

```

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

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

# --- إعدادات الهوية البصرية (جامعة نورة) ---
st.set_page_config(page_title="فزة، تسولفها", 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)

# --- API إعدادات ---
# ☒ st.secrets الطريقة الآمنة: استخدام
# السطر التالي streamlit/secrets.toml أضيف في ملف:
# GEMINI_API_KEY = "مفتاح هنا"
try:
    API_KEY = st.secrets["GEMINI_API_KEY"]
except Exception:
    # GitHub! للاستخدام المحلي المؤقت فقط - لا ترفعيه على
    API_KEY = "ضعي_مفتاح_هنا"

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

# --- واجهة المستخدم 3. ---
st.markdown("<h1>🌸 تسولفها </h1>", unsafe_allow_html=True)
st.markdown("<p style='text-align: center;'>رح محاضرتك كأنها سوالف بين نورة ومنال")

uploaded_file = st.file_uploader("ارفعي ملف المحاضرة (PDF)", 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:
        # ✅ tokens تقليص النص لتجنب تجاوز حد الـ
        full_text = full_text[:15000]

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

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

        final_prompt = ""
        if col1.button("سوالف نجدية 🇸🇪"):
            final_prompt = base_prompt
        if col2.button("ترجمة وسوالف 🇺🇸🇸🇪"):
            final_prompt = " ترجمي النص التالي للعربي ثم " + base_prompt
        if col3.button("🇬🇧 English Session"):
            final_prompt = f"Create a deep-dive conversation between two students

        if final_prompt:
            with st.spinner("نورة ومنال قاعدين يجهزون السوالف... لحظات"):
                try:
                    # توليد الحوار من 1. 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("### 🎧 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.")

```