



ANTEMATTER

# DOCUGPT

**YOUR OWN CHAT-BOT FOR DOCUMENTATIONS**



# PROBLEM STATEMENT

Documentation is always not readable +  
understandable by developers.

How do we solve that?



# PROBLEM STATEMENT


Documentation is always not readable +  
understandable by developers.

How do we solve that?

Welcome docuGPT.



# FEATURES

- 
- 01** SUPPORTS "READTHEDOCS.IO/ORG"
  - 02** JUST LIKE ANY OTHER CHAT-BOT
  - 03** CAN BE USED TO GENERATE EXAMPLES FOR BETTER UNDERSTANDING
  - 04** IS FAST AND RESPONSIVE
  - 05** CLEAN UI/UX EXPERIENCE

# GOALS AND OBJECTIVES

## Objective n° 1

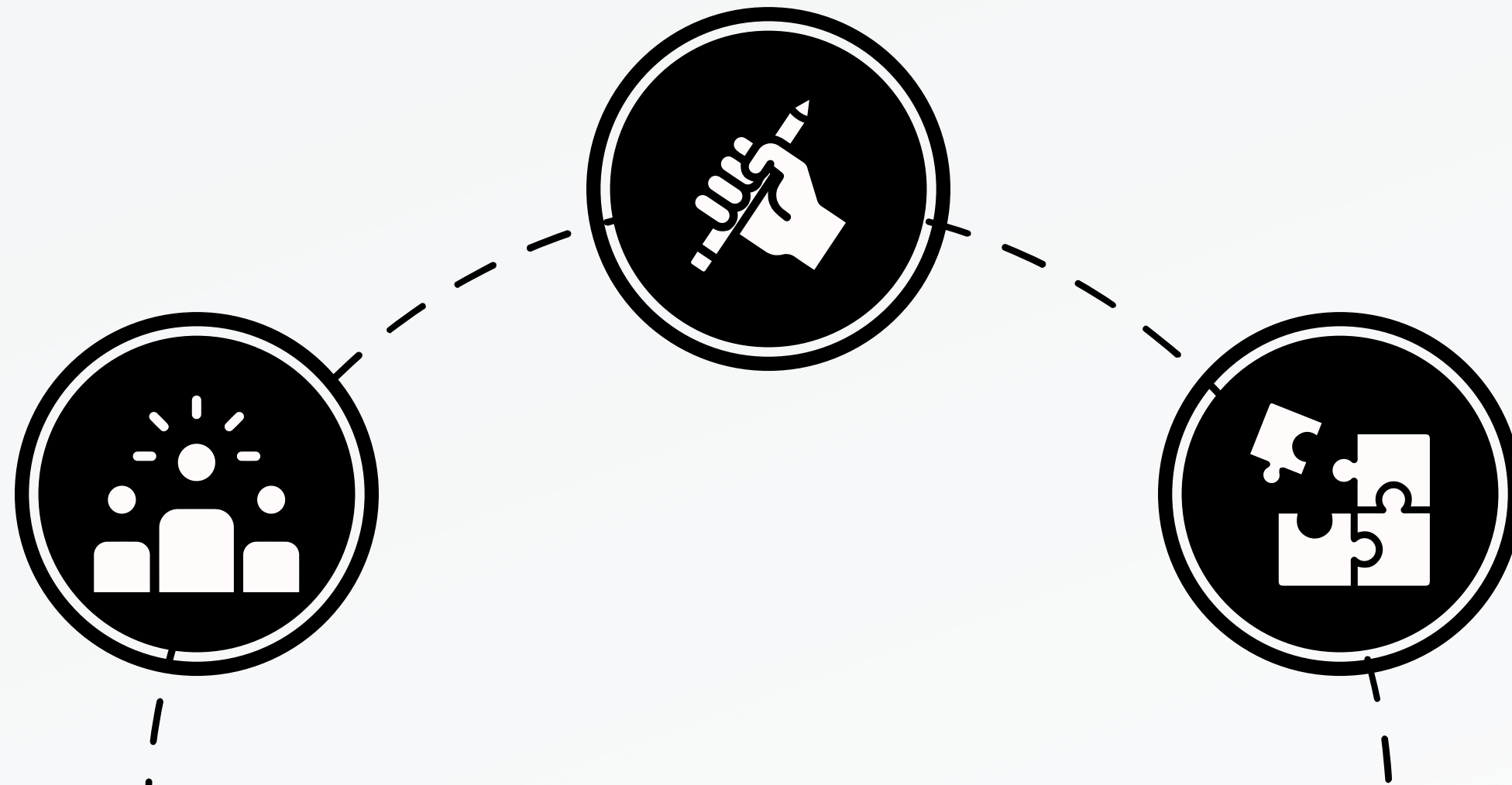
Know and understand documentations better

## Objective n° 2

Being able to find details about a specific thing in the documentation

## Objective n° 3

Reduces time and resources of developers





01

**FRONT-END**

For the front-end, we chose a very reliable tool known as "Streamlit". Streamlit's easy to setup and you don't have to spend too much time for front-end and go straight into the back-end.

02

**BACK-END**

We primarily used Lanchain. Langchain - extremely powerful tool for LLMs and chatbot make the whole process extremely smooth and convenient.

03

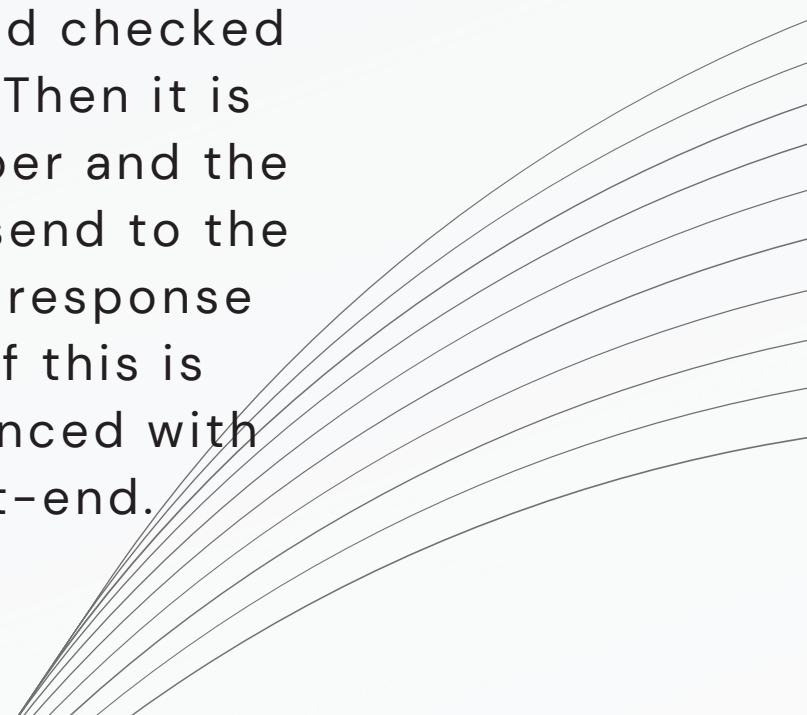
**DATABASE**

For database we first used FAISS but then switched to ChromaDB as the CTO of our team thought it was better.

04

**WORKFLOW**

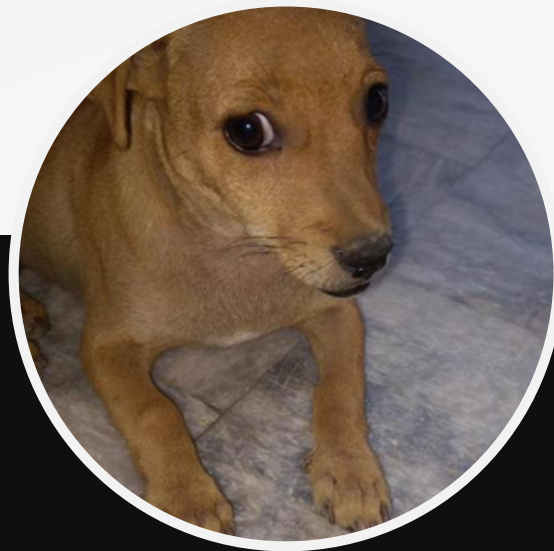
First the link is processed and checked for validity. Then it is send to scraper and the data is then send to the API to get a response back. All of this is perfectly synced with the front-end.



# OUR TEAM



Muhammad  
Rassam  
UI/UX designer



Muhammad  
Saad  
Back-end



Saad  
Mohsin  
Front-end

**THANK'S FOR  
WATCHING**



ANTEMATTER

