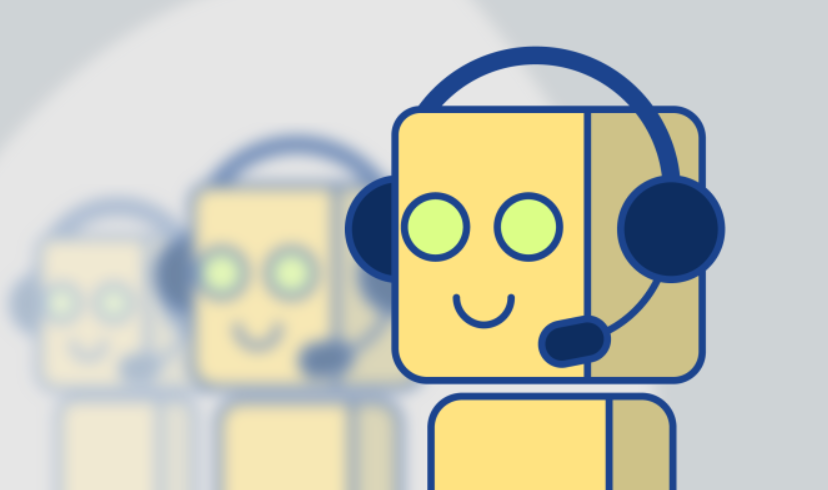
Ecl Chatbot

ECLBot



Suresh Kumar M

Sureshkumar.M@eyecareleaders.com

Agenda

[What is a ChatBot? 2](#_Toc517357569)

[Types of ChatBot? 3](#_Toc517357570)

[Why we need of ChatBot? 4](#_Toc517357571)

[How ChatBot helps ECL? 5](#_Toc517357572)

[ECLBot Architecture? 6](#_Toc517357573)

[Technologies Stack 6](#_Toc517357574)

# What is a ChatBot?

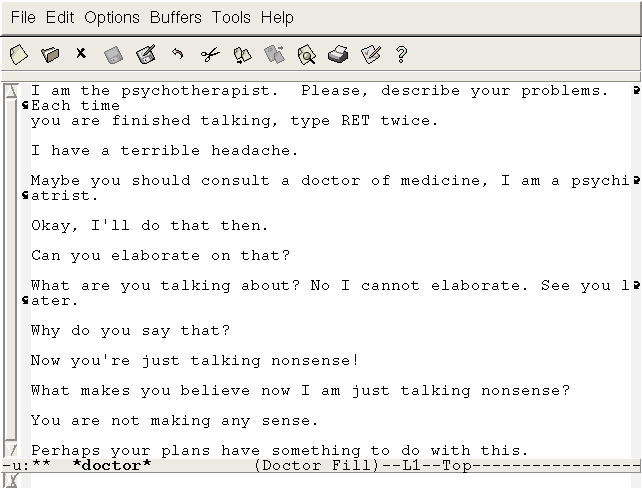
“A computer program designed to simulate conversation with human users, especially over the Internet.”

It is an assistant that communicates with us through text messages, a virtual companion that integrates into websites, applications or instant messengers and helps entrepreneurs to get closer to customers. Such a bot is an automated system of communication with users.



*Quick Fact:*

***ELIZA****is an early*[*natural language processing*](https://en.wikipedia.org/wiki/Natural_language_processing)[*computer program*](https://en.wikipedia.org/wiki/Computer_program)*created from 1964 to 1966*[*[1]*](https://en.wikipedia.org/wiki/ELIZA#cite_note-turing-1)*at the*[*MIT Artificial Intelligence Laboratory*](https://en.wikipedia.org/wiki/MIT_Computer_Science_and_Artificial_Intelligence_Laboratory)*by*[*Joseph Weizenbaum*](https://en.wikipedia.org/wiki/Joseph_Weizenbaum)*.*



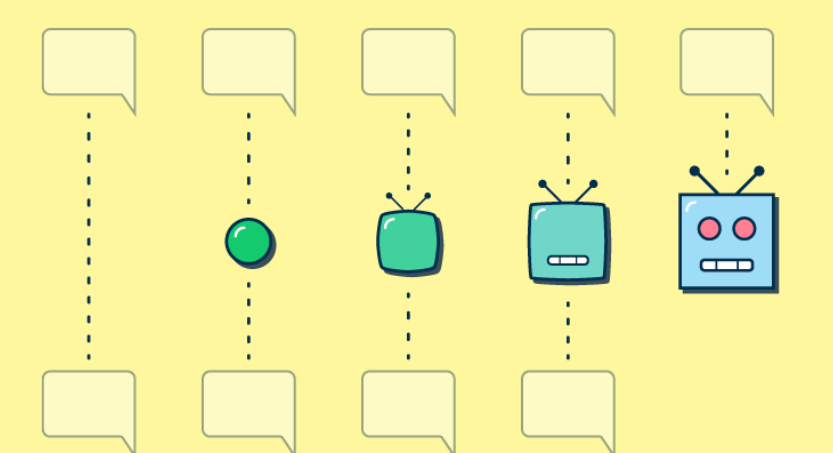
# Types of ChatBot?

Depending on how the specific bots were programmed, we can divide them into two large groups:

* Working according to pre-prepared commands (simple ChatBot).
* Trained (smart or advanced ChatBot).

**Simple ChatBot’s** work based on pre-written keywords that they understand. Each of these commands must be written by the developer separately using regular expressions or other forms of string analysis or through any programming logic. If the user has asked a question without using a single keyword, the robot cannot understand it and, as a rule, responds with messages like “sorry, I did not understand”.

**Smart ChatBot’s** rely on [artificial intelligence](https://anadea.info/blog/artificial-intelligence-pandoras-box-or-the-holy-grail) when they communicate with users. Instead of pre-prepared answers, the robot uses NLP algorithms and responds with adequate suggestions on the topic. In addition, all the words said by the customers are recorded for later processing.



# Why we need of ChatBot?

1. **TO INCREASE YOUR OPERATIONS**

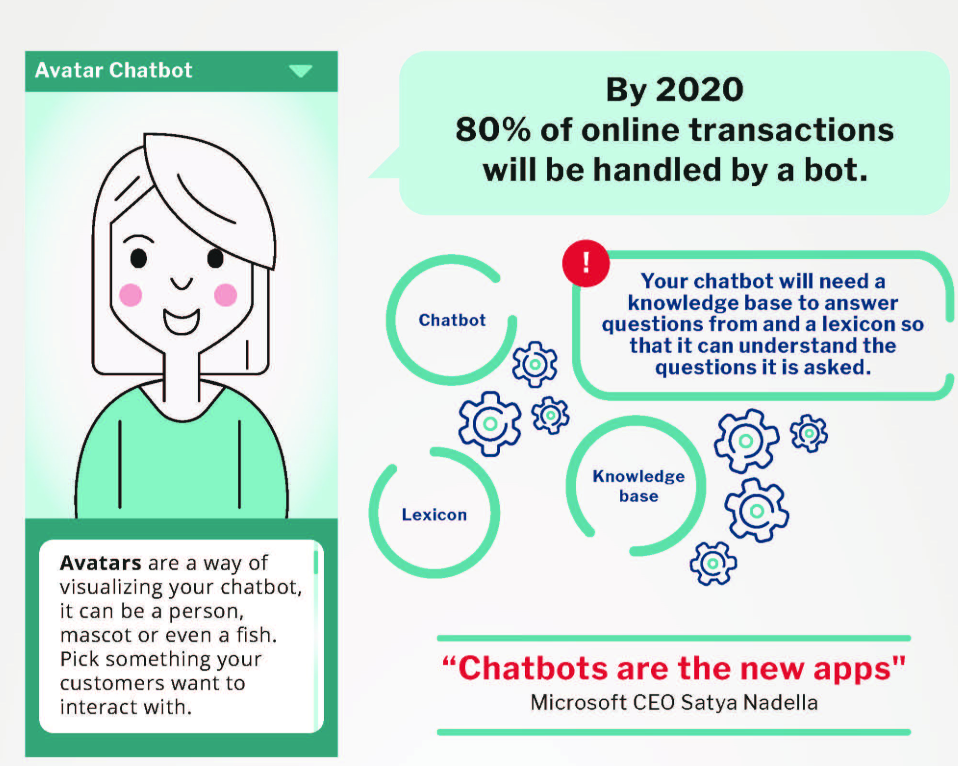
ChatBot’s do not have the same limitations that human agents have. A person can only have two or three conversations at a time, while a ChatBot has no such limit.

1. **GET MANY QUESTIONS FROM CUSTOMERS**

If your company gets a lot of questions, ChatBot’s can take over from the work of the customer support team. By acting as your first contact point, the ChatBot scans the customers' phones and redirects them to human sales agents when necessary.

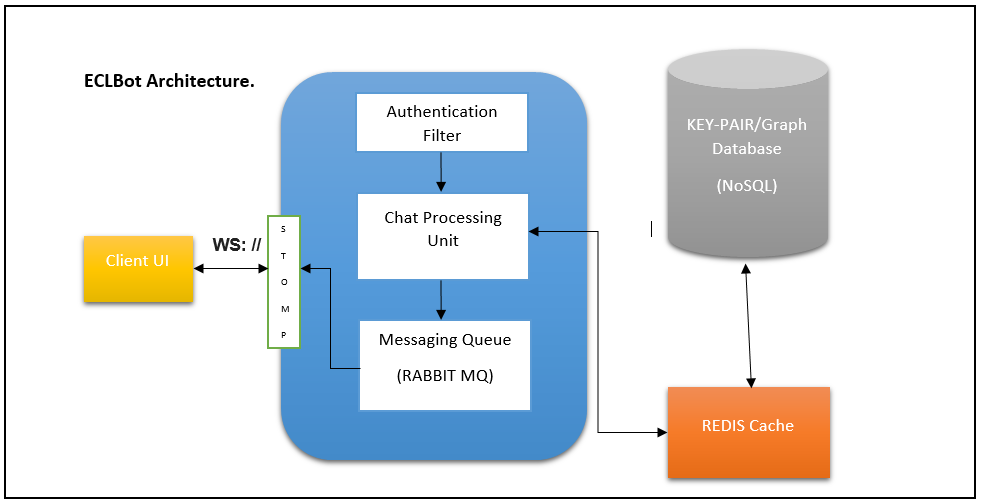
1. **3. YOU HAVE A SERIES OF ALMOST IDENTICAL PRODUCTS OR SERVICES**

If you sell similar goods or services, your customers will need help choosing the right product. Customers typically ask for advice when buying expensive items such as mobile phones, camera accessories, etc. ChatBot’s can assist customers in purchasing the right product or services.



# How ChatBot helps ECL?

# ECLBot Architecture?



This is a rough high level architecture of ECLBot (Security and Redis part yet to be finalized)

# Technologies Stack

* **WebSockets** enable bi-directional communication between the server and the client. Any time you want to push data to a client, web sockets are what you want to use.

WebSockets enable the server and client to send messages to each other at any time, after a connection is established, without an explicit request by one or the other. This is in contrast to HTTP, which is traditionally associated with the challenge-response principle — where to get data one has to explicitly request it. In more technical terms, WebSockets enable a [full-duplex connection](https://en.wikipedia.org/wiki/Duplex_%28telecommunications%29#Full_duplex) between the client and the server.

Especially for the applications like Chat, Stock Exchanges etc., where we have continuous data exchange between client-server, Web Sockets are efficient.

* **STOMP** is the Simple (or Streaming) Text Orientated Messaging Protocol.

STOMP provides an interoperable wire format so that STOMP clients can communicate with any STOMP message broker to provide easy and widespread messaging interoperability among many languages, platforms and brokers.

* **Spring Boot** aims to make it easy to create Spring-powered, production-grade applications and services with minimum fuss.

Spring Boot comes with implementation wrapper classes for various open sources like Messaging brokers (Rabbit MQ, Apache Kafka), Redis Cache, Netflix OSS and Various cloud technologies which helps developers to create Micro service architectures seamlessly.

* **Redis** is an open source (BSD licensed), in-memory data structure store, used as a database, cache and message broker.

Fewer database accesses reducing database traffic.

You can run **atomic operations** like [appending to a string](https://redis.io/commands/append).