Cluma

PAY-WORTHY

The Pay-Worthy Product Playbook. For startups that built their MVP with AI.
How to turn it into a product customers trust and pay for?



PLAYBOOK
2026

SKILLS
5

BY
Cluma

Built your MVP with AI?

Then every screen *probably* looks a little different.

A product that looks stitched together is a hard product to pay for.

AI made building fast, not consistent. Each screen your team generates in Cursor, Lovable or Claude Code is built alone, with no shared components.

Put them side by side: a different button here, a different gray there, spacing that changes from page to page. Your customers notice.

Your developers notice too. They rebuild the same button for every new screen and fix one change in many places. That is **design debt**, and it grows with each screen you ship.

Two in five hours of a developer's week go to old mistakes.

Stripe asked developers how they spend a 41.1-hour week: 17.3 hours went to technical debt and bad code (Stripe, 2018). Design debt is the same problem on screen. Every messy screen brings its own button, gray and spacing, so one change has to be fixed everywhere.



SURVEY

17.3 of 41.1 weekly hours (Stripe, 2018)

- Debt and bad code
- Everything else

NOBODY HAS TIME TO *clean up.*



Renaming colors, merging four buttons into one, fixing the spacing on 60 screens: tedious work that never feels urgent.

But it has to happen. Every week it waits, AI builds new screens on top of the mess. And a product on one system simply looks more professional.

Check your own *screens first.*

Before you clean anything up, see how far your screens have drifted. Paste this prompt into Claude Code or Cursor at the root of your repo:

Audit this product for UI consistency. Score each check 0 (bad), 1 (partly) or 2 (good) and show the evidence:
1. Colors: every distinct value. Do they all come from one palette?
2. Type: how many font sizes are in use? Fewer than 8 is good.
3. Spacing: values that are not on a 4 or 8 px scale.
4. Buttons: every button style. One component, or a new one per screen?
5. Duplicates: modals, inputs, cards or tables that exist twice.
6. States: screens without empty, loading or error states.
7. Tokens: the share of hardcoded style values.
8. Figma: ask me whether a Figma file matches what we ship.
9. AI rules: is there a rules file that points our AI tools to the system?
10. Fixes: ask me how many of last sprint's frontend tickets were UI fixes.
Return one table and the total out of 20. Do not change any code.

Copy the prompt



WHAT YOUR TOTAL SCORE REALLY MEANS

Add up your 10 checks: 0 (bad), 1 (partly) or 2 (good) each. The total tells you how far your screens have drifted.

At 16 to 20 you ship at full speed. At 9 to 15 every new feature drifts further and fixing it eats speed. Below 9, each sprint makes the next one slower.

Wherever you landed, the rest of this playbook shows how to get every screen back on one system.

Three bands for your total:

16–20 CONSISTENT

9–15 DRIFTING APART

0–8 COMPOUNDING



CONTENTS

Copy *what* you need.

01 **Where speed gets stuck** p. 08

AI made engineering fast. Design didn’t keep up

02 **How to make every screen match** p. 14

Into Figma, then five steps from plan to developer spec

03 **More bets per month** p. 29

Test more ideas, then see the before and after

04 **Or let us do it** p. 35

What clients say and a first call

PART 01

THE PROBLEM

AI made engineering fast. Now design is the bottleneck.

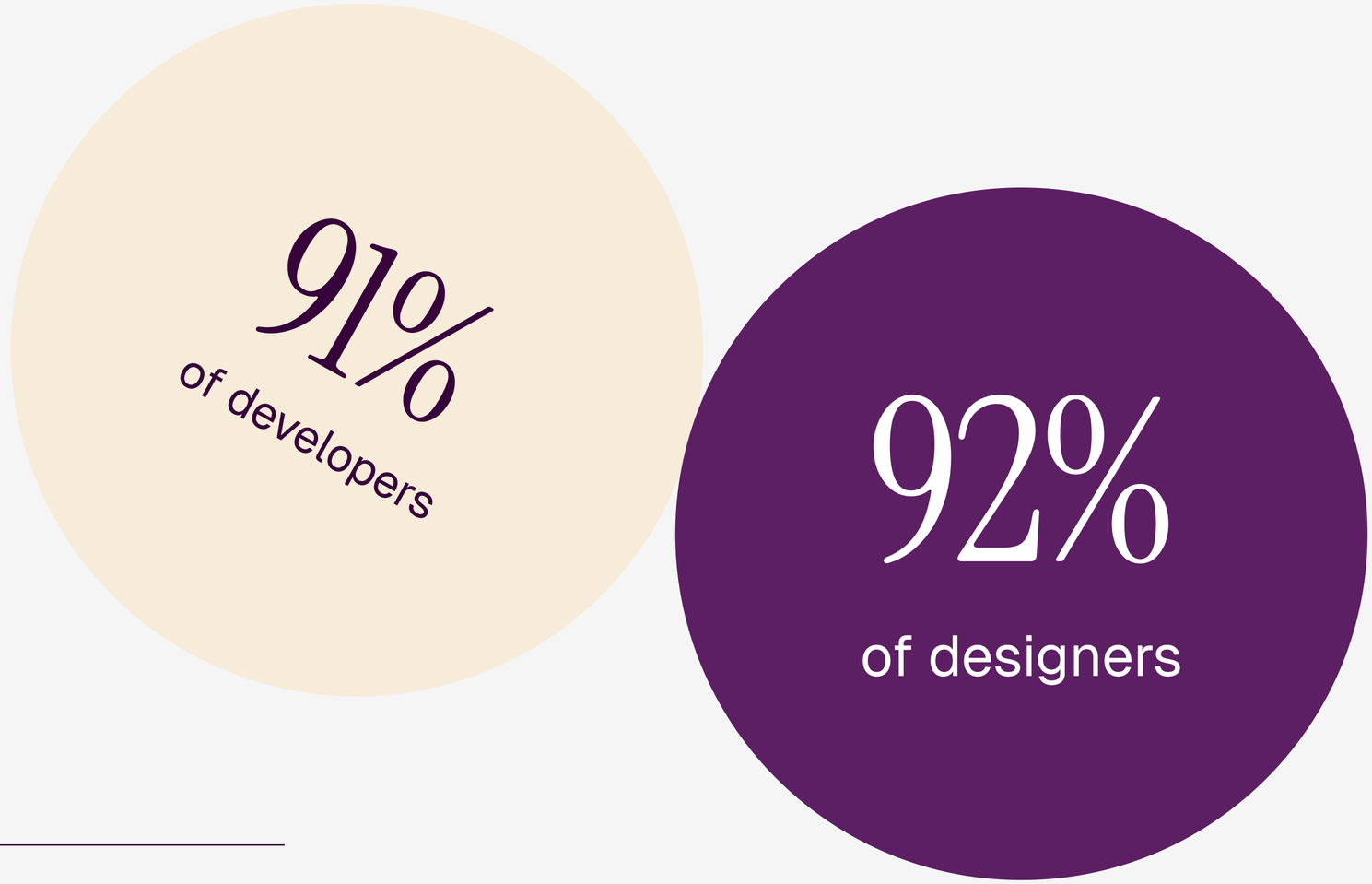


Cluma

Then design became the *bottleneck.*

Your developers open Cursor in the morning, and by lunch a new feature runs. A flow used to take a week to design and three weeks to build. Now the build takes two days, and the ticket waits for design.

Both sides feel it: 91% of developers and 92% of designers say the handoff between them needs improvement (Figma, 2025).



WHAT HAPPENS NEXT

So someone stops waiting and lets the AI design it too.

Purple, rounded primary button

A slightly different gray

Blue, square button

Green button on billing

System font in reports

16 px padding

Gray #71717A

Gray #6B7280

24 px padding

Gray #64748B

Rounded cards

Square cards

Card shadows

No card shadows

Icons from two libraries

Three modal styles

14 px body text

Inter in onboarding

15 px body text

Two date formats

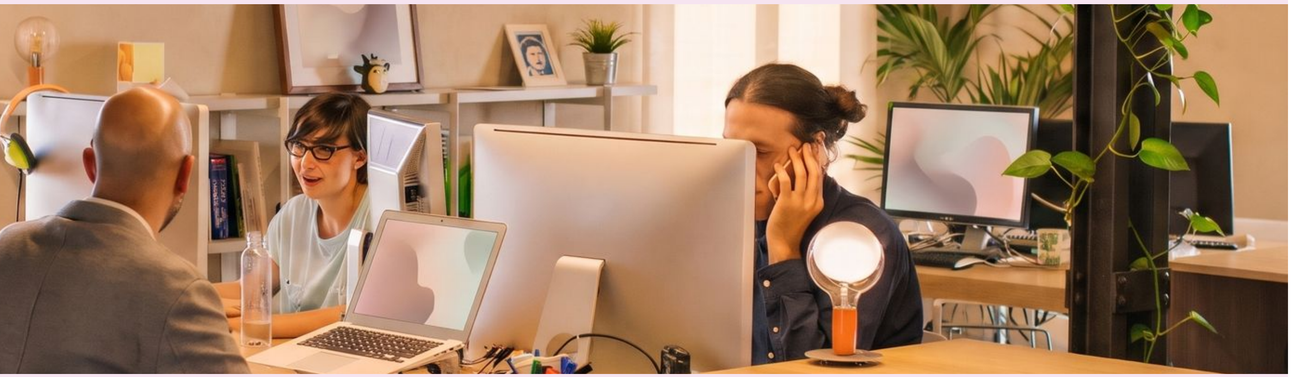
Four button styles

Outline and filled inputs

22

of these were never decided.
The AI picked each one,
one prompt at a time.

THE PROTOTYPE WAS THE EASY PART



Getting it to run was fast. Getting it ready for real users is the slow part, and it lands on your developers.

Cluma

“I built the prototype in an hour. Getting it ready for real users took about a hundred.”



Puzzled Flicker
Part-time button counter

Every new screen starts with the question which of the four buttons is right. Every fix in one place breaks the look somewhere else.

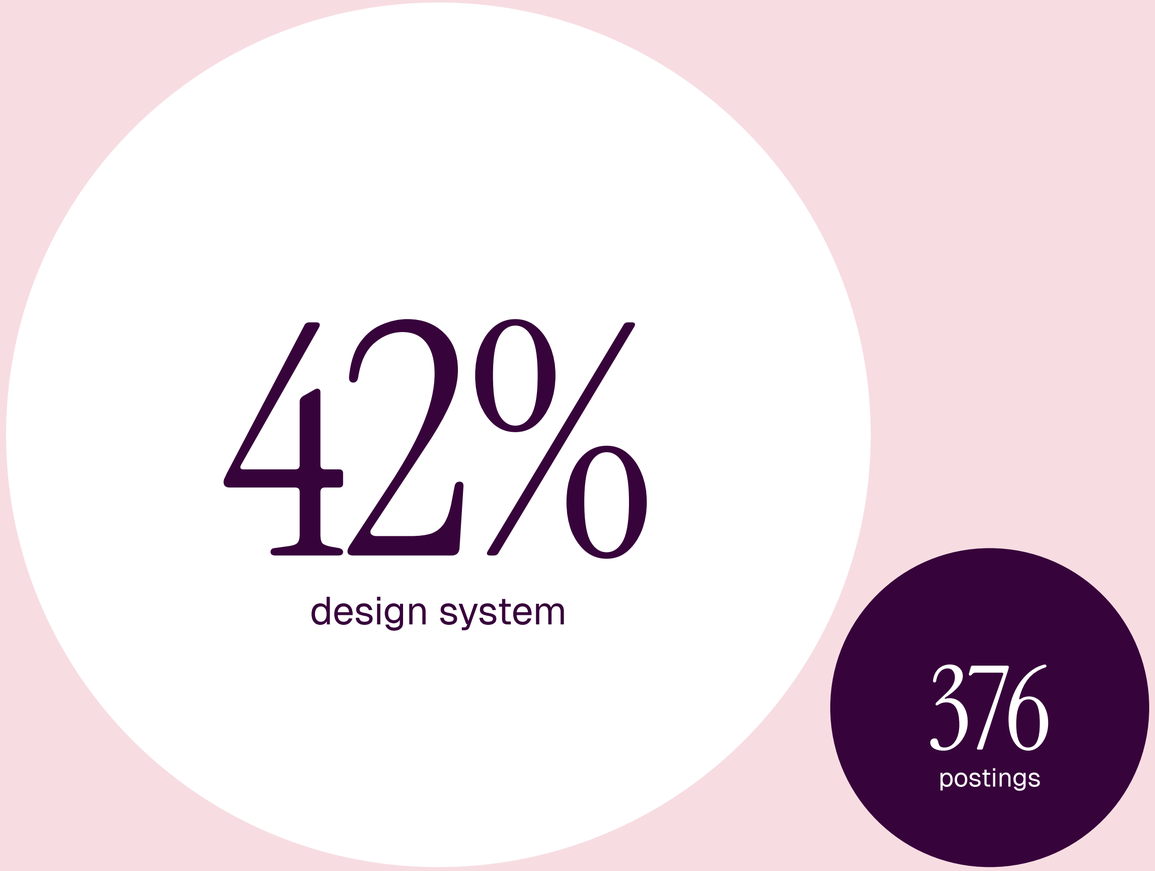
You are not the only one:

376 job postings.

~30%

of the postings ask for a designer who can keep up with engineering pace (Cluma, 2026).

We analyzed 376 design job postings from startups with fewer than 200 employees, 2023 to October 2026.



AND THE BIGGEST ASK

42% hire a designer to build a design system, so every screen stays consistent (Cluma, 2026).

MESSY SCREENS GET PAID FOR TWICE.

ONCE TO BUILD IT

ONCE TO FIX IT

SAME BUTTON, BUILT 4 TIMES

ONE COLOR, 12 FILES

FIX HERE, BREAKS THERE

REBUILT AGAIN

ONE MORE GRAY

BUILT FAST, FIXED SLOW

NEW SCREEN,

NEW SPACING

DEMO CLEAN-UP



Cluma

Cluma

So how do you make every screen *match?*





FIRST, GET YOUR SCREENS INTO FIGMA

Your product lives in Lovable, a Claude artifact or straight in code. None of it is in Figma yet. Figma shows every screen side by side and gives your developers one system to rebuild from. (Noodle counted the grays twice. He hasn’t slept since.)

Capture tools bring them in as loose frames with no components. Clean layers need skills that know your setup, and ours took months to get reliable. Ours then merge the duplicate buttons, name the grays and unify the spacing for you.

One long chat forgets what it decided after a few screens. So we split the clean-up into five steps, each run by a skill we trained for months.

Claude Code works right inside your Figma file, one fresh session per screen.

Research and plan

01

Reads every screen:
a build plan you review

Set up the file

02

Pages, colors, type
and a component library

Build

03

Every screen rebuilt
on the system

Audit and fix

04

Checks each screen against
our rules, fixes it



Consolidate

05

Smallest component
set, tokens, developer
spec

Cluma

Research and *plan.*

Read every screen, then plan the build.



01

SKILL 01 · RESEARCH AND PLAN

Nothing gets built before you have seen the *plan.*

01

Research

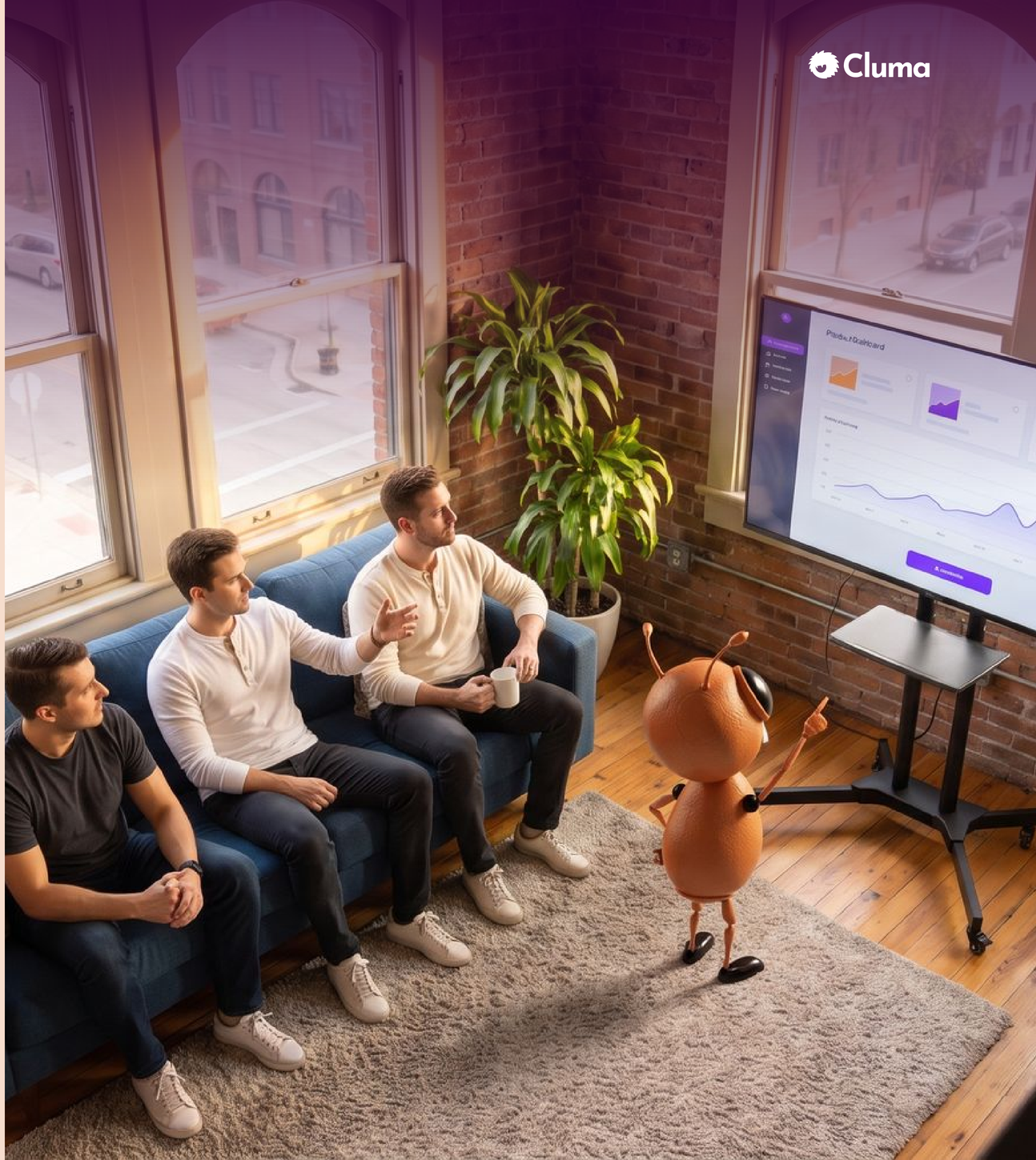
Reads every screen and lists what repeats: buttons, cards, inputs, colors, type sizes and spacing. An existing Figma library is checked first, so nothing gets built twice.

02

Plan

Lists every screen in build order, shared components first. You review it, change the order or cut screens, and only then does the build start.

Cluma



Cluma

Set up the *file.*

Pages, tokens and a component library first.

02



Setup lays four layers, and each one is built from the one below:

01 Tokens, defined once as Figma variables



02 Components, each built once

03 Patterns, built from components

04 Screens: your product, from patterns

Cluma

SKILL

02

Tokens: colors, type sizes, spacing, radii and shadows. Components: buttons, inputs, selects, modals, tables, cards, empty states. Patterns: forms, lists, settings pages, onboarding steps.

Cluma

Build every *screen.*

One fresh session per screen, on the system.

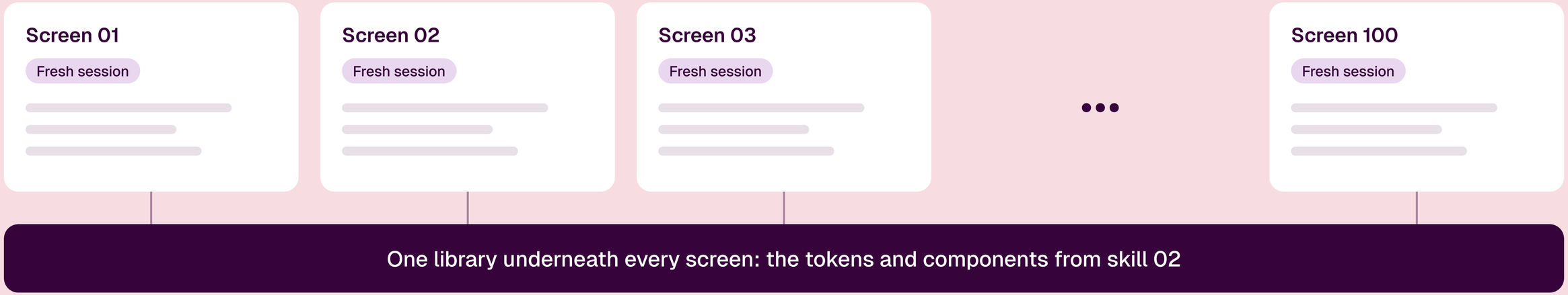
03





Every screen, *rebuilt.*

The build skill rebuilds your screens in Figma one at a time. Each screen gets a fresh session that reads the plan and what earlier screens taught it, so screen 100 follows the same rules as screen 1.



No new colors, no one-off spacing, no third modal.

Library falls short? The skill stops and flags it.

Nothing gets invented on the fly, so you review gaps, not guesses.

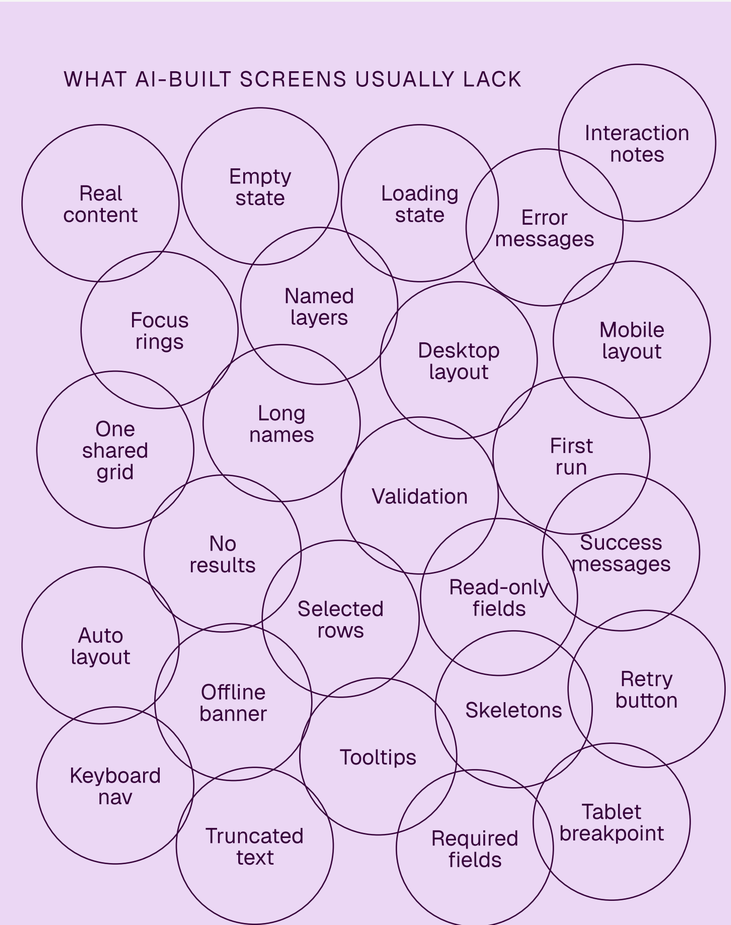
What it *adds*

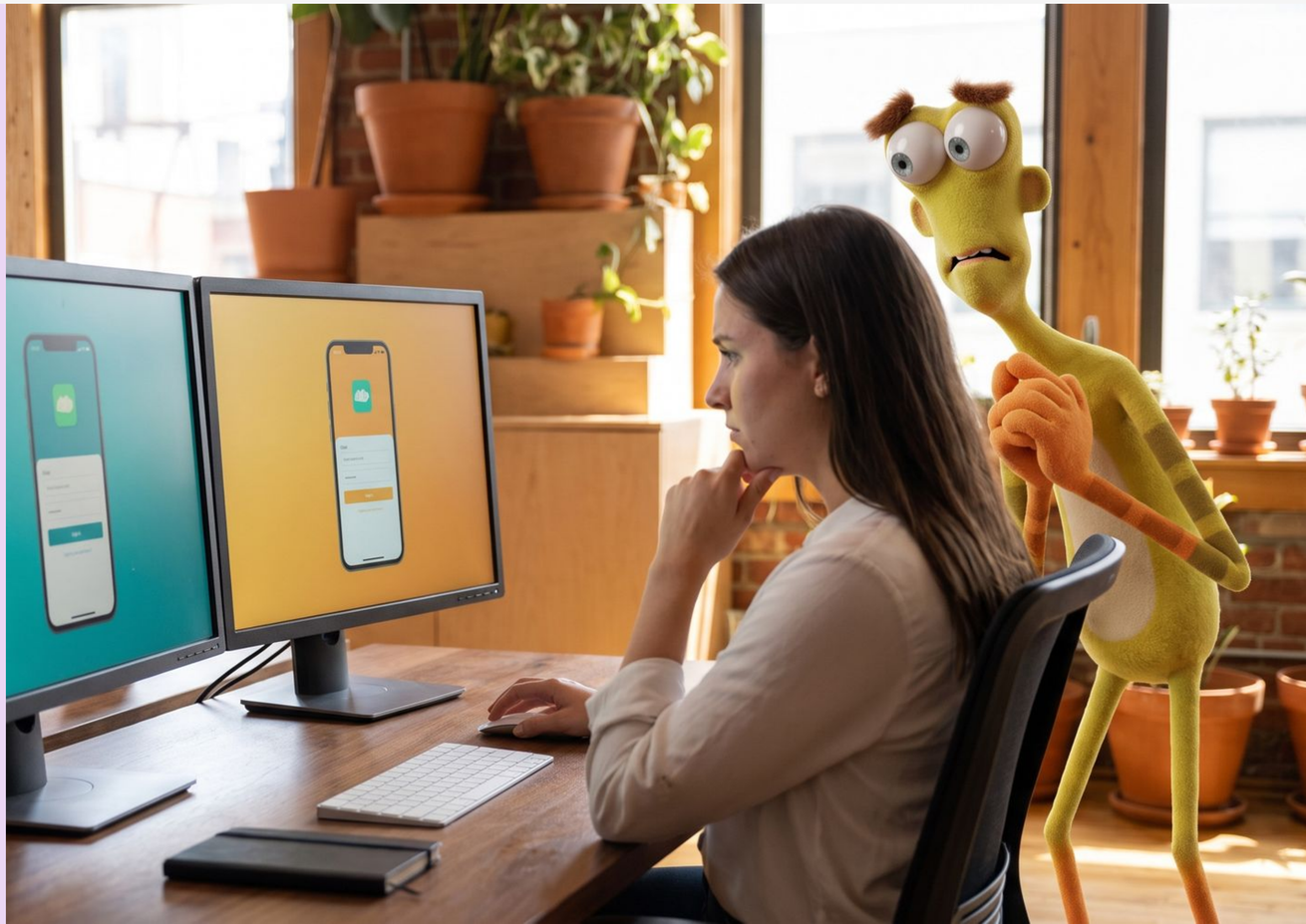
Real content instead of placeholder text, empty, loading and error states, notes for every interaction, and desktop and mobile layouts on one grid.

Every rebuilt screen is checked in skill 04 before the next starts.

SKILL

03



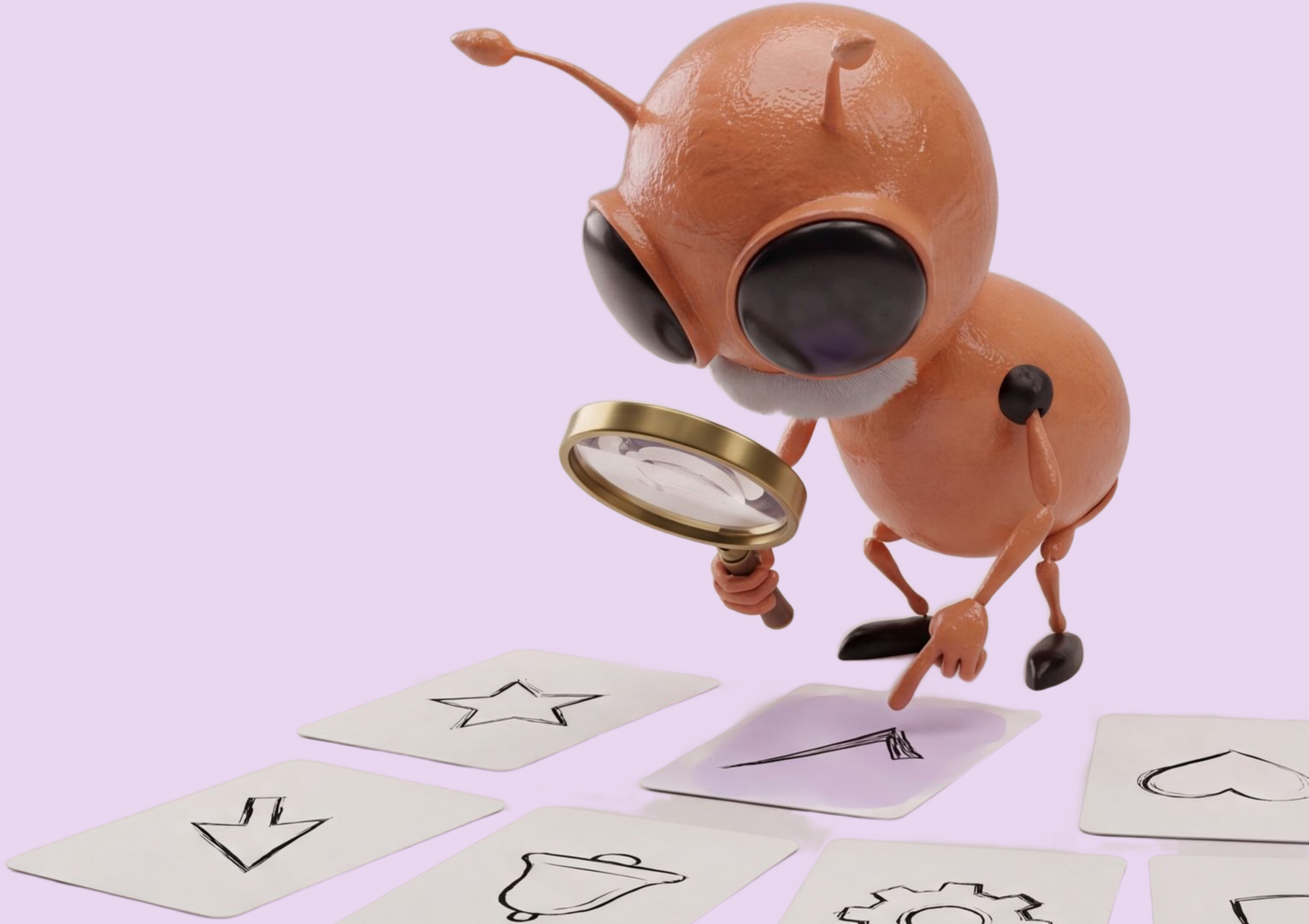


Cluma

Audit and *fix.*

Every screen checked against our rules.

04





3 audits run on every screen: content and structure, accessibility, and mobile or platform fit. Every finding gets fixed before the screen counts as done.

Cluma

Every screen

checked

A screen is only done when it is built and clean. The skill reads the saved file again, not just what it drew, so a broken screen can’t slip through.

Smallest system: every component used at least twice, no one-off variants.

Handoff: no raw values, auto layout everywhere, one type scale.

Accessibility: contrast and focus checked on every screen.

Cluma

Consolidate it *all.*

The smallest system, ready for your developers.

05



SKILL

05

Keep only what you use *in three steps.*



Once every screen is built, the consolidation pass looks at the whole file. Near-duplicates get merged, and a component used only once goes back into its screen.

What stays is the smallest component set that still covers every screen, in a file your developers can build from. (Grizzle throws out the rest. It’s the closest he gets to smiling.)

Tokens

01

Bind every value to one token scale.

Components

02

Merge near-duplicates, keep what is used twice.

Spec

03

Write the token file and developer questions.



SKILL

05

When the last screen is done, the skill hands your developers one small file with every design choice in it.

Each color, text size and spacing gets one name, and it is the same name in Figma and in the code. Nobody has to guess, and nobody has to ask. Gloober set it up once and hasn’t moved since.

WHAT IS FAST, REALLY?

Not more screens.
More validated bets
per month.



LEARN FASTEST

Not who ships most.

35,000+ startups



The startups that win

The startups that win are not the ones that ship the most. They are the ones that learn the fastest. In a peer-reviewed study of over 35,000 startups, those that adopted systematic A/B testing performed 30% to 100% better after a year (Koning et al., 2022).

SYSTEMATIC A/B TESTING

30%

better after a year: the low end

100%

better after a year: the top of the range

How the *loop* runs in three steps.

Before AI, a flow took about a week to design and three weeks to build, so design stayed ahead without trying. Now the build takes days.

So design has to run on a new cadence. With the five skills it can, and the next bet is ready before the last one is built.

THE LOOP IN THREE STEPS:

Plan one small bet and how to measure it *01*

Design it on the system with the skills *02*

Ship it, measure it, then plan the next *03*



One finished screen today is worth more than ten half-finished ones next week.

Small batches beat big ones; Zap would start all ten. In DORA's research on software teams, the fastest performers deploy most often and also have the lowest failure rates (DORA, 2024). We work the same way in design.

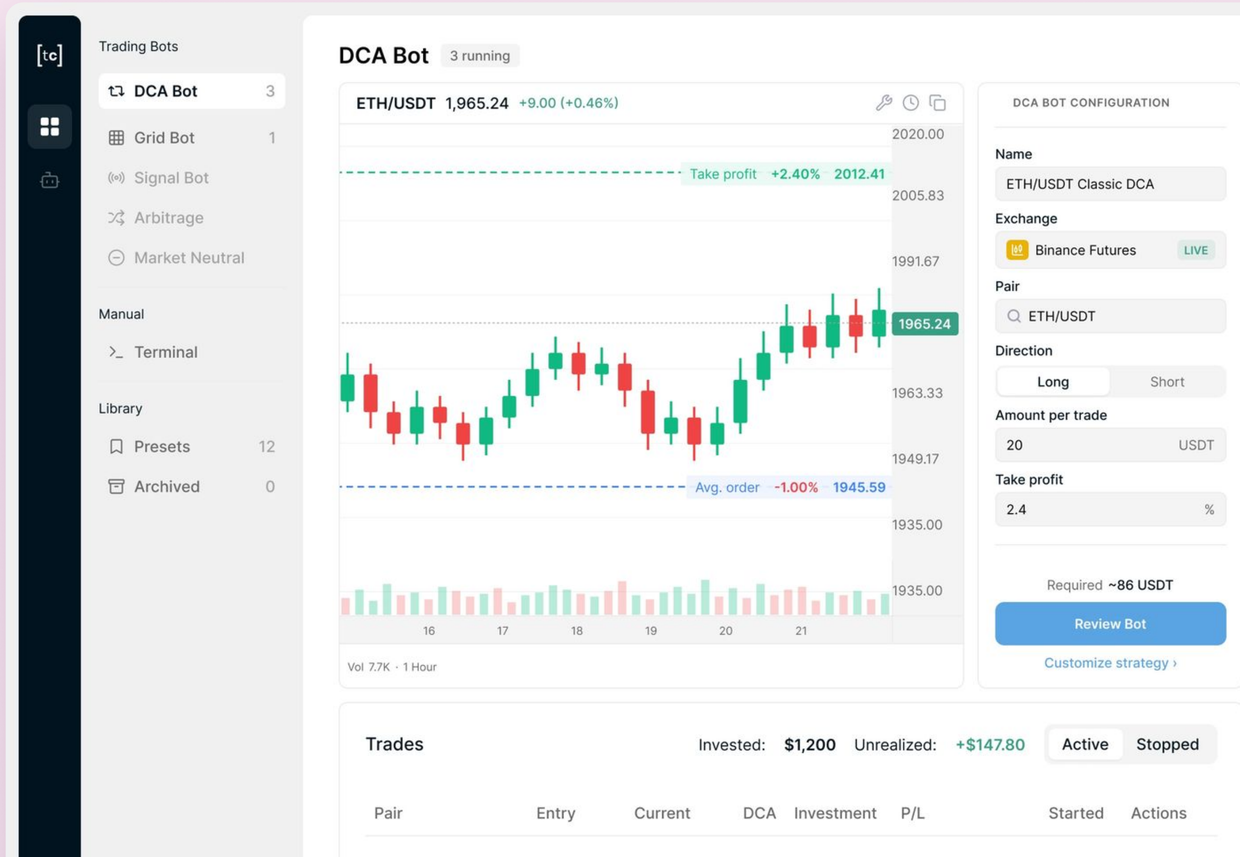
Before: purple gradients, glowing pills, glass *cards, rocket emojis, four* button styles, eleven grays and *no empty states.*



After: the same screens, on one *system.*

The product still works the same. Every screen now uses the same tokens and components. The next 50 will match.

The design now has one button with three variants, a nine-step gray scale, one modal, an 8 px spacing scale, and designed empty, loading and error states.



Cluma

Skip the training. Let us *do it.*



Need *help?*

Skip the months of training.

Each skill in this playbook only got reliable after months of running it on real products. We have already done that work, so we can take this part off your team.



WHAT CHANGES FOR YOU

We take over

AUDIT SYSTEM FIGMA FILE

We bring your existing screens onto one system and hand over a build-ready Figma file and tokens.

Your team builds

DEVELOPMENT

Your developers build from the approved file, the way they always have. Every new screen matches the last one.

They didn't do it *alone.*

Founders and executives behind Inkt, Microsoft-backed Kaasai and software used by the world's largest research firms.

Each one handed us the slow part, so their teams could build.



MARTIN S.

Founder, Inkt · SVP Sales, Sony Music

"Cluma was fantastic to work with and very communicative throughout the process. They listened and made recommendations based on our priorities."

inkt



MARYANN C.

CEO, Kaasai · backed by Microsoft

"Pleasure working with Cluma on a tight deadline. They delivered and we got the result we wanted!"

kaasai



BERNHARD W.

Founder, SurveyTester · used by Ipsos, Kantar, YouGov

"I learned that a good website design is more than fonts, colors and images. It is about the visitor and how to guide them through the mass of information we have."

Survey Tester



MAXIMILIAN S.

Co-founder, Pantares · ex-Deloitte

"We were really excited when we saw the results since we really liked the color palette as well as the unique design elements. The UI/UX prototype really helped getting a first touchpoint how our product will look like in the end."

Cluma

Who is behind them

The people we design for have played at the highest level. Our clients' founders cut their teeth at Sony Music, Goldman Sachs, Deloitte, CNBC and Harvard Innovation Labs, so they know what a product looks like when the stakes are high.

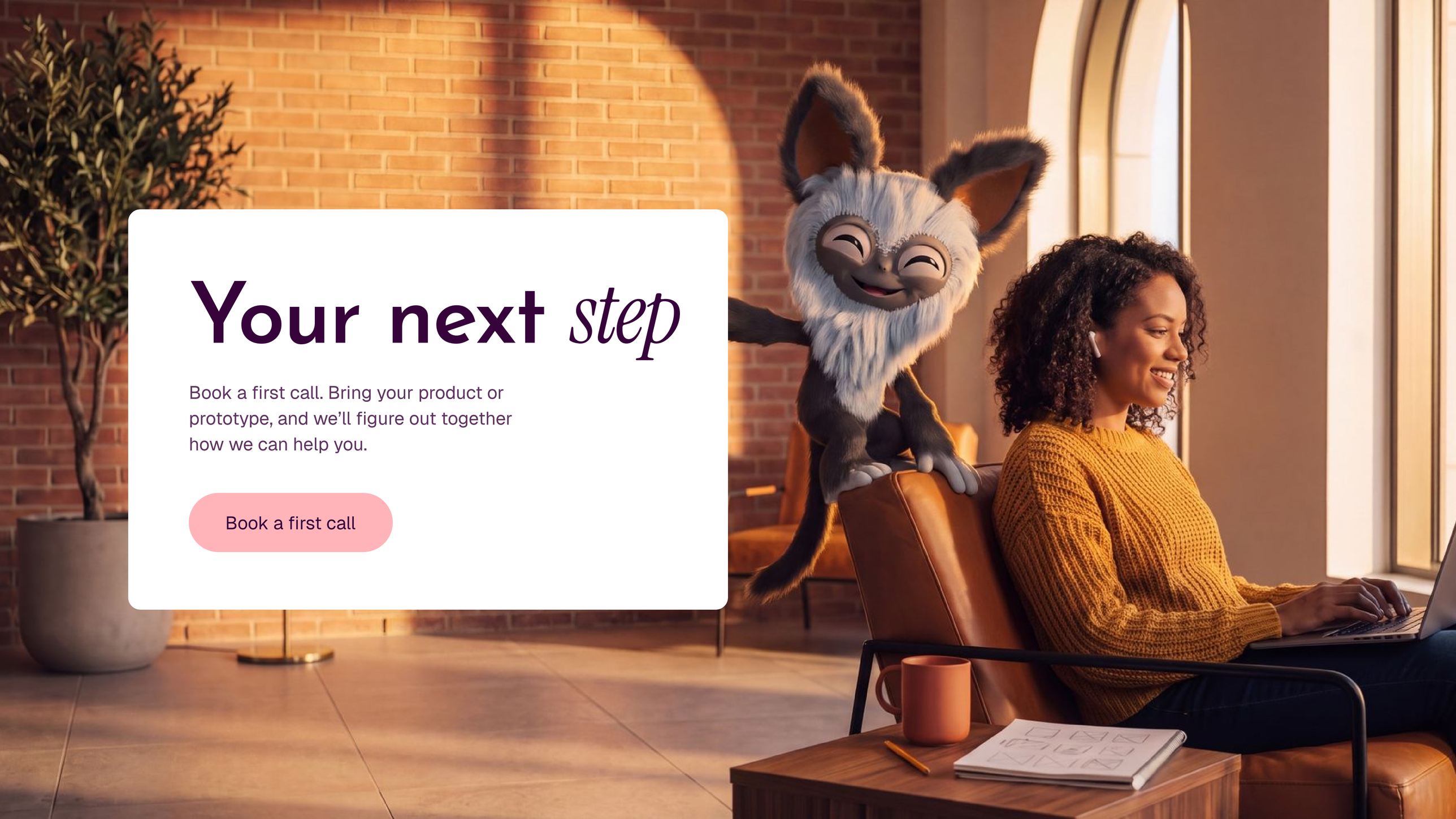
That track record brings serious backers. Microsoft and the UK government, including the Ministry of Defence, stand behind our clients. Others build with partners like BYD.

Where their products land

And the work doesn't stay on a test server. Their products reach Chipotle, Barnes & Noble, Siemens Energy and the U.S. government, and one even made it into the room with Netflix.

Products like these don't get there on a rough prototype. Someone has to turn the brief into a build-ready file that holds up in front of buyers and backers, and that is the part we do.





Your next *step*

Book a first call. Bring your product or prototype, and we'll figure out together how we can help you.

Book a first call

Our *sources*

Cluma. (2026). Analysis of 376 design job postings from startups with fewer than 200 employees, 2023 to October 2026 [Unpublished raw data].

DORA. (2024). Accelerate state of DevOps report 2024. Google Cloud. https://dora.dev/research/2024/dora-report/

Figma. (2025). What's working, what's not, for designers and developers? State of the Designer 2025. https://www.figma.com/reports/designer-developer-trends/

Koning, R., Hasan, S., & Chatterji, A. (2022). Experimentation and start-up performance: Evidence from A/B testing. Management Science, 68(9), 6434–6453. https://doi.org/10.1287/mnsc.2021.4209

Stripe. (2018). The developer coefficient. https://stripe.com/files/reports/the-developer-coefficient.pdf



Same product. One system. Now *build yours.*
cluma.design

Cluma