

# A Landing Page That Takes Payments

Go from idea to a real page that collects money, using AI for the build, a payment link for the checkout, and the legal basics you actually need. No developer.

---

You have something to sell. A study guide, a Notion template, a one-hour tutoring session, a design service for local businesses. The thing exists, or it could exist by tonight. What you do not have is a way for a stranger to find a page, type in their card, and pay you. That gap is the only thing between you and your first dollar online, and people treat it like it needs a developer, a year of Squarespace fiddling, or a CS degree. It does not. With AI doing the heavy lifting on the page and a payment tool that was built for exactly this, you can have a live page taking real cards by the end of an afternoon. Here is how the Desk actually does it, step by step, so you can copy it.

[atlas.elevenviews.io](https://atlas.elevenviews.io)

## 1. Decide what you are selling and pick the money model first

Before you touch any tool, get clear on three things, because every later step depends on them. What is the product, what does it cost, and is it a one-time charge or a subscription. Write it as one sentence: "I sell a Notion job-application tracker for \$19, one time." or "I sell weekly accountability calls for \$40 a month." That sentence is your whole business model and it tells the payment tool what to do.

Now pick your payment tool, because this decision shapes everything. You have three honest options for a beginner.

Stripe Payment Links: the cleanest if you have a real bank account and you are selling in the US to US buyers. No monthly fee, no extra charge for the link itself. You pay the standard 2.9% plus 30 cents per card charge. So on a \$19 sale you keep about \$18.15. The catch: you are the merchant of record, which means sales tax is your problem (more on that in the legal section, and for most students starting out it is a non-issue until you hit volume).

Lemon Squeezy: this is what we reach for when you sell digital products (templates, guides, courses) and you do not want to think about tax at all. They are the merchant of record, meaning they collect and remit sales tax and VAT for you, worldwide. The cost: 5% plus 50 cents per sale on top of card processing. On a \$19 sale you keep roughly \$17.05. You pay more, but you buy yourself out of the entire tax headache, which for a 20-year-old selling to people in five countries is worth it.

Gumroad: the fastest to set up and the most forgiving if you have zero infrastructure. Also a merchant of record, so tax is handled. The cost is higher: 10% plus 50 cents per sale. On a \$19 sale you keep about \$16.60. Use it when you want zero friction and you are testing whether anyone will buy at all.

The Desk rule: selling a digital download to a broad audience, start with Lemon Squeezy. Selling a service or you already have Stripe, use Stripe Payment Links. Just want to test demand tonight with no setup, use Gumroad. You can always switch later.

## 2. Create the payment link before you build the page

This is the step most people do last, and it is why their page sits half-finished for weeks. Do it first. The payment link is the actual money machine. The page is just the wrapper. Build the machine, then wrap it.

Stripe Payment Links: go to [dashboard.stripe.com](https://dashboard.stripe.com), finish the account setup (legal name, address, bank account for payouts, this takes about ten minutes and a debit card or bank login). In the left menu click Payment Links, then Create payment link. Choose "Products" and add a new product: name, price, one-time or recurring. Hit Create link. You now have a URL like [buy.stripe.com/xxxxx](https://buy.stripe.com/xxxxx) that opens a checkout page Stripe hosts and secures. Test it in test mode first (toggle in the dashboard), then switch to live.

Lemon Squeezy: sign up at [lemonsqueezy.com](https://lemonsqueezy.com), create a Store, then click New Product. Set it as a digital product, upload the file you are delivering (the PDF, the template, the zip), set the price, and publish. Lemon Squeezy gives you a hosted checkout URL and even handles delivering the file to the buyer automatically after payment. You do not build a delivery system. They do it.

Gumroad: sign up at [gumroad.com](https://gumroad.com), click New product, pick "Digital product," upload your file, set a price, publish. You get a [gumroad.com/l/xxxx](https://gumroad.com/l/xxxx) link instantly. Same deal: they host checkout and deliver the file.

Whatever you pick, you now have one thing that matters: a working URL where someone can pay you. Open it on your phone, run a real \$1 test charge on yourself if you can, confirm the money shows up. Once that works, the hard part is over.

03

### 3. Write the page copy with AI before you design anything

A page that takes payments needs words that make someone want to pay. Design is the easy part now that AI does it. Copy is what converts. Spend twenty minutes here.

Open Claude or ChatGPT and give it the prompt in the swipeable below. The trick is to feed it real specifics: who the buyer is, the exact problem, what they get, the price, and proof if you have any (a testimonial, a number, your own result). Generic in means generic out. Specific in means a page that sounds like a real person made it for a real buyer.

You want the AI to give you these blocks, in this order, because this is the structure that sells on a single page:

- **Headline:** the outcome the buyer gets, not the name of your product. "Stop rewriting your resume for every job" beats "Resume Tracker Pro."
- **Subhead:** one sentence that says who it is for and what it is.
- **Three to five bullets:** what they actually get, concrete.
- **Proof:** a testimonial, a stat, or an honest line about why you built it.
- **The offer:** price, what is included, the button text.
- **A short FAQ:** the two or three objections that stop a buyer (refunds, format, time to deliver).

Read what the AI gives you out loud. If a sentence sounds like a brochure, cut it. You are talking to one person who is about to spend money. Keep it plain.

04

### 4. Build the page itself with an AI site builder

Now the visual page. You have two good paths depending on how much control you want. Both get you live today.

The fastest path, Carrd: go to [carrd.co](https://carrd.co). Pick a one-page template, drop in your AI-written copy, and add a button. Set the button to link to your payment URL from step 2. That is it. Carrd is \$19 a year for the pro features (custom domain, forms) and it is the lowest-friction real website you can ship. A complete payment-taking page in under an hour, no code.

The AI-generated path, Vercel v0: go to v0.app and describe the page in plain English. Paste the prompt in the swipeable. v0 generates a real, modern-looking page as actual code, shows you a live preview, and lets you tweak by chatting ("make the button bigger," "change the headline color to deep blue"). When it looks right, click Deploy and v0 puts it live on a real URL for free. You drop your payment link into the button. This gives you a page that looks custom-built without you writing a line of code.

The in-between path, Framer: framer.com has an AI generator (Framer AI) that builds a page from a prompt, with more design polish than Carrd and less fiddling than raw code. Free to start, paid plan for a custom domain.

The Desk rule: testing an idea fast, use Carrd. Want a page that looks like a funded startup made it, use v0. Either way, the one job of this page is to get someone to click the button that goes to your payment link. Do not add a menu, a blog, an "About" novel. One page, one offer, one button.

05

## 5. Connect the button to the money and test the whole flow

This is where a page becomes a business. Every visual builder has the same move: select your main button, find the link or URL field, and paste your payment link from step 2.

In Carrd: click the button element, in the right panel set Type to "Link" and paste the URL. In v0: tell it "make the button link to [your payment URL] and open in a new tab," or edit the href in the code it generated. In Framer: select the button, open the Link panel, paste the URL.

Now run the full test as if you were a stranger. Open your live page on your phone, on cellular not wifi, so you see what a real visitor sees. Read the page top to bottom. Click the button. Confirm it opens the checkout. If you can, do one real purchase of a \$1 test product, or use Stripe test mode, and confirm three things: the charge succeeds, the buyer gets what they paid for (the file, the booking link, the confirmation), and the money appears in your dashboard.

This end-to-end test is non-negotiable. A broken button is the single most common reason a page that "went live" never makes a dollar. Test it on someone who is not you. Watch them try to buy without help. Whatever confuses them, fix that.

06

## 6. Handle the legal basics, the honest minimum

You are taking real money from real people, so you need a few things. This is not legal advice, it is the practical floor that keeps you out of trouble and keeps the payment companies happy. Stripe, Lemon Squeezy, and Gumroad can all freeze an account that has none of this.

The minimum every payment page needs:

1. A refund policy. State it plainly on the page. "Digital product, no refunds" is legal and common, but say it clearly so a buyer is not surprised. For services, "full refund within 7 days if we have not started" is fair. Buyers trust a page that is upfront about this.

2. Terms of Service and a Privacy Policy. You can generate solid starter versions for free. Termly.io and GetTerms.io both have free generators where you answer a few questions and get pages you can link in your footer. If you use Lemon Squeezy or Gumroad as merchant of record, they cover a lot of this, but having your own linked pages still helps.

3. Contact info. A real email address on the page. Buyers and payment processors both want to see that a human is reachable.

4. Sales tax. If you use Stripe and you are the merchant of record, you are technically responsible for collecting sales tax once you cross certain thresholds in a state (in most states that is tens of thousands of dollars or 200 transactions, so most people starting out are well under it). If that worries you at all, this is the single best reason to use Lemon Squeezy or Gumroad: as merchant of record, they handle tax collection and remittance for you, everywhere. For a student making their first sales, Stripe is fine and you are almost certainly under any threshold. Just know the rule exists.

Use the AI prompt in the swipeables to draft your refund policy and a plain-English FAQ in two minutes. Add a footer with links to your Terms, Privacy, and refund policy. Now you look like a real business, because you are one.

07

## 7. Get a domain and ship it (then drive traffic)

A page on carrd.co or vercel.app already works and takes money. But a custom domain costs about \$12 a year and makes you look real. Buy one at Namecheap or Cloudflare (Cloudflare sells at cost, slightly cheaper). Then connect it: Carrd, v0, and Framer all have a one-click custom domain setting that walks you through pointing the domain at your page. It takes a few minutes and a copy-paste of a DNS record.

Then, the part nobody tells you: a live page with no traffic makes zero dollars. The page is step one. Now you need eyeballs on it. The cheapest traffic for this audience is your own network and short video. Post the page link in the relevant places: your group chats, the right subreddit, a Discord for your niche, a LinkedIn post if it is professional, a TikTok or Reel if it is visual. The short at the end of this guide is a template for exactly that. One good video pointing at one good page is how the first sales actually happen.

Watch your first ten visitors. Most payment tools and v0 show you basic analytics. If people land and leave without clicking the button, your headline or price is the problem, not your design. Change the words, not the colors. Re-test. The whole loop, idea to live page to first sale, is something you can run in a weekend and repeat for every new thing you want to sell.

08

## How the Desk turns this PDF into the companion video

The Desk reads this guide as a shot list. Each of the seven sections becomes a chapter in the video, and each swipeable becomes an on-screen moment where the prompt or checklist appears full-frame so viewers can pause and copy it. The narration agent (Aura) writes the voiceover from the section bodies, keeping the same plain tone. The screen-capture agent records the real flows: a Stripe Payment Link being created, a v0 prompt generating a page live, a button being linked, and a test

purchase going through. The point of the video is to show the clicks the PDF can only describe. Nothing is faked or mocked up. Every screen is the real tool on the real day, so a viewer following along sees the exact buttons in the exact places.

09

## Companion video narration script (full)

[COLD OPEN, 0:00] You have something to sell. Maybe it exists, maybe you'll make it tonight. The only thing missing is a page where a stranger types in their card and pays you. Let's build that. Live, in one sitting, no developer.

[CHAPTER 1, 0:12] First, one sentence. What you sell, the price, one-time or monthly. Say it out loud: "I sell a Notion job tracker for nineteen dollars, one time." That sentence decides everything next. Now pick your money tool. Three options. Stripe Payment Links if you've got a bank account and you're selling in the US. You keep about eighteen-fifteen on a nineteen-dollar sale, and you handle your own tax. Lemon Squeezy if you're selling a digital file and you never want to think about tax: they're the merchant of record, they handle it worldwide, you keep around seventeen. Gumroad if you just want to test demand tonight with zero setup: easiest, but you keep about sixteen-sixty. Digital product to a wide audience, start with Lemon Squeezy. A service, use Stripe. Just testing, use Gumroad.

[CHAPTER 2, 0:55] Now the move most people get wrong: build the payment link first, before the page. Watch. In Stripe, finish account setup, click Payment Links, Create, add your product, price, one-time. Hit create, and there's your URL. In Lemon Squeezy, new product, upload your file, set the price, publish, and it delivers the file to the buyer for you. In Gumroad, same thing, even faster. Whatever you pick, you now have one link where someone can pay. Open it on your phone. That link is the business. The page is just the wrapper.

[CHAPTER 3, 1:40] Words make people pay, so we write them before we design. Open Claude. Feed it the real stuff: who buys, the problem, what they get, the price, any proof. Specific in, specific out. You want a headline about the buyer's outcome, a subhead, four concrete bullets, one proof line, the offer, a short FAQ. Read it out loud. If it sounds like a brochure, cut it. You're talking to one person who's about to spend money.

[CHAPTER 4, 2:20] Now the page. Fastest path, Carrd: pick a template, drop in your copy, add a button, nineteen dollars a year. Sharpest path, Vercel v0: describe the page in plain English, watch it generate real code with a live preview, tweak by chatting, click deploy, it's live for free. We'll use v0 here. Paste the prompt, and there's a page that looks like a startup built it. One page, one offer, one button. No menu, no blog, no About novel.

[CHAPTER 5, 3:00] This is where a page becomes a business. Select the button, paste your payment link. In v0, just tell it: link the button to this URL, open in a new tab. Now test the whole thing like a stranger. Phone, cellular, not wifi. Read it, click the button, confirm the checkout opens, run a real one-dollar test, confirm the money lands. A broken button is the number one reason a live page never makes a dollar. Watch someone who isn't you try to buy. Whatever confuses them, fix it.

[CHAPTER 6, 3:45] You're taking real money, so cover the basics. A refund policy in plain words on the page. Terms and a Privacy policy, generate free starters at termly.io. A real contact email. And sales tax: with Stripe you handle it, but you're almost certainly under the threshold starting out. Want it

handled for you, that's the reason to use Lemon Squeezy or Gumroad. They're merchant of record, tax is their job. Add a footer with those links and now you look like a real business, because you are one.

[CHAPTER 7, 4:25] Last, ship it. Twelve dollars for a domain at Cloudflare or Namecheap, one-click connect in your builder. Then the part nobody says: a page with no traffic makes zero dollars. Post the link where your people are. The short on our free page is a template for exactly that. Watch your first ten visitors. If they leave without clicking, change the words, not the colors. Idea to live page to first sale, one weekend, every time.

[CLOSE, 4:55] The full step-by-step, every prompt, every checklist, is in the free pack. Grab it, build your page, and go get your first dollar. Link below.

10

## Scene map

Scene 1 (0:00-0:12) Cold open: hand holding a phone showing an empty Stripe dashboard, text overlay "idea to paid in one sitting." Scene 2 (0:12-0:55) Split screen of the three payment tools' logos with the keep-rates animating in. Scene 3 (0:55-1:40) Screen recording: real Stripe Payment Link creation, then Lemon Squeezy product upload, ending on a checkout URL opening on a phone. Scene 4 (1:40-2:20) Claude window with the copy prompt typed, output blocks highlighting one by one. Scene 5 (2:20-3:00) v0.app generating the page from the prompt, live preview building in real time, deploy click. Scene 6 (3:00-3:45) Button being linked, then a phone running a \$1 test purchase, success screen, dashboard showing the charge. Scene 7 (3:45-4:25) Termly generator, footer links appearing, refund policy text. Scene 8 (4:25-4:55) Cloudflare domain purchase, domain connecting, then a social post with the link. Close (4:55-5:10) Free-pack end card with QR code and atlas.elevenviews.io.

11

## Asset checklist

Screen recordings (real, 1080p min): Stripe Payment Link creation; Lemon Squeezy product+file upload; Gumroad quick product; Claude copy prompt and output; v0 page generation and deploy; button link step in v0/Carrd; live \$1 test purchase end-to-end including success and dashboard; Termly policy generator; Cloudflare/Namecheap domain purchase and connect. Graphics: three-tool comparison card with keep-rates (\$18.15 / \$17.05 / \$16.60 on a \$19 sale); seven-chapter lower-third titles; the four swipeable prompts/checklist as full-frame pause cards. Audio: Aura VO track; light underscore bed, no lyrics; soft UI click SFX on button moments. Brand: Eleven Views logo bug, gold accent, end card with QR to atlas.elevenviews.io. Captions: burned-in SRT for the whole runtime (this audience watches muted).

12

## Render recipe

Master at 1920x1080, 30fps, H.264, target 8-10 Mbps for the long-form companion. VO recorded at 48kHz, normalized to -16 LUFS for web. Screen recordings captured at native resolution then scaled, with a subtle 1.05x slow zoom on key clicks so the button being pressed is unmistakable. Pause cards (the swipeable prompts) held on screen 4 full seconds each with a "pause to copy" nudge. Cut to the beat of the narration, no shot under 1.5 seconds except in the cold open. Color: clean, slightly warm,

Eleven Views gold (#hex from brand kit) only on titles, buttons, and the end card. Export a 16:9 master for YouTube and a separate 9:16 cutdown sourced from the same recordings for the short. Deliver MP4 plus the SRT caption file.

## Swipe file

Copy, paste, adjust. These are the exact prompts and templates.

### AI PROMPT: WRITE THE FULL SALES PAGE COPY

Paste into Claude or ChatGPT, fill the brackets:

"You are a direct-response copywriter. Write the copy for a one-page landing page that sells [PRODUCT] for [\$PRICE], [one-time / per month]. The buyer is [WHO: e.g. college students applying to internships]. The problem they have is [PROBLEM]. What they get is [LIST WHAT'S INCLUDED]. My proof is [TESTIMONIAL / RESULT / WHY I MADE IT, or write 'none yet'].

Give me, as labeled blocks: 1) a headline focused on the buyer's outcome not the product name, 2) a one-sentence subhead saying who it's for, 3) four concrete bullets of what they get, 4) one short proof line, 5) the offer block with price and button text, 6) a 3-question FAQ covering refunds, format, and delivery time.

Rules: talk to one person. Short sentences. No corporate words. No hype. Plain English a 20-year-old would actually say."

Then read it out loud and cut anything that sounds like a brochure.

### AI PROMPT: GENERATE THE PAGE IN VERCEL V0

Paste into v0.app (v0.app):

"Build a single-page landing page to sell a digital product. Modern, clean, lots of white space, one accent color [pick: deep blue / warm orange]. Sections in this order: a big headline and subhead with one primary button, a row of 4 benefit bullets with small icons, a short proof/testimonial card, a pricing block showing [\$PRICE] with a second button, a 3-item FAQ accordion, and a small footer with links labeled Terms, Privacy, Refunds and a contact email. Make both buttons say [BUTTON TEXT] and link to [PAYMENT URL], opening in a new tab. Mobile-first, fast, no stock-photo clutter."

Then refine by chatting: "tighten the spacing," "make the headline bigger," "deploy it."

### CHECKLIST: BEFORE YOU CALL THE PAGE LIVE

- [ ] Payment link works on its own (tested with a real \$1 or Stripe test charge)
- [ ] Page button opens the checkout (tested on phone, on cellular)
- [ ] Buyer receives the product/booking automatically after paying
- [ ] Money confirmed landing in your dashboard
- [ ] Headline is about the buyer's outcome, not the product name

- Price is visible without scrolling forever
- Refund policy stated in plain words on the page
- Footer links: Terms, Privacy, Refunds, contact email
- One person who is not you bought it without needing help
- Custom domain connected (optional but looks real)

#### AI PROMPT: DRAFT REFUND POLICY + LEGAL FAQ

Paste into Claude or ChatGPT:

```
"Write a short, plain-English refund policy and a 3-question legal FAQ for a landing page selling [PRODUCT] for [$PRICE]. It's a [digital download / live service]. My refund stance is [no refunds on digital / 7-day refund if not started / your choice]. Keep it honest and simple, no legalese, 2-4 sentences for the policy. Then write FAQ answers for: 'Can I get a refund?', 'How do I receive what I bought?', 'How do I contact you?' Use [YOUR EMAIL] as the contact."
```

Note: for full Terms of Service and Privacy Policy, use the free generators at [termly.io](https://termly.io) or [getterms.io](https://getterms.io) and link them in your footer.

#### VO DIRECTION NOTES FOR AURA (THE NARRATION AGENT)

Voice: calm, direct, a sharp older friend who has actually done this, not a hype channel. Pace: conversational, slightly faster on the lists, slow down on the one money sentence and the 'test the whole flow' warning. Emphasize: 'build the payment link first', 'one page, one offer, one button', 'a broken button is the number one reason a live page never makes a dollar', 'change the words, not the colors'. Never read URLs as 'h-t-t-p'; say 'v0 dot app', 'termly dot io'. No exclamation-point energy. End on the free-pack CTA, warm not salesy.

#### ON-SCREEN TEXT OVERLAYS (IN ORDER)

0:00 "Idea to paid. One sitting." / 0:30 "Keep on a \$19 sale: Stripe \$18.15 | Lemon Squeezy \$17.05 | Gumroad \$16.60" / 0:55 "Build the payment link FIRST" / 1:40 "Words make people pay" / 2:20 "One page. One offer. One button." / 3:00 "Test it like a stranger." / 3:15 "A broken button = \$0" / 3:45 "Cover the legal basics" / 4:25 "No traffic = no money" / 4:50 "Change the words, not the colors" / 4:55 "Free step-by-step pack: [atlas.elevenviews.io](https://atlas.elevenviews.io)"

## Want this built into your business?

If this saved you time, that is the whole idea. The full walkthrough is the paid video track, and if you want it built for you, that is what the Desk does. Book a call at [atlas.elevenviews.io/book](https://atlas.elevenviews.io/book).