COLIBRI GROUP · CASE STUDY

Colibri
Achieve More

AI-Assisted Design System Modernization

How design judgment, controlled AI execution, and verification turned a fragmented library into a usable system for design and engineering.

ROLE	SCOPE	DURATION	HEADLINE OUTCOME
Senior UX Designer	Token-based Figma design system	3-day modernization sprint	660 token bindings clarified

This case study is structured to Nielsen Norman Group standards for UX case studies.

SLIDE 2 · THE RESULT

A fragmented library became a system a team can trust

660

Existing token bindings clarified across color, typography, spacing, radius, and elevation, with component integrity preserved.

11

Organized system pages in the modernized library

120

Live specimen instances, still linked to their main components

45

Button variants preserved in the master component set

0

Variants deleted or recreated; no instance detached

The modernization in numbers: one three-day sprint reorganized the library and clarified its foundations without breaking the components teams already depended on.

SLIDE 3 · THE PROBLEM

A robust library existed, but it was hard to use and hard to present

The problem was operational: make the system maintainable without breaking the implementation surface.

- **Source inventory:** components, variants, and product patterns were piled onto working pages. The source needed protection, not cosmetic rearrangement.
- **Inconsistent presentation:** documentation pages lacked a consistent visual model and readable hierarchy for reviewers or future contributors.
- **Binding gaps:** raw values and historic styling existed alongside foundations, and a blanket “fix everything” pass would have created risk.

The task was not to replace the system. It was to preserve its intent, organize it, and make it governable, without breaking the components teams already depended on.

THE GOVERNING CONSTRAINT

Every change had to preserve the components, variants, and live relationships teams already depended on: nothing broken, nothing detached, nothing recreated.

SLIDE 4 · THE REFRAME

The brief said organize. The audit said separate.

IT STARTED AS

A modernization of a fragmented Figma library: organize what existed, clarify the token bindings, and make the system presentable.

THE EVIDENCE THAT BROKE IT

The starting-point audit showed the failures were structural, not cosmetic: source material and presentation were piled onto the same working pages, and raw values sat alongside foundations, so rearranging pages or forcing a cleanup pass would have put the implementation surface at risk.

THE REFRAMED PROBLEM

Separate the working library from the presentation layer. Source pages remain accurate for implementation; specimen pages show how the system is intended to be used. The distinction made the library safer to evolve and easier to demonstrate.

SLIDE 5 · MY ROLE AND MANDATE

I owned the strategy and the governance. AI executed inside approved scope.

I owned

Strategy, governance, and judgment

- Senior UX Designer at Colibri Group, 03/2021 – 11/2022, leading the modernization end to end.
- Design-system strategy and component governance.
- Standards, mapping approvals, exception decisions, and the publish decision.
- Every structural decision and every quality check.

I led with partners

The governed delivery model

- The seven-step model run with AI assistance: Assess, Propose, Review, Approve, Execute, Verify, Publish.
- Buttons as the controlled test, proving the specimen pattern before it scaled.
- A low-risk Cover-page pilot before the token pass touched the broader system.

Partners owned

AI execution and reporting

- Inventory of the components, variants, and product patterns in the source library.
- Proposals, and execution of approved scope only.
- Verification reporting, back for human review.

SLIDE 6 · RESEARCH AND INSIGHTS

Evidence replaced opinion in the modernization plan

FINDINGS	DECISIONS
Forcing spacing into the component set was the wrong fix	Instead of forcing spacing into a component-set boundary, the system used a separate specimen page with live instances and true auto layout: source masters for implementation, specimens for explanation.
A blanket token pass would have created risk	An audit separated clear mappings from intentional exceptions. Values without a sensible nearby token were retained and documented rather than force-mapped, so the foundation strengthened without visually distorting components.
A pilot surfaces platform quirks safely	A low-risk Cover-page pilot exposed a corner-radius API quirk before it could affect the broader system. Figma page canvases, which cannot be live-bound to variables, were matched visually to neutral/700 and documented as a platform exception.

Method note: the work started with evidence: an audit of the source library before anything was standardized. Every live mutation after that was scoped, reviewed, and verified, and each completed pass closed with structural and visual checks.

Buttons proved the model: specimens, not a rearranged source

✓ CHOSEN

A separate specimen page: live Button instances organized by size, type, and state inside padded, auto-layout documentation sections, while the master component set stayed untouched.

✕ REJECTED

Forcing spacing into a component-set boundary, and cosmetic rearrangement of the source pages: the source needed protection, not rearrangement.

? WHY

Implementation and explanation are different jobs. Source masters serve implementation; specimens show how the system is intended to be used.

WHAT EVIDENCE DECIDED IT

All 45 Button variants remained in the master component set: none deleted or recreated. The same documentation-card model then scaled to navigation, forms, feedback, and data.

MASTER COMPONENT SET

45 variants

Preserved in place: the implementation surface teams already depend on.

↓

SPECIMEN PAGE

Live instances

Organized by size, type, and state, each one still linked to its main component.

The structural fix in one move: source masters for implementation, specimens for explanation.

A controlled token pass, not blind cleanup

✓ CHOSEN

A controlled pass over the existing bindings: an audit separated clear mappings from intentional exceptions, and 660 bindings were clarified across color, typography, spacing, and radius.

✕ REJECTED

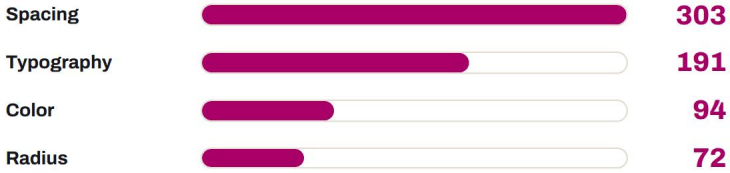
A blanket “fix everything” pass: force-mapping every raw value and historic style to the nearest token, on faith.

? WHY

Token binding required verification, not automation on faith. Force-mapping would have risked visually distorting components just to make the file look tidy.

WHAT EVIDENCE DECIDED IT

Verification after the pass: the Components page remained structurally intact, with the same 40 top-level nodes and 83 components, and Button focus and semantic strokes were preserved.



The 660 clarified bindings, by foundation. Exceptions stayed visible: retained, documented, and never force-mapped.

SLIDE 9 · WHAT SHIPPED

One file that works as a system and presents as a story

Source system

- Foundations: tokens for color, typography, spacing, radius, and elevation
- Components: preserved masters, properties, variants, and implementation relationships
- Patterns: the product patterns the library already carried

Presentation system

- Specimens: live-instance pages for Buttons, Navigation, Forms, Feedback, and Data
- Read Me: usage, governance, maintenance, and engineering-handoff context
- Interview narrative: the story the file can now tell

Two layers, one file: source pages stay accurate for implementation, and specimen pages show how the system is intended to be used.

1. **Assess** Audit the source library before standardizing it
2. **Propose** Mappings and scope proposed for review
3. **Review** Every live mutation reviewed before it runs
4. **Approve** Mappings and exceptions approved by the human
5. **Execute** The approved scope, and only the approved scope
6. **Verify** Structural and visual checks on every pass
7. **Publish** The human owns the publish decision

The governed delivery model the system now runs on: AI accelerated execution; human judgment controlled the decisions.

SLIDE 10 · OUTCOMES

Measured against the library that was inherited

MEASURE	BEFORE	AFTER THE SPRINT
Token bindings clarified	Raw values and historic styling beside the foundations	**660**
Organized system pages	Inventory piled onto working pages	**11**
Live specimen instances	No specimen layer	**120**
Button variants preserved	At risk in any cleanup pass	**45 of 45**

Timeframe and framing: a three-day sprint under the governed model. AI accelerated inventory and execution; every structural decision and quality check stayed human. Verification confirmed the Components page structurally intact and all 120 specimen instances still linked to their main components. The outcome claimed is the system itself: organized, documented, and safe to evolve.

WHAT THIS DEMONSTRATES

“AI can help modernize a design system. It still needs a grown-up in the room: governance, accessibility, and verification are what turn AI speed into a system a team can trust.”

STEVE MOORE · FROM THE CASE STUDY SUMMARY

SLIDE 11 · WHAT I LEARNED

Design-system leadership is an operating model, not a Figma skill

Lesson 01

Start with evidence

Audit the product or source library before standardizing it. The audit, not taste, decides what changes.

Lesson 02

Protect implementation surfaces

Preserve source components; use specimens to explain them. The point was never merely a cleaner file; it was continued trust in the library.

Lesson 03

Use AI responsibly

Pilot, scope, verify, and document exceptions. AI accelerates the work while humans retain decision authority.

The operating lesson I carry into every design-system project since: a modern process is controlled, evidence-based, and human-led, and the strongest proof is the judgment behind the system.

SLIDE 12 · THANK YOU

Thank you for reviewing this case study.

The summary page holds the outcomes, the deliverables, and the full document.