

```

import streamlit as st
import google.generativeai as genai
from pypdf import PdfReader
import edge_tts
import asyncio
import os
import nest_asyncio

# --- تطبيق الحل لمشكلة asyncio مع Streamlit
nest_asyncio.apply()

# --- إعدادات الهوية البصرية (جامعة نورة) 1.
st.set_page_config(page_title="فرزعة، تسلفها", page_icon="🌸", layout="centered")

st.markdown("""
<style>
@import url('https://fonts.googleapis.com/css2?family=Tajawal:wght@400;700&di
html, body, [class*="css"] { font-family: 'Tajawal', sans-serif; direction: r
.stButton>button {
    width: 100%; border-radius: 25px; height: 3.5em;
    background-color: #8A1538; color: white; border: none; font-weight: bold;
}
.stButton>button:hover { background-color: #FCE4EC !important; color: #8A1538
h1, h2, h3 { color: #8A1538; text-align: center; }
.stAudio { margin-top: 20px; }
</style>
""", unsafe_allow_html=True)

# --- إعداد الـ 2. API ---
#  الطريقة الآمنة: استخدام st.secrets
# السطر التالي .streamlit/secrets.toml :أضيفي في ملف
# GEMINI_API_KEY = "مفتاحك هنا"
try:
    API_KEY = st.secrets["GEMINI_API_KEY"]
except Exception:
    # للاستخدام المحلي المؤقت فقط - لا ترعيه على GitHub!
    API_KEY = "ضعـيـ_مـفـتـاحـكـ_هـنـا"

genai.configure(api_key=API_KEY)

# --- دالة لتوليد الصوت الطبيعي ---
async def generate_natural_audio_async(text, output_file):
    voice = "ar-SA-ZariyahNeural"
    communicate = edge_tts.Communicate(text, voice, rate="+10%")

```

```

    await communicate.save(output_file)

def generate_audio(text, output_file):
    """ تشغيل الدالة بشكل آمن داخل asyncio.get_event_loop()
    loop = asyncio.get_event_loop()
    loop.run_until_complete(generate_natural_audio_async(text, output_file))

# --- واجهة المستخدم . 3 ---
st.markdown("<h1>✿ فزعة ، تسولفها رح محاضرتك كأنها سوال بين نورة ومنال</h1>", unsafe_allow_html=True)
st.markdown("<p style='text-align: center;'> ارفعي ملف المحاضرة</p> (PDF)", type="pdf")

uploaded_file = st.file_uploader("ارفعي ملف المحاضرة", type="pdf")

if uploaded_file:
    reader = PdfReader(uploaded_file)
    full_text = "".join([p.extract_text() for p in reader.pages if p.extract_text])

    if full_text:
        # ✅ تقليل النص لتجنب تجاوز حد الـ tokens
        full_text = full_text[:15000]

        st.success("تم رفع الملف! اختياري وش تبين تسمعين")
        col1, col2, col3 = st.columns(3)

        base_prompt = """
اديمي التالي إلى حوار "سكريبت" طويل ومفصل بين بنتين سعوديتين (نورة وفال)
- نورة: هي الدافورة اللي تشرح بذكاء وحماس -
- منال: هي اللي تسأل أسئلة ذكية وتبي تفهم التفاصيل -
- الأسلوب: سوال فجيدة عميقه ، ممتعة ، وبدون اختصار -
- اشرحي كل شيء في النص -
لا تكتبي (نورة:) و (فال:) في النص، اجعليه حواراً متصلًا كأنه جلسة تسجيل -
        {full_text}
"""

        final_prompt = ""
        if col1.button("سوال فجيدة"):
            final_prompt = base_prompt
        if col2.button("ترجمة وسؤال"):
            final_prompt += " ترجمي النص التالي للعربية ثم"
        if col3.button("English Session"):
            final_prompt = f"Create a deep-dive conversation between two students"

        if final_prompt:
            with st.spinner("نورة وفال قاعددين يجهزون السوال... لحظات"):
                try:
                    # 1. توليد الحوار من Gemini
                    model = genai.GenerativeModel('gemini-1.5-flash')

```

```
response = model.generate_content(final_prompt)
generated_script = response.text

# 2. تحويل الحوار لصوت طبيعي ✓ الطريقة الصحيحة
audio_file = "faza_audio.mp3"
generate_audio(generated_script, audio_file)

# 3. عرض النتيجة
st.markdown("---")
st.markdown("### 🎧: جا هز ! اسمعي الفزعة")
st.audio(audio_file)

with open(audio_file, "rb") as f:
    st.download_button("تحميل المحادثة MP3", f, file_name="no

except Exception as e:
    st.error(f"حدث خطأ: {e}")

else:
    st.error("الملف غير قابل للقراءة")
```