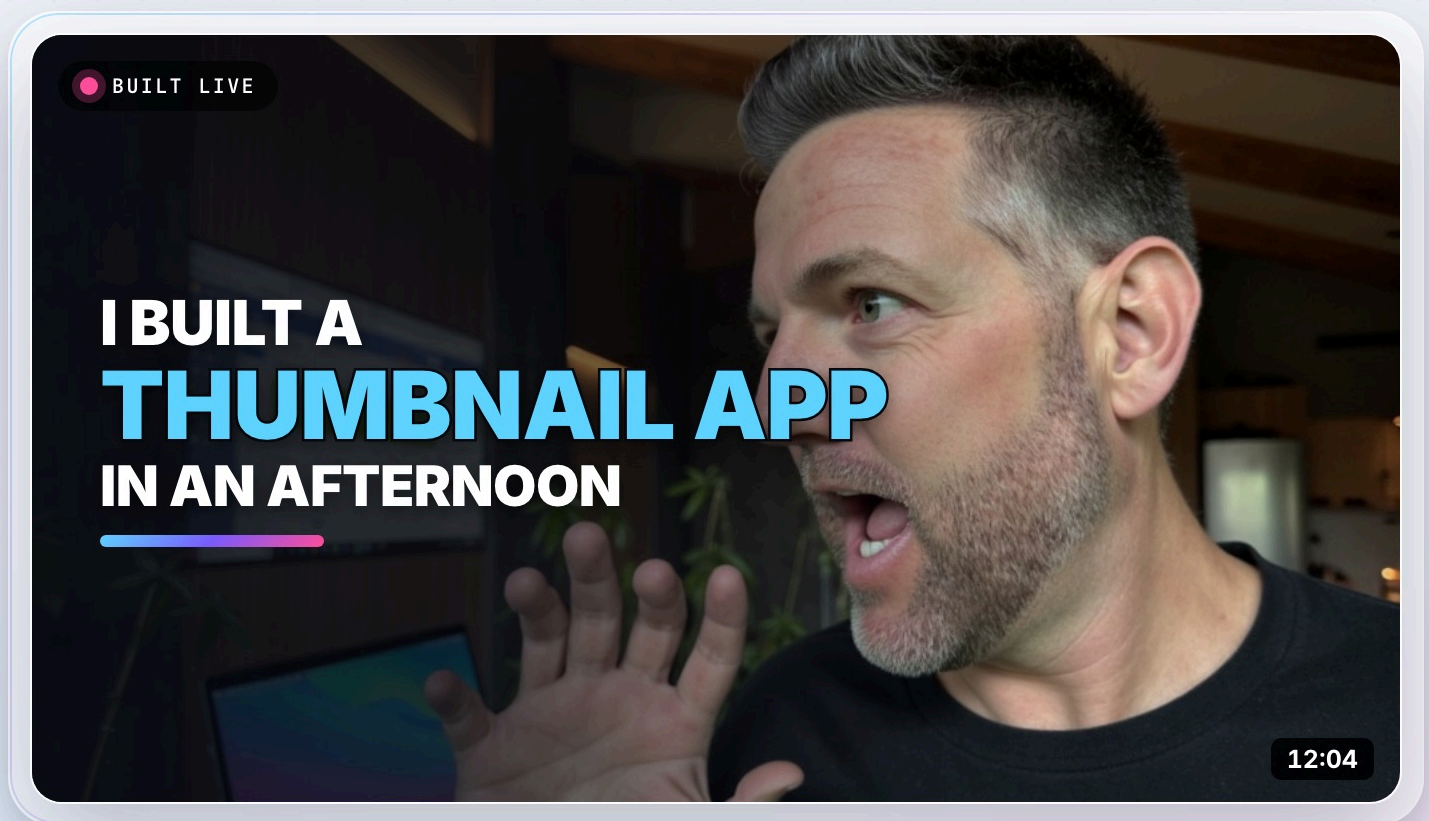


— NICK WHARTON AI · BUILD LOG

Build Your Own YouTube Thumbnail App With **Zero Code**

Every prompt you need, in order. The exact recipe to build your own, even if you've never written a line of code.



Built with Claude Code · Runs on my Mac · Zero code, start to finish

— READ ME FIRST

This is the build, prompt by prompt.

This is the real build, the exact set of prompts to make your own thumbnail app, in the order you'd type them into Claude Code. You won't write a line of code. Each step is one ask, and what it gets you.

The prompts here are templates. Swap in your channel, your face, your style. The wording can change. The order is the part that matters.

THE WHOLE LOOP IS JUST THIS

Tell it what you want



Look at what comes back



Tell it what to fix

Over and over, until it's right. That's the entire job.

WHAT YOU NEED BEFORE YOU START

THE THINKING

Claude Code

On a Max plan, so the brain that writes the code and the concepts is already paid for.

THE PICTURES

A Gemini API key

With a few dollars of billing on it. This is the part that draws the actual images.

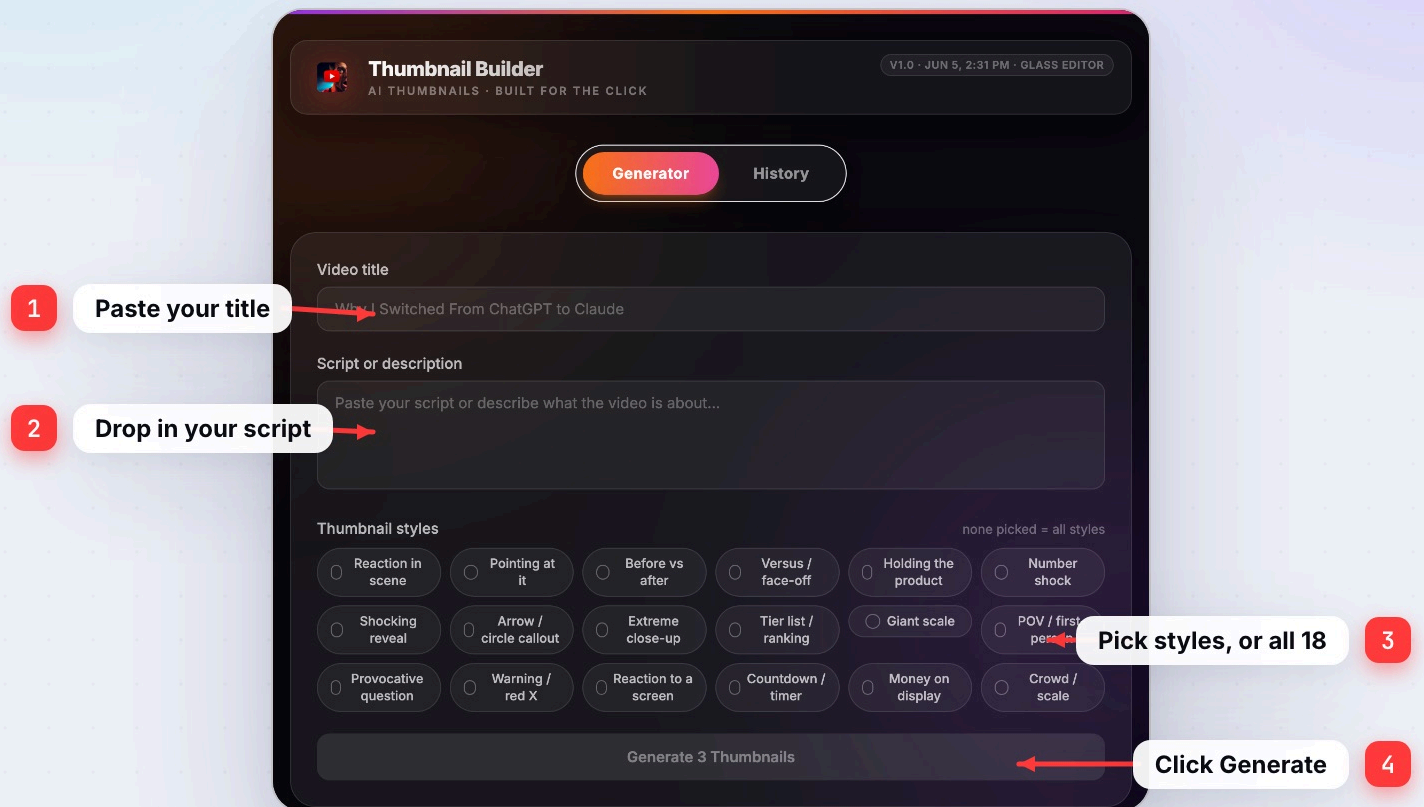
YOUR FACE

A fal.ai key

Only if you want it trained to look like you. Skip it and it still works with photos.

You paste a title. It hands you three thumbnails.

The whole thing is one screen. You fill in two boxes, click one button, and wait about a minute while it does the design work for you.



ABOUT 60 SECONDS LATER, THREE FINISHED OPTIONS ARE SITTING IN FRONT OF YOU.

— WHAT COMES OUT

Real thumbnails, about a minute each.

Every one of these came out of the app. It uses your real face, adds its own text and layout, and gives you three fresh options every run. Here are a few of mine.



YOU GET THREE OPTIONS EVERY RUN, THEN PICK ONE, NUDGE THE TEXT, AND DOWNLOAD IT.

Start by describing the app.

You don't plan it out or write any code. You tell Claude Code what you want in plain language, and it builds the first version for you to react to.

1 Describe what to build

> WHAT TO TYPE

```
Build me a YouTube thumbnail generator web app. I paste in a video title and a short description, and it gives me three thumbnail options designed to get clicks, each one with my face and bold text on it. Keep the interface clean and modern, and walk me through it step by step. I don't know how to code.
```

→ It scaffolds a working app and explains each piece as it builds, so you can follow along without knowing the tech.

2 Make the expensive part free

> WHAT TO TYPE

```
Set it up so the thinking, the concepts and the headline text, runs on my existing Claude plan, and only the image generation uses a paid model like Google's Gemini. I want running this to cost a few cents per thumbnail, not dollars.
```

→ Two brains: **Claude handles the creative direction for free on your plan**, and a paid image model draws the pictures for pennies.

You don't need an original idea or a spec. You need a clear description and a willingness to keep correcting it.

Get three real options, fast.

The first version works but the images look like AI. You fix that by being specific about what you want, and making it go learn the rules.

3 Ask for three concepts, not one

> WHAT TO TYPE

Every time I generate, give me three different thumbnail concepts at once. For each one, write a bold headline of five words or less, and a one-line note on who it's for and why it would get the click. Lay them out in a grid like a YouTube feed.

→ Three options a run, each with its reasoning, so you're picking the best, not hoping the one you got is good.

4 Force real quality and the right size

> WHAT TO TYPE

Every image has to be a true 16:9 and photo-realistic, not cartoonish. Go research what makes YouTube thumbnails get clicks (bright punchy contrast, one clear focal point, bold readable text, a real-looking background) and apply those rules to every single image.

→ Locks the YouTube size and **bakes click-tested rules into every image**, instead of you re-explaining it each time.

When something looks off, don't accept it. Name exactly what's wrong and tell it to go find the fix.

Make it look like you.

Out of the box the face won't be yours. Skip the shortcuts: plain reference photos drift once there's any movement, and face-swapping tends to go soft and blurry. The reliable way is to train a small model on close-ups of your own face.

5 Train it on your face

> WHAT TO TYPE

The faces don't look like me. Set up a face-training step: I'll give you a folder of close-up photos of my face from different angles, train a small face model on them, and use that on every image so the likeness stays consistent. Hide the photo upload from the main screen, since it's trained on me now.

→ Trains a small face model (a Flux LoRA on fal.ai) so **your likeness holds across every thumbnail**, not just a lookalike.

TRAIN IT ON PHOTOS LIKE THESE



FEED IT 20 TO 30 TIGHT CLOSE-UPS OF JUST YOUR FACE, VARIED ANGLES, NO BODY. THAT CONSISTENCY IS WHAT MAKES IT WORK.

Make it smart about scenes and styles.

6 Ask for real scenes, not clip-art

> WHAT TO TYPE

Don't put me on plain gradients with clip-art props. Build a real scene that fits the video: a relevant background, real-looking objects, a company's logo when the video is about that company, me holding or reacting to the actual thing. Keep my face close to the camera.

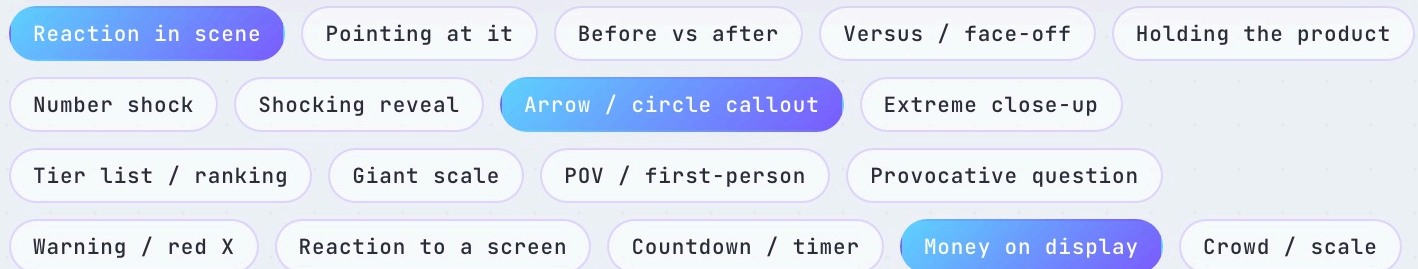
→ Thumbnails that tell a story at a glance, instead of a cut-out on a background.

7 Give it the proven thumbnail formats

> WHAT TO TYPE

Build in a set of proven thumbnail styles (reaction, before-vs-after, arrow or circle callout, tier list, number shock, versus, holding the product, and so on). Let me pick which to use, or leave it blank and you choose the best fit for the video. Show a hover example for each one.

→ A style menu the AI chooses from per video, so it isn't guessing the format.



A SOLID STARTER SET. ADD YOUR OWN ONCE YOU SEE WHAT YOUR AUDIENCE CLICKS.

Make every thumbnail editable.

A finished image you can't touch isn't enough. The step that makes it actually useful is letting you adjust the text and the picture after the fact, for free.

8 Make the text draggable and editable

> WHAT TO TYPE

Make the headline text editable right in the app. Let me click to change the words, drag them to reposition, and adjust the font, color, size and outline. Bake the final text into the downloaded image, and do it in the browser so it costs nothing extra.

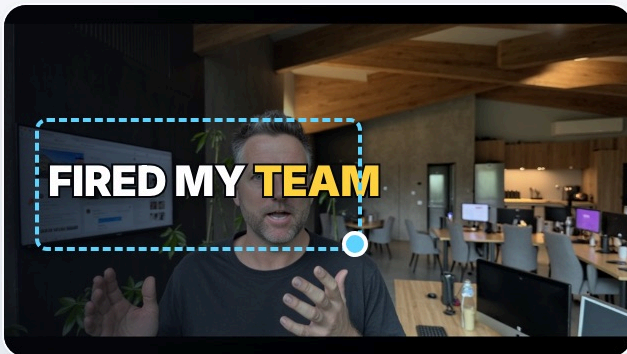
→ Full control of the text, free, no image regeneration needed.

9 Let yourself edit the picture in plain English

> WHAT TO TYPE

Under each thumbnail, give me a box to request one change in plain English (like "make the background a storm rolling in") and redraw only that, keeping everything else the same. Add buttons to generate one more, or three more variations.

→ You reshape the image by describing the change, no editing software.



Drag the text where you want it



"make the background a storm rolling in"

Edit

Turn it into a real app.

By now it works in a browser tab you have to start by hand. Two prompts make it feel like a real product you just open.

10 Get it off the command line

> WHAT TO TYPE

Package this as a proper desktop app with its own icon and a spot in the dock, always running, so I can open it any time without starting a server. Give it a logo that looks like a punchy YouTube thumbnail.

→ A standalone, always-on app, not a browser tab you babysit.

11 Add the quality-of-life touches

> WHAT TO TYPE

Add a button to generate three more options if I don't like the first set. Keep the last generation on screen when I switch tabs and come back. And restyle the whole interface to feel more modern, like frosted glass.

→ The small stuff that turns a working demo into something you'll actually reach for.

Notice the pattern across all of these: not one prompt is technical. Each is a regular person describing what would make it better to use.

The snags you'll hit (and what to say).

A few things will fight you. Each one is fixed the same way, by telling it exactly what's wrong. Keep these in your back pocket.



If dragging the text feels laggy

Tell it: "the text dragging is delayed and sketchy, switch it to pointer events and make it snappy." Then ask it to stamp a version number in a corner, so you can always confirm you're seeing the newest build.



If an edit hands back the same image

It will sometimes claim it changed something and return the identical picture. Tell it bluntly: "that gave me the exact same image, actually regenerate it instead of reusing the last one."



If text doubles up on finished images

Older saved images may already have text baked in. Tell it: "skip the editable text layer on any image that already has text on it."

WHAT IT COSTS TO RUN

The thinking · Claude

On the plan you already pay for

\$0 extra

Each picture · the image AI

A few cents an image, three options a run

~5¢

A month of real use

Thumbnails for every video you put out

~\$3 / mo



NICK WHARTON

— AI —

— YOUR TURN

That's the whole build. Now go make **your own**.

Read it back. Every step was plain language, describing what you want and fixing what looks wrong. If you can type what you want, you can build this, zero coding background and all.

Yours will come out different from mine, because your prompts will be different. That's the point.

COME WATCH ME BUILD THE NEXT ONE

 @nicolas_wharton

 @realnickwharton

 @nickwharton

I'm building this stuff in public as I go, the wins and the screw-ups both. New one every week, so come along for the ride.

Next up: the editor that cuts my videos and writes the titles, tags and descriptions for me. That one's a bit wild.