# TEAM APPLE

## 1. Problem Statement (Elaborated)

In today's fast-paced digital landscape, businesses, content creators, and marketing teams
often struggle to convert static scripts into engaging motion videos. Traditional video production
is time-consuming, expensive, and requires technical expertise in animation, editing, and
design. Many users lack the skills or resources to bring their ideas to life in a visually compelling
way.
The challenge is to develop a system that automatically transforms written scripts into dynamic
motion videos with animations, voiceovers, and background music. The solution should be
intuitive, require minimal user input, and generate professional-quality videos that enhance
engagement across various platforms.

## User Story

**Title:** Automating Motion Video Creation from Scripts
**As a** content creator,
**I want to** generate a professional motion video from my provided script automatically,
**So that** I can save time, reduce production costs, and create engaging visual content without
requiring video editing skills.
**Acceptance Criteria:**

- Users should be able to input a script through a simple interface.
- The system should analyze the script and break it into scenes.
- AI should generate relevant animations, transitions, and background elements.
-

The system should provide auto-generated voiceovers based on the script.

- Users should have options to customize visuals, voiceover styles, and background music.
- The final video should be exportable in various formats for social media and professional use.

CODE:

```
import os
import random
import torch
import moviepy.editor as mp
from gtts import gTTS
from PIL import Image
import requests
from io import BytesIO
from TTS.api import TTS

class MotionVideoGenerator:
    def __init__(self, voice_model="tts_models/multilingual/multi-dataset/xtts_v2"):
        self.device = "cuda" if torch.cuda.is_available() else "cpu"
        self.tts = TTS(voice_model).to(self.device)

    def generate_voiceover(self, script, output_file="voiceover.mp3"):
        tts = gTTS(text=script, lang="en", slow=False)
        tts.save(output_file)
        return output_file

    def get_images(self, keywords, num_images=5):
        image_urls = [
            f"https://source.unsplash.com/800x450/?{keyword}" for keyword in
keywords[:num_images]
        ]
        images = []
        for url in image_urls:
            response = requests.get(url)
            img = Image.open(BytesIO(response.content))
            img_path = f"scene_{random.randint(1000,9999)}.jpg"
```

```
            img.save(img_path)
            images.append(img_path)
        return images

    def create_video(self, script, output_video="final_video.mp4",
background_music="music.mp3"):
        lines = script.split(". ")
        keywords = [word for line in lines for word in line.split() if len(word) > 3]
        images = self.get_images(keywords)
        voiceover = self.generate_voiceover(script)
        clips = []
        for img in images:
            img_clip = mp.ImageClip(img).set_duration(3).fadein(0.5).fadeout(0.5)
            clips.append(img_clip)
        video = mp.concatenate_videoclips(clips, method="compose")
        audio = mp.AudioFileClip(voiceover)
        if os.path.exists(background_music):
            bg_music = mp.AudioFileClip(background_music).subclip(0, audio.duration)
            final_audio = mp.CompositeAudioClip([audio, bg_music.volumex(0.3)])
        else:
            final_audio = audio
        video = video.set_audio(final_audio)
        video.write_videofile(output_video, fps=24, codec="libx264", audio_codec="aac")
        return output_video

if __name__ == "__main__":
    script = """The future of technology is evolving rapidly. Artificial Intelligence is transforming
industries.
    Businesses are adapting to new digital trends. The importance of automation has never been
greater."""
    mv = MotionVideoGenerator()
    final_video = mv.create_video(script)
```