



# AI-Based College Admission Chatbot

**Gurudevan A, Akash V R Danish, Logith S, M Umamathy M E**

Department of Artificial Intelligence and Data Science, The Kavary Engineering College (Autonomous), Salem,  
Tamil Nadu, India

**ABSTRACT:** The AI-based College Admission Chatbot is designed to automate the process of providing admission-related information to students. Traditional admission systems rely on manual communication methods such as websites, emails, and enquiry desks, which often lead to delays and increased administrative workload. The proposed system utilizes artificial intelligence and natural language processing techniques to understand user queries and provide instant, accurate responses. The chatbot interacts with users through a conversational interface and retrieves relevant information from a structured database managed by an administrator. The system ensures 24/7 availability, reduces response time, and improves communication efficiency between institutions and students. Experimental results demonstrate that the chatbot effectively handles repetitive queries and enhances user experience. The proposed system provides a scalable and reliable solution for modern educational institutions.

**KEYWORDS:** Chatbot, Artificial Intelligence, NLP, Admission System, Automation

## I. INTRODUCTION

The college admission process requires students to gather information about courses, eligibility criteria, fee structures, and procedures. Traditional methods such as websites and enquiry offices are time-consuming and inefficient, especially during peak admission periods.

With advancements in artificial intelligence and natural language processing, chatbots have emerged as effective tools for automated communication. These systems allow users to interact using natural language and receive instant responses.

This paper proposes an AI-based chatbot that provides admission-related information through an interactive platform. The system reduces manual workload and improves accessibility for students.

## II. RELATED WORK

Several studies have explored chatbot applications in educational institutions. AI-based chatbots have been developed to automate student queries and improve communication efficiency. NLP-based systems enable accurate interpretation of user queries and enhance information retrieval.

Recent research highlights the use of machine learning, cloud computing, and personalization techniques in chatbot systems. These approaches improve response accuracy, scalability, and user satisfaction. However, many existing systems lack real-time efficiency and adaptability, which the proposed system aims to address.

## III. PROPOSED SYSTEM

The proposed system introduces an AI-based chatbot that provides automated responses to student queries. The system includes:

- **Admin Module:** Manages course details, fees, and admission data
- **Database:** Stores structured institutional information
- **NLP Module:** Processes user queries and identifies intent
- **Chatbot Engine:** Generates appropriate responses

Users can interact with the chatbot using natural language. The system analyzes the query and retrieves relevant information from the database.

### Advantages

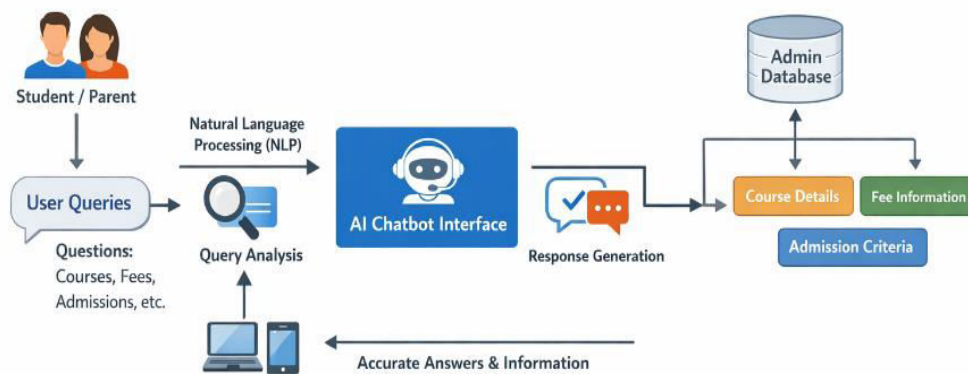
- Instant responses
- Reduced administrative workload
- 24/7 availability
- Improved user experience

## IV. IMPLEMENTATION DETAILS

The system is implemented using:

- **Frontend:** HTML, CSS, Bootstrap
- **Backend:** Python (Flask Framework)
- **Database:** MySQL
- **Libraries:** NLTK, NumPy

### Working Process:



The chatbot uses tokenization and keyword matching techniques to interpret queries and provide accurate responses.

## V. RESULTS AND DISCUSSION

The system was tested under different scenarios to evaluate performance and accuracy.

### Observations:

- Accurate responses for most queries
- Fast response time
- Handles multiple users simultaneously

### Testing Conducted:

- Unit Testing
- Integration Testing
- Functional Testing
- Performance Testing

The chatbot successfully handled admission-related queries and reduced dependency on manual support systems.

### Sample Program:

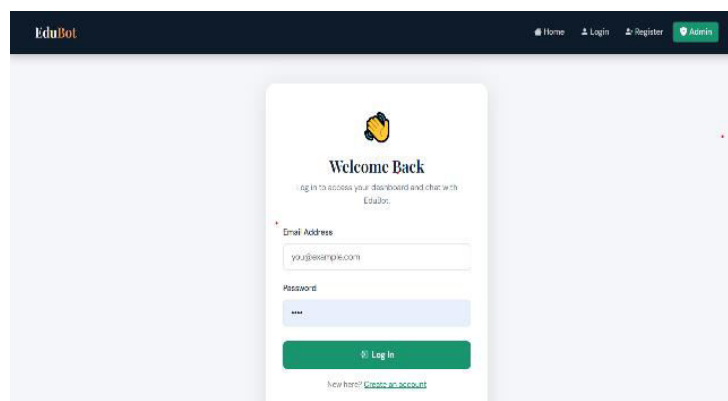
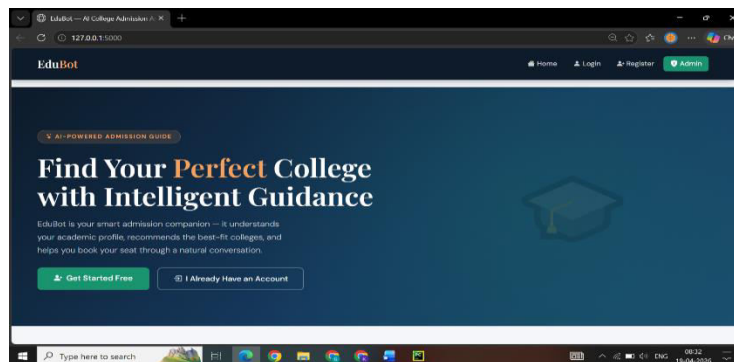
```
from flask import Flask, render_template, request, jsonify
import nltk
```

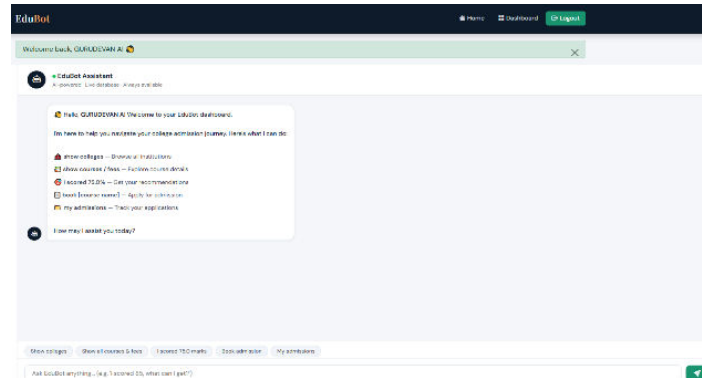
```
from nltk.stem import PorterStemmer
import json
import random
import numpy as np
# Initialize Flask and NLP tools
app = Flask(__name__)
stemmer = PorterStemmer()
nltk.download('punkt')
# 1. LOAD AND PREPROCESS DATA
with open('intents.json') as file:
    data = json.load(file)
words = []
classes = []
documents = []
ignore_letters = ['?', '!', '.', ','] # Organize data for "training" logic
for intent in data['intents']:
    for pattern in intent['patterns']:
        # Tokenize each word
        word_list = nltk.word_tokenize(pattern)
        words.extend(word_list)
        # Add to documents
        documents.append((word_list, intent['tag']))
        # Add to classes
        if intent['tag'] not in classes:
            classes.append(intent['tag'])
# Stem and lower each word and remove duplicates
words = sorted(list(set([stemmer.stem(w.lower()) for w in words if w not in ignore_letters])))
classes = sorted(list(set(classes)))
# 2. NLP UTILITY FUNCTIONS
def clean_up_sentence(sentence):
    sentence_words = nltk.word_tokenize(sentence)
    sentence_words = [stemmer.stem(word.lower()) for word in sentence_words]
    return sentence_words
def bag_of_words(sentence):
    sentence_words = clean_up_sentence(sentence)
    bag = [0] * len(words)
    for s in sentence_words:
        for i, w in enumerate(words):
            if w == s:
                bag[i] = 1
    return np.array(bag)
def predict_class(sentence):
    # For a full ML project, you'd use model.predict() here.
    # This is a robust manual matcher for college-level implementation:
    bow = bag_of_words(sentence)
    # Simple Intent Matching Logic
    input_words = set(clean_up_sentence(sentence))
    best_match = None
    max_overlap = 0
    for intent in data['intents']:
        for pattern in intent['patterns']:
            pattern_words = set(clean_up_sentence(pattern))
            overlap = len(input_words.intersection(pattern_words))
            if overlap > max_overlap:
```

```

max_overlap = overlap
best_match = intent['tag']
return best_match if max_overlap > 0 else "fallback"
def get_response(intent_tag):
    for i in data['intents']:
        if i['tag'] == intent_tag:
            return random.choice(i['responses'])
    return "I'm sorry, I don't have information on that. Can you try asking about courses or fees?"
# 3. FLASK ROUTES (Rendering & API)
@app.route('/')
def index():
    """Renders the Student Landing Page"""
    return render_template('index.html')
@app.route('/admin')
def admin():
    """Renders the Admin Dashboard with Graphs"""
    return render_template('admin.html')
@app.route('/get_response', methods=['POST'])
def chat():
    """The API endpoint the chatbot calls"""
    user_message = request.json.get("message")
    if not user_message:
        return jsonify({"response": "Empty message received."})
    intent = predict_class(user_message)
    response = get_response(intent)
    return jsonify({"response": response})
if __name__ == "__main__":
    app.run(debug=True, port=5000)

```





## VI. CONCLUSION

The AI-based College Admission Chatbot provides an efficient solution for automating student enquiries. The system improves communication, reduces workload, and ensures accurate information delivery. By integrating AI and NLP techniques, the chatbot enhances accessibility and user experience. The system is reliable, scalable, and suitable for modern educational institutions.

## VII. FUTURE WORK

Future enhancements may include:

- Voice-based interaction
- Multilingual support
- Integration with mobile applications
- Advanced machine learning models

These improvements will further enhance system performance and usability.

## REFERENCES

1. Darshan, P., et al., "AI Chatbot for Educational Institution," 2025.
2. Sinha, D., et al., "Chatbot-powered College Support Systems," 2025.
3. Nadir, Z., et al., "SQUbot: Personalized Chatbot System," 2025.
4. Akshaya, D. K., et al., "Intelligent College Enquiry Chatbot," 2025.
5. Devi, M. N., et al., "Smart DIGI-DESK System," 2025.
6. Harini, K., et al., "AI-Based Student Assistance Chatbot," 2025.
7. Ramani, D. R., et al., "Campus Companion Chatbot," 2025.
8. Ramalingam, C., et al., "Cloud-Based Chatbot," 2025.
9. Thamilselvan, R., et al., "AI Chatbot for College Websites," 2024.
10. Razak, M. S., et al., "Chatbot using NLP and ANN," 2024.



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  [ijirset@gmail.com](mailto:ijirset@gmail.com)



[www.ijirset.com](http://www.ijirset.com)

Scan to save the contact details