PROJECT REPORT ON

COLLEGE QUERY CHATBOT

SUBMITTED BY:

EBENZER VANLALTHLANAWMI

ROLL NO. 2223BCA003

MH LALHRUAIZELI

ROLL NO. 2223BCA014

Under the guidance of

Mr H. Lalruatkima

Assistant Professor

BACHELOR OF COMPUTER APPLICATION

(HATIM)

MIZORAM UNIVERSITY, AIZAWL 2024

HIGHER AND TECHNICAL INSTITUTE, MIZORAM

KAWMZAWL, LUNGLEI – 796701

CERTIFICATE

This is to certify that Ebenezer Vanlalthlanawmi and MH. Lalhruaizeli of Group 2 has fully completed the project entitled, “COLLEGE QUERY CHATBOT” in order to meet the requirement of the Mizoram University for the V Semester Bachelor of Computer Application in the year 2024(JULY - DECEMBER). It is to certify that all the corrections/ suggestions indicated for internal assessments has been incorporated in the project. The project report has been approved as it satisfies the academic requirements in respects of the project work prescribed for the BCA Course.

H. LALRUATKIMA	**VUANSANGA VANCHHAWNG**	**K. LALMUANPUIA**
Project Guide	Principal	Head of Department
Dept. of Computer Science	H.A.T.I.M	Dept. of Computer Science

Name of Examiner

Signature of Examiner

_____

_____

Contents

Acknowledgements i

1. Introduction

1.1 Overview 1

1.2 Objectives 1

1.3 Scopes 1

2. System Requirements Specifications

2.1 Hardware Requirements 2

2.2 Software Requirements 2

3. Details of Hardware and Software

3.1 Details of Hardware 3

3.2 Details of Software 3

4. System Analysis

4.1 Entity Relationship Diagram 6

4.2 Data Flow Diagram 7

5. System Design and Implementation

5.1 User/Guest Side

5.1. Chatbot 8

6. Coding System

6.1 Back-end

6.1.1 Training.py 9

6.1.2 App.py 14

6.1.3 Data.json 18

6.2 Front-end

6.2.1 Style.css 27

6.2.2 Index.html 33

7. Drawbacks and Limitations 38

8. Future Enhancements and Planning 39

9. Conclusion 40

ACKNOWLEDGEMENT

First, we would like to give thanks to God for guiding us throughout the process of our project, giving us with good health so that we are able to get this far.

We would also like to give thanks to all our faculty member and for our computer science department teacher, our Principal Mr Vuansanga Vanchhawng, our head of Department K.Lalmuanpuia for giving us their full co-operation and support. We would also like to give thanks to all the people who help us and involve in the making of this project, it's mainly because of them that we are able to complete it in time.

We also would like to give a words of gratitude especially to our project invigilator, Sir H.Lalruatkima for guiding us from the beginning till the end, providing us with all the necessities required for our project helping us in the completion of our project. Last but not the least; we thank our parents for their moral support and encouragement.

1. INTRODUCTION

1.1 OVERVIEW

This project is mainly focus on creating an AI chatbot with a knowledge based on the college administration and other information. The user can inquiry the chatbot and it will generate a response based on the questions.

1.2 OBJECTIVE

The main objective of this project is to create user-friendly and easily accessed chatbot.

1. Our project is an AI chatbot that has extensive knowledge about the college administration and other information.

2. It can be trained to understand the context of user's queries and answer it.

1.3 SCOPE

The scope of this project is to provide information about the college and give easy access to the students and whoever is concerned like:

1. The course it offers.
2. The details of fee for each department.
3. Important details of each department.

2. SYSTEM REQUIREMENTS

2.1 Hardware Requirements

This project is designed in such a way that it can be implemented almost on any Computer system. Thus, the hardware requirement is not high.

(i) Color Monitor

(ii) Processor- Pentium IV and above

(iii) RAM 256 MB and above

(iv) Secondary memory (not more than 1 GB)

(v) QWERTY or IBM-compatible keyboard

(vi) Optical Mouse

2.2 Software Requirements

The software used for designing the user interface (front-end) and for the making of the server side (back-end) are as follows:

- PYTHON
- Flask
- NumPy
- NLTK
- CSS
- Keras
- Tensorflow
- Pickle

3. Details of Hardware and Software

3.1 Details of Hardware

The main details of the hardware and scripting language used to create this project are given below:

1. Color Monitor: A color monitor to view the normal output settings. A monitor is one of the most important output devices for a computer which displays all processes and applications done by the machine in the form of a Graphical User Interface or Command Line Interface.

2. Processor: The processor is the main processing unit used to run the project on the computer.

3. RAM: The main memory is where the application is stored while the process is running. For a project like this, the RAM need not be high; it can be run at full performance even with 128MB of memory.

4. Secondary memory: The secondary memory is the memory in which the application can be stored as a backup file in the computer.

5. QWERTY or IBM-compatible Keyboard: The Keyboard is the main input device to enter the necessary information and data.

3.2 Details of Software

1. Python

Python is a versatile, high-level programming language widely used for various applications, including web development, data analysis, machine learning, and automation. Known for its simplicity and readability, Python supports multiple programming paradigms such as object-oriented, procedural, and functional programming. It features an extensive standard library, enabling applications in a wide range of domains, from scientific computing to artificial intelligence (AI).

2. Flask

Flask is a micro web framework for Python, designed to make the development of web applications quick and easy. It is minimalistic, allowing developers to add

only the components they need by integrating extensions. Flask is particularly suited for small to medium-sized web applications, offering flexible routing, request handling, and templating. It can be extended to add features like database integration, form validation, and user authentication.

3. NumPy

NumPy (Numerical Python) is a core library for scientific computing with Python, providing powerful support for working with large, multi-dimensional arrays and matrices. It includes an array object that supports fast vectorised operations and a variety of mathematical, logical, and statistical functions. NumPy serves as the foundation for most other data science libraries, such as Pandas and SciPy, and is essential for data manipulation and numerical analysis.

4. NLTK

The Natural Language Toolkit (NLTK) is a Python library for processing and analysing human language data. It offers easy-to-use interfaces for tasks such as tokenization, stemming, lemmatization, part-of-speech tagging, and named entity recognition. NLTK is widely used in natural language processing (NLP) applications such as sentiment analysis, text classification, and information retrieval, providing developers with tools and datasets to handle text data efficiently.

5. CSS (Cascading Style Sheets)

CSS is a stylesheet language used to define the presentation and layout of web pages. It allows developers to control the appearance of HTML elements, including fonts, colors, spacing, and positioning. CSS facilitates the separation of structure (HTML) from design, ensuring better maintainability and flexibility in web development. It also supports responsive design techniques, enabling websites to adapt to various screen sizes and devices through media queries.

6. Keras

Keras is a high-level API for building and training deep-learning models in Python. It is designed for fast experimentation and allows for the easy creation of neural networks. Keras operates as an interface over lower-level frameworks like TensorFlow, providing a simplified and user-friendly approach to developing complex machine learning models. Keras supports a wide range of model types, including feedforward, convolutional, and recurrent neural networks, and can efficiently handle large datasets with GPU support.

7. TensorFlow

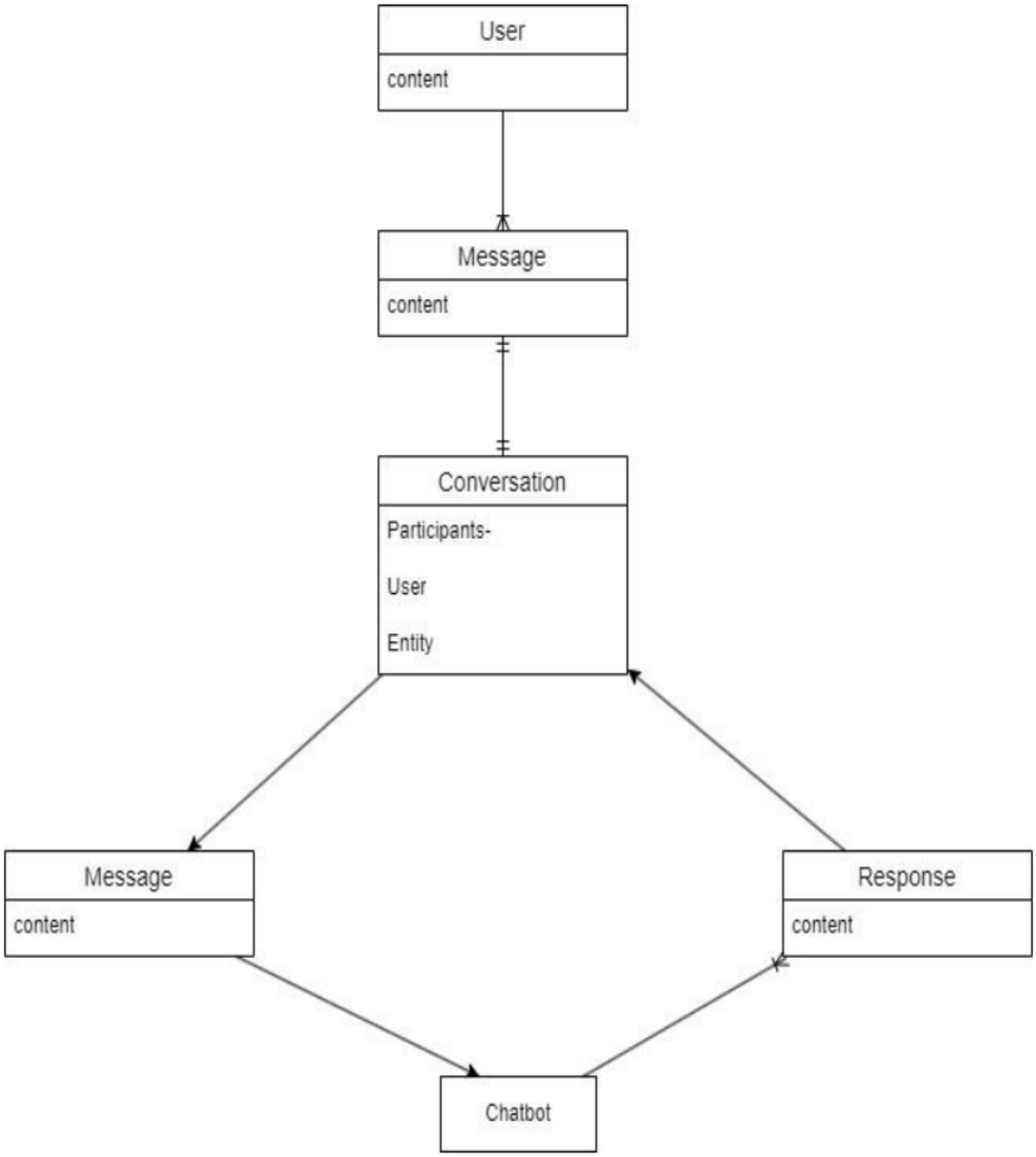
TensorFlow is an open-source machine learning framework developed by Google. It is widely used for building and deploying machine learning and deep learning models. TensorFlow provides a comprehensive ecosystem of tools, libraries, and community resources, allowing developers to design, train, and deploy models at scale. The framework supports both CPU and GPU computation, making it suitable for large-scale training of models, and integrates well with Keras for simplified model development.

8. Pickle

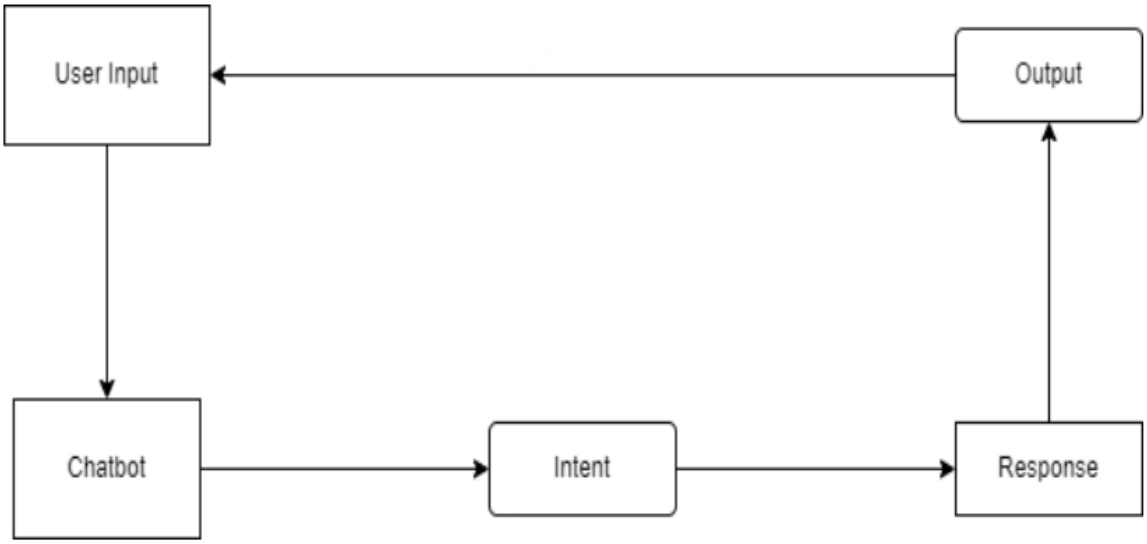
Pickle is a Python library used for serializing (pickling) and deserializing (unpickling) Python objects. Serialization refers to the process of converting a Python object into a byte stream, while deserialization converts the byte stream back into a Python object. Pickle is commonly used for saving machine learning models, program states, or any complex Python objects to disk, allowing for their storage and later retrieval without the need to recreate them from scratch.

4. System Analysis

4.1 Entity-Relationship Diagram



4.2 Data Flow Diagram



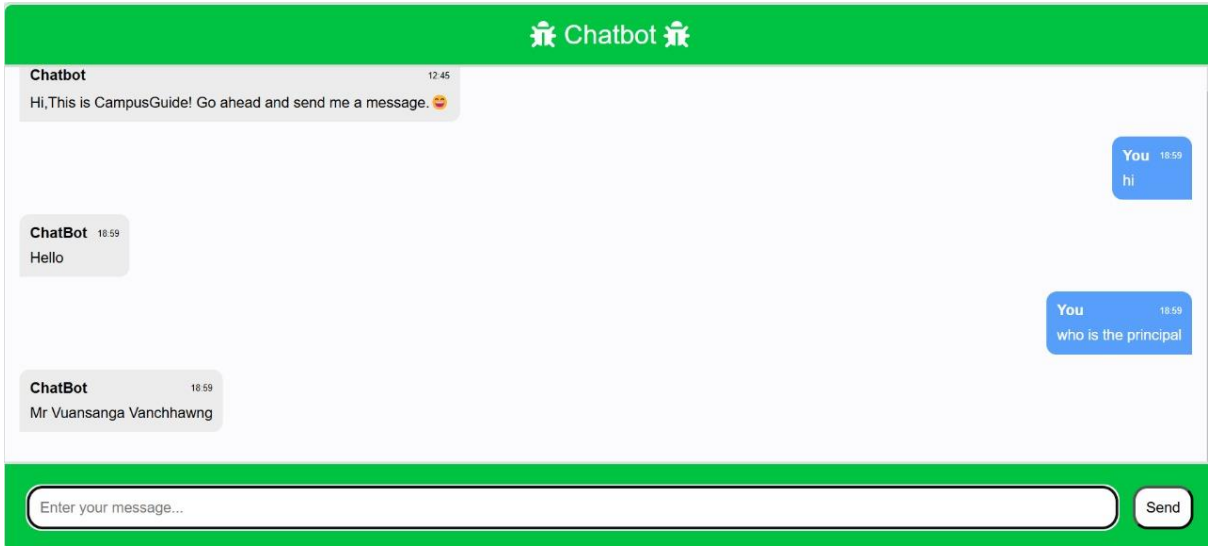
5. System Design and Implementation

5.1 USER SIDE CONTENT

5.1. CHATBOT

5.1 HOMEPAGE

Here the users would communicate with the Chatbot.



6. CODING SYSTEM

6.1 Back-end

6.1.1 training.py

```
import os
os.environ['TF_USE_LEGACY_KERAS'] = '0'

import nltk
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
import json
import pickle
import numpy as np
import tensorflow as tf
from tensorflow.python.keras.models import Sequential
from tensorflow.python.keras.models import Dense, Dropout
from tensorflow.python.keras.models  import SGD
import random

# Initialize variables
words = []
classes = []
documents = []
ignore_words = ['?', '!']

# Load and preprocess data
```

```
with open('data.json') as data_file:
    intents = json.load(data_file)

for intent in intents['intents']:
    for pattern in intent['patterns']:
        # Tokenize each word
        w = nltk.word_tokenize(pattern)
        words.extend(w)
        # Add documents in the corpus
        documents.append((w, intent['tag']))

        # Add to our classes list
        if intent['tag'] not in classes:
            classes.append(intent['tag'])

# Lemmatize and lower each word and remove duplicates
words = [lemmatizer.lemmatize(w.lower()) for w in words if w not in ignore_words]
words = sorted(list(set(words)))

# Sort classes
classes = sorted(list(set(classes)))

# Documents = combination between patterns and intents
print(len(documents), "documents")
print(len(classes), "classes", classes)
print(len(words), "unique lemmatized words", words)
```

```
# Save words and classes
pickle.dump(words, open('texts.pkl', 'wb'))
pickle.dump(classes, open('labels.pkl', 'wb'))

# Create training data
training = []
output_empty = [0] * len(classes)

for doc in documents:
    # Initialize bag of words
    bag = []
    pattern_words = doc[0]
    # Lemmatize each word
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in pattern_words]
    # Create bag of words array
    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)

    # Output is '0' for each tag and '1' for current tag
    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1
    training.append([bag, output_row])

# Shuffle and convert to np.array
random.shuffle(training)
```

```
training = np.array(training, dtype=object)

# Create train and test lists
train_x = list(training[:, 0])
train_y = list(training[:, 1])

# Ensure train_x is of the correct shape
# Determine the maximum length of sequences in train_x
max_length = max(len(x) for x in train_x)

# Pad each sequence to ensure uniform length
train_x = np.array([np.pad(x, (0, max_length - len(x))) for x in train_x])

# Convert train_y to a numpy array
train_y = np.array(train_y)

# Print shapes to verify
print("Shape of train_x:", train_x.shape)
print("Shape of train_y:", train_y.shape)

# Create model
model = Sequential()

model.add(Dense(128, input_shape=(train_x.shape[1],), activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
```

```
model.add(Dropout(0.5))
model.add(Dense(len(classes), activation='softmax'))


# Compile model
sgd = tf.keras.optimizers.legacy.SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
model.summary()
# Fit and save the model
hist = model.fit(train_x, train_y, epochs=200, batch_size=5, verbose=1)
model.save('model.h5', hist)
print("Model created")
```

6.1.2 app.py

```
import requests
import nltk
nltk.download('popular')
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
import pickle
import numpy as np
import tensorflow as tf

model = tf.keras.models.load_model('model.h5')
import json
import random
intents = json.loads(open('data.json').read())
words = pickle.load(open('texts.pkl','rb'))
classes = pickle.load(open('labels.pkl','rb'))

def clean_up_sentence(sentence):
    # tokenize the pattern - split words into array
    sentence_words = nltk.word_tokenize(sentence)
    # stem each word - create short form for word
    sentence_words = [lemmatizer.lemmatize(word.lower()) for word in sentence_words]
    return sentence_words

# return bag of words array: 0 or 1 for each word in the bag that exists in the sentence
```

```
def bow(sentence, words, show_details=True):
    # tokenize the pattern
    sentence_words = clean_up_sentence(sentence)
    # bag of words - matrix of N words, vocabulary matrix
    bag = [0]*len(words)
    for s in sentence_words:
        for i,w in enumerate(words):
            if w == s:
                # assign 1 if current word is in the vocabulary position
                bag[i] = 1
                if show_details:
                    print ("found in bag: %s" % w)
    return(np.array(bag))

def predict_class(sentence, model):
    # filter out predictions below a threshold
    p = bow(sentence, words,show_details=False)
    res = model.predict(np.array([p]))[0]
    ERROR_THRESHOLD = 0.25
    results = [[i,r] for i,r in enumerate(res) if r>ERROR_THRESHOLD]
    # sort by strength of probability
    results.sort(key=lambda x: x[1], reverse=True)
    return_list = []
    for r in results:
        return_list.append({"intent": classes[r[0]], "probability": str(r[1])})
```

```
    return return_list

def getResponse(ints, intents_json):
    tag = ints[0]['intent']
    list_of_intents = intents_json['intents']
    for i in list_of_intents:
        if(i['tag']== tag):
            result = random.choice(i['responses'])
            break
    return result

def chatbot_response(msg):
    ints = predict_class(msg, model)
    res = getResponse(ints, intents)
    return res

from flask import Flask, render_template, request

app = Flask(__name__)
app.static_folder = 'static'

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

@app.route("/get")
```

```
def get_bot_response():
    userText = request.args.get('msg')
    return chatbot_response(userText)

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

6.1.3 data.json


```
{
   "intents": [
   {
      "tag": "greeting",
      "patterns": ["Hi there", "Hi", "Hello","Is anyone there?","Hey","Hola", "Hello", "Good day"],
      "responses": ["Hello", "Hi there, how can I help?","Good to see you again"],
      "context": [""]
   },
   {
      "tag": "goodbye",
      "patterns": ["Bye", "See you later", "Goodbye", "Nice chatting to you, bye", "Till next time"],
      "responses": ["See you!", "Have a nice day", "Bye! Come back again soon."],
      "context": [""]
   },
   {
      "tag": "thanks",
      "patterns": ["Thanks", "Thank you", "That's helpful", "Awesome, thanks", "Thanks for helping me"],
      "responses": ["Happy to help!", "Any time!", "My pleasure"],
      "context": [""]
   },
   {
      "tag": "noanswer",
      "patterns": [],
```

"responses": ["Sorry, can't understand you", "Please give me more info", "Not sure I understand"],

"context": [""]

},

{

"tag": "options",

"patterns": ["How you could help me?", "What you can do?", "What help you provide?", "How you can be helpful?", "What support is offered"],

"responses": ["I can guide you through the college administration", "Offering college information"],

"context": [""]

},

{

"tag": "principal",

"patterns": ["Who is the current principal?", "Who is the principal of the college?", "What is the name of the principal?"],

"responses": ["Mr Vuansanga Vanchhawng"],

"context": [""]

},

{

"tag": "vice_principal",

"patterns": ["Who is the current vice principal?", "What is the name of the vice principal?"],

"responses": ["R. Lalnunthara"],

"context": [""]

},

{

"tag": "course",

"patterns": ["Which course does the college offers?", "How many course does the college offer?", "Which course does the college offer?"],

"responses": ["BA(English, Philosophy, Psychology, Education, History), BCA, BSW, BCOM", "It offers six courses - BA(English, Philosophy, Psychology, Education, History), BCA, BCOm, BSW"],

"context": [""]

},

{

"tag": "bca_hod",

"patterns": ["Who is the head of department of computer science?", "Computer Science HOD", "bca HOD"],

"responses": ["Mr K. Lalmuanpuia"],

"context": [""]

},

{

"tag": "english",

"patterns": ["Who is the head of department of english?", "English Department HOD", "English HOD"],

"responses": ["HT.Zuali"],

"context": [""]

},

{

"tag": "commerce_hod",

"patterns": ["Who is the head of department of commerce?", "Commerce Department HOD", "Commerce HOD"],

"responses": ["Vankhawpuimawii"],

"context": [""]

},

{

"tag": "history_hod",

"patterns": ["Who is the head of department of history?", "History Department HOD", "History HOD"],

"responses": ["Benjamin Lalnunfima"],

"context": [""]

},

{

"tag": "philosophy_hod",

"patterns": ["Who is the head of department of philosophy?", "Philosophy Department HOD", "Philosophy HOD"],

"responses": ["Dr.John C.Lalduhsaka"],

"context": [""]

},

{

"tag": "psychology_hod",

"patterns": ["Who is the head of department of psychology?", "Psychology Department HOD", "Psychology HOD"],

"responses": ["Deborah Zonunpuii"],

"context": [""]

},

{

"tag": "education_hod",

"patterns": ["Who is the head of department of education?", "Education Department HOD", "Education HOD"],

"responses": ["Rebek Lalramtiami"],

"context": [""]

},

{

"tag": "bsw",

"patterns": ["Who is the head of department of bsw?", "Social Work Department HOD", "BSW HOD"],

"responses": ["Rosangliana Khiangte"],

"context": [""]

},

{

"tag": "college_year",

"patterns": ["On which year did the college is established?", "Which year did the college is established?", "In what year that they establish the college?"],

"responses": ["On 2007", "2007"],

"context": [""]

},

{

"tag": "first_principal",

"patterns": ["Who is the first principal of HATIM?", "Who is the first principal?"],

"responses": ["Madam Chawngthanpari", "Chawngthanpari", "Pi Chawngthanpari"],

"context": [""]

},

{

"tag": "eco_club",

"patterns": ["Who is the head of the eco club of HATIM?", "Who is the leader of the eco club?"],

```
        "responses": ["Lalruatfela Ralte", "Sir Rfa", "Tv Ruatfela"],
        "context": [""]
    },
    {
        "tag": "college_motto",
        "patterns": ["What is the motto of the college?", "Motto of Hatim"],
        "responses": ["Seek, Share, Serve"],
        "context": [""]
    },
    {
        "tag": "theme_song",
        "patterns": ["What is the theme song of HATIM?", "College theme song?"],
        "responses": ["How Great Thou Art"],
        "context": [""]
    },
    {
        "tag": "college_bible_verse",
        "patterns": ["What is the college bible verse of HATIM?", "College bible verse?"],
        "responses": ["(Romans 11:36) for from Him, and to Him and through Him are all things..."],
        "context": [""]
    },
    {
        "tag": "fees",
        "patterns": ["What is the amount of fees of each department?", "Fees for each course?"],
```

"responses": ["BCA: Rs 3300, BA: Rs 2800, BSW: Rs 3300, BCOM: Rs 2800 "],

"context": [""]

},

{

"tag": "fee_payment",

"patterns": ["When is the fee payment available?", "When can i pay fees?","Is there any restriction regarding fee payment?", "Fee payment deadline"],

"responses": ["From 1st - 15th day of every month. There will be late fees if it is not paid during these days."],

"context": [""]

},

{

"tag": "bca_faculty",

"patterns": ["Who are the faculty member of BCA?", "Name the BCA faculty member?","Name of BCA teachers?"],

"responses": ["K.Lalmuanpuia(HOD), R.Vanlalawmpuia, H.Lalruatkima, R.Lalruatfela, Joseph Lalhunmawia, H. Lalrinawma, N.Lalrochani"],

"context": [""]

},

{

"tag": "commerce_faculty",

"patterns": ["Who are the faculty member of BCom?", "Name the Bcom faculty member?","Name of Bcom teachers?"],

"responses": ["Vankhawpuimawii(HOD), C.Lalrinsangi, Vanlalkimliani, C.Lalnunkimi, Lalvenpuia, Ruth C.Lalhruaipuii"],

"context": [""]

},

```
{
    "tag": "bsw_faculty",
    "patterns": ["Who are the faculty member of BSW?", "Name the BSW faculty member?","Name of BSW teachers?"],
    "responses": ["Rosangliana Khiangte(HOD), C.Lalremtluangi, Dr.R.Lallianzela, RTC.Lalremruata, Rebecca LP Lalnunhlui"],
    "context": [""]
},
{
    "tag": "psychology_faculty",
    "patterns": ["Who are the faculty member of Psychology?", "Name the Psychology faculty member?","Name of Psychology teachers?"],
    "responses": ["Deborah Zonunpuii(HOD), H. Rinmuanawmi, Lalvenhimi Ralte, R. Lalfamkimi, Vanlalawmpuii Hmar"],
    "context": [""]
},
{
    "tag": "philosophy_faculty",
    "patterns": ["Who are the faculty member of Philosophy?", "Name the Philosophy faculty member?","Name of Philosophy teachers?"],
    "responses": ["Dr. John C. Lalduhsaka(HOD), Ludi Lalneihpuii, H.C. Lalhmangaihzuali, Erika Zorinpuii, H. Lalromawia"],
    "context": [""]
},
{
    "tag": "history_faculty",
    "patterns": ["Who are the faculty member of history?", "Name the history faculty member?","Name of history teachers?"],
```

"responses": ["Benjamin Lalnunfima(HOD), Lalruathlui, A. Lalremtluangi, Dr. Lalhminghlua, Linda Chongthu"],

"context": [""]

},

{

"tag": "english_faculty",

"patterns": ["Who are the faculty member of english?", "Name the english faculty member?","Name of english teachers?"],

"responses": ["HT. Zuali(HOD), Immanuel Lalramenkima, Remruatpuii, Hannah Lalnunpuii Khiangte, Samuel Malsawmkima, Janet Lawmsangzuali, Lalremliana Chhakchhuak"],

"context": [""]

},

{

"tag": "education_faculty",

"patterns": ["Who are the faculty member of education?", "Name the education faculty member?","Name of education teachers?"],

"responses": ["Rebek Lalramtiami(HOD), F. Lalnuntluangi, Scully R. Laldintharia, Malsawmsangi"],

"context": [""]

}

]

}

6.2 Front-end

6.2.1 Style.css

```
:root {
  --body-bg: linear-gradient(135deg, #f5f7fa 0%, #c3cfe2 100%);
```

```
  --msger-bg: #fff;
  --border: 2px solid #ddd;
  --left-msg-bg: #ececec;
  --right-msg-bg: #579ffb;
}
html {
  box-sizing: 12;
}

*,
*:before,
*:after {
  margin: 12;
  padding: 12;
  box-sizing: inherit;
}

body {
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
  background-image: var(--body-bg);
  font-family: Helvetica, sans-serif;
}
```

```
.msger {
  display: flex;
  flex-flow: column wrap;
  justify-content: space-between;
  width: 100%;
  max-width: 867px;
  margin: 25px 10px;
  height: calc(100% - 50px);
  border: var(--border);
  border-radius: 5px;
  background: var(--msger-bg);
  box-shadow: 0 15px 15px -5px rgba(0, 0, 0, 0.2);
}

.msger-header {
  /* display: flex; */
  font-size: large;
  justify-content: space-between;
  padding: 10px;
  text-align: center;
  border-top-left-radius: 5px;
  border-top-right-radius: 5px;
  border-bottom: var(--border);
  background: rgb(0, 196, 65);
  color: #ffffff;
}
```

```
.msger-chat {
  flex: 1;
  overflow-y: auto;
  padding: 10px;
}
.msger-chat::-webkit-scrollbar {
  width: 6px;
}
.msger-chat::-webkit-scrollbar-track {
  background: #ddd;
}
.msger-chat::-webkit-scrollbar-thumb {
  background: #bdbdbd;
}
.msg {
  display: flex;
  align-items: flex-end;
  margin-bottom: 10px;
}

.msg-bubble {
  max-width: 300px;
  padding: 7px;
  border-radius: 7px;
  background: var(--left-msg-bg);
```

```
}.msg-info {
  font-size: x-small;
  display: flex;
  justify-content: space-between;
  align-items: center;
  margin-bottom: 5px;
}
.msg-info-name {
  margin-right: 7px;
  font-weight: bold;
}
.msg-info-time {
  font-size: 0.50em;
}

.left-msg .msg-bubble {
  font-size: x-small;
  border-bottom-left-radius: 0;
}

.right-msg {
  flex-direction: row-reverse;
}
.right-msg .msg-bubble {
  font-size: x-small;
  background: var(--right-msg-bg);
```

```
  color: #fff;
  border-bottom-right-radius: 0;
}

.msger-inputarea {
  display: flex;
  padding: 15px;
  border-top: var(--border);
  border-bottom-left-radius: 5px;
  border-bottom-right-radius: 5px;
  background: rgb(0, 196, 65);
}
.msger-inputarea * {
  padding: 7px;
  border-top-left-radius: 50px;
  border-top-right-radius: 50px;
  border-bottom-left-radius: 50px;
  border-bottom-right-radius: 50px;
  border-radius: 10px;
  font-size: x-small;
}
.msger-input {
  flex: 5;
  background: #ffffff;
}
.msger-send-btn {
```

```
  margin-left: 10px;
  background: rgb(255, 255, 255);
  color: #000000;
  font-weight: normal;
  cursor: pointer;
  transition: background 0.23s;
}
.msger-send-btn:hover {
  background: rgb(21, 150, 255);
  color: #ffffff;
}

.msger-chat {
  background-color: #fcfcfe;
  background-image:none
}
```

6.2.2 Index.html

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
  <meta charset="UTF-8">
  <title>Chatbot</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet" href="{{ url_for('static', filename='styles/style.css') }}">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script
>
</head>

<body>
  <!-- partial:index.partial.html -->
  <section class="msger">
    <header class="msger-header">
      <div class="msger-header-title">
        <i class="fas fa-bug"></i> Chatbot <i class="fas fa-bug"></i>
      </div>
    </header>

    <main class="msger-chat">
      <div class="msg left-msg">
        <div                class="msg-img"                style="background-image:
url(https://image.flaticon.com/icons/svg/327/327779.svg)"></div>

        <div class="msg-bubble">
```

```
        <div class="msg-info">
          <div class="msg-info-name">Chatbot</div>
          <div class="msg-info-time">12:45</div>
        </div>

        <div class="msg-text">
          Hi,This is CampusGuide! Go ahead and send me a message.😄
        </div>
      </div>
    </div>

  </main>

  <form class="msger-inputarea">
    <input type="text" class="msger-input" id="textInput" placeholder="Enter your message...">
    <button type="submit" class="msger-send-btn">Send</button>
  </form>
 </section>
 <!-- partial -->
 <script src='https://use.fontawesome.com/releases/v5.0.13/js/all.js'></script>
 <script>
  const msgerForm = get(".msger-inputarea");
  const msgerInput = get(".msger-input");
  const msgerChat = get(".msger-chat");
  // Icons made by Freepik from www.flaticon.com
```

```
  const BOT_IMG = "https://image.flaticon.com/icons/svg/327/327779.svg";
  const PERSON_IMG = "https://image.flaticon.com/icons/svg/145/145867.svg";
  const BOT_NAME = "    ChatBot";
  const PERSON_NAME = "You";

  msgerForm.addEventListener("submit", event => {
    event.preventDefault();

    const msgText = msgerInput.value;
    if (!msgText) return;

    appendMessage(PERSON_NAME, PERSON_IMG, "right", msgText);
    msgerInput.value = "";
    botResponse(msgText);
  });

  function appendMessage(name, img, side, text) {
    //   Simple solution for small apps
    const msgHTML = `
<div class="msg ${side}-msg">
 <div class="msg-img" style="background-image: url(${img})"></div>

 <div class="msg-bubble">
  <div class="msg-info">
   <div class="msg-info-name">${name}</div>
   <div class="msg-info-time">${formatDate(new Date())}</div>
```

```
    </div>

    <div class="msg-text">${text}</div>
  </div>
</div>
`;

      msgerChat.insertAdjacentHTML("beforeend", msgHTML);
      msgerChat.scrollTop += 500;
    }

    function botResponse(rawText) {

      // Bot Response
      $.get("/get", { msg: rawText }).done(function (data) {
        console.log(rawText);
        console.log(data);
        const msgText = data;
        appendMessage(BOT_NAME, BOT_IMG, "left", msgText);

      });
    }
    // Utils
    function get(selector, root = document) {
      return root.querySelector(selector);
    }
```

```
    function formatDate(date) {
      const h = "0" + date.getHours();
      const m = "0" + date.getMinutes();
      return `${h.slice(-2)}:${m.slice(-2)}`;
    }
  </script>
</body>
</html>
```

7. Drawbacks and Limitations

This project has been undertaken with a careful analysis to meet the specified requirements for this minor project. While we have made efforts to address all aspects of the project, there are certain limitations due to our current level of knowledge, skills, and experience. As a result, the scope of the project is more focused and constrained compared to larger-scale implementations.

First and foremost, due to the lack of contextual understanding, the chatbot may not have the contextual understanding, which can result in inappropriate or inaccurate responses. This limitation may result in inaccurate or irrelevant responses. Additionally, the chatbot's performance heavily depends on the quality of the data it is trained on. Any issues with data quality can significantly impact the accuracy of its responses.

Another limitation is the chatbot's inability to handle highly complex questions or tasks, as it does not possess advanced critical thinking or reasoning capabilities. While the system is designed to provide useful and relevant answers, it cannot fully replicate human-level comprehension or decision-making.

Due to the constraints mentioned above, it was not possible to implement all the features initially planned for the chatbot. Some functionalities had to be deprioritized or postponed due to time and knowledge limitations. Nevertheless, the project still aims to provide a basic, functional chatbot that serves its intended purpose within the scope defined for this minor project.

8. Future Enhancement and Planning

As this project is carried out in a limited time and limit skill, there are many areas in which it can be upgraded. The layout can be upgraded accordingly an also the addition of some features and programs so that the system can function more accuracy, with more human intervention.

9. CONCLUSION

This minor project work is the compilations of our ideas, views and thoughts. In doing this minor project and in the development of our thoughts and ideas Group 2 of 5th Semester BCA, we have benefited a great deal from our interaction with our teachers and friends. We extend our sincere thanks to them.

We are deeply conscious of the fact that this project would neither have been undertaken nor pursued and completed but for the tremendous support that we received from Mr. H. Lalruatkima, our project guide and K. Lalmuanpuia Head of Department, who not only give us their full support but provide us with all kinds of necessities that we required in the project we are working. We would also like to place and record our sincere thanks and gratitude to our Principal, Mr. Vuansanga Vanchhawng, for extending his full support and consent to this project undertaken.

This minor project is done and presented in such a way that it can be understandable. Our firm conviction emboldened us to embark upon this minor project work. This is a very painstaking work; however, we tried our best to satisfy the needs of this minor project. With much efforts this minor project has come into being even though we are conscious of our limited knowledge and skill. But it would be our request that this minor project we have undertaken, be dealt with much consideration and acknowledgement.