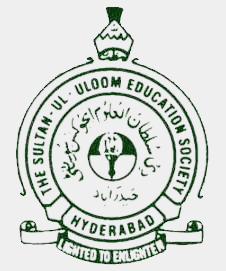
**MUFFAKHAM JAH COLLEGE OF ENGINEERING TECHNOLOGY**

(Affiliated to Osmania University)

Mount Pleasant, 8-2-249, Road No. 3, Banjara Hills, Hyderabad-34.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

***CERTIFICATE***

This is to certify that the Mini Project work titled “**COLLEGE CHATBOT**” is a bonafide work prescribed by the Osmania University for B.E II,III/IV IT I/II Semester during the academic year 2017-2018 carried out by **Ayush Singh (1604-17-737-022), Pranav Hindupur (1604-17-737-004), Samarth Shriram Shetty (1604-17-737-025).**

**Course Coordinator Head-ITD**

**Mr. Gouse Baig Mohd. Dr. Mousmi Ajay Chaurasia**

**A**

**Mini Project Report**

**On**

**COLLEGE CHATBOT**

By

Ayush Singh (1604-17-737-022)

Samarth Shiram Shetty (1604-17-737-022)

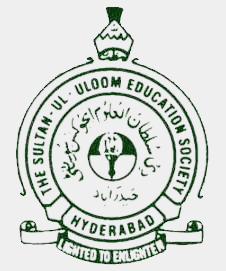
Pranav Hindupur (1604-17-737-022)

**Of**

BE IV Sem (IT-A)

**Under the Guidance of:**

Mr. Gouse Baig



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**MUFFAKHAM JAH COLLEGE OF ENGINEERING AND TECHNOLOGY**

(Affiliated to Osmania University)

Mount Pleasant, 8-2-249, Road No. 3, BanjaraHills, Hyderabad-34.

**ACKNOWLEDGEMENT**

The austerity and satisfaction that one gets on completing a project cannot be fulfilled without mentioning the people who made it possible with gratitude.

We are grateful to the almighty God who helped us all the way throughout the project and also has molded us into what we are today. We express our sincere thanks to our parents who encouraged us always to achieve our goals.

We offer our sincere thanks to Muffakham Jah College of Engineering and Technology for allowing us to do our mini project in their esteemed institution.

We show gratitude to **Dr. K. N. Krishnan sir, M.Tech, PhD** and **Principal** for having provided all the facilities and support. We would like to thank **Dr. Mousmi Ajay Chaurasia,** (**Head of the Department, Information Technology)** for her expert guidance and encouragement at various level of project.

We are Thankful to our guide **Mr. Gouse Baig Sir** and for his sustained inspiring Guidance and cooperation throughout the process of this project report.

We express our deep sense of gratitude and thanks to all the **Teaching** and **Non-Teaching Staff** of our college who stood with us and helped us to make it a successful venture.

**ABSTRACT**

A chatbot is a program that communicates with you. It is a layer on top of, or a gateway to, a service. Sometimes it is powered by machine learning (the chatbot gets smarter the more you interact with it). Or, more commonly, it is driven using intelligent rules (i.e. if the person says this, respond with that).

**PURPOSE OF THE SYSTEM**

The purpose of the chatbot is to drive e-commerce transactions, chat bots have also seen success in delivering daily news digests, weather forecasts and tracking personal habits like meditation, exercise or dietary goals.

**EXISTING SYSTEM**

The chatbots are completely functional. Though the chatbot is providing the service ,it’s being difficult for the service providers to interpret the human responses and inorder to make it reliable everytime feedback is given in order to correct the chatbot response.

**PROPOSED SYSTEM**

To combat this problem, developers are flirting with artificial intelligence (AI) industry and a solution is relatively close at hand. By feeding a machine learning algorithm thousands and thousands of essays, many people believe there’s a good chance of replacing human feedback on essays with AI systems.

**SYSTEM CONFIGURATION:**

**Hardware requirements:**

Processer                     :           Any Update Processer

Ram                             :           Min 1 GB

Hard Disk                   :           Min 100 GB

**Software requirements:**

Operating System       :           Windows family

Technology                 :           Java (1.7/1.8)

Web Technologies      :           Html, Html-5, JavaScript, CSS.

Web Server                 :           Tomcat 7/8

Server side Lang : J2EE

Database                     :           My SQL5.5

UML : Star UML

DFD : DFD Drawer

**CONTENTS**

* **Introduction**
* **Literature survey**
* **Software Requirement Specification**
* **System Design**

**4.1 System Architecture**

* **Implementation**
* **Testing**
* **Screenshots**
* **Conclusion**
* **Future Enhancement (If Any)**
* **Bibliography**

**CHAPTER 1**

**INTRODUCTION**

A Student bot project is built using artificial algorithms that analyzes user’s queries and understand user’s message. This System is a web application which provides answer to the query of the student. Students just have to query through the bot which is used for chatting. Students can chat using any format there is no specific format the user has to follow. The System uses built in artificial intelligence to answer the query. The answers are appropriate what the user queries. If the answer found to invalid, user just need to select the invalid answer button which will notify the admin about the incorrect answer. Admin can view invalid answer through portal via login. System allows admin to delete the invalid answer or to add a specific answer of that equivalent question. The User can query any college related activities through the system. The user does not have to personally go to the college for enquiry. The System analyzes the question and then answers to the user. The system answers to the query as if it is answered by the person. With the help of artificial intelligence, the system answers the query asked by the students. The system replies using an effective Graphical user interface which implies that as if a real person is talking to the user. The user can query about the college related activities through online with the help of this web application. This system helps the student to be updated about the college activities.

**CHAPTER 2**

**LITERATURE SURVEY**

**Evaluating Quality of Chatbots and Intelligent Conversational Agents Nicole Radziwill and Morgan Benton**

Chatbots are one class of intelligent, conversational software agents activated by natural language input (which can be in the form of text, voice, or both). They provide conversational output in response, and if commanded, can sometimes also execute tasks. Although chatbot technologies have existed since the 1960’s and have influenced user interface development in games since the early 1980’s, chatbots are now easier to train and implement. This is due to plentiful open source code, widely available development platforms, and implementation options via Software as a Service (SaaS). In addition to enhancing customer experiences and supporting learning, chatbots can also be used to engineer social harm - that is, to spread rumors and misinformation, or attack people for posting their thoughts and opinions online. This paper presents a literature review of quality issues and attributes as they relate to the contemporary issue of chatbot development and implementation. Finally, quality assessment approaches are reviewed, and a quality assessment method based on these attributes and the Analytic Hierarchy Process (AHP) is proposed and examined.

**Conclusions:** This paper provided a review of the academic literature since 1990, and industry articles since 2015, to 1) gather and articulate quality attributes for chatbots and conversational agents, and 2) discover and synthesize quality assurance approaches to recommend strategies moving forward. There are many ways for practitioners to apply the material in this article:

● The quality attributes (Table 1) can be used as a checklist for a chatbot implementation

team to make sure they have addressed key issues

● Two or more conversational systems can be compared by selecting the most significant

quality attributes, and/or

● Systems can be compared at two points in time to see if quality has improved, which is

particularly useful for adaptive systems that learn as they are exposed to additional

participants and topics The example showed how this goal-oriented approach might be used to evaluate the quality of two different chatbot implementations. Because the method relies on pairwise comparisons, any metric (including those emphasized by the authors in Table 2) can be associated with each quality attribute, and the results will still be valid. Furthermore, this technique can be easily adapted to evaluate different implementations over time, which is essential since most conversational agents learn from experience with users. These factors make the AHP approach particularly robust for assessing the quality of chatbots and conversational agents, resolving the majority of issues identified by previous researchers

**Survey on Chatbot Design Techniques in Speech Conversation Systems**

**Authors:- Sameera A. Abdul-Kader School of Computer Science and Electronic Engineering/University of Essex Colchester/ UK Diyala University/ Diyala/ Iraq Dr. John Woods School of Computer Science and Electronic Engineering/University of Essex Colchester/ UK**

Human-Computer Speech is gaining momentum as a technique of computer interaction. There has been a recent upsurge in speech based search engines and assistants such as Siri, Google Chrome and Cortana. Natural Language Processing (NLP) techniques such as NLTK for Python can be applied to analyze speech, and intelligent responses can be found by designing an engine to provide appropriate human like responses. This type of programme is called a Chatbot, which is the focus of this study. This paper presents a survey on the techniques used to design Chatbots and a comparison is made between different design techniques from nine carefully selected papers according to the main methods adopted. These papers are representative of the significant improvements in Chatbots in the last decade. The paper discusses the similarities and differences in the techniques and examines in particular the Loebner prizewinning Chatbots.

**CONCLUSIONS:** In this paper, the literature review has covered a number of selected papers that have focused specifically on Chatbot design techniques in the last decade. A survey of nine selected studies that affect Chatbot design has been presented, and the contribution of each study has been identified. In addition, a comparison has been made between Chatbot design techniques in the selected studies and then with the Loebner Prize winning Chatbot techniques. From the survey above, it can be said that the development and improvement of Chatbot design is not grow at a predictable rate due to the variety of methods and approaches used to design a Chatbot. The techniques of Chatbot design are still a matter for debate and no common approach has yet been identified. Researchers have so far worked in isolated environments with reluctance to divulge any improved techniques they have found, consequently, slowing down the improvements to Chatbots. Moreover, the Chatbots designed for dialogue systems in the selected studies are, in general, limited to particular applications. General-purpose Chatbots need improvements by designing more comprehensive knowledge bases. Although some commercial products have emerged recently in the market (e.g. Microsoft Cortana) as dialogue Chatbots, improvements need continuous research and lack a common solution. Each researcher needs to robustly document any successful improvements to allow the human computer speech interaction to agree a common approach. This will always be at odds with commercial considerations.

**CHAPTER 3**

**SYSTEM ANALYSIS**

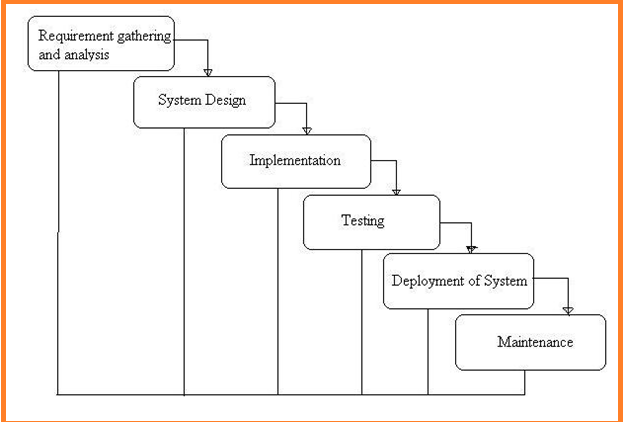


Fig:-1 Project SDLC

• Project Requisites Accumulating and Analysis

• Application System Design

• Practical Implementation

• Manual Testing of My Application

• Application Deployment of System

• Maintenance of the Project

**Requisites Accumulating and Analysis**

It’s the first and foremost stage of the any project as our is a an academic leave for requisites amassing we followed of IEEE Journals and Amassed so many IEEE Relegated papers and final culled a Paper designated “Individual web revisitation by setting and substance importance input and for analysis stage we took referees from the paper and did literature survey of some papers and amassed all the Requisites of the project in this stage

**System Design**

In System Design has divided into three types like GUI Designing, UML Designing with avails in development of project in facile way with different actor and its utilizer case by utilizer case diagram, flow of the project utilizing sequence, Class diagram gives information about different class in the project with methods that have to be utilized in the project if comes to our project our UML Will utilizable in this way The third and post import for the project in system design is Data base design where we endeavor to design data base predicated on the number of modules in our project

**Implementation**

The Implementation is Phase where we endeavor to give the practical output of the work done in designing stage and most of Coding in Business logic lay coms into action in this stage its main and crucial part of the project

**Testing**

**Unit Testing**

It is done by the developer itself in every stage of the project and fine-tuning the bug and module predicated additionally done by the developer only here we are going to solve all the runtime errors

**Manual Testing**

As our Project is academic Leave we can do any automatic testing so we follow manual testing by endeavor and error methods

**Deployment of System**

Once the project is total yare we will come to deployment of client system in genuinely world as its academic leave we did deployment i our college lab only with all need Software’s with having Windows OS

**Maintenance**

The Maintenance of our Project is one time process only

**Functional Requirements**

* **User Sign up :-** all the application users have to give all the mandatory Fields and get an Account in our application to access our application
* **User Login:-** To access the application we are verifying the users login user name and Password
* **Admin Login:-** he is super user of the application where he can login into the application with his/her user name and password
* **Questions:-** Admin will add all the question in to application
* **Answers:-** all answers for the Question add by admin
* **Suggestions:-** will be adding Suggestions to user
* **Chabot :-** Where we can chat & get Results

**Application needs Non-Functional Requisites**

**Expanded System admin security:** overseer to eschew the abuse of the application by PC ought to be exceptionally secured and available.

**Compactness:** The Presentation of this application is facile to utilize so it is looks simple for the using client to comprehend and react to identically tantamount.

**Unwavering quality:** and the functionalities accessible in the application this substructure has high probability to convey us the required inquiries.

**Time take for Reaction:** The time taken by the application to culminate an undertaking given by the client is very fast.

**Multifariousness:** Our application can be stretched out to incorporate the vicissitudes done by applications present now to enhance the performance of the item. This is implicatively insinuated for the future works that will be done on the application.

**Vigor:** The project is blame tolerant concerning illicit client/beneficiary sources of info. Blunder checking has been worked in the platforms to avert platforms disappointmen

**About Project Software’s**

JAVA, Apache Server, MSQL, EDIT ++

In our web Application Development we are using one tier architecture as total applicant will be developed in single system with all the three layers of application development like presentation layer where we use our web technologies to make of GUI of the application like HTML, HTML-5, CSS, JS Etc. and in second layer we have to make our business logic or called as implementation of application where we are using java, J2EE and also we use JDBC to connect from our Business layer to data base layer and final our data base layer where we develop the Data structure of the application

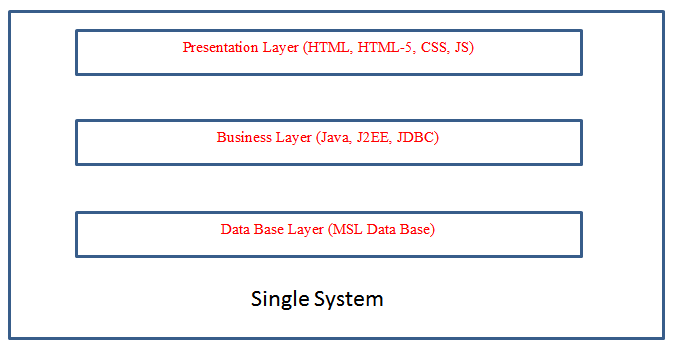


Fig:-3 Single tire Architecture Project Development

**How we used java in our Project Development**

**Installation and Setup in our system**

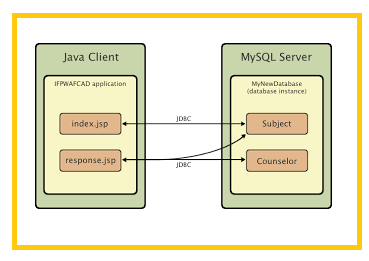
The Software we download form the oracle website as it’s an open source as per the software we have installed it in our system and for we have set the system path of java in our OS location We have used the main logic of our algorithm by core java concepts only for web application we have used all JSP concepts and to connect data base we have used JDBC with all this concept we have done the application in Single tire Architecture Project

**Data Storage in MYSQL**

We have taken open source software MYSQL from the provide website and run in our system we used for creating our project data base related tables as per project requirement’s even for user friendly access of my sql we used Software called SQL Yog where we can do all the operation of mysql by click & use

**About the role of apace tomcat webserver**

As our project is a web applicant we need webserver so for that we used again open sour software where our total project source code will be in webapps of the server form that location the application run into web browser where users can see the implementation of the total project



Application Development Structure

**CHAPTER 4**

**SOFTWARE REQUIREMENT SPECIFICATIONS**

**Hardware requirements:**

Processer                     :           Any Update Processer

Ram                             :           Min 1 GB

Hard Disk                   :           Min 100 GB

**Software requirements:**

Operating System       :           Windows family

Technology                 :           Java (1.7/1.8)

Web Technologies      :           Html, Html-5, JavaScript, CSS.

Web Server                 :           Tomcat 7/8

Server side Lang : J2EE

**CHAPTER 5**

**SYSTEM DESIGN**

**Usecase:**

****

**Class Diagram:**

****

**Sequence and Collaboration Diagram;**

****

****

****

****

**State Chart:**

****

**Activity Diagram:**

****

**Component Diagram:**

****

**Deployment Diagram:**

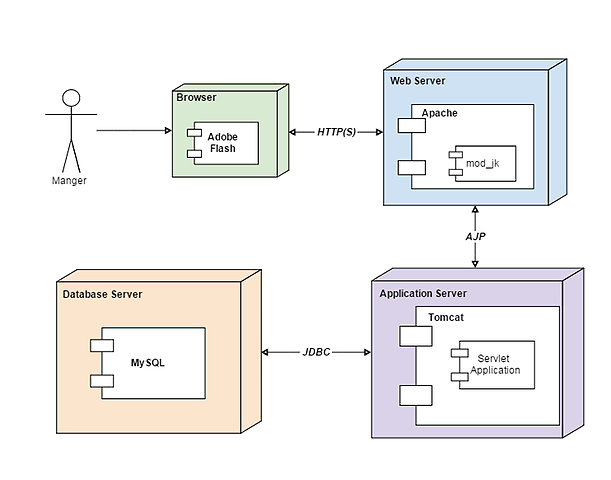
****

Fig:-Java Application Deployment Diagram

**CHAPTER 6**

**IMPLEMENTATION**

**Databasecon.java**

package databaseconnection;

import java.sql.\*;

public class databasecon

{

static Connection co;

public static Connection getconnection()

{

try

{

Class.forName("com.mysql.jdbc.Driver");

co = DriverManager.getConnection("jdbc:mysql://localhost:3306/chatbox","root","root");

}

catch(Exception e)

{

System.out.println("Database Error"+e);

}

return co;

}

}

**CountWords.java**

package CT;

import org.apache.commons.lang.StringUtils;

public class CountWords

{

public static int main(String input, String compare)

{

input=input.toLowerCase();

compare=compare.toLowerCase();

int number = StringUtils.countMatches(input, compare);

return number;

}

}

**GetData.java**

package CT;

import databaseconnection.\*;

import java.sql.\*;

public class GetData

{

static Connection con1=null;

static Statement st1=null;

public static String main(String id) {

String res="";

try{

con1 = databasecon.getconnection();

st1 = con1.createStatement();

String sql=null;;

sql="select \* from questions where sno='"+id+"'";

ResultSet rs=null;

rs=st1.executeQuery(sql);

if(rs.next())

{

res=rs.getString("qns").trim();

}

}

catch(Exception e){

System.out.println(e);

}

finally{

try{

con1.close();

st1.close();

// rs.close();

}

catch(Exception e){

System.out.println(e);

}

}

return res;

}

public static void main(String[] args)

{

System.out.println(main("1"));

}

}

**4. Data Base Tables**

create database if not exists `chatbox`;

USE `chatbox`;

DROP TABLE IF EXISTS `ans`;

CREATE TABLE `ans` (

`sno` varchar(20) DEFAULT NULL,

`ans` varchar(1000) DEFAULT NULL,

`type` varchar(100) DEFAULT 'text'

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `msgs`;

CREATE TABLE `msgs` (

`sno` int(11) NOT NULL AUTO\_INCREMENT,

`msg` longblob,

`user\_` varchar(100) DEFAULT NULL,

`time\_` varchar(100) DEFAULT NULL,

`type\_` varchar(100) DEFAULT NULL,

PRIMARY KEY (`sno`)

) ENGINE=InnoDB AUTO\_INCREMENT=58 DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `profilepic`;

CREATE TABLE `profilepic` (

`email` varchar(200) NOT NULL,

`pic` longblob,

PRIMARY KEY (`email`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `questions`;

CREATE TABLE `questions` (

`Sno` int(11) NOT NULL AUTO\_INCREMENT,

`qns` varchar(500) NOT NULL,

PRIMARY KEY (`Sno`,`qns`)

) ENGINE=InnoDB AUTO\_INCREMENT=8 DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `suggetions`;

CREATE TABLE `suggetions` (

`sno` int(11) NOT NULL AUTO\_INCREMENT,

`msg` varchar(1000) DEFAULT NULL,

PRIMARY KEY (`sno`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `users`;

CREATE TABLE `users` (

`uid` varchar(100) DEFAULT NULL,

`name` varchar(200) DEFAULT NULL,

`pwd` varchar(200) DEFAULT NULL,

`email` varchar(100) DEFAULT NULL,

`ph` varchar(100) DEFAULT NULL,

`addr` varchar(100) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

****

**CHAPTER 7**

**TESTING**

**Software testing**

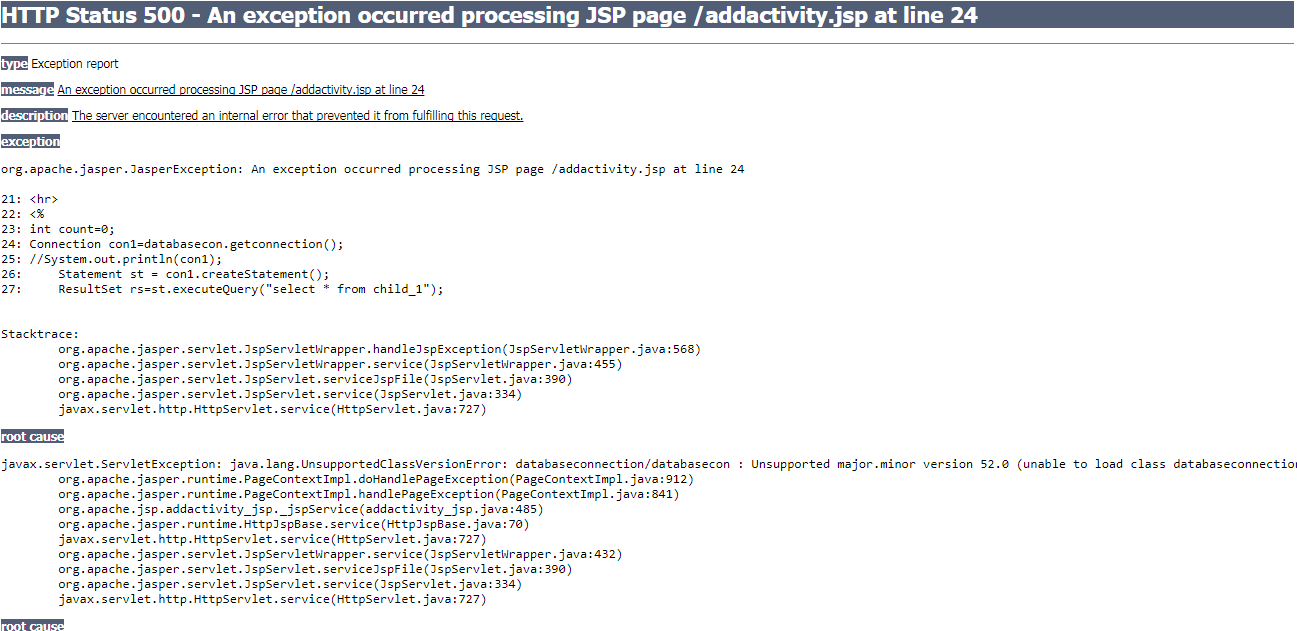
Software testing is one of the main stages of project development life cycle to provide our cessation utilizer with information about the quality of the application and ours, in our Project we have under gone some stages of testing like unit testing where it’s done in development stage of the project when we are in implementation of the application after the Project is yare we have done manual testing with different Case of all the different modules in the application we have even done browser compatibility testing in different web browsers in market, even we have done Client side validation testing on our application.

**Unit testing**

The unit testing is done in the stage of implementation of the project only the error are solved in development stage some of the error we come across in development are given below

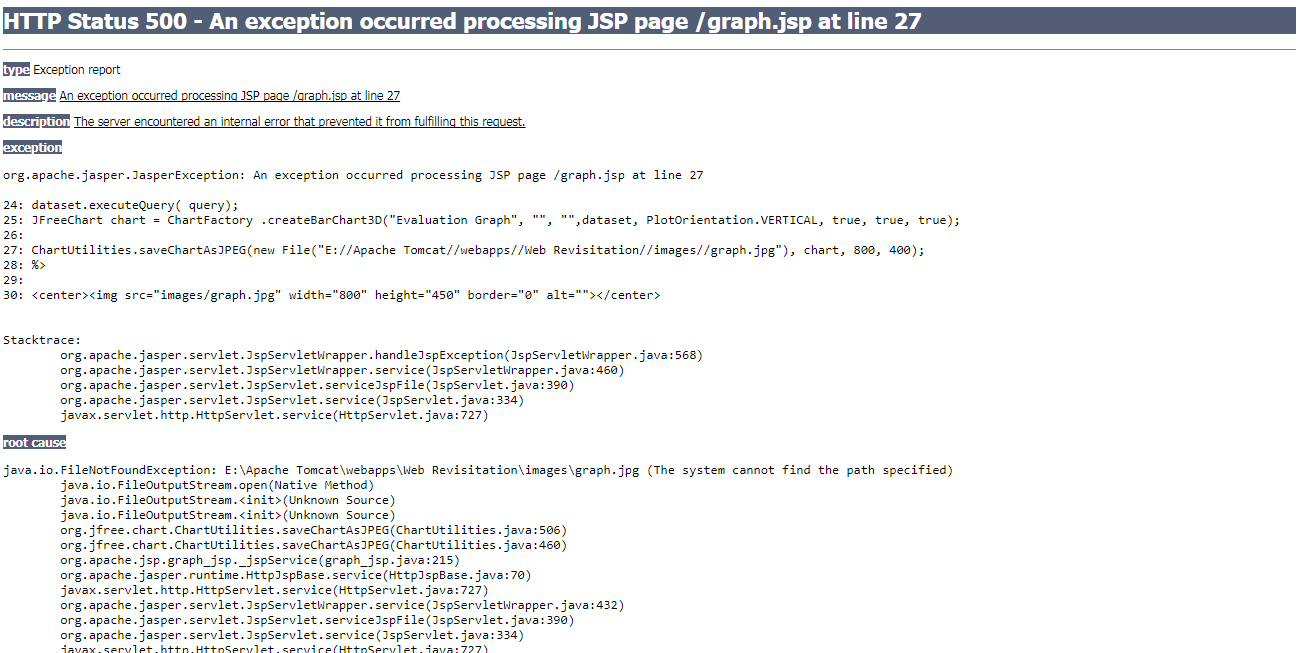
**Testing done when application is in development stage**

**Class version Error in our application**

****

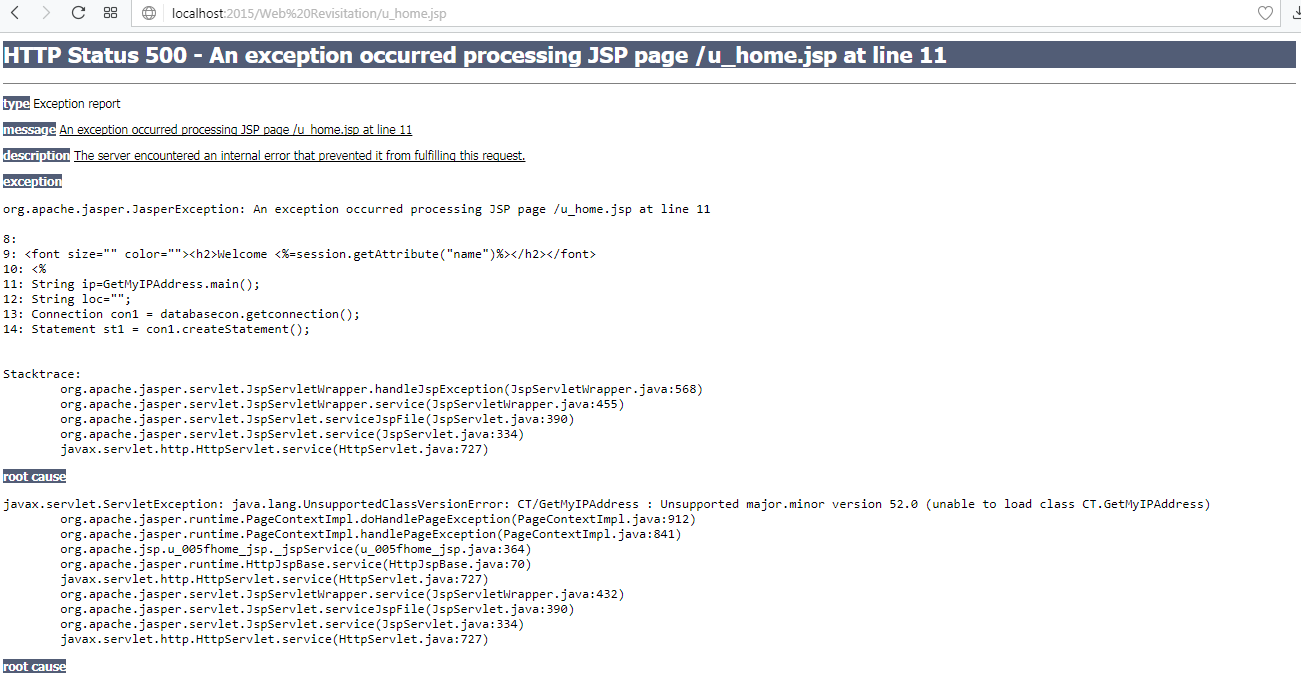
This Error Come when we move our application from one system to other and mainly when we version issues in the software’s we us

**Path related error in our application**

****

This Error Came when I have Performance Metrics to show in graph when I missed my server directly path in the system so we got this error in the applicant in development stage.

**Server Connection Error**

****

**Manual testing on project application**

**TEST CASES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #1 | | **Test Case Description** - Validations in Registration Form | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | User should be Registered | | 1 | Data should be valid | |
| **Test Condition** | | | | | |
| Entering data in registration form | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | User gives First and Last Name | Pop showing email verification message | | Enter valid email/password | Fail |
| 2 | Submitting the form without entering any details | Pop showing email verification message | | Enter email /password | Fail |
| 3 | User enters invalid format of email id | Pop showing email verification message | | Enter valid email id | Fail |
| 4 | User enters a phone number with < 10 digits | Pop showing email verification message | | Enter valid phone number | Fail |
| 5 | Entering valid username and password | Pop showing email verification message | | Pop showing email verification message | Pass |

Table 1 Registration test case

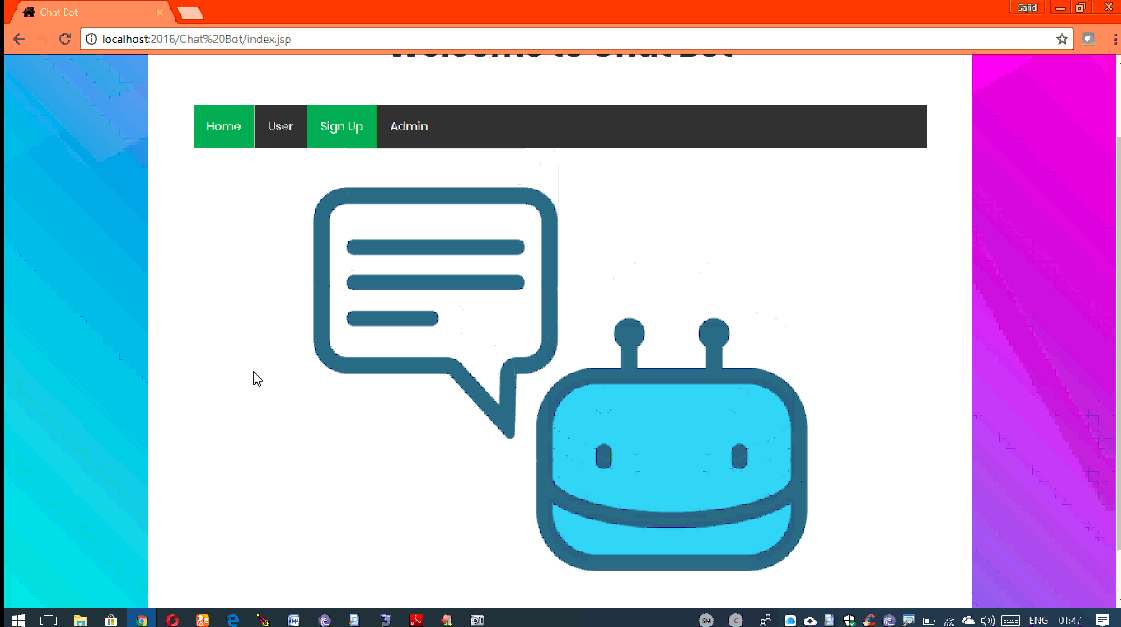
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #2 | | **Test Case Description** - Validations in Login Form | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | User should have an email id | | 1 | Data should be valid | |
| **Test Condition** | | | | | |
| Entering data in login form | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | User gives a email or password of <6 characters | User logged in | | Enter valid email/password | Fail |
| 2 | Submitting the form without entering any details | User logged in | | Enter email /password | Fail |
| 3 | User enters wrong Email and (or) password | User logged in | | Enter correct email /password | Fail |

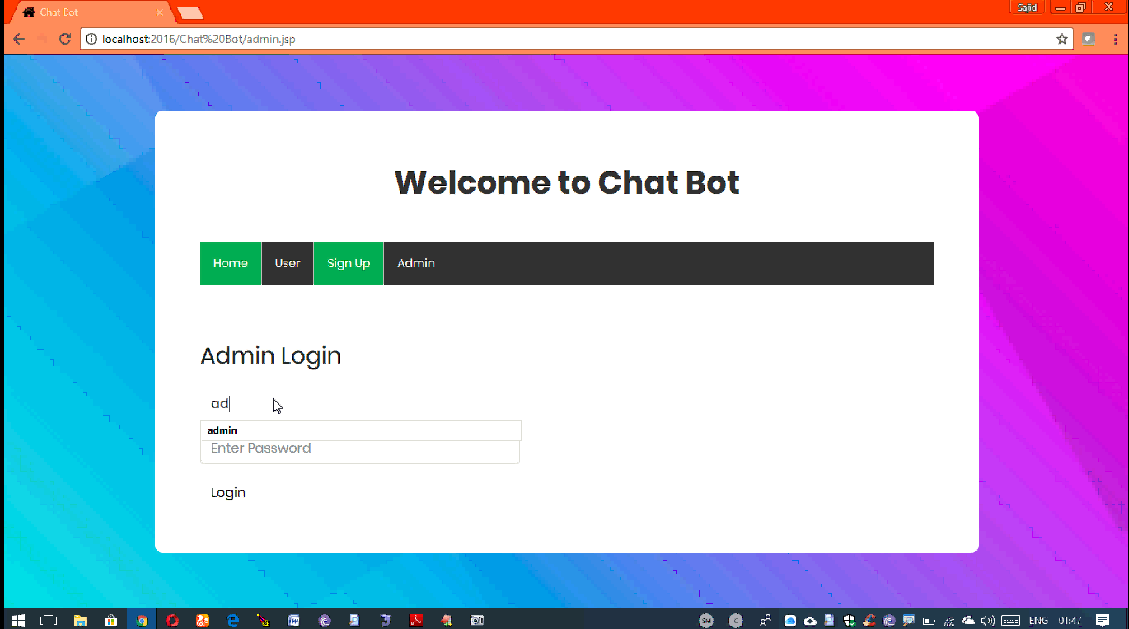
Table 2 Login test case

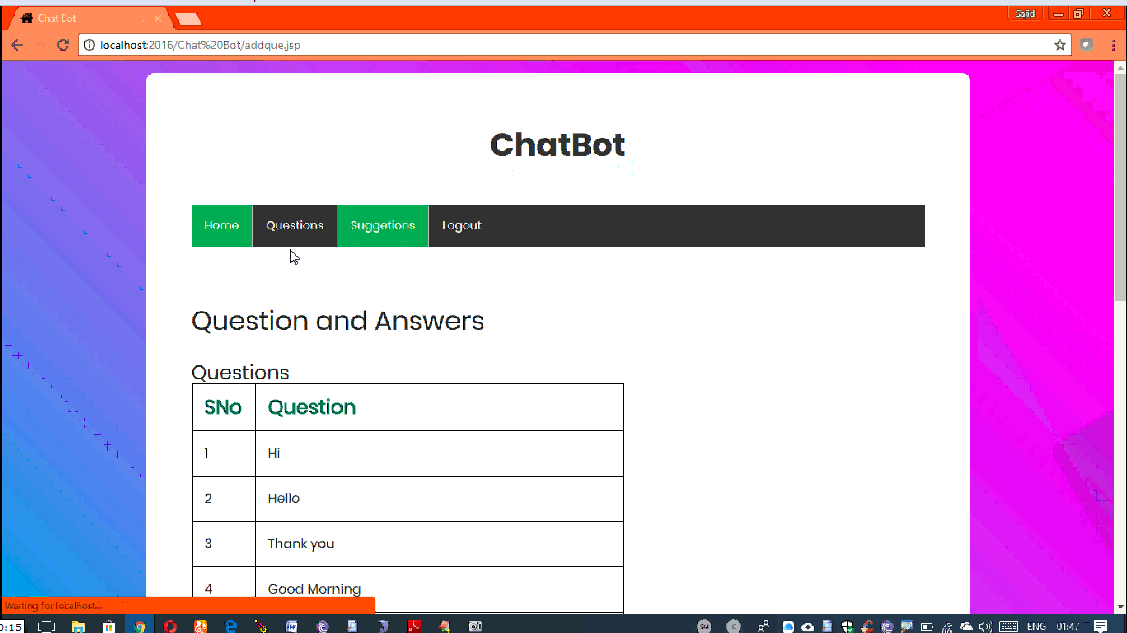
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** #3 | | **Test Case Description** - Search box | | | |
| **S#** | **Prerequisites** | | **S#** | **Test Data Requirement** | |
| 1 | Any user | | 1 |  | |
| **Test Condition** | | | | | |
| Entering data in search box | | | | | |
| **Step #** | **Step Details** | **Expected Results** | | **Actual Results** | **Pass/Fail/Not Executed/Suspended** |
| 1 | User enters a value and clicks on search | Results in the form of url’s | | Nothing happens | Fail |

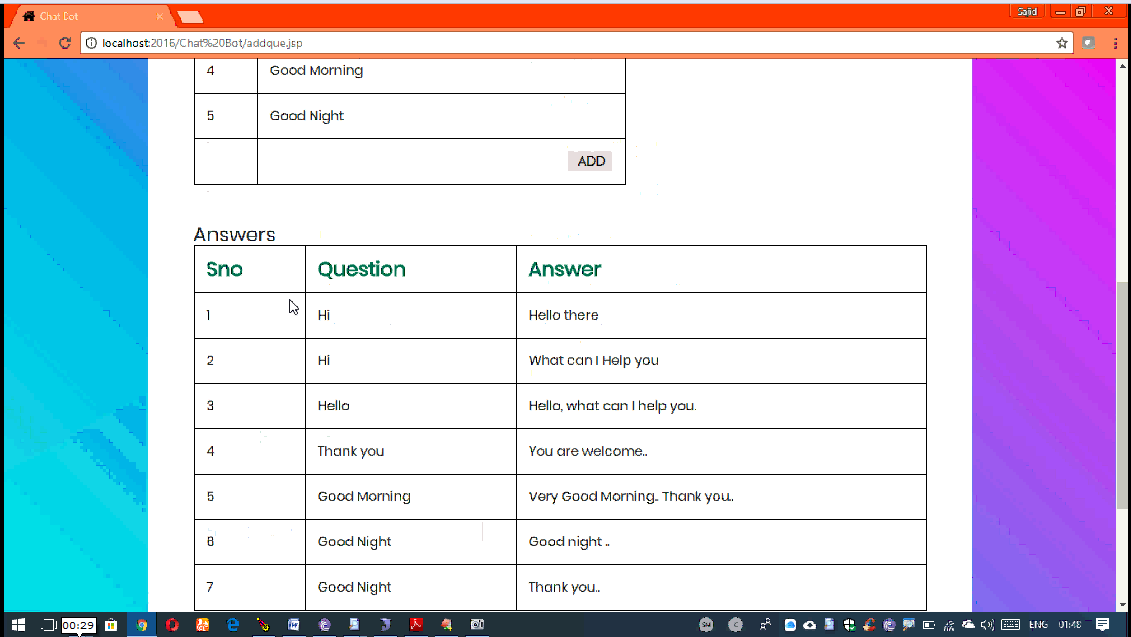
**CHAPTER 8**

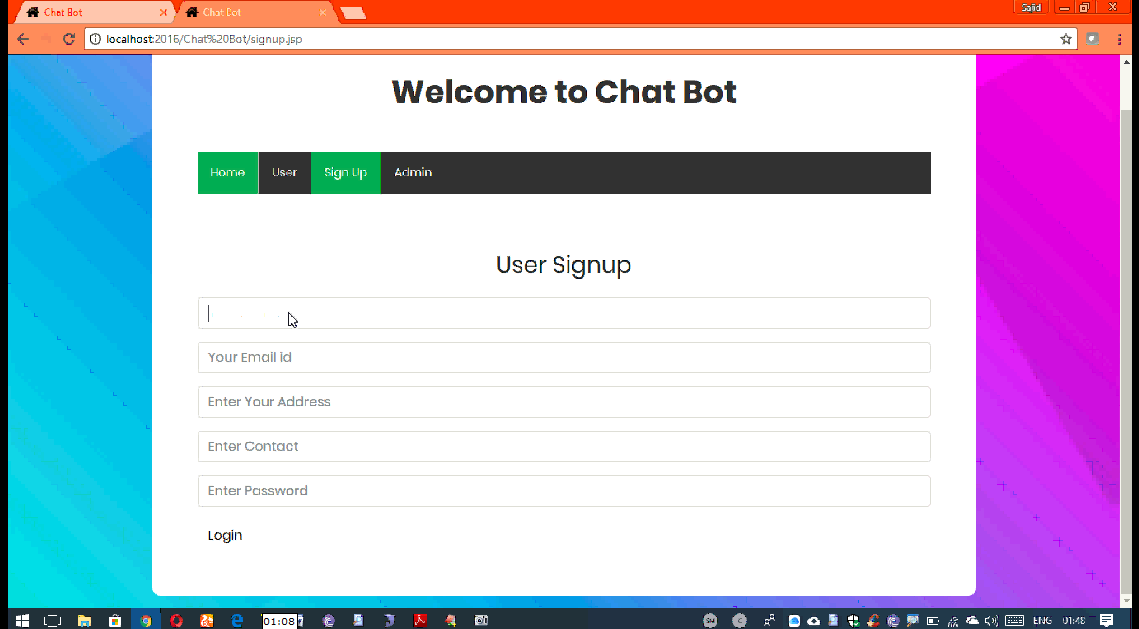
**SCREEN SHOTS**

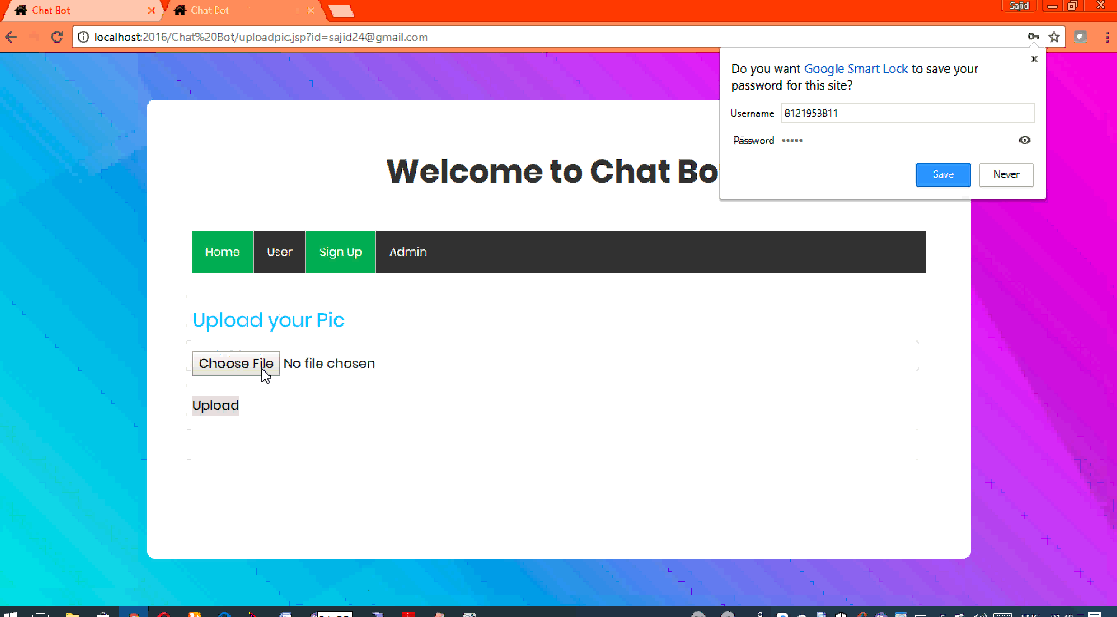
****

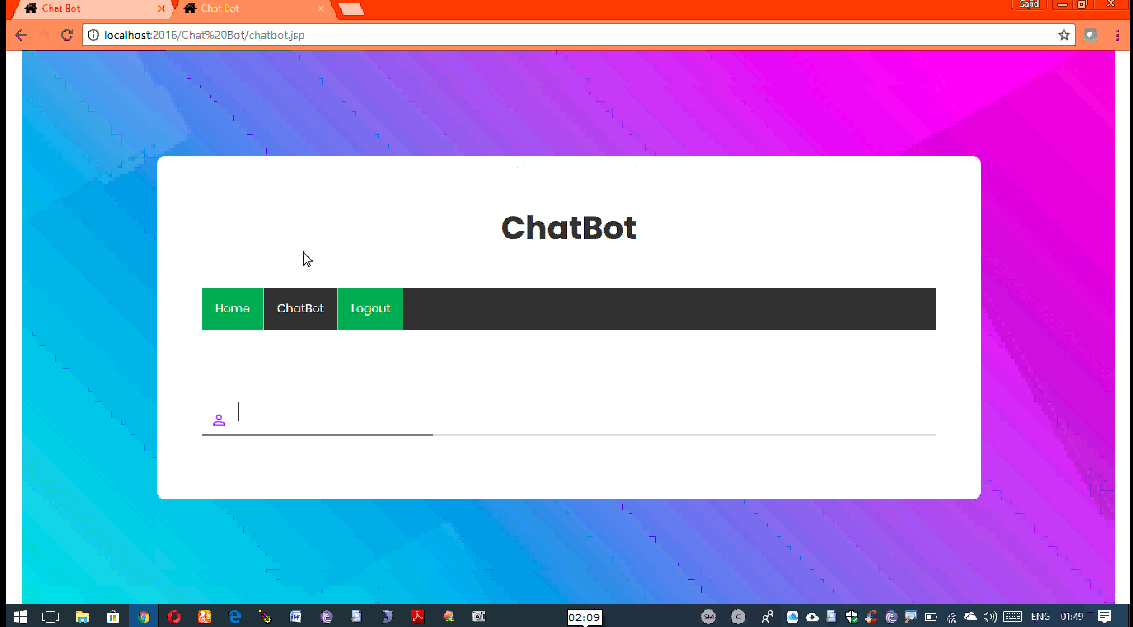
****

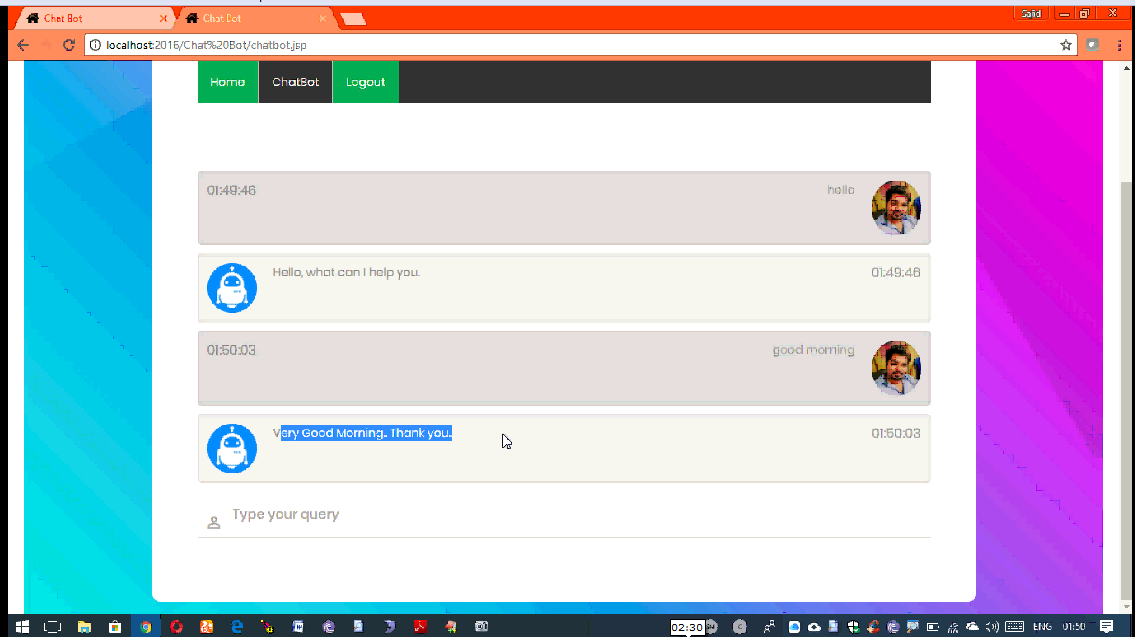
****

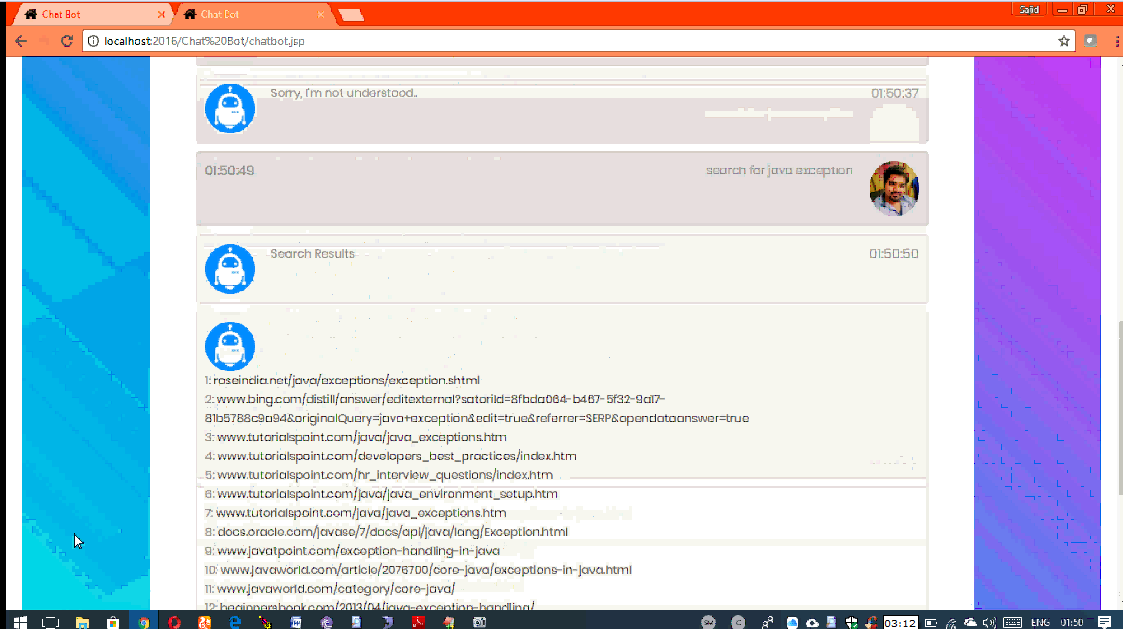
****

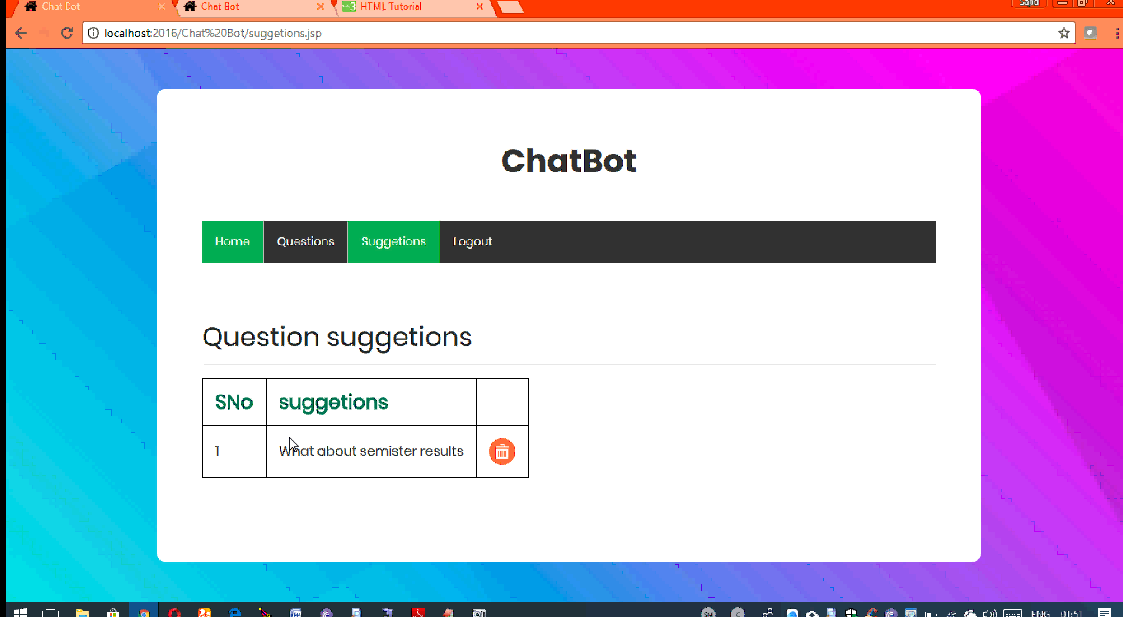
****

****

****

****

****

****

**CHAPTER 9**

**CONCLUSION**

Chatbots are a great way to use artificial intelligence to make it easier for business managers, large and small, to grow business via messaging. The developments in the area of NLP (Natural Language Processing). It is a big challenge for the designers if the user data input is not constrained. In the language, we can express ourselves in many ways, while our intentions behind what we are saying remain the same. You can see that how fast the user can encounter problems if they do not speak in the language of the machine and if they do not only choose the opportunities proposed to them by the machine. Therefore, chat bots may bring revolution in many fields including marketing and services.

**CHAPTER 10**

**Future Enhancement**

In the future, we would like our college to use the chatbot for new students to, find out information about the college, and any other queries.

For example the students can inquire whether college is working on the next day or not. The database has to be uploaded with proper questions and answers.

This is a great and innovative way to bring the college administration closer to students for better interaction.

**CHAPTER 11**

**BIBLIOGRAPHY**

[**https://www.aclweb.org/anthology/E/E17/E17-3.pdf#page=99**](https://www.aclweb.org/anthology/E/E17/E17-3.pdf#page=99)

[**https://www.cognizant.com/whitepapers/the-chatbot-imperative-intelligence-personalization-and-utilitarian-design-codex2469.pdf**](https://www.cognizant.com/whitepapers/the-chatbot-imperative-intelligence-personalization-and-utilitarian-design-codex2469.pdf)

[**http://students.iitk.ac.in/eclub/assets/documentations/summer12/chat-bot.pdf**](http://students.iitk.ac.in/eclub/assets/documentations/summer12/chat-bot.pdf)

[**https://media.readthedocs.org/pdf/chatterbot/latest/chatterbot.pdf**](https://media.readthedocs.org/pdf/chatterbot/latest/chatterbot.pdf)