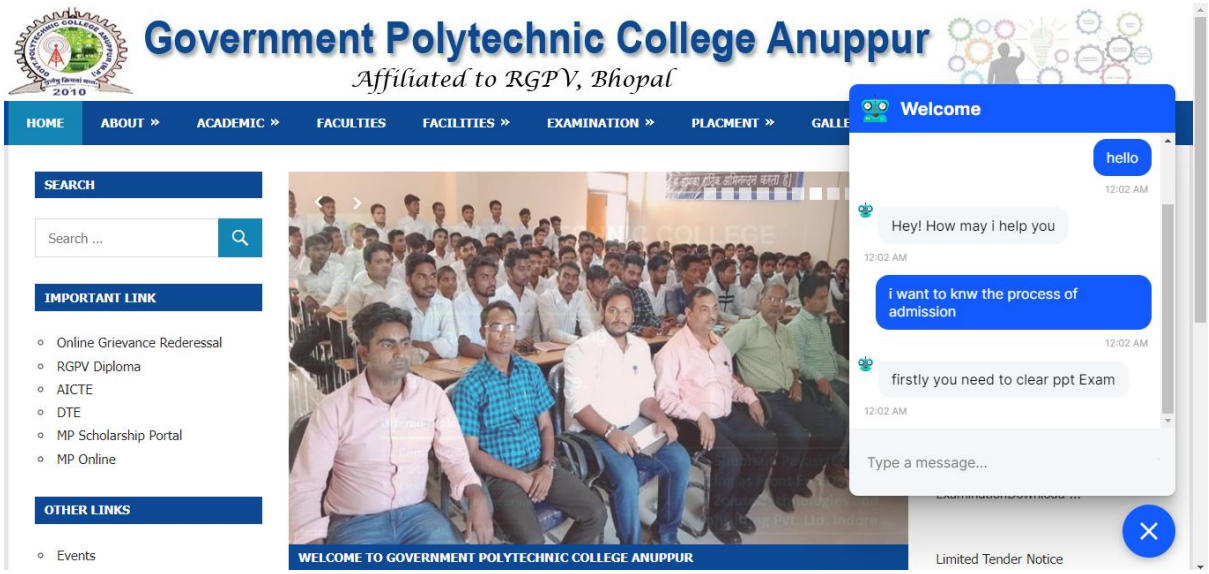
And finally, we also thank all my colleagues who were constant support during the whole project.

1. INTRODUCTION

I'm sure each of us would have interacted with a bot, sometimes without even realizing it! We are in the era of Conversational AI now. Every website uses a Chatbot to interact with the users and help them out. This has proven to reduce the time and resources to a great extent. At the same time, bots that keep sending" Sorry I did not get you " just irritate us. You need to ensure that the performance is satisfactory too.

Rasa open source provides an advanced and smooth way to build your chatbot that can provide satisfactory interaction.

We can put RASA Chatbot on your website and connect to the database after that RASA will manage the user coming to our website like this:-



And with this, we can also make a Bot for **WhatsApp** like this: -

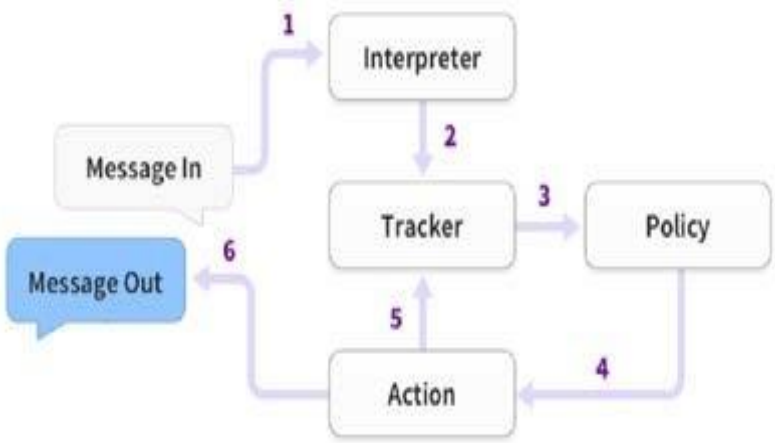
2. OBJECTIVE

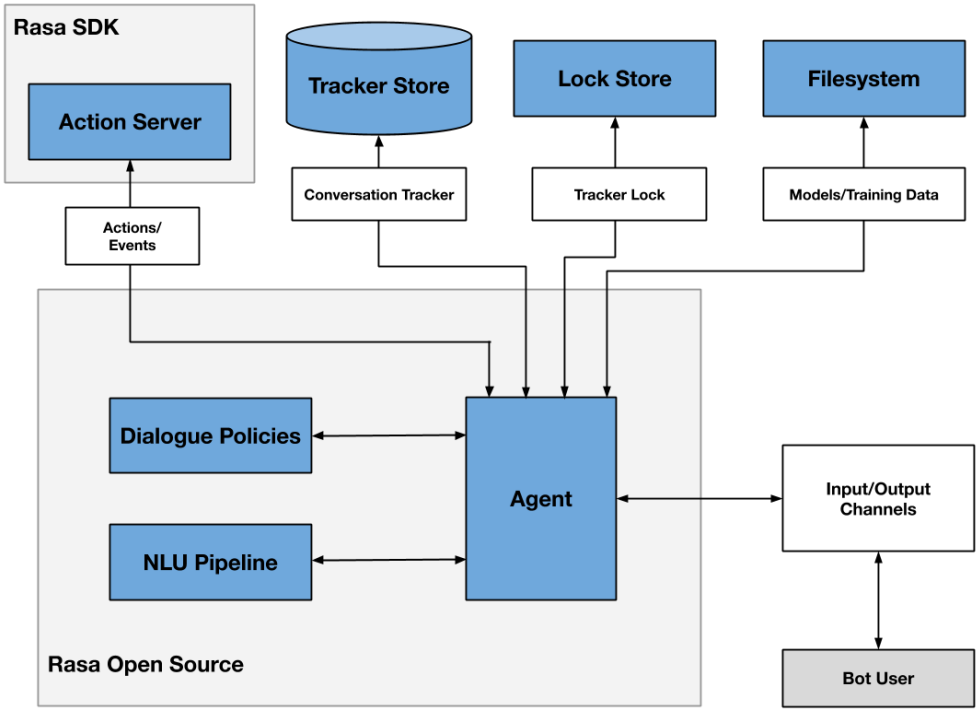
A chatbot can communicate with a real person by behaving like a human.

The objective of the Rasa chatbot is that it can help the college. In most cases, the student does not know how to take admission. How many branches in this college and there are many other problems. With the help of this chatbot, we can solve those problems.

This chatbot can solve all the problems related to the college like can tell the process of admission. It can tell how many branches are there in the college. It can tell about the staff of the college and it can do many other things related to the college.

3. FLOW CHART


1
Interpreter
2
Message In
Tracker
3
Policy
6
Message Out
5
4
Action


Rasa SDK
Action Server
Tracker Store
Lock Store
Filesystem
Conversation Tracker
Tracker Lock
Models/Training Data
Actions/
Events
Dialogue Policies
Agent
Input/Output
Channels
NLU Pipeline
Rasa Open Source
Bot User

4. HARDWARE REQUIREMENT

PROCESSOR i3

RAM 8GB

STORAGE 512 GB

5. SOFTWARE REQUIREMENT

IDE Vs Code

LANGUAGE Python

FRAMEWORK Rasa FrameWork

5.1 VS CODE

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS, and Linux. It comes with built-in support for JavaScript, TypeScript, and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity)

5.2 PYTHON

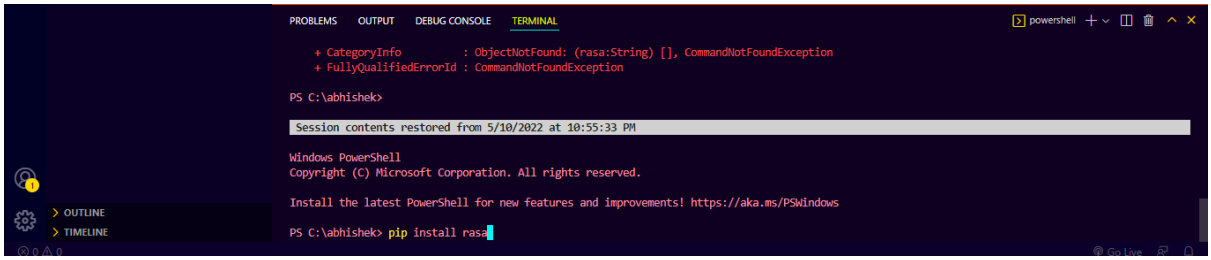
Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components.

5.3 RASA FRAMEWORK

Rasa is a framework for developing AI-powered, industrial-grade chatbots. It's incredibly powerful and is used by developers worldwide to create chatbots and contextual assistants. In this project, we are going to understand some of the most important basic aspects of the Rasa framework and chatbot development.

5.4 INSTALLING PROCESS OF RASA:-

To install rasa you just have to type a simple command ‘ pip install rasa’.

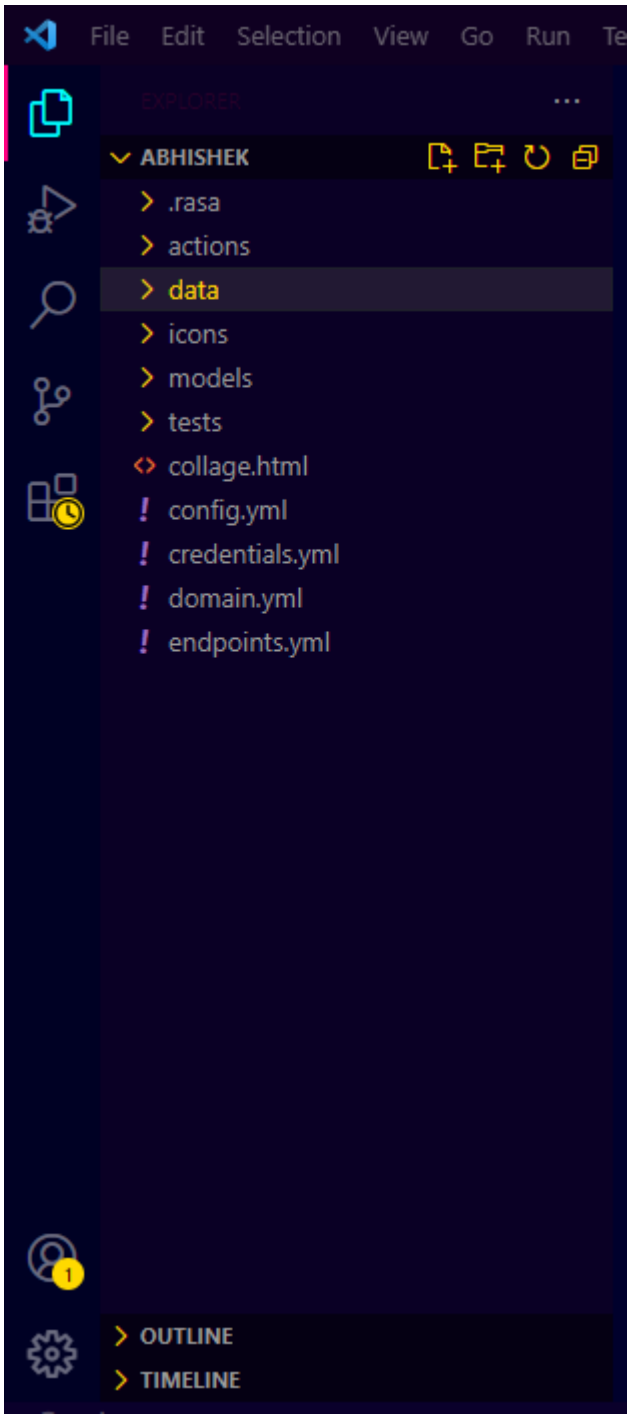


After that ,your rasa framework will start installing

6. WORKING

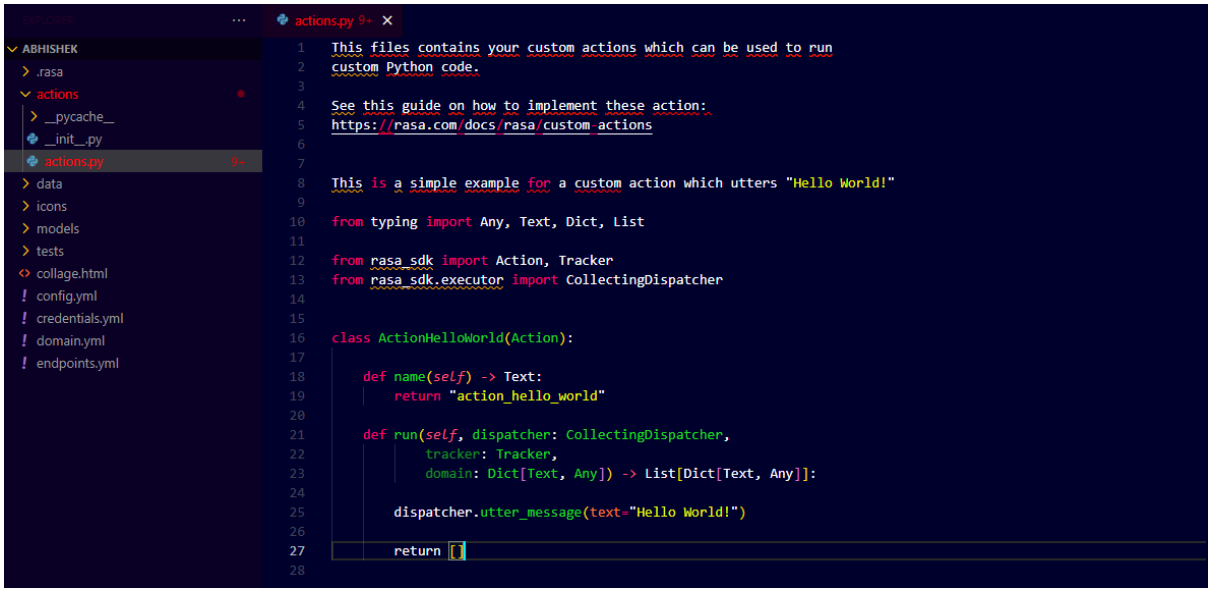
When we install Rasa Framework, we get some preinstalled files and folders these are:-

Action, Data, Model, Domain.yml, Endpoint.yml, Config.yml, and credentials.yml



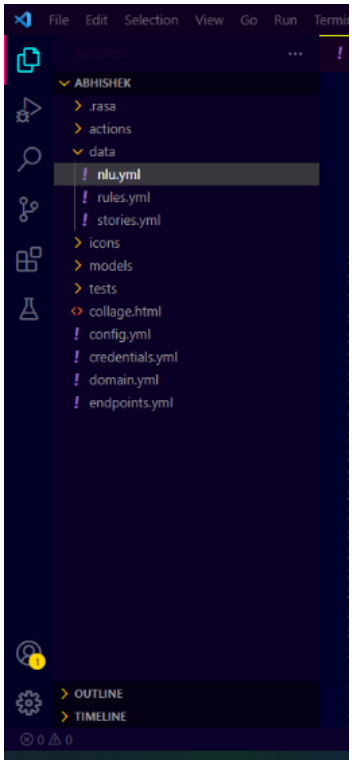
6.1. Actions

Action is a folder where we get the Action.py file it's a Python Which we use to do things like fetch data from API, to fetch data from the database, etc.



6.2. DATA:-

Data is a folder inside which we get some files like Nlu.yml,Rule.yml,Stories.yml

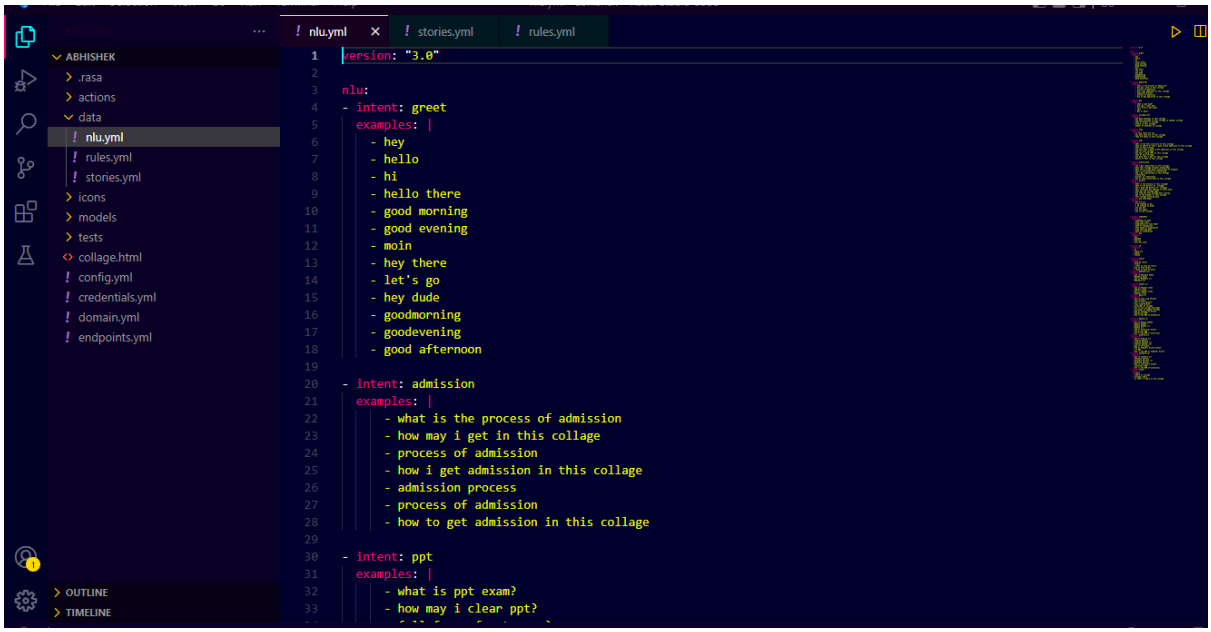


6.3. NLU.yml:-

NLU a file where we train your bot here we specify **Intent** and **Examples**

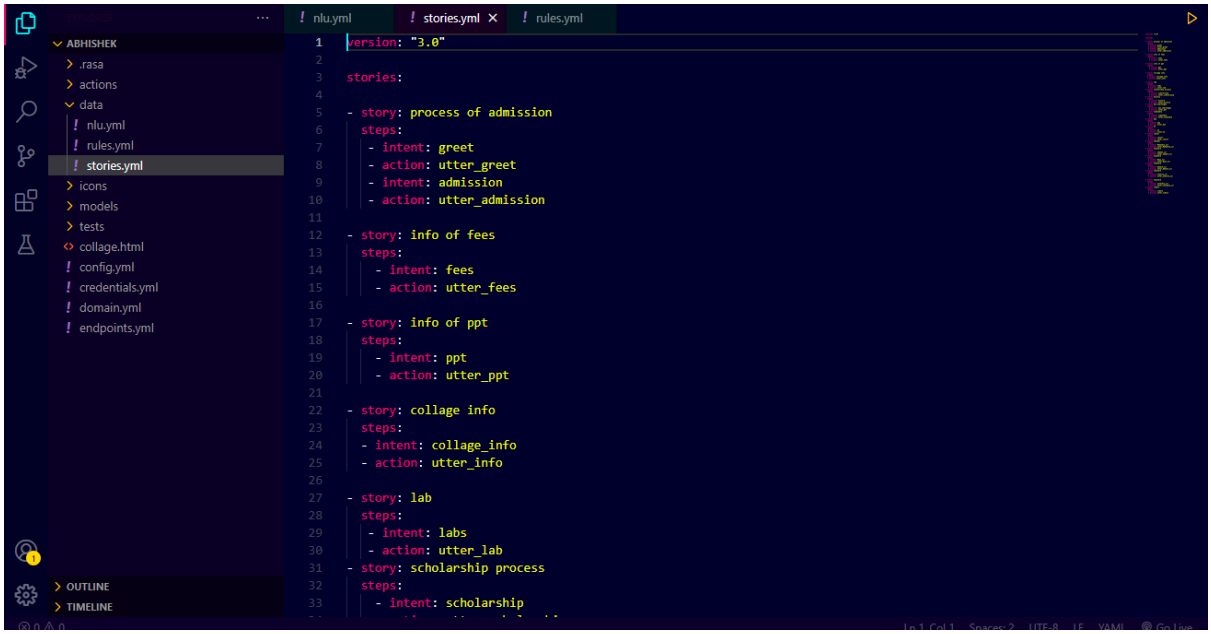
Intent:- If the intent is explained in simple words, then the intent is also called the subject, which we are talking about like **College Information** it's intent in this we ask things related to college like the location of the college, how many branches in college.

<u>Example:-</u> Example is that in which we tell the bot how the user can ask the question to you like Branches in this collage, How many Branches in this collage, Is there C.s Branch in this collage



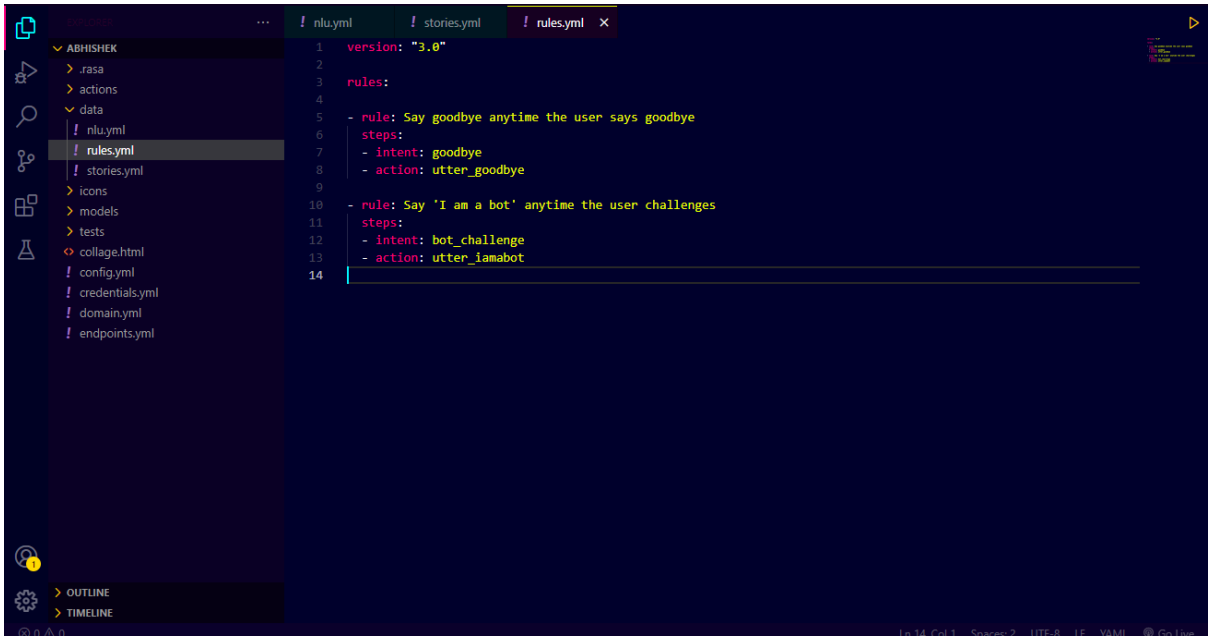
6.4. STORIES.yml:-

Stories.yml is a file that is found inside the Data folder, as we have seen in **nlu.yml** where we specify the **intent,** in **Stories.yml** we specify **an action,** that if this intent occurs then perform this action



6.5. RULE.yml:-

Like Stories.yml Rule.yml is also found in the Data folder, in **Rule.yml** we define some Rules for a bot like whenever a user say Bye, **you immediately exit and says bye to him,** and also whenever the user asked who are you, **you should immediately tell them that I am a bot**



6.6. MODEl:-

Model is a folder where our Rasa model Store Whenever we train a model, it becomes stored in this folder

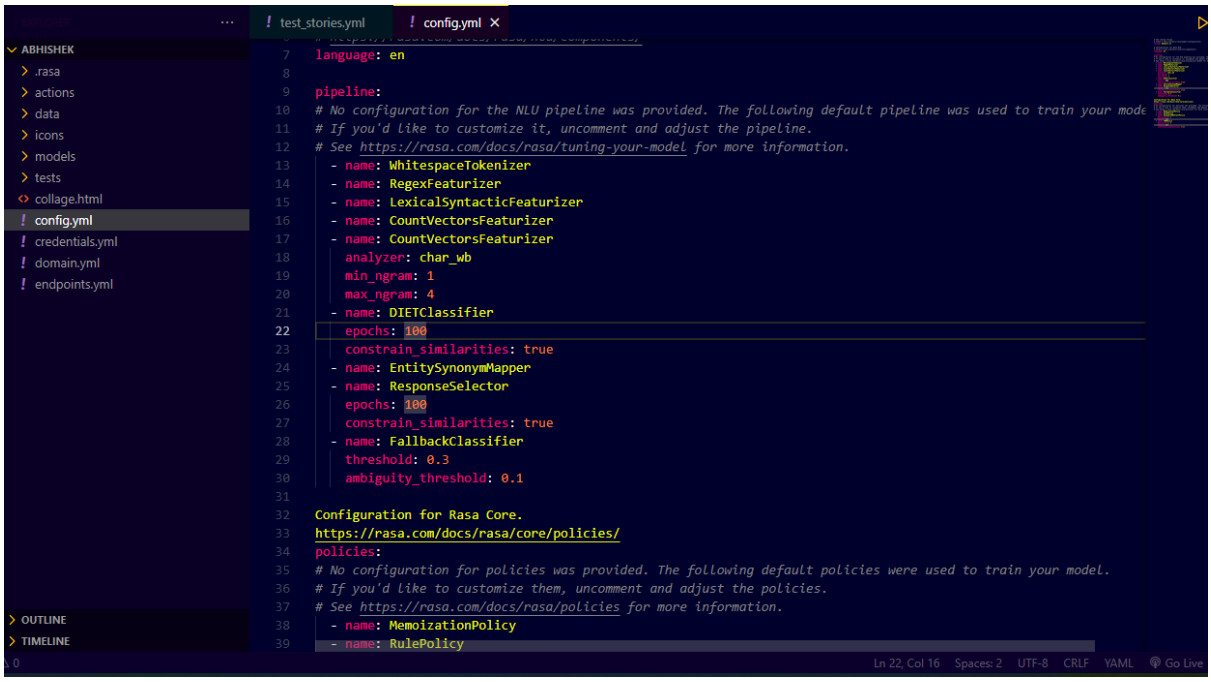
Process of Traning Model:-

Whenever we make a chatbot we need to train them it is similar to how we compile a program by creating it

To train the bot we need to use the command “**Rasa Train”**

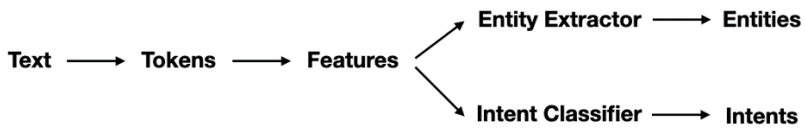
6.7. CONFIG.yml:-

If we talk about the most important thing in the rasa chatbot, it's a config.yml because it has a **Pipeline**



6.7.1 PIPELINE:-

The NLU pipeline defines the processing steps that convert unstructured user messages into intents and entities. It consists of a series of components, which can be configured and customized by developers.



There are different types of components that you can expect to find in a pipeline. The main ones are:

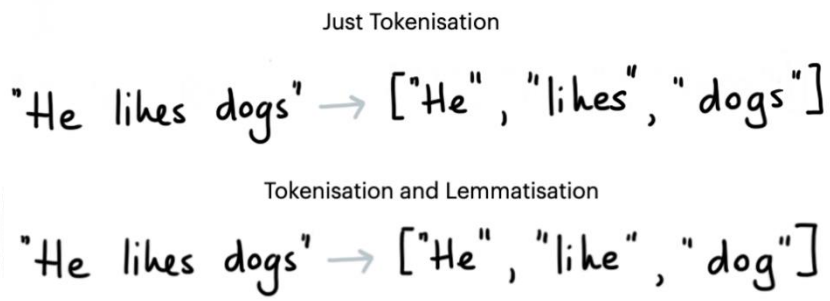
1) Tokenizers
2) Features
3) Intent Classifiers
4) Entity Extractors

6.7.2 Tokenizers:-

The first step is to split an utterance into smaller chunks of text, known as tokens. This must happen before text is featured for machine learning, which is why you'll usually have a tokenizer listed first at the start of a pipeline.

"Hi, my name is Vincent." ⟶ ["Hi", "my", "name", "is", "Vincent"]

Details on Tokenizers.



Some tokenizers also add extra information to the tokens. For example, spaCy is able to also generates lemmas of the tokens which can later be used by the CountVectorizer.

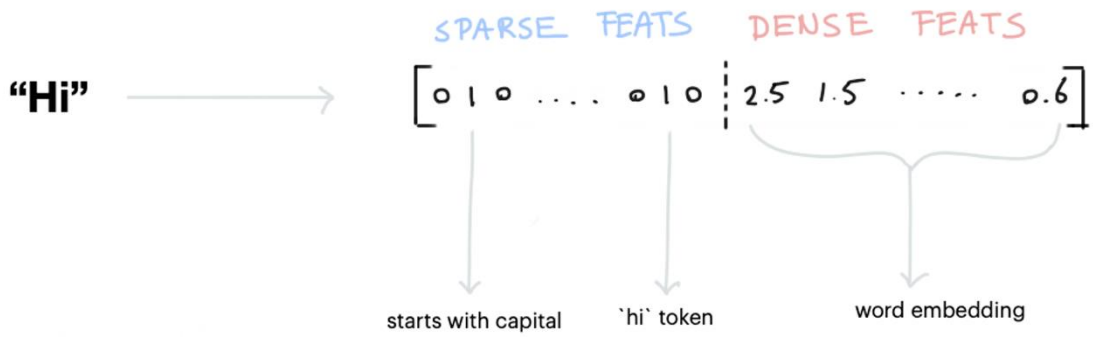
The tokenizer splits each individual word in the utterance into a separate token, and commonly the output of the tokenizer is a list of words. We might also get separate tokens for punctuation depending on the tokenizer and the settings that we pass through.

For English, we usually use the WhiteSpaceTokenizer but for non-English it can be common to pick other ones. SpaCy is a good choice for non-English European languages but Rasa also supports Jieba for Chinese.

Note that tokenizers don't change the underlying text, they only separate text into tokens. That means, for example, that capitalisation remains untouched. It might be that you'd like to only encode the lower case text for your pipeline, but adding this kind of information is the job of a featurizer, which we'll discuss next.

6.7.3 FEATURES:-

Features generate numeric features for machine learning models. The diagram below shows how the word "Hi" might be encoded



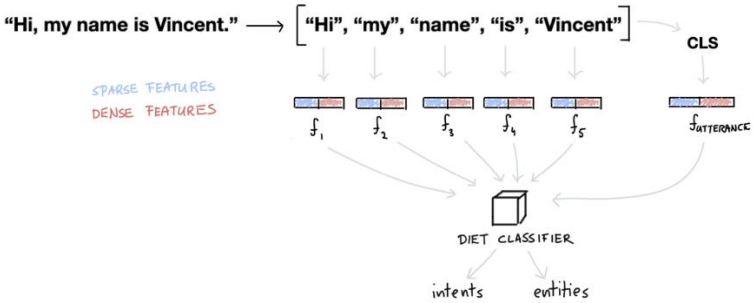
There are two types of features:

Sparse Features: usually generated by a CountVectorizer. Note that these counts may represent subwords as well. We also have a LexicalSyntacticFeaturizer that generates window-based features useful for entity recognition. When combined with spaCy, the LecticalSyntacticFeaturizer can be configured to also include part of speech features.

Dense Features: these consist of many pre-trained embeddings. Commonly from SpaCyFeaturizers or hugging face via LanguageModelFeaturizers. If you want these to work, you should also include an appropriate tokenizer in your pipeline. More details are in the documentation.

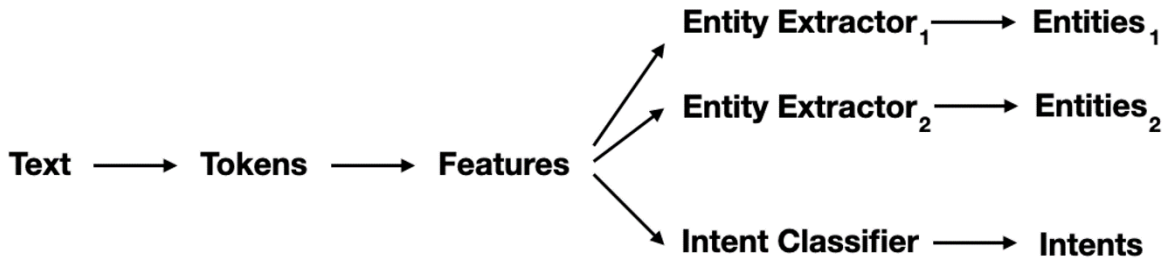
6.7.4 Intent Classifiers:-

Once we've generated features for all of the tokens and the entire sentence, we can pass it to an intent classification model. We recommend using Rasa's DIET model which can handle both intent classification as well as entity extraction. It is the token- as well as sentence features



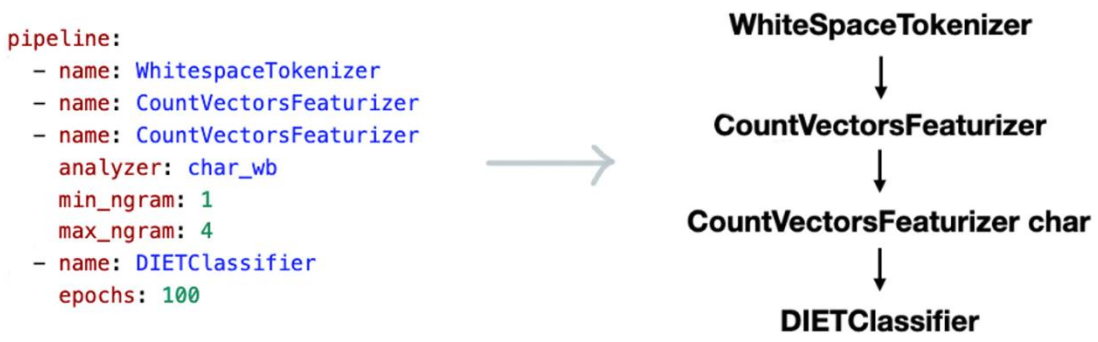
6.7.5 Entity Extraction:-

Even though DIET is capable of learning how to detect entities, we don't necessarily recommend using it for every type of entity out there. For example, entities that follow a structured pattern, like phone numbers, don't need an algorithm to detect them. You can just handle it with a RegexEntityExtractor instead.
This is why it's common to have more than one type of entity extractor in the pipeline



Interaction: Message Passing:-

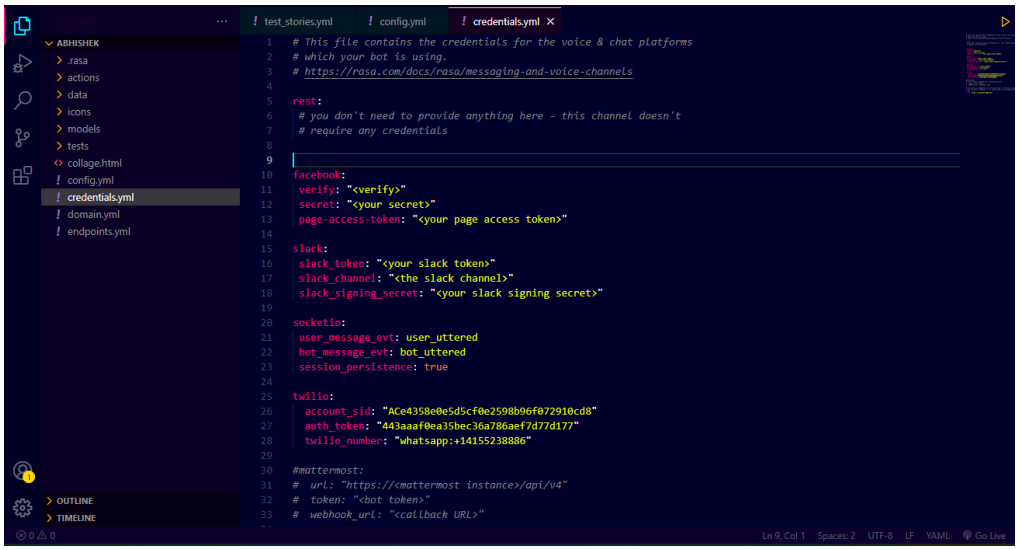
As you can imagine, the components in a Rasa pipeline depend on each other. So you might be wondering how they interact.



7. credentials.yml:-

Here things are related to server like in which port rasa chatbot will run,we need to specify port in which chatbot run like:- **Port:5005**

Here also we write program /code to link the website,Whatspps,Teligram

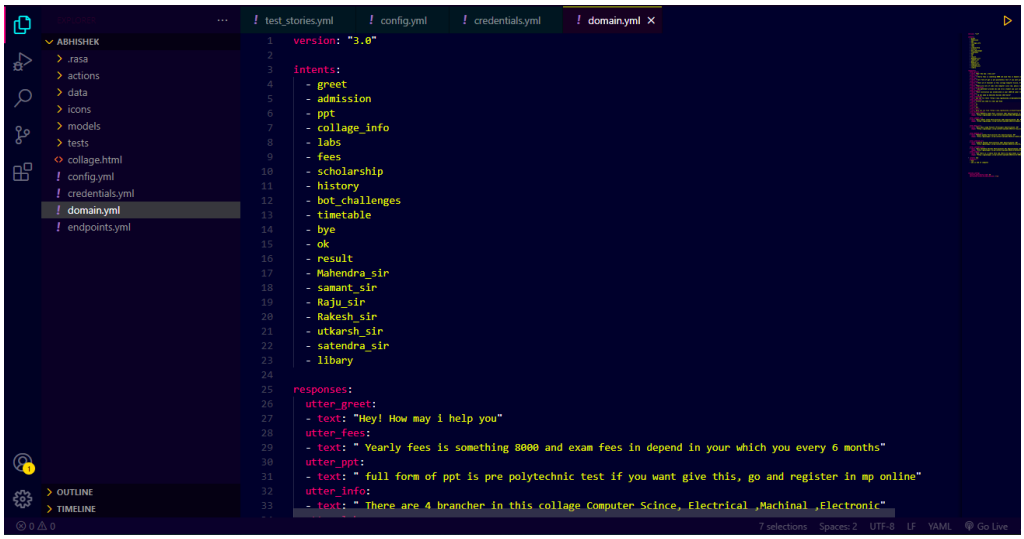


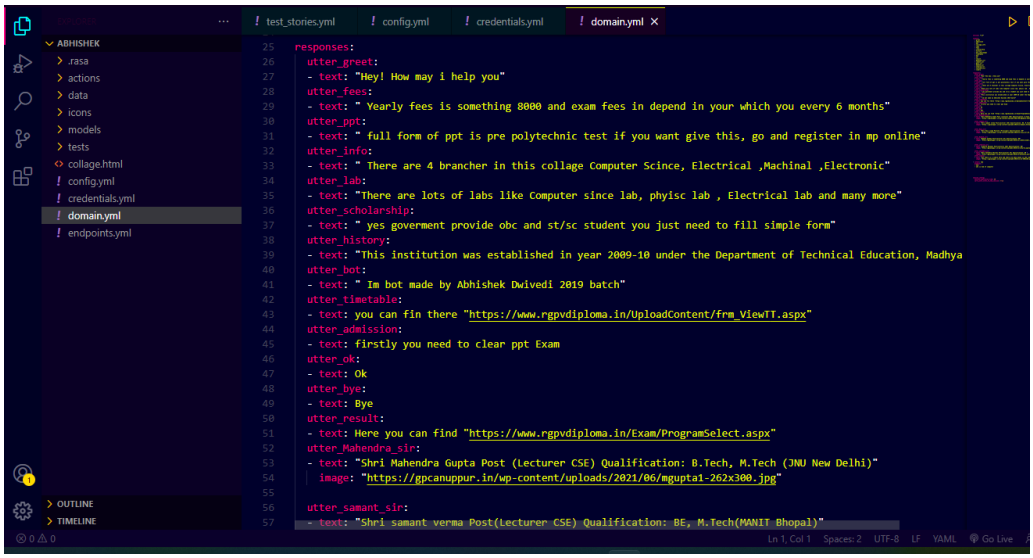
As you can see in the above image that there is **Facebook, Tillow** is connected to all these websites,To put chatbot in a **Website,Account,Server** we need to specify some things like **Account**, **Number,sever name , Token**

8. Domain.yml:-

Domain.yml is a very importanat file. Eariler you saw in stories.yml that we used to classify actions, These are actions are written here

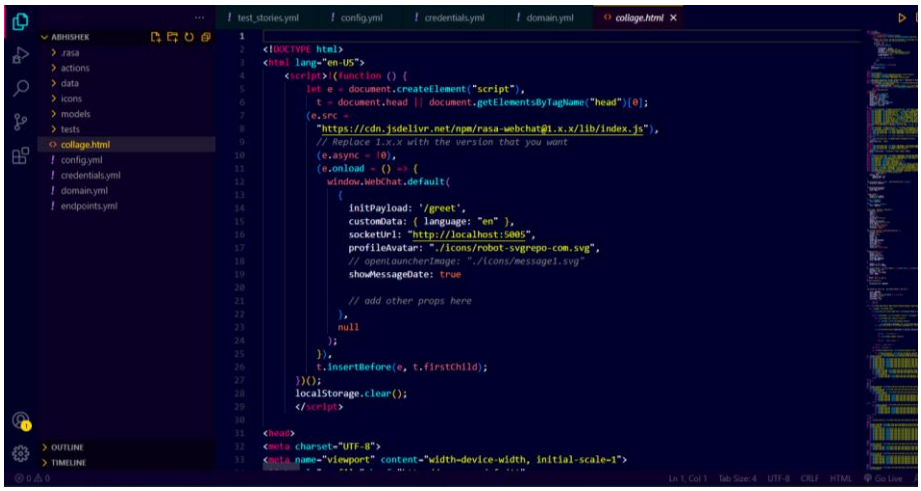
Here we have to write intent which we keep writing in nlu.yml. After that we have to write the actions for intent





9. Process to Deploy Chatbot in Website:-

To Deploy Rasa Chatbot in your website, you have to write some code of rasa chatbot in your **HTML** content .It is written with the help of JavaScript



In this code we need to specify Socket Url is where your website is running if your website is running on your Server you need to specify server Address / Domain name. If your website is not running in any server then simply writte **Localhost**

Also in this code there is a **Payload,** you can understand payload in such a way that whenther the bot will say something automatically, or the user sends a message to him, then he will response

I specify Greeting intent in **Payload** in greeting intent there are some message like hii,hello,how are, goodmorning its means when ever user truns chatbot ,bot will message it self like:-

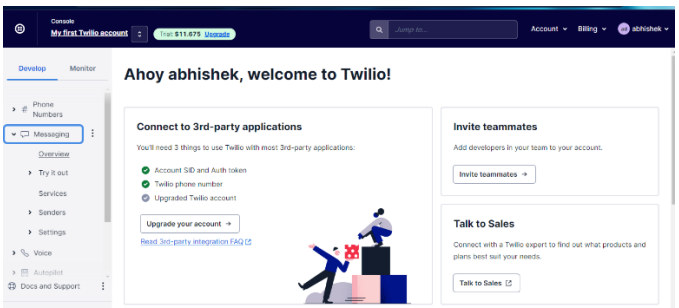
10. Process to Integrate into Whatsapp using rasa:-

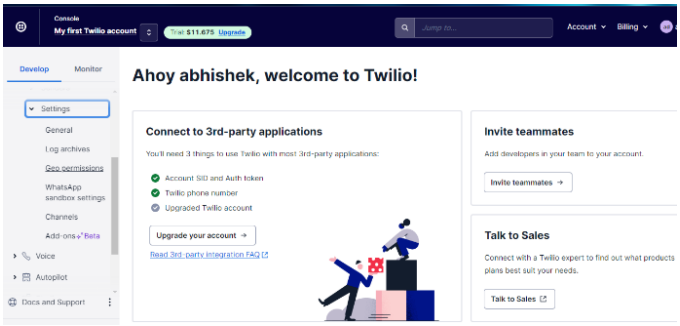
If you don not have server then you need to use Thrid party sever and ng rok software , ng rock software are software that puts local host on internet

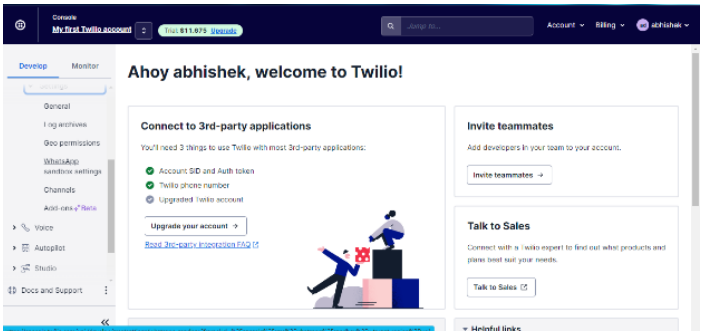
10.1 Stpes to make Whats App Chat bot:-

Step 1:- Create Account in Twilio, Twilio is a sever and also provedive whatapp bussiness Api

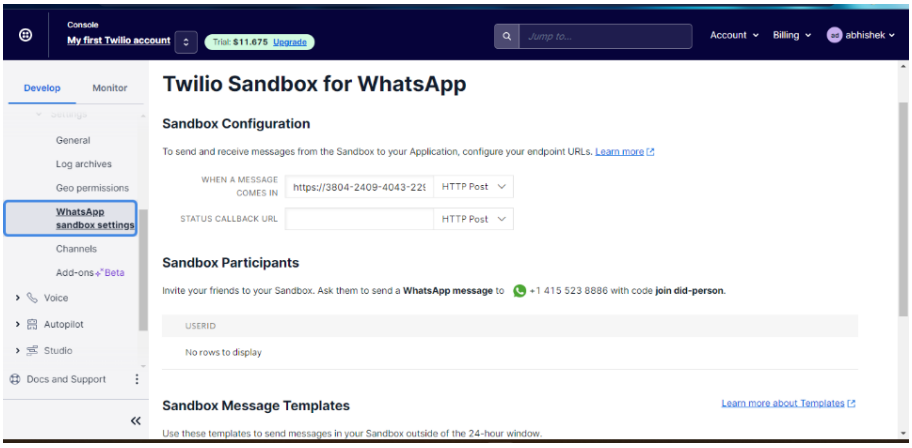
Step2:- After ceating account , you have go to the **Message Section** then in the **Settings section** the **Whatspp sand Box settings**







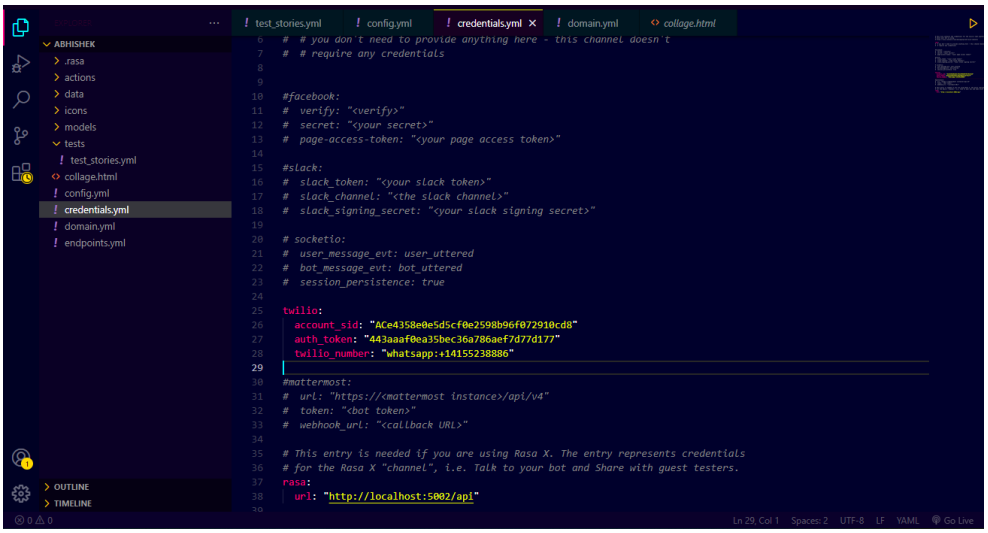
Sept4:- When you reach here , you see like this Inter Face



Here you can see a section **When a Message comes in,** here you have to enter the server link where your chat bot is running , because your botis running in local host then you need help of ng ng rok to put localhost in the internet

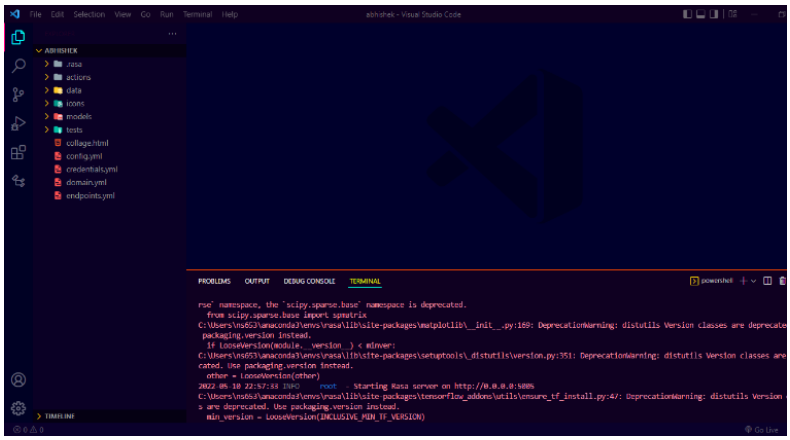
Step5:- But before doing all this, you have to tell rasa chatbot that on which number it will have to be activated or not and which server it will run

Earlier I told you about **credential.yml** where things realted to server are written here you have to **specify SID**, **Authorize Token**, and **Tiwlio number** if you do not have Whats App bussines api, if you have then Simpely past it

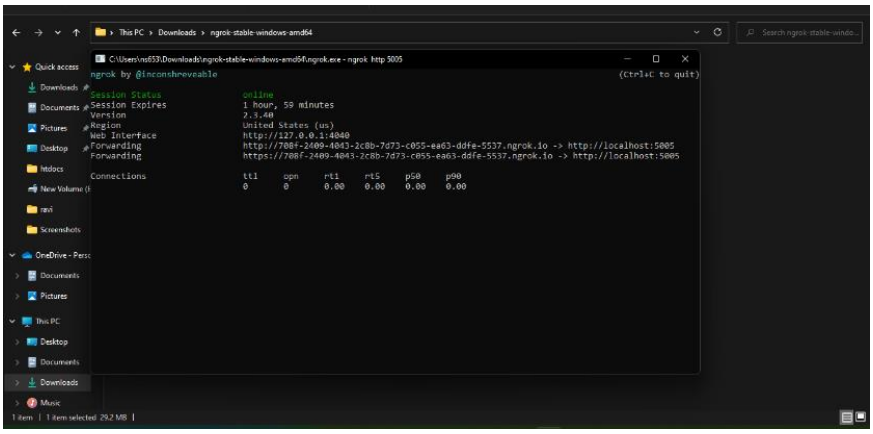


Step6:- After doing all this you simply have to train your rasa model one more time so that he know every thing

Step7:- After doing all this, you just have to type a simple command ,so that your rasa chat bot starts running like this



Step8:- After that you have to put port in which rasa chat bot is running like this



Step9:- As you can see , here after specifying the port we get link we have to put this link in Tiwlio

After that we need to type small command then after our bot start given response like that:-

11. REFERENCES: -

[1]For web site https://youtu.be/eJMT2FovZsM

[2]For whatapp chatbot https://youtu.be/K7boxP8Q50M

[3] Pipeline https://rasa.com/blog/intents-entities-understanding-the-rasa-nlu-pipeline/

[4] What is Rasa https://rasa.com/