

MINI PROJECT REPORT

on

KRISHNA: The Guidance Chat-bot

Submitted By

Mr. Ritesh Y Sharma

Mr. V Devang

Mr. Shreyash Limkar

Mr. Deepak Rahangdale

*Submitted in partial fulfillment of the requirements
of
Mini Project in third year of Bachelor of Engineering*

Guided by-

Dr. Ganesh Regulwar



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

**S.B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT &
RESEARCH, NAGPUR.**

2020-2021

© S.B, J.I.T.M.R Nagpur 2021



**S. B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT
& RESEARCH, NAGPUR.**

(An Autonomous Institute, Affiliated to RTMNU, Nagpur)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Vision: To become a center for quality education in the field of Computer Science & Engineering and to create competent professionals.



Institute Vision:

Emerge as a leading Institute for developing competent and creative Professionals.

Institute Mission:

- Providing Quality Infrastructure and experienced faculty for academic excellence.
- Inculcating skills, knowledge and opportunities for competency and creativity.
- Aligning with Industries for knowledge sharing, research and development.

Department Vision:

To become a center for quality education in the field of computer science & engineering and to create competent professionals.

Department Mission:

- To provide academic ambience and latest software tools to prepare competent Software Engineers with strong theoretical and practical knowledge.
- To foster professionalism and strong work ethics in students for the betterment of Society
- To provide adequate infrastructure as well as experienced & skilled faculty members.
- To encourage the spirit of entrepreneurship and adaptability in our students in view of the ever-changing scenario of the Software Industry.

Program Educational Objectives (PEO's)

- Have analytical, design and implementation skills, to innovate, design and develop software products and systems.
- Have strong work ethics and professionalism, reflected through communication skills, leadership, teamwork, and sense of responsibility towards the society.
- Be successful professionals through lifelong learning with allied objectives of higher education or research.

**S.B. JAIN INSTITUTE OF TECHNOLOGY MANAGEMENT
AND RESEARCH, NAGPUR**

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SESSION 2020-2021

CERTIFICATE

This is to certify that the Mini project titled “**KRISHNA – The Guidance Chat-Bot**” is a bonafide work of **Mr. Ritesh Sharma, Mr. Devang Vogallu, Mr. Shreyash Limkar, Mr. Deepak Rahangdale**, carried out for the partial fulfillment of the requirement of Mini Project in third year of Bachelor of Engineering in **Computer Science & Engineering, Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur.**

Dr. Ganesh Regulwar
Associate Professor
(Project Guide)

Mr. Animesh Tayal
Head of Department

Dr. S. L. Badjate
Principal

External Examiner

Mr/Mrs/Ms/Dr. _____

Designation: _____

Institution: _____

Date: _____

DECLARATION

We, hereby declare that the Mini Project titled “**KRISHNA : The Guidance Chat-bot**” submitted herein has been carried out by us in the Department of Computer Science & Engineering of S. B. Jain Institute of Technology Management and Research, Nagpur under the guidance of **Dr. Ganesh Regulwar**. The work is original and has not been submitted earlier as a whole or in part for the award of any degree / diploma at this or any other Institution / University.

RITESH Y SHARMA

V DEVANG

SHREYASH LIMKAR

DEEPAK RAHANGDALE

Date: - / /

ACKNOWLEDGEMENT

We would like to express deep sense of gratitude to our Project Guide, **Dr. Ganesh Regulwar, Department of Computer Science & Engineering**, for being the cornerstone of our project. It was her incessant motivation and guidance during periods of doubts and uncertainties that has helped us to carry on with this project.

We would like to thank **Mr. Animesh Tayal, Head of Department, Computer Science & Engineering** for providing necessary guidance, support, motivation, and inspiration without which this project would not have been possible.

We would like to extend our special thanks to **Dr. S. L. Badjate, Principal of S.B. Jain Institute of Technology, Management & Research** for his encouragement and best wishes.

We would like to extend our sincere thanks to the **Management of S.B. Jain Institute of Technology, Management & Research** for providing all the necessary infrastructure and laboratory facilities.

We also like to acknowledge the help extended by the **faculty members and non-teaching staff** of Computer Science & Engineering Department for successful completion of our project.

Last but not the least, special thanks to our family members, friends and colleagues for their continuous support.

ABSTRACT

Choosing your career path is a very momentous decision because it decides the future course of your professional life. The major problem identified is that nowadays many students facing the problem of Guidance. We created a Chatbot **“Krishna – The Career Guidance Chat Bot”** which helps the students in guidance of education and career. There are not many projects on this topic, but only one or two are there. But none of them are providing guidance and counseling in education & career track in the same Bot. The aim of career counseling bot is to carry out a conversation between both human and machine. Some knowledge has been embedded into the machine so that it identifies the sentences and deciding itself as response to answer a question. It is not just limited to one subject related guidance, instead of that student mainly faces the problem of decision making, career path selection, confusion in choosing the career track and what should I do after 12th. In most probability, you are likely to consider a career which your friends or classmates have chosen or as your parents’ desire. And they are not even aware of the existing career fields. The Solution to this problem is **“Krishna – The Career Guidance Career Bot”**.

***Key Words** - Machine Learning, ChatterBot, Education Chatbot, NLP, NLTK, Guidance Chatbot.*

INDEX

	PAGE NO.
CERTIFICATE.....	i
DECLARATION.....	ii
ACKNOWLEDGEMENT.....	iii
ABSTRACT	iv
INDEX	v-vi
LIST OF FIGURES.....	vii
LIST OF ABBREVIATIONS.....	vii
LIST OF SYMBOLS.....	viii
CHAPTER 1 INTRODUCTION	
1.1 PROJECT BACKGROUND	1
1.2 PROBLEM STATEMENT	2
1.3 PURPOSE OF STUDY	2
1.4 TECHNOLOGICAL BASE	3
CHAPTER 2 LITERATURE SURVEY	
2.1 RELATED WORK	5
CHAPTER 3 METHODOLOGY/PROPOSED SOLUTION	
3.1 PROPOSED WORK	7
3.2 SYSTEM ARCHITECTURE	8
3.3 SYSTEM FLOWCHART	9
CHAPTER 4 TOOLS/ PLATFORM	
4.1 SOFTWARE REQUIREMENT	10
4.2 HARDWARE REQUIREMENT	14
CHAPTER 5 DESIGN& IMPLEMENTATION	
5.1 SYSTEM DESIGN	15
5.1.1 USE CASE DIAGRAM	
5.1.2 SEQUENCE DIAGRAM	

5.2	IMPLEMENTED MODULES	17
5.3	SAMPLE CODE	18
CHAPTER 6	TESTING, RESULTS & DISCUSSION	
6.1	TESTING	22
	6.1.1 TYPES OF TESTING	
	6.1.2 LEVELS OF TESTING	
	6.1.3 TESTING REPORT	
6.2	RESULTS & DISCUSSION	29
CHAPTER 7	ADVANTAGES & APPLICATIONS	
7.1	ADVANTAGES	31
7.2	APPLICATIONS	31
CHAPTER 8	CONCLUSION AND FURTHER SCOPE	
8.1	CONCLUSION	32
8.2	FURTHER SCOPE	33
	REFERENCES	34
	APPENDIX I PLAGARISM REPORT	
	APPENDIX II INSTRUCTION MANUAL	






LIST OF FIGURES

Fig No.	TITLE OF FIGURE	PAGE NO.
3.2	System Architecture	20
3.3	Flowchart	21
4.1	QuAC Dataset image	26
5.1.1	Use Case	29
5.1.2	Sequence Diagram	30
6.2	Home Screen	44
6.3	Conversation Screen	45
6.4	Webpage(Front-end) Integrate	45
6.4	Output Image	45

LIST OF ABBREVIATION

ABBREVIATION	DESCRIPTION
BOT	Build Operate Transfer
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
PyNLPI	Python Pineapple

LIST OF SYMBOLS

SR.NO	SYMBOL	NAME OF SYMBOL
01		Chat Window
02		NLP Processor
03		PARSING and PAGGING
04		Database
05		Entities Extraction

Chapter 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 PROJECT BACKGROUND

Many times, students end up making a wrong career choice which they regret later and cannot do much about it. To make sure you take a rational and well-thought decision, we come with three biggest challenges you may face.

Lack of Guidance:

Every student has his or her own aptitude and career preference and hence needs customized counselling. There are a very few reliable career guidance counselling mentors, websites or physical centres which investigate this aspect.

Peer and Parent Pressure:

You feel that your peers are making the right career choice, or your parents know what is best for you. To compete with your friends or make your parents happy, you end up with a wrong career.

Salary-Driven Decisions:

Many students choose a career based on how well it will pay or whether it can land you a foreign job. There cannot be a more irrational logic than this parameter to decide your career. You think that a high package or offshore employment is the key to your career success or satisfaction. Well, that is not true.

“Krishna – The Career Guidance Chat Bot” which helps the students in guidance of education and career. There are not many projects on this topic, but only one or two are there. But none of them are providing guidance and counselling in education & career track in the same Bot. We will also shows them the different Career options available which they have not heard about. Will also inform them about the various institutions providing various courses for the concerning fields.

1.2 PROBLEM STATEMENT

Choosing your career path is a very momentous decision because it decides the future course of your professional life. The major problem identified is that nowadays many students facing the problem of Guidance. Yes! Guidance. It is not just limited to one subject related guidance, instead of that student mainly faces the problem of decision making, career path selection, confusion in choosing the career track and what should I do after 12th. In most probability, you are likely to consider a career which your friends or classmates have chosen or as your parents' desire. This kind of pressure is very common among students. And they are not even aware of the existing career fields. The Solution to this problem is **“Krishna – The Career Guidance Career Bot”**.

1.3 PURPOSE OF STUDY

Artificial intelligence, a technology that overlays computer graphics on the real world has its applications in the field of engineering and architecture and has been used to tackle real world problems.

The aim of career counselling bot is to carry out a conversation between both human and machine. Some knowledge has been embedded into the machine so that it identifies the sentences and deciding itself as response to answer a question. The response principle is to extract the tokens from the sentence process on that find the goal of sentence by matching the input sentence from user.

The bot will help the users who have passed SSC and HSC to select their field of interest or a field that would be best for them to build up their future. And the main thing is that our bot not just only deal with the students who face the career selection problem, but also will be helpful for the peoples who are Mentally Depressed and the one who is dealing with the Suicidal Thoughts, Depression and Anxiety. But this part we will cover in the second phase.

Chatbots are computer programs that simulate intelligent conversation. They are situated between games and toys, as their aim is mostly to be entertaining, but the user does not have to follow precise rules when playing with the program. The aim of this

paper is to explore career choices available, as well as to propose several graphs to evaluate area of interest. Currently educational applications have started to emerge as a further development of the idea of intelligent dialog.

Objectives:

- To Provide Guidance and Consultation to students.
- To Reduce the time and efforts of students.
- To help the students for choosing perfect career track.
- To build the platform that deal with real life problems of students.

1.4 TECHNOLOGICAL BASE

This Project can be implemented by using various technologies like-

Python

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum from 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is dynamically typed, and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "battery included" language due to its comprehensive standard library.

Features:

- Easy-to-learn – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- Easy-to-read – Python code is more clearly defined and visible to the eyes.
- Easy-to-maintain – Python's source code is fairly easy-to-maintained.
- Scalable – Python provides a better structure and support for large programs than shell scripting.

Chatterbot:

ChatterBot is a Python library that makes it easy to generate automated responses to a user's input. Chatterbot uses a selection of machine learning algorithms to produce different types of responses. This makes it easy for developers to create chat bots and automate conversations with users.

Features:

- Inbuilt conversational dialog flow and training engine.
- The bot created using this library will get trained automatically with the response it gets from the user.
- It makes it easy to generate automated responses to a user's input
- Process mapping: Use the conversation builder to turn your current processes quickly and easily into automated conversations.

HubSpot's Chatbot Builder:

With HubSpot's free chatbot builder software, you can easily create bots that help you qualify leads, book meetings, provide answers to common support questions, and more.

Feature's:

- No coding Experience Required to make chatbot
- We can Choose prebuilt template based on your bot's goal.

CHAPTER 2
LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

2.1 RELATED WORK

- **Anderson et al. (2001)** tied on these findings and tried to classify learning outcomes with a Chat-Bot matrix consisting of a knowledge and a cognitive process dimension. The knowledge dimension consists of factual, conceptual, procedural, and metacognitive knowledge. Apart from already described factual and procedural knowledge, they added conceptual and metacognitive knowledge. Conceptual knowledge shows the interrelationships among basic concepts and metacognitive knowledge is the knowledge of cognition in general. Chat Bot works in many forms and combines different learning styles and methods:
 - Web or computer-based Asynchronous or synchronous
 - Instructor-led or self-paced
 - Individual-based or team-based (collaborative learning)
- **Sofie Roos et al. (CHATBOTS IN EDUCATION):** Introduce the task of **Visual Dialog** in Chat Bot, which requires an AI agent to hold a meaningful dialog with humans in natural, conversational language about visual content. Specifically, given an image, a dialog history, and a question about the image, the agent must ground the question in image, infer context from history, and answer the question accurately. Visual Dialog is disentangled enough from a specific downstream task to serve as a general test of machine intelligence, while being grounded in vision enough to allow objective evaluation of individual responses and benchmark progress. We develop a novel two-person chat data-collection protocol to curate a large-scale Visual Dialog dataset (VisDial). VisDial v0.9 has been released and contains 1 dialog with 10 question-answer pairs on ~120k images from COCO, with a total of ~1.2M dialog question-answer pairs.

- **BeeBee Blackboard Chat-Bot:**

- AI based chatbot show Higher Education Opportunity.
- Blackboard Chatbot provides a better UI experience that is always available.
- Give students, faculty, and staff the information they need, when they need it, in the way that they prefer to communicate with a self-service chatbot.

- **Gurukul-College Enquiry Chatbot:**

- Gurukul is built using artificial algorithms that analyses user's queries and understand user's message.
- This System is a web application which provides answer to the query of the student. Students just must query through the bot which is used for chatting.
- The user can query about the college related activities through online with the help of this web application.

Our project aim is to develop a chatbot, which helps the students in guidance of education and career. Our chatbot will be an intelligent system that can hold a conversation with a human using natural language in real-time. Chatbots are computer programs that simulate intelligent conversation. They are situated between games and toys, as their aim is mostly to be entertaining, but the user does not have to follow precise rules when playing with the program.

The aim of this paper is to explore career choices available, as well as to propose several graphs to evaluate area of interest. Our chatbot, which will work as a conversational agent, and is a computer software capable of taking a natural language input and providing a conversational output in real-time. This human-chatbot interaction is typically carried out through a graphical user interface based on **Human-computer Interaction (HCI)** principles.

The intent of our project is that to make a chatbot, which will work as a conversational agent, and is a computer software capable of taking a natural language input and providing a conversational output in real time.

CHAPTER 3

METHODOLOGY / PROPOSED SOLUTION

CHAPTER 3

METHODOLOGY / PROPOSED SOLUTION

3.1 PROPOSED SOLUTION

The intent of our project is that to make a chatbot, which will work as a conversational agent, and is a computer software capable of taking a natural language input and providing a conversational output in real time. Our chatbot will be an intelligent system that can hold a conversation with a human using natural language in real-time.

We have divided our project in few modules listed below:

Module 1: Dataset Gathering

- **Creation of self-made Dataset-** We have made our own short Conversational Dataset in YML Format.
- **Gather Train Dataset-** We have used QuAC Dataset for training of our model. It consists of more than 30,000 Question and Answers to train the model.
- **Data Pre-processing and Dataset Void Filtration**

Module 2: Chat-Bot Training/Testing and Website Integration

- **Train Chatbot -** By the help of QuAC Dataset, we have train our chatbot.
- **Frontend Designing –** With the help of HTML, CSS, and BOOTSTRAP, we have designed our frontend.
- **Integrate chatbot with Website (Deployment): -** We have developed the application using Flask framework and integrate it with webpage.

Module 3: Chat-Bot Testing

- Check whether our chatbot replying correctly.
- Check whether our chatbot perfectly integrate with webpage or not.

3.2 SYSTEM ARCHITECTURE

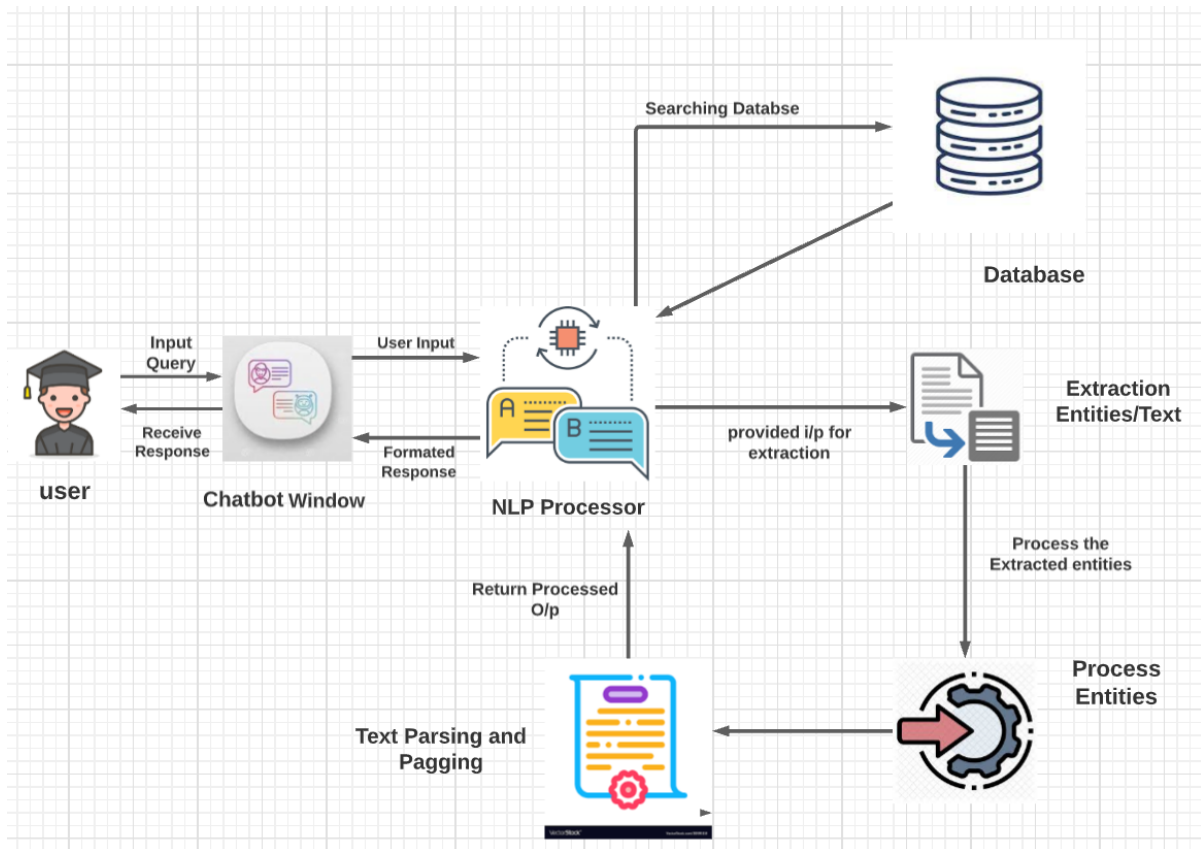


Figure 3.2 System Architecture

In the above figure 3.2, You can see the System Architecture our chatbot, in which User must put some Input Query in Chat Window of chatbot. Then that query will send to NLP Processor for processing. Then NLP Processor compare the query with queries already stored in database.

If the queries have not found any match with the queries stored in database. Then that query will send for extraction by which it extracted in some entities and text or keyword. Then that extracted keywords or entities are processed to find a pattern. The above architecture describes the complete view of the app and its usage in the perfect manner.

3.3 FLOW CHART:

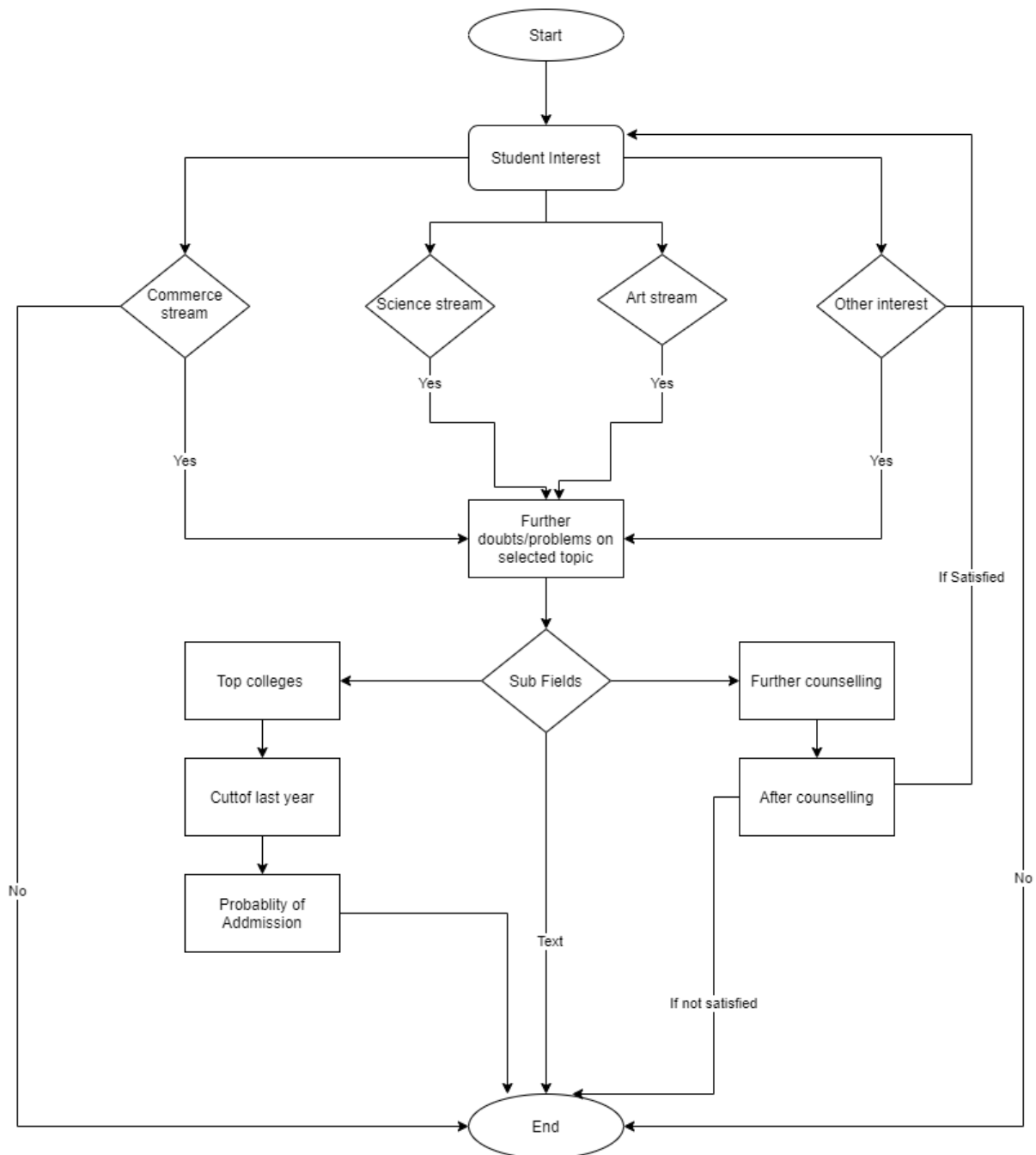


Figure 3.3 Flow Chart

In the above figure 3.3, the Flow Chart of our project. In this, the user just must provide the stream he/she want to choose, then according to that Chatbot shows some colleges.

CHAPTER 4

TOOLS/PLATFORM

CHAPTER 4

TOOLS/PLATFORM

4.1 SOFTWARE REQUIREMENT

a) CLIENT-SIDE TECHNOLOGY: HTML, CSS, Bootstrap.

HTML:

HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages.

Features:

- It is easy to learn and easy to use.
- It is platform independent.
- Images, video, and audio can be added to a web page.
- Hypertext can be added to text.

CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Features:

- You can control the- colour of the text, the style of fonts etc.
- Variations in display for different devices and screen sizes as well as a variety of other effects.

BOOTSTRAP:

Bootstrap is a free and open-source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML,

CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps. Responsive design makes it possible for a web page or app to detect the visitor's screen size and orientation and automatically adapt the display accordingly.

Features:

- Easy to Begin With
- LESS as Well as CSS Files
- Easily Customizable
- Responsive Utility Classes
- Some of the components come pre-styled in Bootstrap

b) **SERVER-SIDE TECHNOLOGY:** Python, Flask

Python:

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is dynamically typed, and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a battery included language due to its comprehensive standard library.

Features:

- Easy to code
- Free and Open Source
- Object-Oriented Language
- High-Level Language
- Dynamically Typed Language
- Interpreted Language

Flask:

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications.

It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Features:

- Built-in development server, fast debugger.
- Integrated support for unit testing.
- RESTful request dispatching.
- Jinja2 Templating.
- Support for secure cookies.

c) **IDE**

d) / **FRAMEWORK:** Visual Studio, Brackets, Flask

Flask:

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Features:

- Built-in development server, fast debugger.
- Integrated support for unit testing.
- RESTful request dispatching.
- Jinja2 Templating.
- Support for secure cookies.

e) **WEB SERVER:**

f) **DATABASE:** Sqlite, YML, QuAC Dataset

SQLite:

SQLite is a relational database management system contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine.

QuAC Dataset:

Question Answering in Context is a dataset for modeling, understanding, and participating in information seeking dialog. Data instances consist of an interactive dialog between two crowd workers: (1) a *student* who poses a sequence of freeform questions to learn as much as possible about a hidden Wikipedia text, and (2) a *teacher* who answers the questions by providing short excerpts (spans) from the text. QuAC introduces challenges not found in existing machine comprehension datasets: its questions are often more open-ended, unanswerable, or only meaningful within the dialog context.

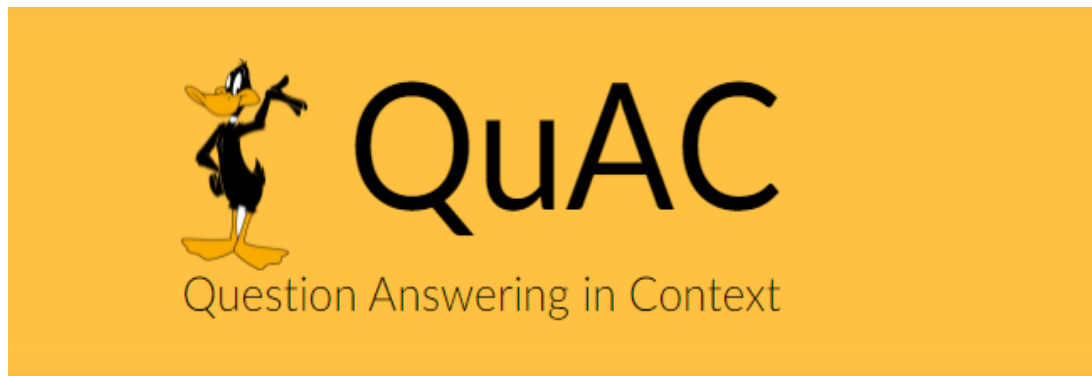


Figure 4.1 QuAC Dataset

g) **OPERATING SYSTEM:** Windows 7 or above

h) **DESIGNING TOOLS:** Lucid chart

Lucid Chart:

Lucidchart is a web-based proprietary platform that allows users to collaborate on drawing, revising, and sharing charts and diagrams.

i) **TESTING TOOL:** MS-EXCEL

For testing purpose, we used MS Excel to observe the results of application.

j) **Libraries:** ChatterBot, NLTK, PyNLPI

ChatterBot:

As the name suggests, chatterbot is a python library specifically designed to generate chatbots. This algorithm uses a selection of machine learning algorithms to fabricate varying responses to users as per their requests. Chatterbot makes it easier to develop chatbots that can engage in conversations.

NLTK:

The **Natural Language Toolkit**, or more commonly **NLTK**, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language.

PyNLPI:

PyNLPI, pronounced as 'pineapple', is a Python library for Natural Language Processing. It contains various modules useful for common, and less common, NLP tasks. PyNLPI can be used for basic tasks such as the extraction of n-grams and frequency lists, and to build simple language model. There are also more complex data types and algorithms.

4.2 HARDWARE REQUIREMENT

PROCESSOR : Min Intel i5 and above.

HARD DISK : Min 128 GB.

RAM : Min 512 Mb and above

CHAPTER 5

DESIGN & IMPLEMENTATION

CHAPTER 5

DESIGN & IMPLEMENTATION

5.1 SYSTEM DESIGN

5.1.1 USE-CASE DIAGRAM

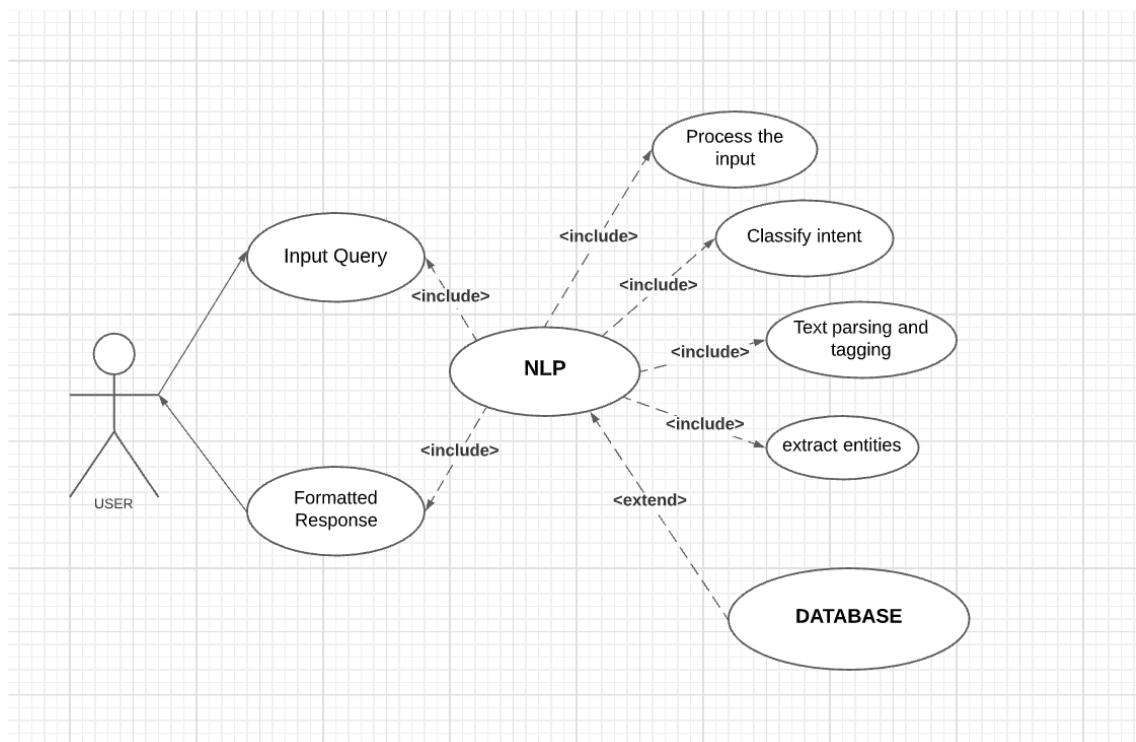


Figure 5.1.1 Use Case Diagram

In the above Figure 5.1.1, You can see the Use Case Diagram simply depicts the task which a user can perform through our Chatbot.

These tasks are as follows:

- User can Input Query
- User can choose the available subject stream
- User can choose the available college as per stream

5.1.2 SEQUENCE DIAGRAM

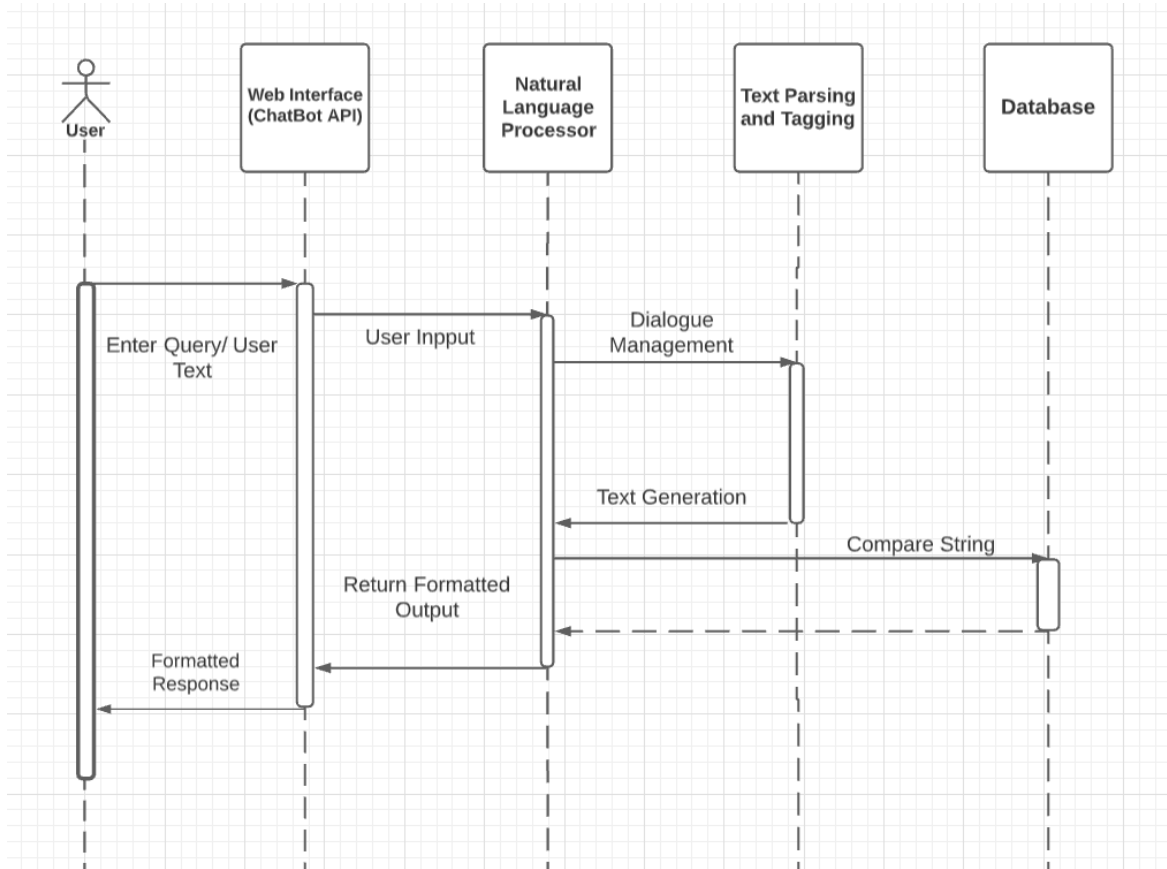


Figure 5.1.2 Sequence Diagram

In the above figure 5.1.2, you can see the Sequence Diagram of the Chatbot:

- User must put some Input Query in Chat Window of chatbot. Then that query will send to NLP Processor for processing.
- Then NLP Processor compare the query with queries already stored in database.
- If the queries have not found any match with the queries stored in database. Then that query will send for extraction by which it extracted in some entities and text or keyword.
- Then that extracted keywords or entities are processed to find a pattern.
- The above Sequence Diagram describes the complete view of the app and its timeline usage in the perfect manner.

5.2 IMPLEMENTED MODULES

We have completed the Creation of self-made Dataset, Gather Train Dataset, Dataset Gathering and Processing. We also Completed the GUI Designing implementation using Flask. In this module, we have completed designing the user interface of web page.

Module 1: Dataset Gathering

- **Creation of self-made Dataset-** We have made our own short Conversational Dataset in YML Format.
- **Gather Train Dataset-** We have used QuAC Dataset for training of our model. It consists of more than 30,000 Question and Answers to train the model.
- **Data Pre-processing and Dataset Void Filtration -** A dataset is collected and pre-processed accordingly for training and testing purpose of the model.

Module 2: Chat-Bot Training/Testing and Website Integration

- **Train Chatbot -** By the help of QuAC Dataset, we have train our chatbot.
- **Frontend Designing –** With the help of HTML, CSS, and BOOTSTRAP, we have designed our frontend.
- **Integrate chatbot with Website (Deployment): -** We have developed the application using Flask framework and integrate it with webpage.

Module 3: Chat-Bot Testing

- Check whether our chatbot replying correctly
- Check whether our chatbot perfectly integrate with webpage or not.

5.3 SAMPLE CODE:

```
from flask import Flask, render_template, request, jsonify, Response
from chatterbot import ChatBot
from chatterbot.trainers import ChatterBotCorpusTrainer, ListTrainer
import json
import os, uuid
from datetime import datetime
from pytz import timezone
from nltk.tokenize import word_tokenize

app = Flask(__name__)

english_bot = ChatBot("Chatterbot",
storage_adapter="chatterbot.storage.SQLStorageAdapter")

#trainer = ListTrainer(english_bot)
#text = open("engineering.yml", "r")
#training = text.readlines()

trainer = ListTrainer(english_bot)

#trainer.train('chatterbot.corpus.english')
text_file = open("sample.txt", "r")
text = text_file.readlines()

sample_text = [s.replace('\n','') for s in text]
trainer.train(sample_text)
text_file = open("conversations.yml", "r")
text = text_file.readlines()
```

```

trainer.train(training_data)
text_file = open("blocked.txt", "r")
block = text_file.readlines()

blocked = [s.replace('\n', '') for s in block]

@app.route("/")
def home():
    return render_template("index.html")

@app.route("/get")
def get_bot_response():

    userText = request.args.get('msg')
    user_words = word_tokenize(userText)

    for word in user_words:
        if word in blocked:
            return str("Please avoid bad words!")

    return str(english_bot.get_response(userText))
    #return str("Hello")
    return str(english_bot.get_response(userText))
    #return str("Hello")

@app.errorhandler(404)
def not_found(error=None):
    message = {
        'status': 404,
        'message': 'Not Found: ' + request.url,
    }
    resp = jsonify(message)
    resp.status_code = 404

```

```

    return resp
@app.errorhandler(500)
def internalservererror(error=None):
    message = {
        'status': 500,
        'message': 'Unexpecteppd server error or Internal Server Error',
    }
    resp = jsonify(message)
    resp.status_code = 500

    return resp

@app.errorhandler(502)

def gatewaytimeout(error=None):
    message = {
        'status': 502,
        'message': 'Gateway time out error',
    }

@app.errorhandler(502)

def gatewaytimeout(error=None):
    message = {
        'status': 502,
        'message': 'Gateway time out error',
    }

    resp = jsonify(message)
    resp.status_code = 502

    return resp

```

```
@app.errorhandler(400)
def Badrequest(error=None):
    message = {
        'status': 400,
        'message': 'Bad request',
    }
    resp = jsonify(message)
    resp.status_code = 400

    return resp

if __name__ == "__main__":
    app.run(debug=True)
```

CHAPTER 6
TESTING, RESULTS &
DISCUSSION

CHAPTER 6

RESULTS & DISCUSSION

6.1 TESTING

6.1.1 TYPES OF TESTING

Manual Testing

Manual testing includes testing a software manually, i.e., without using any automated tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behaviour or bug. There are different stages for manual testing such as unit testing, integration testing, system testing, and user acceptance testing.

Testers use test plans, test cases, or test scenarios to test a software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identify errors in it. Manual Testing is one of the most fundamental testing processes as it can find both visible and hidden defects of the software. The difference between expected output and output, given by the software, is defined as a defect.

Following are the testing techniques that are performed manually during the test life cycle:

- Acceptance Testing
- White Box Testing
- Black Box Testing
- Unit Testing
- System Testing
- Integration Testing

Automation Testing

Automation testing, which is also known as Test Automation, is when the tester writes scripts and uses software to test the product. This process involves automation of a manual process. Automation Testing is used to re-run the test scenarios that were performed manually, quickly, and repeatedly.

Test automation is the practice of running tests automatically, managing test data, and utilizing results to improve software quality. It's primarily a quality assurance measure, but its activities involve the commitment of the entire software production team.

Apart from regression testing, automation testing is also used to test the application from load, performance, and stress point of view. **Test Automation** is the best way to increase the effectiveness, test coverage, and execution speed in software testing.

Test Automation should be used by considering the following aspects of a software:

- Large and critical projects.
- Projects that require testing the same areas frequently
- Requirements not changing frequently
- Manual Testing of all workflows, all fields, all negative scenarios is time and money consuming
- Accessing the application for load and performance with many virtual.
- Stable software with respect to manual testing
- Availability of time

6.1.2 LEVELS OF TESTING

There are four levels of testing: Unit, Integration, System and Acceptance.

1. Unit Testing: A level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed. Unit Testing is done during the development (coding phase) of an application by the developers. Unit Tests isolate a section of code and verify its correctness.

2. Integration Testing: A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.

3. System Testing: A level of the software testing process where a complete, integrated system/software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements. System Testing is a series of different tests whose sole purpose is to exercise the full computer-based system.

4. Acceptance Testing: A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

6.1.3 TESTING REPORT

A	B	C	D	E	F	G	H	I
PROJECT:	KRISHNA- The Guidance Chatbot							
1	MODULE:	Chat-Bot Training and Webpage Integration:						
2	PREPARED BY:	Ritesh Sharma						
3		Devang Vogallu						
4		Shreyash Linkar						
5		Deepak Rahangdale						
6								
7	SR.NO.	TEST CASE ID	TEST OBJECTIVE	STEPS	DATA	PREREQUISITE	EXPECTED RESULT	ACTUAL RESULT
1		TC_ADMIN_01	Check the Preprocessing of Dataset is working fine	Run the command for pre-processing of dataset file	NIL	Dataset	No error should be reported by compiler	No error was reported by terminal
2		TC_ADMIN_02	Check whether the Void or Gap in Dataset is filled or not	after run the command for pre-processing of dataset file, check the result in terminal	NIL	Dataset	No error should be reported by compiler and all the void should be filled by random input	No error was reported by terminal and all the voids are filled
3		TC_ADMIN_03	Check the Chatbot is training with the input Dataset	Check the visible message or result at Terminal.	NIL	Dataset	Terminal should show List Trainer with progressive completion	Terminal had shows the List Trainer with Progressive Bar
4		TC_ADMIN_04	Check whether the Chatbot is perfectly Integrated with webpage or not	Check the visible user interface of the screen.	NIL	NIL	UI should be perfect and chatbot is perfect integrated with webpage displayed.	UI was perfect and chatbot is perfectly integrated with webpage and it was displayed.
9								Pass
10								Pass
11								Pass
12								Pass

A	B	C	D	E	F	G	H	I
5	TC_ADMIN_05	Check whether the Local server is established or not	Run the code	NIL	NIL	At last terminal should reply with Local server address	Terminal successfully established the Local server	Pass
6	TC_ADMIN_06	Check whether the chatbot has replied with formatted response at first	Check the visible chat window	NIL	NIL	Chat Bot should show the "Hi, I am Krishna the guidance Chatbot" message	Chatbot had show "Hi, I am Krishna the guidance Chatbot" message	Pass
7	TC_ADMIN_07	Check whether the input box is taking responses or not	Enter any text or string	NIL	NIL	Input box should take text or string	Input box had takes text or string	Pass
8	TC_ADMIN_08	Check whether the Send Response button is working or not	Press the Send Response button after entering any text	NIL	NIL	Response should be sent after pressing the Send button	Response has been sent after pressing the Send button	Pass

	A	B	C	D	E	F	G	H	I
1	PROJECT:	KRISHNA- The Guidance Chatbot							
2	MODULE:	Chat-Bot Testing:							
3	PREPARED BY:	Ritesh Sharma							
4		Devang Vogallu							
5		Shreyash Linkar							
6		Deepak Rahangdale							
7	SR.NO.	TEST CASE ID	TEST OBJECTIVE	STEPS	DATA	PREREQUISIT E	EXPECTED RESULT	ACTUAL RESULT	STATUS
8	1	TC_USER_01	Check whether the User Interface of Chatbot is perfectly loaded or not	Check the visible user interface of the screen.	NIL	NIL	User Interface should be perfect and chatbot is perfectly integrated with webpage and should be displayed.	User Interface was perfectly loaded and chatbot is perfectly integrated with webpage and it was displayed.	Pass
9	2	TC_USER_02	Check whether the chatbot has replied with formatted response at first	Check the visible chat window	NIL	NIL	Chat Bot should show the "Hi, I am Krishna the guidance Chatbot" message	Chatbot had show "Hi, I am Krishna the guidance Chatbot" message	Pass
10	3	TC_USER_03	Check whether the chatbot has replied with formatted response at first after reload of webpage	Check the visible chat window after Reload	NIL	NIL	Chat Bot should show the "Hi, I am Krishna the guidance Chatbot" message	Chatbot had shows "Hi, I am Krishna the guidance Chatbot" message	Pass
11	4	TC_USER_04	Check whether the chatbot is showing input box or not	Check the visible input box at chat window	NIL	NIL	Chatbot should show the input box with "Type your message.." message..	Chatbot had shows the input box with "Type your message.." message	Pass
12									

A	B	C	D	E	F	G	H	I
5	TC_USER_05	Check whether the chatbot is showing input box or not, after reload of webpage	Check the visible input box at chat window after reload of webpage	NIL	NIL	Chatbot should show the input box with "Type your message.." message, after reload of webpage	Chatbot had shows the input box with "Type your message.." message, after reload of webpage	Pass
6	TC_USER_06	Check whether the input box is taking responses or not	Enter any text or string	NIL	NIL	Input box should take text or string	Input box had takes text or string	Pass
7	TC_USER_07	Check whether the Send Response button is working or not	Press the Send Response button after entering any text	NIL	NIL	Response should be sent after pressing the Send button	Response has been sent after pressing the Send button	Pass
8	TC_USER_08	Check whether the chatbot has replied with formatted response on your query or not	Check the visible chat window	NIL	NIL	Chatbot should reply with formatted Response	Chatbot replied with Formatted Response	Pass
9	TC_USER_09	Check whether the formatted response is correct as per query	Check the visible response at chat window	NIL	NIL	Chatbot should reply with correct formatted response as per query	Chatbot replied with correct formatted response as per query	Pass
10	TC_USER_10	Check whether the formatted response is correct as per different query	Check the visible response at chat window	NIL	NIL	Chatbot should reply with correct formatted response as per query	Chatbot replied with incorrect formatted response as per query	Fail

6.2 RESULTS AND DISCUSSIONS:

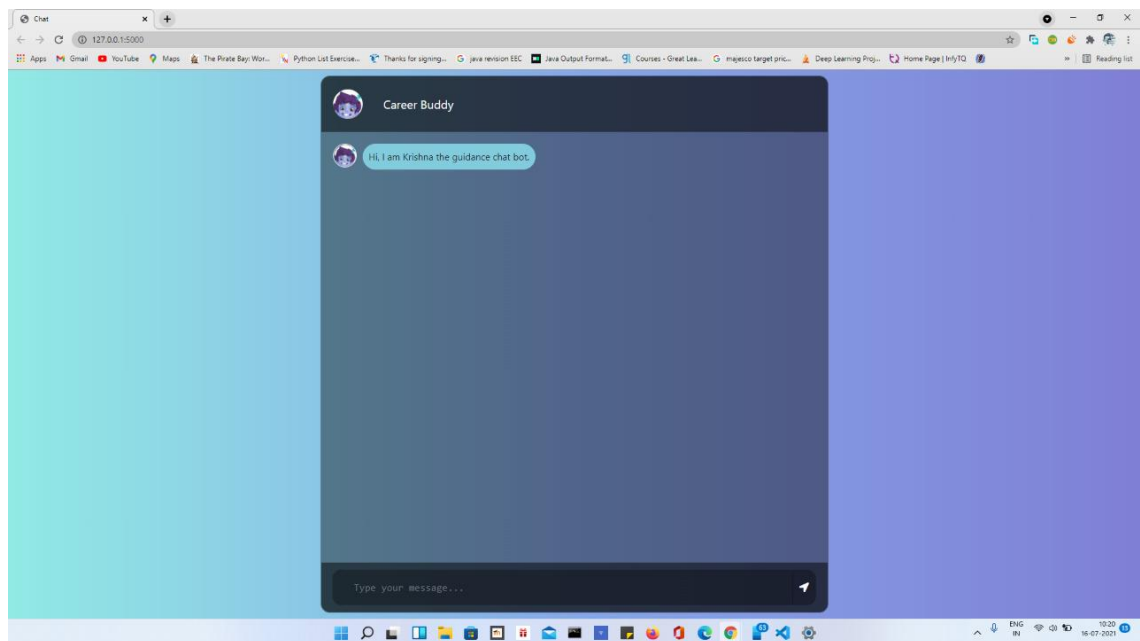
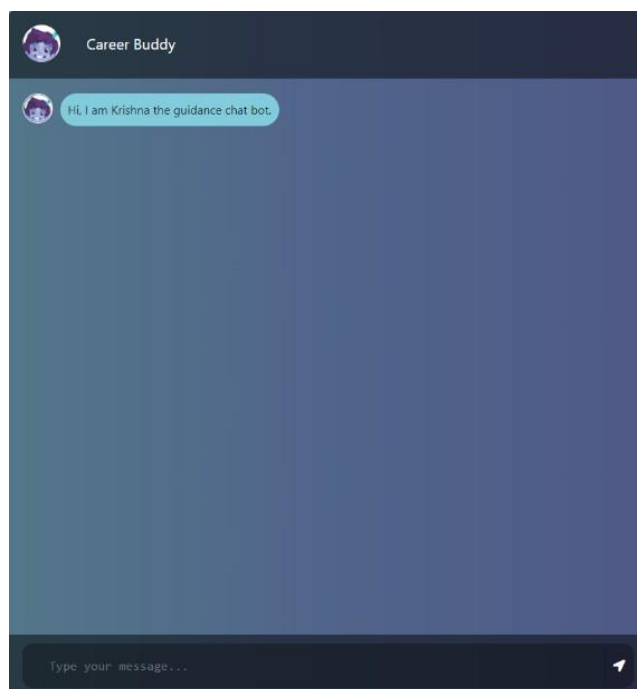


Figure 6.2 Home Screen

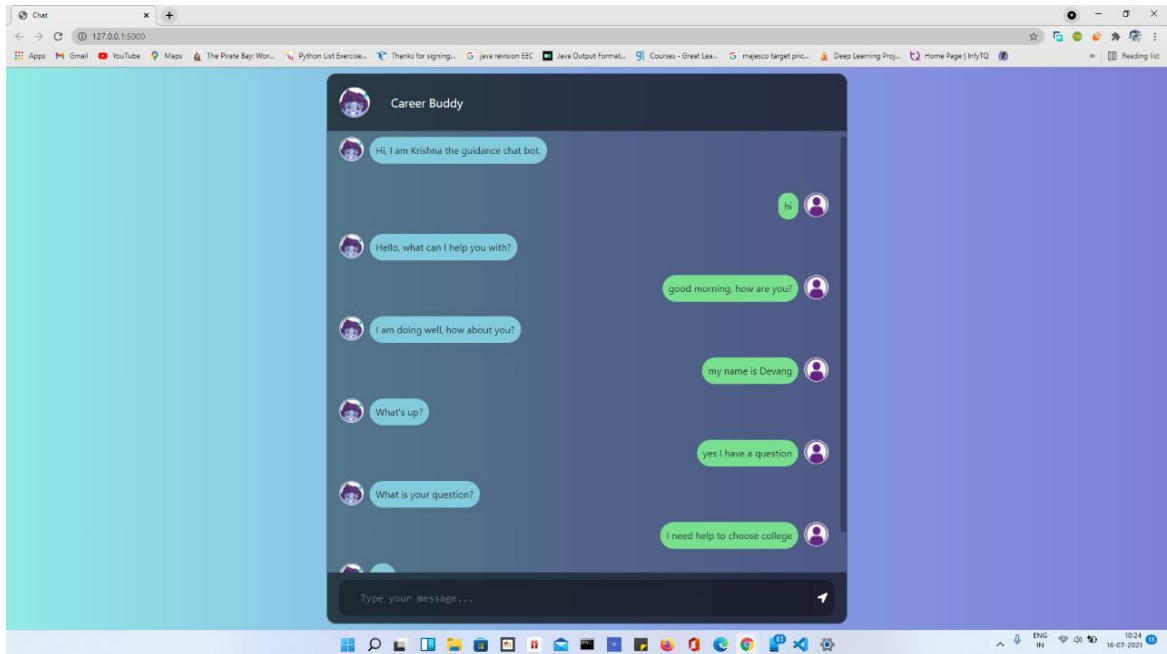
In the above Figure 6.2, You can see the Home-screen of Chatbot. This is Home-screen of “Krishna – The Guidance Chatbot”. User can input any query or message in type box.

Type Box have sent button which is used to send your input query to chatbot.



We will enter or provide some queries to chatbot, to analyse how the chatbot will react or reply to our queries.

Figure 6.3 Conversation screen



We had entered or provide some queries to chatbot, and we analysed how the chatbot will react or replied to our queries and chatbot successfully reacted in correctly manner.

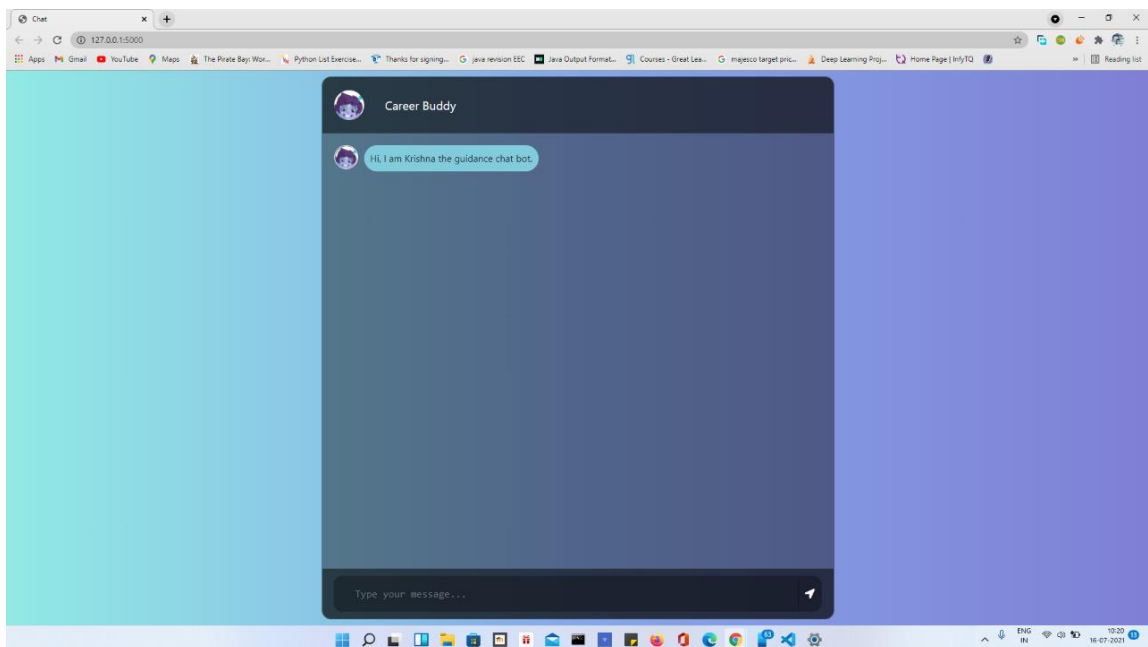


Figure 6.4 Webpage Integration and Output Image

CHAPTER 7
ADVANTAGES AND
APPLICATIONS

CHAPTER 7

ADVANTAGES AND APPLICATIONS

7.1ADVANTAGES

- **Better Interface:**

Provide Chat Bot interface to provide guidance to students, which enable the easiest way to understand or find out the solution to problem.

- **System Scalable:**

System Scalable, User-friendly, Highly Interactive and Easy to use.

- **Speedy Response:**

- We have imported all types of questions from Dataset which train the chatbot in best specific manner that enables the chatbot to give the fast response to user.

Applications:

- **Stream Selection:**

It shows you to select the subject stream from available streams.

It will help the student to practically analyse their potential and field of interest.

- **College Choices:**

It offers varieties of colleges according to your city or choice of stream.

It will provide the students with a clear path so as what to pursue for their higher studies.

- **College Information:**

It also provides all the related college information like location, acceptance rate and available stream.

It would guide them to choose the correct career path asper their interest.

CHAPTER 8
**CONCLUSION & FURTHER
SCOPE**

CHAPTER 8

CONCLUSION & FURTHER SCOPE

8.1 CONCLUSION

We have **designed and developed** a Chat-Bot by applying **engineering knowledge** which provides an approach in building a platform where users can try to solve their different problems in the conversational environment.

As a student, we always encounter many problems in which the **solution strategy** is required and Chatbot can provide a convenient platform for users to make queries. It solved the **societal problem** of users with their different problems in education fields. This chatbot will give a unique platform for the users to find the best colleges and subjects for them.

We have used **modern tools** like Vscod, Brackets, and libraries like NLP, Chatterbot, Numpy, pytz, and nltk.tokenize, etc and Frameworks like Flask to implement this project. During the development of the project, we understood the importance of **individual and teamwork** while **project development and management**. While showcasing our project through various seminars and conferences we enhanced our **communication skills** and displayed **professional ethics** which results in **lifelong learning**.

8.2 FURTHER SCOPE

- **Personality Analysis:**

We will include the feature of Personality analysis option based on their Psychometric Test profile. In this first, Student must give the Psychometric Test and on basis of that or on different traits bot will evaluate their profile.

- **Factors based Analysis:**

We will analyse the different-different factors to choose the career track or stream chosen by student. On what factors does the student have to choose right career track for themselves.

- **Guidance for mental health:**

We will include the feature of Analysis of mental health of students. Many students are currently dealing with Depression, Anxiety and confusion to choose stream.

- **Voice over feature:**

We will include the Voice over feature to our chatbot to make it easy to use and friendly for disabled person.

REFERENCES

PAPERS:

1. Safiya Shaikh, Dipti More, Ruchika Puttoo, “A Survey Paper on Chatbots”, International Research Journal of Engineering and Technology (IRJET). Volume: 06 Issue: 04, Apr 2019
2. Rincy Mariam Thomas, Supriya Punna, Dr. B V RamanaMurthy, “Survey on Artificially Intelligent Chatbot- Chappie”. Volume VI, Issue I, January/2019
3. Ergun Gide, Niti Sandhu, “AI-Chatbots to Enhance Student Learning Experience”, IEEE Explore Paper.
4. Sofie Roos, Ruth Lochan, “CHATBOTS IN EDUCATION”, Department of Informatics and Media Uppsala University, 2017
5. Molnár, G. and Z. Szüts. “The Role of Chatbots in Formal Education”. in 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY). 2018. IEEE.

BOOKS

1. Herbert Schildt, “Python the Complete Reference”, Fifth Edition.

WEBSITES

1. www.shiksha.com
2. [https://towardsdatascience.com/how-to-create-a-chatbot-with-python-deep learning-in-less-than-an-hour-56a063bdfc44](https://towardsdatascience.com/how-to-create-a-chatbot-with-python-deep-learning-in-less-than-an-hour-56a063bdfc44)
3. <https://www.drift.com/learn/chatbot>

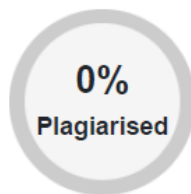
APPENDIX I

PLAGIARISM REPORT



Date: July, 16 2021

PLAGIARISM SCAN REPORT



Excluded Url : None

Content Checked For Plagiarism

Choosing your career path is a very momentous decision because it decides the future course of your professional life. The major problem identified is that nowadays many students facing the problem of Guidance. We created a Chatbot "Krishna – The Career Guidance Chat Bot" which helps the students in guidance of education and career. There are not many projects on this topic, but only one or two are there. But none of them are providing guidance and counseling in education & career track in the same Bot. The aim of career counseling bot is to carry out a conversation between both human and machine. Some knowledge has been embedded into the machine so that it identifies the sentences and deciding itself as response to answer a question. It is not just limited to one subject related guidance, instead of that student mainly faces the problem of decision making, career path selection, confusion in choosing the career track and what should I do after 12th. In most probability, you are likely to consider a career which your friends or classmates have chosen or as your parents' desire. And they are not even aware of the existing career fields. The Solution to this problem is "Krishna – The Career Guidance Career Bot". Key Words - Machine Learning, ChatterBot, Education Chatbot, NLP, NLTK, Guidance Chatbot



APPENDIX II

Instructional Manual

On

“KRISHNA: The Guidance Chat-bot”

Submitted By

Mr. Ritesh Sharma

Mr. V Devang

Mr. Shreyash Limkar

Mr. Deepak Rahangdale

Under the Guidance of

Dr. Ganesh Regulwar



Department of Computer Science & Engineering

**S. B. Jain Institute of Technology Management
and Research
Nagpur-441501**

2020-2021

1. Abstract

Choosing your career path is a very momentous decision because it decides the future course of your professional life. The major problem identified is that nowadays many students facing the problem of Guidance. We created a Chatbot “**Krishna – The Career Guidance Chat Bot**” which helps the students in guidance of education and career. There are not many projects on this topic, but only one or two are there. But none of them are providing guidance and counseling in education & career track in the same Bot. The aim of career counseling bot is to carry out a conversation between both human and machine. Some knowledge has been embedded into the machine so that it identifies the sentences and deciding itself as response to answer a question. It is not just limited to one subject related guidance, instead of that student mainly faces the problem of decision making, career path selection, confusion in choosing the career track and what should I do after 12th. In most probability, you are likely to consider a career which your friends or classmates have chosen or as your parents’ desire. This kind of pressure is very common among students. And they are not even aware of the existing career fields. The Solution to this problem is “**Krishna – The Career Guidance Career Bot**”.

***Key Words** - Machine Learning, ChatterBot, Education Chatbot, NLP, NLTK, Guidance Chatbot.*

2. Modules Implemented

Module 1: Dataset Gathering

- **Creation of self-made Dataset-** We have made our own short Conversational Dataset in YML Format.
- **Gather Train Dataset-** We have used QuAC Dataset for training of our model. It consists of more than 30,000 Question and Answers to train the model.
- **Data Pre-processing and Dataset Void Filtration -** A dataset is collected and pre-processed accordingly for training and testing purpose of the model.

Module 2: Chat-Bot Training/Testing and Website Integration

- **Train Chatbot** - By the help of QuAC Dataset, we have train our chatbot.
- **Frontend Designing** – With the help of HTML, CSS, and BOOTSTRAP, we have designed our frontend.
- **Integrate chatbot with Website (Deployment):** - We have developed the application using Flask framework and integrate it with webpage.

Module 3: Chat-Bot Testing

- Check whether our chatbot replying correctly.
- Check whether our chatbot perfectly integrate with webpage or not.

3. Software and Hardware Requirement:

3.1 Software Requirement

- a) **CLIENT-SIDE TECHNOLOGY:** HTML, CSS, Bootstrap.

HTML:

HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly.

Features:

- It is easy to learn and easy to use.
- It is platform independent.
- Images, video, and audio can be added to a web page.
- Hypertext can be added to text.

CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Features:

- You can control the- colour of the text, the style of fonts etc.
- Variations in display for different devices and screen sizes as well as a variety of other effects.

BOOTSTRAP:

Bootstrap is a free and open-source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps. Responsive design makes it possible for a web page or app to detect the visitor's screen size and orientation and automatically adapt the display accordingly.

Features:

- Easy to Begin With
- LESS as Well as CSS Files
- Easily Customizable
- Responsive Utility Classes
- Some of the components come pre-styled in Bootstrap

b) SERVER-SIDE TECHNOLOGY: Python, Flask**Python:**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is dynamically typed, and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a battery included language due to its comprehensive standard library.

Features:

- Easy to code
- Free and Open Source
- Object-Oriented Language
- High-Level Language
- Dynamically Typed Language
- Interpreted Language

Flask:

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Features:

- Built-in development server, fast debugger.
- Integrated support for unit testing.
- RESTful request dispatching.
- Jinja2 Templating.
- Support for secure cookies.

c) **IDE / FRAMEWORK:** Visual Studio, Brackets, Flask

Flask:

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Features:

- Built-in development server, fast debugger.
- Integrated support for unit testing.
- RESTful request dispatching.
- Jinja2 Templating.
- Support for secure cookies.

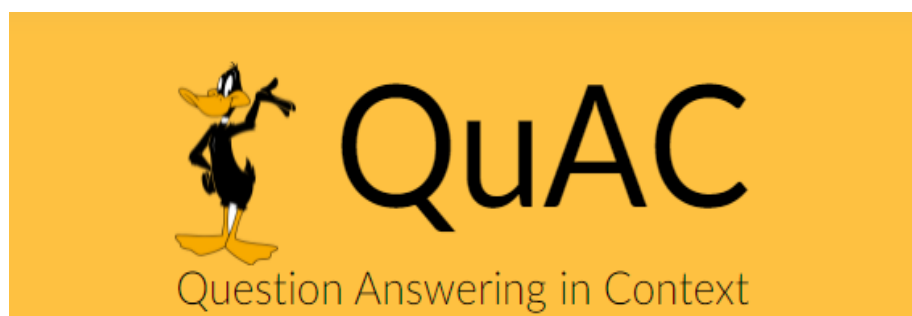
d) **DATABASE:** Sqlite, YML, QuAC Dataset

SQLite:

SQLite is a relational database management system contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program. SQLite generally follows PostgreSQL syntax.

QuAC Dataset:

Question Answering in Context is a dataset for modeling, understanding, and participating in information seeking dialog. Data instances consist of an interactive dialog between two crowd workers: (1) a student who poses a sequence of freeform questions to learn as much as possible about a hidden Wikipedia text, and (2) a teacher who answers the questions by providing short excerpts (spans) from the text. QuAC introduces challenges not found in existing machine comprehension datasets: its questions are often more open-ended, unanswerable, or only meaningful within the dialog context.



e) **OPERATING SYSTEM:** Windows 7 or above

f) **DESIGNING TOOLS:** Lucid chart

Lucid Chart:

Lucidchart is a web-based proprietary platform that allows users to collaborate on drawing, revising, and sharing charts and diagrams.

g) **TESTING TOOL:** MS-EXCEL

For testing purpose, we used MS Excel to observe the results of application.

h) **Libraries:** ChatterBot, NLTK, PyNLPI

ChatterBot:

As the name suggests, chatterbot is a python library specifically designed to generate chatbots. This algorithm uses a selection of machine learning algorithms to fabricate varying responses to users as per their requests. Chatterbot makes it easier to develop chatbots that can engage in conversations.

NLTK:

The **Natural Language Toolkit**, or more commonly **NLTK**, is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for English written in the Python programming language.

PyNLPI:

PyNLPI, pronounced as ‘pineapple’, is a Python library for Natural Language Processing. It contains various modules useful for common, and less common, NLP tasks. PyNLPI can be used for basic tasks such as the extraction of n-grams and frequency lists, and to build simple language model. There are also more complex data types and algorithms.

3.2 Hardware Requirement

PROCESSOR : Min Intel i5 and above.

HARD DISK : Min 128 GB.

RAM : Min 512 Mb and above

Steps to install Software

Python:

You can install Python on your Windows server or local machine in just a few easy steps

Prerequisites:

- A system running Windows 10 with admin privileges
- Command Prompt (comes with Windows by default)
- A Remote Desktop Connection app (use if you are installing Python on a remote Windows server)

Steps:

Step 1: Select Version of Python to Install.

Step 2: Download Python Executable Installer.

Step 3: Run Executable Installer.

Step 4: Verify Python Was Installed on Windows.

Step 5: Verify Pip Was Installed.

Step 6: Add Python Path to Environment Variables (Optional)

Flask:

Step 1: Install Virtual Environment. Install Flask in a virtual environment to avoid problems with conflicting libraries. ...

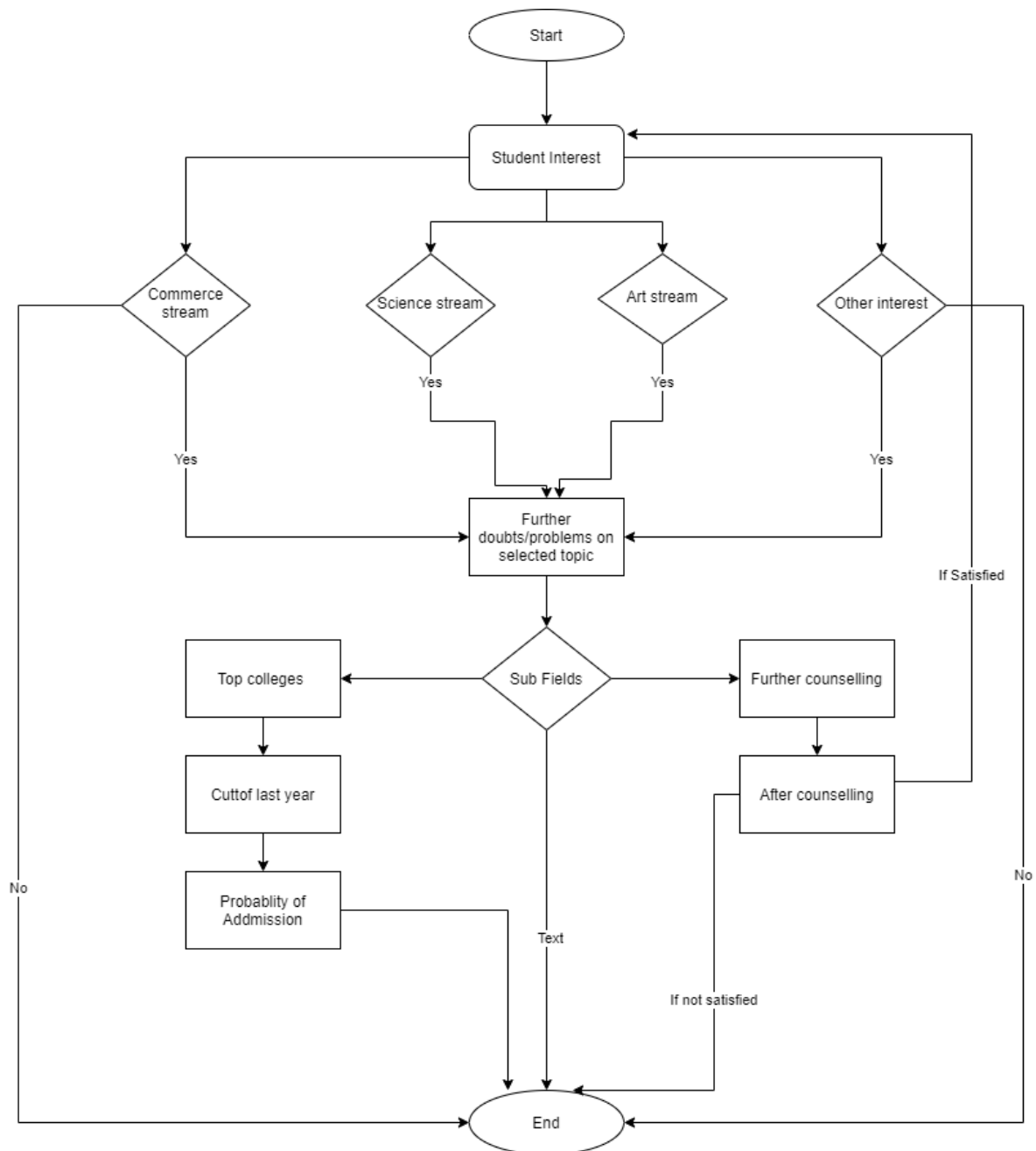
Step 2: Create an Environment. Make a separate directory for your project: mkdir <project name> ...

Step 3: Activate the Environment. ...

Step 4: Install Flask. ...

Step 5: Test the Development Environment.

3.3 Flowchart



4. Steps to Run the Project

- Requirements for running our project are:
 - Flask \geq 0.11
 - chatterbot \geq 0.7.1
 - SQLAlchemy \geq 1.1.11
- Open our file in any of the reputed IDE
- For that we have used vscode as an IDE for running this project

- First, we need to import the following libraries like Flask, chatterbot, chatterbot.trainer
- Then import the project file in the IDE
- Then search for the app.py file
- Simply run that file in your ide with command "python app.py"
- After you run the file, then you have a local server address.
- By directing to that address, you can see the Web Page of our Chatbot.

5. Future Scope:

○ **Personality Analysis:**

We will include the feature of Personality analysis option based on their Psychometric Test profile. In this first, Student must give the Psychometric Test and on basis of that or on different traits bot will evaluate their profile.

○ **Factors based Analysis:**

We will analyse the different-different factors to choose the career track or stream chosen by student. On what factors does the student have to choose right career track for themselves.

○ **Guidance for mental health:**

We will include the feature of Analysis of mental health of students. Many students are currently dealing with Depression, Anxiety, and confusion to choose stream.

○ **Voice over feature:**

We will include the Voice over feature to our chatbot to make it easy to use and friendly for disabled person.

- **Limitations:**

- Sometimes our chatbot did not reply with correct formatted sequence.
- Often it gives answers in irrelevant context.
- Our chatbot oftentimes unable to extract entities or keywords from input queries.