Report

Web Chatbot with Groq API

Submitted by: Husnain Ali
Intern ID: TN/IN01/PY/009
Internship Domain: Python

1. Overview

Objective: Build a minimal web-based chatbot that:

Uses **Groq's API** (Llama 3/Mixtral) for ultra-fast responses Features a clean **text-only interface** (no webcam/media) Runs on Flask backend with HTML/JS frontend

Key Advantage:

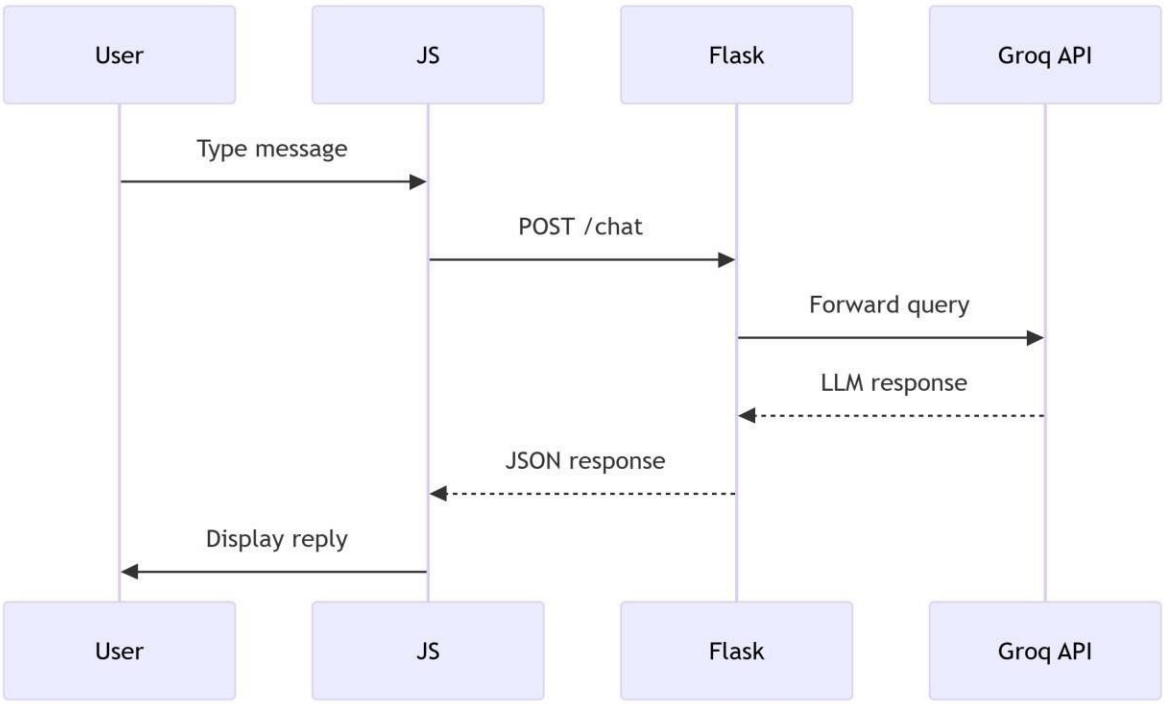
100% text-based: Lower latency, no privacy concerns from camera/mic
Easy to deploy: Single Flask app + static files

2. Tech Stack

Component	Technology
Backend	Flask (Python)
Frontend	HTML/CSS/JS (no React)
LLM	Groq API (Llama 3 70B)
Deployment	Render/AWS EC2 (optional)

3. System Flow

1. **User** types message → **JS** sends to Flask
2. **Flask** forwards to **Groq API**
3. **Groq** generates response → Returns to **Flask** → **User**



4. Implementation Plan

Phase 1: Setup (1 Day)

- Flask app with single `/chatbot` route
- Basic HTML form + JS + Css

Phase 2: Core Features (3 Days)

1. Streaming responses (Groq supports Server-Sent Events)
2. Chat history persistence (localStorage or Flask session)
3. Mobile-responsive CSS Phase 3: Extras (1 Days)

- Rate limiting
- Loading animations

5. Code Preview

Flask Backend (app.py**)**:

chtbot > app.py

```python
from flask import Flask, render_template, request, jsonify
from dotenv import load_dotenv
from langchain_groq import ChatGroq

load_dotenv()

app = Flask(__name__)

llm = ChatGroq(
    model="llama-3.1-8b-instant",
    temperature=0.0,
    max_retries=2,
)

@app.route('/')
def home():
    return render_template('chat.html')  # Make sure your template is named chat.html

@app.route('/get', methods=['POST'])
def get_bot_response():
    user_message = request.form['msg']
    response = llm.invoke(user_message)
    bot_response = response.content
    return jsonify({'response': bot_response})

if __name__ == '__main__':
    app.run(debug=True)
```

Frontend (JS):

JS script.js

chtbot > static > JS script.js > ...

```javascript
document.addEventListener('DOMContentLoaded',
    function(){
    //Toggle chatbot visibility
    const chatbotToggler = document.querySelector(".chatbot-toggler");
    const closeBtn = document.querySelector(".close-btn");
    const chatbot = document.querySelector(".chatbot");

    chatbotToggler.addEventListener("click", () => {
        document.body.classList.toggle("show-chatbot");
    });

    closeBtn.addEventListener("click", () => {
        document.body.classList.remove("show-chatbot");
    });

    // Send message function
    function sendMessage() {
        const userInput = $("#user-input");
        const msg = userInput.val().trim();

        if(!msg) return;

        // Add user message to chatbox
        $("#chatbox").append(
            `<li class="chat outgoing"><p>${msg}</p></li>`
        );

        // Clear input and scroll to bottom
        userInput.val("");
        scrollToBottom();

        // Show "Thinking..." message
```

6. Expected Output

- A working demo at http://localhost:5000with:
  - Text input box oReal-time chat display
  - **Speed**: Responses in <1 second (Groq benchmark)