AI IN PRODUCT DESIGN

Claude + a bridge to Figma. AI works right inside the files.

How I built a process where an AI agent works directly in layout and design system files, following the team's rules, while the designer keeps the final say.

Oleksii Repetii

Less time on routine work ✦ AI is a tool, not a replacement ✦ A tidy design system

THE PROBLEM

Routine work used to eat up designers' hours. Now AI does it in minutes.

01

Hundreds of components

Renaming, rebinding styles, Auto Layout and descriptions — all by hand, one at a time.

02

Checking by eye

Off-scale spacing, stray styles and broken instances get caught by chance.

03

Knowledge lives in heads

DS rules aren't written down anywhere, so new designers and developers can't see them.

THE IDEA

AI handles the routine. People make the decisions.

DESIGNER

Sets tasks, makes decisions and owns quality. Art director and design system owner.

AI AGENT

Doesn’t just advise — it works inside Figma files: does the job, checks it against the rules and reports back.

HOW IT'S BUILT

How a task travels: from words to an edit in Figma

1

Designer

Describes the task in plain words, in the terminal or the project chat

→

2

AI agent

Claude reads the rules and writes a small script for the task

→

3

Local bridge

Sends the script to Figma on the designer's machine

→

4

Figma

The script runs in the DS and layout files

Before the task ↓

The agent reads the shared rules and the product profile

After the task ↑

A report for the designer: what was done, which rules were broken, and what to publish

UNDER THE HOOD

A direct tunnel from AI to Figma

AGENT

Claude Code

Writes JS against the Figma Plugin API for each task

HTTP →

BRIDGE

Local server

Runs on the designer's machine, with a queue per file

WebSocket →

FIGMA

Executor plugin

Runs in Figma Desktop and executes the code in the file

Where tasks come from

From the Mac terminal directly, or from a cloud project via a remote-control session on the same Mac.

What comes back

The result, a log and errors with hints. The agent reviews them and writes a report.

Ready-made scripts

Common actions aren't written from scratch every time: checking a layout against the rules, listing DS components, loading states.

WHY THIS WAY

Free from standard Figma MCP limits

	STANDARD MCP	DIRECT BRIDGE
Limits	200 calls a day; 6 a month on View and Collab seats	✓ None — everything runs locally
Writing to files	Mostly reads designs; writing is beta-only	✓ Creates, edits and binds styles
Scale	Individual calls, one at a time	✓ Hundreds of layers in a single script
Multiple files	A single context	✓ In parallel, with a queue per file

Figma limits per the documentation as of October 2026. Bridge limitations: Figma Desktop must be open; libraries are published manually.

MODEL INDEPENDENCE

Models change. The system stays.

VARIABLE PART

AI model

- Claude — today
- Another model — tomorrow
- A new version — when it ships

Any model that can write code and use tools will work



CONSTANT PART

The process itself

- Rules and product profiles
- Gates and audits
- Bridge to Figma
- Ready-made scripts
- Memory and decision log
- Designer’s decisions

Stays the same: process, rules, checks and files

Depends on the model: decision quality and speed

THE KEY IDEA

The intelligence lives in the rules, not in the tool

Shared rules	For every product: how we work, what's allowed and what isn't, a product registry
Product profile	Grids, spacing and radius scales, fonts, screen assembly rules, checks, exceptions
Project memory	Decisions carried between tasks: "already decided, don't suggest it again"

QUALITY GATES

0 Plan from the DS
Components and styles are chosen before the first command

1 Assembly rules
DS instances, styles and scales only

2 Audit
A script checks the build and reports

3 Exceptions
Anything outside the DS becomes a candidate for review

HOW IT WORKS

Two task flows

DS EDITS

1 Task
In plain words, with context and a goal

2 Rules
AI reads the rules and the profile

3 Pilot
Backup, then a trial run on one set

4 “Yes”
The designer reviews and decides

5 At scale
Apply to the whole set, then audit

6 Report
A summary plus a “to publish” list

NEW SCREENS

1 Plan from the DS
What we take from the DS and what’s missing

2 Assembly
Following DS rules and scales

3 Audit
A script checks the build

4 Options
Each one side by side in its own frame

5 Decision
The designer picks an option and what to add to the DS

Each decision goes straight into the product profile, so next time AI already knows it.

TWO ENTRY POINTS

Small tasks in the terminal, big ones in the project

Terminal on my computer

- A quick fix to a single component
- Questions about the current Figma selection
- Quick checks
- Cheaper and faster

Cloud project

- Cleaning up DS sections to a plan
- Audits with reports, screen assembly
- A separate branch and fresh context per task
- Tasks can be sent from a phone

Both read the same rules — so the results are consistent.

THE WORKDAY

Switch on — work — switch off

09:00

Start

One command launches the bridge and opens the plugin in Figma

→

all day

Working with AI

Small edits in the terminal. Bigger tasks in the cloud project, each in its own branch. Tasks can be sent from a phone.

→

end of day

Wrap-up

Publish the library from the list, commit new rules to git

→

18:00

Stop

One command shuts the bridge down

Rules, profiles and scripts live in a private git repo: full history and easy rollback.

MULTIPLE PRODUCTS

A new product means a new profile, not a new process

1

Onboarding

Product, files, flows

2

DS review

AI reads the file without changing anything

3

Profile

Built from a template: scales, grids, rules

4

AI-ready DS

Descriptions, consistent names and states so AI can read the system

5

Generation

Index, templates, checks

One set of shared rules plus a profile per product. Products with different rules run side by side without mixing.

SCREEN GENERATION

The design system is a foundation, not a cage

DS candidate — a new element the design system doesn't have yet. AI builds it with DS styles, explains why it's needed and waits for the designer's call.

1
Finding the best solution
Best for the screen, not just what already exists

→

2
A missing piece
AI builds it with DS colors, fonts and spacing

→

3
A label
Marked "DS candidate", with a note on why

→

4
Decision
The designer decides whether to keep it

→

5
Into the system
Once approved, it becomes a DS component

Example: a screen needs a timer, but the DS has none. AI builds one in the system's style, and you decide whether it becomes a component.

Colors, fonts, spacing and grids come only from the DS. AI gets freedom, and the system grows under control.

EVOLUTION

Every problem becomes a rule

1

Problem

Something went wrong or cost too much

→

2

Analysis

We look for the cause, not someone to blame

→

3

Rule

Written so that AI can follow it

→

4

Record

Added to the shared rules or the product profile

📖 Every case goes into a “problem → fix → rule” log, so the process gets more reliable with every task.

PRINCIPLES

Rules that keep files safe

Backup before every edit

Always something to roll back to

Test small first

One set first, then everything

Nothing changes without a “yes”

The designer always has the final say

No deleting before migrating

Layouts never break

Decisions get recorded right away

AI doesn’t forget what was agreed

One task, one session

AI doesn’t go beyond the brief

SUMMARY

The designer decides. AI does the work. The rules are shared.

The process is captured in rules and profiles, so it can be repeated on any product.

Telegram @gnt_alex

repetii.space

Oleksii (Alex) Repetii · Product Design Lead