TABLE OF CONTENTS

CHAPTER 1 : INTRODUCTION

1.1 BACKGROUND 3

1.2 OBJECTIVES 3

1.3 PURPOSE AND SCOPE

1.3.1 PURPOSE 4

1.3.2 SCOPE 4

1.5 ORGANIZATIONAL REPORT 5

CHAPTER 2 : SURVEY OF TECHNOLOGY

2.1 FRONT END 6

2.2 BACK END 9

CHAPTER 3 : REQUIREMENT & ANALYSIS PROBLEM DEFINATION

3.1 FLOW CHART 11

3.2 USE CASE DIAGRAM 13

3.3 SEQUENCE DIAGRAM 14

3.4 GANTT CHART 15

3.5 PERT DIAGRAM 16

CHAPTER 4: SYSTEM DESIGN

4.1 PSEUDO CODE 17

CHAPTER 5: RESULTS AND DISCUSSIONS

5.1 IMPLEMENTATION APPROACH 18

5.2 CODING DETAILS AND CODE EFFICIENCY 20

5.2.1 DATABASE CONNECTION 20

5.2.2 FLASK UI 21

5.2.3 BOT ACTIONS (RSA-NLU) 23

5.2.4 DATA TRAINING 26

5.3 TESTING APPROACH

5.3.1 ERROR HANDLING TESTING 27

5.3.2 EXECUTION TESTING 28

5.3.3 GRAPHICAL USER INTERFACE TESTING 29

5.3.4 COMPATIBILITY TESTING 32

5.3.5 PERFORMANCE TESTING 33

5.4 MODIFICATIONS 35

5.5 TEST CASES 36

5.6 SCREENSHOTS OF INPUT 38

CHAPTER 6: RESULTS AND DISCUSSIONS

6.1 TEST CASES 48

6.2 SCREENSHOTS OF OUTPUT 50

6.2.1 SCREENSHOTS OF DATABASE 59

6.3 TEST REPORT 64

CHAPTER 7: CONCLUSION

7.1 CONCLUSIONS 67

7.2 LIMITATIONS 69

7.3 FUTURE SCOPE 70

BIBLIOGRAPHY

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my teacher Mrs.Navneet Kaur Nagpal as well our principal Dr. Dyaneshwar Doke who gave us the golden opportunity to do this wonderful project on the topic Campus Assist Bot, which also helped me in doing a lot of research and I came to know about many more things. We have used all our knowledge of the past three years of 'IT' in implementing this project. While creating this project we came across many new things which enlightened us and which would definitely help us in our near future. I was privileged enough to complete this project in the given period of time and this was only possible due to the efforts and guidance of our coordinator Mrs.Archana Talekar and our project guide Mrs.Navneet Kaur Nagpal.

CHAPTER 1: INTRODUCTION

1.1 Background:

- Earlier in our college i.e. M.L.Dahanukar college no such system existed. As a result , many times due to wrong information or unavailability of the college staff at that moment it was difficult for new person in the premises to find the required place in the campus.
- This college/ campus assist Chatbot will work on the principle of Machine Learning.
- Machine learning is a field of computer science that uses statistical techniques to give computers ability to learn Big Data without being explicitly programmed.
- Machine learning explores the study and construction of algorithm that can learn from and make predictions on the data and such algorithms overcome strictly static program instructions.
- So with the help of this Chabot it would be very easy query and doubt solving and correct information would be provided.

1.2 Objectives:

- Machine learning is an application of Artificial Intelligence that provides system the ability to automatically learn and improve from the experience without being explicitly programmed.
- Intelligence demonstrated by machines in contrast to natural intelligence displayed by Humans and other animals.
- Colloquially the term ‘AI’ is applied when machines mimics ”cognitive” functions that humans mind such as learning and problem solving.
- We are going to develop a Chat bot which will answer the difficulties related to the college
- Any visitor should have basic idea about the college.

1.3.1 Purpose:

- The Chatbot will answer questions such as : ”How will an admission process should be carried out ?”. It will also answer the general queries of locating classrooms, canteen, college office, SFC office, college Gym etc.
- It also will provide with correct and proper information about the names office and coordinators of each stream.
- It will also provide information about the events, holidays and guest lectures.
- Previously the visitors used to disturb college staff for the queries, so this would be reduced after implementation of Chatbot.

1.3.2 Scope.

- The non-teaching and teaching staff faced a tedious job of answering the queries raised by the visitors.
- Due to which they had to leave their current work midway to attend the visitors and this scenario was at its peak while on going admission procedure.
- This problem and limitation can be solved with the help of this chatbot.
- The main aim of the project is to reduce the burden of staff members and it also can help college in becoming more techno-savvy.

1.5 Organization of Report:

- ❖ In the following chapters that is :

- Chapter2:
  - ➢ The survey of requirement of chatbot is done.
  - ➢ The questions asked to the users will be related to the need of installation of chatbot; whether they are going to use the chatbot.
  - ➢ This survey will be done of minimum 20 users.
  - ➢ This will help us to find out what the users actually want.

- Chapter3:
  - ➢ In the 3rd chapter the requirement will be taken into consideration for example which languages should be used in frontend and which database can be used as backend.
  - ➢ For developing the application python is the frontend and sql is the backend.
  - ➢ If problems arise while developing the application it should be divided and solved.
  - ➢ This chapter will also consist of Use-Case, Sequence, Activity and flowchart.
- Chapter 4:
  - ➢ In this chapter the designing part will be taken into consideration.
  - ➢ It will also consist of security issues for e.g.:- The user will not be allowed the admin's page.
  - ➢ It contains the basic user interface.
  - ➢ Pseudocode of the program.

CHAPTER 2: SURVEY OF TECHNOLOGY

2.1 **Frontend**

- Frontend of application is the visible part of it that user interacts with, It will analyze the code design & debugging applications along with ensuring a seen less user experience. There are particular set of technical skills tools plus knowledge based requirements for frontend development .The following programming language can be used :

1. Java

- Java is a programming language that produces software for multiple platforms.
- When a programmer writes a Java application, the compiled code (known as bytecode) runs on most operating systems (OS), including Windows, Linux and Mac OS.
- Java derives much of its syntax from the C and C++ programming languages.
- Java applets run in a Web browser with Java Virtual Machine (JVM), which translates Java bytecode into native processor instructions and allows indirect OS or platform program execution.
- JVM provides the majority of components needed to run bytecode, which is usually smaller than executable programs written through other programming languages. Bytecode cannot run if a system lacks required JVM.

2. C#

- C# is a hybrid of C and C++, it is a Microsoft programming language developed to compete with Sun's Java language.
- C# is an object-oriented programming language used with XML-based Web services on the .

- NET platform and designed for improving productivity in the development of Web applications.
- C# boasts type-safety, garbage collection, simplified type declarations, versioning and scalability support, and other features that make developing solutions faster and easier, especially for COM+ and Web services.
- Microsoft critics have pointed to the similarities between C# and Java.

3. Python

- Python is an interpreted, object-oriented programming language similar to PERL that has gained popularity because of its clear syntax and readability.
- Python is said to be relatively easy to learn and portable, meaning its statements can be interpreted in a number of operating systems, including UNIX-based systems, Mac OS, MS-DOS, OS/2, and various versions of Microsoft Windows 98 Python offers dynamic data type, ready-made class, and interfaces to many system calls and libraries.
- It can be extended, using the C or C++ language.

We have chosen Python as our frontend because of following points.

1. Python is an interpreted high-level programming language for general purpose programming.
2. Python is an open source software.
3. Python is meant to be an easily readable language it often uses English keywords where other language use punctuations.
4. Python has fewer syntactic exceptions & special cases.
5. Python has -
    a. Easy syntaxes
    b. Readability
    c. High-level language
    d. Widely supported

6.Python also has a strong community support, many third party packages are available and many package like, scikit, pandas are used to provide a seamless way to develop machine learning solutions.

2.2 BackEnd :

- Backend usually means the part that do work but the user is unaware of it or cannot be seen .
- Backend is basically how the site works, updates and changes. Backend usually contains database to handle the data generated during dynamic transactions.
- It indirectly supports the front end services which communicate with the required resources.
- Some of the backend are mentioned below.

1. **MongoDB :**

   It is an open source document database. It is a leading NoSQL database. MongoDB is highly scalable & performance oriented database .MongoDB provides a cross platform document oriented database that gives highly availability & easy scalability. MongoDB stores the data in unstructured format rather than going for structured format. MongoDB uses Json-like document with schematic structure, MongoDB provides various features like Indexing, Replication & sharding. MongoDB is great when the data is large.

2. **Oracle:**

   Management system produced & masked by Oracle Co-operation. It is database commonly used for running online transaction processing, data warehousing & mixed database workloads. It is multi-model database. Oracle Database licensed software Oracle provides various services like cloud services with database & offline database. It has high reliability & scalability. It helps in data mining & analysis of data using various algorithms.

3. **MySQL** – is an open source relational database management system. It is a combination of my & SQL the abbreviation for structured query Language. It was first released on 23 may 1995 & was developed with the help of C&C++. Basically it has a class platform features i.e. it can work on almost all platforms. It is widely used RDBMS in the market. It is most trusted and faster RDBMS software.

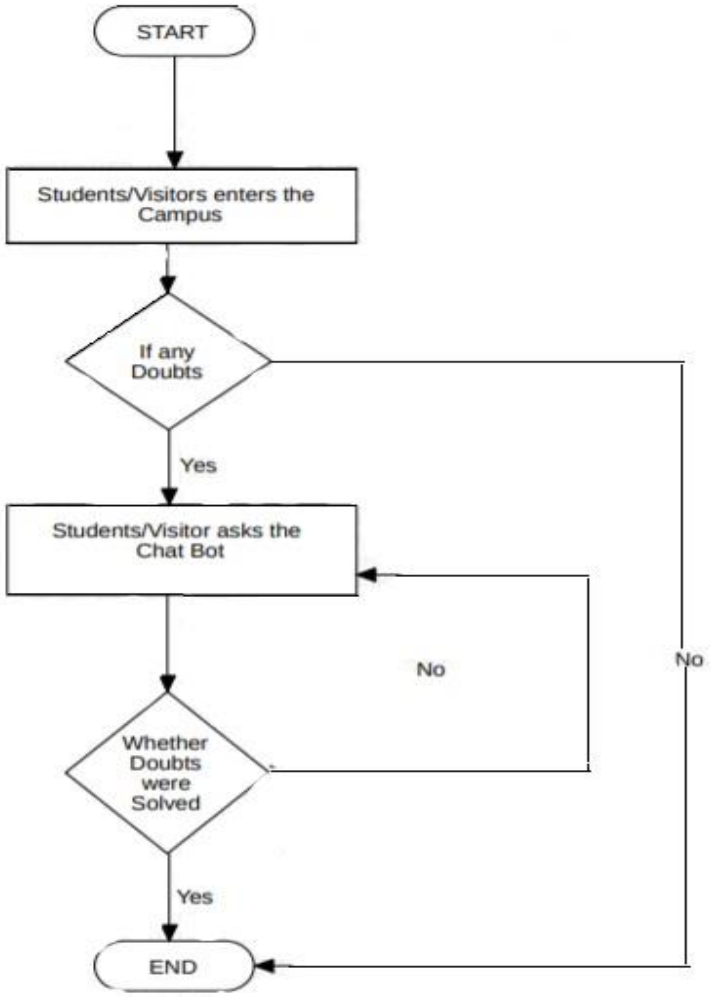
We have chosen MySQL as our backend for the following reasons –

1. MySQL server is one of the most widely used relational database management system.
2. Over 100 million copies of MySQL have been downloaded.
3. MySQL is an open source & freeware database server that provides numerous advanced database functionalities.
4. Google, Yahoo, Facebook, Nokia are famous organizations have employed MySQL.
5. MySQL is relatively easier to execute that MongoDB.
6. MySQL has these features –
    - Portability
    - Seamless connectivity
    - Security

CHAPTER 3 : REQUIREMENT & ANALYSIS

PROBLEM DEFINATION

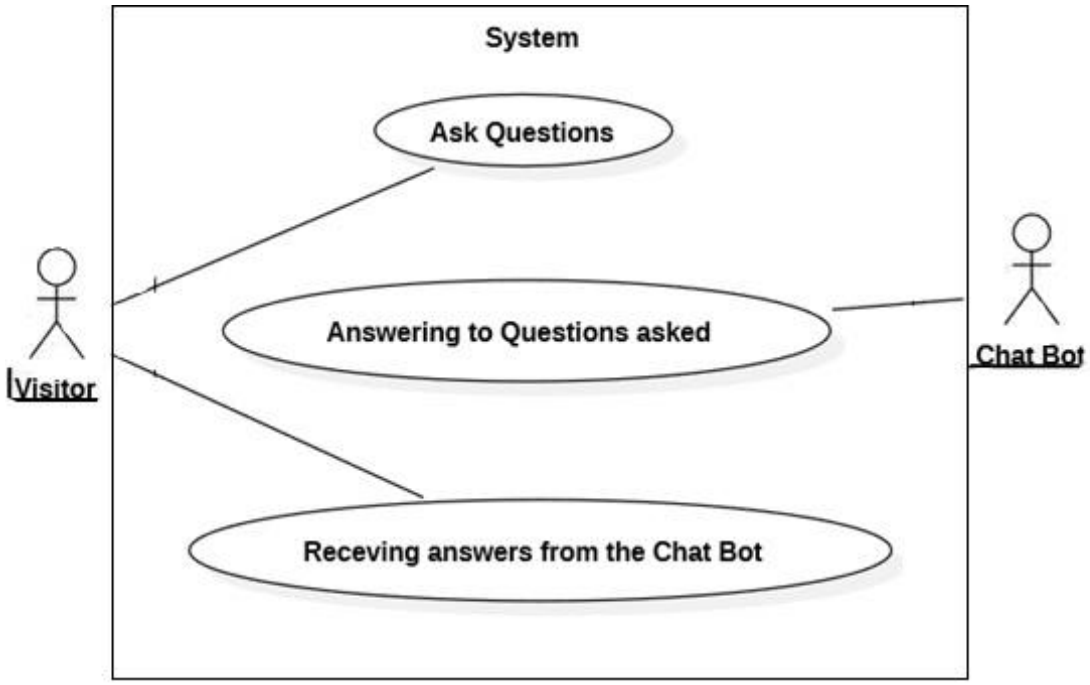
3.1 Flow Chart



EXPLANATION:

- Program starts
- Students or visitors enter the campus
- If they have any doubts they can ask the chatbot
- The communication will start after asking doubts to the chatbox
- The chatbot will ask them if their doubts were solved
- If Yes the communication will terminate
- If No the visitors/students can ask the question again or can ask another question

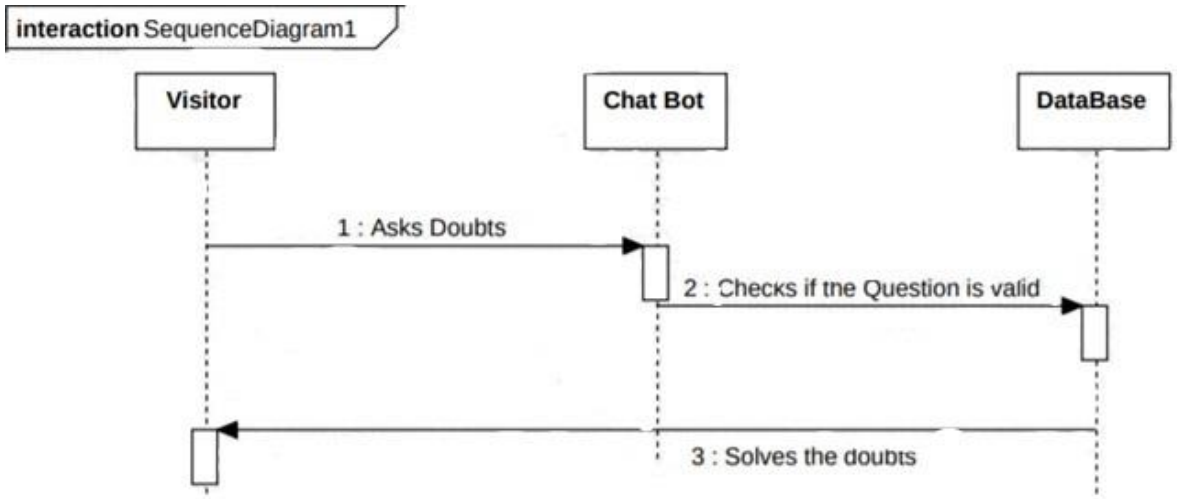
3.2 UseCase Diagram



EXPLANATION:

- The visitors will ask the question to the system
- The chatbot will answer the questions asked by the visitors
- The visitors will receive the answer from the chatbot

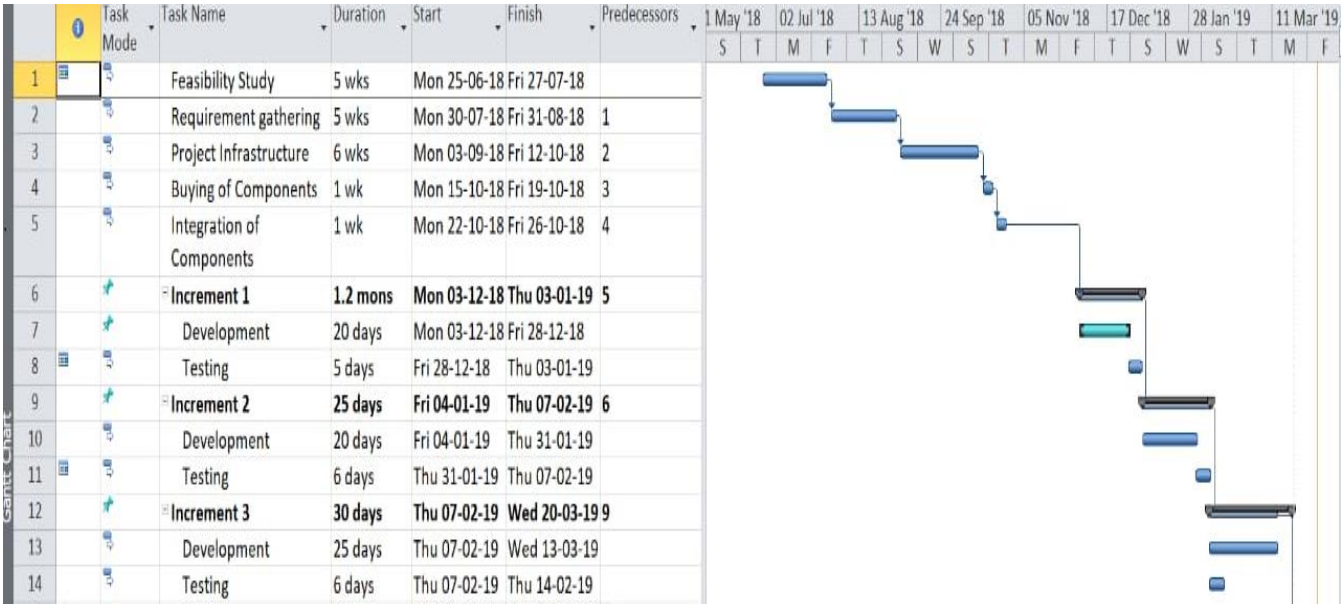
3.3 Sequence Diagram



EXPLANATION:

- The visitors will ask question to chatbot
- The chatbot will check with the database if the question is valid
- If the question is valid the chatbot will solve the doubt of the visitor

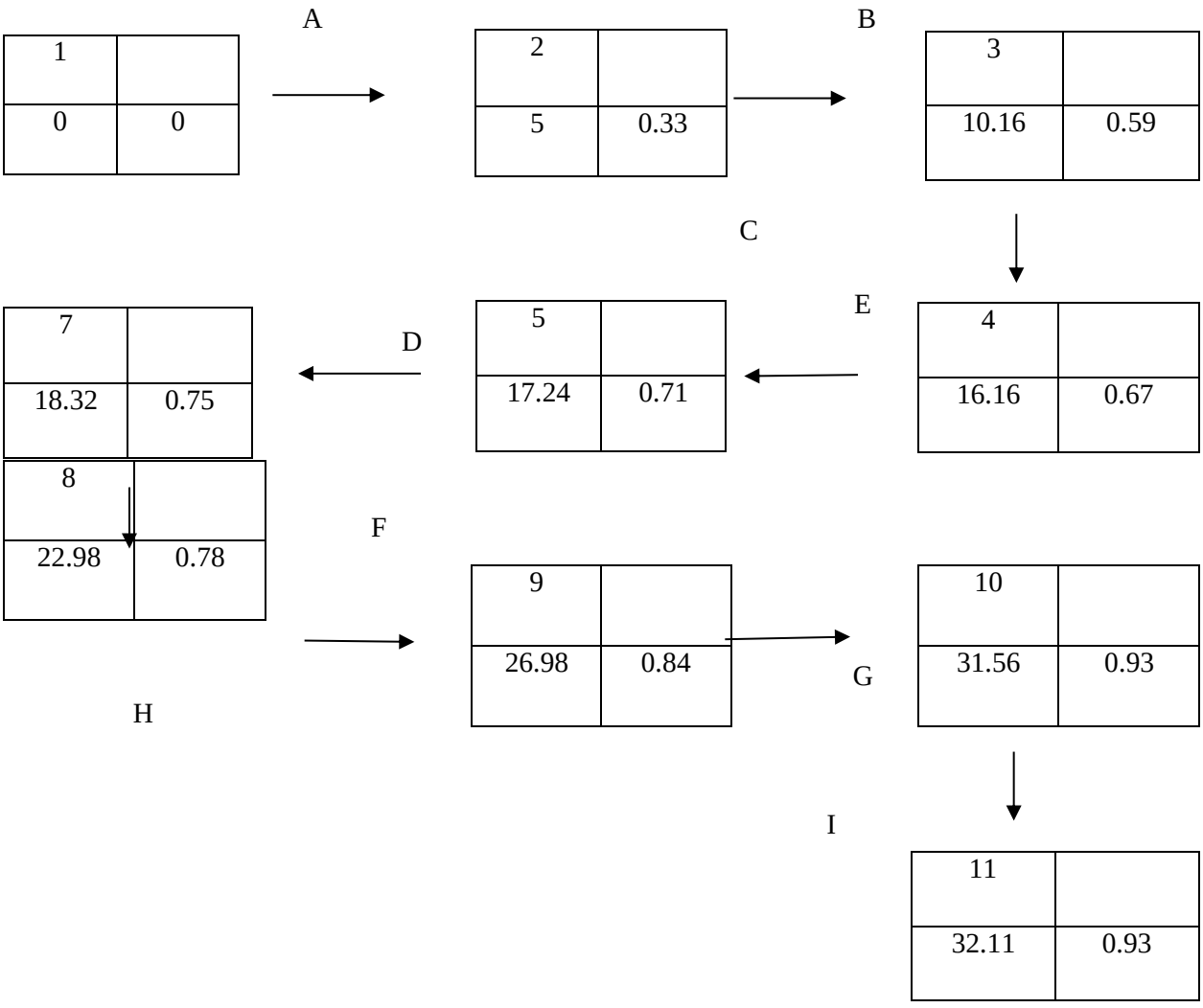
3.4 Gantt Chart



	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1		Feasibility Study	5 wks	Mon 25-06-18	Fri 27-07-18	
2		Requirement gathering	5 wks	Mon 30-07-18	Fri 31-08-18	1
3		Project Infrastructure	6 wks	Mon 03-09-18	Fri 12-10-18	2
4		Buying of Components	1 wk	Mon 15-10-18	Fri 19-10-18	3
5		Integration of Components	1 wk	Mon 22-10-18	Fri 26-10-18	4
6		**Increment 1**	**1.2 mons**	**Mon 03-12-18**	**Thu 03-01-19**	**5**
7		Development	20 days	Mon 03-12-18	Fri 28-12-18	
8		Testing	5 days	Fri 28-12-18	Thu 03-01-19	
9		**Increment 2**	**25 days**	**Fri 04-01-19**	**Thu 07-02-19**	**6**
10		Development	20 days	Fri 04-01-19	Thu 31-01-19	
11		Testing	6 days	Thu 31-01-19	Thu 07-02-19	
12		**Increment 3**	**30 days**	**Thu 07-02-19**	**Wed 20-03-19**	**9**
13		Development	25 days	Thu 07-02-19	Wed 13-03-19	
14		Testing	6 days	Thu 07-02-19	Thu 14-02-19	

3.5 Pert Diagram

4. Activity	Optimistic	Most Likely	Pessimistic	Expected	S. D
A	4	5	6	5	0.33
B	4	5	7	5.16	0.5
C	5	6	7	6	0.33
D	0.5	1	2	1.08	0.25
E	0.5	1	2	1.08	0.25
F	4	4.5	6	4.66	0.33
G	3	4	5	4	0.33
H	3.5	4.5	6	4.58	0.41
I	0.3	0.5	1	0.55	0.11



CHAPTER 4: System design

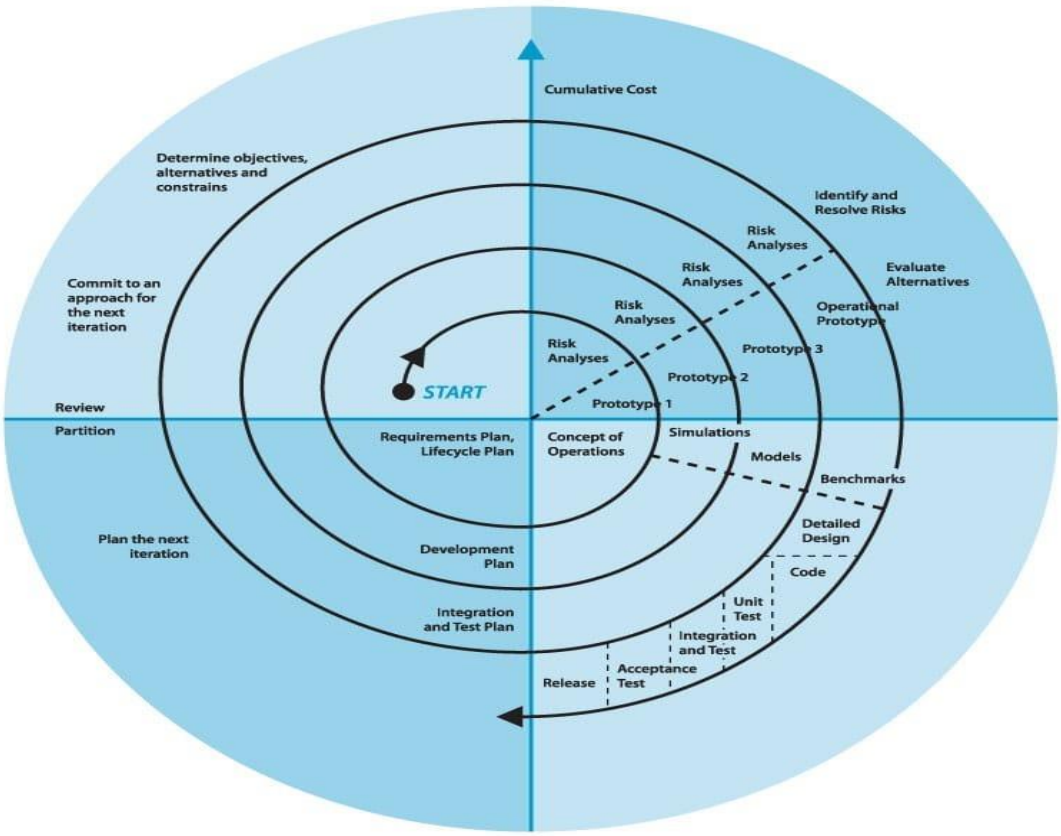
4.1 Pseudo Code:

1. Start
2. Initialize Chatbot
3. Accept Questions
4. If(Question valid)
5. Print Answer
6. Else
7. Print ask another question
8. Stop.

Chapter 5: Results and Discussions

5.1 Implementation Approach.

SPIRAL MODEL: -



- Spiral Model is a combination of a waterfall model and iterative model. Each phase in spiral model begins with a design goal and ends with the client reviewing the progress. The spiral model was first mentioned by Barry Boehm in his 1986 paper.
- The development team in Spiral-SDLC model starts with a small set of requirements and goes through each development phase for those set of requirements. The software engineering team adds functionality for the additional requirement in every-increasing spirals until the application is ready for the production phase.
- The development team in Spiral-SDLC model starts with a small set of requirements and goes through each development phase for those set of requirements. The software engineering team adds functionality for the additional requirement in every-increasing spirals until the application is ready for the production phase.
- Spiral model is used while implementation of the project because of its advantage and feasibility of moving back to the stages which are already implemented.

- Whatever the part of software development lifecycle it may be at any stage it is feasible to move back to the stages which are already done and do the modifications if necessary.

Planning: -

- It includes estimating the cost, schedule and resources for the iteration. It also involves understanding the system requirements for continuous communication between the system analyst and the customer.

Risk Analysis: -

- Identification of potential risk is done while risk mitigation strategy is planned and finalized.

Engineering: -

- It includes testing, coding and deploying software at the customer site.

Evaluation: -

- Evaluation of software by the customer. Also, includes identifying and monitoring risks such as schedule slippage and cost overrun.

Advantages: -

- Spiral Model mostly concentrates on risk analysis.
- Spiral Model used if requirement changing frequently.
- Customer evaluation phase made this model useful.
- Most useful for large and risk projects.
- Focused model for all phases.
- Most suitable for Real-Time Systems.

5.2 Coding Details and Code Efficiency

5.2.1 Data Base connector

```
import pyodbc

class BotDB(object):
    '''
    classdocs
    '''
    def __init__(self):
        '''
        Constructor
        '''
        self.DRIVER="ODBC Driver 13 for SQL Server"
        self.SERVER="desktop-du0svca\sqlexpress"
        self.DATABASE="Campus Assist Bot Final"

    def connect(self):
        cnxn = pyodbc.connect("Driver="+self.DRIVER+";"
                    "Server="+self.SERVER+";"
                    "Database="+self.DATABASE+";"
                    "Trusted_Connection=yes;")


        cursor = cnxn.cursor()
        return cursor

if __name__ == '__main__':
    b = BotDB()
    c = b.connect()
    c.execute('SELECT * from DateDetails')
    for r in c:
        print(r)
```

5.2.2 Flask UI

```
from flask import Flask,request,render_template,session, redirect
import os
from rasa_nlu.model import Metadata, Interpreter
from Dialog import BotActions
app=Flask(__name__)

interpreter =
Interpreter.load(r'D:\Python\Workspace\CollegeChatBot\CampusAssist\src\Training\models\nlu\
default\CampusNLU')

@app.route('/')
def main():
    return render_template('chatwindow.html')

@app.route('/process',methods=['POST'])
def process():
    user_input=request.form['user_input']
    bot_response=''
    global interpreter
    print(user_input)
    statementObj = interpreter.parse(user_input)
    if statementObj["intent"]["name"] == "person_enquiry":
            bot_response=BotActions.ActionPerson().run(statementObj)
    elif statementObj["intent"]["name"] == "location_search":
            print("Entering location Search")
            bot_response=BotActions.ActionLocation().run(statementObj)
    elif statementObj["intent"]["name"] == "dates_enquiry":
            bot_response=BotActions.ActionDates().run(statementObj)
    elif statementObj["intent"]["name"] == "greet":
            bot_response="Hello, How may I help you?"
    elif statementObj["intent"]["name"] == "thankyou":
            bot_response="My Pleasure. Is there anything else I can assist you with?"

    return render_template('chatwindow.html',user_input=user_input,
        bot_response=bot_response
        )

def run_nlu(message):
    interpreter = Interpreter.load('./models/nlu/default/CampusNLU')
    return interpreter.parse(message)

if __name__ == '__main__':
    app.secret_key = os.urandom(12)
    app.run(debug=True)
```

```
'''
print("BOT: Welcome, how may I assist you?")
while msg != "exit()":
    msg = input("YOU: ")
    statementObj = interpreter.parse(msg)
    if statementObj["intent"]["name"] == "person_enquiry":
        BotActions.ActionPerson().run(statementObj)
print("BOT: Good Bye!!")
'''
```

5.2.3 Botactions(rsanlu):

```
from rasa_core.actions.action import Action
from rasa_core.events import SlotSet
from Dialog import DatabaseConnector as DB
from _ast import Try
#from cytoolz.itertoolz import result

class ActionLocation(object):
    '''
    classdocs
    '''
    def __init__(self):
        pass

    def name(self):
        return "Location Search Action"

    def run(self,statementObj):
        try:
            print(statementObj)
            entity={}
            for s in statementObj["entities"]:
                entity[s["entity"]]=s["value"]

            print(entity)

            cursor = DB.BotDB().connect()
            if ("Academic_Year" in entity.keys()) and ("course" in entity.keys()):
                print("SELECT Blocknumber FROM TimeTableDetails WHERE
Coursename='"+entity["course"]+"' AND Year="+entity["Academic_Year"])
                cursor.execute("SELECT Blocknumber FROM TimeTableDetails WHERE
Coursename='"+entity["course"]+"' AND Year="+entity["Academic_Year"])
            else:
                print("SELECT Blocknumber FROM LocationDetails WHERE
Type='"+entity["location"]+"'")
                cursor.execute("SELECT Blocknumber FROM LocationDetails WHERE
Type='"+entity["location"]+"'")
            op=[]
            for row in cursor:
                print(row[0])
                op.append(row[0])

            return("Block no-> "+op[0])
```

```
        except Exception as e:
            return("I did not get the question")
            print(str(e))



class ActionPerson(object):
    '''
    classdocs
    '''
    def __init__(self):
        pass

    def name(self):
        return "Location Search Action"

    def run(self,statementObj):
        try:
            print(statementObj)
            intent = statementObj["intent"]["name"]
            mandatory = {"Teacher_Head":""}

            for ent in statementObj["entities"]:
                mandatory[ent['entity']] = ent['value']

            var=""

            if "course" in mandatory:
                var=var+" AND Coursename='"+mandatory["course"]+"' "

            cursor = DB.BotDB().connect()

            cursor.execute("SELECT Employeename FROM EmployeeDetails WHERE
Designation='"+mandatory["Teacher_Head"]+"'"+var)
            print("SELECT Employeename FROM EmployeeDetails WHERE
Designation='"+mandatory["Teacher_Head"]+"'"+var)
            op=[]
            for row in cursor:
                print(row[0])
                op.append(row[0])

            return(op[0])

        except Exception as e:
```

```
            return("I did not get the question")
            print(str(e))

class ActionDates(object):
    '''
    classdocs
    '''
    def __init__(self):
        pass

    def name(self):
        return "Dates enquire"

    def run(self,statementObj):
        try:
            print(statementObj)
            intent = statementObj["intent"]["name"]
            mandatory = {"Event":""}

            for ent in statementObj["entities"]:
                print(ent)
                mandatory[ent['entity']] = ent['value']

            var=""

            #if "course" in mandatory:
             #   var=var+" AND Coursename='"+mandatory["course"]+"' "

            cursor = DB.BotDB().connect()

            cursor.execute("SELECT Dates FROM DateDetails WHERE
Event='"+mandatory["event"]+"'"+var)
            print("SELECT Dates FROM DateDetails WHERE
Event='"+mandatory["event"]+"'"+var)
            op=[]
            for row in cursor:
                print(row[0])
                op.append(row[0])

            return(op[0])
        except Exception as e:
            return("I did not get the question")
            print(str(e))
```

5.2.4 Data Training

```
from rasa_nlu.training_data import load_data
from rasa_nlu import config
#from rasa_nlu.config import RasaNLUConfig
from rasa_nlu.model import Trainer
from rasa_nlu.model import Metadata, Interpreter

def train_nlu(data, config_file, model_dir):
    training_data = load_data(data)
    trainer = Trainer(config.load(config_file))
    trainer.train(training_data)
    model_directory = trainer.persist(model_dir, fixed_model_name = 'CampusNLU')

def run_nlu(message):
    #interpreter = Interpreter.load('./models/nlu/default/CampusNLU',
RasaNLUConfig('config_spacy.json'))
    interpreter = Interpreter.load('./models/nlu/default/CampusNLU')
    return interpreter.parse(message)


if __name__ == '__main__':
    data = "TrainData//nlu.md"
    config_file = "TrainData//nlu_config.yml"
    model_dir = "./models/nlu"

    ''' Used for Training the BOT using nlu.md'''
    train_nlu(data, config_file, model_dir)

    print("----- Started ------")
    message="Where is BMS classroom."
    intentData = run_nlu(message)
    print(intentData)
    #intent = intentData['intent']['name']
    #entity = intentData['entities']['entity']
    #value = intentData['entities']['value']
    #print(intent)
    #if intent == "location_search":
        #ActionLocation().run('', '', '','Bscit','Ty')


if __name__=="__main__":
    ActionLocation().run('', '','')
```

5.3 Testing Approach

5.3.1 Error handling testing:

- When normal users are working with the system, it may be possible that they may enter wrong data or select wrong options.
- Application is expected to help the user through error messages, if anything unexpected happened with the system.
- A run-time error takes place during the execution of a program, and usually happens because of adverse system parameters or invalid input data.
- An example is the lack of sufficient memory to run an application or a memory conflict with another program.
- In our project while coding we have included several ‘try’ and ‘catch’ block of codes which are ‘class’ specific and are capable of handling errors as per the requirement of that specific module.
- For e.g.: try:

```
        print(statementObj)
        intent = statementObj["intent"]["name"]
        mandatory = {"Teacher_Head":""}

        for ent in statementObj["entities"]:
            mandatory[ent['entity']] = ent['value']

        var=""

        if "course" in mandatory:
            var=var+" AND Coursename='"+mandatory["course"]+"' "

        cursor = DB.BotDB().connect()

        cursor.execute("SELECT Employeename FROM EmployeeDetails WHERE
Designation='"+mandatory["Teacher_Head"]+"'"+var)
        print("SELECT Employeename FROM EmployeeDetails WHERE
Designation='"+mandatory["Teacher_Head"]+"'"+var)
        op=[]
        for row in cursor:
            print(row[0])
            op.append(row[0])

        return(op[0])

    except Exception as e:
        return("I did not get the question")

        print(str(e))
```

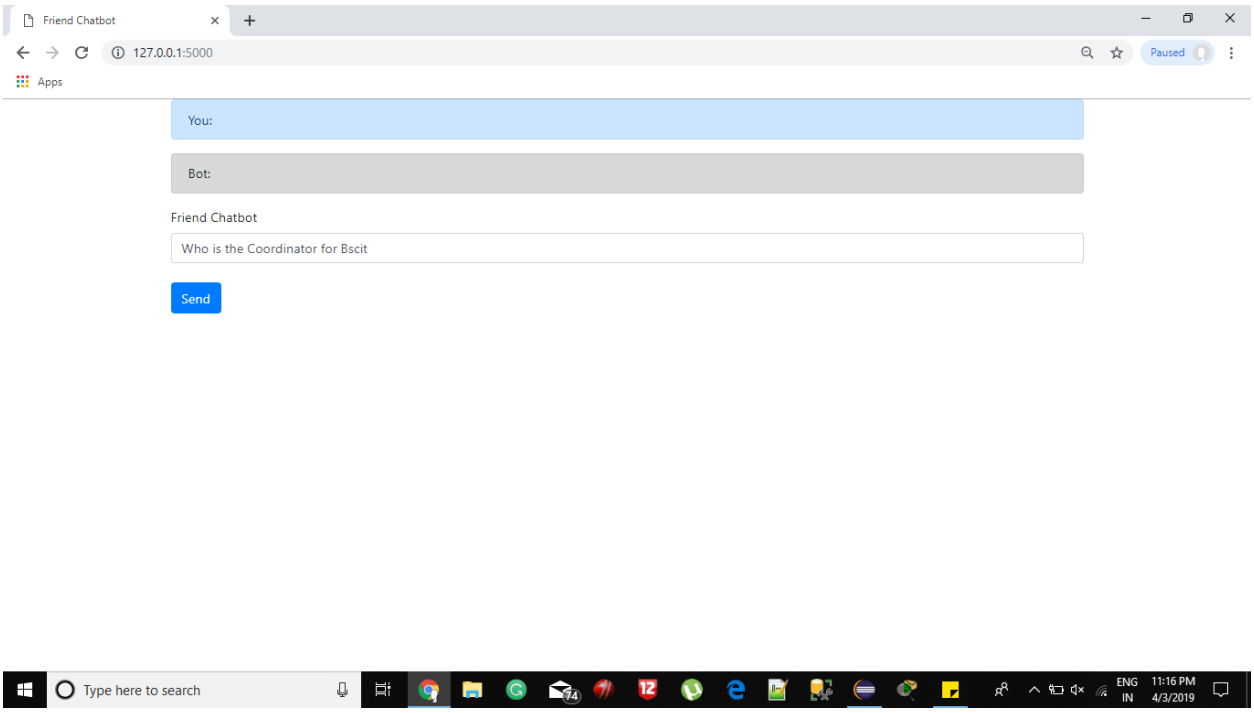
- Like if user inputs ‘Who is the coordinator of BBA?’ the chatbot will throw exception ‘I did not get the question’ because our college doesn’t have BBA course in the curriculum.
- This error handling also solves the problem of wrong data entry through preventive messages i.e. when a user tries to enter the wrong data system identifies wrong entry and prevents such entry by throwing an error message.

5.3.2 Execution Testing.

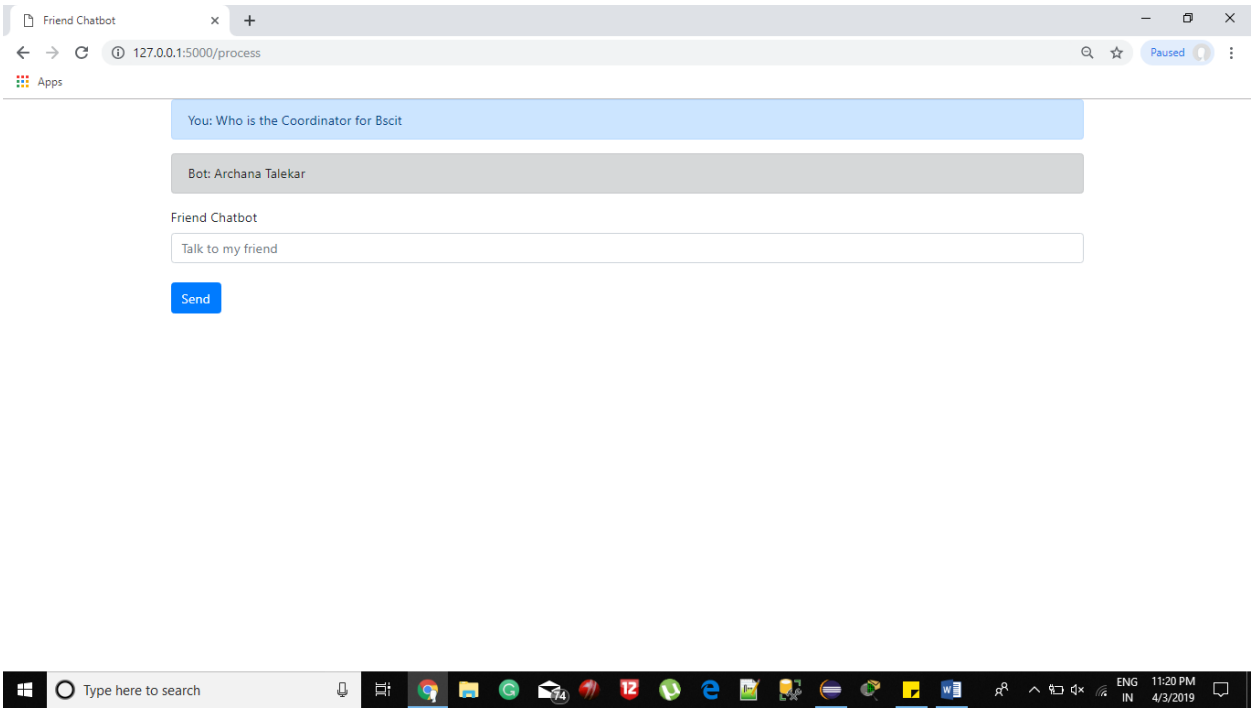
- Execution testing is performed to ensure that system achieves desired level of proficiency in production environment when normal users are using it in normal circumstances
- Execution Testing is successfully carried out when the system objectives defined in the requirements are met and the same answer is thrown as the actual output expected by the user.
- **Test Execution** is an extension of Software **Testing** Life **Cycle** and is said to be the most important and “happening” part of Software **Testing** Life **Cycle** (STLC) and the entire software development.
- It is in short describes as the process of **executing** the code and comparing the expected and actual results.
- It may be considered as alpha testing if it is done in development environment , or Beta testing if done in user environment.
- In our Chatbot our basic aim of the project was when an user inputs any query or a question the chat bot should give a correct response.
- Example:- SS
- As you can see in the above example the user inputs ‘who is the coordinator of BscIt?’ and as stated in the requirement and development phase chatbot correctly responds ‘Archana Talekar’.

5.3.3 Graphical user interface testing

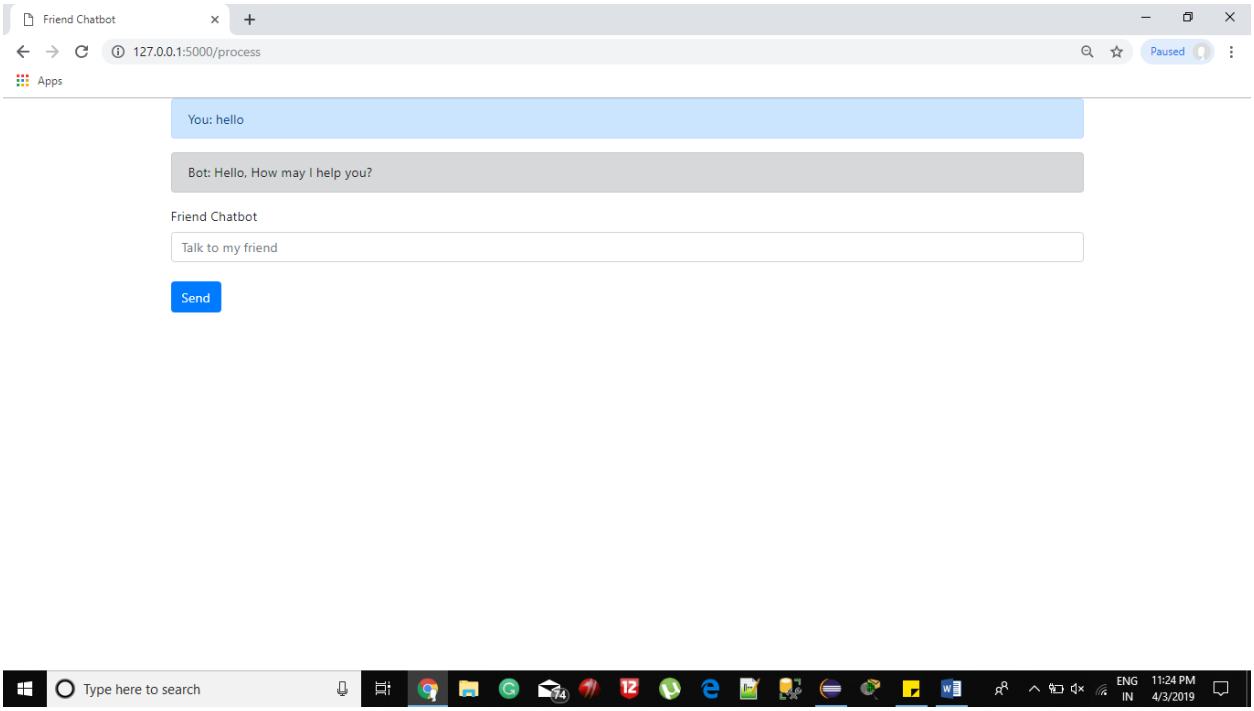
- Graphical user interface is the most important part of the application along with the functionality, as it may have a effect on usability.
- Generally, application system testing starts with functionality testing and is followed by graphical user interface testing.
- Graphical user Interface testing is also known as ‘GUI’ testing or ‘UI’ testing.
- Graphical user interface testing has major impact on the users.
- User interface should be simple and easily understandable by the user.
- Complex user interface may lead to problem for the users as not every user is professional or trained to use a specific software or application.
- This user interface should always be simple and easily understandable by common people.
- In this project while designing the User interface if chatbot we considered all the aspects and designed it very simple so as the students ,parents and other common public can access the chatbot easily and can get their work done with help of this chatbot.
- Minimal use of buttons for less confusion can be seen in this chatbot’s user interface for smooth and hassle free working for users.
- As the background colour scheme of the interface, colour combinations, brightness of the screen, content on the screen, font used etc matters a lot we have particularly selected cool colours in the background such as blue, grey, white etc.
- Error messages are also prompted by the system if the user has input wrong value.
- Graphical user interface defects are generally considered as high priority defects as it has direct relationships with usability testing, look and feel of an application.
- It affects the emotion of people and can improve acceptability of the application
- Easy and simple and user friendly interface can be seen below:



User interface of the chatbot when user is asking question about the BSc It coordinator to the system.



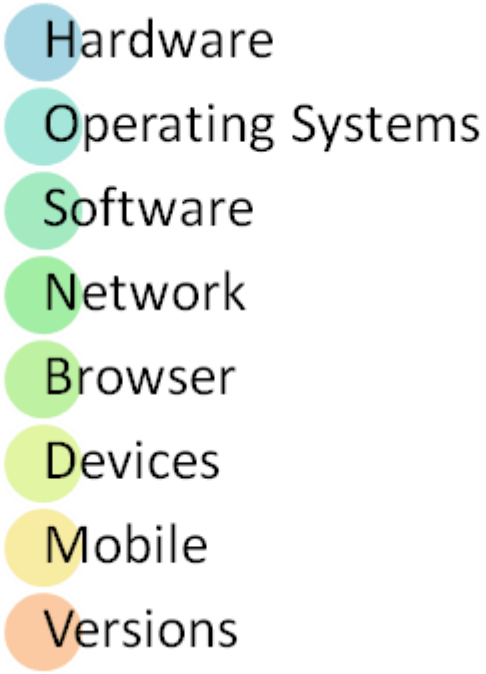
User interface of system's reply to the user for question asked.



Snippet of the simple and easy user interface.

5.3.4 Compatibility testing

- Any system is made up of components.
- In case of software system there may be components such as different types of machines ,printers, hardware, databases, different operating systems and communication systems.
- As world is growing there is possibility of these components being present or absent in the system
- Compatibility Testing is a type of Non-functional testing
- **Types of Compatibility Tests**



- How to do Compatibility Testing

1. The initial phase of compatibility testing is to define the set of environments or platforms the application is expected to work on.
2. The tester should have enough knowledge of the platforms/software/hardware to understand the expected application behavior under different configurations.
3. The environment needs to be set-up for testing with different platforms, devices, networks to check whether your application runs well under different configurations.
4. Report the bugs. Fix the defects. Re-test to confirm Defect fixing.

- In our project we have considered all these aspects mentioned above in the compatibility testing and have designed our chatbot in such a manner that it will be compatible in any environment and one can ensure smooth working of it whatever the environment maybe.

5.3.5 Performance Testing

- Performance Testing is defined as a type of software testing to ensure software applications will perform well under their expected workload.
- Features and Functionality supported by a software system is not the only concern.
- A software application's performance like its response time, reliability, resource usage and scalability do matter.
- The goal of Performance Testing is not to find bugs but to eliminate performance bottlenecks.
- The focus of Performance Testing is checking a software program's

1. **Speed** - Determines whether the application responds quickly
2. **Scalability** - Determines maximum user load the software application can handle.
3. **Stability** - Determines if the application is stable under varying loads

- Performance Testing is popularly called “Perf Testing” and is a subset of performance engineering.
- In the project chatbot machine is a high power system with top quality parts and applications used to develop it.
- Response time of the project plays an important role hence implementation is RSA-NLU has been done for a fast and quick reply.
- Security also plays a major role in this type of testing hence it is implemented with the use of SQL management by providing passwords to confidential information in the databases.
- Back up of the information has already been done in case of any future failure of the system.

5.4 Modifications

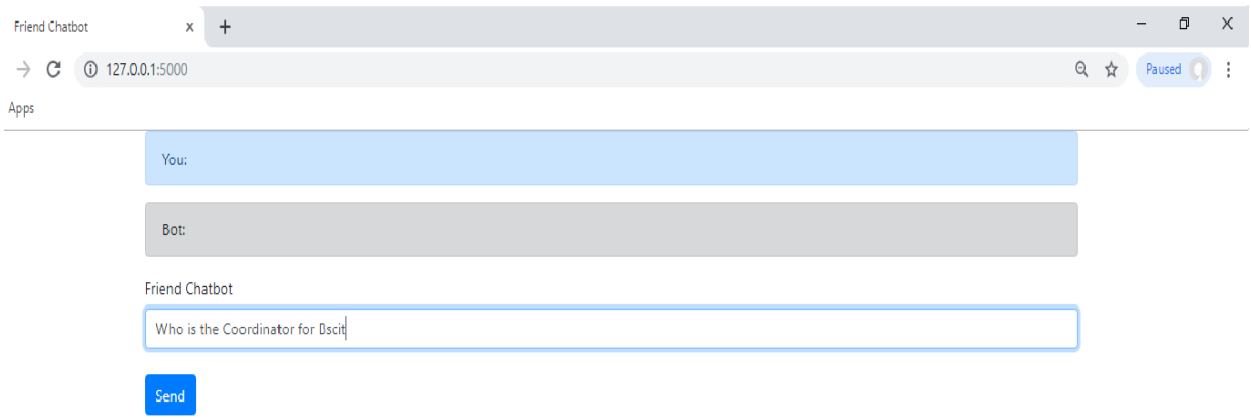
- Earlier the back end of our project included workbench but due to some additional functionalities in SQL management we had to change our backend i.e. database to SQL management server which has all the facilities and features that were required for successful working of the project.
- Initially our database was not properly managed it lacked the table system which is important to differentiate between the values in the table and to carry out the changes i.e. inserting, adding, deleting, modifying values in the database for its sooth working from the developer side.
- Gradually as data was trained it was noticed that some intents were failing to get trained due to the similarity in their basic unique key such intents were completely changed and then were replaced by altogether different unique key for their unique identification.
- We also initiated some changes in the database connection coding part as guided by our project guide resulting in successful connection of the database and smooth working of the ChatBot.
- User Interface which was initially planned lack user-friendliness, hence later on we changed our user interface to ‘Flask UI’ which is indeed user friendly and very easy to use.

5.5 Test Cases

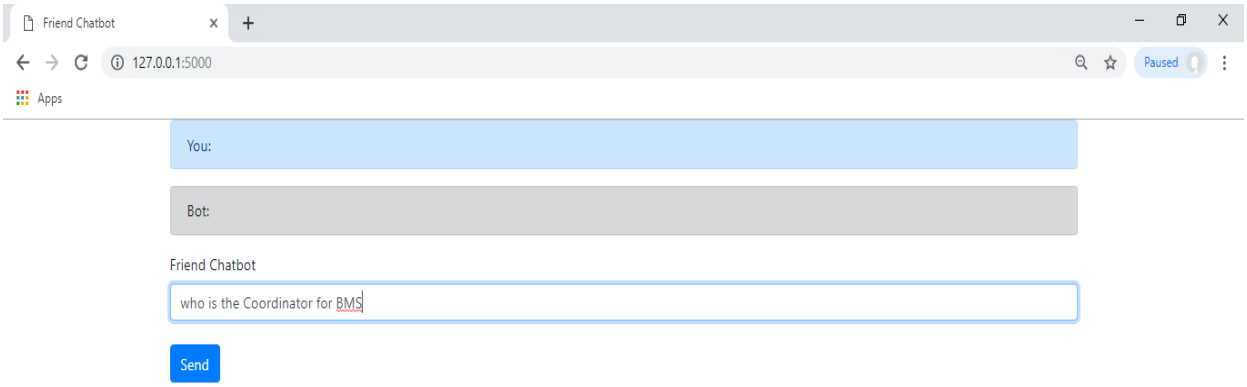
TEST CASE	TITLE	PRIORITY	REQUIREMENTS	PURPOSE	STATUS
TC01	Greet For e.g. (Hi, hello, Good morning, good afternoon etc.)	P1	System should be turned ON	To increase User friendliness of system	Pass
TC02	Greet For e.g. (Hi, hello, Good morning, good afternoon etc.)	P1	System should be turned ON	To increase User friendliness of system	Fail
TC03	Classroom and blockNos enquiry(includes staff office, gymkhana, canteen, library, principle's cabin etc.)	P2	Avoid asking questions in single word.(it might confuse the system and result outcome could be different than expected)	For guiding visitor to the required place without disturbing the staff and other faculty members.	Pass
TC04	Classroom and blockNos enquiry(includes staff office, gymkhana, canteen, library, principle's cabin etc.)	P2	Avoid asking questions in single word.(it might confuse the system and result outcome could be different than expected)	For guiding visitor to the required place without disturbing the staff and other faculty members.	Fail

TC05	Enquiry about the teaching staff in college	P3	Enquiring about the correct subjects which are in the curriculum of the college.	Subject enquiry or any other personal work with staff members.	Pass
TC05	Enquiry about the teaching staff in college	P3	Enquiring about the correct subjects which are in the curriculum of the college.	Subject enquiry or any other personal work with staff members.	Fail

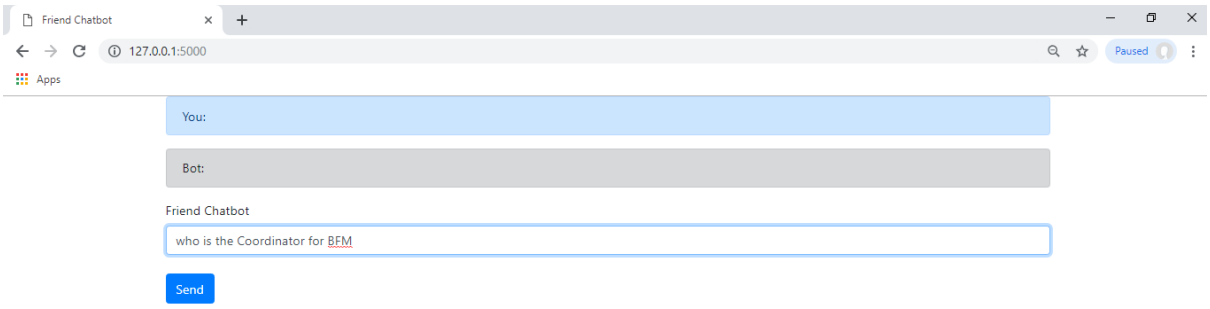
5.6 Screenshots of inputs.



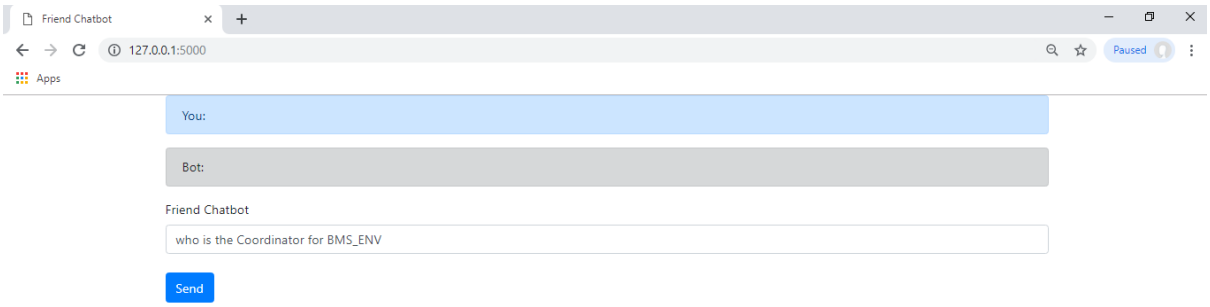
Visitor has a doubt and asks question about “Who is the Coordinator for Bscit?”



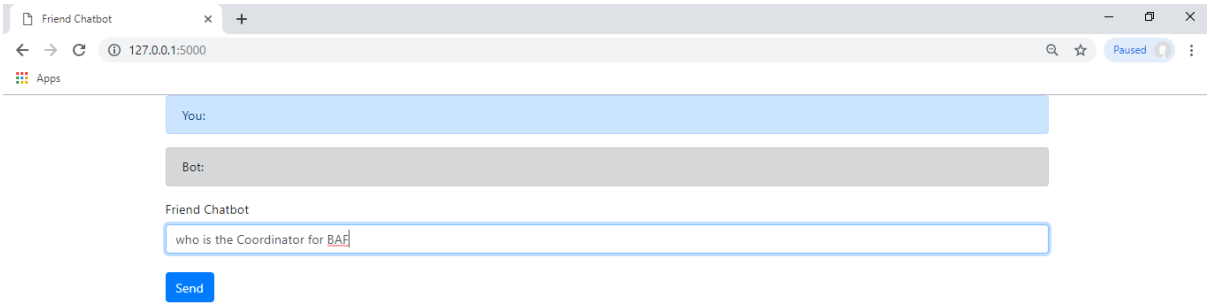
Visitor has a doubt and asks question about BMS coordinator



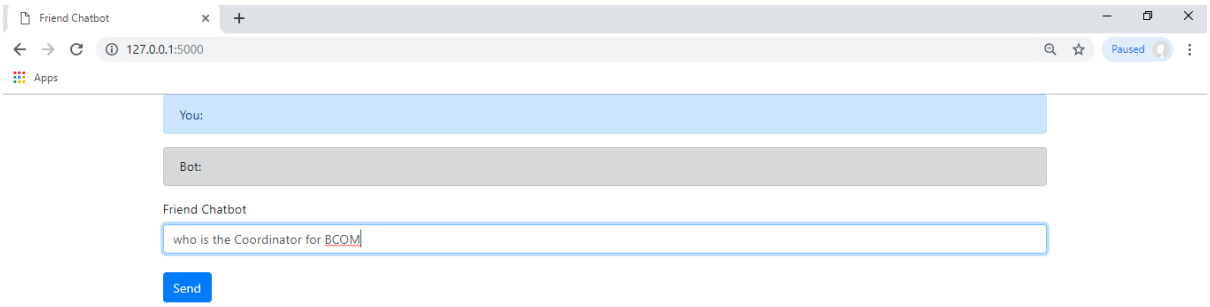
Visitor has a doubt and asks question about BFM coordinator



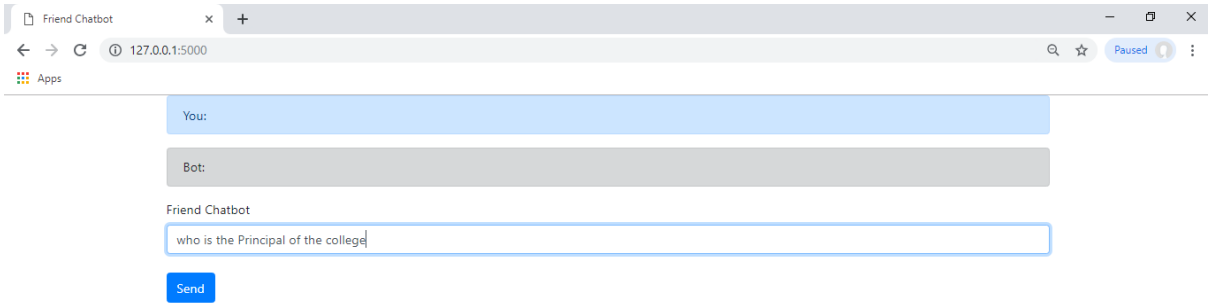
Visitor has a doubt and asks question about BMS_ENV coordinator



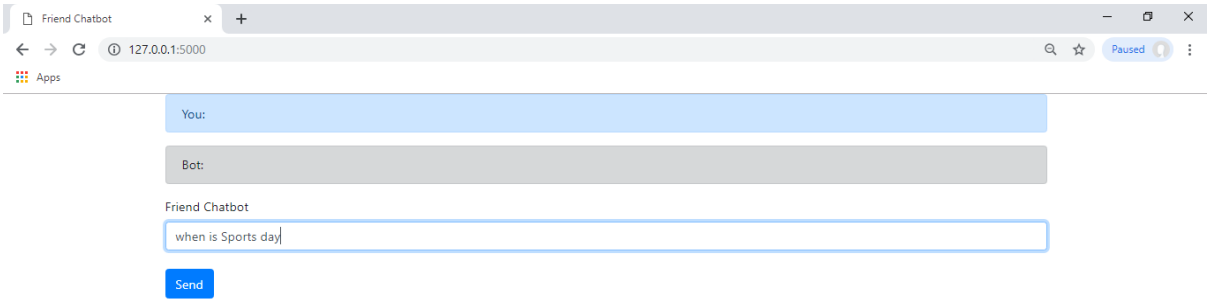
Visitor has a doubt and asks question about BAF coordinator



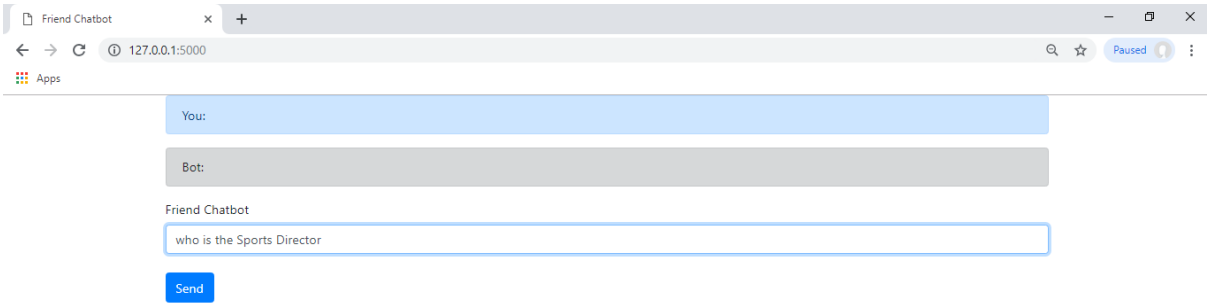
Visitor has a doubt and asks question about coordinator of BCOM



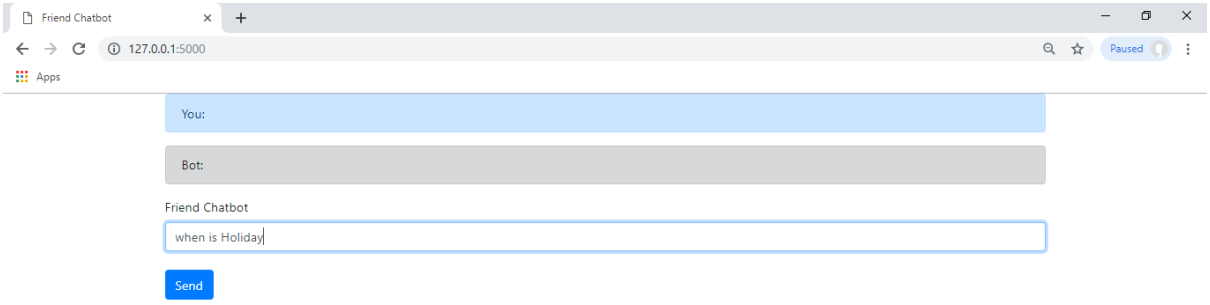
Visitor has a doubt and enquires about principle of college



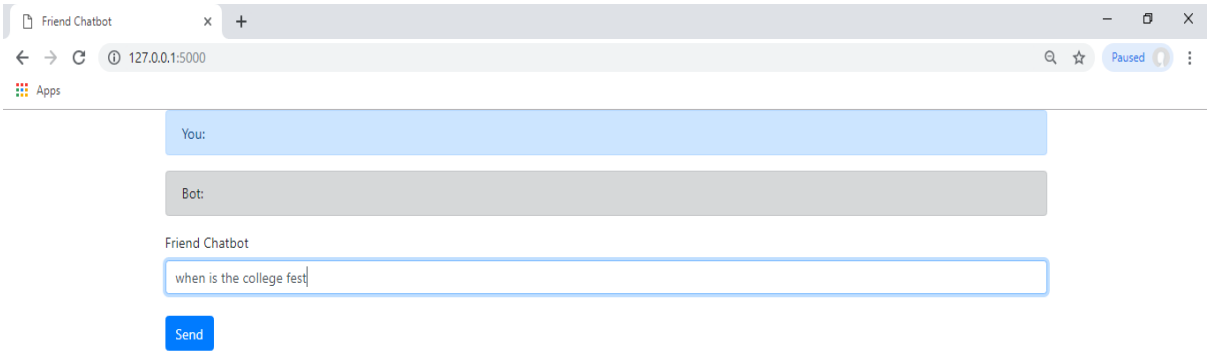
Visitor has a doubt and asks question about dates of sports day



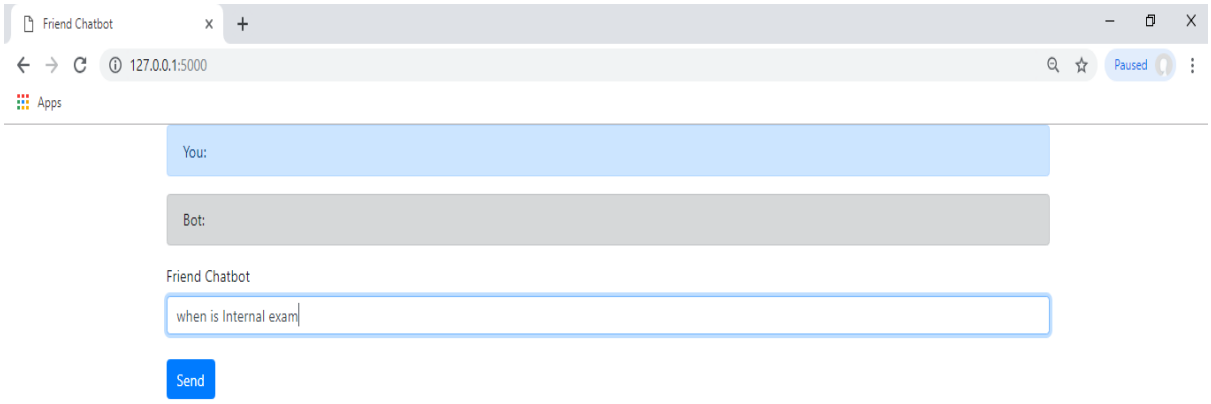
Visitor has a doubt and enquires about the sports director



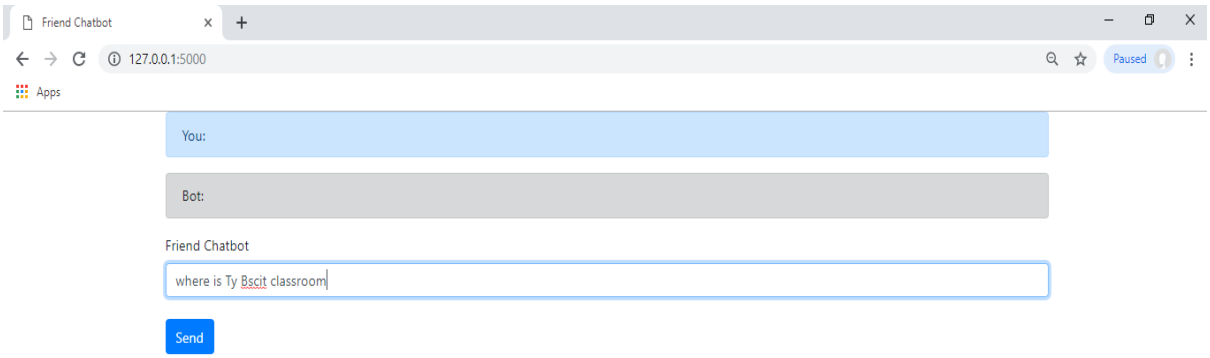
Visitor has a doubt and asks question about Holidays



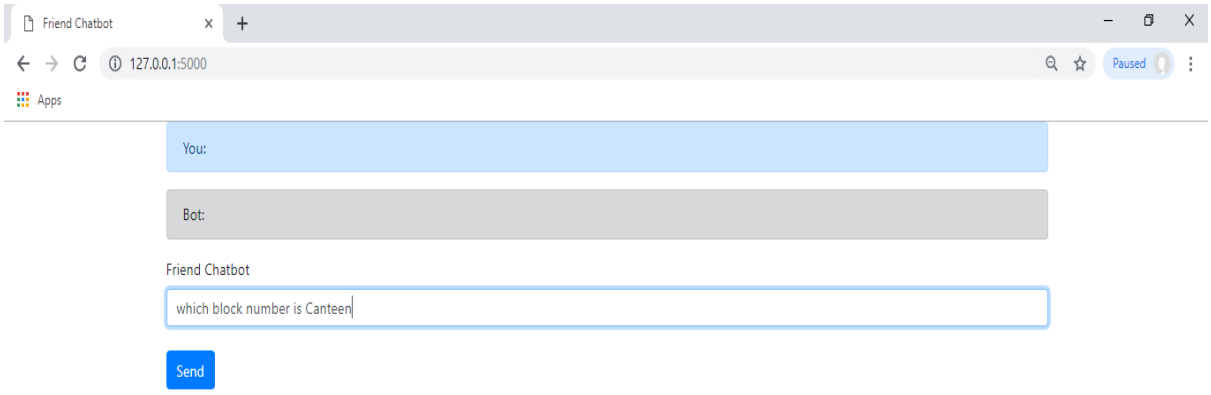
Visitor has a doubt and asks question about college festival



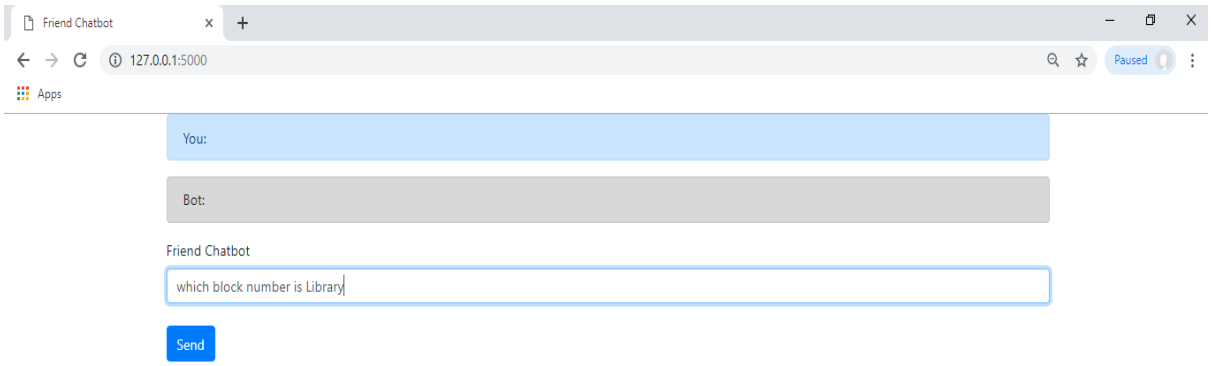
Visitor has a doubt and asks question enquiring about internal examinations



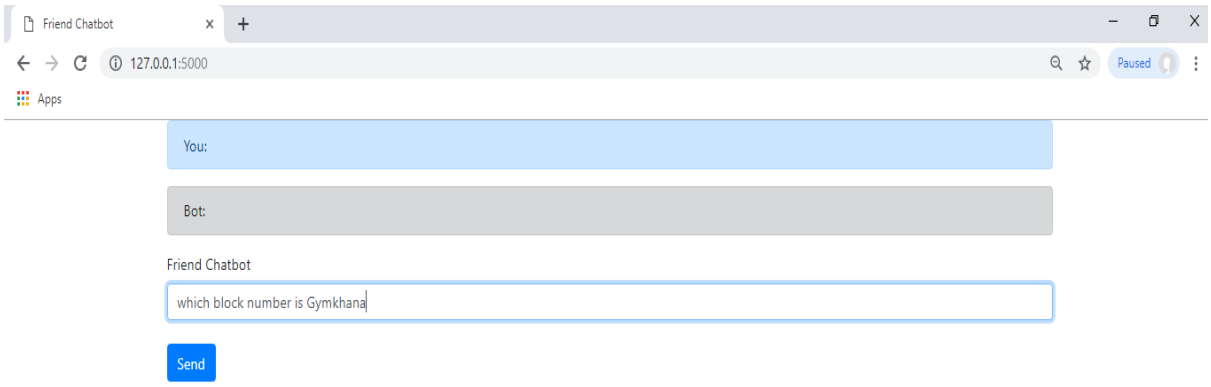
Visitor has a doubt and asks question about location or block number of TYBscIt classroom



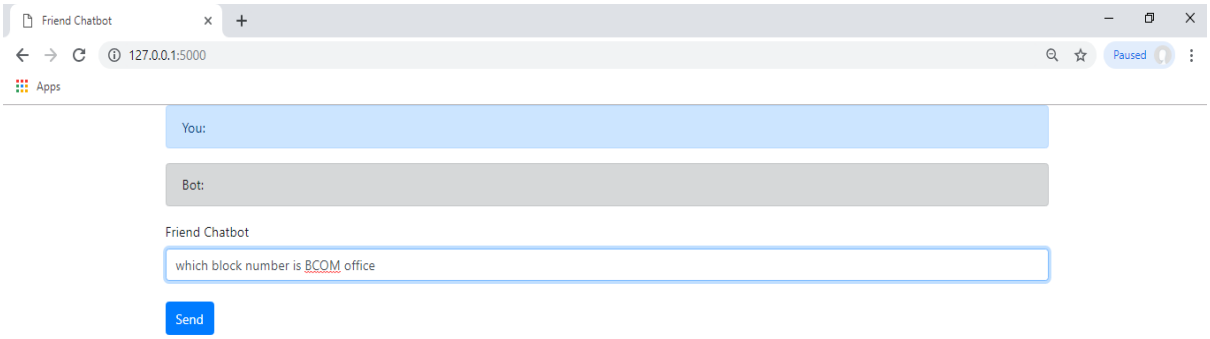
Visitor has a doubt and asks question about location of canteen or a place to eat!



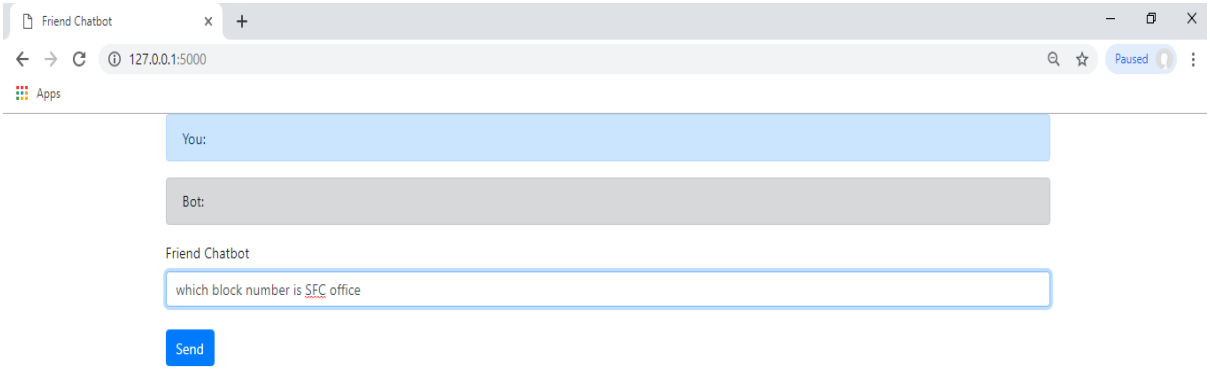
Visitor has a doubt and asks question about location of Library!



Visitor has a doubt and asks question about location of gymkhana!



Visitor has a doubt and enquires about BCOM office



Visitor has a doubt and asks enquires about the SFC office!

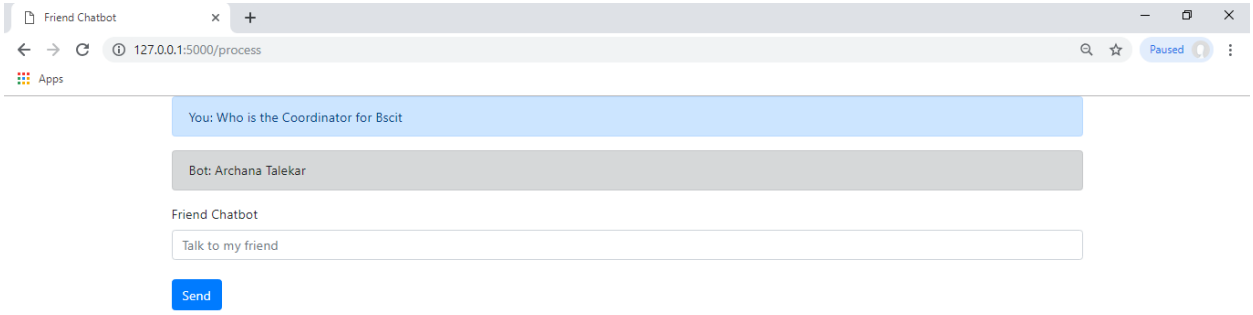
Chapter 6: Results and Discussions.

6.1 Test Cases

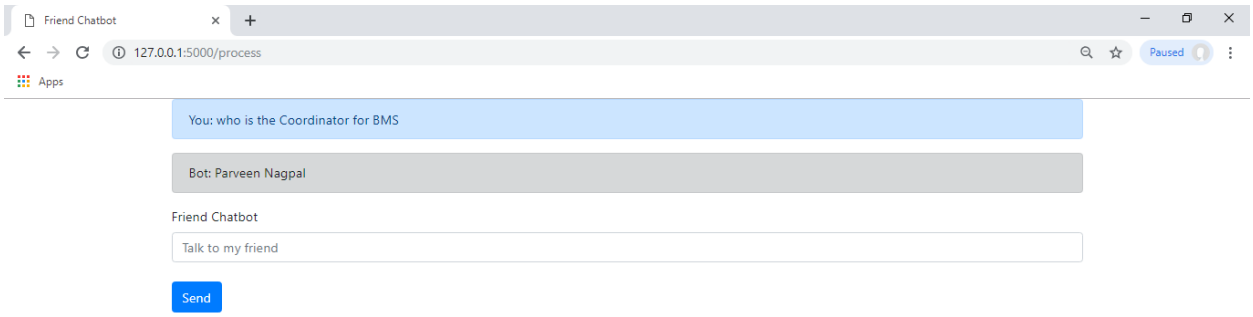
TEST CASE	TITLE	PRIORITY	REQUIREMENTS	PURPOSE	EXPECTED RESULT	ACTUAL RESULT	STATUS
TC 01	Greet For e.g. (Hi, hello, Good morning, good afternoon etc.)	P1	System should be turned ON	To increase User friendliness of system	Greet back	Pass	Pass
TC 02	Greet For e.g. (Hi, hello, Good morning, good afternoon etc.)	P1	System should be turned ON	To increase User friendliness of system	Greet back	System doesn't reply	Fail
TC 03	Classroom and blockNos enquiry(includes staff office, gymkhana, canteen, library, principle's cabin etc.)	P2	Avoid asking questions in single word.(it might confuse the system and result outcome could be different than expected)	For guiding visitor to the required place without disturbing the staff and other faculty members.	Chatbot guides the user properly as per the query asked by the user	Pass	Pass
TC 04	Classroom and blockNos enquiry(includes staff office, gymkhana, canteen, library, principle's cabin etc.)	P2	Avoid asking questions in single word.(it might confuse the system and result outcome could be different than expected)	For guiding visitor to the required place without disturbing the staff and other faculty members.	Chatbot guides the user properly as per the query asked by the user	Chatbot Misguides user due to some unknown error occurred during	Fail

						question phase	
TC 05	Enquiry about the teaching staff in college	P3	Enquiring about the correct subjects which are in the curriculum of the college.	Subject enquiry or any other personal work with staff members.	Guiding correctly with name and block number of their office .	Apt reply from the system.	Pass
TC 05	Enquiry about the teaching staff in college	P3	Enquiring about the correct subjects which are in the curriculum of the college.	Subject enquiry or any other personal work with staff members.	Guiding correctly with name and block number of their office.	Incorrect guidance from the system throwing answer about the college festival dates	Fail

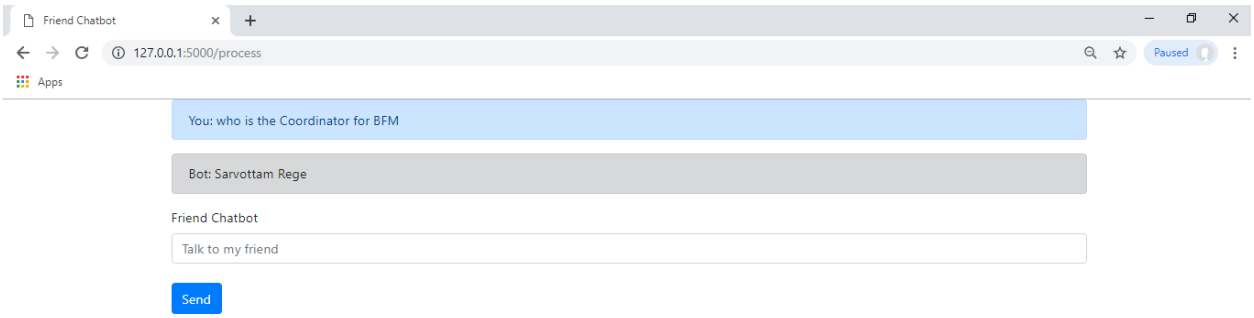
6.2 Screenshots of outputs.



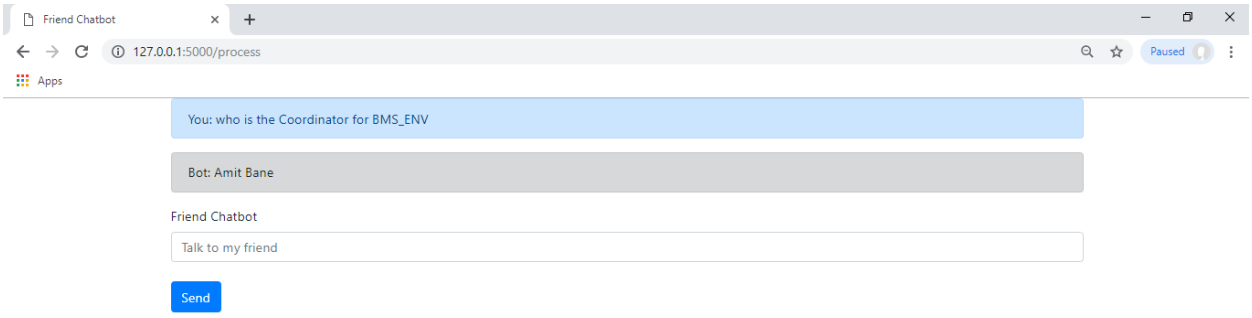
In this Screenshot one can see chatbots reply to the users asked question.



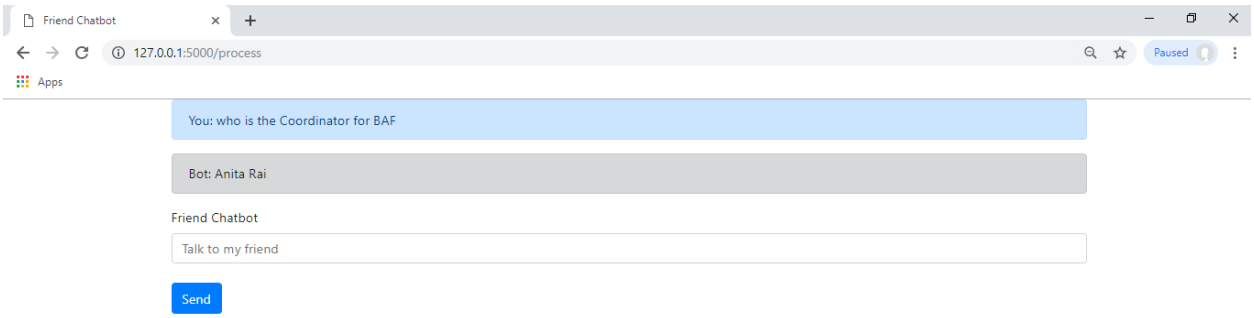
Chatbot replies user with correct answer of coordinator of BMS.



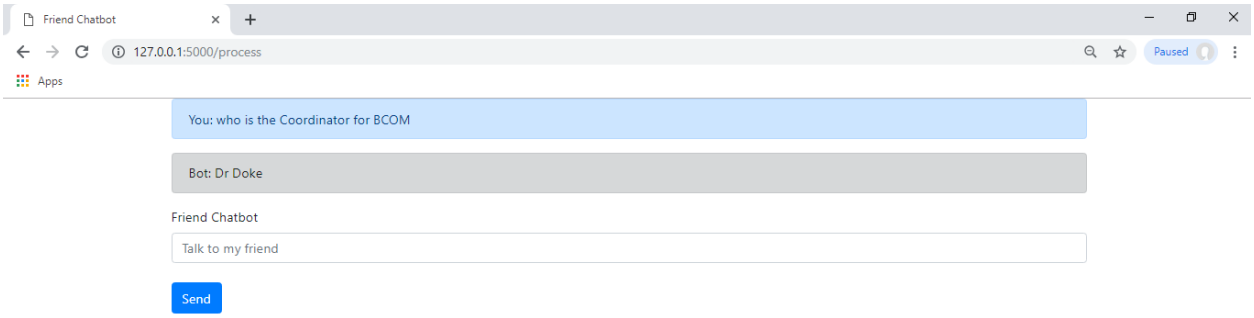
Chatbot replies user with the correct answer of the question asked by the user and throws answer Sarvottam Rege who is the coordinator of BFM.



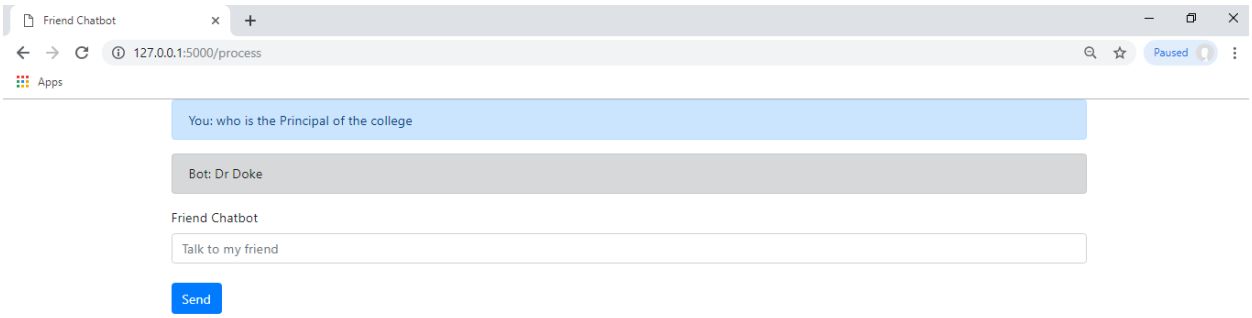
Chatbot replies correctly to the user asked question.



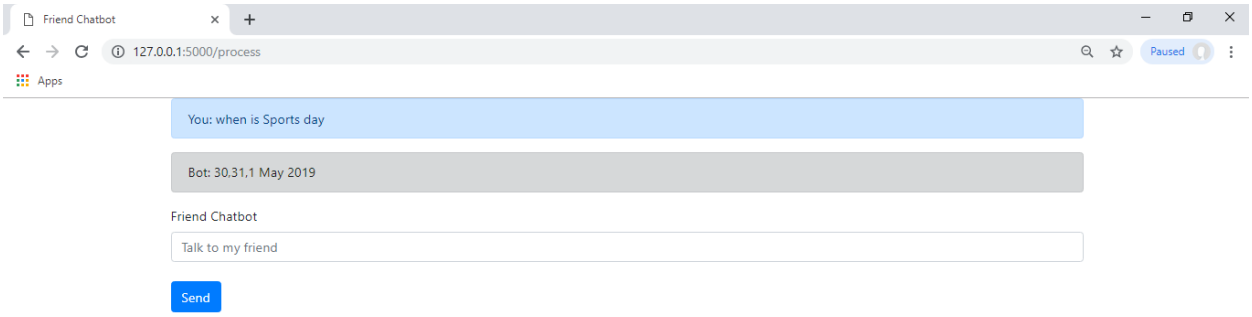
Chatbot replies correctly to the user asked question.



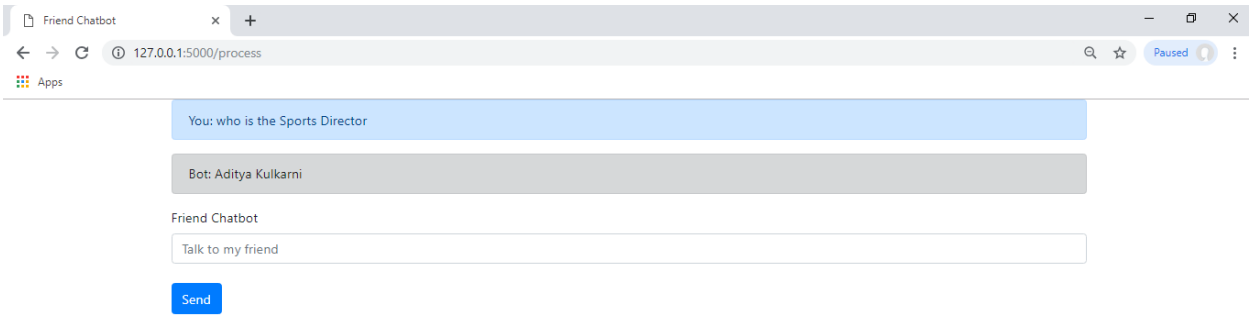
Chatbot replies correctly to the user asked question.



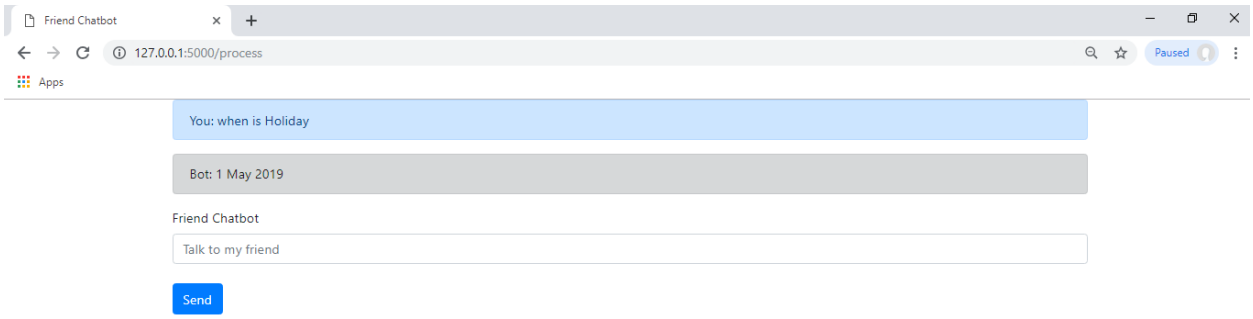
Chatbot replies correctly to the user asked question.



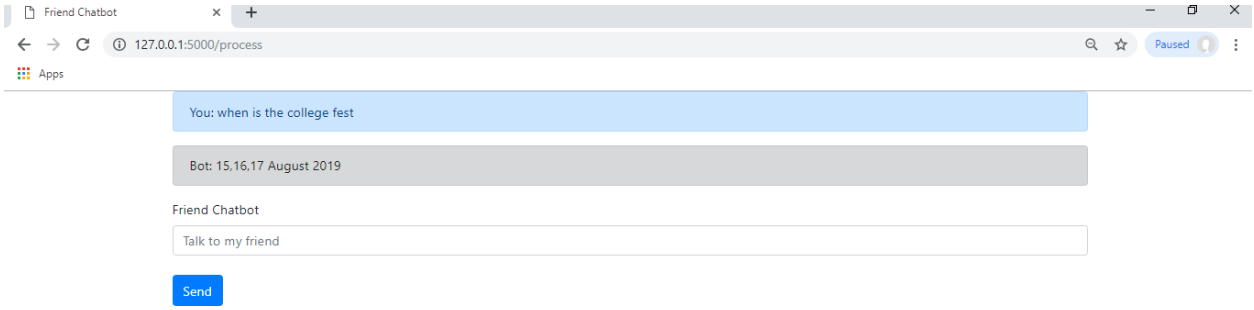
Chatbot replies correctly to the user asked question.



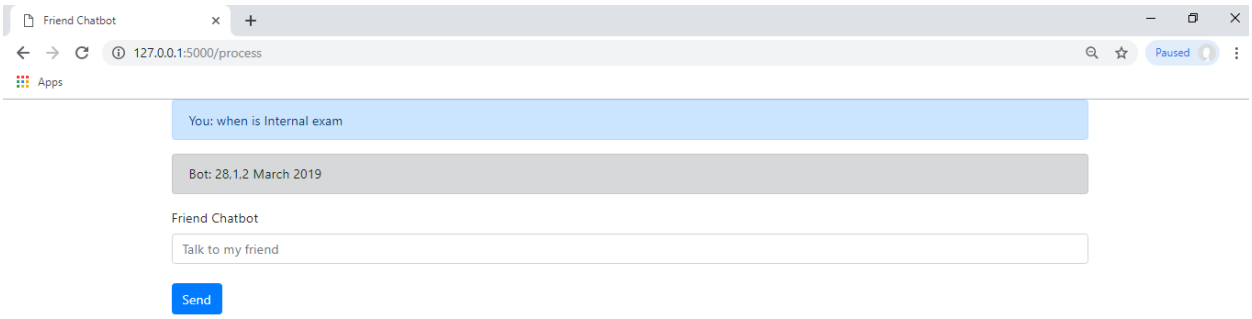
Chatbot replies correctly to the user asked question.



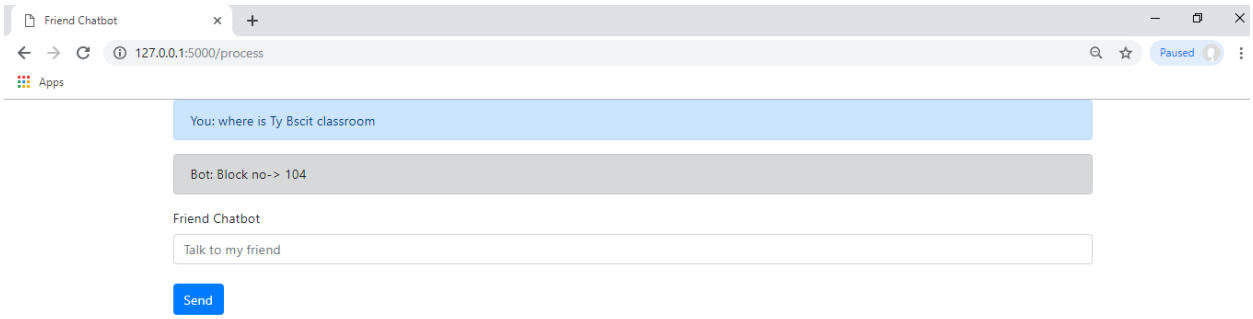
Chatbot replies correctly to the user asked question



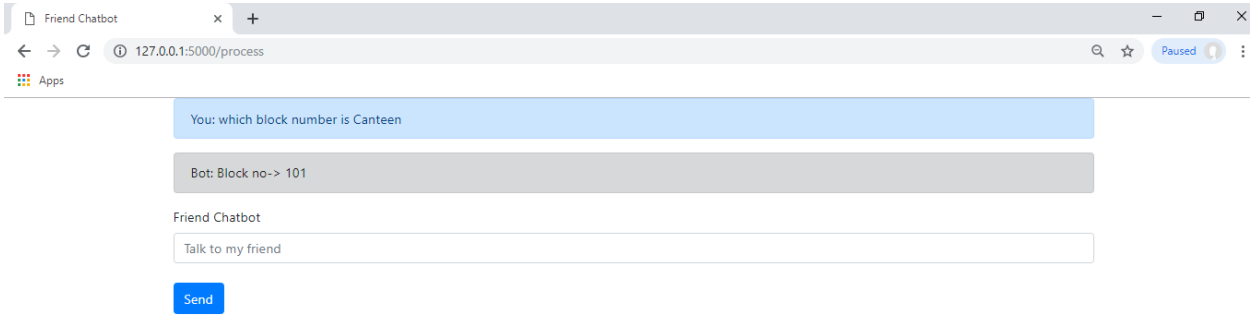
Chatbot replies correctly to the user asked question.



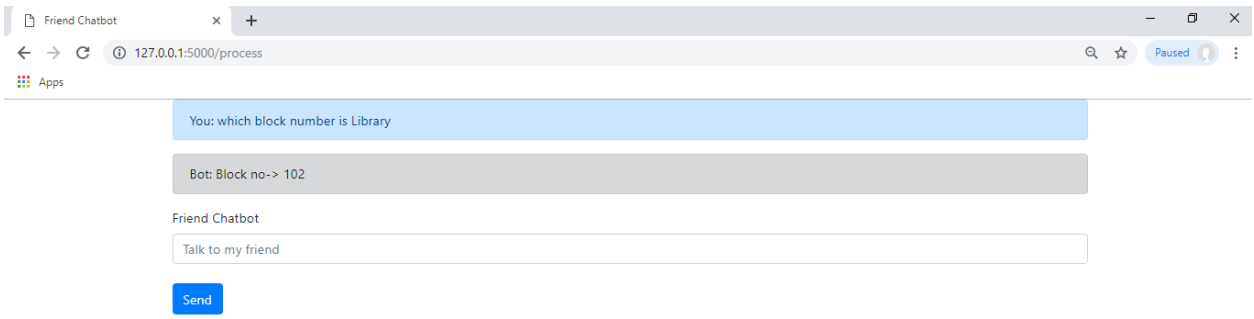
Chatbot replies correctly to the user asked question.



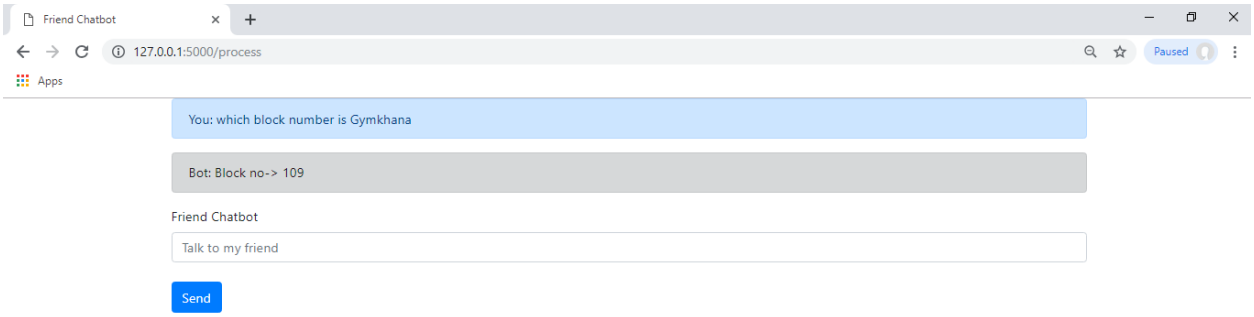
Chatbot replies correctly to the user asked question.



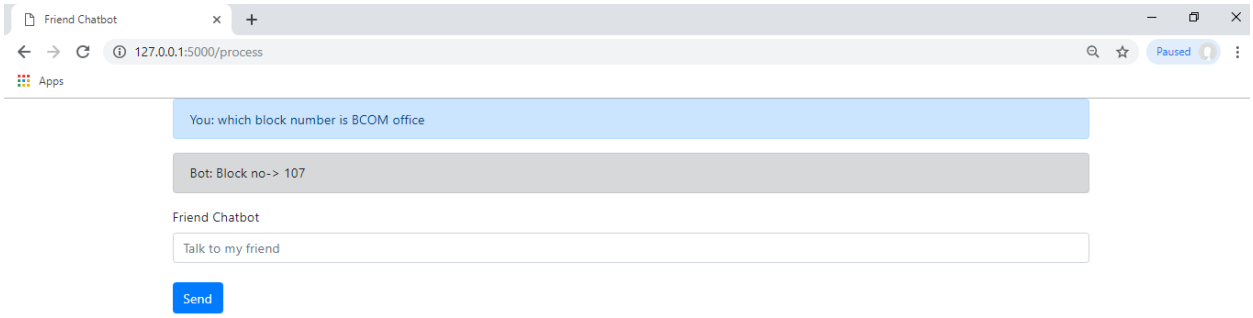
Chatbot replies correctly to the user asked question.



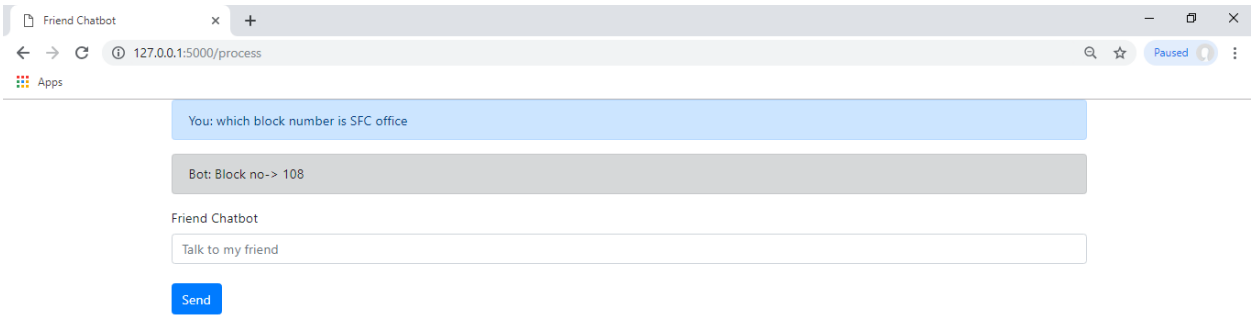
Chatbot replies correctly to the user asked question.



Chatbot replies correctly to the user asked question.



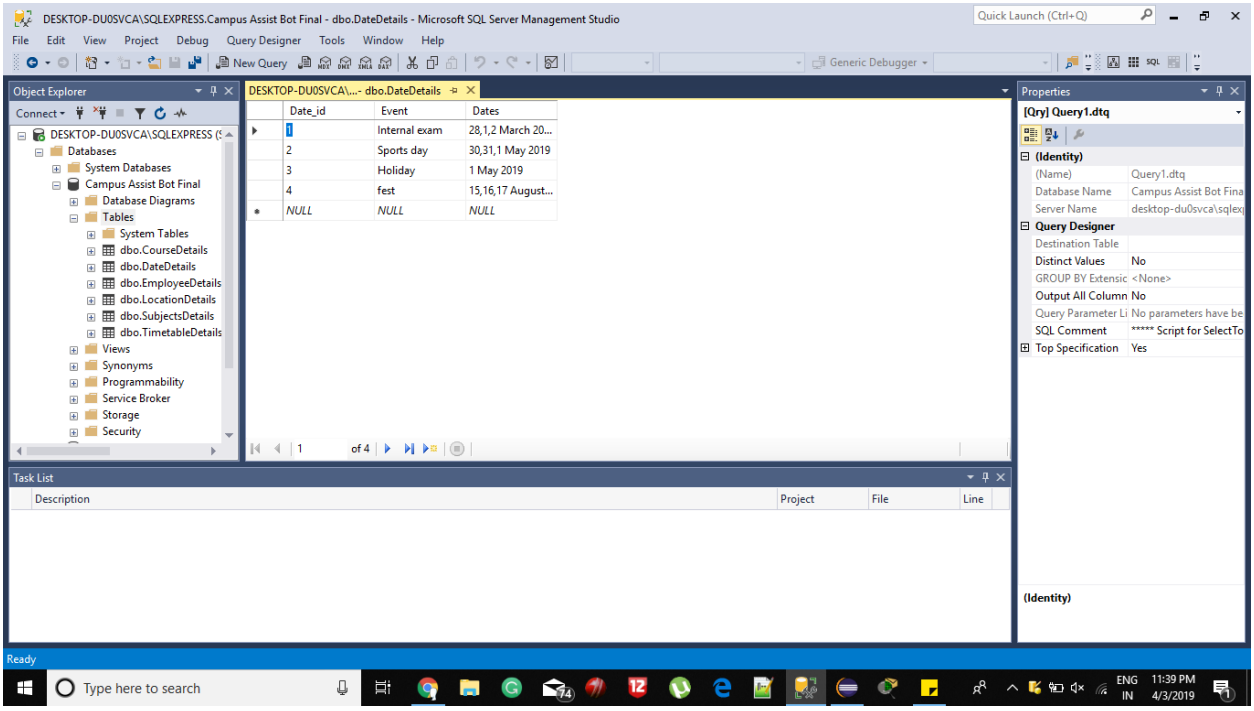
Chatbot replies correctly to the user asked question.



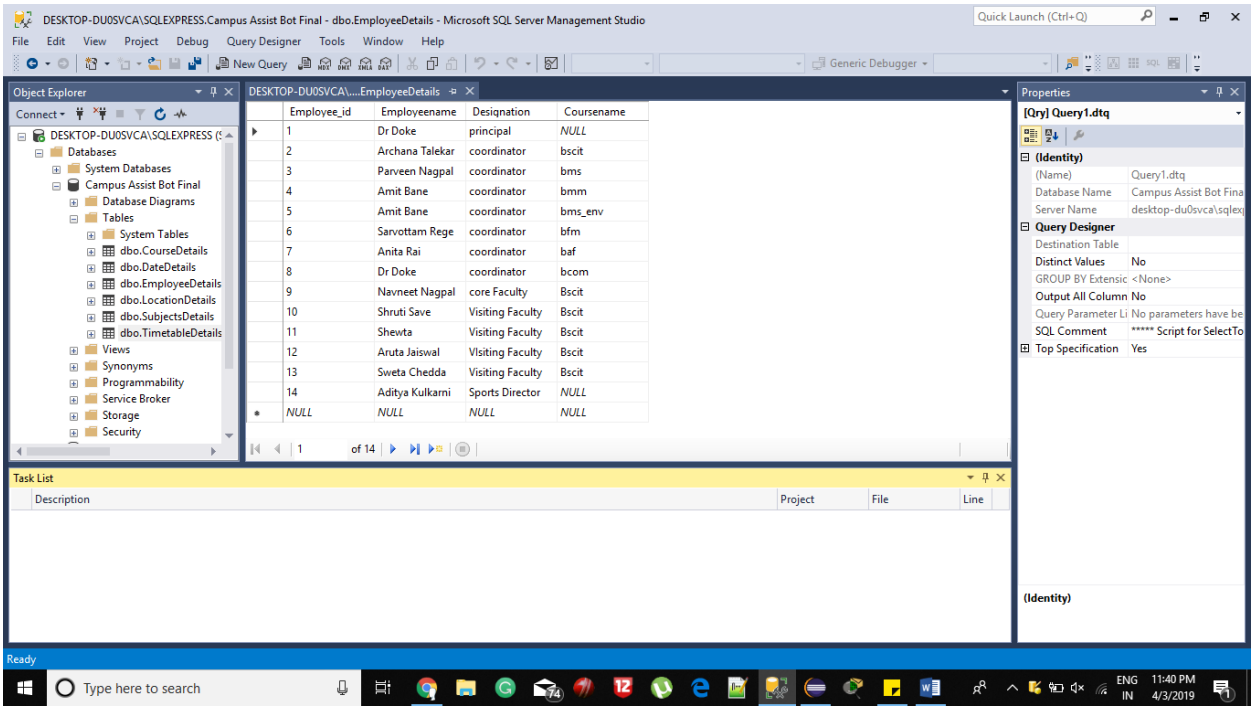
Chatbot replies correctly to the user asked question

6.2.1 Screenshots of database.

6.2.1.1 Table – DateDetails:

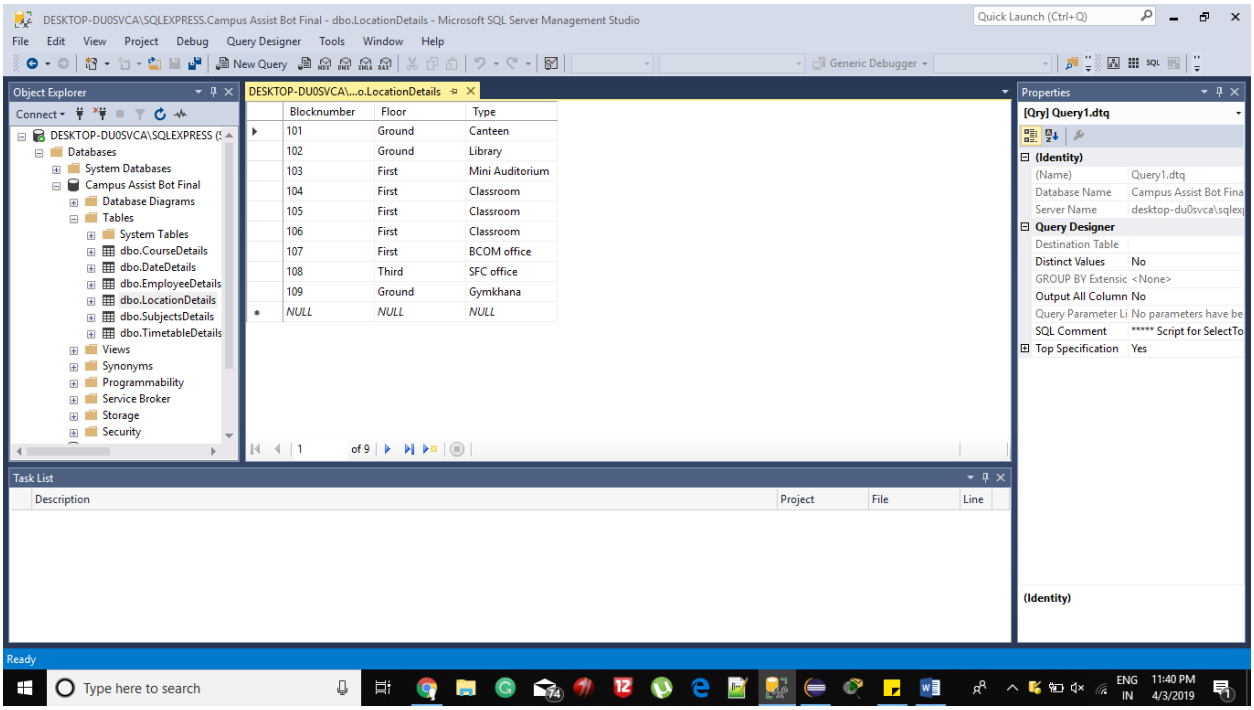


6.2.1.2 Table EmployeeDetails:

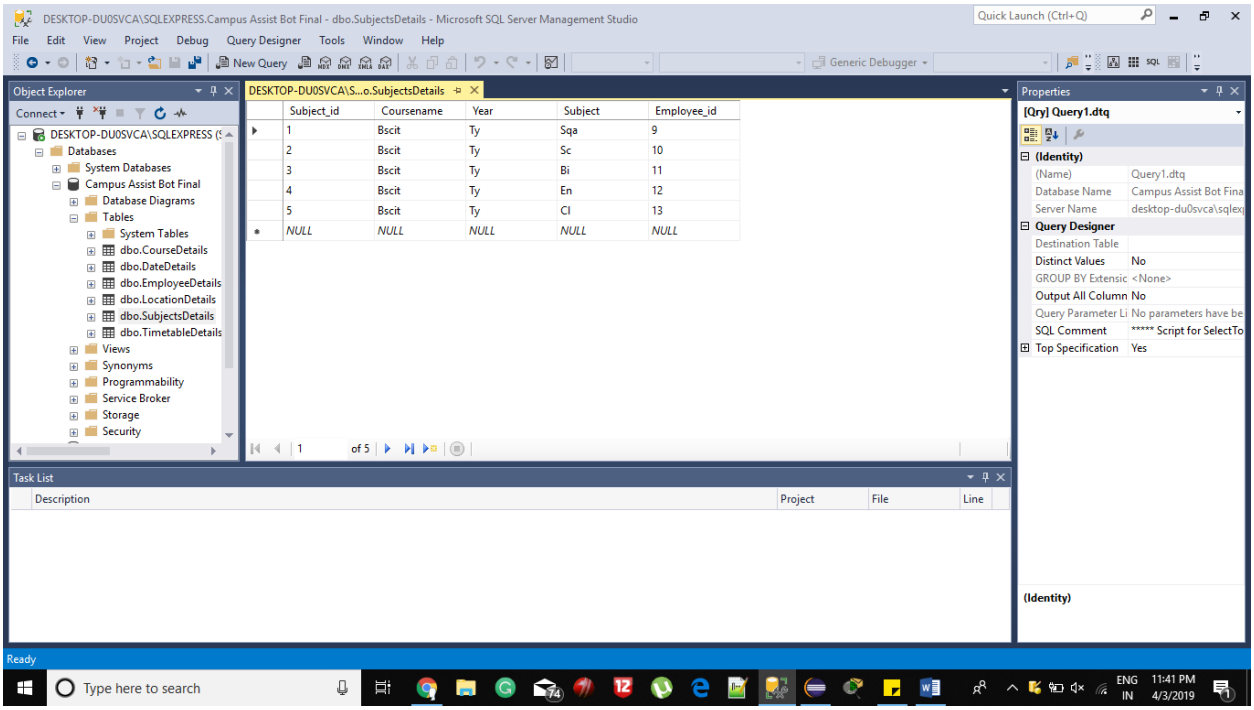


Employee_id	Employeename	Designation	Coursename
1	Dr Doke	principal	NULL
2	Archana Talekar	coordinator	bscit
3	Parveen Nagpal	coordinator	bms
4	Amit Bane	coordinator	bmm
5	Amit Bane	coordinator	bms_env
6	Sarvottam Rege	coordinator	bfm
7	Anita Rai	coordinator	baf
8	Dr Doke	coordinator	bcom
9	Navneet Nagpal	core Faculty	Bscit
10	Shruti Save	Visiting Faculty	Bscit
11	Shewta	Visiting Faculty	Bscit
12	Aruta Jaiswal	VIsiting Faculty	Bscit
13	Sweta Chedda	Visiting Faculty	Bscit
14	Aditya Kulkarni	Sports Director	NULL
NULL	NULL	NULL	NULL

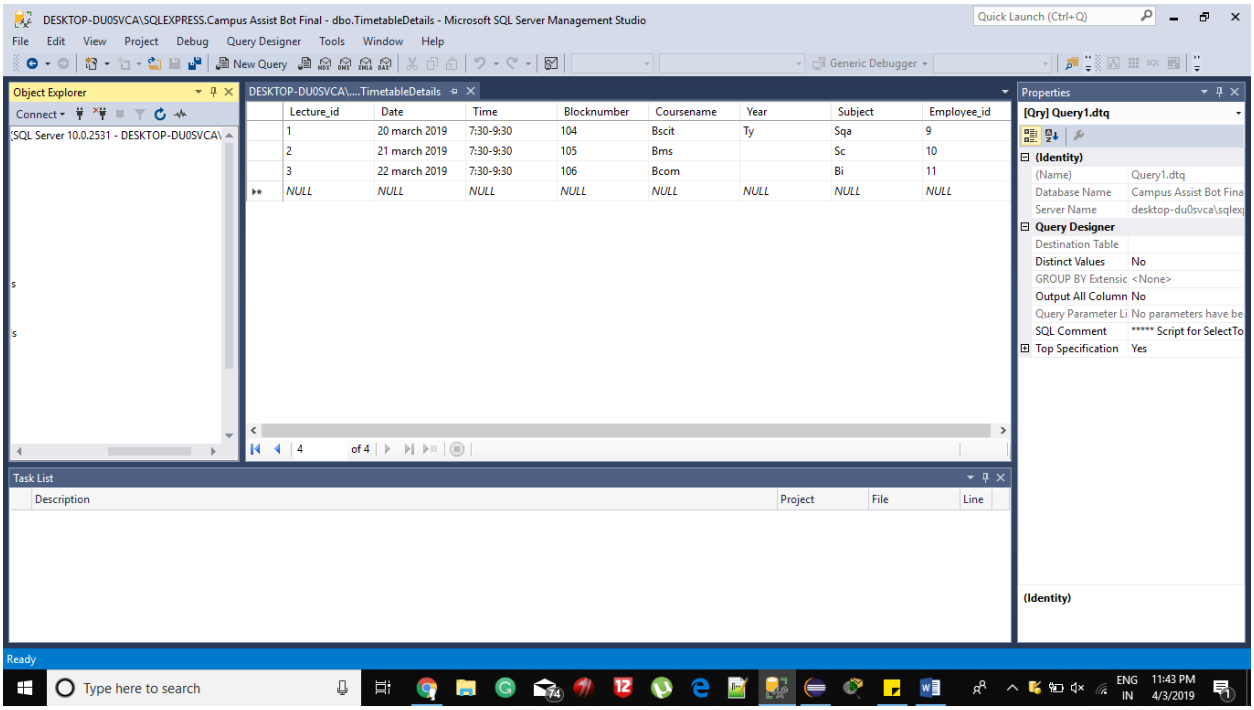
6.2.1.3 Table LocationDetails



6.2.1.4 Table SubjectDetails



6.2.1.5 Table TimeTableDetails



6.3 Test Report

- Initially we planned to make a chat bot system for our college as college didn't had any such system.
- Such system was necessary to reduce burden of the college faculty
- Hence we initiated the working of chatbot by starting with the research and analyzing the difficulties faced by college staff.
- Explaining more about the machine learning and AI.
- Machine learning is an application of Artificial Intelligence that provides system the ability to automatically learn and improve from the experience without being explicitly programmed.
- -Intelligence demonstrated by machines in contrast to natural intelligence displayed by Humans and other animals.
- -Colloquially the term 'AI' is applied when machines mimics "cognitive" functions that humans mind such as learning and problem solving.
- -We are going to develop a Chat bot which will answer the difficulties related to the college
- -Any visitor should have basic idea about the college.
- Whole purpose of building the chatbot was:
- The Chatbot will answer questions such as : "How will an admission process should be carried out ?". It will also answer the general queries of locating classrooms, canteen, college office, SFC office, college Gym etc.
- It also will provide with correct and proper information about the names office and coordinators of each stream.
- It will also provide information about the events, holidays and guest lectures.
- Previously the visitors used to disturb college staff for the queries, so this would be reduced after implementation of Chatbot.
- Our vision behind creating and developing a chatbot was:
- The non-teaching and teaching staff faced a tedious job of answering the queries raised by the visitors.

- Due to which they had to leave their current work midway to attend the visitors and this scenario was at its peak while on going admission procedure.
- This problem and limitation can be solved with the help of this chatbot.
- The main aim of the project is to reduce the burden of staff members and it also can help college in becoming more techno-savvy.
- With the help of our project mentor we successfully developed a chatbot and it also went through various testing methods to measure its capability in the real world.
- Hence we carried out making of our chatbot with the specifications and the objectives defined which were defined in earlier chapters were met successfully.

Chapter 7: Conclusion

7.1 Conclusions:

- In our college i.e. M.L.Dahanukar college no such system existed.
- As a result, many times due to wrong information or unavailability of the college staff at that moment it was difficult for new person in the premises to find the required place in the campus.
- This college/ campus assist Chatbot will work on the principle of Machine Learning.
- Machine learning is a field of computer science that uses statistical techniques to give computers ability to learn Big Data without being explicitly programmed.
- Machine learning explores the study and construction of algorithm that can learn from and make predictions on the data and such algorithms overcome strictly static program instructions.
- So with the help of this Chabot it would be very easy query and doubt solving and correct information would be provided.
- Machine learning is an application of Artificial Intelligence that provides system the ability to automatically learn and improve from the experience without being explicitly programmed.
- Intelligence demonstrated by machines in contrast to natural intelligence displayed by Humans and other animals.
- We are going to develop a Chat bot which will answer the difficulties related to the college.
- Any visitor should have basic idea about the college.
- The Chatbot will answer questions such as : ”Who is the Coordinator of Bscit ?”. It will also answer the general queries of locating classrooms, canteen, college office, SFC office, college Gym etc.
- It will also provide information about the events, holidays and exams.
- Previously the visitors used to disturb college staff for the queries, so this would be reduced after implementation of Chatbot.
- The non-teaching and teaching staff faced a tedious job of answering the queries raised by the visitors.

- Due to which they had to leave their current work midway to attend the visitors and this scenario was at its peak while on going admission procedure.
- This problem and limitation can be solved with the help of this chatbot.
- The main aim of the project is to reduce the burden of staff members and it also can help college in becoming more techno-savvy.
- The basic idea behind creating a chatbot was to help the visitors to get the required information about the college.
- By the implementation of this bot the college staff doesn't have to attend the visitors leaving their work mid-way.
- This bot is pretty much user friendly and can be used my most of the people.
- As the Chat bot is made with machine learning technique to add new data is not that difficult.
- This bot will surely reduce the burden of the college staff.

7.2 Limitation:

There are few limitations to our project such as -

- When the user askes a location based question i.e. where is the canteen the bot is trained only to throw the block number of canteen, but it cannot give directions to the user.
- For the bot to work smoothly data needs to be trained, the data training model is a tedious job.
- The data is needed to be trained a minimum of 10 times.
- This is one of the most important limitation of our project.
- Currently the intent identifier is case sensitive, in future we tend to make it case in-sensitive.
- This is another limitation of the project.
- Currently the model trainer is not exactly identifying multiple intents in the same question regularly, we try to fix this limitation in the near future.
- Like if the user inputs where is Ty Bscit classroom , so the model trainer identifies only the classroom as a location intent but it fails to identify Ty as a Year intent every time .
- Currently the bot can be accessed only through text inputs, this is another limitation we would try to solve by adding text to speech and speech to text functions.
- The current UI is basic in nature , we tend to link the current UI with Facebook Messenge.

7.3 Future Scope :

- Currently the intent identifier is case sensitive, in future it can be made case in-sensitive.
- Currently the module is needed to be trained a minimum of 10 times for the intent to get identified, in future with the evaluation of the training module in such a way that it will take the least amount of training.
- Currently the model trainer is not exactly identifying multiple intents in the same question regularly, with the advancement in the versions this limitation cab be fixed in the near future.
- Like if the user inputs where is Ty Bscit classroom , so the model trainer identifies only the classroom as a location intent but it fails to identify Ty as a Year intent every time . Currently we have used Flask module of python for UI of the project, in future we tend to link our chatbot on Facebook messenger, so any individual just has to call our chatbot from his Facebook messenger to interact with it.
- Currently our project has limited scope we will surely tend to increase our scope and push the limits further.
- Voice feature, like a text to speech and speech to text . Where the visitors can speak with the bot instead of typing cab be added in future.
- Including some IoT functions can help in location search better.
- Some features of Geotag to show the exact location of the searched locations can be implemented successfully.
- In the near future way to reach the desired location can be shown.
- A additional feature can be added wherein users would be allowed to add things which are not built in , like if a certain question is not known to the bot he user can add the question and give the answer as well , the bot will store that in the database and then it can be used by others .

Bibliography

- Data Camp
- Youtube video (Rsa-NLU Chatbot)
  https://www.youtube.com/watch?v=xu6D_vLP5vY
- Wikipedia
- https://apps.worldwritable.com/tutorials/chatbot
- https://pythonprogramming.net/chatbot-deep-learning-python-tensorflow/

Table of Diagrams

Diagram name	Page Number
1.Flowchart	Pg.No- 13
2.Use Case	Pg.No- 15
3.Sequence Diagram	Pg.No- 16
4.Gantt Chart	Pg.No- 17
5.Pert Diagram	Pg.No- 18
6.Spiral Model Diagram	Pg.No- 20
7.Input Screenshots	Pg.No-39-48
8.Output Screenshots	Pg.No- 51- 59
9.Database Screenshots	Pg.No-60- 64