PARSHVANATH CHARITABLE TRUST'S

A.P. Shah Institute of Technology

Thane, 400615

Academic Year: 2021-22

Department of Computer Engineering

CSL605 SKILL BASED LAB COURSE: CLOUD COMPUTING

Mini Project Report

- **Title of Project** :Handyman Service Booking Bot
- **Year and Semester** : T.E. (Sem VI)
- **Group Members Name and Roll No.** :
  JANHAVI ANAP(03)
  NIDHI SINGH(69)
  ISHANEE REVANKAR(54)
  NIDHI HENIYA(21)

Table of Contents

Problem Definition:

Booking any type of service for any problem that one faces is tedious since one has to physically visit the shop to book a handyman or a worker. To ease this out, this project aims to build a system or a platform which is a web-based Chatbot for Handyman service booking. Handyman Service Booking Chatbot is designed to let a user book a handyman service in just one click.

Introduction:

A **handyman**, also known as a **fixer**, **handyperson** or **handyworker**, is a person skilled at a wide range of repairs, typically around the home. These tasks include trade skills, repair work, maintenance work, are both interior and exterior, and are sometimes described as "side work", "odd jobs" or "fix-up tasks". Specifically, these jobs could be light plumbing jobs such as fixing a leaky toilet or light electric jobs such as changing a light fixture or bulb. The term *handyman* increasingly describes a paid worker, but it also includes non-paid homeowners or do-it-yourselfers. Many people can do common household repairs. Handyman service booking bot have a User Interface through which users interact with the chatbot.Once the user sends respective service request or utterance the chatbot replies accordingly.It asks for information like name,address,phone number and pin code for the purpose of user booking confirmation. Further it asks for the problem faced by the user.Once the user inputs their problem,related service is booked (on the basis of problem type) after confirmation from the user.Bot books the service via our interactive frontend.

The cloud services used in this project are:

- Amazon Lex
- Amazon EBS

Kommunicate

We are integrating our interactive frontend with the chatbot using Kommunicate

Description:

Cloud services used:
Amazon Lex
AWS Elastic Beanstalk

Software requirements:
For frontend:
HTML
CSS

Integration Tool:
Kommunicate

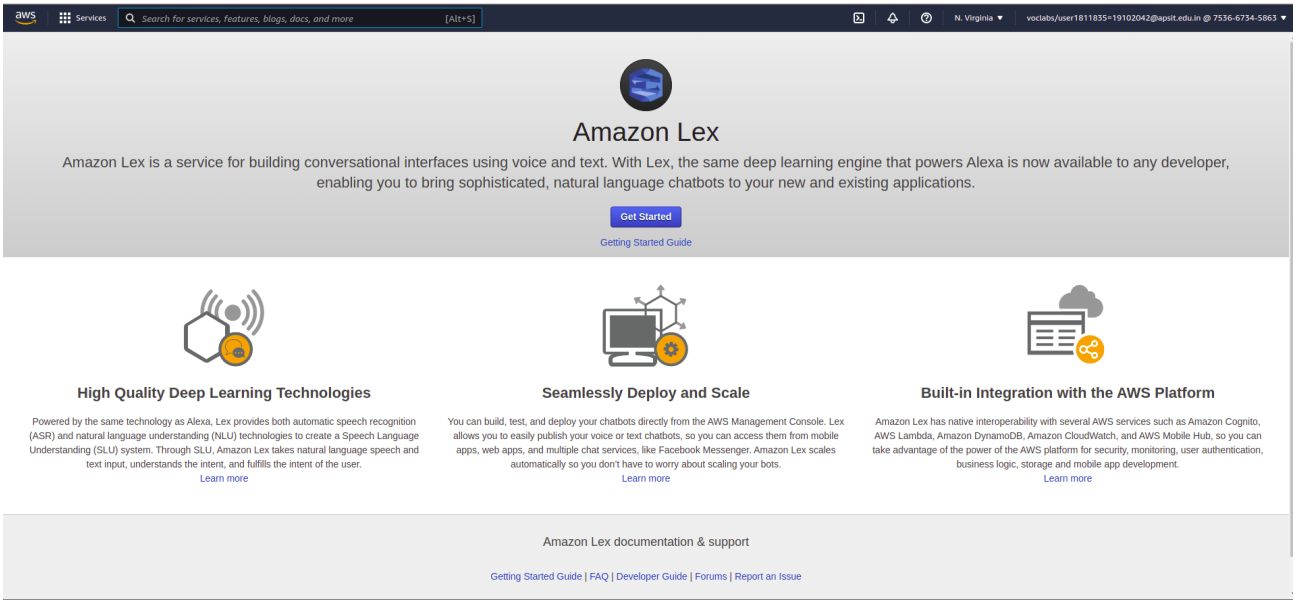
Amazon Lex:

Amazon Lex is an AWS service for building conversational interfaces for applications using voice and text. Amazon Lex provides the deep functionality and flexibility of natural language understanding (NLU) and automatic speech recognition (ASR) .

It helps to build conversational chatbots quickly. We just have to specify the basic conversation flow in the Amazon Lex console,without the knowledge of deep learning. Amazon Lex manages the dialogue and dynamically adjusts the responses in the conversation. Using the console, you can build, test, and publish your text or voice chatbot.This bots can be later integrated into web applications,mobile applications,etc.

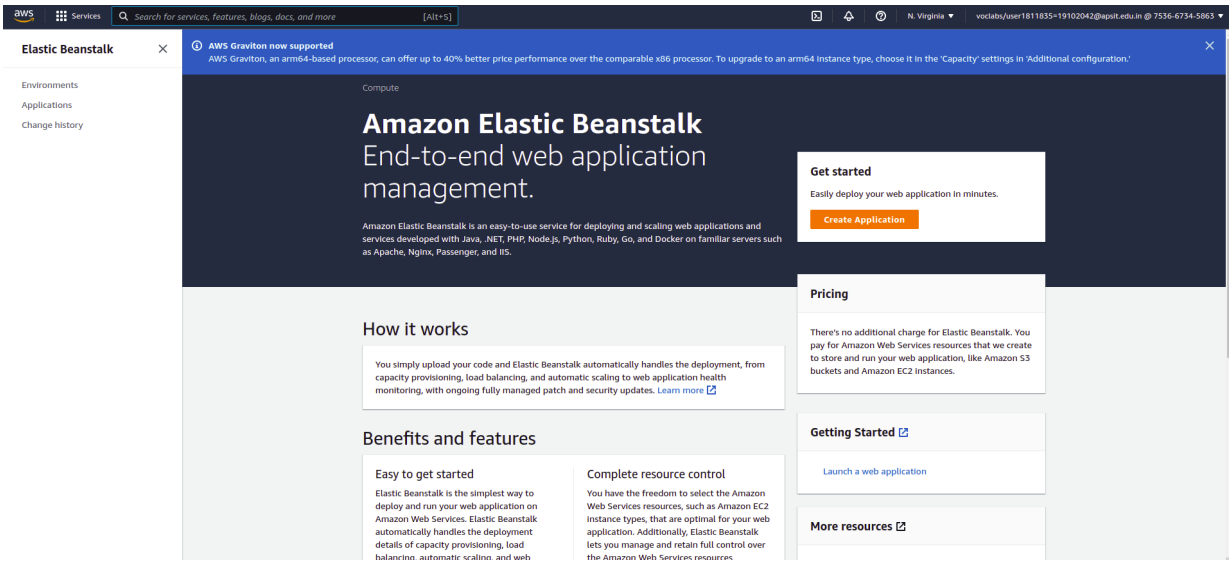
Typical steps performed when working with Amazon Lex:

1. Create a bot and configure it with one or more intents that you want to support. You add the configuration so that the bot is able to understand the user's goal (intent), engage in conversation with the user to elicit information, and, after the user provides the necessary data, fulfill the user's intent.
2. Test the bot. You can use the test window client provided by the Amazon Lex console.
3. Publish a version and create an alias.
4. Deploy the bot. You can deploy the bot on platforms such as mobile applications or messaging platforms such as Facebook Messenger.



AWS Elastic Beanstalk:

AWS Elastic Beanstalk is used for deploying web applications and services.It handles all the deployment,from capacity provisioning,load balancing,auto-scaling to application health monitoring.AWS Elastic Beanstalk allows you to quickly deploy applications and services without having to worry about configuring underlying resources, services, operating systems or web servers
It is pay only for the resources needed to store and run your applications.



Kommunicate:

Kommunicate is a human+bot hybrid customer support automation platform that provides real-time, proactive, and personalized support for growing businesses. Kommunicate is used to integrate chatbots with the website.

Lex bot can be integrated into your website in a few simple steps:

Step 1: Create a free Kommunicate account

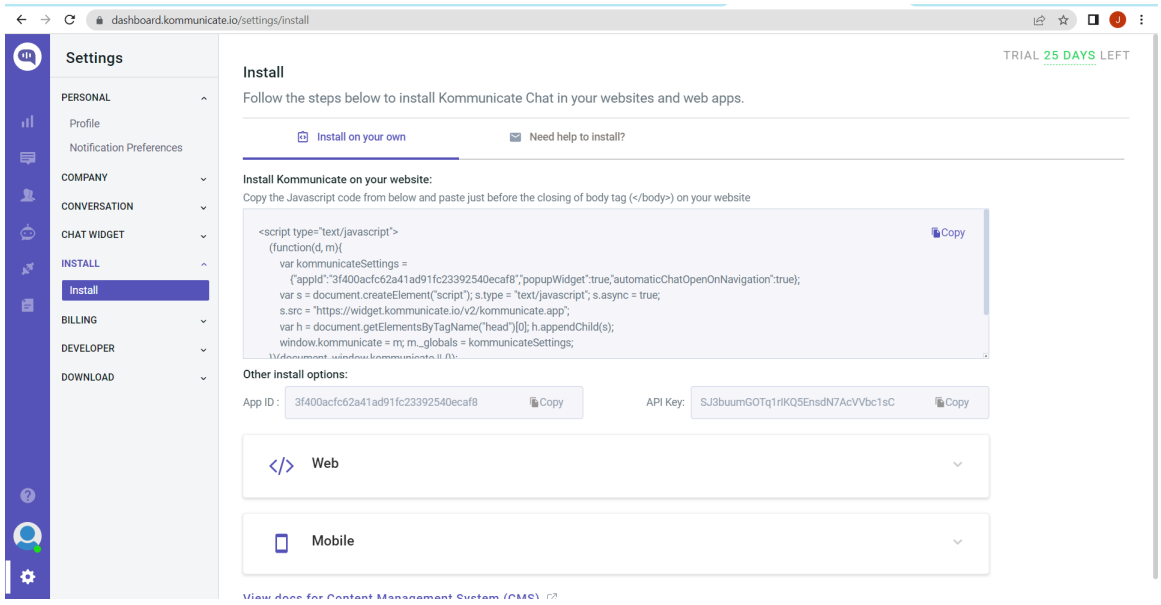
Step 2: Connect your Amazon Lex bot

Step 3: Give your bot an identity

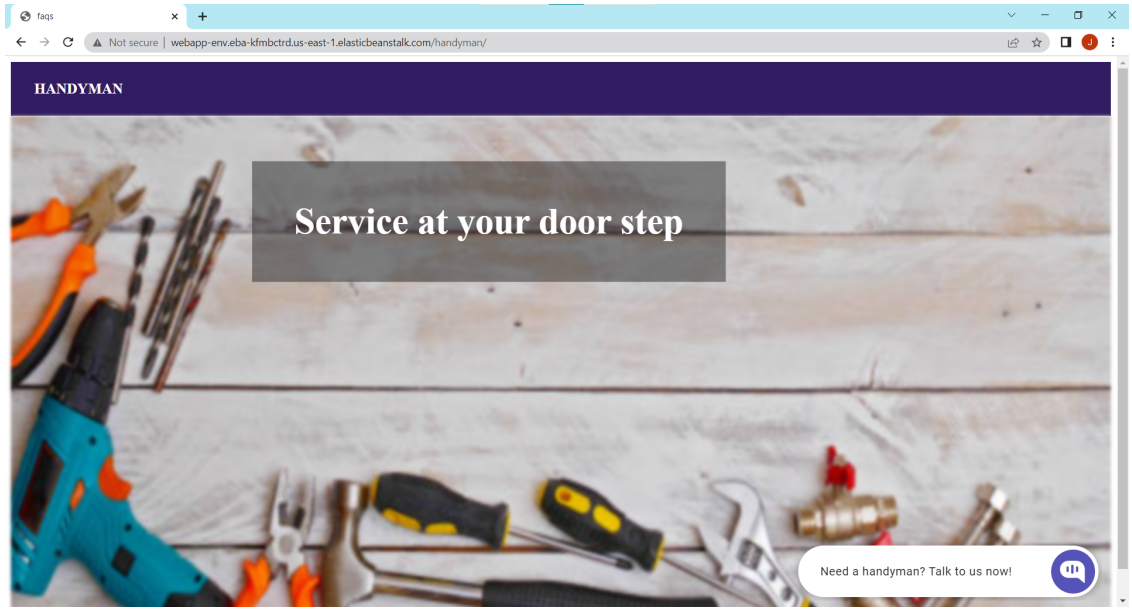
Step 4: Enable/Disable human handoff

Step 5: Assign all the incoming conversations to your Lex bot

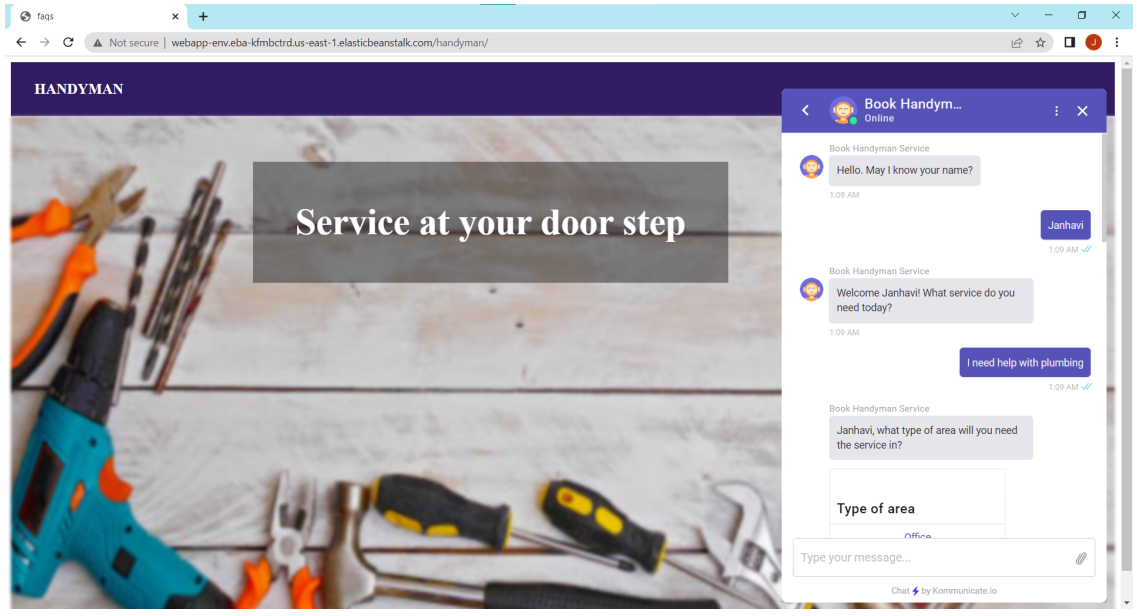
Step 6: Install the Kommunicate on your website by pasting the JavaScript code in the source code



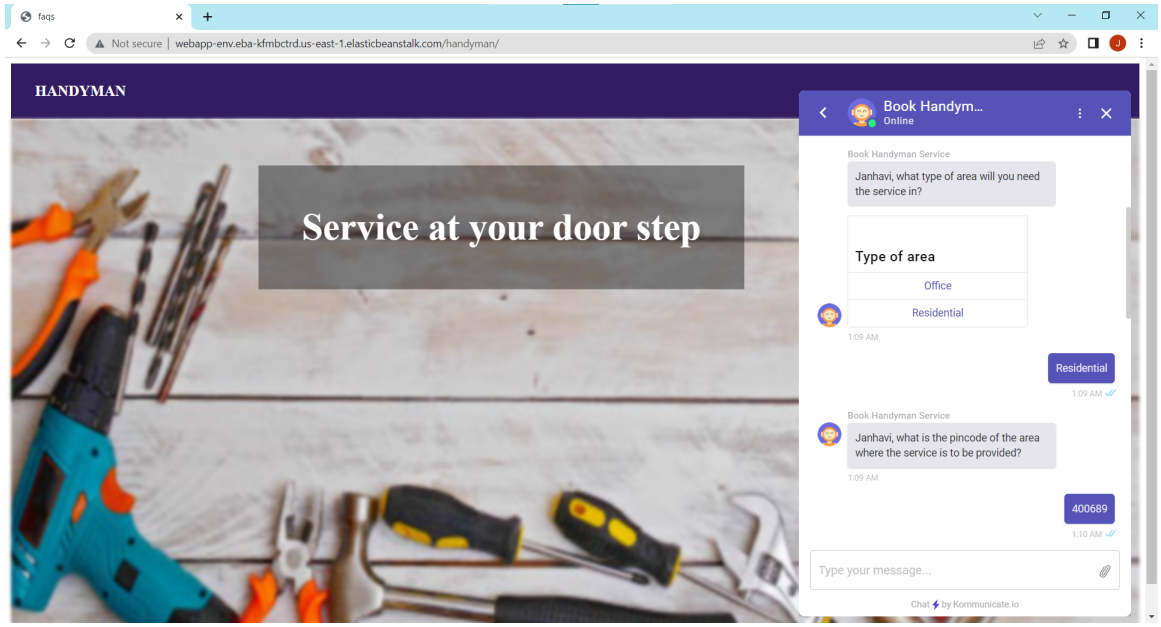
Implementation:



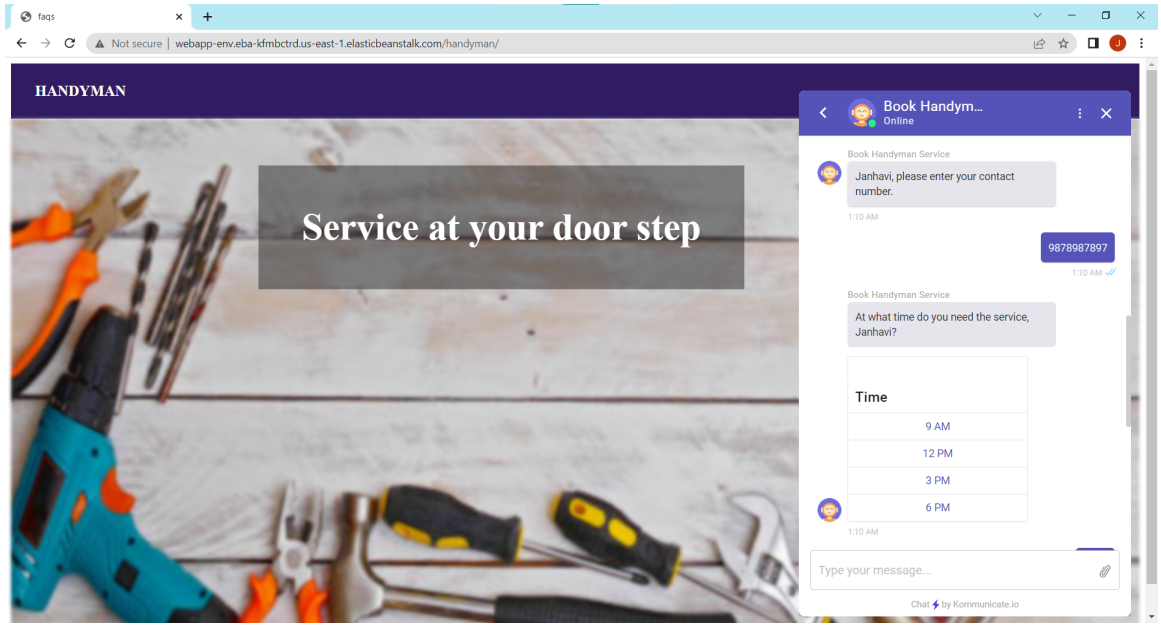
On opening the website link the chatbot pops up text to grab the attention of the user. By clicking on the bot icon the chat window opens.



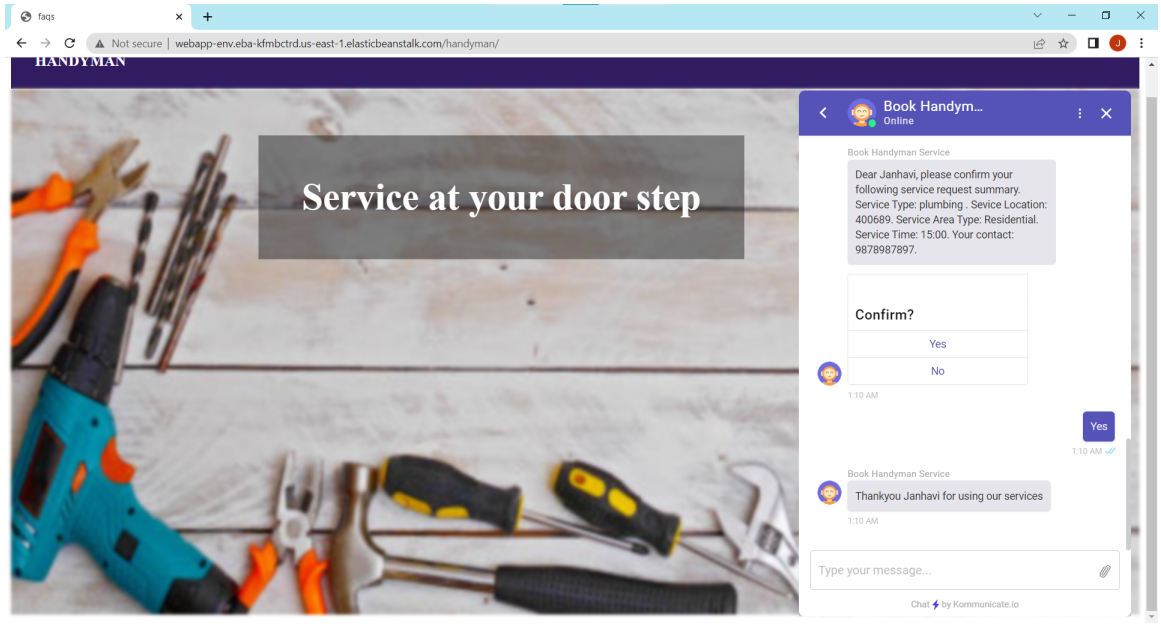
The bot initiates the conversation by first asking the name of the user. It stores that name and addresses all the questions using that name.
It then asks the user what service is needed to which user should reply. The bot detects the keywords and stores the type of service needed.



The bot then asks what type of area the service is needed in. The user can enter the response by either typing or by selecting from the cards provided.
The bot then asks for the pincode in which the user wants the service in.



The bot asks the user to enter their contact details and stores the reply as numeric value. It then asks the user to enter the time when the service is required. Again, the user can input the time by typing or by selecting one of the cards.



The bot also asks the user to confirm the details. Upon user's confirmation the bot confirms the order.

Learning outcomes:

We have learnt to use various cloud service like:

- Learnt to use Amazon Lex ,for creating a chatbot
- Used AWS Elastic Beanstalk ,for deploying the web application on cloud

We also learnt to use Kommunicate to integrate the Lex bot with the frontend