Department of computer Engineering and Applications GLA University,17 km. Stone NH#2, Mathura-Delhi Road, Chaumuhan, Mathura – 281406 U.P (India)

Declaration

We hereby declare that the work which is being presented in the Mini Project "Cloud based Student Chatbot **",**in fulfilment of the requirements for Mini Project viva voice, is an authentic record of our own work carried under the supervision of **Mr. Ambrish Gangal, Assistant Professor, Department of CEA.**

Signature:

_____

Name of Candidate:

Shubham Mishra

University Roll No. :171510051

Signature:_____

Name of Candidate:

Pranav Kumar Gupta

University Roll No. : 171510039

Signature:_____

Name of Candidates:

Akshat Gaur

University Roll No. : 171510007

ACKNOWLEDGEMENT

This project consumed huge amount of work, research and dedication. Still, implementation would not have seen possible if we did not have a support of many individuals and organization. Therefore, we would like to extend our sincere gratitude to all them.

First of all, we are thankful to **Mr. Ambrish Gangal ,GLA University, Mathura** for providing necessary guidance concerning projects implementation.

We are also grateful to for provision of expertise, and technical support in the implementation. Without their superior knowledge and experience, the project would like in quality of outcomes, and thus their support has been essential.

We would like to express our sincere thank towards volunteer researchers who devoted their time and knowledge in the implementation of this project.

Nevertheless, we express our gratitude toward our families and colleague for their kind cooperation and encouragement which help us in completion of this project.

ABSTRACT

Abstract—Human-Computer Text-Based recognition is gaining thrust as a medium of computer interaction. It's being very important for new students to know about their college, as many students need help for solving their queries related to a college campus, canteens, classrooms, etc but college faculty or senior students are not able to reach out to every student and solve their problems related to college. This paper proposes a system architecture that will try to overcome the above problem by using an automated text-chatbot that will initiate text-based conversion among that customer(student) and help the customer for resolving the issue by providing a human way interaction. To provide a highly robust, scalable architecture, this system is implemented on AWS services like Amazon Lex.

CONTENTS

List of Figures

Chapter 1

Introduction

As a student, we faced many problems when we came to college as a new student and then we realized that it is very difficult for a new student to get a full conclusion about the college. In college, there are many challenges to be faced by new students like not aware of the college people, seniors, places, classrooms, canteens, faculty. Also, new student hesitates in asking their question and not solving their problem. New students are thinking that they are new, so, there is no one to solve their problem. If a new student is a hosteler, then it is tougher to face these challenge.

Solution for this is that student must have someone assosiated with them to assist them but it is not posssible for one person to perform such task for every student so we should intoduce one technology that can return responses to multiple querries at a time. So we are trying to create a chatbot that can response regards to querry.

1.1 OVERVIEW AND MOTIVATION

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.

1.2 OBJECTIVE

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 botwhich 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 loginSystem 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

1.2 ORGANISATION OF THE PROJECT REPORT

- In the next chapter we deal with the softwarer equirement.
- In the next chapter the technology review is defined which includes terms like Aws lex,Aws polly, Aws cloud Formation.
- In the next chapter we deal with the software design. It includes the various dataflow diagrams, use case diagram etc.
- The following chapter implementation which includes all the outputscreens.
- The next chapter deals with the softwaretesting.
- Final chapter has the conclusionpart.

Chapter 2

Software Requirement Analysis

In this chapter we define the feasibility analysis to various non-functional requirements like performance, reliability, availability, scalability, security, maintainability, and portability. It then consists of the module description. In the end use case aregiven.

2.1 DEFINE THE PROBLEM

The first step of analysis process involve the identification of need. The success of a system depends largely on how accurately a problem is defined, thoroughly investigated and satisfy the Students needs by providing user friendly environment. The origin of the problem istohelp customer, to provide a seamless way to get a quote easily, Apart from quotes, user can also know about transportation work through our website instead of making an appointment. After successful completion of the project user can able to book vehicle according to their need.

2.2 REQUIREMENTANALYSIS

2.2.1 HARDWAREREQUIREMENT

Minimum requirement will be as follows:

- RAM: 2GB
- A good internet connection
- A computer or laptop

2.2.2 SOFTWAREREQUIREMENTS

- Operating System: Window 7 or a b o v e , l i n u x .
- Aws account
- Web Browser

Chapter 3

Technology Review

3.1 AWS

Amazon Web Services (AWS) is a subsidiary of Amazon that provides on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered pay-as-you-go basis. In aggregate, these cloud computing web services provide a set of primitive abstract technical infrastructure and distributed computing building blocksandtools. One of these services is Amazon Elastic Compute Cloud, which allows users to have at their disposal a virtual cluster of computers, available all the time, through theInternet.

Aws lex

Amazon Lex is a service for building conversational interfaces into any application using voice and text. Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognize the intent of the text, to enable you to build applications with highly engaging user experiences and lifelike conversational interactions. With Amazon Lex, the same deep learning technologies that power Amazon Alexa are now available to any developer, enabling you to quickly and easily build sophisticated, natural language, conversational bots chatbots.

Speech recognition and natural language understanding are some of the most challenging problems to solve in computer science, requiring sophisticated deep learning algorithms to be trained on massive amounts of data and infrastructure. Amazon Lex democratizes these deep learning technologies by putting the power of Amazon Alexa within reach of all developers. Harnessing these technologies, Amazon Lex enables you to define entirely new categories of products made possible through conversational interfaces.

As a fully managed service, Amazon Lex scales automatically, so you don't need to worry about managing infrastructure. With Amazon Lex, you pay only for what you use. There are no upfront commitments.

Benefits

Easy to use

Amazon Lex provides an easy-to-use console to guide you through the process of creating your own chatbot in minutes, building conversational interfaces into your applications. You supply just a few example phrases and Amazon Lex builds a complete natural language model through which your user can interact using voice and text, to ask questions, get answers, and complete sophisticated tasks.

Seamlessly deploy and scale

With Amazon Lex, you can build, test, and deploy your chatbots directly from the Amazon Lex console. Amazon Lex enables you to easily publish your voice or text chatbots to mobile devices, web apps, and chat services such as Facebook Messenger, Slack, and Twilio SMS. Once published, your Amazon Lex bot processes voice or text input in conversation with your end-users. Amazon Lex is a fully managed service so as your user engagement increases, you don't need to worry about provisioning hardware and managing infrastructure to power your bot experience.

Built in integration with AWS

Amazon Lex provides built-in integration with AWS Lambda, AWS MobileHub and Amazon CloudWatch and you can easily integrate with many other services on the AWS platform including Amazon Cognito, and Amazon DynamoDB. You can take advantage of the power of the AWS platform for security, monitoring, user authentication, business logic, storage and mobile app development.

Cost effective

With Amazon Lex, there are no upfront costs or minimum fees. You are only charged for the text or speech requests that are made. Amazon Lex' pay-as-you-go pricing and low cost per request make it a cost-effective way to build conversational interfaces anywhere. With the Amazon Lex free tier, you can easily try Amazon Lex without any initial investment.

AWS3 Cloud Formation

AWS CloudFormation provides a common language for you to model and provision AWS and third party application resources in your cloud environment. AWS CloudFormation allows you to use programming languages or a simple text file to model and provision, in an automated and secure manner, all the resources needed for your applications across all regions and accounts. This gives you a single source of truth for your AWS and third party resources.

BENEFITS

Model it all

AWS CloudFormation allows you to model your entire infrastructure and application resources with either a text file or programming languages. The AWS CloudFormation Registry and CLI make it easy to manage third party resources with CloudFormation. This provides a single source of truth for all your resources and helps you to standardize infrastructure components used across your organization, enabling configuration compliance and faster troubleshooting.

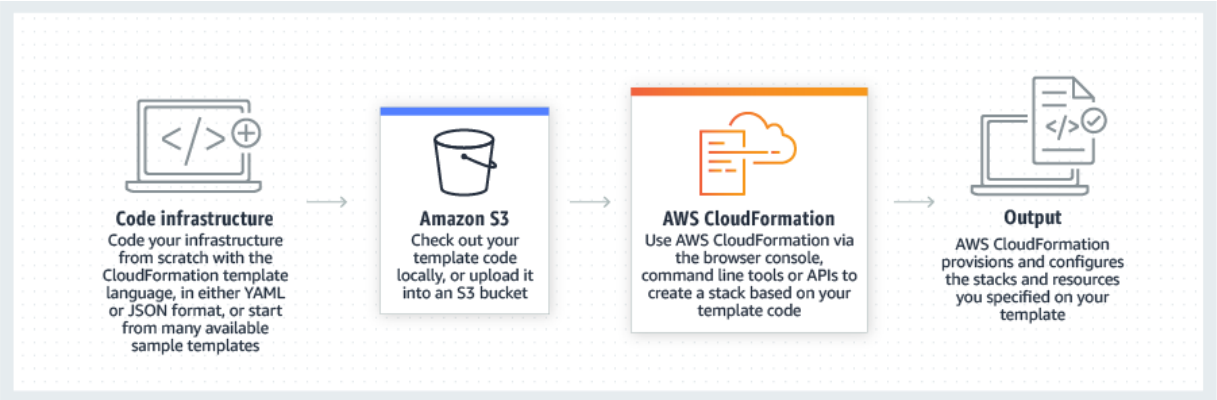
Automate & deploy

AWS CloudFormation provisions your application resources in a safe, repeatable manner, allowing you to build and rebuild your infrastructure and applications, without having to perform manual actions or write custom scripts. CloudFormation takes care of determining the right operations to perform when managing your stack, orchestrating them in the most efficient way, and rolls back changes automatically if errors are detected.

It's just code

Codifying your infrastructure allows you to treat your infrastructure as just code. You can author it with any code editor, check it into a version control system, and review the files with team members before deploying into production.

Working

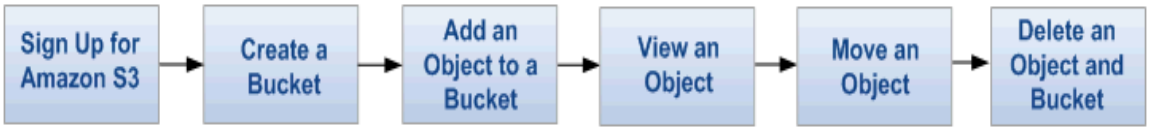


New – Amazon Connect and Amazon Lex Integration

Amazon Connect is a self-service, cloud-based contact center service that makes it easy for any business to deliver better customer service at lower cost. Amazon Lex is a service for building conversational interfaces using voice and text. By integrating these two services you can take advantage of Lex‘s automatic speech recognition (ASR) and natural language processing/understading (NLU) capabilities to create great self-service experiences for your customers. To enable this integration the Lex team added support for 8kHz speech input – more on that later. Why should you care about this? Well, if a bot can solve the majority of your customer’s requests your customers spend less time waiting on hold and more time using your provision

Amazon Simple Storage Service

Amazon Simple Storage Service (Amazon S3) is storage for the Internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web. You can accomplish these tasks using the AWS Management Console, which is a simple and intuitive web interface. This guide introduces you to Amazon S3 and how to use the AWS Management Console to complete the tasks shown in the following figure.



AWS LAMBDA

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app

Benefits

NO SERVERS TO MANAGE

AWS Lambda automatically runs your code without requiring you to provision or manage servers. Just write the code and upload it to Lambda.

CONTINUOUS SCALING

AWS Lambda automatically scales your application by running code in response to each trigger. Your code runs in parallel and processes each trigger individually, scaling precisely with the size of the workload.

SUBSECOND METERING

With AWS Lambda, you are charged for every 100ms your code executes and the number of times your code is triggered. You pay only for the compute time you consume.

CONSISTENT PERFORMANCE

With AWS Lambda, you can optimize your code execution time by choosing the right memory size for your function. You can also enable Provisioned Concurrency to keep your functions initialized and hyper-ready to respond within double digit milliseconds.

Chapter4

Software Design

Software design is the process of implementing software solutions to one or more sets of problem. One of the main components of software design is the software requirements analysis (SRA).

4.1 Data FlowDiagram:

A DFD also known as 'bubble chart', has the purpose of clarifying system requirements and identifying those transformations. It shows the flow of data through a system. It is a graphical tool because it represents a picture. The DFD may be partitioned into levels that represents increasing information flow and functional details. Four simple notation are used to complete a DFD. These notation are given below:-

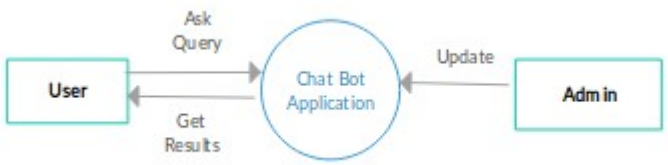
Data Flow:-The data flow is used to desired the movement of information from one part of the system to another part. Flow represent data in motion. It is pipe line through which information flow.

Data flow is represented by an arrow.

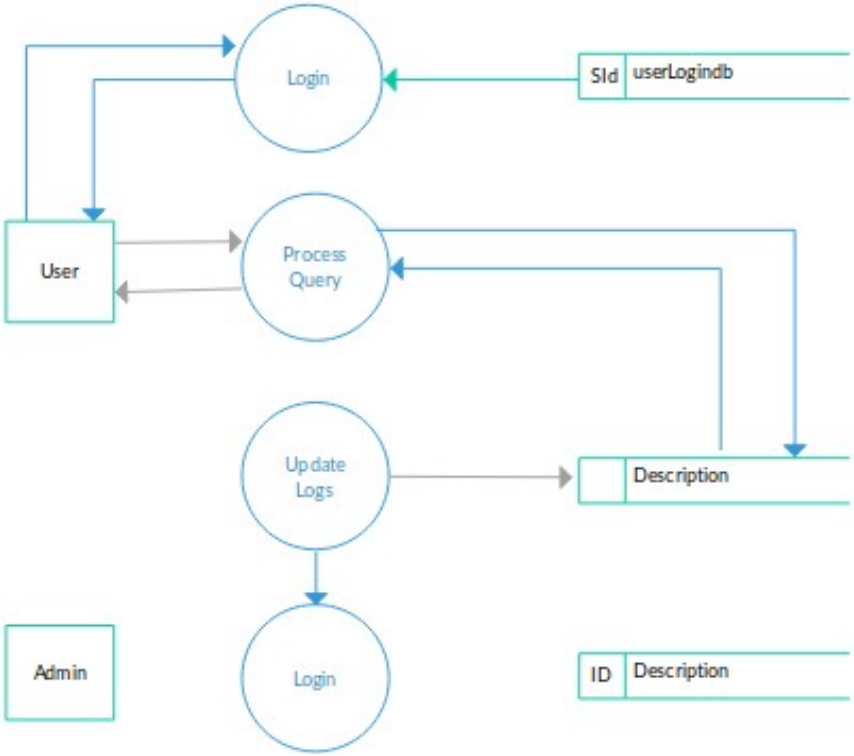
Process:- A circle or bubble represents a process that transform incoming data to outgoing data. Process shows a part of the system that transforms input to outputs.

External Entity :-A square define a source or destination of system data. External entities represent any entity that supplies or receive information from the system but is not a part of the system,

Data Store:-The data store represent a logical file. A logical file can represent either a data store symbol which can represent either a data structure or a physical file on disk. The data store is u-sed to collect data at rest or a temporary repository of data. It is represented by open rectangle.



Data flow diagram 0



DATA FLOW DIAGRAM1

4.2 USE CASE Diagram:

Use Case Diagram gives a graphic overview of the actors involves in the system, different functions needed by those actors and how these different functions are interacted. The purpose of this is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose. Use Case Diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements.

Fig 4.2.1 Use Case Diagram

4.3 ActivityDiagram:

An activity diagram is a replacement for flow charts. It captures actions and their results. It supports and encourages parallel activities. It is important for business modelling. It is useful for concurrent programs. It is used to describe use cases. It is used in workflow modelling, multithreaded programming, analyzing a use case, and understanding workflow across different use cases.

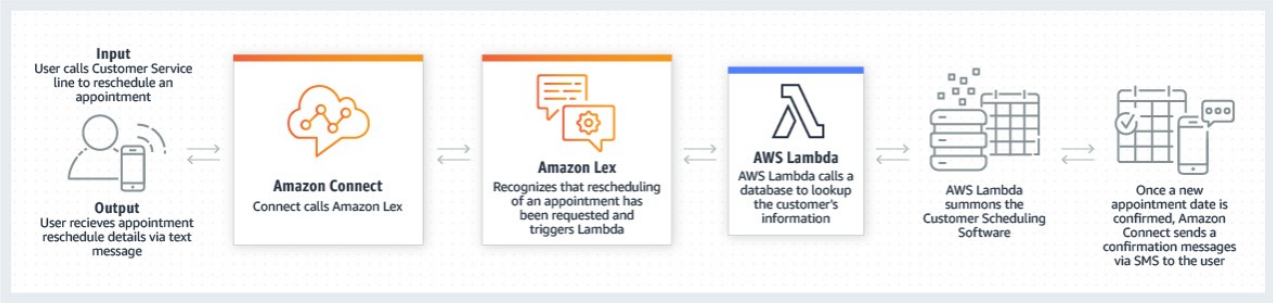


Fig 4.3.1 Activity Diagram

Chapter 5

Implementation and User Interfaces

In this chapter we describe the various interfaces in the project like user interfaces, hardware interface, software interfaces, and communication interfaces. In the end output screens are shown depicting various screens in the system.

5.1 USER INTERFACES

User interface is part of website and is designed such a way that it is expected to provide the user insight of the software. The user interface provides fundamental platform for human-computer interaction. UI can be graphical, text-based, audio-video based, depending upon the underlying hardware and software combination.

5.2 HARDWARE INTERFACES

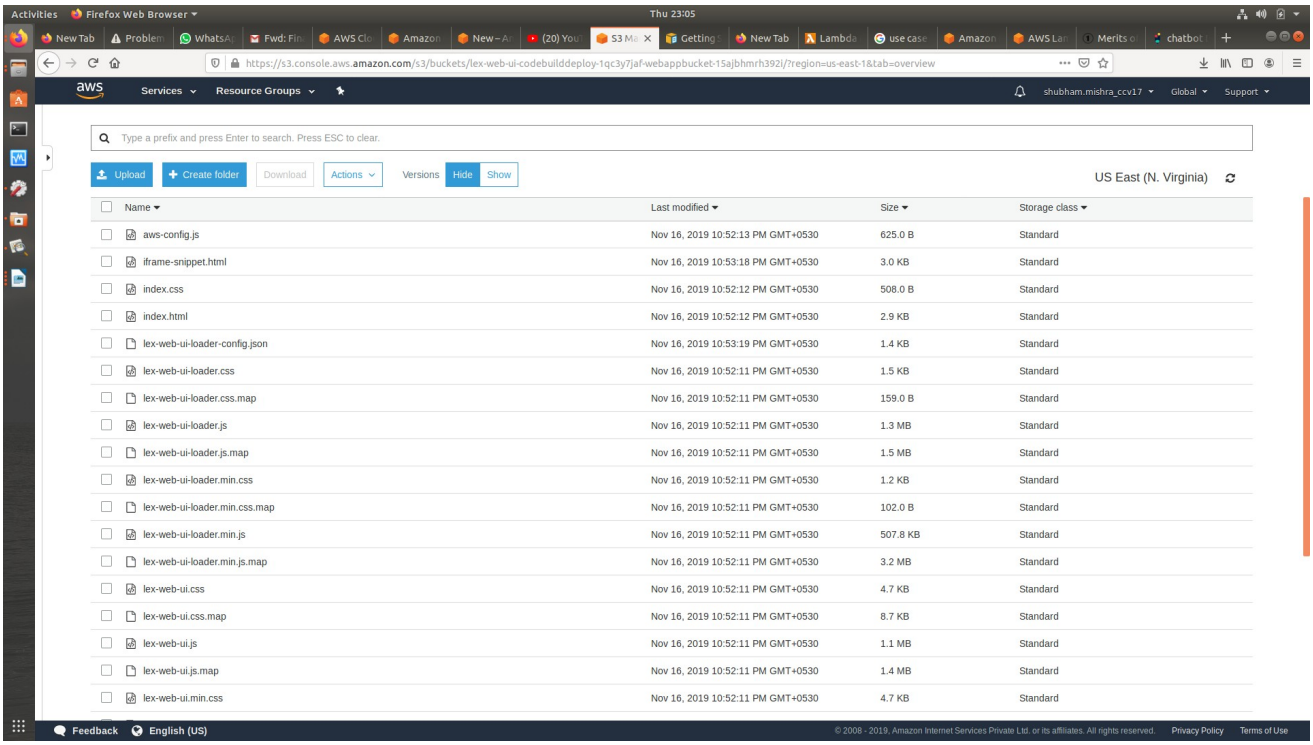
- RAM: 2GB
- A good internet connection
- A computer or laptop

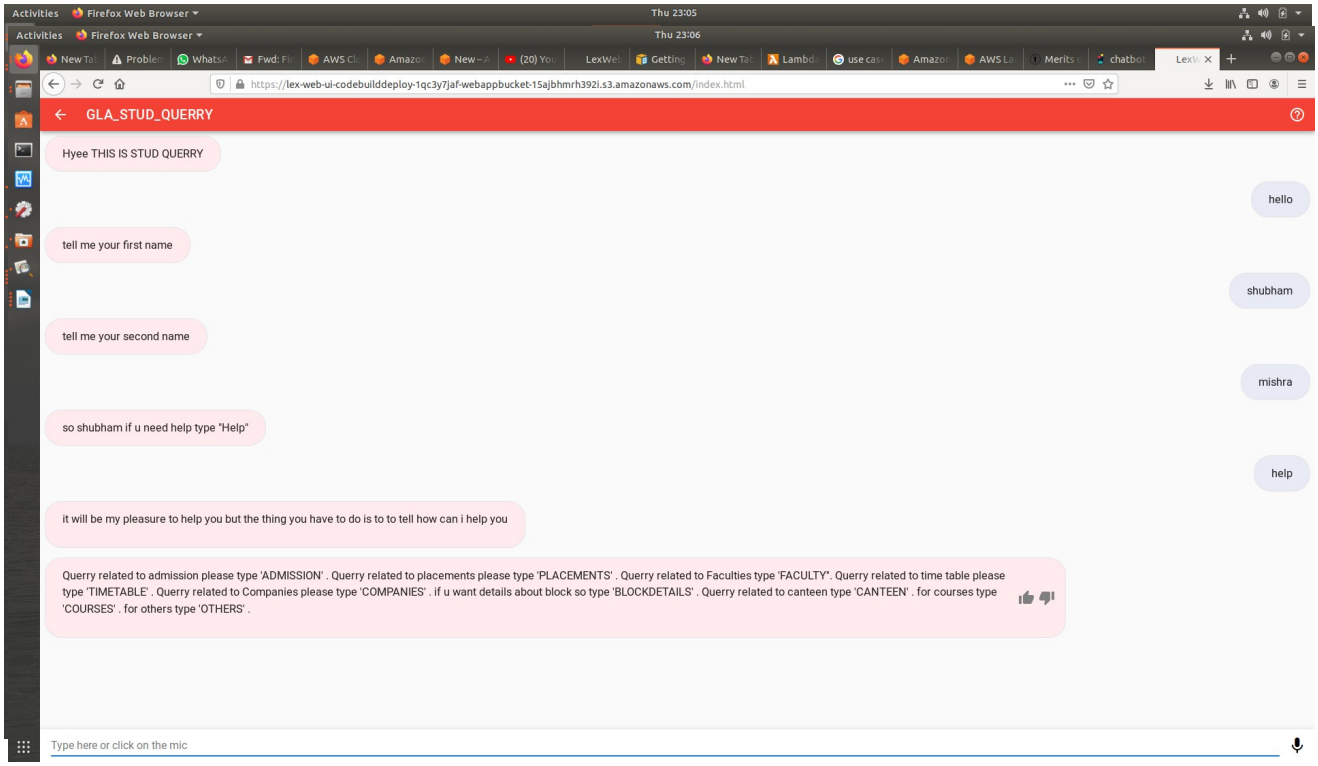
5.3 SOFTWARE INTERFACES

- Operating System: Window 7 or a b o v e
- aws
- web browser

ATTACHMENTS

ATTACHMENTS





Chapter 7

Conclusion

The main objectives of the project were to develop an algorithm that will beused to identify answers related to user submitted questions. To develop a database were all the related data will be stored and to develop a web interface. The web interface developed had two parts, one for simple users and one for the administrator. Abackground research took place, which included an overview of the conversation procedure and any relevant chat bots available. A database was developed, which stores information about questions, answers, keywords, logs and feedback messages. A usable system was designed, developed and deployed to the web server on two occasions. An evaluation took place from data collected by potential students of the University. Also after received feedback from the first deployment, extra requirements were introduced andimplemented

BIBLOGRAPHY

For Aws services- https://docs.aws.amazon.com/