

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
content="width=device-width, initial-
scale=1.0">
  <title>CineMorph AI – Long Form Video
Generator</title>
  <script src="https://
cdn.tailwindcss.com"></script>
  <link href="https://cdnjs.cloudflare.com/
ajax/libs/font-awesome/6.4.0/css/
all.min.css" rel="stylesheet">
  <link href="https://
fonts.googleapis.com/css2?
family=Outfit:wght@300;400;500;600;700
&family=Rajdhani:wght@500;600;700&dis
play=swap" rel="stylesheet">
  <style>
    :root {
```

```
--primary: #6366f1;  
--secondary: #ec4899;  
--bg-dark: #09090b;  
--bg-card: #18181b;  
}  
  
  
body {  
  font-family: 'Outfit', sans-serif;  
  background-color: var(--bg-dark);  
  color: white;  
  overflow: hidden; /* App-like feel */  
}  
  
  
h1, h2, h3, .logo-font {  
  font-family: 'Rajdhani', sans-serif;  
}  
  
  
/* Glassmorphism & Glows */  
.glass-panel {  
  background: rgba(24, 24, 27, 0.7);  
  backdrop-filter: blur(12px);
```

```
    border: 1px solid rgba(255, 255,  
255, 0.08);  
}  
  
/* Custom Range Slider */
```

```
.neon-text {  
    text-shadow: 0 0 10px rgba(99,  
102, 241, 0.5);  
}
```

```
.gradient-border {  
    position: relative;  
    background: linear-gradient(var(--  
bg-card), var(--bg-card)) padding-box,  
                linear-gradient(45deg, var(--  
primary), var(--secondary)) border-box;  
    border: 1px solid transparent;  
    border-radius: 0.75rem;  
}
```

```
input[type=range] {
```

```
-webkit-appearance: none;  
background: transparent;  
}  
  
input[type=range]::-webkit-slider-  
thumb {  
    -webkit-appearance: none;  
    height: 16px;  
    width: 16px;  
    border-radius: 50%;  
    background: #fff;  
    cursor: pointer;  
    margin-top: -6px;  
    box-shadow: 0 0 10px  
rgba(255,255,255,0.5);  
}  
  
input[type=range]::-webkit-slider-  
runnable-track {  
    width: 100%;  
    height: 4px;  
    background: #3f3f46;  
    border-radius: 2px;
```

}

```
/* Animation for loading */
@keyframes scanline {
    0% { transform: translateY(-100%); }
    100% { transform:
translateY(100%); }
}

.scanning-bar {
    position: absolute;
    left: 0;
    right: 0;
    height: 2px;
    background: linear-gradient(to
right, transparent, #6366f1, transparent);
    animation: scanline 2s linear
infinite;
    z-index: 20;
}

/* Scrollbars */
```

```
    ::-webkit-scrollbar { width: 6px; }
    ::-webkit-scrollbar-track {
background: #09090b; }
        ::-webkit-scrollbar-thumb {
background: #3f3f46; border-radius: 3px; }
        ::-webkit-scrollbar-thumb:hover {
background: #52525b; }
```

```
.video-overlay {
    background: linear-gradient(0deg,
rgba(0,0,0,0.8) 0%, transparent 40%);
}
```

```
</style>
```

```
</head>
```

```
<body class="h-screen flex flex-col">
```

```
<!-- Top Navigation -->
<nav class="h-16 border-b border-
white/10 flex items-center justify-
between px-6 bg-black/50 z-50">
    <div class="flex items-center gap-3">
```

```
<div class="w-8 h-8 rounded bg-gradient-to-br from-indigo-500 to-pink-500 flex items-center justify-center">  
    <i class="fa-solid fa-infinity text-white text-sm"></i>  
    </div>  
    <span class="text-2xl font-bold logo-font tracking-wide">Cine<span  
        class="text-indigo-500">Morph</span></span>  
    <span class="text-[10px] bg-indigo-500/20 text-indigo-400 px-2  
        py-0.5 rounded border border-indigo-500/30 ml-2">PRO BETA</span>  
    </div>
```

```
<div class="flex items-center gap-6">  
    <div class="flex items-center  
        gap-2 text-sm text-gray-400">  
        <i class="fa-solid fa-coins text-
```

```
yellow-500"></i>
    <span class="text-white">∞
Credits</span> (Free Mode)
</div>
<div class="h-8 w-8 rounded-full
bg-zinc-800 border border-white/10 flex
items-center justify-center cursor-
pointer hover:bg-zinc-700 transition">
    <i class="fa-regular fa-bell"></i>
</div>
<div class="h-8 w-8 rounded-full
bg-gradient-to-r from-purple-500 to-
indigo-500 cursor-pointer"></div>
</div>
</nav>
```

```
<!-- Main Workspace -->
<div class="flex-1 flex overflow-
hidden">
<!-- LEFT SIDEBAR: Controls -->
```

```
<aside class="w-80 bg-[#0c0c0e]  
border-r border-white/10 flex flex-col  
overflow-y-auto z-10">  
  <div class="p-5 space-y-6">  
  
    <!-- Input Mode Switch -->  
    <div class="bg-zinc-900/50 p-1  
rounded-lg flex border border-white/5">  
      <button  
        onclick="setMode('text')" id="btn-mode-  
text" class="flex-1 py-2 text-xs font-  
semibold rounded bg-indigo-600 text-  
white shadow-lg transition-all">  
        TEXT TO VIDEO  
      </button>  
      <button  
        onclick="setMode('image')" id="btn-  
mode-image" class="flex-1 py-2 text-xs  
font-semibold rounded text-zinc-400  
hover:text-white transition-all">  
        IMAGE TO VIDEO  
      </button>  
    </div>  
  </div>  
</aside>
```

```
</button>
</div>

<!-- Prompt Input -->
<div id="text-input-section"
class="space-y-2">
    <div class="flex justify-
between items-center">
        <label class="text-xs font-
semibold text-gray-400 uppercase
tracking-wider">Prompt</label>
        <button class="text-xs text-
indigo-400 hover:text-indigo-300"
onclick="enhancePrompt()"><i class="fa-
solid fa-wand-magic-sparkles mr-1"></
i>Enhance</button>
    </div>
    <textarea id="prompt-area"
class="w-full bg-zinc-900 border border-
white/10 rounded-xl p-3 text-sm text-
gray-200 focus:outline-none"
```

```
focus:border-indigo-500/50 focus:ring-1  
focus:ring-indigo-500/50 resize-none  
h-32 transition-all placeholder-gray-600"  
placeholder="Describe your scene in  
detail... e.g. An astronaut walking through  
a bioluminescent forest, cinematic  
lighting, 8k, smooth motion..."></textarea>  
</div>
```

```
<!-- Image Upload (Hidden by  
default) -->
```

```
<div id="image-input-section"  
class="space-y-2 hidden">  
    <label class="text-xs font-  
semibold text-gray-400 uppercase  
tracking-wider">Source Image</label>  
    <div class="border-2 border-  
dashed border-zinc-700 rounded-xl p-6  
text-center hover:bg-zinc-800/50  
hover:border-indigo-500/50 transition-all  
cursor-pointer relative overflow-hidden
```

group">

```
    <input type="file" id="image-
upload" accept="image/*" class="absolute
inset-0 opacity-0 cursor-pointer z-20"
onchange="handleImageUpload(event)">
        <div id="upload-placeholder"
class="z-10">
            <i class="fa-solid fa-
cloud-arrow-up text-2xl text-gray-500
mb-2 group-hover:text-indigo-400
transition-colors"></i>
            <p class="text-xs text-
gray-500">Click or Drag Image</p>
        </div>
        <img id="preview-img"
class="absolute inset-0 w-full h-full
object-cover hidden z-10" />
        <button id="remove-img-
btn" onclick="removeImage(event)"
class="absolute top-2 right-2 z-30 bg-
black/60 rounded-full p-1 w-6 h-6 flex"
```

```
items-center justify-center text-xs  
hover:bg-red-500/80 hidden text-white">  
    <i class="fa-solid fa-  
times"></i>  
    </button>  
  </div>  
</div>  
  
  <!-- Settings -->  
  <div class="space-y-4 pt-4  
border-t border-white/5">  
    <label class="text-xs font-  
semibold text-gray-400 uppercase  
tracking-wider flex items-center gap-2">  
      <i class="fa-solid fa-  
sliders"></i> Configuration  
      </label>  
  
    <!-- Aspect Ratio -->  
    <div class="grid grid-cols-3  
gap-2">
```

```
<button class="aspect-btn  
bg-zinc-800 border border-white/10  
rounded py-2 text-xs text-gray-300  
hover:bg-zinc-700 active-aspect ring-1  
ring-indigo-500 text-white bg-zinc-700"  
onclick="setRatio(this, '16:9')">16:9</  
button>
```

```
<button class="aspect-btn  
bg-zinc-800 border border-white/10  
rounded py-2 text-xs text-gray-300  
hover:bg-zinc-700"  
onclick="setRatio(this, '9:16')">9:16</  
button>
```

```
<button class="aspect-btn  
bg-zinc-800 border border-white/10  
rounded py-2 text-xs text-gray-300  
hover:bg-zinc-700"  
onclick="setRatio(this, '1:1')">1:1</button>  
</div>
```

```
<!-- Duration -->
```

```
<div>
    <div class="flex justify-
between text-xs text-gray-400 mb-1">
        <span>Duration</span>
        <span id="duration-val"
class="text-indigo-400">5s</span>
    </div>
    <input type="range" min="3"
max="60" value="5" class="w-full"
oninput="updateDuration(this.value)">
        <div class="flex justify-
between text-[10px] text-gray-600 mt-1">
            <span>3s</span>
            <span>30s</span>
            <span>60s</span>
        </div>
    </div>
```

```
<!-- Motion Strength -->
<div>
    <div class="flex justify-
```

```
between text-xs text-gray-400 mb-1">
    <span>Motion Strength</span>
        <span id="motion-val">Medium</span>
        </div>
        <input type="range" min="1" max="10" value="5" class="w-full" oninput="document.getElementById('motion-val').innerText = this.value">
    </div>
```

```
<!-- Negative Prompt -->
<div>
    <span class="text-xs text-gray-500 cursor-pointer hover:text-gray-300 transition"
        onclick="document.getElementById('negative-prompt').classList.toggle('hidden')">
        Advanced: Negative
        Prompt <i class="fa-solid fa-chevron-
```

```
down ml-1 text-[10px]"></i>
        </span>
        <textarea id="neg-prompt-
box" class="hidden w-full mt-2 bg-
zinc-900 border border-white/10
rounded-lg p-2 text-xs text-gray-400
focus:outline-none resize-none h-16"
placeholder="blurry, distorted, low quality,
watermark..."></textarea>
```

```
        </div>
```

```
    </div>
```

```
<!-- Generate Button -->
```

```
    <button id="generate-btn"
onclick="startGeneration()" class="w-full
py-4 rounded-xl bg-gradient-to-r from-
indigo-600 to-purple-600 hover:from-
indigo-500 hover:to-purple-500 text-
white font-bold text-sm tracking-widest
uppercase transition-all shadow-
[0_0_20px_rgba(99,102,241,0.3)]
```

```
  hover:shadow-[0_0_30px_rgba(99,102,241,0.5)] relative  
  overflow-hidden group">  
    <span class="relative z-10 flex  
  items-center justify-center gap-2">  
      <i class="fa-solid fa-film"></i> Generate Video  
    </span>  
    <div class="absolute inset-0  
  bg-white/20 translate-y-full group-  
  hover:translate-y-0 transition-transform  
  duration-300"></div>  
  </button>  
  
</div>  
</aside>  
  
<!-- CENTER: Canvas / Viewer -->  
<main class="flex-1 bg-[#050505]  
  relative flex flex-col">
```

```
<!-- Canvas Toolbar -->
<div class="h-12 border-b border-white/5 flex items-center justify-between px-4 bg-zinc-900/30">
  <div class="text-xs text-gray-500 flex gap-2">
    <span class="hover:text-white cursor-pointer transition">Project:
    Untitled_01</span>
    <span>/</span>
    <span id="res-display"
    class="hover:text-white cursor-pointer transition">1080p</span>
  </div>
  <div class="flex gap-3">
    <button class="text-gray-400
    hover:text-white transition"><i class="fa-solid fa-undo"></i></button>
    <button class="text-gray-400
    hover:text-white transition"><i class="fa-solid fa-redo"></i></button>
```

```
<div class="w-px h-4 bg-
white/10"></div>
    <button class="text-gray-400
hover:text-white transition"><i class="fa-
solid fa-expand"></i></button>
</div>
</div>
```

```
<!-- Main Viewing Area -->
<div class="flex-1 relative flex
items-center justify-center p-8 bg-
[radial-
gradient(#1f1f22_1px,transparent_1px)]
[background-size:16px_16px]">
```

```
<!-- Empty State -->
<div id="empty-state"
class="text-center opacity-40 select-
none pointer-events-none">
    <i class="fa-solid fa-
clapperboard text-6xl text-gray-600
```

mb-4"></i>

<h2 class="text-xl font-semibold text-gray-400">Ready to Create</h2>

<p class="text-sm text-gray-600 mt-2">Configure settings and hit Generate to start</p>

</div>

<!-- Processing State -->

<div id="processing-state" class="hidden absolute inset-0 z-20 bg-black/80 backdrop-blur-sm flex items-center justify-center">

<div class="w-96 p-6 rounded-2xl glass-panel border border-indigo-500/30 text-center relative overflow-hidden">

<div class="scanning-bar"></div>

<div>

<h3 class="text-lg font-bold

text-white mb-1 logo-font tracking-wide">GENERATING...</h3>

<p id="process-step" class="text-xs text-indigo-300 mb-6 font-mono">Initializing Neural Engine</p>

<!-- Progress Bar -->

<div class="h-1.5 w-full bg-zinc-800 rounded-full overflow-hidden mb-2">

<div id="progress-bar" class="h-full bg-gradient-to-r from-indigo-500 to-pink-500 w-0 transition-all duration-300"></div>

</div>

<div class="flex justify-between text-[10px] text-gray-500 font-mono">

<span>00:00</span>

<span id="percent-text">0%</span>

```
<span id="est-time">EST:  
10s</span>  
</div>
```

```
<!-- Terminal Output Effect
```

```
-->
```

```
<div class="mt-4 h-24 bg-  
black/50 rounded p-2 text-left overflow-  
hidden font-mono text-[10px] text-  
green-500/80 leading-tight border  
border-white/5" id="terminal-log">  
    > System ready...<br>  
</div>  
</div>  
</div>
```

```
<!-- Result Video Player -->  
<div id="video-container"  
class="hidden relative w-full h-full max-  
w-4xl max-h-[80vh] bg-black rounded-lg  
shadow-2xl overflow-hidden border
```

```
border-white/10 group">
    <video id="output-video"
class="w-full h-full object-contain" loop
playsinline>
    <source src="" type="video/
mp4">
</video>

<!-- Controls Overlay -->
<div class="absolute inset-0
bg-black/40 flex items-center justify-
center opacity-0 group-
hover:opacity-100 transition-opacity
duration-300" id="play-overlay"
onclick="togglePlay()">
    <div class="w-16 h-16
rounded-full bg-white/10 backdrop-blur-
md flex items-center justify-center
hover:bg-white/20 cursor-pointer
transition transform hover:scale-110">
        <i id="play-icon"
```

```
        class="fa-solid fa-play text-2xl text-white ml-1">></i>
      </div>
    </div>

    <!-- Bottom Info -->
    <div class="absolute bottom-0 left-0 right-0 p-4 video-overlay translate-y-full group-hover:translate-y-0 transition-transform duration-300">
      <div class="flex justify-between items-end">
        <div>
          <h3 class="text-sm font-semibold text-white">Generated Sequence_001</h3>
          <p class="text-xs text-gray-400">HQ Render • <span id="final-dur">5s</span></p>
        </div>
        <button>
```

```
    onclick="downloadVideo()" class="bg-white text-black px-4 py-1.5 rounded text-xs font-bold hover:bg-gray-200 transition">  
        <i class="fa-solid fa-download mr-1"></i> Download MP4  
    </button>  
  </div>  
</div>  
  
</div>  
  
<!-- Bottom Timeline (Visual only<br/>for "Long Video" feel) -->  
  <div class="h-32 bg-[#0c0c0e]  
border-t border-white/10 flex flex-col">  
    <div class="h-8 border-b border-white/5 flex items-center px-4 justify-between bg-zinc-900/50">  
      <span class="text-[10px] font-
```

**bold text-gray-500 uppercase tracking-widest">Timeline</span>**

<div class="flex gap-2 text-gray-500 text-xs">

<i class="fa-solid fa-minus cursor-pointer hover:text-white"></i>

<i class="fa-solid fa-plus cursor-pointer hover:text-white"></i>

</div>

</div>

<div class="flex-1 p-2 overflow-x-auto whitespace nowrap" id="timeline-track">

<!-- Timeline segments added here dynamically -->

<div class="inline-block h-full w-full bg-zinc-800/30 rounded border border-white/5 relative overflow-hidden flex items-center justify-center text-gray-600 text-xs font-mono">

NO CLIPS

```
    </div>
    </div>
</div>
```

```
</main>
```

```
</div>
```

```
<script>
```

```
  // --- Logic & State ---
```

```
  let currentMode = 'text';
```

```
  let isGenerating = false;
```

```
  let videoDuration = 5;
```

```
// Stock videos to simulate generation  
results
```

```
const stockVideos = [
```

```
  "https://assets.mixkit.co/videos/  
preview/mixkit-stars-in-space-  
background-1610-large.mp4", // Space
```

```
  "https://assets.mixkit.co/videos/
```

```
preview/mixkit-white-sand-beach-
background-1564-large.mp4", // Beach
    "https://assets.mixkit.co/videos/
preview/mixkit-ink-swirling-in-
water-2022-large.mp4", // Abstract Ink
    "https://assets.mixkit.co/videos/
preview/mixkit-going-down-a-curved-
highway-through-a-mountain-
range-41576-large.mp4" // Road
];

```

### // ---- UI Functions ----

```
function setMode(mode) {
    currentMode = mode;
    const textBtn =
document.getElementById('btn-mode-
text');
    const imgBtn =
document.getElementById('btn-mode-
image');
```



```
    py-2 text-xs font-semibold rounded bg-pink-600 text-white shadow-lg transition-all";
```

```
        textBtn.className = "flex-1  
py-2 text-xs font-semibold rounded text-zinc-400 hover:text-white transition-all";
```

```
textSection.classList.add('hidden');
```

```
imgSection.classList.remove('hidden');
```

```
    }  
}
```

```
function setRatio(btn, ratio) {
```

```
document.querySelectorAll('.aspect-btn').forEach(b => {  
    b.className = "aspect-btn bg-zinc-800 border border-white/10 rounded  
py-2 text-xs text-gray-300 hover:bg-zinc-700";
```

```
});  
  btn.className = "aspect-btn bg-zinc-800 border border-white/10 rounded py-2 text-xs text-gray-300 hover:bg-zinc-700 active-aspect ring-1 ring-indigo-500 text-white bg-zinc-700";  
  document.getElementById('res-display').innerText = ratio === '16:9' ? '1080p' : (ratio === '9:16' ? 'Vertical' : 'Square');  
}
```

```
function updateDuration(val) {  
  videoDuration = val;  
  
  document.getElementById('duration-val').innerText = val + 's';  
  // Scale generation time slightly  
  // based on duration requested  
  document.getElementById('est-time').innerText = `EST: ${Math.min(15,
```

```
Math.floor(val/2)))s`;
```

```
}
```

```
function enhancePrompt() {
```

```
    const area =
```

```
document.getElementById('prompt-area');
```

```
    if(area.value.trim() === "") {
```

```
        area.value = "Cinematic
```

```
masterpiece, highly detailed, 8k  
resolution, photorealistic, dramatic  
lighting, trending on artstation...";
```

```
    } else {
```

```
        area.value += ", cinematic
```

```
lighting, hyper-realistic, 8k, unreal engine  
5 render, smooth motion";
```

```
}
```

```
}
```

```
// --- Image Upload Handling ----
```

```
function handleImageUpload(event) {
```

```
    const file = event.target.files[0];
```

```
if(file) {  
    const reader = new FileReader();  
    reader.onload = function(e) {  
        const img =  
document.getElementById('preview-img');  
        img.src = e.target.result;  
        img.classList.remove('hidden');  
  
document.getElementById('upload-  
placeholder').classList.add('opacity-0');  
  
document.getElementById('remove-img-  
btn').classList.remove('hidden');  
    };  
    reader.readAsDataURL(file);  
}  
}  
  
function removeImage(e) {  
    e.stopPropagation(); // prevent  
opening file dialog
```

```
document.getElementById('image-
upload').value = "";

document.getElementById('preview-
img').classList.add('hidden');

document.getElementById('preview-
img').src = "";

    document.getElementById('upload-
placeholder').classList.remove('opacity-0')
;

    document.getElementById('remove-
img-btn').classList.add('hidden');
}
```

// --- Generation Logic ---

```
function startGeneration() {
    if(isGenerating) return;
```

// Simple validation

```
if(currentMode === 'text' &&
document.getElementById('prompt-
area').value.trim() === "") {
    alert("Please enter a text
prompt.");
    return;
}
if(currentMode === 'image' &&
document.getElementById('image-
upload').files.length === 0) {
    alert("Please upload a source
image.");
    return;
}

isGenerating = true;

// UI Update
document.getElementById('empty-
state').classList.add('hidden');
document.getElementById('video-
```

```
container').classList.add('hidden');
```

```
document.getElementById('processing-state').classList.remove('hidden');
```

```
// Terminal Logs
```

```
const log =
```

```
document.getElementById('terminal-log');
```

```
log.innerHTML = "> Job
```

```
queued...<br>> Allocation GPU Node
```

```
[A100]...<br>";
```

```
const steps = [
```

```
 { pct: 10, msg: "Parsing Prompt..." }
```

```
,
```

```
 { pct: 30, msg: "Diffusing Noise..." }
```

```
,
```

```
 { pct: 50, msg: "Generating Keyframes..." },
```

```
 { pct: 70, msg: "Interpolating Motion..." },
```

```
        { pct: 90, msg: "Upscaling &  
Rendering..." },  
        { pct: 100, msg: "Finalizing..." }  
    ];
```

```
let stepIndex = 0;  
let currentPct = 0;
```

```
// Calculate simulation duration  
based on video length user chose
```

```
// Short videos gen fast, long  
videos gen slower to feel real
```

```
const simulationTime =  
Math.max(3000, videoDuration * 200);
```

```
const intervalTime =  
simulationTime / 100;
```

```
const timer = setInterval(() => {  
    currentPct++;
```

```
// Update Progress UI
```

```
document.getElementById('progress-bar').style.width = currentPct + "%";
```

```
document.getElementById('percent-text').innerText = currentPct + "%";
```

```
// Update steps
if(stepIndex < steps.length &&
currentPct >= steps[stepIndex].pct) {

document.getElementById('process-step').innerText = steps[stepIndex].msg;
log.innerHTML += `> ${steps[stepIndex].msg}<br>';
log.scrollTop = log.scrollHeight;
stepIndex++;
}
```

```
if(currentPct >= 100) {
    clearInterval(timer);
```

```
        finishGeneration();  
    }  
}, intervalTime);  
}  
}
```

```
function finishGeneration() {  
    isGenerating = false;  
  
    document.getElementById('processing-  
state').classList.add('hidden');  
    document.getElementById('video-  
container').classList.remove('hidden');  
  
    // Pick a random video to simulate  
    result  
    const randomVid =  
stockVideos[Math.floor(Math.random() *  
stockVideos.length)];  
    const videoEl =  
document.getElementById('output-video');  
    videoEl.src = randomVid;
```

```
videoEl.play();

document.getElementById('final-
dur').innerText = videoDuration + 's';

// Add to timeline
addToTimeline();

}

function addToTimeline() {
    const track =
document.getElementById('timeline-
track');

    // Remove "No clips" text if present
    if(track.innerText.includes("NO
CLIPS")) track.innerHTML = "";

    // Create clip element
    const clipWidth = Math.min(200,
videoDuration * 5); // Visual width based
on duration
```

```
const clip =  
document.createElement('div');  
clip.className = "inline-block  
h-24 rounded-lg bg-indigo-900/40  
border border-indigo-500/50 mr-2  
relative group overflow-hidden cursor-  
pointer hover:brightness-110 transition";  
clip.style.width = clipWidth + "px";  
clip.innerHTML = `  
    <div class="absolute top-1 left-2  
text-[10px] text-white font-mono  
z-10">CLIP_ $  
{Math.floor(Math.random()*1000)}</div>  
    <div class="absolute bottom-1  
right-2 text-[10px] text-gray-300 z-10">$  
{videoDuration}s</div>  
    <div class="absolute inset-0 flex  
items-center justify-center opacity-0  
group-hover:opacity-100 bg-black/40  
transition">  
        <i class="fa-solid fa-play text-
```

```
white"></i>
        </div>
        <!-- Fake waveform visual -->
        <div class="absolute bottom-0
left-0 right-0 h-8 opacity-30 flex items-
end gap-0.5 px-1">
    ${Array(20).fill(0).map(() =>
`<div class="w-1 bg-indigo-400"
style="height: ${Math.random()*100}%"></
div>`).join("")}
    </div>
`;
// Logic to play this specific clip
// (simulated by replaying current video)
clip.onclick = () => {
    const videoEl =
document.getElementById('output-video');
    videoEl.currentTime = 0;
    videoEl.play();
}
```

```
    track.appendChild(clip);
}
```

```
// ---- Video Player Controls ----
```

```
const videoEl =
```

```
document.getElementById('output-video');
```

```
const playIcon =
```

```
document.getElementById('play-icon');
```

```
function togglePlay() {
```

```
    if(videoEl.paused) {
```

```
        videoEl.play();
```

```
        playIcon.className = "fa-solid
```

```
fa-pause text-2xl text-white ml-0";
```

```
        document.getElementById('play-
```

```
overlay').style.opacity = "0";
```

```
    } else {
```

```
        videoEl.pause();
```

```
        playIcon.className = "fa-solid
```

```
fa-play text-2xl text-white ml-1";
```

```
        document.getElementById('play-
overlay').style.opacity = "1";
    }
}

videoEl.addEventListener('play', () => {
    document.getElementById('play-
overlay').classList.add('opacity-0');
    document.getElementById('play-
overlay').classList.remove('group-
hover:opacity-100');
});

videoEl.addEventListener('pause', () => {
    document.getElementById('play-
overlay').classList.remove('opacity-0');
    document.getElementById('play-
overlay').classList.add('group-
hover:opacity-100');
});
```

```
function downloadVideo() {  
    const link =  
document.createElement('a');  
    link.href = videoEl.src;  
    link.download = `CineMorph_Gen_$  
{Date.now()}.mp4`;  
    document.body.appendChild(link);  
    link.click();  
    document.body.removeChild(link);  
}  
</script>  
</body>  
</html>
```