DAY – 33

12 September 2025

Updated Objective

- Use **English + Punjabi** understanding (bilingual).
- Enable **farmers** to communicate with **website developers**.
- Use **basic NLP rules (NLTK)** for now.
- Future-ready for **ML/LLM integration**.

What Needs to Change or Add?

1. Expand Pairs to Include Punjabi

You can add Punjabi phrases as pattern matches directly into the pairs list:

```
[
   r"ਮੇਰਾ ਨਾਮ (.*) ਹੈ",
   ["ਸਤ ਸ੍ਰੀ ਅਕਾਲ! ਤੁਸੀਂ ਕਿਵੇਂ ਹੋ?",]
],
[
   r"ਹੈਲੋ|ਹਾਈ|ਸਤ ਸ੍ਰੀ ਅਕਾਲ",
   ["ਸਤ ਸ੍ਰੀ ਅਕਾਲ! ਤੁਹਾਡਾ ਸੁਆਗਤ ਹੈ",]
],
[
   r"ਤੁਸੀਂ ਕੌਣ ਹੋ?",
   ["ਮੈਂ ਇੱਕ ਚੈਟਬੋਟ ਹਾਂ ਜੋ ਕਿਸਾਨਾਂ ਅਤੇ ਵਿਕਾਸਕਾਰਾਂ ਦੀ ਮਦਦ ਲਈ ਬਣਾਇਆ ਗਿਆ ਹੈ.",]
],
[
   r"ਮੈਨੂੰ ਸਹਾਇਤਾ ਦੀ ਲੋੜ ਹੈ",
   ["ਕਿਰਪਾ ਕਰਕੇ ਦੱਸੋ ਕਿ ਤੁਸੀਂ ਕਿਸ ਚੀਜ਼ ਦੀ ਮਦਦ ਚਾਹੁੰਦੇ ਹੋ?",]
],
```

You can mix both Punjabi and English pairs together in the same list.

---

2. Customize Responses to Fit Use Case

Edit your responses so it feels like a **communication assistant between farmer and web developer**.

Examples:

```
[
    r"I want a new feature on my website",
    ["Sure, could you please describe the feature you want? I will share it with the developer."]
],
[
    r"ਮੈਂ ਆਪਣੀ ਵੈੱਬਸਾਈਟ 'ਤੇ ਨਵੀਂ ਵਿਸ਼ੇਸ਼ਤਾ ਚਾਹੁੰਦਾ ਹਾਂ",
    ["ਕਿਰਪਾ ਕਰਕੇ ਦੱਸੋ ਕਿ ਤੁਸੀਂ ਕੀ ਵਿਸ਼ੇਸ਼ਤਾ ਚਾਹੁੰਦੇ ਹੋ। ਮੈਂ ਇਹ ਡਿਵੈਲਪਰ ਨਾਲ ਸਾਂਝੀ ਕਰਾਂਗਾ।"]
],
```

---

3. Rename Chatbot Identity

Rename the bot as a bridge between farmer & dev:

["I am *KisanBot*, here to help farmers and developers communicate."]

---

4. Keep Flask Integration As-Is

No changes needed in your Flask setup — it already:

- Loads the chatbot on a webpage.
- Uses GET API to fetch response dynamically.

Final Tips for Better Punjabi Handling

Since NLTK has **limited support for Punjabi NLP**, here's what you can do:

▶ **Optionally add Transliteration:**

Use indic-transliteration to normalize input.

```
pip install indic-transliteration
```

▶ **Or Add Translation (for LLM support later):**

Use googletrans to translate Punjabi to English, process in English, then translate back.

```
pip install googletrans==4.0.0-rc1
```

Folder Structure

```
farmer_chatbot/
├── app.py  ← Your Flask + Chat code
├── templates/
│   └── index.html  ← Frontend UI
├── static/
│   └── style.css (optional)
```

Here's What We'll Build:

A **Flask-based chatbot** using the **Google multilingual BERT model** or a similar multilingual NLP model that:

- Accepts farmer input in English or Punjabi
- Uses NLP to understand and respond
- Responds as a helpful assistant between **farmer and developer**

Step-by-Step Solution

1. Install Required Libraries

```
pip install flask flask-cors transformers torch
```

---

2. Python Code (app.py) – NLP Model with Punjabi + English

```
from flask import Flask, request, jsonify
from flask_cors import CORS
from transformers import pipeline

# Initialize Flask app
app = Flask(__name__)
CORS(app)

# Load multilingual model
chatbot = pipeline("text2text-generation", model="google/mt5-small")

# Basic intent examples (could be expanded with RAG or custom model later)
def generate_response(user_input):
    # Simple prompt style (could be replaced with LangChain or RAG later)
    prompt = f"Reply in the same language: {user_input}"
    result = chatbot(prompt, max_length=100)[0]["generated_text"]
    return result

# Route for chatbot API
@app.route("/chat", methods=["POST"])
def chat():
    data = request.json
    user_message = data.get("message", "")
    response = generate_response(user_message)
    return jsonify({"response": response})
```

```
# Run Flask app
if __name__ == "__main__":
    app.run(debug=True)
```

---

3. Frontend (index.html) – Farmer Chat UI

```
<!DOCTYPE html>
<html>
<head>
    <title>Farmer-Developer Chatbot</title>
</head>
<body>
    <h2>ਕਿਸਾਨ ਚੈਟਬੋਟ | Farmer Chatbot</h2>
    <input type="text" id="message" placeholder="Type in English or Punjabi">
    <button onclick="sendMessage()">Send</button>
    <div id="chat"></div>

    <script>
        async function sendMessage() {
            const msg = document.getElementById("message").value;
            const res = await fetch("http://127.0.0.1:5000/chat", {
                method: "POST",
                headers: { "Content-Type": "application/json" },
                body: JSON.stringify({ message: msg })
            });
            const data = await res.json();
            document.getElementById("chat").innerHTML += `<p><strong>You:</strong> ${msg}</p><p><strong>Bot:</strong> ${data.response}</p>`;
        }
    </script>
</body>
</html>
```

Output Example:

Input (Punjabi):

ਮੈਂ ਆਪਣੀ ਵੈੱਬਸਾਈਟ ਵਿੱਚ ਇੱਕ ਨਵਾਂ ਫੀਚਰ ਜੋੜਵਾਉਣਾ ਚਾਹੁੰਦਾ ਹਾਂ।

Output:

ਕਿਰਪਾ ਕਰਕੇ ਦੱਸੋ ਤੁਸੀਂ ਕਿਹੋ ਜਿਹਾ ਫੀਚਰ ਜੋੜਵਾਉਣਾ ਚਾਹੁੰਦੇ ਹੋ, ਮੈਂ ਇਹ ਡਿਵੈਲਪਰ ਨੂੰ ਭੇਜ ਦਿਆਂਗਾ।

Input (English):

I want to add a new feature to my website.

Output:

Sure! Please describe the feature you want to add. I will share it with the developer.

What NLP Model is Used?

- google/mt5-small — supports many Indian languages including **Punjabi** and **English**
- You can **replace it with OpenAI GPT or Gemini** for higher quality