BENITES IA · EFEITOS

25 efeitos de vídeo com o Claude

Prompts e instalação: Claude Code + HyperFrames, passo a passo, com os números reais dos Reels.

1. O que é e o que você precisa
2. Instalar a skill HyperFrames
3. Primeiro vídeo com a sua gravação
4. O recorte da pessoa
5. Os 25 efeitos, um prompt para cada
6. Erros comuns e como corrigir

01 Parede de manchetes · 02 Zoom pelo corte · 03 Vertigo · 04 Fundo trocado · 05 Celular na tela · 06 Zoom digital · 07 Recorte de papel · 08 Mundo de interface · 09 Órbita · 10 Quebra de quadro · 11 Rabisco vivo · 12 Texto adesivo · 13 Antes e depois · 14 Recuo revelador · 15 Letras espalhadas · 16 Balão de comentário · 17 Título cinético · 18 Letras com gravidade · 19 Zoom no rosto · 20 Barra de prompt · 21 Timeline de editor · 22 Conta-gotas + paleta · 23 Look de cinema · 24 Cartão que vira celular · 25 Lista lateral

Conferido em 24/10/2026 · https://benites.blendagency.com.br/tutoriais/efeitos

O que é e o que você precisa

Chegou pelo comentário **EFEITOS** (ou STUDIO)? Aqui estão os 25 efeitos de edição dos Reels do @benites.ia, cada um com um prompt pronto. Todos foram feitos com o **HyperFrames**, uma ferramenta gratuita e de código aberto (licença Apache 2.0) da HeyGen. Ela transforma uma página HTML com animações em vídeo, quadro a quadro. Quem escreveu esse HTML foi o **Claude Code**, a partir de pedidos em português. Você não precisa saber programar: descreve o efeito, o Claude escreve o código e você confere no navegador antes de exportar o MP4.

Sua gravação→Claude Code + skill HyperFrames→Prévia no navegador→MP4

Como foram feitos, de verdade: os efeitos vêm de dois projetos. O **"16 efeitos de vídeo que o Claude cria"** (efeitos 01 a 16) e o **"Esse vídeo foi editado inteiro pelo Claude"** (efeitos 17 a 25, mais as variantes do 13 e do 16). Nos dois, o apresentador é um avatar da HeyGen com a voz clonada do Alexandre. O vídeo foi acelerado para 1,25x e transcrito palavra por palavra. Com isso, o Claude escreveu um gerador (`build.py`) que monta as cenas e dispara cada efeito na palavra certa da fala. Nada saiu pronto de um único pedido: houve várias rodadas de ajuste. Os prompts deste guia trazem os números reais desses projetos, para você começar de um ponto que já funcionou.

- **Claude Code** no computador (aplicativo do Claude, aba Code, ou no Terminal). Nunca abriu? Siga primeiro o guia Seu primeiro vídeo com IA.
- **Node.js 22 ou mais novo** e **FFmpeg**. São os dois requisitos do HyperFrames, e o passo 2 mostra como conferir.
- **Uma gravação sua**, em pé (9:16), de preferência com câmera parada e fundo que contraste com você. Pode ser do celular. O avatar da HeyGen é **opcional**.
- **Tempo:** cerca de 30 minutos para instalar e ver a primeira prévia. Cada efeito leva de uma a três rodadas de ajuste.

Custos: o HyperFrames é gratuito e renderiza no seu computador. O Claude Code exige plano pago do Claude ou créditos de API, e vídeos com muitos efeitos consomem bastante uso. A HeyGen (avatar e voz) cobra créditos. Confira os preços atuais nos sites oficiais antes de começar.

Instale a skill HyperFrames, passo a passo

A **skill** é um pacote de instruções que ensina o Claude a escrever vídeos no formato do HyperFrames. Os comandos abaixo vão no **Terminal** (Mac: Aplicativos → Utilitários → Terminal; Windows: PowerShell). A conversa com o Claude vem depois.

Passo 1: se o `node -v` der erro ou mostrar versão menor que 22, instale a versão LTS em nodejs.org e abra um Terminal novo.

1. Confira o Node.js (precisa mostrar v22 ou maior)

```
node -v
```

2. Confira o FFmpeg (deve mostrar a versão)

```
ffmpeg -version
```

3. Sem FFmpeg no Mac? Instale com o Homebrew

```
brew install ffmpeg
```

4. Instale as skills (escolha o grupo “Core Skills”)

```
npx skills add heygen-com/hyperframes
```

Alternativa: instalar todas as 7 skills sem perguntas

```
npx skills add heygen-com/hyperframes --all
```

5. Confira a instalação

```
npx hyperframes skills check
npx hyperframes doctor
```

6. Abra uma sessão nova do Claude Code e confirme

```
Verifique se as skills do HyperFrames estão disponíveis nesta sessão (hyperframes, hyperframes-cli, hyperframes-registry e
media-use). Rode npx hyperframes doctor e me explique em português simples o que está faltando no meu computador e como
instalar. Não crie nenhum projeto ainda.
```

- **Passo 3, Windows:** o comando `brew` é só do Mac. No Windows, baixe o FFmpeg pelos links de ffmpeg.org ou peça ao Claude: “instale o FFmpeg no meu Windows e coloque no PATH”.
- **Passo 4:** o instalador abre uma lista em que nada vem marcado. Selecione o grupo **Core Skills**, que traz as quatro skills principais, e escolha o Claude Code quando ele perguntar o agente. O `--all` instala as 7 skills publicadas em todos os agentes compatíveis, sem perguntar nada.
- **Passo 5:** o `skills check` avisa se alguma skill está desatualizada. O `doctor` mostra Node, FFmpeg e Chrome com ✓ ou ✗. Itens marcados como *optional* (whisper-cpp, Kokoro, MusicGen) não impedem o primeiro vídeo.
- **Passo 6:** uma skill só aparece nas sessões abertas depois da instalação. Feche e abra o Claude Code.

Outro caminho, só para o Claude Code: o README oficial também oferece o HyperFrames como plugin. Nesse caso a skill é chamada como `/hyperframes:hyperframes`, e as atualizações passam a ser feitas pelo menu `/plugin`.

```
claude plugin marketplace add heygen-com/hyperframes
claude plugin install hyperframes@hyperframes
```

Use só um dos caminhos (skills ou plugin) para não ter duas cópias. Fontes: README do repositório heygen-com/hyperframes e a ajuda da CLI 0.8.185, conferidas em 24/10/2026.

Primeiro vídeo: prepare o projeto com a sua gravação

Crie a pasta do projeto no Terminal. O `init` monta a estrutura básica: `index.html`, a pasta `assets/`, um `CLAUDE.md` com as regras para o agente e os atalhos `npm run dev`, `check` e `render`.

1. Crie o projeto (formato vertical 1080x1920)

```
npx hyperframes init meu-video --resolution portrait
cd meu-video
```

2. Copie a sua gravação para meu-video/assets/base.mp4, abra o Claude Code dentro da pasta meu-video e cole:

```
Use a skill /hyperframes. Quero montar um Reel vertical (1080x1920, 30 fps) com a minha gravação em assets/base.mp4. Não renderize ainda: pare na prévia.

1. Velocidade (opcional): crie uma versão 1,25x mais rápida mantendo o tom da voz (ffmpeg com setpts=PTS/1.25 e atempo=1.25). Use essa versão como assets/base.mp4 e guarde a original em assets/original.mp4.
2. Transcrição: transcreva o vídeo JÁ acelerado em português, com tempo por palavra (npx hyperframes transcribe assets/base.mp4 --language pt). Se o texto vier em inglês ou estranho, rode de novo com --model large-v3. Corrija à mão os nomes que o reconhecimento erra (ex.: "Cloud" → "Claude") e salve os tempos em tools/words.json.
3. Estrutura: index.html com o vídeo de fundo (data-has-audio="true") e uma sub-composição por efeito em compositions/. Escreva um gerador tools/build.py que lê words.json e calcula o início de cada efeito na 1ª sílaba do nome dele, para eu poder trocar a fala sem refazer os tempos à mão.
4. Legendas por palavra: grupos de até 3 palavras (ou até a pontuação), Inter 900, 58 px, branco com sombra preta, na altura do peito (y ≈ 1168). Palavras-chave em amarelo #F5E663 e 66 px. Cada grupo entra com opacidade 0→1 e escala 0,92→1 em 0,08 s. Nos efeitos que já escrevem o próprio nome na tela, esconda a legenda.
5. Lista dos efeitos (opcional, como no Reel "16 efeitos"; outra opção é a lista lateral do efeito 25): título "[N] efeitos de vídeo / que o Claude cria" no topo (Inter 900, 70 px, o número em amarelo) e duas colunas de vidro escuro (rgba(18,18,22,.72), borda branca 14%, raio 26 px) com os itens 01–08 e 09–16 em 29 px. O item falado fica amarelo e em negrito, com um empurrão de 10 px para a direita (0,12 s, ida e volta).
6. Rode npx hyperframes check, corrija os erros e abra a prévia com npx hyperframes preview. Me diga o endereço para eu assistir.
```

3. Para assistir no navegador (o Claude também pode abrir por você)

```
npx hyperframes preview
```

4. Só quando estiver bom: exporte o MP4

```
npx hyperframes render -o meu-reel.mp4
```

- **Por que acelerar antes de transcrever:** no Reel, a fala da HeyGen tinha 26,75 s e ficou com 21,4 s em 1,25x. Os tempos por palavra foram tirados do vídeo já acelerado. Se você transcrever o original e acelerar depois, todos os efeitos saem atrasados.
- **Vídeo do iPhone (HEVC/MOV):** o `init` aceita `--video caminho/do/video.mov` e converte para H.264, mas transcreve em inglês por padrão. Por isso o pedido acima faz a transcrição em português num passo separado.
- **A prévia** abre por padrão em `localhost:3002`. Se essa porta estiver ocupada, use `--port`.

Antes dos efeitos: o recorte da pessoa

Seis dos 25 efeitos precisam separar você do fundo: **vertigo, fundo trocado, recorte de papel, órbita, rabisco vivo e letras espalhadas**. No projeto do Reel “16 efeitos”, a separação foi feita em três peças, todas no próprio computador:

- **Recorte em vídeo** (`cutout.webm`): gerado pelo comando `npx hyperframes remove-background`, que roda um modelo de IA local. Foram 642 quadros em cerca de 3 minutos num Mac M2 Pro (CoreML). Esse WebM transparente fica por cima do vídeo original, no mesmo tempo, e tudo o que fica entre os dois parece estar atrás de você.
- **Placa limpa** (`plate.jpg`): uma foto do fundo sem a pessoa. Foi feita com um script Swift que usa a segmentação de pessoas da Apple (Vision): a máscara é dilatada em 28 px e a área é preenchida com as cores do entorno, borradas. Serve para o vertigo, em que o fundo precisa crescer sem levar você junto.
- **Contorno**: a mesma segmentação, aplicada a um único quadro, com detecção de contornos. Gera os pontos da silhueta que viram o rabisco vivo.

Pedido para preparar as camadas (faça uma vez por vídeo)

```
Prepare as camadas de recorte deste projeto (assets/base.mp4, 1080x1920). Não altere o vídeo
original.

1. Recorte da pessoa: rode npx hyperframes remove-background assets/base.mp4 -o assets/
cutout.webm --quality best. O resultado é um WebM transparente com o mesmo tempo do vídeo.
2. Placa limpa (o fundo sem a pessoa): pegue um quadro do vídeo e a máscara da pessoa nesse
quadro, dilate a máscara uns 28 px e preencha a área da pessoa com as cores do entorno (média
borrada, raios de 40 e 120 px). Salve como assets/plate.jpg. No Mac, você pode usar o Vision da
Apple (VNGeneratePersonSegmentationRequest) num script Swift. Em outro sistema, use o canal
alfa do cutout.webm como máscara.
3. Contorno da silhueta (para o rabisco vivo): num quadro em que eu esteja parado, extraia o
maior contorno da máscara, simplifique (epsilon 0,003) e salve os pontos normalizados de 0 a 1
em tools/layout.json, na chave "contour". Calcule também o centro do meu rosto em pixels e
salve em "face".
4. Monte o "sanduíche" no index.html, dentro de um contêiner #cam de 1080x1920 que se move
inteiro: vídeo original → placas e cenários (escondidos por padrão) → elementos que passam
ATRÁS de mim → cutout.webm (muted, mesmo data-start e data-duration do vídeo) → elementos na
FRENTE.
5. Tire snapshots em 3 momentos (npx hyperframes snapshot --at 2,8,14) e me mostre se a borda
do recorte está boa nas mãos, no cabelo e embaixo.
```

Limite real: no Reel, o recorte automático deixou uma mancha clara na região da cadeira e das pernas e uma borda imperfeita na mão. Funcionou porque cada efeito dura menos de 1,2 s. Grave com fundo liso, boa luz e as mãos longe das bordas do quadro para ter um recorte mais limpo.

01 · Parede de manchetes

Sem recorte · Reel “16 efeitos” · entra aos 2,56 s · dura 1,04 s

O que faz: recortes de jornal com borda rasgada pipocam em volta da pessoa, girando de leve, e uma manchete grande com marca-texto amarelo domina o centro.

Quando usar: na abertura, para dar ar de notícia ou para mostrar “o que estão falando” de um assunto.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie a cena "parede de manchetes" começando na 1ª sílaba da palavra [PALAVRA] e terminando ~0,6 s depois da próxima fala.
- 3 recortes pequenos (~300 px de largura) nas laterais, em papel #F4EEDF, com borda rasgada (clip-path polygon irregular, 18 pontos), sombra 0 14px 30px rgba(0,0,0,.45). Texto em EB Garamond 700, cor #1A1712, com uma linha de chamada em JetBrains Mono 28 px maiúscula (ex.: "JORNAL DO CRIADOR").
- 1 card de reel (250x360, borda branca de 3 px, raio 18) com uma miniatura do meu vídeo e um rótulo amarelo.
- 1 manchete grande (780 px de largura, 66 px): "[SUA MANCHETE]", com a segunda linha em marca-texto amarelo #F5E663 (gradiente de 52% a 92% da altura da linha).
- Entrada: cada recorte vem de opacidade 0, escala 1,5 e rotação 3x maior para a rotação final (entre −5° e 4°), em 0,22 s, ease back.out(1.6), um a cada 0,09 s. A manchete grande entra em 0,3 s: escala 1,4 e rotação −6° → escala 1 e −2°, 0,24 s, back.out(1.6).
Rode o check e me mostre um snapshot no meio da cena.
```

Dica: deixe a área do rosto livre. No Reel, os recortes ocupam a faixa entre y 700 e y 1100, abaixo do rosto e acima das mãos.

02 · Zoom pelo corte

Sem recorte · Reel “16 efeitos” · entra aos 3,60 s · dura 0,96 s

O que faz: a câmera “mergulha” dentro da manchete: ela cresce 9 vezes com desfoque, a imagem acompanha com um zoom rápido e um flash branco faz o corte para a cena seguinte.

Quando usar: como transição entre dois assuntos, ou logo depois da parede de manchetes (é a continuação dela).

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), na 1ª sílaba de [PALAVRA], faça o "zoom pelo corte" a partir
da manchete grande da cena anterior:
- Os recortes pequenos somem em 0,12 s.
- A manchete grande vai para escala 9, rotação 0, x −140, y −120 e blur(10px) em 0,36 s com
ease expo.in. Use transformOrigin em cima das palavras marcadas (no nosso, 62% 72%), para o
mergulho ser "para dentro" do texto. Em 0,3 s ela some (0,06 s).
- Ao mesmo tempo, o contêiner da câmera (#cam, com o vídeo) vai de escala 1 para 1,28 em 0,36
s, expo.in, e volta seco para 1 em 0,38 s.
- Flash branco de tela cheia: opacidade 0 → 0,85 em 0,04 s aos 0,34 s, e de volta a 0 em 0,14
s.
Me mostre snapshots aos 0,1 s, 0,3 s e 0,4 s depois do início.
```

Dica: o ease `expo.in` é o segredo: o crescimento começa devagar e acelera no fim, como se a câmera atravessasse o papel. Com um ease suave o efeito vira um zoom comum.

03 · Vertigo

Precisa do recorte · Reel “16 efeitos” · entra aos 4,56 s · dura 0,64 s

O que faz: o efeito “dolly zoom” do cinema: o fundo cresce 1,5 vez atrás da pessoa, que quase não muda de tamanho, com uma vinheta escura nas bordas. Dá a sensação de vertigem.

Quando usar: em uma revelação, numa frase de impacto ou num “espera aí…”.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), usando a placa limpa (assets/plate.jpg) e o recorte (assets/
cutout.webm) já preparados, faça o "vertigo" na 1ª sílaba de [PALAVRA], até a próxima fala:
- Mostre a placa limpa ATRÁS do recorte e escale de 1,0 para 1,5 durante toda a duração, ease
power1.inOut, com transformOrigin no centro do meu rosto (layout.json → face).
- Ao mesmo tempo, escale o contêiner #cam (que leva a pessoa junto) de 1,0 para só 1,08, também
power1.inOut.
- Ligue uma vinheta: radial-gradient(ellipse at 50% 40%, transparent 45%, rgba(0,0,0,.55)).
- No fim, esconda a placa e a vinheta e volte a câmera para escala 1 de uma vez (corte seco).
```

Dica: a diferença entre 1,5 (fundo) e 1,08 (pessoa) é o que cria a vertigem. Se a placa tiver um “fantasma” da pessoa, aumente a dilatação da máscara antes de preencher.

04 · Fundo trocado

Precisa do recorte · Reel “16 efeitos” · entra aos 5,20 s · dura 0,8 s

O que faz: por 0,16 s aparece uma tela verde de chroma key atrás da pessoa e, em seguida, o fundo vira outro cenário: no Reel, um espaço com estrelas e um planeta com anel.

Quando usar: para “teletransportar” você para outro lugar ou mostrar que o fundo é editável.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), faça o "fundo trocado" na 1ª sílaba de [PALAVRA], entre o
vídeo original e o recorte (assets/cutout.webm):
- 0,00–0,16 s: placa verde chroma (#00B140) cobrindo todo o fundo.
- A partir de 0,16 s: cenário [DESCREVA O CENÁRIO]. Se for o espaço, como no nosso: fundo
radial-gradient(ellipse at 50% 120%, #3B1F6B, #140A2E 45%, #05030D 80%), 140 estrelas brancas
de 3 a 6 px com posições sorteadas com semente fixa (nada de Math.random sem semente), um
planeta de 360 px em degradê laranja com anel inclinado em −14° e uma superfície marrom
embaixo.
- O cenário entra com escala 1,12 → 1 em 0,6 s, ease power2.out.
- No fim da cena, esconda tudo de uma vez.
Use apenas imagens ou desenhos que eu possa usar (nada de clipe com direitos autorais).
```

Dica: no Reel original o fundo era um clipe musical famoso. Aqui ele foi trocado por um cenário desenhado em código, que não tem problema de direitos autorais.

05 · Celular na tela

Sem recorte · Reel “16 efeitos” · entra aos 6,00 s · dura 0,88 s

O que faz: um celular entra pela esquerda, gira de leve e mostra uma página rolando: no Reel, “VOCÊ GRAVA. O CLAUDE EDITA.”, uma miniatura, um botão e uma lista de tarefas.

Quando usar: para mostrar um app, um site, um post ou uma oferta sem sair do vídeo.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie a cena "celular na tela" na 1ª sílaba de [PALAVRA], até a próxima fala:
- Celular de 280x570 px em x 40, y 730, raio 46, corpo #0B0B0E com borda de 8 px #2A2A30, notch de 72x22 e sombra 0 30px 60px rgba(0,0,0,.6).
- Conteúdo da tela, em uma coluna mais alta que o celular: [TÍTULO EM 2 LINHAS, a 2ª em amarelo #F5E663], uma imagem, um botão vermelho #E8483B "[TEXTO DO BOTÃO]" e uma lista com ✓.
- Entrada: x −420 e rotação −14° → x 0 e −4°, em 0,32 s, power3.out.
- Rolagem: a coluna sobe 300 px em 0,42 s, power2.inOut, começando aos 0,36 s.
- Saída: volta para x −420 e rotação −12° em 0,14 s, power2.in, nos últimos 0,15 s da cena.
```

Dica: troque o conteúdo por um print real do seu site ou app. Se o print for vertical e comprido, a rolagem fica natural.

06 · Zoom digital

Sem recorte · Reel “16 efeitos” · entra aos 6,88 s · dura 0,8 s

O que faz: aparece uma interface de câmera (cantos de foco, “● REC” piscando e um contador de zoom de 1,00x a 1,30x) enquanto a imagem se aproxima do rosto em velocidade constante.

Quando usar: para dar ênfase a uma frase, simulando um operador de câmera.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "zoom digital" na 1ª sílaba de [PALAVRA], até a próxima fala:
- 4 cantos de foco brancos de 108x108 px com borda de 6 px em volta do rosto (no nosso: x 300 e 672, y 390 e 782). Eles entram com escala 1,25 → 1 em 0,25 s, power3.out.
- Moldura fina de 2 px rgba(255,255,255,.35) e, embaixo, em JetBrains Mono 32 px: "● REC 00:07" com a bolinha vermelha #FF3B30 piscando (opacidade 0,2, 0,2 s, ease steps(1), 3 repetições) e um contador "1.00x".
- O contêiner #cam escala de 1 para 1,3 durante toda a cena, ease "none" (linear), com origem no centro do meu rosto. O contador acompanha de 1.00 a 1.30 no mesmo tween (onUpdate).
- Fim: tudo some e a câmera volta para 1 num corte seco.
```

Dica: o ease linear é proposital: zoom digital de verdade não tem aceleração. Use transformOrigin no rosto para ele não sair do quadro.

07 · Recorte de papel

Precisa do recorte · Reel “16 efeitos” · entra aos 7,68 s · dura 0,8 s

O que faz: a pessoa vira um adesivo com borda branca grossa sobre papel quadriculado com uma fita amarela, e as letras “RECORTE DE PAPEL” entram uma a uma, como recortes de revista.

Quando usar: em conteúdo criativo, colagens, bastidores ou para dar um respiro de estilo.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), faça o "recorte de papel" na 1ª sílaba de [PALAVRA], até a próxima fala:
- Fundo (atrás do recorte): papel #EFE8D8 com grade de 60 px em linhas de 2 px rgba(0,0,0,.07) e uma faixa de fita amarela rgba(245,230,99,.85) de 200 px de altura, girada −6°, na altura de y 300.
- Uma segunda cópia do cutout.webm com borda branca de 7 px: filter drop-shadow(7px 0 0 #fff) drop-shadow(−7px 0 0 #fff) drop-shadow(0 7px 0 #fff) drop-shadow(0 −7px 0 #fff) drop-shadow(0 12px 18px rgba(0,0,0,.35)).
- A câmera "pula": escala 0,94 e rotação −1,5° em 0,2 s, back.out(2).
- Letras "[PALAVRA 1]" e "[PALAVRA 2]" em duas linhas (y 1000 e 1110): cada letra num bloco de 84x96 px, alternando 8 estilos (fundo/cor/fonte: branco/preto Playfair Display; preto/branco Archivo Black; amarelo/preto Space Mono; azul #2D5BFF/branco Oswald; branco/vermelho Archivo Black; vermelho #E8483B/branco EB Garamond; branco/preto League Gothic; preto/amarelo Playfair Display), rotação entre −6° e 6° e desnível de 8 px nas letras alternadas.
- As letras entram com escala 1,8 → 1 e opacidade 0 → 1, 0,12 s, back.out(2), uma a cada 0,055 s, começando aos 0,08 s.
- Esconda a legenda durante essa cena.
```

Dica: fontes fora do pacote do HyperFrames precisam de arquivo em `assets/fonts/` com `@font-face`. Peça ao Claude para conferir isso e trocar por uma fonte parecida se faltar o arquivo.

08 · Mundo de interface

Sem recorte · Reel “16 efeitos” · entra aos 8,80 s · dura 1,12 s

O que faz: cartões de vidro flutuam em volta da pessoa como um app: um prompt sendo digitado, uma barra “editando… 0→100%”, uma timeline com miniaturas e uma prévia “● AO VIVO”.

Quando usar: para explicar um processo, mostrar uma ferramenta trabalhando ou falar de tecnologia.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "mundo de interface" na 1ª sílaba de [PALAVRA], até a próxima fala. Quatro cartões de vidro (fundo rgba(18,18,22,.72), borda 2 px rgba(255,255,255,.14), raio 26 px, sombra 0 18px 40px rgba(0,0,0,.35)), sem cobrir o rosto:
1. PROMPT (430x150, esquerda): rótulo em JetBrains Mono 28 px e o texto "[SEU PROMPT]▏" em Inter 800, 44 px, aparecendo letra a letra com clip-path inset(0 100% 0 0) → inset(0 0% 0 0) em 0,42 s, ease steps(14), a partir de 0,12 s.
2. EDITANDO (430x112): "editando…" e um contador que vai de 0 a 100% junto com uma barra amarela #F5E663 (scaleX 0 → 1), 0,7 s, power1.inOut, a partir de 0,2 s.
3. PRÉVIA (350x430, direita): "PRÉVIA" e "● AO VIVO" em verde #3DDC97, com uma miniatura do meu vídeo.
4. TIMELINE (640x128, embaixo): 6 miniaturas de 96x66 e uma agulha amarela andando 560 px em 0,9 s, ease none.
Entrada: cada cartão sobe 40 px com opacidade 0 → 1 em 0,32 s, power4.out, com atrasos de 0 / 0,04 / 0,08 / 0,12 s. Saída: sobem 30 px e somem em 0,1 s no fim da cena.
```

Dica: para as miniaturas, peça ao Claude para extrair quadros do seu próprio vídeo (no Reel foram imagens de `assets/stills/`). Fica coerente e não usa imagem de terceiros.

09 · Órbita

Precisa do recorte · Reel “16 efeitos” · entra aos 9,84 s · dura 0,56 s

O que faz: seis cards de reel giram em carrossel 3D em volta da pessoa, passando por trás e pela frente do recorte.

Quando usar: para mostrar vários exemplos, produtos ou trabalhos de uma vez.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), faça a "órbita" na 1ª sílaba de [PALAVRA], até a próxima
fala:
- 6 cards de 190x320 px (raio 16, borda branca de 3 px), cada um com uma miniatura do meu vídeo
e um rótulo amarelo embaixo: [6 RÓTULOS].
- Crie DUAS cópias de cada card: uma camada atrás do cutout.webm e outra na frente.
- Um único tween de ângulo de 0° a 250° durante a cena toda, ease power1.inOut. Para cada card
i: t = (ângulo + i·60)°; x = sen(t)·400 px; y = −40·cos(t); escala = 0,78 + 0,28·(cos(t)+1)/2;
rotateY = −sen(t)·35°.
- Quando cos(t) > 0, o card está na frente: mostre a cópia da frente e esconda a de trás.
Quando cos(t) < 0, o contrário.
- Centro da órbita na altura do peito (cards com left 445, top 660).
Tudo calculado a partir do tempo da timeline, sem requestAnimationFrame.
```

Dica: a ilusão de 3D vem das duas cópias, não de 3D de verdade. Se um card “atravessar” o corpo na hora errada, confira o sinal do cosseno.

10 · Quebra de quadro

Sem recorte · Reel “16 efeitos” · entra aos 10,40 s · dura 0,8 s

O que faz: o vídeo encolhe e vira um card de Reel (curtir, comentar, enviar, @perfil, Seguir) enquanto o quadro de fundo se estilhaça em 24 cacos que voam para fora.

Quando usar: para falar de redes sociais, de publicação ou para virar a chave entre duas partes do vídeo.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), faça a "quebra de quadro" na 1ª sílaba de [PALAVRA], até a próxima fala:
- Atrás da câmera: fundo radial-gradient(circle at 50% 45%, #2a1416, #070708 70%) e 24 cacos (grade de 4 colunas x 6 linhas, de 270x320 px, cada canto deslocado entre −30 e 30 px com semente fixa). Cada caco usa um quadro congelado do vídeo como fundo, recortado com clip-path polygon.
- Os cacos voam para fora a partir do centro (540, 860): ~900 px na direção do próprio centro ±80 px, rotação entre −70° e 70°, opacidade 1 → 0, 0,55 s, power3.out, com atrasos sorteados de 0 a 0,08 s.
- A câmera (#cam) encolhe para 0,62 e sobe 120 px em 0,28 s, power3.out, com origem no centro do quadro.
- Na frente: moldura de Reel de 670x1188 (raio 34, borda branca de 4 px), ícones ♡ 💬 ➤ à direita, "[@SEUPERFIL]" com botão "Seguir" e uma linha de legenda. Entra com escala 1,6 → 1 e opacidade 0 → 1 em 0,3 s, power3.out, e some nos últimos 0,12 s.
- No fim, a câmera volta para escala 1 num corte seco.
```

Dica: sorteie os cacos com semente fixa (por exemplo, `random.Random(7)` no gerador). O HyperFrames proíbe sorteio sem semente porque o render precisa sair igual à prévia.

11 · Rabisco vivo

Precisa do recorte · Reel “16 efeitos” · entra aos 11,20 s · dura 0,88 s

O que faz: contornos desenhados à mão (branco, amarelo, rosa e azul) seguem a silhueta da pessoa e “fervem”: o traço é redesenhado a cada 0,1 s. Círculos rabiscados sobem pela tela.

Quando usar: para dar energia, num momento divertido ou para destacar quem está falando.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), faça o "rabisco vivo" na 1ª sílaba de [PALAVRA], até a
próxima fala, usando o contorno da minha silhueta salvo em tools/layout.json ("contour", pontos
de 0 a 1):
- Gere no build.py 4 traços paralelos ao contorno, deslocados para FORA pela normal: 26 px
(branco, 9 px de espessura), 50 px (amarelo #F5E663, 7 px), 74 px (rosa #FF7AB6, 5 px) e 92 px
(azul #7FD3FF, 4 px), cada ponto com tremor aleatório de ±9 px. Corte os trechos nas bordas do
quadro e abaixo de y 1330. Curvas suaves (Q) num SVG de 1080x1920, linecap round.
- Gere 4 variantes do desenho com sementes diferentes. A primeira aparece "sendo desenhada"
(pathLength=1, strokeDashoffset 1 → 0 em 0,28 s, power2.out). Depois, troque de variante a cada
0,1 s (tl.set de opacidade). Esse é o "ferver".
- 3 círculos de contorno (raios 20 a 26 px, traço de 6 px) sobem de y +60 a −120 com opacidade
0 → 1 em 0,7 s, power1.out, um a cada 0,08 s, começando aos 0,1 s.
- O SVG fica atrás do recorte, para o traço não passar por cima do corpo.
```

Dica: o contorno sai de um único quadro. Funciona porque a pessoa está quase parada. Se você se mexe muito, peça um contorno a cada 0,1 s em vez de um só.

12 · Texto adesivo

Sem recorte · Reel “16 efeitos” · entra aos 12,08 s · dura 0,88 s

O que faz: uma etiqueta amarela com o número e o nome do efeito cai do alto, quica e gruda no peito. Como fica dentro da câmera, acompanha qualquer zoom.

Quando usar: para nomear um tópico, marcar um preço, um passo ou uma palavra-chave.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "texto adesivo" na 1ª sílaba de [PALAVRA], até a próxima fala:
- Etiqueta amarela #F5E663, raio 16 px, padding 14px 30px 14px 22px, sombra dura 0 10px 0 rgba(0,0,0,.35), texto "[TEXTO]" em Inter 900, 48 px, preto. Antes do texto, um número "[NÚMERO]" numa caixinha preta com texto amarelo de 30 px.
- Posição no peito (no nosso, x 520, y 960). Coloque DENTRO do contêiner #cam para seguir os movimentos da câmera.
- Entrada: de y −700, rotação −30°, escala 1,3 e opacidade 0 para y 0, rotação −6°, escala 1, em 0,36 s, ease bounce.out.
- Some de uma vez no fim da cena. Esconda a legenda enquanto a etiqueta estiver na tela.
```

Dica: o `bounce.out` faz o quique. Para uma batida mais seca, troque por `back.out(2)`.

13 · Antes e depois

Sem recorte · Reel “16 efeitos” · entra aos 12,96 s · dura 0,72 s (no “editado pelo Claude”: slider P&B aos 24,56 s)

O que faz: compara duas versões do mesmo vídeo no mesmo quadro. Variante 1 (Reel “16 efeitos”): uma faixa de luz varre a tela da direita para a esquerda e revela a versão com a cor tratada, com os rótulos “antes” e “● depois”. Variante 2 (Reel “editado pelo Claude”): uma divisória com alça “‹›” anda pela tela, com o vídeo em preto e branco de um lado e colorido do outro.

Quando usar: para mostrar resultado de edição, cor, produto, reforma, qualquer comparação.

Variante 1 · faixa de luz

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), faça o "antes e depois" com FAIXA DE LUZ na 1ª sílaba de [PALAVRA], até a próxima fala:
- Prepare uma segunda versão do vídeo com a cor tratada ([MAIS CONTRASTE E SATURAÇÃO / O TOM QUE EU QUERO]), com exatamente a mesma duração e enquadramento: assets/graded.mp4 (muted). Use npx hyperframes media-treatment ou ffmpeg; não use filtro CSS no vídeo.
- Coloque graded.mp4 por cima do original e revele com clip-path inset(0 0 0 100%) → inset(0 0 0 0%) durante a cena (menos 0,08 s), ease power1.inOut. A direita é o "depois".
- Faixa de luz: 16 px de largura, branca, com box-shadow 0 0 40px 18px rgba(255,255,255,.75), indo de x 1080 a x −40 no mesmo tempo e ease da revelação.
- Rótulos em pílula (Inter 800, 32 px, fundo rgba(0,0,0,.55)): "antes" à esquerda e "● depois" à direita, na altura de y 1060, com pop de escala 0,7 → 1 (0,3 s, back.out(1.8)).
- Se houver camada de recorte por cima, esconda durante a cena para não cobrir a versão tratada.
```

Variante 2 · slider preto e branco

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), faça o "antes e depois" com SLIDER na 1ª sílaba de [PALAVRA], até a próxima fala:
- Gere uma cópia do vídeo em preto e branco, com a mesma duração e enquadramento: assets/base-pb.mp4 (muted). Use npx hyperframes media-treatment ou ffmpeg, não filtro CSS.
- Coloque a cópia P&B por cima do original, dentro de um wrapper com clip-path que mostra só a parte à ESQUERDA da divisória.
- Divisória: linha vertical branca de 4 px, da altura da área útil (y 168 a 1408), com sombra 0 0 12px rgba(0,0,0,.4), uma alça redonda de 68 px no meio (borda branca de 4 px, fundo escuro, texto "‹ ›" em JetBrains Mono) e os rótulos "ANTES" (à esquerda da linha) e "DEPOIS" (à direita) em JetBrains Mono 28 px, letter-spacing 3 px, fundo rgba(0,0,0,.55).
- A divisória anda de x 160 até x 620 durante a cena (menos 0,35 s), ease power2.inOut, começando aos 0,1 s. O clip-path do P&B segue exatamente a mesma posição, duração e ease.
- Título opcional no topo: "ANTES" em Archivo Black 100 px e "e depois." em Playfair Display itálico dourado (#D8B46A) 72 px, entrando com blur 10 px → 0 em 0,18 s.
- No fim, a linha some em 0,12 s e o P&B é cortado.
```

Dica: a faixa (ou a divisória) e a revelação precisam do mesmo ease e da mesma duração. Senão a linha se descola da borda entre as duas versões.

14 · Recuo revelador

Sem recorte · Reel “16 efeitos” · entra aos 13,68 s · dura 1,2 s

O que faz: a câmera recua até 40% e mostra que o vídeo era a prévia dentro de um editor: checklist sendo marcado, painel de efeitos, timeline colorida e forma de onda.

Quando usar: para mostrar bastidores, o processo ou “como isso foi feito”.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "recuo revelador" na 1ª sílaba de [PALAVRA], até a próxima fala:
- Atrás da câmera, um editor escuro (#0C0D11): à esquerda um cartão de vidro "checklist" com 3 itens ("○ [ITEM 1]", "○ [ITEM 2]", "○ [ITEM 3]") que viram "✓ ..." em verde #3DDC97 aos 0,45 / 0,65 / 0,85 s; à direita um cartão "efeitos.reel" com 3 linhas, a última em amarelo com ✓; embaixo (y 1350) uma timeline com 14 blocos coloridos de 44 px de altura e larguras variadas (semente fixa), uma forma de onda verde de 113 barras e uma agulha branca andando 700 px em 1,2 s, ease none.
- O contêiner #cam recua para escala 0,40 em 0,42 s, ease power3.inOut, com origem no centro do quadro (540, 960).
- Aos 0,3 s aparece uma moldura branca de 10 px com raio 40 em volta do vídeo encolhido.
- Os cartões entram subindo 40 px (0,32 s, power4.out) aos 0,2 e 0,26 s, e a timeline aparece em 0,3 s.
- No fim, corte seco de volta para escala 1.
```

Dica: esse é o efeito mais longo (1,2 s) porque o olho precisa de tempo para ler o editor. Abaixo de 0,8 s ele vira só um encolhimento.

15 · Letras espalhadas

Precisa do recorte · Reel “16 efeitos” · entra aos 14,88 s · dura 1,04 s

O que faz: duas palavras gigantes se montam a partir de letras espalhadas e desfocadas: “LETRAS” em branco, atrás da cabeça, e “ESPALHADAS” em amarelo, na frente do corpo. A profundidade vem do recorte.

Quando usar: para o título de um tema, uma palavra de impacto ou o nome de um produto.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), faça as "letras espalhadas" na 1ª sílaba de [PALAVRA], até a
próxima fala:
- Palavra 1 "[PALAVRA ATRÁS]": Inter 900, 190 px, branca, letter-spacing -4 px, centralizada em
y 690, numa camada ENTRE o vídeo original e o cutout.webm (passa atrás da cabeça).
- Palavra 2 "[PALAVRA NA FRENTE]": Inter 900, 132 px, amarela #F5E663, centralizada em y 960,
na FRENTE do recorte.
- Cada letra é um span separado. Ela parte de uma posição sorteada (x entre -500 e 500, y entre
-400 e 400), escala sorteada entre 0,3 / 2,6 / 3,2 / 0,4, opacidade 0 e blur(16px), e vai para
a posição final, escala 1 e blur 0 em 0,42 s, power3.out. Atraso de 0,035 s entre letras na
palavra de trás e de 0,03 s na da frente (que começa aos 0,18 s).
- Semente fixa nos sorteios. Esconda a legenda nessa cena.
```

Dica: a palavra de trás fica parcialmente escondida pela cabeça. No Reel, “LETRAS” apareceu como “LE··AS”. Escolha uma palavra que ainda se leia com o meio coberto, ou suba a palavra para acima da cabeça.

16 · Balão de comentário

Sem recorte · Reel “16 efeitos” · entra no “comenta aqui” e vai até o fim

O que faz: mostra o comentário sendo escrito e publicado. Variante 1 (Reel “16 efeitos”): campo com contorno amarelo, a palavra é digitada e enviada, sobe como balão com coração e vem um card “@benites.ia Seguir → Seguindo ✓”. Variante 2 (Reel “editado pelo Claude”): caixa branca no topo digita “EFEITOS”, o “Publicar” acende, aparece a resposta “Te mandei no direct ✓”, a palavra cai em letras gigantes e chega a notificação do direct.

Quando usar: no final, para mostrar exatamente o que a pessoa deve comentar e pedir para seguir.

Variante 1 · balão + seguir

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o CTA "balão de comentário" começando na palavra "[COMENTA]" e indo até o fim do vídeo:
- Campo em pílula de 940x112 (x 70, y 1000, raio 56, fundo rgba(16,16,20,.88), borda 3 px rgba(255,255,255,.18)) com avatar cinza, placeholder "Adicione um comentário…" e botão de envio redondo. Entra subindo 40 px (0,32 s, power4.out).
- Na palavra "[AQUI]", a borda fica amarela #F5E663 (0,12 s).
- Na palavra "[PALAVRA-CHAVE]": o placeholder some e "[PALAVRA-CHAVE]|" é digitado em Inter 900, 54 px, com clip-path em 0,36 s, ease steps(6). Depois o botão de envio fica amarelo (0,1 s).
- Na fala "[ME SEGUE]": o campo desce 30 px e some (0,14 s, power2.in); o comentário enviado sobe como balão branco ("você" em cinza + "[PALAVRA-CHAVE]" em 66 px preto), de y +150 a 0 em 0,3 s, power3.out; um coração rosa #FF2D6F faz pop aos 0,3 s; e um card "[@SEUPERFIL]" com botão azul "Seguir" (#0095F6) sobe 60 px.
- Na palavra "[FERRAMENTAS]", o botão faz um clique (escala 0,9, ida e volta) e vira "Seguindo ✓" cinza.
- Loop: nos últimos 0,22 s, limpe o balão e o card para que o último quadro fique igual ao primeiro.
```

Variante 2 · resposta + direct

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o CTA de comentário com RESPOSTA começando na palavra "[COMENTA]" e indo até o fim:
- Caixa de comentário branca no topo (x 30, y 190, 700x100, raio 50, sombra 0 14px 34px rgba(0,0,0,.4)) com avatar, placeholder "Adicione um comentário…" e o botão "Publicar" em cinza. Entra com opacidade 0, y -30 e scaleX 0,4 (origem à esquerda) → normal em 0,25 s, power3.out.
- Na palavra "[PALAVRA-CHAVE]": "[PALAVRA-CHAVE]|" é digitado em Inter 800, 38 px, com clip-path em 0,3 s, ease steps(8). Logo depois, "Publicar" fica vermelho (#E0412F) com texto branco.
- 0,34 s depois do início da digitação, a caixa vira o comentário publicado ("você · agora" + [PALAVRA-CHAVE] + ♥), com escala 0,92 → 1, 0,22 s, back.out(1.8). 0,24 s depois, a resposta "[@SEUPERFIL] · agora / [Te mandei no direct ✓]" desce 40 px até o lugar (0,25 s, expo.out).
- Ao mesmo tempo, embaixo (y 1040): "[PALAVRA-CHAVE]" gigante em Archivo Black 150 px, letra por letra caindo 120 px com bounce.out (0,36 s, 0,05 s entre letras), com uma barra vermelha vertical ao lado.
- Na fala "[SEGUE]": a palavra gigante encolhe e some (0,18 s) e entra uma pílula vermelha "+ SEGUE [@SEUPERFIL]" (escala 0,5 → 1, 0,3 s, back.out(1.8)).
- Por fim, comentário e resposta somem e chega uma notificação de direct no topo ("[@SEUPERFIL] · agora / [Os prompts completos 📎]"), descendo 80 px em 0,26 s, expo.out. Opcional: um som curto de notificação num <audio> com id, no mesmo instante.
```

Dica: só prometa “te mando no direct” se o envio estiver mesmo configurado, por exemplo com uma automação de DM. O primeiro Reel não prometia envio porque o material ainda não existia.

17 · Título cinético com máscara

Sem recorte · Reel “editado pelo Claude” · entra aos 4,40 s

O que faz: o título aparece letra por letra: cada letra nasce escondida atrás de um bloco vermelho que encolhe para a direita e a descobre.

Quando usar: na abertura de um tópico ou para dar peso a um título curto (uma ou duas palavras por linha).

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "título cinético com máscara" na 1ª sílaba de [PALAVRA]:
- Texto "[LINHA 1]" / "[LINHA 2]" em Archivo Black 100 px, branco, letter-spacing -2 px, com sombra 0 6px 24px rgba(0,0,0,.55), no topo à esquerda (x 36, y 180 e y 280).
- Cada letra é um span com overflow hidden e, dentro dele, um bloco vermelho #E0412F do tamanho da letra (110% de largura) por cima.
- As letras aparecem uma a cada 0,045 s a partir de 0,02 s. Os blocos encolhem com scaleX 1 → 0 e transformOrigin "right center" em 0,22 s, ease expo.out, também a cada 0,045 s, a partir de 0,06 s. O bloco "sai pela direita" e descobre a letra.
- O título fica parado até [PALAVRA SEGUINTE] e então some com opacidade 0 e blur(8px) em 0,15 s.
```

Dica: o efeito pede fonte bem grossa (Archivo Black, Inter 900). Com fonte fina, o bloco fica maior que a letra e a máscara aparece demais.

18 · Letras com gravidade

Sem recorte · Reel “editado pelo Claude” · entra aos 8,00 s

O que faz: uma palavra grande em serifada cai letra por letra do alto e quica ao chegar. Acima dela, uma palavra pequena em itálico dourado entra com desfoque.

Quando usar: para uma palavra que precisa de impacto, como um nome, um número ou um “atenção”.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), crie as "letras com gravidade" na 1ª sílaba de [PALAVRA]:
- Palavra pequena "[palavra de apoio]" em Playfair Display itálico 600, 56 px, dourado #D8B46A,
em x 44, y 184. Entra com blur(10px) → 0 e opacidade 0 → 1 em 0,18 s, power2.out.
- Palavra grande "[PALAVRA]" em Playfair Display 700, 112 px, branca, em x 36, y 240. Cada
letra é um span inline-block.
- Cada letra cai de y -420 até 0 em 0,42 s, ease bounce.out, com 0,055 s entre letras, a partir
de 0,05 s.
- Uma barra vertical vermelha (#E0412F, 20x96 px) à esquerda da palavra cresce de cima para
baixo (scaleY 0 → 1, 0,2 s, power3.out).
- Saída: tudo desce 40 px e some em 0,16 s, power3.in, quando começar a próxima frase.
```

Dica: durante a queda as letras atravessam o topo da tela, e o `check` avisa sobreposição. É esperado: dura menos de meio segundo.

19 · Zoom no rosto (punch-in)

Sem recorte · Reel “editado pelo Claude” · entra aos 12,08 s · e em cada palavra forte

O que faz: cortes secos de zoom no rosto nas palavras fortes: a imagem “pula” para 115% e volta na frase seguinte. Na demonstração aparecem uma moldura de foco e o selo “115%”.

Quando usar: em quase todo vídeo falado, para dar ritmo e marcar ênfase sem precisar de outro plano.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), aplique "zoom no rosto" (punch-in):
- Coloque o <video> dentro de um wrapper #cam-zoom de 1080x1920 e anime SÓ o wrapper, nunca o
<video> direto. transformOrigin no centro do rosto (no nosso, 490px 600px).
- Use a transcrição para achar as palavras fortes: [LISTA DE PALAVRAS]. Em cada uma, mude a
escala de uma vez para 1,15 (tl.set, corte seco). Na frase seguinte, volte para 1,0. Para ritmo
extra, alterne 1,07 e 1,0 no início de cada tópico.
- Na demonstração: uma moldura de foco de 400x500 px em volta do rosto, com 4 cantos de 70 px e
borda branca de 7 px, entrando com escala 1,5 → 1 e opacidade 0 → 1 em 0,3 s, expo.out. Um selo
"115%" em JetBrains Mono 64 px, texto escuro sobre creme #F4EDE2, raio 16, faz pop (0,28 s,
back.out(1.7)).
```

Dica: corte seco (tl.set) parece edição de verdade. Se preferir um zoom que deslize, use 0,12 s com power3.out, mas não passe disso, senão vira “câmera flutuando”.

20 · Barra de prompt digitando

Sem recorte · Reel “editado pelo Claude” · entra aos 16,16 s · uma barra por tópico

O que faz: uma barra branca em forma de pílula entra no topo, o texto é digitado letra por letra com um cursor vermelho piscando, e o botão de enviar vira um ✓ verde.

Quando usar: para mostrar o pedido que gerou cada resultado, numa busca, num prompt ou num comando.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), crie a "barra de prompt" na fala [TRECHO DA FALA]:
- Pílula branca de 680x156 px em x 36, y 190, raio 34, sombra 0 14px 34px rgba(0,0,0,.4).
Dentro: rótulo "PROMPT [01]" em JetBrains Mono 28 px cinza #9A8F86 e o texto "[O QUE VAI SER
DIGITADO]" em Inter 700, 33 px, cor #171313, com até 2 linhas (largura 540 px).
- Entrada: opacidade 0, y −30 e scaleX 0,35 (origem à esquerda) → normal em 0,24 s, power3.out.
- Digitação: cada caractere é um span que acende (opacidade 0 → 1) em sequência. A duração
total é entre 0,7 e 1,4 s (número de caracteres ÷ 34), começando 0,08 s depois da entrada.
Cursor "|" vermelho #E0412F que pisca a cada 0,3 s depois de terminar.
- Botão redondo de 66 px à direita: vermelho com "↑" e, ao terminar a digitação, troca por um
verde #1E8E5A com "✓" (escala 0,4 → 1, 0,22 s, back.out(2.2)).
- Saída: sobe 20 px, blur(6px) e some em 0,14 s, power2.in, logo antes do próximo tópico.
```

Dica: escreva o mesmo texto que você fala. Barra e legenda dizendo a mesma coisa se reforçam; textos diferentes disputam a leitura.

21 · Timeline de editor

Sem recorte · Reel “editado pelo Claude” · entra aos 20,72 s

O que faz: embaixo da tela surge a timeline de um editor de vídeo, com timecode correndo, trilha de clipes, forma de onda, marcadores e uma agulha vermelha andando junto com a fala.

Quando usar: para falar de edição, de etapas de um processo ou de tempo.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie a "timeline de editor" na 1ª sílaba de [PALAVRA], até o próximo tópico:
- Cartão escuro (#141A16) de 1020x216 px em x 30, y 1000, raio 14, borda 2 px #ffffff22. Rótulo "Timeline" em Inter 800 28 px creme (#F4EDE2) e um timecode em JetBrains Mono 28 px no formato 00:00:05:13 que avança em 20 passos ao longo da cena (tl.set).
- Trilha V1: 5 clipes de 168x40 px, raio 6, alternando #E8D9B8 e #D9C49A. Trilha A1: forma de onda de 72 barras verdes #7CF2A6 de 7 px de largura e alturas variadas. 8 marcadores amarelos #F2C94C de 22x14 px.
- Agulha vermelha #E0412F (4x150 px, com cabeça de 22x16) andando 860 px durante toda a cena, ease none, sincronizada com o relógio do vídeo.
- Entrada: o cartão sobe de y +260 com opacidade 0 → 1 em 0,32 s, expo.out. Saída: desce 260 px e some em 0,18 s, power3.in, 0,2 s antes do fim da cena.
- No topo: título "TIMELINE" em Archivo Black 100 px revelado com clip-path da esquerda para a direita (0,25 s, power3.out) e "de editor." em Playfair itálico dourado 72 px.
```

Dica: para a forma de onda parecer real, peça as alturas a partir do volume do seu áudio, com um valor por barra. Assim ela sobe quando você fala mais alto.

22 · Conta-gotas + paleta

Sem recorte · Reel “editado pelo Claude” · entra aos 28,88 s

O que faz: um cursor de conta-gotas passeia por quatro pontos da cena (pele, cabelo, parede e camiseta) e, a cada parada, monta uma amostra na barra de paleta com o código hex. As cores são as reais do quadro.

Quando usar: para falar de cor, identidade visual, estilo ou “o que tem nesta imagem”.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "conta-gotas + paleta" na 1ª sílaba de [PALAVRA], até o próximo tópico:
- Antes, num script Python (PIL), pegue um quadro do vídeo e meça a cor média de 4 pontos (média de uma grade de 5x5 pixels, de 3 em 3 px, em volta de cada ponto): [PELE (x,y)], [CABELO (x,y)], [FUNDO (x,y)], [ROUPA (x,y)]. Se o vídeo estiver reenquadrado (escala ou deslocamento), converta as coordenadas da tela para o quadro original. Salve em assets/palette.json com nome, x, y e hex.
- Barra branca #FBF8F3 de 1020x140 px em x 30, y 1030, raio 18, com o rótulo "PALETA" em JetBrains Mono e 4 espaços: quadrado de 96 px (raio 12) + hex em maiúsculas (JetBrains Mono 28 px) + nome (Inter 700, 28 px). Entra de y +200 em 0,3 s, expo.out.
- Cursor de conta-gotas (SVG de 64 px) vai a cada ponto com ease power2.inOut. O tempo por ponto é o maior entre 0,35 s e (duração da cena − 0,6 s) ÷ 4, e o deslocamento usa 60% desse tempo. Na chegada: um anel de 90 px com borda de 6 px da cor amostrada se abre (escala 0,3 → 1, opacidade 1 → 0, 0,3 s) e o quadrado correspondente cresce de 0 a 1 (0,2 s, back.out(2)).
- Saída: barra e cursor descem 40 px e somem em 0,16 s.
```

Dica: no Reel, as cores amostradas foram #76423C (pele), #27231C (cabelo), #6A302B (parede) e #31201F (camiseta). Use sempre a cor medida do quadro, nunca uma cor “parecida” escolhida à mão.

23 · Look de cinema (letterbox)

Sem recorte · Reel “editado pelo Claude” · entra aos 32,48 s

O que faz: duas faixas pretas entram por cima e por baixo como num filme, a imagem ganha um tom mais quente e aparecem “look de cinema.” e “2.39 : 1”.

Quando usar: para uma frase dramática, um antes de virada ou para mudar o clima de um trecho.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie o "look de cinema" na 1ª sílaba de [PALAVRA], até o próximo tópico:
- Duas faixas #050505 de 1080x250 px (cerca de 13% da altura cada): a de cima entra de y -250 e a de baixo de y +250 até o lugar em 0,26 s, power3.out.
- Ao mesmo tempo, um tom quente: no Reel foi uma camada por cima do vídeo com linear-gradient(180deg, rgba(255,140,60,.10), rgba(20,30,60,.16)) e mix-blend-mode soft-light, entrando em 0,3 s. Para uma correção de cor de verdade no vídeo, use npx hyperframes media-treatment.
- Texto "look de cinema." em Playfair itálico dourado #D8B46A, 64 px, no topo, e "2.39 : 1" em JetBrains Mono 28 px à direita, os dois com blur → nítido em 0,18 s.
- Fique [0,8–3] s e saia na mesma velocidade: faixas para fora em 0,2 s, power3.in, e o tom some junto.
```

Dica: mantenha legenda e textos fora das faixas pretas (no Reel: abaixo de y 250 e acima de y 1670). Se a legenda cair na faixa, ela some.

24 · Cartão de título que vira celular

Sem recorte · Reel “editado pelo Claude” · entra aos 36,72 s

O que faz: um cartão em tela cheia com timecode, título em serifada e um fio fino encolhe até virar a tela de um celular, onde o vídeo continua rodando. Depois o celular cresce de volta até o vídeo ocupar a tela toda.

Quando usar: para separar capítulos, mostrar “isso aqui no celular” ou fazer uma transição elegante.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos
por palavra em tools/words.json), crie o "cartão de título que vira celular" na 1ª sílaba de
[PALAVRA], até o próximo tópico:
- Cartão de tela cheia na cor [#171313]. Timecode em JetBrains Mono 30 px laranja #FF6B4A
("00:00:31:12"), "[TÍTULO]" em Playfair Display 150 px creme #F4EDE2 aparecendo letra a letra
(0,07 s entre letras), "[subtítulo]" em Playfair itálico dourado 76 px e um fio de 200x3 px
crescendo (scaleX 0 → 1, 0,3 s, power3.out).
- Na fala [MOMENTO DO ENCOLHER]: o cartão encolhe para escala 0,46 com borderRadius 90, em 0,35
s, power3.inOut, com transformOrigin 207px 960px (área livre à esquerda), e some em 0,14 s.
- Ao mesmo tempo, o contêiner do vídeo assume escala 0,46 com a mesma origem e clip-path
inset(0 round 96px), dentro de uma moldura de celular creme (#F4EDE2) com borda de 22 px e raio
96. O vídeo continua rodando dentro do celular.
- Pelo menos 1 s depois (ou 0,45 s antes do próximo tópico): o celular cresce de volta para
escala 1 em 0,35 s, power3.inOut. A moldura some em 0,2 s e o clip-path volta para 0.
- Durante essa cena, desligue os punch-ins de zoom.
```

Dica: encolha para o lado vazio do quadro (no Reel, o lado esquerdo, porque o painel da lista ficava à direita). Assim o celular não cobre outra informação.

25 · Lista lateral com progresso

Sem recorte · Reel “editado pelo Claude” · o vídeo inteiro

O que faz: um painel escuro à direita lista os tópicos do vídeo com o cabeçalho “NESTE VÍDEO” e um contador “01/10”. O item falado vira uma barra clara, os já falados ganham ✓ e, no final, o rodapé vira uma pílula amarela.

Quando usar: em vídeos de lista (“10 coisas”, “5 erros”), para a pessoa saber quanto falta e ficar até o fim.

Prompt pronto para colar

```
No projeto HyperFrames desta pasta (Reel 1080x1920, 30 fps, gravação em assets/base.mp4, tempos por palavra em tools/words.json), crie a "lista lateral" durante o vídeo todo, com os itens [LISTA DOS ITENS]:
- Painel em x 728, y 300, 330 px de largura, raio 14, fundo rgba(23,19,19,.9), borda 2 px #ffffff1c, sombra 0 18px 40px rgba(0,0,0,.35). Cabeçalho "NESTE VÍDEO" em JetBrains Mono 28 px creme (#F4EDE2) e o contador "00/[TOTAL]" em laranja #FF6B4A à direita, com um fio de 2 px embaixo.
- Uma linha a cada 56 px: número em JetBrains Mono 28 px (50% de opacidade) + nome em Inter 700, 28 px, creme.
- Entrada (quando a lista começa a ser falada): de x +90 e blur(8px) para o lugar em 0,35 s, expo.out.
- Quando eu falo o número/nome de cada item (tempos da transcrição): uma barra creme cresce atrás do item (scaleX 0,15 → 1, origem à esquerda, 0,16 s, power3.out), o texto fica escuro (#171313) e o contador avança. No item seguinte, a barra some e o anterior ganha um ✓ laranja (escala 1,8 → 1, 0,2 s, back.out(2)).
- No último item: um anel laranja de 5 px em volta do painel (escala 1,08 → 1, 0,25 s) e o painel pulsa uma vez para 1,035.
- No CTA: o rodapé "[+ os prompts]" vira uma pílula amarela #F2C94C com texto escuro (escala 0,7 → 1, 0,25 s, back.out(2)).
- O painel não pode cobrir o rosto.
```

Dica: o painel ocupa um terço da largura. Deixe os outros efeitos do lado esquerdo e a legenda embaixo, abaixo do painel.

Erros comuns e como corrigir

- **“ffmpeg not found” ou ✗ FFmpeg no doctor:** instale o FFmpeg (passo 2) e abra um Terminal novo. A prévia pode abrir sem ele, mas o render precisa.
- **“npx: command not found” ou erro de versão:** falta o Node.js ou ele é anterior ao 22. Instale a versão LTS e confira com `node -v`.
- **O Claude não usa a skill:** abra uma sessão nova, rode `npx hyperframes skills check` e comece o pedido com “Use a skill /hyperframes”. Se instalou como plugin, o nome é `/hyperframes:hyperframes`.
- **Efeitos atrasados em relação à fala:** a transcrição foi feita no vídeo original, mas você acelerou depois. Transcreva o arquivo final (já em 1,25x) e rode o `build.py` de novo.
- **Transcrição em inglês ou com nomes errados:** use `--language pt` e, se precisar, `--model large-v3`. Corrija no JSON os nomes próprios (no Reel, “Cloud” virou “Claude” à mão).
- **Recorte com mancha ou borda ruim:** rode o `remove-background` com `--quality best`, grave com fundo liso e boa luz, e use os efeitos com recorte por pouco tempo (menos de 1,2 s).
- **Aparece na prévia e sai diferente no MP4:** algum efeito usa sorteio sem semente, relógio ou timer. Peça: “troque todo Math.random por sorteio com semente fixa e anime tudo pela timeline”. O `npx hyperframes check` aponta isso.
- **Texto cortado ou fora da área segura:** rode `npx hyperframes check`. Ele aponta sobreposição, texto pequeno e contraste. No Reel, todo texto ficou entre y 168 e y 1330, com 28 px ou mais.
- **Render lento:** para testar, use `npx hyperframes render --quality draft`. Deixe a qualidade padrão só para a versão final.
- **Vídeo do iPhone não aparece:** HEVC/MOV costumam falhar no navegador. A prévia gera uma cópia compatível sozinha (proxy). Se não gerar, converta para H.264 MP4 antes.

Fontes e outros tutoriais

Comandos conferidos em **24/10/2026** na ajuda da CLI `hyperframes@0.8.185` (init, skills, preview, render, transcribe, remove-background, check, snapshot, doctor), no instalador `npx skills` e no README oficial. Os números de cada efeito foram copiados do código dos dois Reels. A ferramenta muda rápido: se algo estiver diferente, vale a documentação oficial.

- Repositório heygen-com/hyperframes no GitHub (Apache 2.0) · documentação oficial
- Node.js · FFmpeg · GSAP: lista de eases

Continue por aqui:

- Seu primeiro vídeo com IA: abrir o Claude Code pela primeira vez, com Remotion ou HyperFrames.
- VIDEO2: sua fala vira edição: legenda que cresce, letras que mudam de forma e movimento lateral.
- Você fala. A edição acompanha.: como gravar pensando nos efeitos.

Fez algum dos 25? Siga @benites.ia no Instagram e comente no Reel qual efeito você usou primeiro.

Seguir @benites.ia