```python
1   import os
2   import time
3   import json
4   import queue
5   import threading
6   import numpy as np
7   import av
8   import streamlit as st
9
10  # -------------------------------------------------
11  # Page Config MUST be first
12  # -------------------------------------------------
13  st.set_page_config(
14      page_title="⚡ GROQ PULSE",
15      page_icon="⚡",
16      layout="wide",
17      initial_sidebar_state="expanded",
18  )
19
20  # -------------------------------------------------
21  # Load secrets from Streamlit's .toml config
22  # -------------------------------------------------
```

PROBLEMS 6   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

```
● PS C:\Users\aqsa amber> cd "C:\groq"
● PS C:\groq> Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
● PS C:\groq> .\.venv\Scripts\Activate.ps1
● (.venv) PS C:\groq> python -m pip install --upgrade pip
  Requirement already satisfied: pip in c:\groq\.venv\lib\site-packages (25.0.1)
● (.venv) PS C:\groq> pip install -r requirements.txt
  Requirement already satisfied: streamlit==1.37.1 in c:\groq\.venv\lib\site-packages (from -r requirements.txt (line 2)) (1.37.1)
  Requirement already satisfied: streamlit-webrtc==0.47.1 in c:\groq\.venv\lib\site-packages (from -r requirements.txt (line 3)) (0.47.1)
  Requirement already satisfied: av==10.0.0 in c:\groq\.venv\lib\site-packages (from -r requirements.txt (line 4)) (10.0.0)
  Requirement already satisfied: numpy<1.25,>=1.21.6 in c:\groq\.venv\lib\site-packages (from -r requirements.txt (line 5)) (1.24.4)
  Requirement already satisfied: scipy<1.11,>=1.7.3 in c:\groq\.venv\lib\site-packages (from -r requirements.txt (line 6)) (1.10.1)
```

```python
22  # -------------------------------------------------
23  api_key = st.secrets.get("GROQ_API_KEY", "")
24  api_url = st.secrets.get("GROQ_API_URL", "https://api.groq.com/openai/v1/chat/completions")
25  tts_enabled_default = st.secrets.get("TTS_ENABLED", False)
26  tts_provider = st.secrets.get("TTS_PROVIDER", "pyttsx3")
27
28  st.write("API Key Loaded:", api_key[:5] + "..." if api_key else "⚠️ Not set")
29
30  # -------------------------------------------------
31  # Local utils
32  # -------------------------------------------------
33  import sys
34  sys.path.append(os.path.dirname(os.path.abspath(__file__)))
35
36  from utils.groq_client import GroqClient
37  from utils.ocr import extract_text_from_file
38  from utils.tts import speak_text_if_enabled
39  from utils.db import ConversationStore
40
41  # -------------------------------------------------
42  # WebRTC (browser mic support) - Hybrid STT
43  # -------------------------------------------------
44  from streamlit_webrtc import webrtc_streamer, AudioProcessorBase, WebRtcMode
```

PROBLEMS 6   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

```
  Requirement already satisfied: pydeck<1,>=0.8.0b4 in c:\groq\.venv\lib\site-packages (from streamlit==1.37.1->-r requirements.txt (line 2)) (0.9.1)
  Requirement already satisfied: tornado<7,>=6.0.3 in c:\      Open folder in new window (ctrl + click)     streamlit==1.37.1->-r requirements.txt (line 2)) (6.4.2)
  Requirement already satisfied: watchdog<5,>=2.1.5 in c:\                                                  m streamlit==1.37.1->-r requirements.txt (line 2)) (4.0.2)
  Requirement already satisfied: aiortc<2.0.0,>=1.4.0 in c:\groq\.venv\lib\site-packages (from streamlit-webrtc==0.47.1->-r requirements.txt (line 3)) (1.9.6
  Requirement already satisfied: comtypes in c:\groq\.venv\lib\site-packages (from pyttsx3==2.90->-r requirements.txt (line 16)) (1.4.12)
  Requirement already satisfied: pypiwin32 in c:\groq\.venv\lib\site-packages (from pyttsx3==2.90->-r requirements.txt (line 16)) (223)
  Requirement already satisfied: pywin32 in c:\groq\.venv\lib\site-packages (from pyttsx3==2.90->-r requirements.txt (line 16)) (311)
  Requirement already satisfied: charset-normalizer<4,>=2 in c:\groq\.venv\lib\site-packages (from requests<3.0,>=2.28->-r requirements.txt (line 7)) (3.4.3)
  Requirement already satisfied: idna<4,>=2.5 in c:\groq\.venv\lib\site-packages (from requests<3.0,>=2.28->-r requirements.txt (line 7)) (3.10)
  Requirement already satisfied: urllib3<3,>=1.21.1 in c:\groq\.venv\lib\site-packages (from requests<3.0,>=2.28->-r requirements.txt (line 7)) (2.2.3)
  Requirement already satisfied: certifi>=2017.4.17 in c:\groq\.venv\lib\site-packages (from requests<3.0,>=2.28->-r requirements.txt (line 7)) (2025.8.3)
  Requirement already satisfied: greenlet!=0.4.17 in c:\groq\.venv\lib\site-packages (from sqlalchemy<2.1,>=1.4->-r requirements.txt (line 19)) (3.1.1)
```

```python
43   # -------------------------------------------------
44   from streamlit_webrtc import webrtc_streamer, AudioProcessorBase, WebRtcMode
45
46   USE_VOSK = False
47   vosk_model = None
48
49   try:
50       from vosk import Model, KaldiRecognizer
51       model_path = "vosk-model-small-en-us-0.15"
52       if os.path.exists(model_path):
53           vosk_model = Model(model_path)
54           USE_VOSK = True
55           st.success("✅ Using Vosk offline speech recognition")
56       else:
57           st.warning("⚠️ Vosk model not found, will use Google SpeechRecognition")
58   except ImportError:
59       st.warning("⚠️ Vosk not installed, using Google SpeechRecognition instead")
60
61   # Google fallback
62   import speech_recognition as sr
63
64   # -------------------------------------------------
```

```python
65   # AudioProcessor classes
66   # -------------------------------------------------
67   class VoskAudioProcessor(AudioProcessorBase):
68       def __init__(self):
69           self.q = queue.Queue()
70           self.recognizer = KaldiRecognizer(vosk_model, 16000)
71           self.running = True
72           self._last_text = None
73           self.thread = threading.Thread(target=self._process_audio, daemon=True)
74           self.thread.start()
75
76       def recv_audio_frame(self, frame: av.AudioFrame):
77           audio = frame.to_ndarray().flatten()
78           if audio.dtype != np.float32:
79               audio = audio.astype(np.float32)
80           np.clip(audio, -1.0, 1.0, out=audio)
81           pcm16 = (audio * 32767).astype(np.int16).tobytes()
82           try:
83               self.q.put_nowait(pcm16)
84           except queue.Full:
85               pass
86           return frame
```

```python
130              st.markdown(f" <style>{f.read()}</style> ", unsafe_allow_html=True)
131
132     # -----------------------------------------------
133     # Session initialization
134     # -----------------------------------------------
135     if "theme" not in st.session_state:
136         st.session_state.theme = "light"
137     if "settings" not in st.session_state:
138         st.session_state.settings = {
139             "temperature": 0.2,
140             "max_tokens": 512,
141             "top_p": 1.0,
142             "model": "llama3-8b-8192"
143         }
144     if "current_conversation" not in st.session_state:
145         st.session_state.current_conversation = None
146     if "messages" not in st.session_state:
147         st.session_state.messages = []
148     if "debug_mode" not in st.session_state:
149         st.session_state.debug_mode = False
150
151     # -----------------------------------------------
152     # Initialize Groq client and database
```

```python
267         finally:
268             elapsed = time.time() - start_time
269
270         response_placeholder.markdown(
271             f"<div class='msg assistant'>🖥 {assistant_content}</div>", unsafe_allow_html=True)
272
273         st.session_state.messages.append({"role": "assistant", "content": assistant_content})
274         st.session_state.current_conversation["messages"] = st.session_state.messages
275         store.save_conversation(st.session_state.current_conversation)
276         store.log_interaction(text_in, assistant_content, elapsed,
277             conversation_id=st.session_state.current_conversation["id"])
278
279         if st.session_state.get("tts_enabled"):
280             speak_text_if_enabled(assistant_content)
281
282         st.rerun()
283
284     # -----------------------------------------------
285     # Footer
286     # -----------------------------------------------
287     st.markdown("<div class='footer'>GROQ PULSE • Powered by Groq</div>", unsafe_allow_html=True)
288     st.markdown("</div>", unsafe_allow_html=True)
289
```