Smart resume generator Sample code:

```
import streamlit as st

import google.generativeai as genai

# Configure the API key for the Gemini API

api_key = "AIzaSyB3j11kVF4FGQ1vmkpkpXSdmredoSSY9GI"  # Replace with your actual API key

genai.configure(api_key=api_key)

# Configure the model generation settings

generation_config = {

    "temperature": 1,

    "top_p": 0.95,

    "top_k": 64,

    "max_output_tokens": 1024,

    "response_mime_type": "text/plain",

}

# Function to generate resume using Google Generative AI API

def generate_resume(name, job_title):

    # Create the generative model instance

    model = genai.GenerativeModel(

        model_name="gemini-1.5-pro",

        generation_config=generation_config,

    )

    # Construct context dynamically based on input
    context = f"Name: {name}\nJob Title: {job_title}\nWrite a resume
based on the above data."
    # Start the chat session with the generative model
    chat_session = model.start_chat(
        history=[
            {
                "role": "user",
                "parts": [context],
            }
        ]
    )
    # Get the response from the model
    response = chat_session.send_message(context)
    # Extract the resume content
    text = response.text  # Simplified access assuming the API returns a
text attribute
    return text
# Function to clean the resume text
def clean_resume_text(text):
    # Replace placeholder text within square brackets
    cleaned_text = text.replace("[Add Email Address]", "[Your Email
Address]")
    cleaned_text = cleaned_text.replace("[Add Phone Number]",
"[Your Phone Number]"
```

```
 cleaned_text = cleaned_text.replace("[Add LinkedIn Profile URL (optional)]")
```