Mini Project: Chatbot for Customer Support

1. Objective

Create an AI-powered chatbot to automate responses to customer queries using Natural Language Processing (NLP).

2. Tools & Technologies

- Programming Language: Python
- Libraries/Frameworks:
  * NLTK / spaCy / Rasa (for NLP)
  * Flask (for web integration)
  * Dialogflow (optional, for no-code platform)
- Interface: Web-based or terminal-based

3. Project Workflow

Step 1: Intent Recognition

Define common customer intents (e.g., Order Status, Return Policy). Train using sample data.

Step 2: Chatbot Logic

Develop a function that matches input queries to predefined intents and generates responses.

Example Code:

```
def get_response(user_input):
    if "order" in user_input:
        return "Please provide your order ID."
    elif "return" in user_input:
        return "You can return products within 30 days."
    else:
        return "Sorry, I didn't understand that."
```

Step 3: Integration (Optional)

Integrate chatbot into a web interface using Flask.

Example Route:

```
@app.route('/chat', methods=['POST'])
def chat():
    user_input = request.form['message']
    response = get_response(user_input)
    return jsonify({'reply': response})
```

4. Skills Learned

- NLP for conversational agents

- Chatbot framework integration

- Query handling and response automation

5. Real-World Applications

Automating customer support in businesses and services. Enhancing user experience with 24/7 support.

6. Optional Enhancements

- Add speech recognition

- Deploy on Telegram or WhatsApp

- Use machine learning for improved intent matching