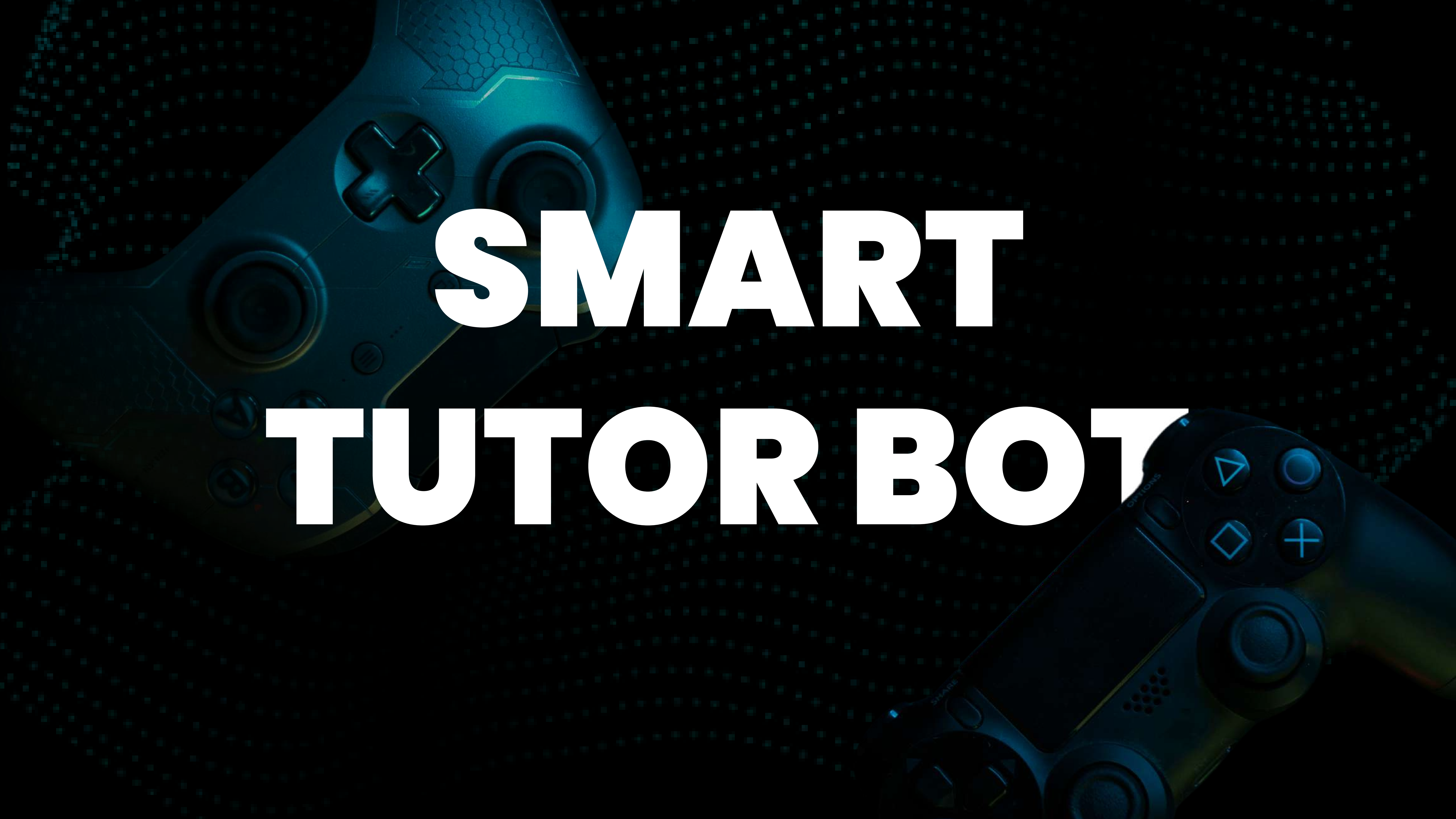


SMART TUTOR BOT

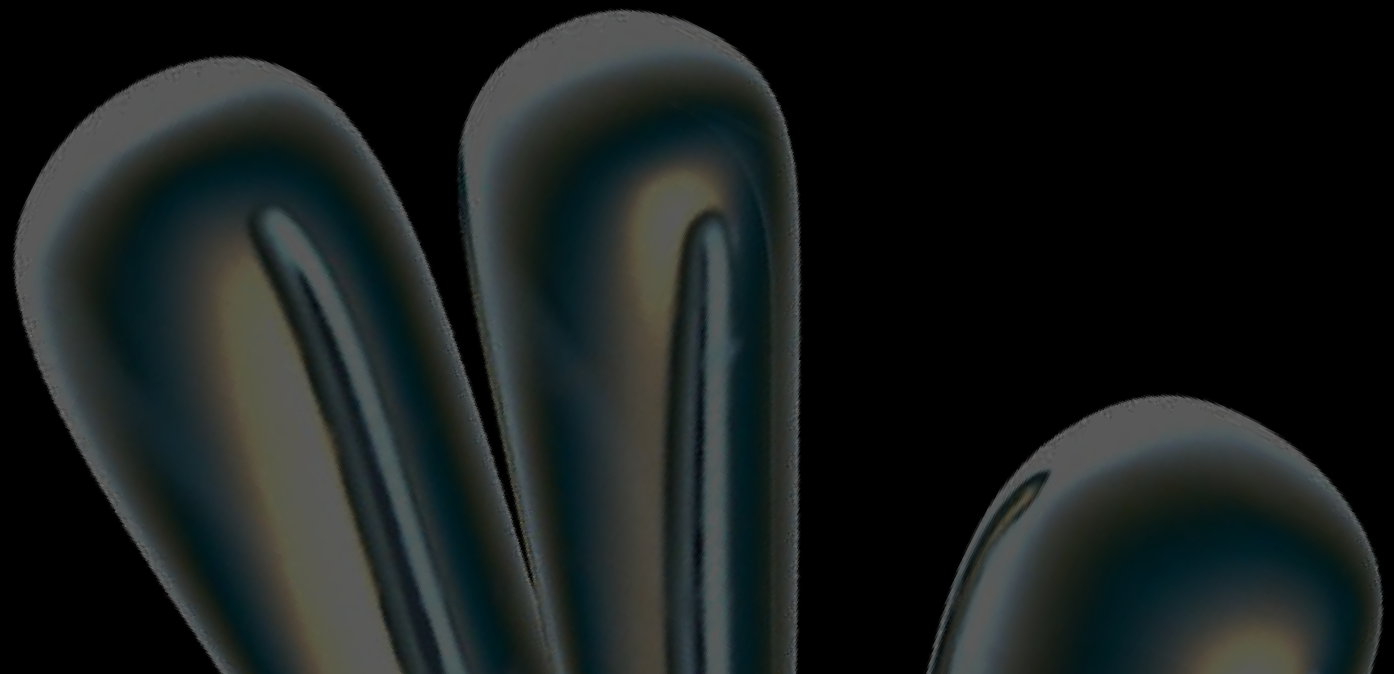
TOPICS WE HAVE COVERED

01.

Introduction

02.

Types of chatbots



03.

History of chatbot

04.

Training a tutor chat be like

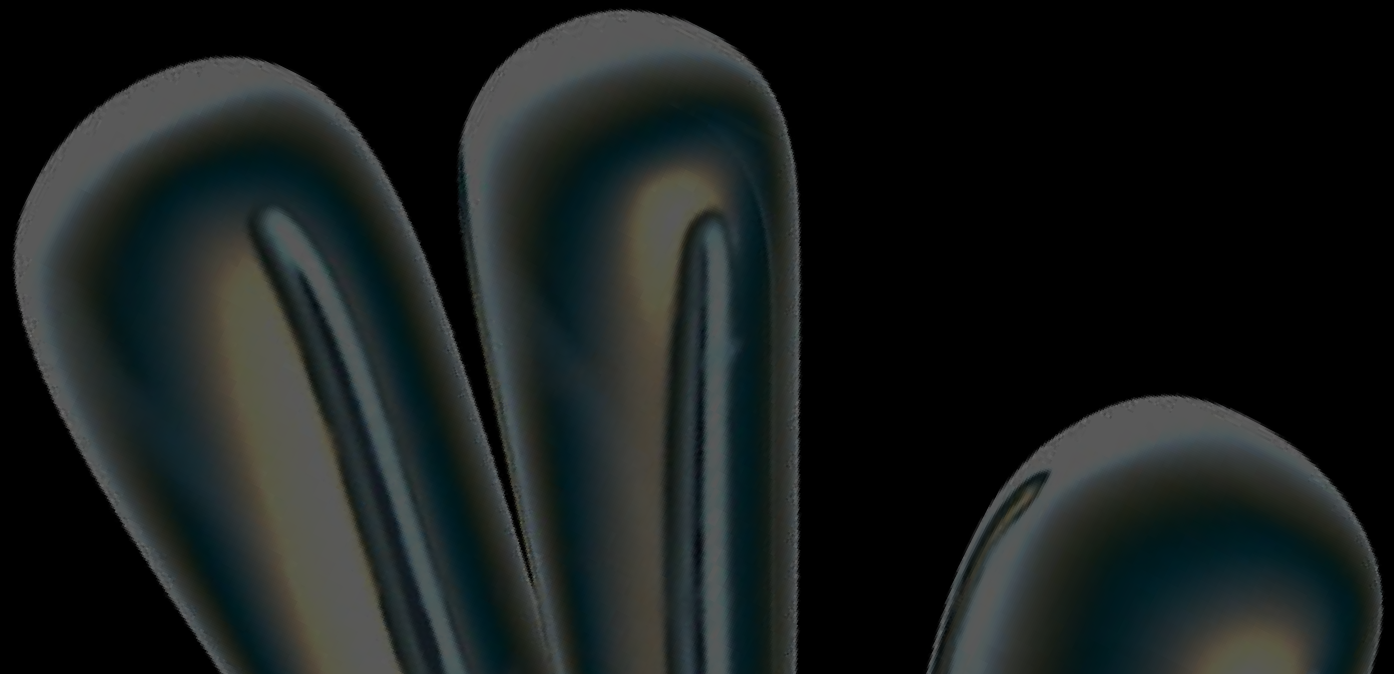
TOPICS WE HAVE COVERED

05.

Integration

06.

Responsiveness



07.

Ensuring the accuracy

08.

Calculating its performance

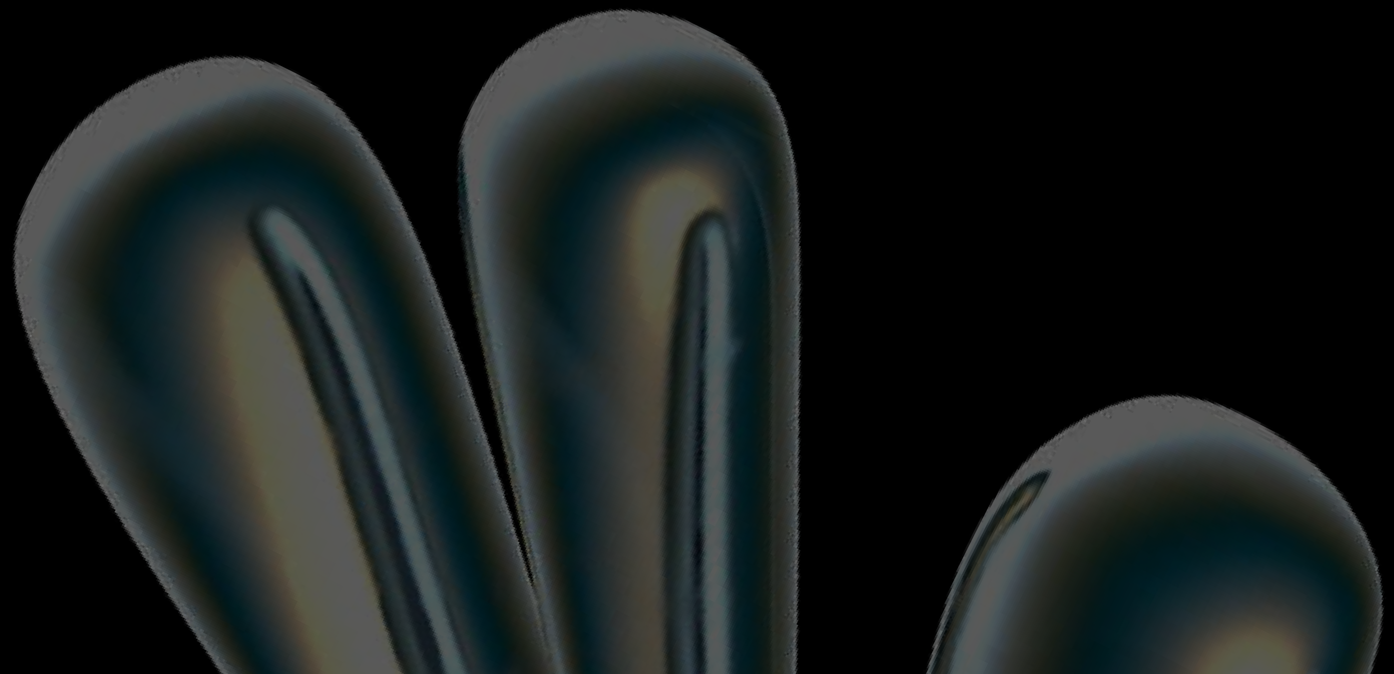
TOPICS WE HAVE COVERED

05.

Integration

06.

Responsiveness



07.

Ensuring the accuracy

08.

Calculating its performance

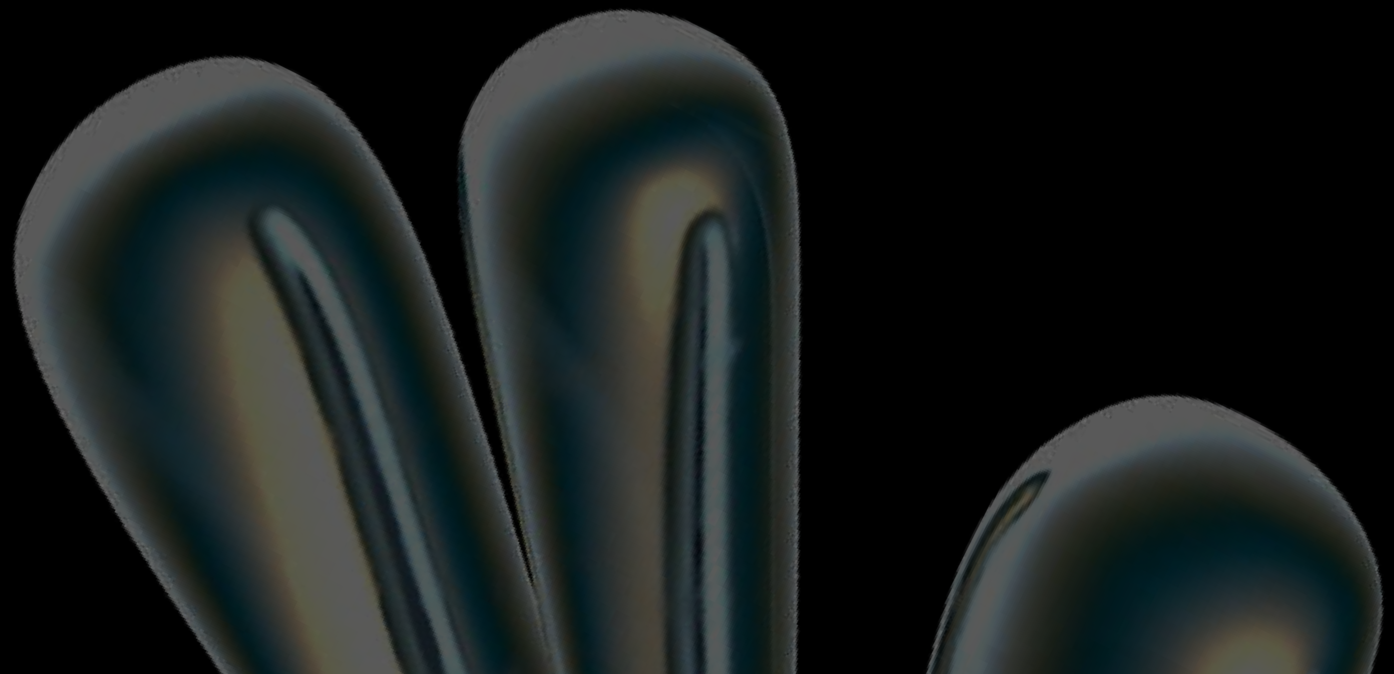
TOPICS WE HAVE COVERED

09.

Conclusion

10.

Future Enhancement



INTRODUCTION

What's So interesting in developing a chat bot and is that mandatory?

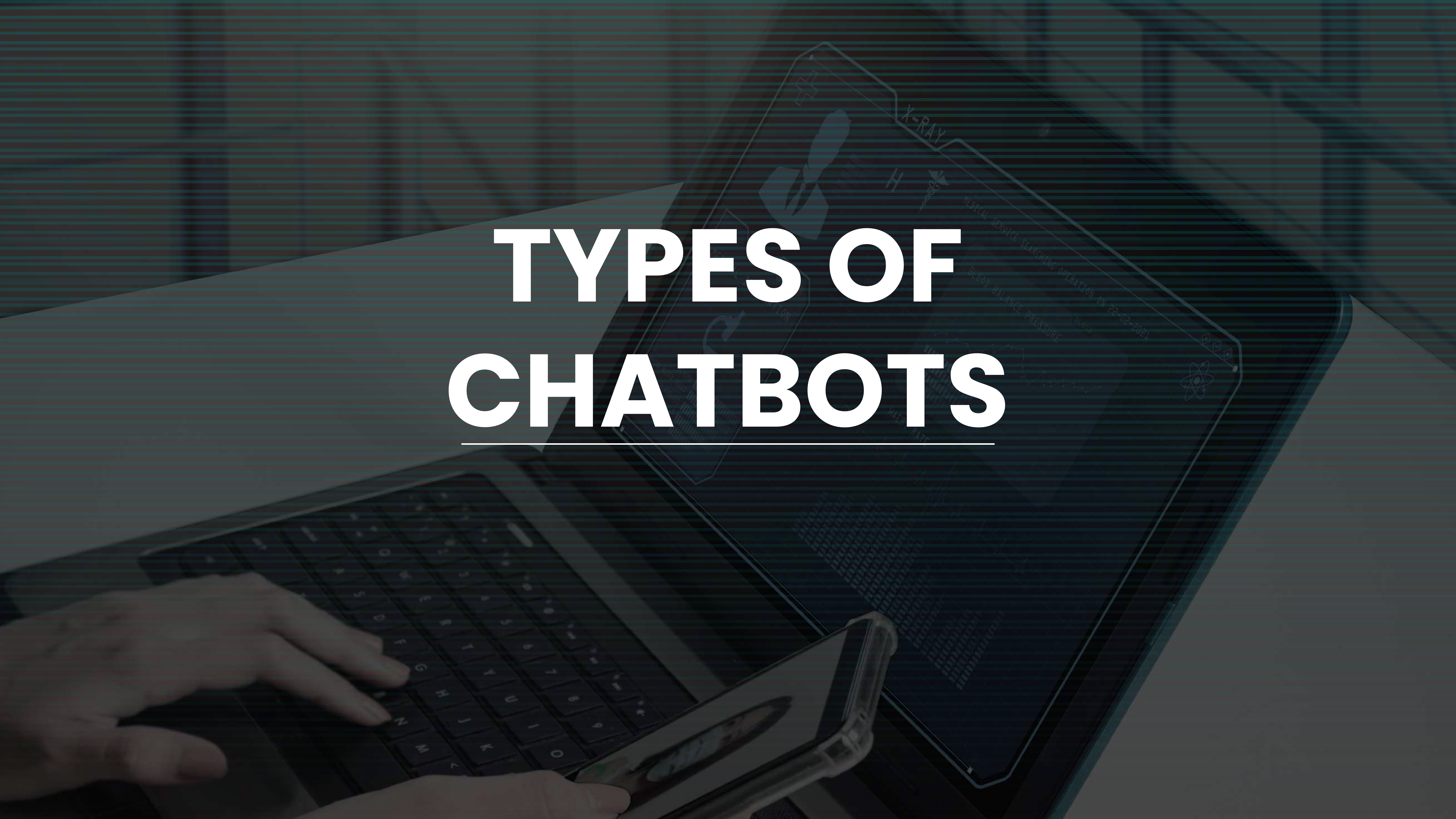
We people live in a modern technology where the technology gets updated within each and every second. In this competative era it becomes a complicated task for people to get updated according to the changes in order to make these type of tasks simpler these chat bots are implemented.

You people might be wondering what has these bots have done so magically or something like that

Apart from considering the factors like these type of implementations save time and reduce the manual work by reducing the work stress of humans, it plays a major role in providing precise,clean and a simple format for those who are actually working on how to upskill their knowledge.

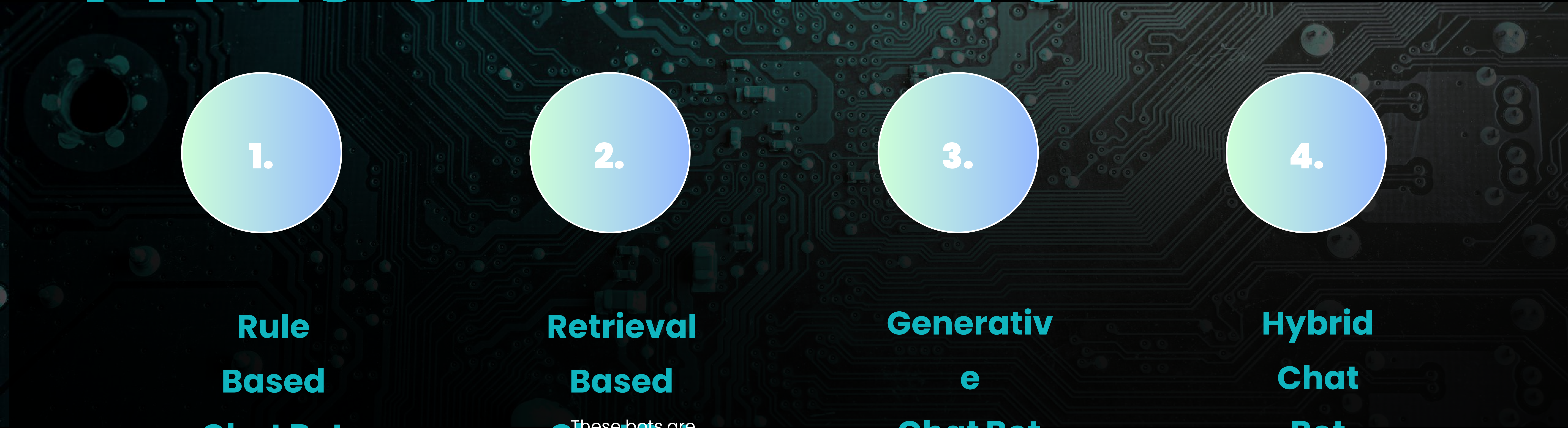
We have developed a ChatBot which provides a clean and precise format whatever the user provides.

Let's Get Started ➡



TYPES OF CHATBOTS

WHAT ARE THE TYPES OF CHATBOTS



1.

Rule Based Chat Bot

These bots work and provide according to specific rules given by the user.

2.

Retrieval Based Chat Bot

These bots are implemented in such a way that they provide solutions according to the user's queries and also turn their own dataset according to the operational designing

3.

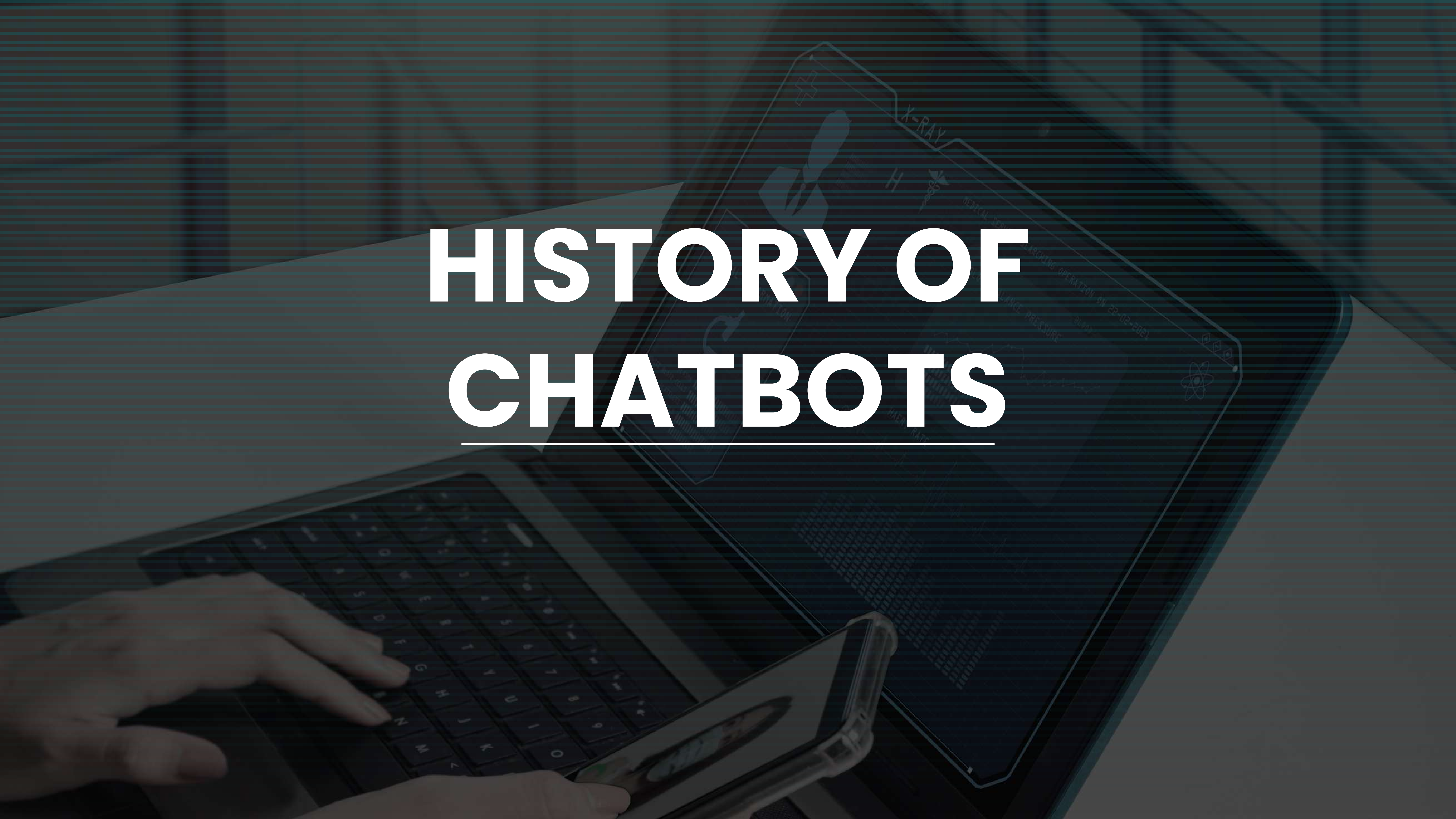
Generativ e Chat Bot

Provides response from scratch rather than providing those from the pre-defined ones.

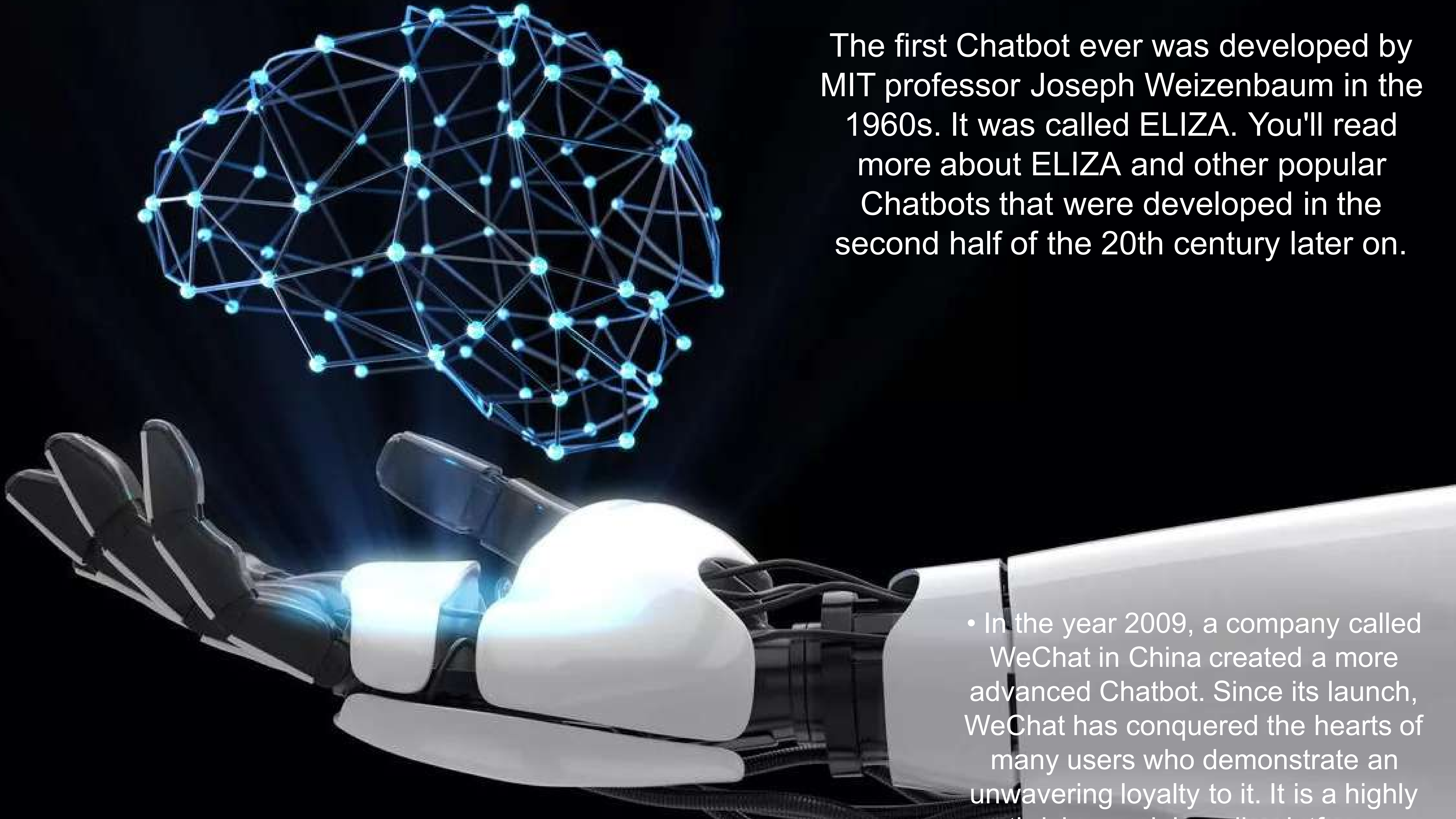
4.

Hybrid Chat Bot

Combination of both machine learning algorithms and predefined rules.



HISTORY OF CHATBOTS



The first Chatbot ever was developed by MIT professor Joseph Weizenbaum in the 1960s. It was called ELIZA. You'll read more about ELIZA and other popular Chatbots that were developed in the second half of the 20th century later on.

- In the year 2009, a company called WeChat in China created a more advanced Chatbot. Since its launch, WeChat has conquered the hearts of many users who demonstrate an unwavering loyalty to it. It is a highly

STEPS INVOLVED FOR DEVELOPING

Set Up the Development Environment

Install the required libraries

Gather training data

Preprocess the data

Implement NLP techniques

Train the chatbot model

Define chatbots responses and habits

Integrate internal services and API'S

Test and refine

Deploy the ChatBot

FUTURE INNOVATIONS

Personalized Learning

This means that the responses generated by the chatbot is easier to the user and the user or students feel it easier to understand the concept.

Real-time Feedback and Assessment

This part focuses on how well the students feel it easier to understand and how good the responses are..

Multimodal Learning Experiences

Integrating multimedia elements interactive simulations into the chatbot interface can enrich the learning experience and cater to different learning styles

Integration with Learning Management Systems

enhance accessibility and usability for students and educators. This includes interoperability with course materials, grading systems, and student progress tracking tools, streamlining the learning process for all stakeholders..

THE LANGUAGE MODEL FOR AI

We have developed our chat bot using Natural Language Processing(NLP) and languages like BERT(Bidirectional Encoder Representations from Transformers) and Open AI gpt 2(Generative Pre-trained Transformer)
We have implemented various techniques like tokenization,transformers and word count

These three techniques focuses on how well a bot can be developed by analyzing the user input and how to convert that into numerical values through BERT and Gpt these focuses on how well the interactions are done and how quick the responses are given.

Properties like tokenization focuses on the understanding part of the machine.
This method splits the words into various parts and retrieve them separately through this the machine gets a clear idea on what type of query is and how good the response must be made.

PROBLEM & SOLUTION MADE

PROBLEM &SOLUTION

Problem 1

Develop a chatbot that uses natural language processing to understand and answer students' queries on various subjects. This project involves data collection from educational resources, cleaning and preprocessing this data, training a language model, and integrating it into a chat interface.

Solution

We have developed our chat bot in such a way it is capable of analyzing,turing the problem statement and provides a precise response for the user.

We have developed a pre-trained model which focuses on providing fast replies.

INTEGRATION

Define Requirements:

Clearly outline the goals and requirements for integrating the smart tutor chatbot.

Choose a Platform or Framework:

Select a suitable platform or framework for developing and deploying the chatbot.

Develop the Chatbot:

Design and develop the chatbot using appropriate tools and technologies.

Implement AI and NLP:

Integrate artificial intelligence (AI) and natural language processing (NLP) capabilities into the chatbot to enable intelligent interactions with users.

Test and Refine:

Thoroughly test the chatbot to ensure that it functions correctly and meets the specified requirements.

Deploy and Monitor:

RESPONSIVENESS

Multi-platform compatibility:

Ensure that the chatbot is compatible with a wide range of devices and platforms, including web browsers, mobile devices, messaging apps, and voice assistants

Adaptive layout:

Design the chatbot interface to adapt to different screen sizes and resolutions.

Optimized performance:

Optimize the performance of the chatbot to ensure fast response times and smooth interactions, regardless of the user's device or network connection.

Context-awareness:

Design the chatbot to be context-aware, meaning it can remember and understand the context of previous interactions with the user.

Consistent user experience:

Maintain a consistent user experience across different devices and platforms to ensure that users can seamlessly transition between them without any confusion or disruption

ENSURING THE ACCURACY

Quality Training Data:

Start by ensuring that your chatbot is trained on high-quality, relevant data.

Fine-tuning and Iteration:

Continuously fine-tune and iterate on your chatbot's training data and algorithms based on user feedback and real-world usage.

Validation & Testing:

Use test suites and sample conversations to assess the chatbot's performance across different scenarios and use cases, and measure key metrics such as precision, recall, and F1 score.

Error Handling and Escalation:

Implement robust error handling mechanisms to gracefully handle cases where the chatbot is unable to provide a satisfactory response.

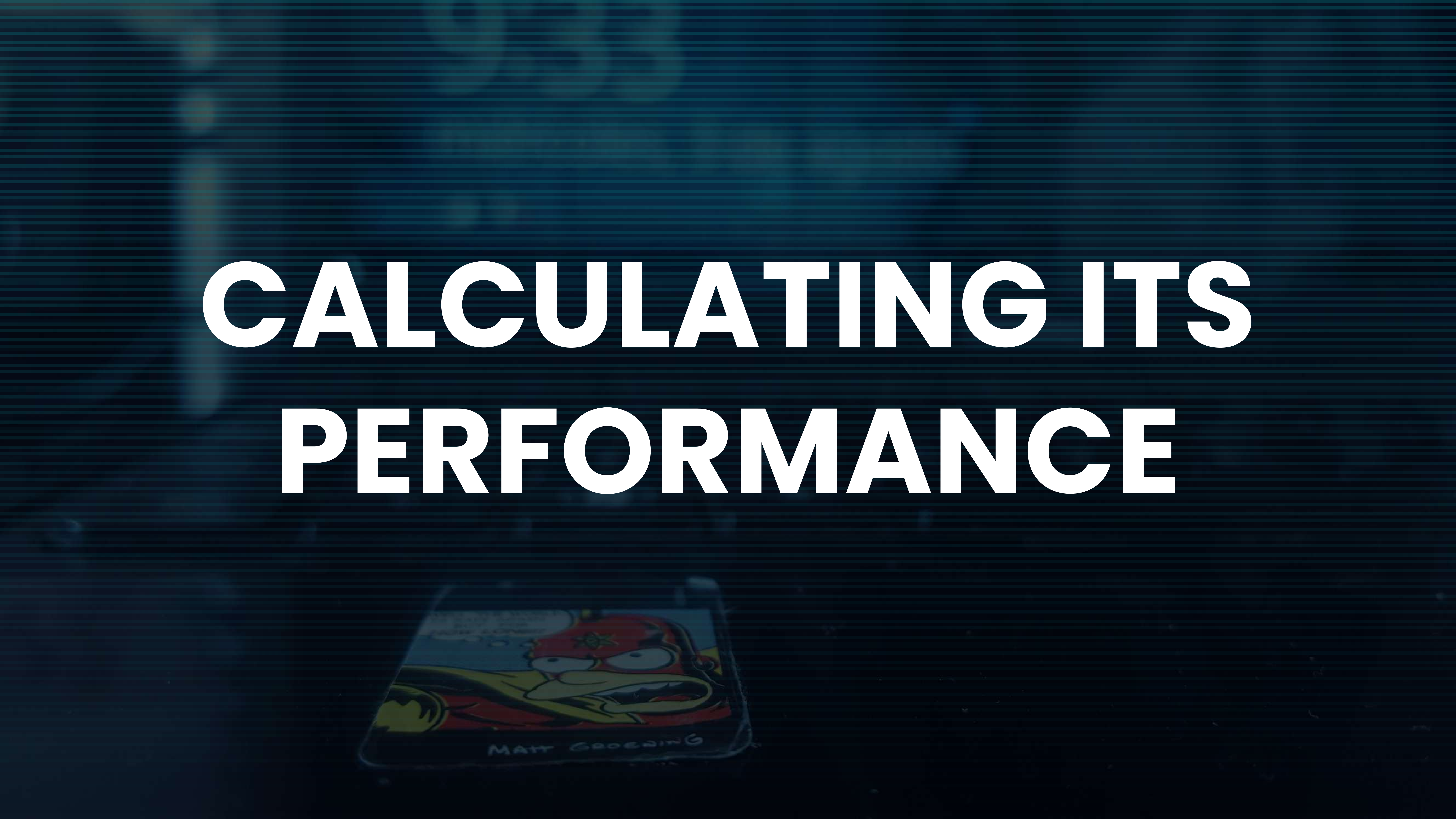
User Feedback Loops:

Establish feedback loops to gather input from users about their interactions with the chatbot.

Continuous Learning:

Implement mechanisms for continuous learning and adaptation based on real-time user interactions and feedback

CALCULATING ITS PERFORMANCE



Efficient Algorithms:

Use efficient algorithms and data structures for performing calculations within the chatbot

Optimize the code:

Minimize unnecessary computations, avoid redundant code, and optimize loops and conditional statements for better performance.

Caching and Memoization:

Implement caching and memoization techniques to store the results of expensive calculations and avoid recomputing them unnecessarily.

Asynchronous Processing:

Use asynchronous processing techniques to handle long-running calculations or external API calls without blocking the chatbot's main thread.

Performance Monitoring:

Monitor the performance of your chatbot's calculations in real-time to identify bottlenecks, latency issues, or areas for optimization.

Load Testing:

THANK YOU