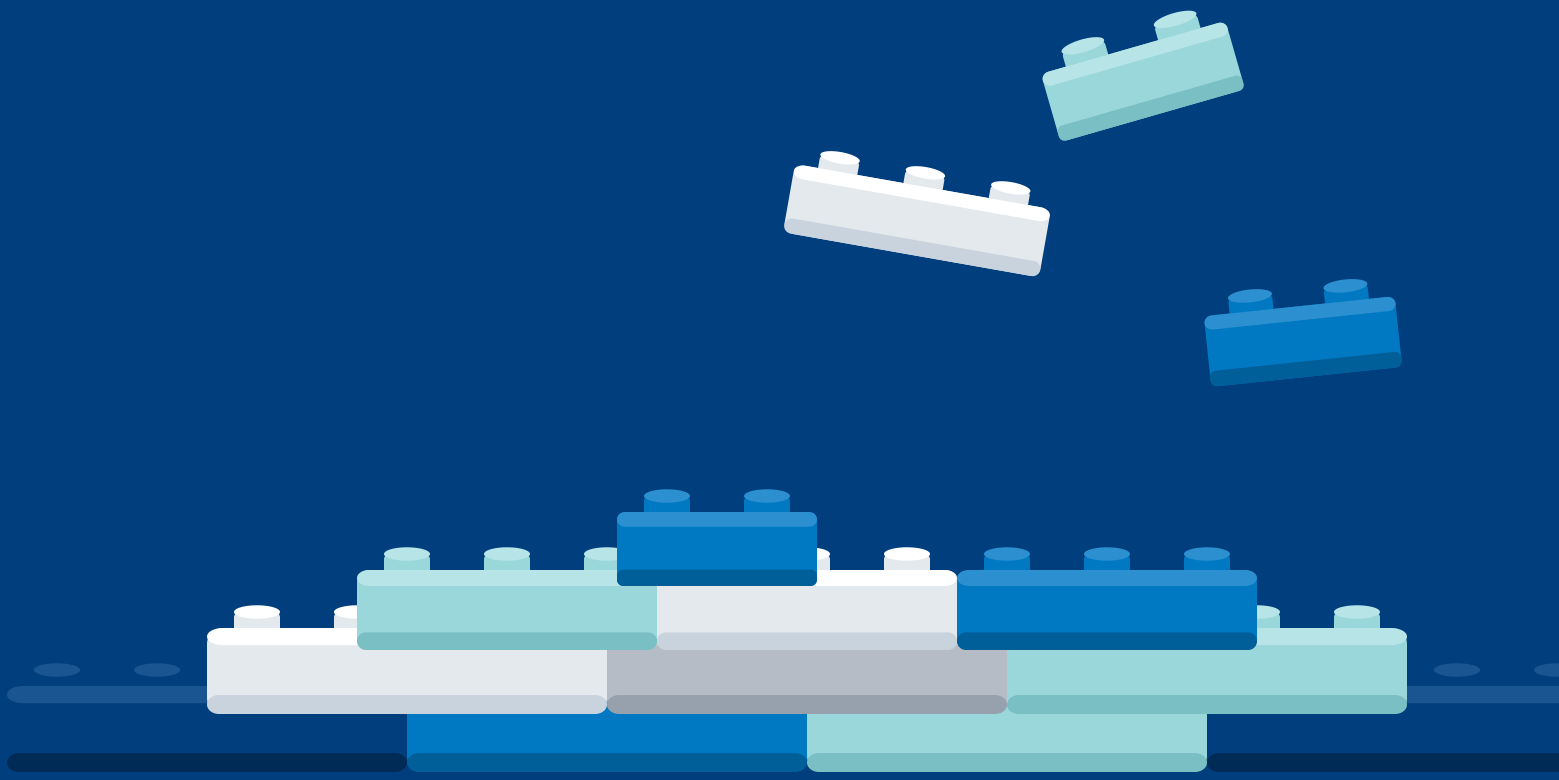
PRACTICAL GUIDE · LINÉALE

Brick by brick

Designing with AI, from Figma to code, without losing the method.

The designer isn't replaced, they're augmented.

Tristan Fraud



Before you start

A complete iPhone app, rebuilt from Figma all the way to code and then a live prototype, with an AI agent at my side. Here's how, and what it cost me.

This guide won't teach you a tool. Tools change every six months. What lasts is the way you think, frame and decide.

I'll walk you through one main project, an outdoor iPhone app, from code to Figma, then from Figma to a live prototype, told through a short film. A few detours through other projects will shed light on it. You'll see what worked, what cost me time, and above all what I understood along the way.

This guide is for designers, beginners and experienced alike, who are starting to design with AI and want to stay in control of the method.

The split of roles is simple. The agent (an AI that chains actions: reading, editing, checking) executes and verifies. You frame, judge and decide.

Go at your own pace. Take what helps you, and build your own method, brick by brick.

The designer isn't replaced, they're augmented.



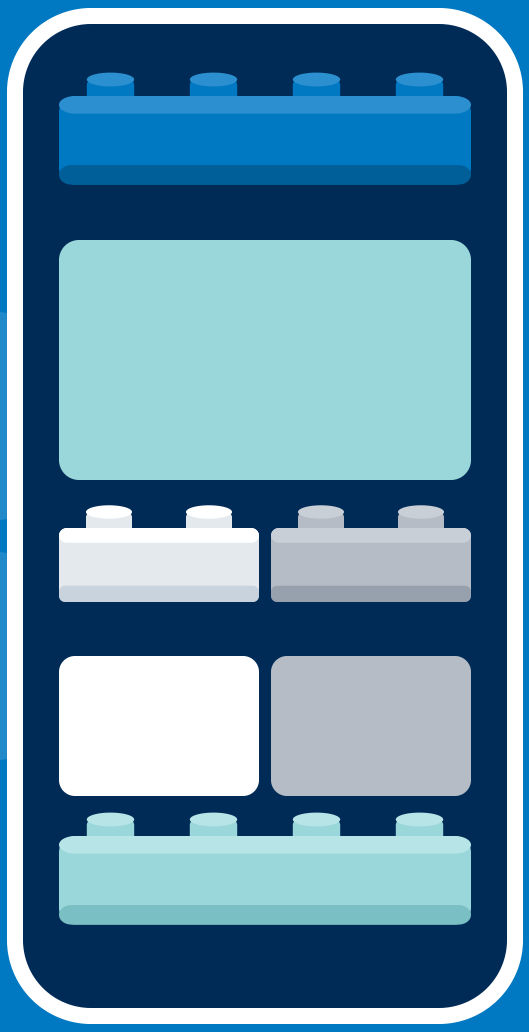
The result, first

92
icons

34
screens

2
languages

2
themes, light and dark



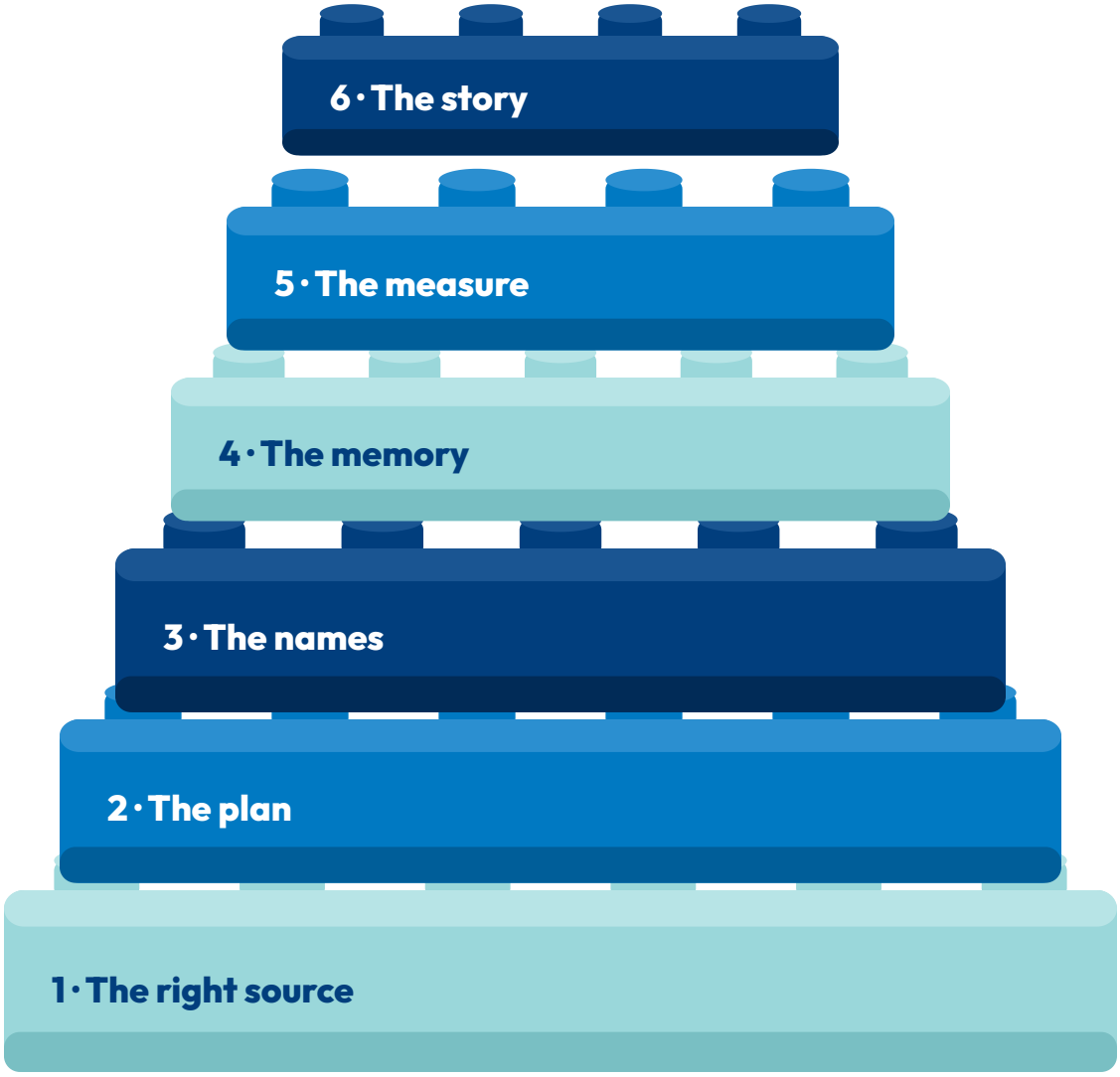
In the end, every screen was assembled from the same bricks: the colours, the icons, the components.

Switching to French, to the dark theme or to another interface generation cost no extra screen: you just turned on a mode.

This result didn't come on the first try. It's the fruit of a method built through mistakes. I'll tell you about them in order.

CONTENTS

The method in 6 bricks



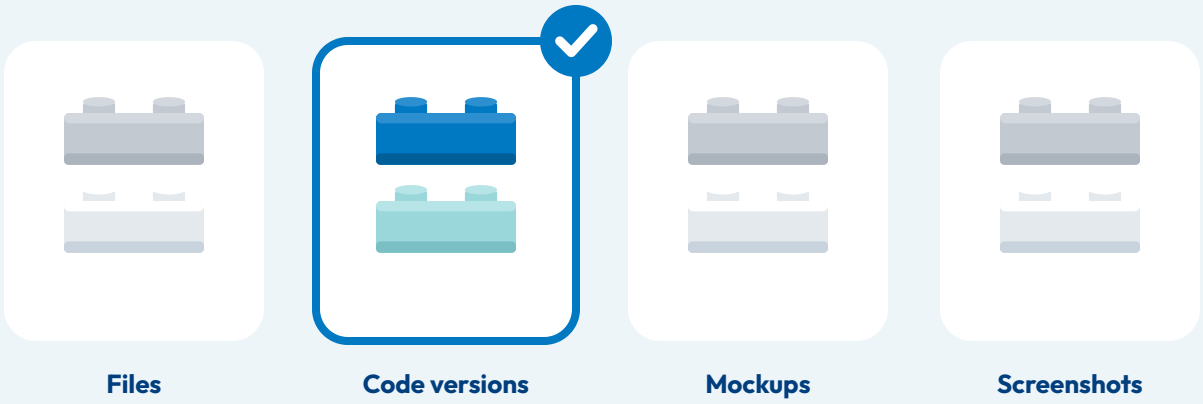
ALSO INSIDE

The result, first
What changed for me
The 6 reflexes
Your first brick
Words of the guide

“Are you sure you have the right app?”

Before you build, find the version that is authoritative.





Which one is authoritative?

THE STORY

“Are you sure you have the right app?”

The project: a native outdoor iPhone app, with its own design system, in light and dark, in two languages. This time we started from the app’s own code, not from a mockup. I was sure I had the right version.

A good part of the project had already been rebuilt when a doubt crept in: what if this wasn’t the right app? Maybe there was more in the code.

There was more. The real app lived in another version of the code, with its own screens, screenshots and design system. Everything built so far rested on an old base. We had to start again from the right one.

THE CLICK

Before framing anything, I now take an inventory of every source: files, code versions, mockups, screenshots. Then I ask a single question: which one is authoritative? What developers call the source of truth.

1

single source of truth

IN ONE SENTENCE

Before you build, find the version that is authoritative.

WHY IT MATTERS WHEN YOU'RE AUGMENTED

An agent builds on whatever you hand it, without ever asking whether it's the right base. Checking the starting point is your first move as a designer.

MY ADVICE

Ask "is this really the latest version?" on day one. It sounds naive; it keeps you from building in the wrong place.

✦ A PROMPT TO THINK WITH

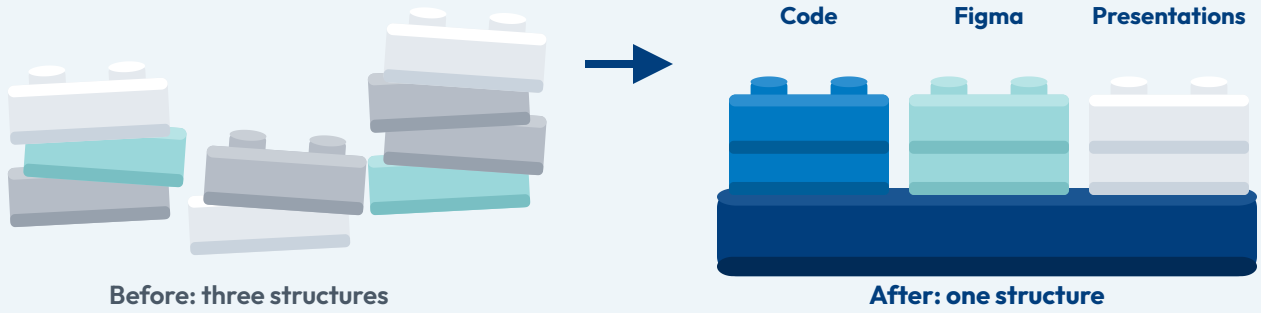
"Before we start, list every source you can consult for this project. For each one, tell me what it contains, its date, and what makes you think it is or isn't the reference. Don't build anything until I've decided."

2

The plan before the screens

One structure, set before the first screen.





Before: three structures

After: one structure

THE STORY

The plan before the screens

On an earlier project, I had drawn three flows in Figma, the coded demo had four visitor profiles, and the personas followed yet another split. Three structures for a single project. As long as nobody looked closely, it held. The day I wanted to show one flow from start to finish, nothing fit anymore.

On the outdoor app, I started differently: a single structure, set before drawing a single screen.

THE CLICK

Flows first, then one persona per flow. The code, Figma and the presentations are then all built on that one structure.

1

page for all your flows

IN ONE SENTENCE

One structure, set before the first screen.

> **WHY IT MATTERS WHEN YOU'RE AUGMENTED**
>
> When the agent produces fast, the smallest inconsistency at the start multiplies fast too. The plan is what keeps everything consistent as the pace picks up.

MY ADVICE

Before you design, write your flows on a single page. If someone can't read them back in a minute, they're not ready.

✦ **A PROMPT TO THINK WITH**

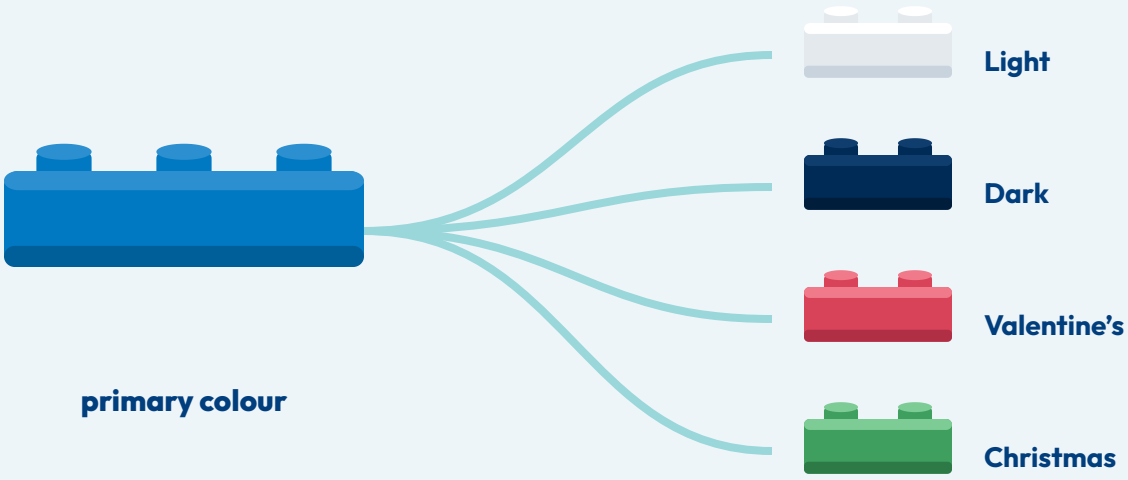
"Here are my flows on one page. Look for the gaps: a persona without a flow, a flow without a starting point, a screen that belongs to no flow. Don't suggest solutions; ask me questions."

3

The pipeline, behind the scenes

Name it once, use it everywhere.





THE STORY

The pipeline, behind the scenes

For this app, everything started from the code. Claude, an AI assistant, read the app's colours and components, then recreated them in Figma through an MCP server, a bridge that lets the agent read and edit a Figma file. First the colours in light and dark, then 92 icons, then 34 screens rebuilt with those same bricks.

The secret was a detail: Figma and the code used exactly the same names. When one said "primary colour", the other knew what it meant, with no translation.

I learned that lesson on an earlier project: a red from the wrong brand guidelines and a frozen peach tint, impossible to fix in one go. I had to hunt them down screen by screen. Since then, I reverse the order: colours and sizes first, named once; screens after. Even Valentine's Day and Christmas became nothing more than two modes you switch on in one click.

THE CLICK

That's what tokens are for: give each value a name, then let everything else refer to it. If you catch yourself typing a colour code by hand, a brick is missing: create it first, draw the screen after.



0

colour codes typed by hand

IN ONE SENTENCE

Name it once, use it everywhere.

> **WHY IT MATTERS WHEN YOU'RE AUGMENTED**
>
> An agent is excellent at faithfully reproducing what has been named. It's bad at guessing what you haven't defined. The clearer your system, the more accurately your assistant works. You no longer draw every screen: you design the rules that create them.

MY ADVICE

Before opening a new screen, check that every colour and size you need already has a name. What has no name doesn't exist for the agent.

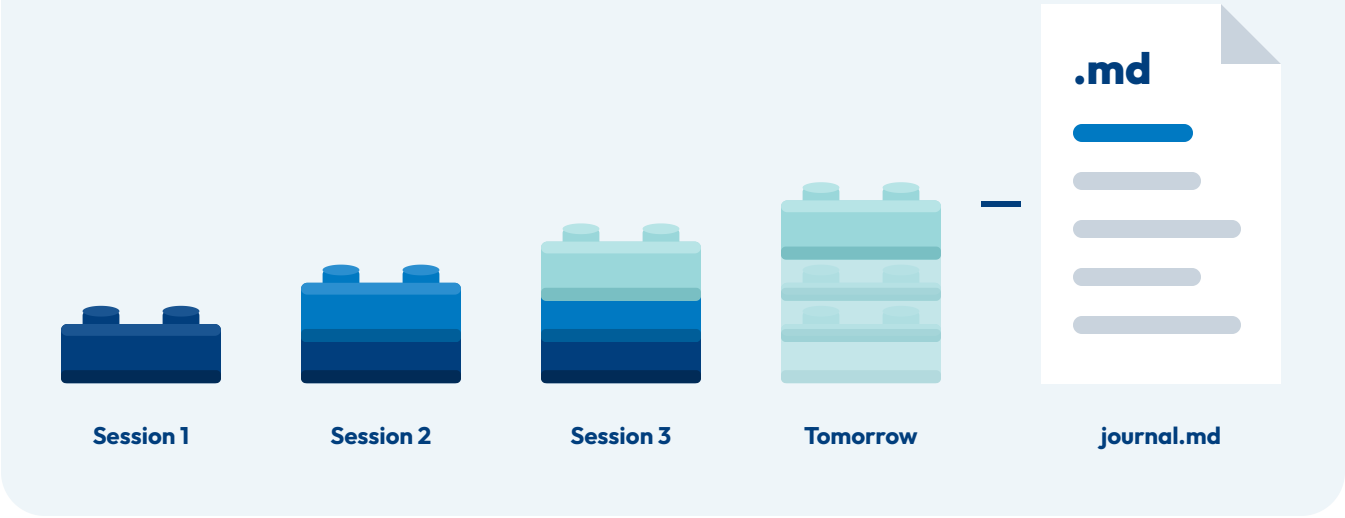
✦ A PROMPT TO THINK WITH

"Here is my screen. Find every value typed by hand: colours, spacing, text sizes. For each one, tell me whether an existing token could replace it, or whether a new one is needed. Don't change anything; give me a list."

A memory for your agent

What is written down isn't lost.





THE STORY

A memory for your agent

As the project went on, one problem kept coming back: every new session started from scratch. The agent forgot the rules, the decisions, the mistakes already fixed. I kept repeating myself, and repeating myself badly.

The solution fits in simple text files: MD (Markdown) files. Versioned with the code and reread by the agent at every session, they became the project's memory.

In fact, I write everything down. Every decision, every mistake, every rule fits in a dated note. Thanks to them, I could pick up each project after every interruption without losing anything, and this guide grew out of them.

THE CLICK

An MD file can serve several purposes. Instructions to set up the agent: how you work, what it must respect. A checklist, for accessibility for example. A project memory: the decisions made, and why. And living documentation that evolves with the product.

3

lines of journal a day

IN ONE SENTENCE

What is written down isn’t lost.

WHY IT MATTERS WHEN YOU’RE AUGMENTED

The agent has no memory between two sessions, but neither do you, not really, after three weeks. These files are what let you delegate without losing the thread. It’s method in writing, which any colleague or any agent can pick up.

MY ADVICE

Keep a project journal, even three lines a day. It’s the most underrated brick.

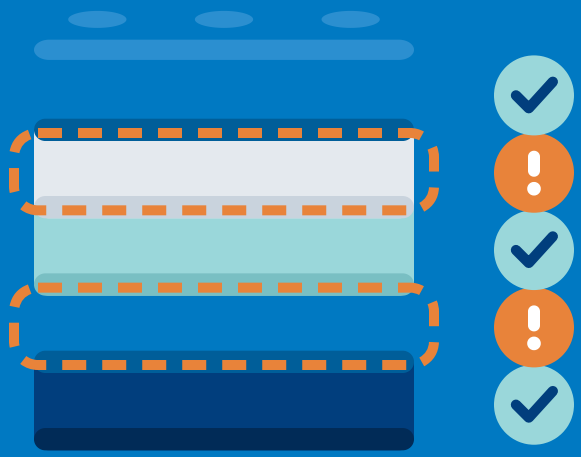
✦ A PROMPT TO THINK WITH

“Based on this session, summarise in a few lines the decisions we made, what we tried without success, and the rules I want you to follow next time. Write it as a dated note you can reread tomorrow without context.”

5

Your agent checks, you judge

What can be measured can be delegated.







44 px

touch target

Aa

12 px

minimum text



320–430

mobile width

Aa Aa

Aa

light and dark



0

accessibility errors

THE STORY

Your agent checks, you judge

On another project, a bilingual mini-site with a dark mode, the prototype was going to be reviewed by a seasoned developer who tolerates no visual error. My eyes were no longer enough.

I handed the repetitive checks to the agent: text legibility, button size, screens that overflow, alignment. A script found a navigation bar that stuck out of the grid by 16 pixels on a large screen, invisible to the eye. The accessibility audit came back with zero errors, in light and dark alike.

That same project had also taught me a lesson the hard way. By raising small text from 12 to 13 pixels to make it more legible, I had widened a table: on small phones, the home page overflowed sideways, in a version already published. I hadn't seen it, because I was only looking at the screen I was fixing.

THE CLICK

After every fix, however tiny, I rerun the checks on every screen size. It takes a minute, and it spares you the bad surprise. The same principle applies to mobile: I had built the desktop first and derived the phone afterwards, and it took me six rounds to reach a clean version. A screen that fits on a small phone scales up easily; the reverse is a headache.

44 px

for a touch target

IN ONE SENTENCE

What can be measured can be delegated.

WHY IT MATTERS WHEN YOU'RE AUGMENTED

The agent checks what can be measured. The designer judges what can't: meaning, tone, what matters first. If you delegate judgement, you're no longer augmented, you're replaced. If you refuse to delegate measurement, you waste your time where the agent is at its best.

MY ADVICE

Keep a short list of checks to redo after every change, and never skip it, especially when you're in a hurry.

✦ A PROMPT TO THINK WITH

"Here is my screen and my criteria: legibility, touch target size, width from 320 to 430 pixels, accessibility. Check each one, in light and dark, and tell me what you measured and what you couldn't measure. Don't judge tone or meaning; that's my job."

Telling the story to get a decision

Show it, make them want it, get a decision.





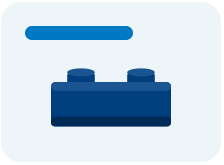
The prototype

to live the flow



The film

to spark interest



The slides

to get a decision

THE STORY

Telling the story to get a decision

At the end of every project, one question remained: how do I show it? A prototype, however polished, doesn't tell its own story.

I got into the habit of preparing three things: the prototype, to live the flow; a short film with music, under a minute, to spark interest; and a few slides, to get a decision. Always the same thread: the problem in the client's own words, the flow that solves it, the decision expected.

IN ONE SENTENCE

Show it, make them want it, get a decision.

3

formats to tell it

IN ONE SENTENCE

Show it, make them want it, get a decision.

WHY IT MATTERS WHEN YOU'RE AUGMENTED

The agent produces fast, but it doesn't know what your client needs to hear to decide. Storytelling is a designer's move: choosing what to show, in what order, and what to leave out.

MY ADVICE

End every presentation by saying exactly what you expect: approve, test, launch.

✦ A PROMPT TO THINK WITH

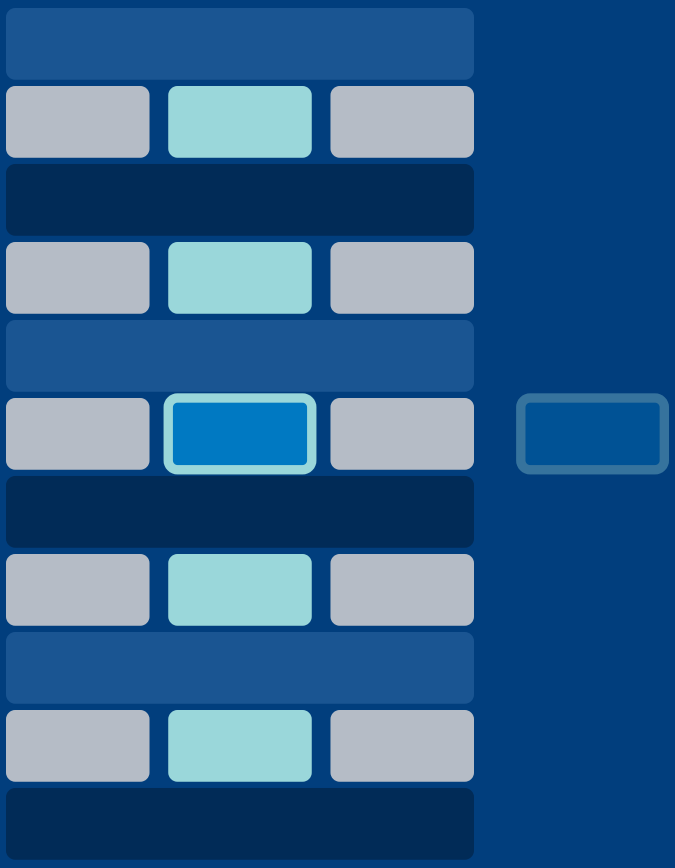
"Here is my client's problem, in their own words, and the flow I designed. Help me find the sentence that connects the two, then ask me three questions about what my client needs to decide after seeing the prototype."

EPILOGUE

What changed for me

I spend much less time pushing pixels, and much more framing, critiquing and keeping the whole coherent. I've become less of a draughtsman and more of an architect.

And I doubt better. AI moves fast and speaks with confidence; my job is to check what it claims, then decide. Screens churned out on an assembly line don't make a product. What makes one is judgement.



Less draughtsman, more architect.

The 6 reflexes

BRICK 1 · THE RIGHT SOURCE

Before you build, find the version that is authoritative.

1 single source of truth



BRICK 2 · THE PLAN

One structure, set before the first screen.

1 page for all your flows



BRICK 3 · THE NAMES

Name it once, use it everywhere.

0 colour codes typed by hand



BRICK 4 · THE MEMORY

What is written down isn't lost.

3 lines of journal a day



BRICK 5 · THE MEASURE

What can be measured can be delegated.

44 px for a touch target



BRICK 6 · THE STORY

Show it, make them want it, get a decision.

3 formats to tell it

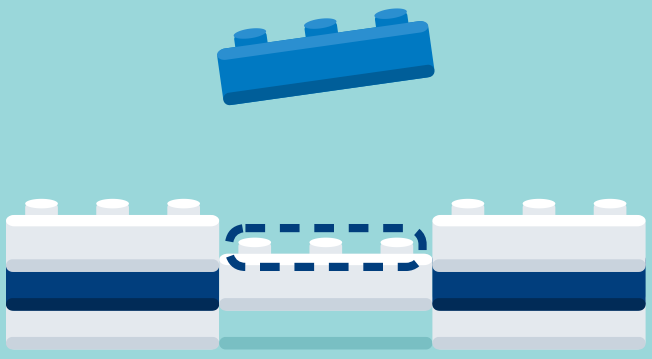


Your first brick

You don’t need a big project to start. Take a small real case, a screen you know well, and try three things this week.

1. **Write its flows on one page, before touching the design.**
2. **Name three colours and two sizes, and replace every value typed by hand.**
3. **Review it on a small phone, and ask yourself what your agent could check for you.**

You’ll have laid your first brick. The next one will come faster. What matters isn’t going fast; it’s being able to start again without taking everything apart.



One brick at a time.

Words of the guide



DESIGN SYSTEM

Design system: the box that holds tokens, components and their rules. A shared language between design and code.

Token: a value given a name, for example “primary colour” or “medium spacing”. Changing it once updates everything.

Component: a reusable element, like a button or a card, built with tokens.

Mode: another version of the same tokens (light, dark, campaign, language), without duplicating screens.

Light and dark mode: two colour modes for the same interface.

Figma variable: the way Figma stores a token, with one value per mode.

Flow: a person’s path, screen after screen, to reach their goal.

Persona: the portrait of a typical user, which helps make design decisions.

Prototype: a clickable version of the product, to test before building.

Grid: the spacing framework, often 8 or 16 pixels, on which elements align.

Touch target: the area you can touch with a finger. It should measure at least 44 pixels.

Accessibility: designing so that everyone can use the product, including people with disabilities or in difficult conditions.

Words of the guide



VISUAL IDENTITY

Visual identity: everything that makes a brand recognisable: logo, colours, typefaces, images, tone.

Logotype: the brand name drawn in letters, often paired with a symbol to form the logo.

Brand guidelines: the document that sets the rules for using the visual identity: where to place the logo, which colours to use, which to avoid.

Style guide: a product's reference document: colours, typefaces, components and usage examples. Brand guidelines speak about the brand; the style guide speaks about the product.



TOOLING

Figma: the interface design tool where mockups and the design system live.

MCP server: the bridge that lets Claude read and edit a Figma file.

Git repository: where the code lives, with the history of every change.

VS Code: a code editor, where you read and edit a project's files.

Claude Design: Claude's tool for designing and exploring interfaces.

Claude Code: Claude's agent that works directly in a project's code.

MD (Markdown) file: a simple text file with light formatting, used as instructions, memory or documentation for the agent.

Source of truth: the version that is authoritative when several sources contradict each other.



PLATFORMS

Native app: an app built specifically for one system, like an iPhone app written in Swift. It accesses the phone's capabilities directly.

Web app: an app that runs in the browser, with no installation.

Hybrid app: web content wrapped in an installable app.



ARTIFICIAL INTELLIGENCE

Agent: an AI that doesn't just answer, but chains actions: reading a file, editing code, checking a result.

Prompt: the instruction you give the AI. The more precise it is, the more precise the answer.

Context: all the information the agent has in front of it during a session.

Automated audit: a check performed by a script or an agent, for example on accessibility or legibility.

Augmented designer: a designer who delegates execution and checking to the agent, but keeps framing, judgement and decision-making.

lineale

Your judgement. The speed of AI.

Let’s talk about your product

lineale.ca