CAPSTONE PROJECT

FUN CHATBOT

PRESENTED BY : K.SUHITHA

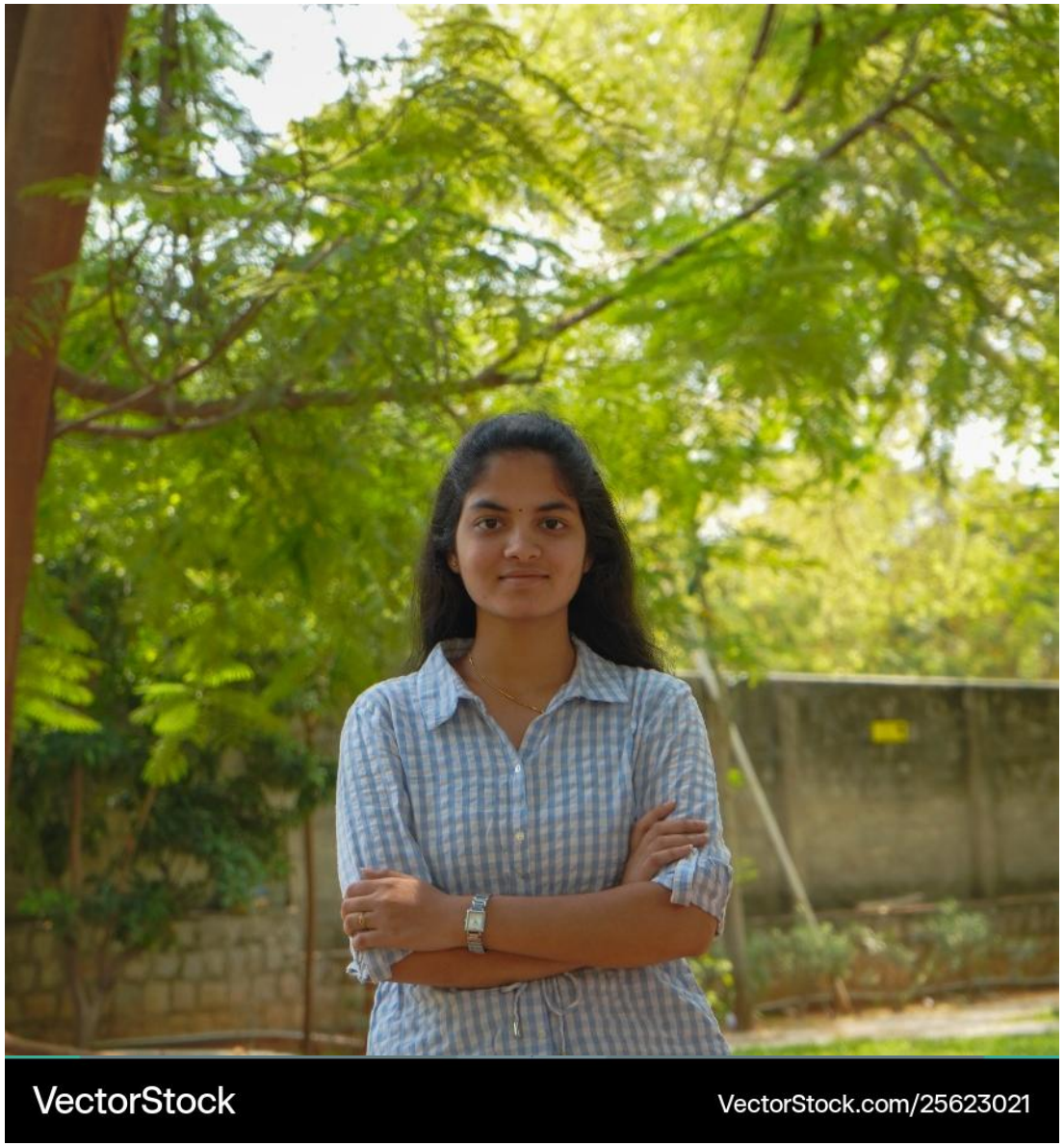
STUDENT NAME: K.SUHITHA

COLLEGE NAME: GEETHANJALI COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT: CSE (AI & ML)

EMAIL ID: ksuhitha2004@gmail.com

AICTE STUDENT ID: STU672b78da18d521730902234



OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result (Output Image)
- Conclusion
- Future Scope
- References

PROBLEM STATEMENT

In many learning environments, students often lack hands-on experience in building intelligent systems. A simple, interactive chatbot project provides a low-barrier entry point into the world of AI. Students need a way to demonstrate basic AI concepts like response generation, user interaction, and logic flow.

PROPOSED SOLUTION

- The proposed solution is a lightweight AI chatbot developed in Python. It uses simple rule-based logic and randomized responses to simulate conversation. The chatbot can greet users, tell jokes, and end conversations naturally. This is ideal for students demonstrating AI fundamentals during internships or mini-projects.

SYSTEM APPROACH

System Requirements:

- Python 3

Libraries Used:

- random
- time

Development:

- Console-based chatbot
- Uses keyword detection for response logic
- Randomized outputs for natural feel

ALGORITHM & DEPLOYMENT

- Algorithm:

- Match input text with known keywords (e.g., 'hello', 'joke', 'bye')

- Respond with a random choice from predefined responses

- Deployment:

- Local execution in Python console

- No external dependencies

- Easily extendable with more rules or NLP tools

RESULT

The chatbot responds to greetings, tells jokes, and ends conversations naturally.

Example Output:

User: Tell me a joke

Bot: Why did the computer go to therapy? Because it had too many bytes. 🧠💻

CONCLUSION

The chatbot project successfully demonstrates basic AI logic in an interactive format. It is easy to implement and serves as a fun introduction to conversational AI without needing external APIs.

FUTURE SCOPE

The chatbot can be enhanced with:

- Natural Language Processing
- Voice input/output
- GUI/Web interface
- Integration with Azure AI or OpenAI APIs for advanced interaction

REFERENCES

- Python documentation
- Stack Overflow examples
- ChatGPT for logic assistance
- https://www.geeksforgeeks.org

GitHub Link: https://github.com/Suhithareddy6/Fun-AI-ChatBot

Thank you