```python
import requests
import json
import base64
from pathlib import Path

def convert_ad_to_speech(api_key, url, output_filename="ad_audio.wav"):
    """
    Converts ad copy to a WAV audio file using IBM Watson Text-to-Speech API
    without SSML, relying on the voice's natural delivery.

    Args:
        api_key (str): Your IBM Watson API key.
        url (str): The IBM Watson service URL.
        output_filename (str, optional): The name of the output audio file.
                                         Defaults to "ad_audio.wav".
    """
    # The SSML text is hardcoded to ensure consistent ad delivery.
    # It includes tags for pauses (<break>), emphasis (<emphasis>), and
    # expressive style (<express-as>) to make the voice sound natural.
    ssml_text = """
<speak>
    You’re One PDF Away From Changing Everything

    Tired of searching endlessly online, only to find outdated, half-baked content?

    This PDF is not just another download. It’s a shortcut to what you actually need.

    Why This PDF?

    ✅ Proven strategies you won’t find for free
    ✅ Actionable steps – no fluff, just results
    ✅ Real-world value in every single page
    ✅ Exclusive insights from [Your Niche/Field] experts

    This is the exact resource that [thousands/yourself] used to [solve a problem/gain success]—and now, it’s your turn.

    Limited-time launch offer: Get it now before the price doubles.

    🎁 Bonus inside for first 100 downloads.

    💬 Real buyers say:

    "I wish I had this sooner."
    "It’s worth 10x the price."

    🛒 Ready to stop guessing and start winning?

    Click Buy Now. You won’t regret it.

    Instant access. No waiting. No BS.
</speak>
"""
    # Using an Expressive Neural Voice for speaking styles and a professional tone
    voice = "en-US_MichaelExpressive"
    full_url = f"{url}/v1/synthesize?voice={voice}"

    # Prepare authentication (Basic Auth with API Key)
    auth_string = f"apikey:{api_key}"
    encoded_auth = base64.b64encode(auth_string.encode("utf-8")).decode("utf-8")

    # Headers for the API call
    headers = {
        "Content-Type": "application/json",
        "Accept": "audio/wav",
        "Authorization": f"Basic {encoded_auth}"
    }

    # Create the JSON payload
    payload = {
        "text": ssml_text
    }

    print(f"Synthesizing audio for ad text...")
    try:
        response = requests.post(full_url, headers=headers, data=json.dumps(payload), stream=True)
        response.raise_for_status()  # Raise an HTTPError for bad responses (4xx or 5xx)

        # Save the audio file
        with open(output_filename, 'wb') as audio_file:
            for chunk in response.iter_content(chunk_size=1024):
                audio_file.write(chunk)

        print(f"Audio saved to '{output
    except requests.exceptions.Request
        print(f"Error during API call: {e}")
        if response is not None:
            print(f"Status Code: {response.status_code}")
```

What can I help you build?

```
            print(f"Response Body: {response.text}")

# --- How to run this code in a Python environment like Google Colab ---
if __name__ == '__main__':
    # You must provide your own API key and URL here.
    # Replace "YOUR_API_KEY" and "YOUR_URL" with your actual credentials.
    # Note: The script is already configured with your API key and URL for this example.
    api_key_placeholder = "UGZoVQeeMoaHOZ1wc-a8bSGonViCef6sxSFMCDneI7ES"
    url_placeholder = "https://api.au-syd.text-to-speech.watson.cloud.ibm.com/instances/8d847898-36ee-40ae-a981-0ce733e2b8e5"

    convert_ad_to_speech(api_key=api_key_placeholder, url=url_placeholder, output_filename="instagram_ad_natural.wav")
```

```
Synthesizing audio for ad text...
Audio saved to 'instagram_ad_natural.wav'
```

Start coding or generate with AI.