DAY – 88

8 December 2025

.env file

```
HF_API_KEY=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
GEMINI_API_KEY=AIzaSyD4t00x72WBWr0dPEhjLyYtlF_C0IyrqB8

# OPTIONAL (only if needed)
HF_TOKEN=your_huggingface_token

# Flask session secret
FLASK_SECRET_KEY=f5e5cfee8ac603a3576b739c92810d9540083e38d85272da0edcf701f81ca
6ee
```

Dataset.txt

[english]

what is your name = I am your assistant chatbot.

hello = Hello! How can I help you today?

bye = Goodbye! Have a great day.

okay = great□

thanku = welcome□

What is Comparative Command Statement = This report compares two command areas or outlets side by side. It helps evaluate water distribution, land utilization, and system performance. Officials rely on it for planning improvements and resource allocation. It highlights differences in irrigation demand and supply. The comparative report is generated in clear tabular format.

What is Command Statement = The Command Statement provides a summary of the command area under a specific outlet or channel. It includes land area, irrigation coverage, and operational details. Officials use it for planning and monitoring water supply. It helps compare expected vs actual irrigation coverage. The statement is used for administrative decision-making.

What is Jamabandi = In the context of land records, Jamabandi is an official "Record of Rights" document that includes details about a specific parcel of land's ownership, cultivation, and revenue status. While "Jamabandi" itself is primarily a term for general land records, its connection to water resources comes from the fact that these records often document the source and nature of irrigation for agricultural land.

What is A form = A-Form is a supporting land record used for validating farmer applications. It includes essential landholding details such as owner name, area, and plot numbers. Irrigation authorities use it to verify application authenticity. It forms part of the documentation for processing irrigation requests. It ensures accuracy in field-level verification.

What is Naksha nakal = "Naksha nakal" refers to obtaining a certified copy (nakal) of a land map (naksha). In the context of water resources, this likely relates to accessing official maps or records that detail land use, including the location of water bodies or irrigation infrastructure on a specific parcel of land.

Session id

```
# === Session Configuration ===
app.secret_key = "chatbot_secret_key"
app.config["SESSION_TYPE"] = "filesystem"
app.config["SESSION_FILE_DIR"] = os.path.join(os.getcwd(), "flask_session")  # persistent session folder
app.config["SESSION_PERMANENT"] = False
app.config["PERMANENT_SESSION_LIFETIME"] = 3600  # 1 hour

Session(app)
```

```
   @app.route("/get", methods=["POST"])

def chatbot_reply():
    data = request.get_json()
    user_message = data.get("msg", "").strip()
    user_lang = data.get("lang", "english")

    if not user_message:
        return jsonify({"response": "□ Please enter a message."})

    # Auto-load training data if needed
    if check_reload_needed():
        load_and_train()
    # Load chat history from session (list of (user, bot) tuples)
    history = session.get("chat_history", [])

    # Pass the full history to get_response — get_response will append the new
pair
    response = get_response(user_message, user_lang, history)

    # Save updated history back to session
    session["chat_history"] = history
    session.modified = True  # Force Flask to save the updated session

    return jsonify({"response": response})

if __name__ == "__main__":
    app.run(debug=False)
```

requests

python-dotenv

nltk

PyPDF2

pytesseract

pdf2image

scikit-learn

# optional (only if you want RAG & LLM integration):

# langchain

# chromadb

# sentence-transformers

# huggingface-hub

# groq-client