

**THE CHATBOT**

Submitted in partial fulfillment of the requirements for the award of degree of

**BACHELOR OF ENGINEERING  
IN  
COMPUTER SCIENCE & ENGINEERING**



**Submitted to: Er. Nilesh Bharti  
(ECODE: E11105)**

**Submitted By:-Keshav Kant Mishra**

**Project members:-**

**Aji Nair (19BCS1903)**

**Keshav Kant Mishra (19BCS1887)**

**Anil Kr. Pradhan(19BCS1898)**

**Project Teacher  
(Er. Nilesh Bharti-E11105)**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**Chandigarh University, Gharuan**

**August,2021**

## **CERTIFICATE**

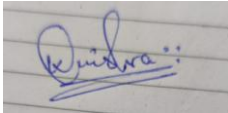
This is to certify that the work embodied in this Project Report entitled “**The Chatbot**” being submitted by “**Anil Pradhan** ” - UID “**19BCS1898**” , 5<sup>th</sup> Semester for partial fulfillment of the requirement for the degree of “ **Bachelor of Engineering in Computer Science & Engineering** ” discipline in “ **Chandigarh University** ” during the academic session Aug-Dec 2021 is a record of bonafide piece of work, carried out by student under my supervision and guidance in the “ **Department of Computer Science & Engineering** ”, **Chandigarh University**.

**APPROVED & GUIDED BY: Er. Nilesh Bharti\_(Project teacher)**

## DECLARATION

I, student of **Bachelor of Engineering in Computer Science & Engineering, 5<sup>th</sup> Semester** , session: **Aug – Dec 2021, Chandigarh University**, hereby declare that the work presented in this Project Report entitled “**The Chatbot**” is the outcome of my own work, is bona fide and correct to the best of my knowledge and this work has been carried out taking care of Engineering Ethics. The work presented does not infringe any patented work and has not been submitted to any other university or anywhere else for the award of any degree or any professional diploma.

**Keshav Kant Mishra**  
**(19BCS1887)**



**STUDENT DETAIL**



**APPROVED & GUIDED BY:** **Er. Nilesh Bharti (Project teacher)**

To our parents, teachers and all the well wishers out there . . .

## **ACKNOWLEDGEMENT**

This project is a synergistic product of many minds and has been accumulated over the last few months. This has been a special project brought to fruition through the efforts of some very special people. Many people contributed enthusiastically to this project, which really came together in the last few weeks before deadline. For their continuous guidance and valuable advice I would like to take this opportunity to thank: Firstly, I would like to thank Chandigarh University who has granted me this opportunity to prepare a project which has helped me to gain knowledge beside my studies and which is also definitely going to be useful in future. I would like to place on record my deep sense of gratitude to my parents for financial wisdom and inspiration that have guided and helped me from day one. My project guide, Er Nilesch Bharti(Project teacher), who came in and batted down the hatches when things were flying about. And I would like to thank all those who have helped me contribute their valuable insights and time for this project.

## **ABSTRACT**

Chatbot is widely popular now-a-days and catching speed as an application of computer communication. Some programs respond intelligently like human. This type of program is called a Chatbot. This paper addresses the design and implementation of a Chatbot system. We will also study another application where Chatbots could be useful and techniques used while designing a Chatbot. A Chat-bot is a software application used to conduct an online chat conversation via text or text-to speech, instead of providing direct contact with a live human agent.

Designed to convincingly simulate the way a human would behave as a conversational partner. In the proposed system, we presented a chatbot that generates a dynamic response for online client's queries. The Proposed System is based on Artificial Intelligence-powered Chatbot. The web based platform provides a vast intelligent base that can help simulate problem- solving for humans.

This proposed chatbot identifies the user context which triggers the particular intent for a response. Since it is responding dynamic response the desired answer will be generated for the user. The proposed system used machine learning algorithms to learn the Chatbot by experiencing various user's responses and requests. After referring to 17 IEEE papers and 13 Standard papers our research finding a state that the strong point of Chat-bot is that it comes to use innumerable fields of our daily life.

Nowadays chat-bot is started to becoming so robust because Artificial Intelligence aids the human touch in every conversation, chat-bot understand the user's query, and trigger an accurate response. The objective of this project is that chatbots can help to reduce the dependency of an organization on humans and also minimize the need for a different system for different processes.

## Table of Contents

|  |  |
|--|--|
| CERTIFICATE.....                                 |  |
| DECLARATION .....                                |  |
| ACKNOWLEDGEMENT .....                            |  |
| ABSTRACT.....                                    |  |
| Glossary .....                                   |  |
| INTRODUCTION .....                               |  |
| <b>“What is AI &amp; ML?”</b> .....              |  |
| How does AI & ML work.....                       |  |
| 1. Rich library ecosystem .....                  |  |
| 2. Flexibility .....                             |  |
| 3. Simple and Consistent .....                   |  |
| 4. Platform Independent.....                     |  |
| What exactly is Deep Learning?.....              |  |
| Why is Deep Learning is Popular these Days?..... |  |
| <b>Background</b> .....                          |  |
| Chatbots in the Recent Past.....                 |  |
| Chatbots: Then and Now .....                     |  |
| <b>Summary</b> .....                             |  |
| System Design .....                              |  |
| 3.1 Project Overview .....                       |  |
| DESIGN OF CHATBOT.....                           |  |
| IMPLEMENTATION PROCESS .....                     |  |
| 4. Experimental Results and Discussion.....      |  |
| Result from first Testing .....                  |  |
| Future Scope .....                               |  |
| Summary.....                                     |  |
| Conclusion .....                                 |  |
| <b>BIBLIOGRAPHY</b> .....                        |  |

# **Glossary**

Action task – Action tasks collect, modify, and post information in systems of record, like scheduling an appointment, searching for a product, or updating critical information.

Administration & analytics – The Kore.ai Platform provides enterprise grade visibility and control of all enterprise bots, user groups, and security, as well as a commitment to compliance in even the most highly regulated areas like healthcare and financial services. [Learn more](#)

Alert task – Alert tasks deliver timely, relevant, and personalized information to customers and employees directly from enterprise systems. Bots poll the system for user requested updates in real-time.

API – An API, or Application Programming Interface, is a set of definitions, protocols, and tools for building application software. It helps developers by essentially providing the building blocks for a program.

Artificial Intelligence (AI) – AI is the development of computer systems that are able to perform tasks that normally require human-like intelligence, like decision making, speech recognition and understanding, translation between languages, and more.

Auto-NLP – A term we use at Kore.ai to describe our synonym-based approach to natural language processing. It allows chatbots to communicate understand intent variations right out of the gate, thus being speech enabled “automatically”

Automatic Message Formatting – The pre-programmed responses for tasks built into Kore.ai’s NL engine.

Automated Speech Recognition (ASR) – Our Platform can integrate Automated Speech Recognition Engine to enable bots to process voice-driven interactions and communicate outside of traditional text-based interfaces.

**B**

Bots (also see Chatbot) – Short for a computer program that acts as an intelligent intermediary between people, digital systems, and Internet-enabled things

Bot Builder Tool – Kore.ai’s web-based Bot Builder allows enterprises to customize chatbot use cases, channels, tasks, security, and more. It also provides a framework in which developers can design, test, and deploy chatbots in a consistent process at scale.

Bot Framework – A framework is a skeleton that provides some basic building blocks and generic functionality for building chatbots (like ML/ NLP or a Dialog Builder), but requires additional user-written code or other third-party services (to match the functionality of actual platform). Frameworks often are composed of piecemeal components from different vendors.

Bots Platform – Kore.ai’s Platform is the only enterprise-grade chatbot platform offered as a SaaS which gives enterprises every component necessary for designing, developing, deploying, and managing AI-rich chatbots.



**Bot Connector** – For enterprises with systems behind firewalls, the bot connector is a tool that allows you to securely access the data in those systems for use in a bot.

**Bot context** – User or task information that is captured at the bot level and can be used in context with some or all of the users of that bot.

## C

**Channel** – A channel is another word for any of the various communication platforms where a bot can live such as SMS, email, mobile apps, websites, messaging apps and more.

**Chatbot** – Short for a computer program that acts as an intelligent intermediary between people, digital systems, and Internet-enabled things. Find out more about chatbots with our Bots 101 guide.

**Chat logs** – Histories of all recorded human-to-bot interactions.

**Cisco Spark** – Cisco’s all-in-one communication platform, which is also a supported channel for the Kore.ai bots and an integration partner for chatbot development.

**Cloud (or Cloud computing)** – Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand.

**Cloud Connector** – Provides an agent that runs behind your enterprise firewall that acts as a bridge to facilitate secure data exchanges between on-prem systems and Kore.ai’s cloud based infrastructure.

**Component reusability** – The ability for developers to use components they’ve already built in the Bot Builder, like APIs, synonyms, tasks, etc. and apply them to other bots.

**Context** (see also, Bot Context, Enterprise Context, Session Context, User Context) – The information that a chatbot pulls from a conversation with a user that it can leverage when performing tasks. Contextual data can vary in importance, utility, and lifespan.

**Conversational Commerce** – A term coined by Chris Messina in 2016, which is another way of describing how digital economies will be driven by text and voice based interfaces and experiences.

**Conversational UI** – Another way of describing text and voice-based interfaces, which don’t require graphical elements for use, like Amazon’s Alexa or Apple’s Siri.

## D

**Data retention** – The continued storage of an organization’s data for compliance or business matters.

**Deep learning** – An area of Machine Learning that is based on learning data representations as opposed to task specific algorithms.

**Deployment** – The process of publishing a bot to communication channel where it will be engaged by users.

**Dialog task** – Dialog tasks are advanced tasks that developers design with logic-driven business processes and pre-established workflows. Bots key off the primary request intent to

accomplish the task at hand, then go above and beyond to execute sub-intents and additional workflows.

**Dialog Builder** – The Kore.ai Dialog Builder gives designers and developers the flexibility to manipulate the entire dialog process of a bot interaction and string together complex workflows in a GUI-based tool.

**E**

**Ecommerce** – Any monetary transaction conducted on the internet.

**Ediscovery** – Any process in which electronic data is sought, located, secured, and searched with the intent of legal use. The Kore.ai Platform supports e-discovery.

**Email** – A supported channel for Kore.ai bots.

**Encryption** – The process of converting information or data into a code, especially to prevent unauthorized access.

**End-to-end** – A way of describing the Kore.ai Bots Platform which signifies that it includes all the component features to take enterprises from the very beginning of the chatbot development process, through deployment and management.

**Enterprise analytics** – The central dashboard within the Kore.ai Platform where administrators can get visibility into key metrics, pull detailed reports, and track bot usage (i.e. number of executed tasks, most popular channels, most active users, user enrollment, etc.)

**Enterprise bots store** – A bot store that an enterprise sets up for a select group of users to access any custom built bots.

**Enterprise-grade** – A way of describing all of the components and capabilities of the Kore.ai Bots Platform that are specifically designed to match the highest enterprise standards, including administration, analytics, security, compliance, and more.

**Enterprise context** – Information that represents company-wide rules and standards that apply to all users and bots, such as a company travel policy, or expense limits.

**Entity** – Entities are the fields, data, or words the developer designates are necessary for a chatbot to complete the user's request. An entity could be a date, a time, a location, a description or any number of things.

**Entity extraction** – This is the process by which the Kore.ai NL engine identifies words from a user's utterance to ensure all available fields match the task at hand. If the chatbot needs an entity to complete the task after initial extraction, it will prompt the user for it.

**F**

**Facebook Messenger** – A supported channel for chatbots built on Kore.ai's Bots Platform, primarily used when companies build bots for end customers.

**FAQ** – The primary data source chatbots use to pull information to complete knowledge tasks. Coming soon to the Platform will be the ability for website and data based knowledge and document-based knowledge.

**Framework** – A framework is a skeleton that provides some basic building blocks and generic functionality for building chatbots (like ML/ NLP or a Dialog Builder), but requires additional user-written code or other third-party services (to match the functionality of an actual platform). Frameworks often are composed of piecemeal components from different vendors.

**Fundamental Meaning** – Fundamental Meaning is an approach to NLP that's all about understanding words themselves. Each user utterance is broken down word-for-word to search for intent (what the user is asking it to do) and entities (the necessary data needed to complete a task). Learn more about this approach and the Kore.ai NL engine.

## G

**Glip** – A supported channel for chatbots built on Kore.ai's Bots Platform.

**Graphic User Interface (GUI)** – A visual way of interacting with an app or system, such as buttons, images, windows, icons, menu forms, and more.

## H

**Hosting** – Enterprises have the choice of hosting the Kore.ai Bots Platform on prem or in the cloud via AWS.

## I

**Information and Communication Technologies (ICT)** – ICT refers to technologies that provide access to information through telecommunications. It is similar to Information Technology (IT), but focuses primarily on communication technologies. This includes the Internet, wireless networks, cell phones, and other communication mediums.

**Information task** – Information tasks lookup data or pull reports based on specific parameters and quickly return easy-to-consume results that are convenient for users.

**Interface** – A shared

## M

**Machine Learning** – Using algorithms, patterns, and training data, machine learning allows computers to find hidden insights without being explicitly programmed. Learn more about the way Kore.ai uses machine learning for natural language enablement.

**Managed Services Provider (MSP)** – Most often an IT provider that manages and assumes responsibility for providing a defined set of services to its clients either proactively or as the MSP determines that services are needed

**Memory** – Bots can remember actions, data, and contextual details to maintain conversation continuity and take helpful actions. The developer can designate how long the bot remembers information as either short term or long term memory.

**Message Broker** – Consumes all user inputs and system outputs, standardizes for a common messaging paradigm, and redirects to the appropriate endpoints.

Middleware – The Kore.ai Platform Middleware contains the Message Broker, Message Store, and built-in encryption to create a flawless conversational experience by ensuring messages are received, secured, and exchanged in real time.

N

Natural Language (NL) – The method by which users can talk to systems in everyday language like text and speech, rather than programming language.

Natural Language Processing (NLP) – The process by which a chatbot or any other system understands and processes requests in common language, rather than programming language. NLP is typically enabled via machine learning, but Kore.ai uses a dual-pronged approach which includes intent recognition and entity extraction.

Natural Language Training – The processes in which you refine a chatbot’s ability to understand and process NL requests, and test accordingly. It can be done by adding synonyms to the chatbot’s vocabulary via the Kore.ai Bot Builder, or training with complete utterances via machine learning. You can learn more about how to NL train a bot by watching [How To Build A Chatbot In 5 Minutes](#).

Neural networks – A computer system modeled on the human brain and nervous system.

Nodes – A node is where different points of a dialog or workflow intersect. The Kore.ai Dialog Builder has multiple node types so developers can steer the human-to-bot conversation in different directions, including intent nodes, service nodes, message nodes, confirmation nodes, webhook nodes, and JavaScript nodes. Learn more about how you can use nodes to build more complex and free flowing chatbot dialogs

Omni-channel (deployment) – The process of building one chatbot that is “channel agnostic” (meaning the bot can live in any channel), and deploying it to the communication channels of your choice.

---

# ***Project Description***

---

# **INTRODUCTION**

“What is AI & ML?”

AI (Artificial Intelligence)

What comes to mind when I ask you about Artificial Intelligence (AI)? Is it a case of robots taking over the world? Or something you might have seen in a science fiction film? Don't be concerned; it happens to all of us! Even when I first started learning about it, I had this thought. But as I dug deeper into AI, I realized that it is nothing like that, and yet it is so much more. If the twenty-first century is to be remembered for anything, it must be AI and the changes it has brought. Artificial Intelligence refers to the intelligence demonstrated by machines. Artificial intelligence has grown in popularity in today's world. It is the simulation of natural intelligence in machines that are programmed to learn and mimic human actions.

These machines can learn from experience and perform human-like tasks. As artificial intelligence (AI) technology advances, it will have a significant impact on our quality of life.

It is only natural that everyone today wants to connect with AI technology in some way, whether as an end user or as a developer. Humans created an intelligent entity. Capable of intelligently performing tasks without being explicitly instructed. Capable of rational and humane thought and action.

## **How does AI & ML work?**

While it's one thing to know what AI is, it's another to understand the underlying functions. Artificial intelligence operates by processing data through advanced algorithms. It combs large data sets with its algorithms, learning from the patterns or features in the data. There are many theories and subfields in AI systems including:

**Machine learning.** Machine learning uses neural networks to find hidden insights from data, without being programmed for what to look for or what to

conclude. Machine learning is a common way for programs to find patterns and increase their intelligence over time.

**Deep learning.** Deep learning utilizes huge neural networks with many layers, taking advantage of its size to process huge amounts of data with complex patterns. Deep learning is an element of machine learning, just with larger data sets and more layers.

**Cognitive computing.** Cognitive computing has a goal for a human-like interaction with machines. Think robots that can see and hear, and then respond as a human would.

**Computer vision.** In AI, computer vision utilizes pattern recognition and deep learning to understand a picture or video. This means the machine can look around and take pictures or videos in real time, and interpret the surroundings. The overall goal of AI is to make software that can learn about an input, and explain a result with its output. Artificial intelligence gives human-like interactions, but won't be replacing humans anytime soon.

## **Why is Python So Popular for AI and Machine Learning?**

**With each passing minute, AI and machine learning are grabbing more eyeballs than ever. Who'd have thought that there could exist a self-driven car or smartphones that forecast what weather it will be tomorrow! But today, all this is a reality.**How Does Home Automation Work?

Companies like Uber, Tesla, JP Morgan Chase, Apple, and other industry giants have accepted these technologies. With this, they've also befriended one programming language that is flexible, stable, with a variety of tools available: PYTHON.

Here are the top reasons that make Python so popular for AI and machine learning:

### **1. Rich library ecosystem**

A programming language library refers to a module that comes with a pre-written code that helps the user to use the same functionality to perform different actions. Python contains libraries that help in saving developer's time as they do not have to start from scratch.

List of some common libraries used for AI and machine learning:

- Pandas
- Scikit-learn
- Keras
- TensorFlow
- Caffe
- PyBrain

With the help of these libraries, AI and ML algorithms can be implemented more easily. These libraries are useful for data analysis, deep learning, machine learning, computer vision, and advanced computing. This helps in the faster development of the product as the developers can now resolve complex problems without rewriting code lines.

## 2. Flexibility

Python is a flexible language, which means that it can be used along with other programming languages to achieve the desired result. It offers an option to the developer to choose between OOPs or scripting. Also, it does not require recompilation of the source code, making it easier to view the results. Due to its flexibility, it gives the developer a safe environment and reduces the chances of mistakes.

## 3. Simple and Consistent

This programming language offers concise, readable codes. As complex algorithms stand behind AI and ML, the simplicity of the language helps in developing reliable systems. Now the entire focus is on solving an ML problem instead of worrying about the technical details of the language.

Another reason which makes Python so popular is that it is an easy-to-learn programming language. Due to its easier understandability by humans, it is easier to make models for machine learning. Furthermore, many coders say that Python is more intuitive than other programming languages. It is suitable for a collaborative implementation as and when multiple developers are involved. Being a general-purpose language, it allows you to build prototypes faster so that you can test your product for machine learning.

## 4. Platform Independent

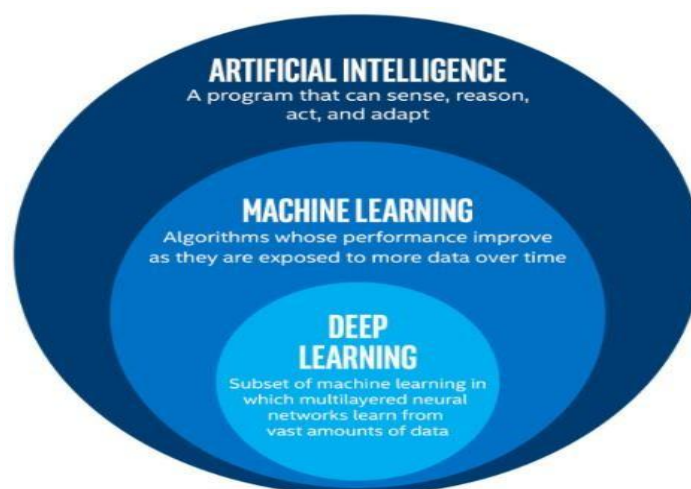


Platform independence of a programming language means that it can run on a variety of platforms and software architectures. The code has to be written once and it can be compiled and run on multiple platforms.

Python is easy to learn and use and scores high on versatility. It can run on any platform, be it Windows, MacOS, Linux, Unix, and more. If one wants to run the code of different platforms, packages like PyInstaller come in handy. Let's say a coder wants to shift from one platform to another, it is far easier with Python. This saves time and money for tests on multiple platforms. As a result, the overall process becomes more convenient.

What exactly is Deep Learning?

Deep Learning is a subset of Machine Learning, which on the other hand is a subset of Artificial Intelligence. Artificial Intelligence is a general term that refers to techniques that enable computers to mimic human behavior. Machine Learning represents a set of algorithms trained on data that make all of this possible.



AI. vs ML. vs DL.

Deep Learning, on the other hand, is just a type of Machine Learning, inspired by the structure of a human brain. Deep learning algorithms attempt to draw similar conclusions as humans would by continually analyzing data with a given logical structure. To achieve this, deep learning uses a multi-layered structure of algorithms called neural networks.

The design of the neural network is based on the structure of the human brain. Just as we use our brains to identify patterns and classify different types of information, neural networks can be taught to perform the same tasks on data.

The individual layers of neural networks can also be thought of as a sort of filter that works from gross to subtle, increasing the likelihood of detecting and outputting a correct result.

The human brain works similarly. Whenever we receive new information, the brain tries to compare it with known objects. The same concept is also used by deep neural networks.

Neural networks enable us to perform many tasks, such as clustering, classification or regression. With neural networks, we can group or sort unlabeled data according to similarities among the samples in this data. Or in the case of classification, we can train the network on a labeled dataset in order to classify the samples in this dataset into different categories.

Why is Deep Learning is Popular these Days?

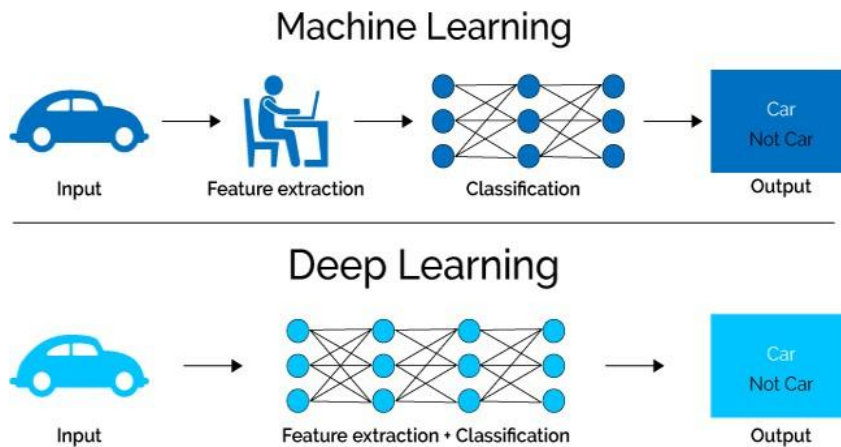
Why is deep learning and artificial neural networks so powerful and unique in today's industry? And above all, why are deep learning models more powerful than machine learning models? Let me explain it to you.

*The first advantage of deep learning over machine learning is the needlessness of the so-called feature extraction.*

Long before deep learning was used, traditional machine learning methods were mainly used. Such as Decision Trees, SVM, Naïve Bayes Classifier and Logistic Regression.

These algorithms are also called flat algorithms. Flat here means that these algorithms can not normally be applied directly to the raw data (such as .csv, images, text, etc.). We need a preprocessing step called Feature Extraction.

The result of Feature Extraction is a representation of the given raw data that can now be used by these classic machine learning algorithms to perform a task. For example, the classification of the data into several categories or classes.



## **Background**

While many would agree that chatbot has become a buzzword recently, but the concept has its existence from the time when people started developing ways to interact with computers. The first-ever chatbot was introduced even before the launch of personal computers. It was developed by MIT Artificial Intelligence Laboratory by Joseph Weizenbaum in 1966 and was named Eliza.

Eliza examined the keywords received as input and then triggered the output according to a defined set of rules.

This methodology of generating output is still used by a number of chatbots. The next chatbot was Parry, written by psychiatrist Kenneth Colby at the Stanford University in an attempt to simulate a person with paranoid schizophrenia.

Then came the A.L.I.C.E, which was developed in 1995 by Richard Wallace. While A.L.I.C.E won the Loebner prize thrice, it failed to pass the Turing test. A Turing test examines whether or not a machine is able to think intelligently like humans. The next chatbot was Parry, developed by psychiatrist Kenneth Colby at the Stanford University. It was an attempt to simulate a person with paranoid schizophrenia. Then came the A.L.I.C.E, which was developed in 1995 by Richard Wallace. While A.L.I.C.E won the Loebner prize thrice, it failed to pass the Turing test. A Turing test examines whether or not a machine

can think intelligently like humans. After this, various virtual assistants were launched. Siri by Apple was the first one to introduce conversational assistants. The concepts gained popularity and soon after Google launched their Google Assistant for Android. Following the lead, Microsoft brought Cortana into existence. Taking this a step forward, smart speakers were introduced which made voice conversation possible between humans and the bots. Amazon Alexa & Google Home represents a different category of conversational UI.

**Chatbots in the Recent Past** Chatbots have an edge in the recent past because of two prominent reasons: Messaging Platforms Technology

In 2015, Telegram opened its app for developers to build and deploy bots. Following this, Facebook opened its messaging platform, the Facebook Messenger for bot development and it garnered the attention of users as well as developers.

Now, almost every popular platform including Whatsapp, Slack, We Chat, IMO offers the facility to build a chatbot. Over time, there has been an improvement in the abilities of a chatbot and technology has been a real contributor to it. Artificial intelligence and its technologies have been helping developers in taking chatbots and their functionality to the next level. Today, chatbots are performing financial transactions, processing orders, resolving customer queries, and a lot more. We will be talking about it in detail in the upcoming segment.

### **Chatbots: Then and Now**

The journey of chatbots from 1960 till date has been interesting. Chatbots have evolved in functionality, interface, and their significance to the business world cannot be ignored. Here is a glimpse of how progressive chatbots have been so far. Tasks and Roles Performed: Chatbots have been a great conversational agent for businesses.

For a long time, they were an efficient replacement of a customer service provider who is there for the customers 24/7. However, with chatbots being able to take new roles and responsibilities, they have become an alternative to mobile apps. From simple conversations to complex transactions, chatbots are made to do almost everything.

They are now used by businesses, irrespective of the industry and are used for placing orders, checking the status of the order, book flight tickets, make financial transactions, handle complex queries of the clients/customers, improve marketing campaigns, and a lot more.

In short, chatbots are new apps for businesses. Enhancement in UI Elements: As the chatbots evolve with better abilities, there is a shift in their UI elements as well. They have a Conversational User Interface (CUI) which brings a change to the way human-computer interaction occurs.

CUI uses Natural Language Processing (NLP) that enables computers to understand, analyze, and create meaning from human language. NLP also takes up and process unstructured inputs (such as questions with spelling mistakes), helps in sentiment analysis, and more.

Depending upon the functionalities performed, several graphic elements are added to the chatbot. Along with text messages, there are carousels, buttons, quick reply, web view, group chat, option to share audio, video, document file, image, GIF, and more. Technology Stack for Development: Chatbots are made to do a lot of things. Depending upon what a chatbot has to do, different technologies are adopted by the developers to build a chatbot. There are tools (like ChatFuel, API.ai, Botsify) that help to build a chatbot with less technical expertise. Then, there are technologies like Machine Learning, Natural Language Processing, RPA that can take chatbot abilities to the next level.

Such abilities can be integrated by opting for custom chatbot development. Modes of Communication: Chatbots have evolved to enable text and voice communication possible. Chatbots working in collaboration with virtual assistants or smart speakers allow users to give input and receive output in a voice format. Here is a perfect example of it- Mark Zukerberg's JARVIS.

## **Summary**

In this project, we have introduced a chatbot that is able to interact with users. This chatbot can answer queries in the textual user input. For this purpose, AIML with program-o has been used. The chatbot can answer only those questions which he has the answer in its AIML dataset. So, to increase the knowledge of the chatbot, we can add the APIs of Wikipedia, Weather Forecasting Department, Sports, News, Government and a lot more.

The next step towards building chatbots involves helping people to facilitate their work and interact with computers using natural language or using their set of rules.

As future work, we can make a chatbot that is based on AIML and LSA. This technology will enable a client to interact with a chatbot in a more natural fashion. We can enhance the discussion by including and changing patterns and

templates for general client queries using AIML and the right response are given more often than LSA.

## **System Design**

This section explains in details how the project was approached. All hardware and software components that will be used will be explained in this section of the report. Problems encountered and solutions to these problems will not be mentioned in this section of the report.

### **3.1 Project Overview**

A chatbot is an artificial intelligence (AI) software that can simulate a conversation (or a chat) with a user in natural language through messaging applications, websites, mobile apps or through the telephone. A chatbot is often described as one of the most advanced and promising expressions of interaction between humans and machines. However, from a technological point of view, a chatbot only represents the natural evolution of a Question Answering system leveraging Natural Language Processing (NLP). Formulating responses to questions in natural language is one of the most typical Examples of Natural Language Processing applied in various enterprises' end-use applications.

## **DESIGN OF CHATBOT**

A Chatbot refers to a chatting robot. It is a communication simulating computer program. It is all about the conversation with the user. The conversation with a Chatbot is very simple. It answers to the questions asked by the user. During designing a Chatbot, how does the Chatbot speak to the user? And how will be the conversation with the user and the Chatbot is very important . The design of a Chatbot is represented using diagram as follows:

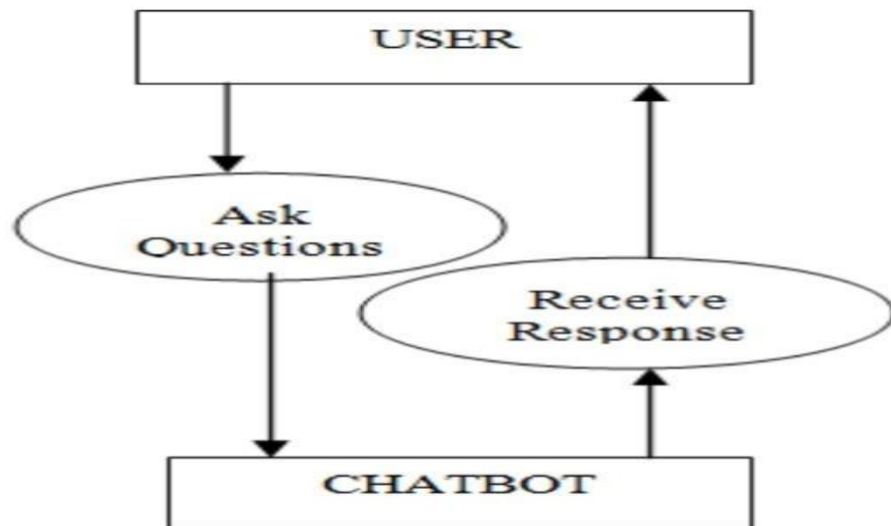


Fig.1: Use Case Diagram of Chatbot Design.

The following facts are kept in mind during designing a Chatbot :

**A. Selection of OS**

Windows is used for this project because it is user friendly. It is also robust.

**B. Selection of Software**

Eclipse software is used for programming in java. Because it contains basic workspace and it is mostly used for java applications.

**C. Creating a Chatbot**

For creating a Chatbot, a program has to be written. Java programming language is used for programming. The Chatbot is created in such a way to help the user, improve the communication and amuse the user.

**D. Creating a Chat**

The chat is created using a pattern that is known to the user and could be easy to understand. Chat dialog box show up to create conversation. This dialog box is created using java applets.

**E. Pattern Matching**

It is a technique of artificial intelligence used in the design of a Chatbot. The input is matched with the inputs saved in the database and corresponding response is returned.

#### F. Simple

The design of a Chatbot is very simple. It just answers to the questions asked by the user, if the question is found in the database.

#### G. Conversational and Entertaining

The Chatbot responses are a way known to the user. The conversation follows a Basic English language and interacts in an easy to read manner. The conversation between the user and the Bot is entertaining. It is like talking to other person.

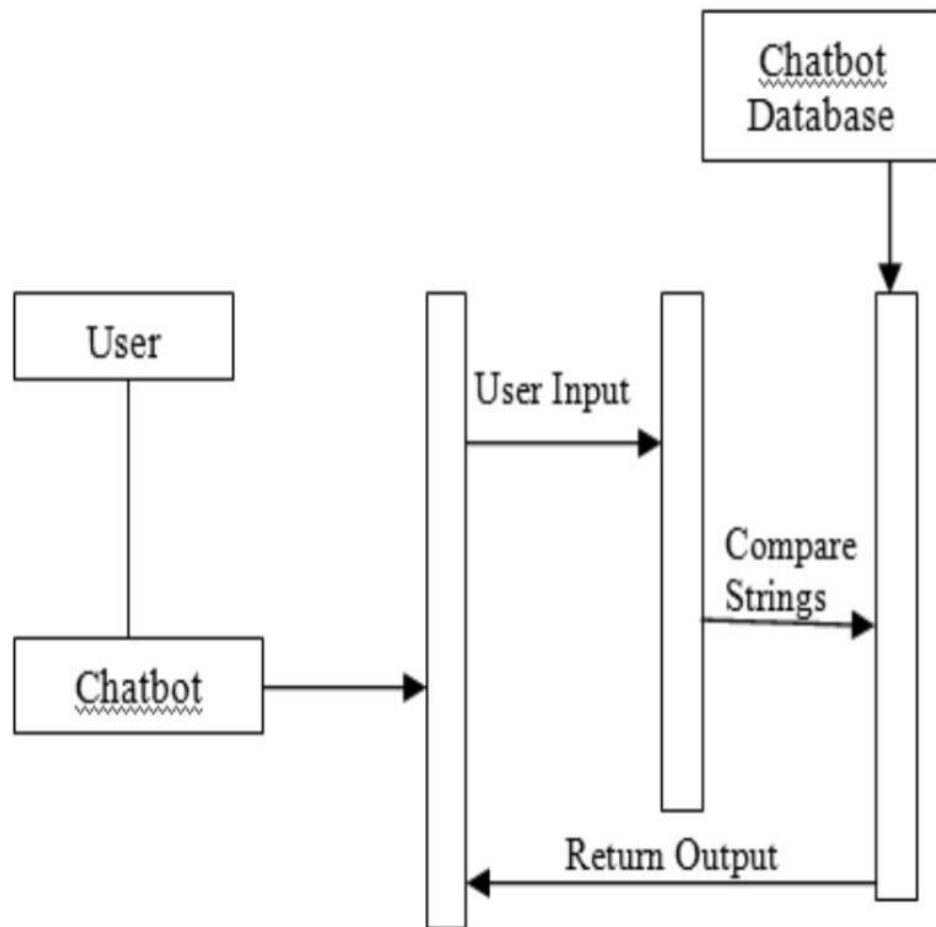


Fig. 2: Sequence Diagram Representing Design of the Chatbot.

### IMPLEMENTATION PROCESS

Chatbot is a computer application which uses artificial intelligence to mimic human conversation. It helps the user by answering the questions asked by them. The program is implemented using Java programming language.



Particularly Java applets are used. Applets are used because it is easy to create the dialog box required for the conversation between the user and the bot. Detailed implementation is given below:

#### A. Fundamental Design Techniques and Approaches

##### ☐ Creating the dialog box

All the packages required for creating the dialog box are imported. The size of the dialog box and text area inside the dialog box is given. Vertical scrollbar is used so that the screen is scrolled as the conversation goes on. Horizontal scrollbar is never used because the size of the dialog box is fixed.

##### ☐ Creating a database

Two dimensional string arrays are applied to build a database. Rows in the array are used for request and response. All the even rows contain the request or questions and all the odd rows contain the response or answers.

Columns in the array are applied to save different types of questions that could be asked by the user and responses that a Chatbot can answer. There is one row in the array which contains default responses which is used when the matching question is not found in the array.

#### B. Modules Description

The description of the modules used in the implementation is given below :

##### ☐ Chatbot()

In this function, all the variables used for creating the dialog box are added. Default close operation is set to EXIT\_ON\_CLOSE so that the dialog box closes on exit. Required background colour is set using inbuilt set Background () function.

##### ☐ Random()

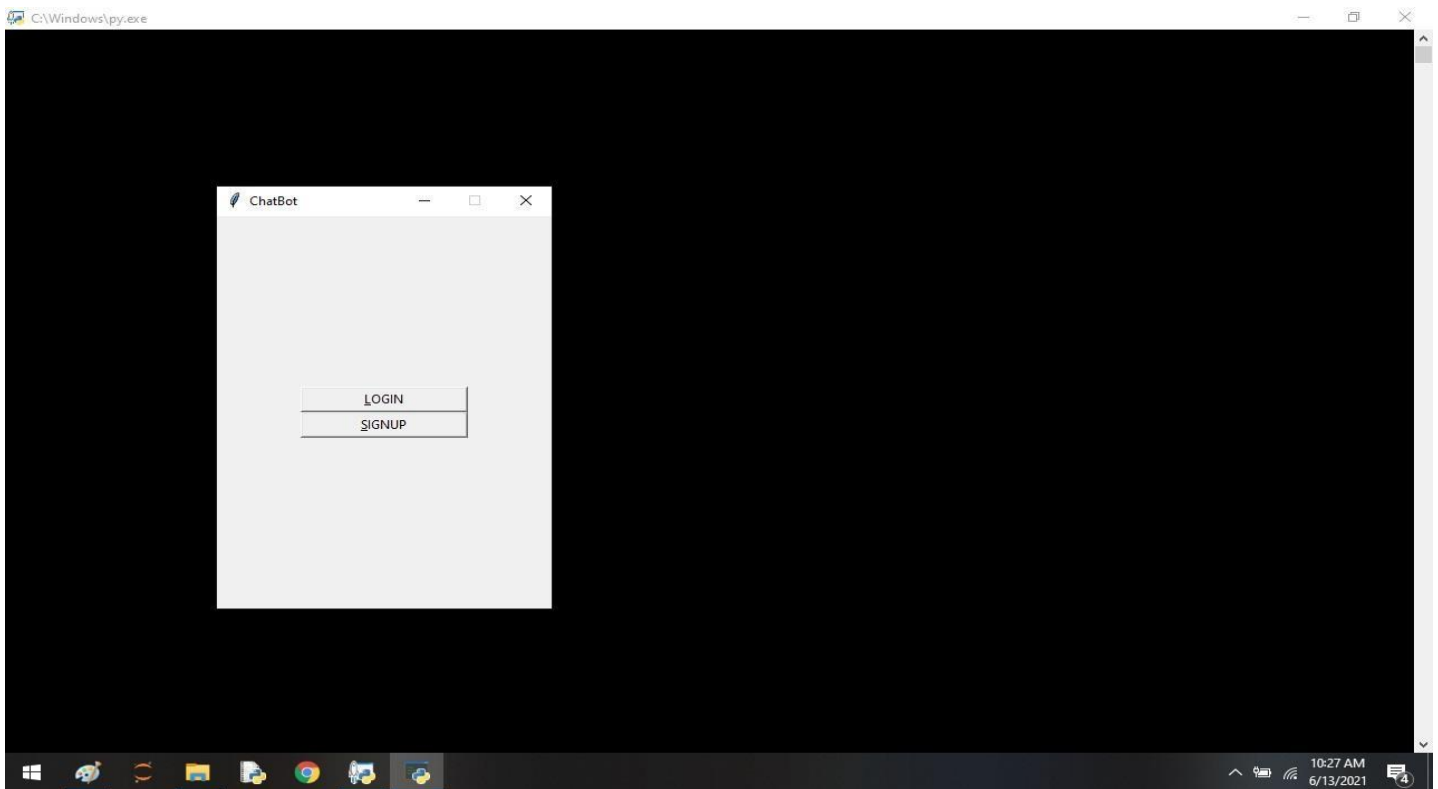
The input from the user is taken using get Text () function .All the punctuation marks in the users input are remove during trim () function. The uppercase letters are converted to lower case. A variable called response is used to hold a byte value and it is set to 0.

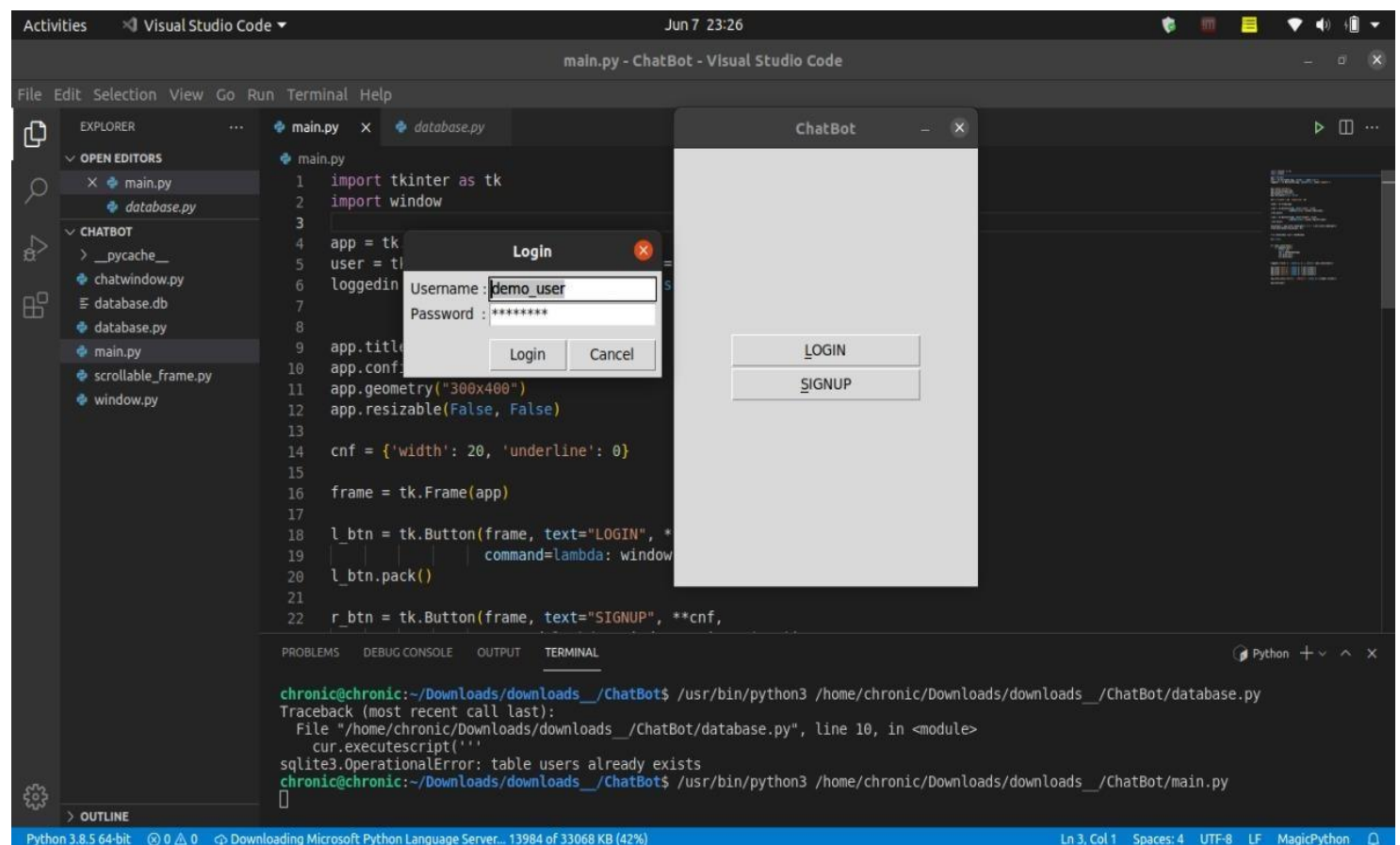
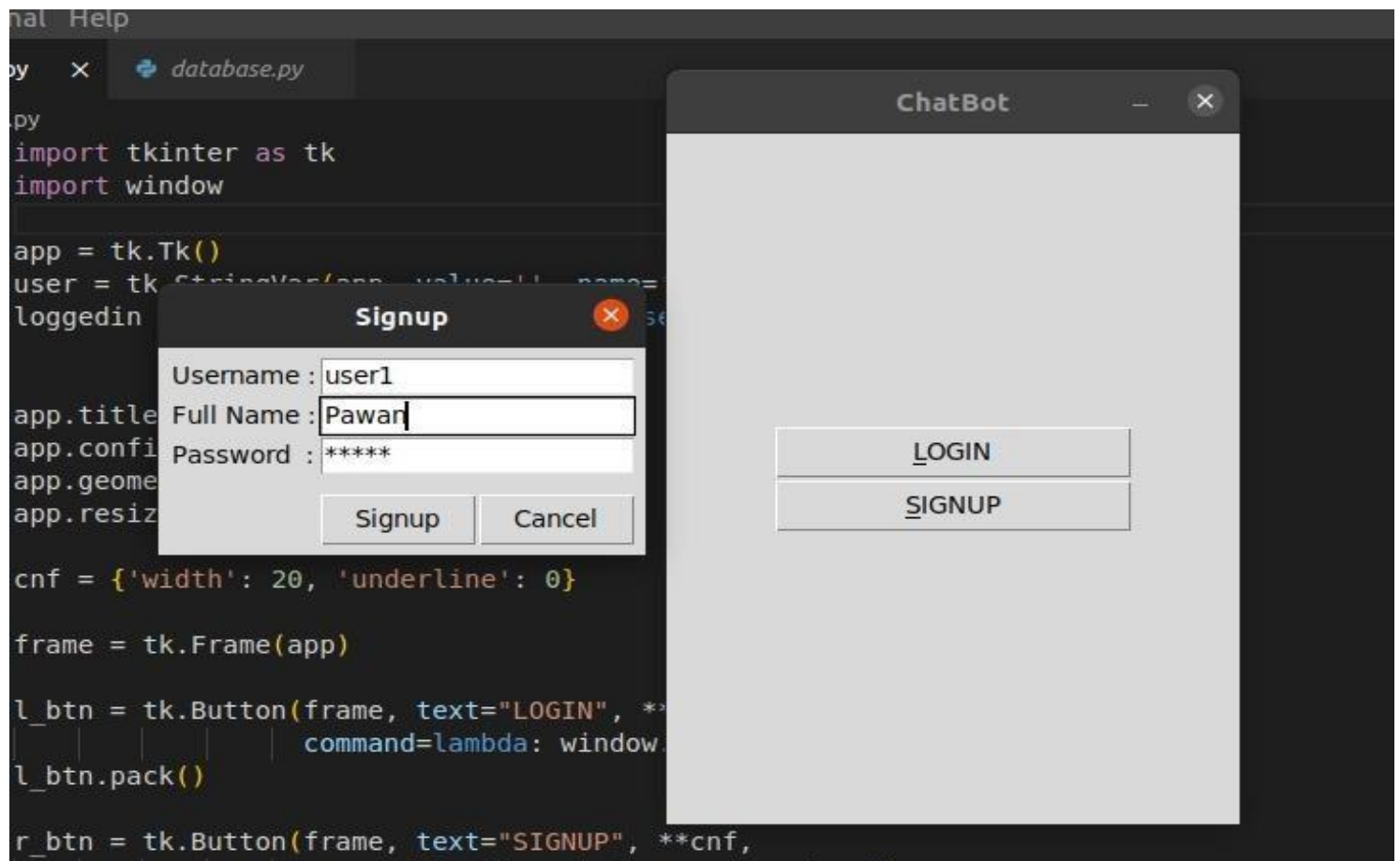
While response is 0, the match for the input is found in the database and it is returned as a response which is displayed in the text area. If the response is 1, then the match for the input is not found in the database. In this case, a default response is returned. Random () function is used to choose the response saved in the database.

☐ AddText()All the texts or strings used in input and output are added to the text area in the dialog box.

## ❑ InArray()

This is used as a pattern matching function. A variable match is used to hold a Boolean value and it is set to false. If the match for the users input is found in the database, true is returned else false is returned as a result. This value is returned to key Pressed() function and the result is displayed in the dialog box.







#### **4. Experimental Results and Discussion**

All of our participants reported that they had interacted with chatbots before, but had very little knowledge about how they worked. They found the chatbot to be nice to interact with and enjoyed that it had a friendly and casual tone. One of the participants said that she did not want a chatbot that felt too 'human-like', and that the prototype did not feel 'human-like' at all.

This became clear when the same error message appears several times during the test. They found it hard to get the right answer but when they did they were very satisfied with the answers. "It was a good answer when I finally got the right one..". It was pointed out that the chatbot was not a smart chatbot, but that it provided the most necessary information sparing them from precious time spent on 'Google'.

They also reported that they trusted the answers they got, and they all pointed out that it was good that the chatbot provided a source along with the information it gave. The gif and the pictures were also very popular among the participants, they said that this made the chatbot fun to interact with. One of the participants said that: “It’s casual, and extra fun with GIF’s”. One of the participants also stated: “I liked that the chatbot was casual and cute. I don’t want a formal and boring chatbot, then I could have tried to find it on the university's web-pages.” It was also pointed out that it was preferably that the chatbot could provide diverse information, “Usually, the information is so spread that you don’t know where to look”.

When testing the last prototype we got findings suggesting that the participants did not have a problem with getting information from a chatbot instead of a human. The information that they got was not seen as less trustworthy, this could be supported by the fact that the chatbot provided a source for the information it gave. It has been interesting to investigate how the participants interacted with the chatbot and how they reported on it afterwards.

Our findings have some indicators leading towards that a chatbot could be a good alternative for acting as a helpful friend for freshmans at a new school. Still we have to stress the fact that the chatbot was not very intelligent and that the evaluators had to adjust their language to match the chatbots.



## **Result from first Testing**

The first participant enjoyed talking to the bot, but stressed the fact that you had to “talk like “a dummy” for it to understand what you were asking. The participant pointed out that this really would have come in handy in his first weeks at the university, as he didn’t always know who to ask - especially if he was in a hurry.

He pointed out that the prototype needs to get more features like tell you exam dates, or “if life-hacks, like get your coffee before all of the students have their break”. The second participant was a bit frustrated that the chatbot wasn’t flexible enough . “I don’t like having to guess what questions to ask”. He would liked more instructions to know how to get more out of the chatbot. The third participant had also problems with understanding what the chatbot could do. When given a hint for what the chatbot could do, the chatbot did not function properly.

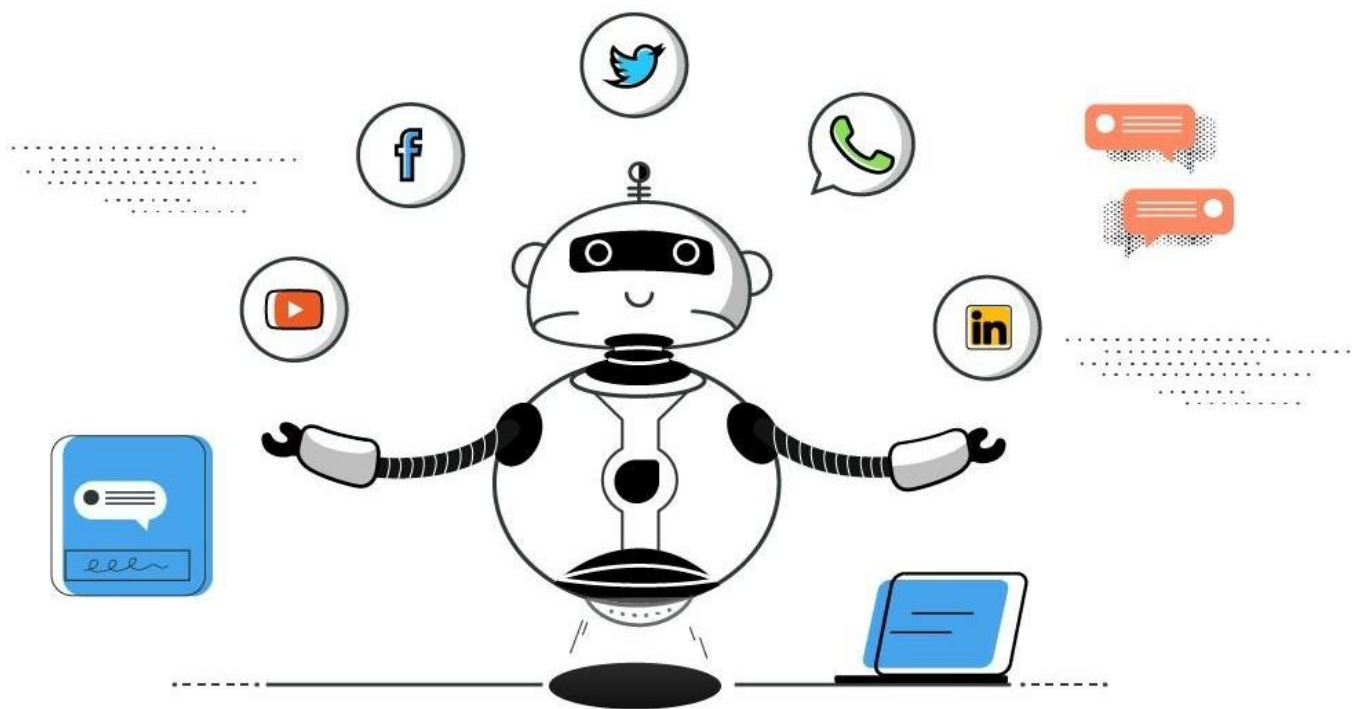
Here we tried to restart the system and then the chatbot displayed it’s welcome message what it could do. Afterwards it was more clear what the participant could ask it, but the chatbot did not always give the response that the participant wanted.



## **Future Scope**

Many experts expect chatbots to continue growing in popularity. In the future, AI and ML will continue to evolve, offer new capabilities to chatbots and introduce new levels of text and voice-enabled user experiences that will transform CX. These improvements may also affect data collection and offer deeper customer insights that lead to predictive buyer behaviors.

Voice services have also become common and necessary parts of the IT ecosystem. Many developers place an increased focus on developing voice-based chatbots that can act as conversational agents, understand numerous languages and respond in those same languages



## **Summary**

In this project, we have introduced a chatbot that is able to interact with users. This chatbot can answer queries in the textual user input. For this purpose, AIML with program-o has been used. The chatbot can answer only those questions which he has the answer in its AIML dataset. So, to increase the knowledge of the chatbot, we can add the APIs of Wikipedia, Weather Forecasting Department, Sports, News, Government and a lot more. The next step towards building chatbots involves helping people to facilitate their work and interact with computers using natural language or using their set of rules. As future work, we can make a chatbot that is based on AIML and LSA. This technology will enable a client to interact with a chatbot in a more natural fashion. We can enhance the discussion by including and changing patterns and templates for general client queries using AIML and the right response are given more often than LSA.

## **Conclusion**

In this project, we have introduced a chatbot that is able to interact with users. This chatbot can answer queries in the textual user input. For this purpose, AIML with program-o has been used. The chatbot can answer only those questions which he has the answer in its AIML dataset. So, to increase the knowledge of the chatbot, we can add the APIs of Wikipedia, Weather Forecasting Department, Sports, News, Government and a lot more. In such cases, the user will be able to talk and interact with the chatbot in any kind of domain. Using APIs like Weather, Sports, News and Government Services, the chatbot will be able to answer the questions outside of its dataset and which are currently happening in the real world.

The next step towards building chatbots involves helping people to facilitate their work and interact with computers using natural language or using their set of rules. Future Such chatbots, backed by machine-learning technology, will be able to remember past conversations and learn from them to answer new ones. The challenge would be conversing with the various multiple bot users and multiple users.



As future work, we can make a chatbot that is based on AIML and LSA. This technology will enable a client to interact with a chatbot in a more natural fashion. We can enhance the discussion by including and changing patterns and templates for general client queries using AIML and the right response are given more often than LSA.

## **BIBLIOGRAPHY**

Bayan Abu Shawar and Eric Atwell, 2007 “Chatbots: Are they Really Useful?”

[2] LDV Forum - GLDV Journal for Computational Linguistics and Language Technology.

[3] [http://www.ldv-forum.org/2007\\_Heft1/Bayan\\_AbuShawar\\_and\\_Eric\\_Atwell.pdf](http://www.ldv-forum.org/2007_Heft1/Bayan_AbuShawar_and_Eric_Atwell.pdf)

[4] Bringing chatbots into education: Towards natural language negotiation of open learner models. Know.-Based Syst. 20, 2 (Mar. 2007), 177-185.

[5] Intelligent Tutoring Systems: Prospects for Guided Practice and Efficient Learning. Whitepaper for the Army's Science of Learning Workshop, Hampton, VA. Aug 1-3, 2006.

[6] <http://en.wikipedia.org/wiki/Chatterbot>

[7] ALICE. 2002. A.L.I.C.E AI Foundation, <http://www.alicebot.org/>

[8] Kumar, M Naveen, PC Linga Chandar, A Venkatesh Prasad, and K Sumangali (2016). “Android based educational Chatbot for visually impaired people”. In:

International Conference on Computational Intelligence and Computing Research (ICCIC), 2016 IEEE. IEEE, pp. 1–4.

[9] Ranoliya, Bhavika R, Nidhi Raghuwanshi, and Sanjay Singh (2017). “Chatbot for University Related FAQs”. In: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI). Udupi, pp. 1525–1530.

] Mikic, Fernando A, Juan C Burguillo, Martín Llamas, Daniel A Rodríguez, and Eduardo Rodríguez (2009).

“Charlie: An aiml-based chatterbot which works as an interface among ines and humans”. In: EAEEIE Annual Conference, 2009. IEEE.

[11] Deryugina, OV (2010). “Chatterbots”. In: Scientific and Technical Information Processing 37.2, pp. 143–147.

[12] Lue Lin, Luis Fdo. D’Haro, and Rafael Banchs. A Webbased Platform for Collection of Human Chatbot Interactions, Paper accepted in HAI 2016 to appear in Oct. 2016.

[13] Wallace, Richard S. "The anatomy of ALICE." Parsing the Turing Test. Springer Netherlands, 2009. 181-210.

[14] Steven Bird, Ewan Klein, and Edward Loper – Natural Language Processing with Python, Published by: O’Reilly Media, Inc., June 2009

[15] Weizenbaum, Joseph (1966). “ELIZA—a computer program for the study of natural language communication between man and machine”. In: Communications of the ACM 9.1, pp. 36–45.

[16] Emanuela Haller and Traian Rebedea, “Designing a Chatbot that Simulates an Historical Figure”, IEEE Conference Publications, July 2013.

[17] Maria das Graças Bruno Marietto, Rafael Varago de Aguiar, Gislene de Oliveira Barbosa, Wagner Tanaka

Botelho, Edson Pimentel, Robson dos Santos França, and Vera Lúcia da Silva, “ARTIFICIAL INTELLIGENCE MARKUP LANGUAGE: A BRIEF TUTORIAL,”

International Journal of Computer science and engineering Survey (IJCSES), July 2013.

[18] N. Thomas, “An e-business chatbot using aiml and lsa,” in Advances in Computing, Communications and Informatics (ICACCI), 2016 International Conference on. IEEE, 2016, pp. 2740–2742