CHORUS

LEAD SENIOR UX/UI DESIGNER
Nancy O'Donnell

PRODUCT DESIGN CASE STUDY

Chorus UX Builder

A no-code experience builder that lets business teams design, configure, and publish enterprise forms and web apps - with an embedded AI assistant. One form travels across three people and two platforms: the **Designer** who builds it, the **End User** who fills it, and the **Processor** who verifies it inside the task platform.

DISCIPLINE
Product & UX Design

PLATFORM
Angular 17 · Web

DESIGN SYSTEM
Bluestone Light

Contents

01 - OVERVIEW

One artifact, three people, two platforms

Chorus UX Builder is where an enterprise experience is *designed*. But a form's life doesn't end at publish - it's *filled* by an end user, then *verified* by a processor inside SS&C's task platform. Because that platform shares the same design system and AI assistant, the artifact stays coherent from first drag to final sign-off.

0
lines of code to design & ship a form

3
personas served by one shared surface

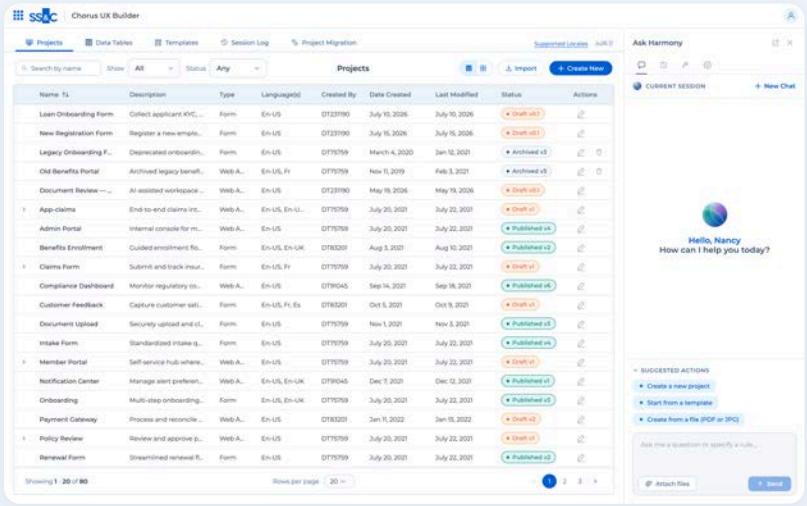
1
governed AI assistant across both platforms

The problem

Enterprise form-building meant developer tickets and inconsistent UI - and once a form was submitted, it landed in a disconnected back-office tool that looked and behaved nothing like where it was built. Context, branding, and AI governance were lost at every handoff.

The response

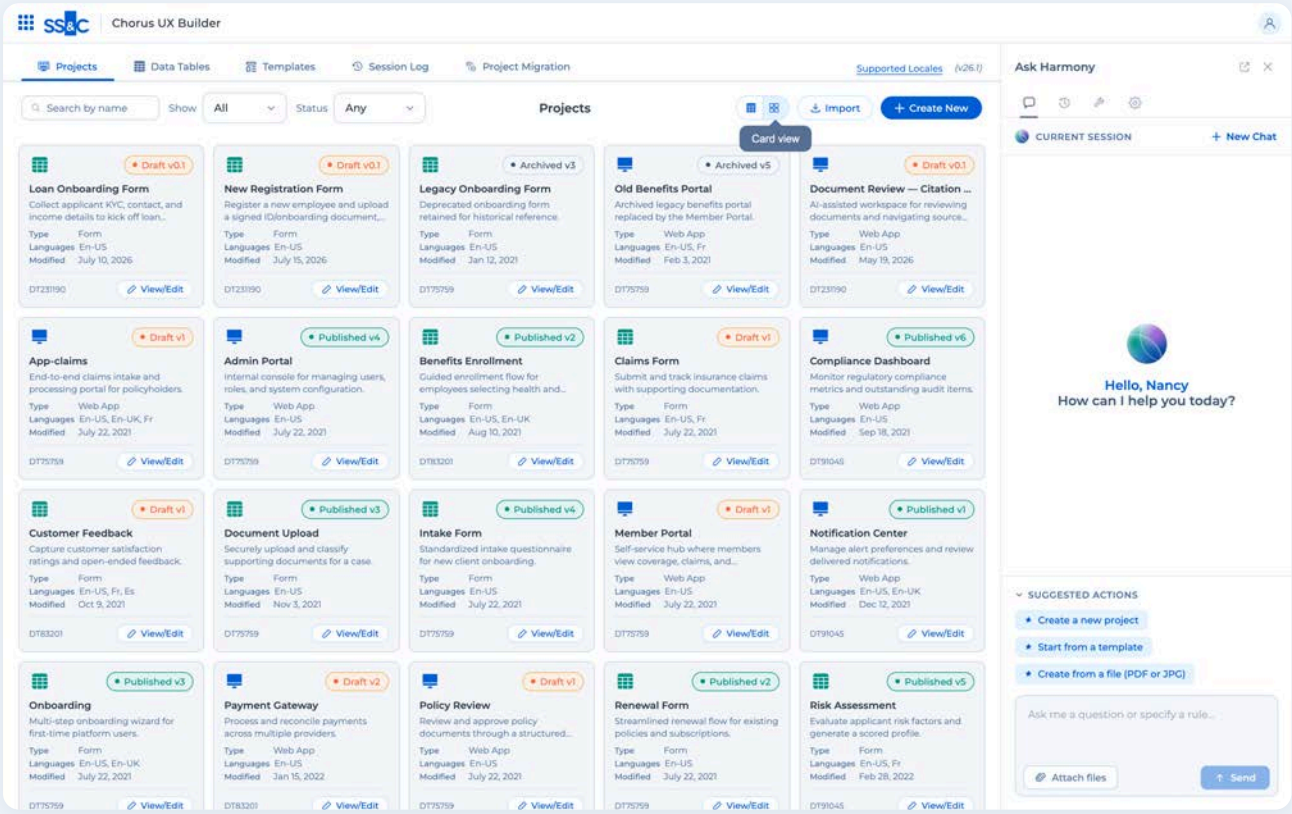
A visual builder plus a shared runtime: the same form renders identically as a **work item** inside the Processor task platform, carrying its theme and its governed AI panel with it. Nothing is re-implemented downstream.



The Projects dashboard (table view) - a designer's home base for every form and web app across the organization.

The same projects, at a glance

Designers manage a growing library of forms and apps, so the dashboard offers both a dense table and a scannable card view. The card view leads with status and type - ideal for quickly spotting what's in draft, what's published, and what needs attention.



The Projects dashboard in card view - each project as a tile with status badge, type, languages, owner, and a direct View/Edit action.

Status at a glance

Color-coded badges - Draft, Published, Archived - with version numbers make the state of every project instantly readable.

Form vs Web App

A type icon distinguishes forms from web apps, so designers filter the library by what they're looking for.

One-click entry

View/Edit opens the project straight into the right editor - form designer or builder canvas - no intermediate screens.

Who touches the form

A single form is handled by three people with different jobs and different tools. The product's task is to serve each - power for the builder, simplicity for the filler, control and evidence for the verifier - without fracturing the experience between them.

D The Designer

Business analyst · builds the experience in UX Builder

BUILDER

GOALS

- Ship a branded form without engineering
- Configure & govern the AI assistant

FRUSTRATIONS

- Developer tickets for small changes
- Inconsistent styling across teams

E The End User

Applicant · completes the published form

CONSUMER

GOALS

- Complete the task quickly & confidently
- Avoid re-typing what's already in a document

FRUSTRATIONS

- Cluttered, confusing forms
- Tedious manual data entry

P The Processor

Back-office reviewer · verifies the submission in Processor (BPM)

VERIFIER

GOALS

- Verify a submission against source docs fast
- Clear the work queue within SLA

FRUSTRATIONS

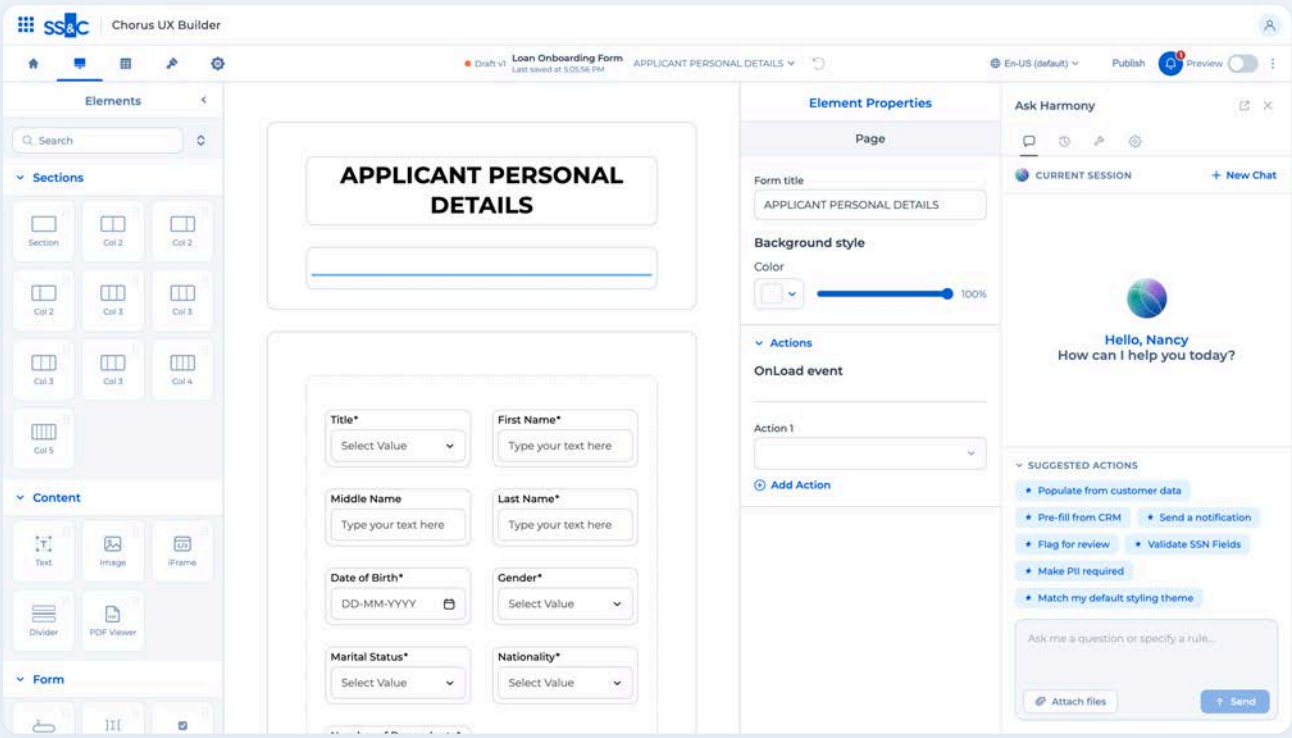
- Re-keying data into a separate tool
- No evidence trail; AI they can't trust

THE THROUGH-LINE

The same form must feel native in three places: the builder canvas, the end-user's browser, and the Processor work list. Shared design system + shared governed AI = one artifact, no re-implementation, no lost context.

Building the experience

The Designer works on a live canvas. Layout elements structure the page; content and form fields drop into sections; drag handles reorder everything in place. What they build is what everyone downstream gets - no translation layer, no handoff.



The drag-and-drop canvas - layout sections on the left palette, live element properties on the right.

Structure first

Layout elements (Section, Columns, Tabs) form the skeleton. Content and form elements can only be dropped inside them - enforcing a clean, consistent hierarchy by design.

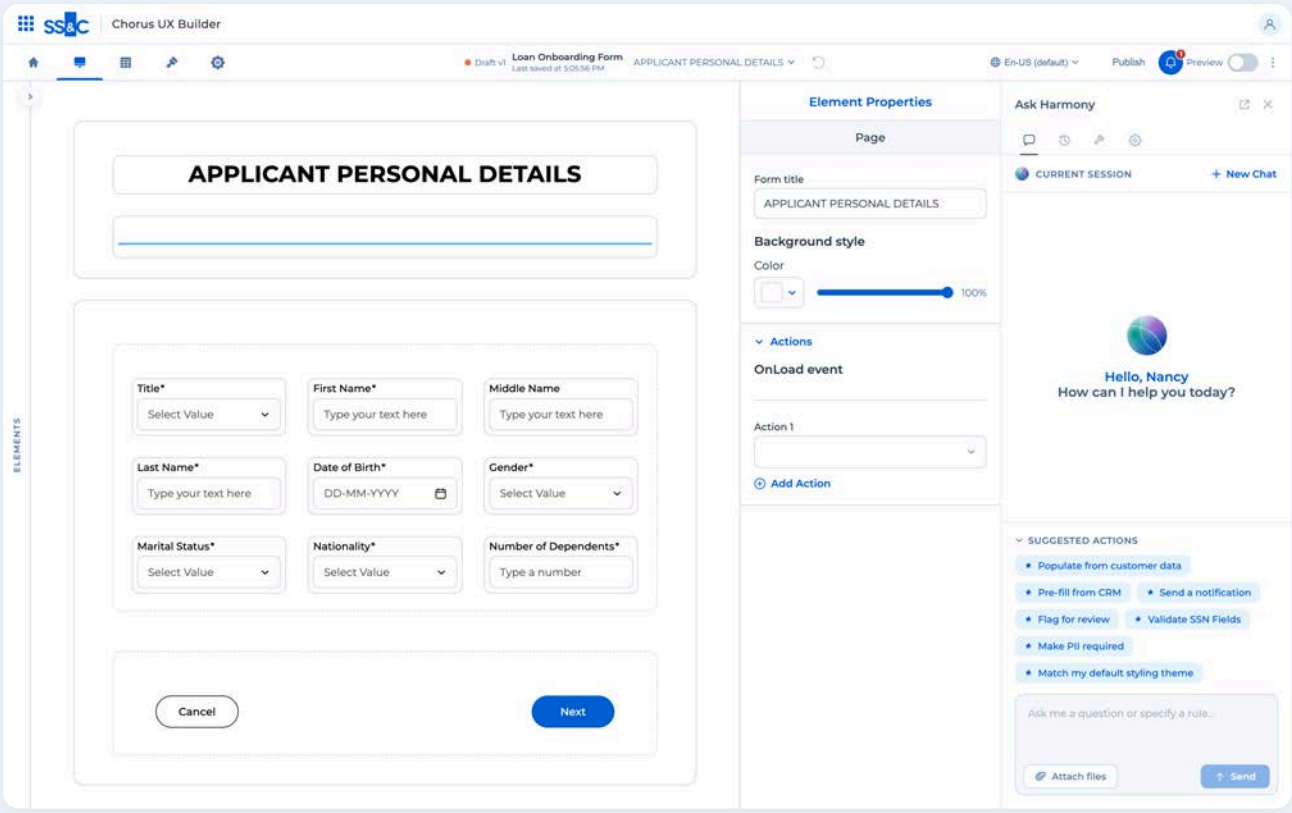
Direct manipulation

Every section and element carries an inline drag handle to reorder in place, plus contextual tools to duplicate, delete, or add columns - no modal round-trips.

03 - PERSONA A · THE DESIGNER

From blank canvas to branded form

The form designer renders exactly what the End User will experience, styled with the organization's Bluestone Light theme. The Designer edits the real thing - headers, sections, fields, and the embedded "Ask Harmony" assistant - all in context.



The Loan Onboarding Form under construction - real fields, real theme, real AI assistant, live properties panel.

HEADER SECTION

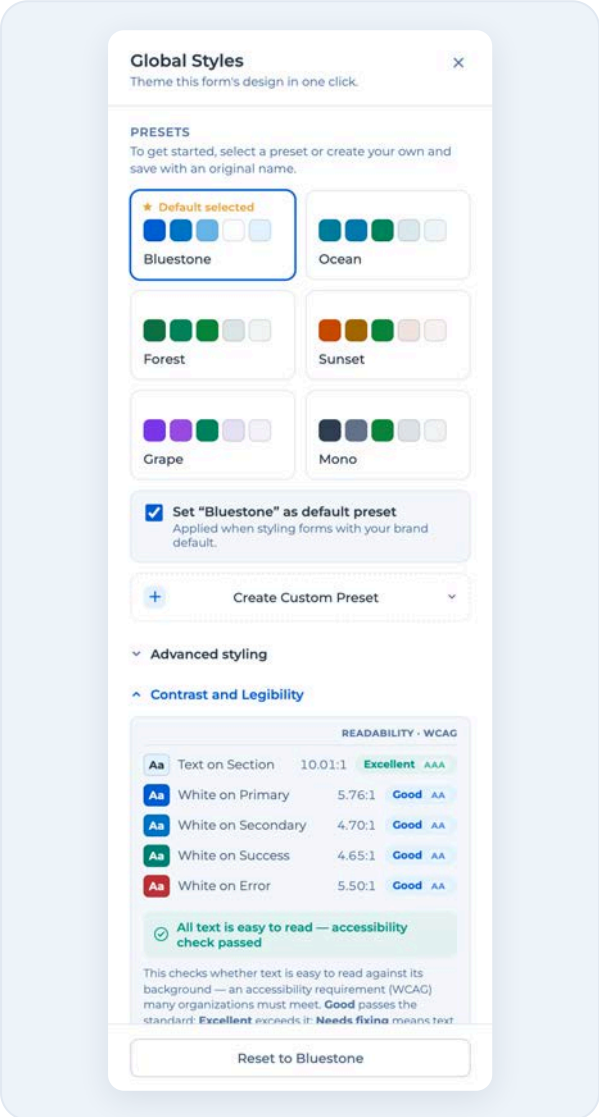
NESTED SECTIONS

FIELD-LEVEL PROPERTIES

REORDER HANDLES

Theming in one click

Branding shouldn't require a stylesheet. Global Styles themes an entire form from a single panel - pick a preset, set it as the organization's brand default, and every new form starts on-brand. A built-in contrast checker keeps it accessible before it ever ships.



1 Preset palettes

Six curated themes - Bluestone, Ocean, Forest, Sunset, Grape, Mono. One click restyles the whole form's colors, type, and radii.

2 Brand default

Star a preset as the organization's default so every new form starts on-brand - no manual setup, no drift across teams.

3 Custom presets

"Create Custom Preset" saves a team's own palette, design tokens, and type scale under an original name for reuse.

4 WCAG built in

A live Contrast & Legibility check scores text pairs against WCAG AA / AAA - with a one-tap Fix-all when anything falls short.

Primary #0760D5 **Success #009688**

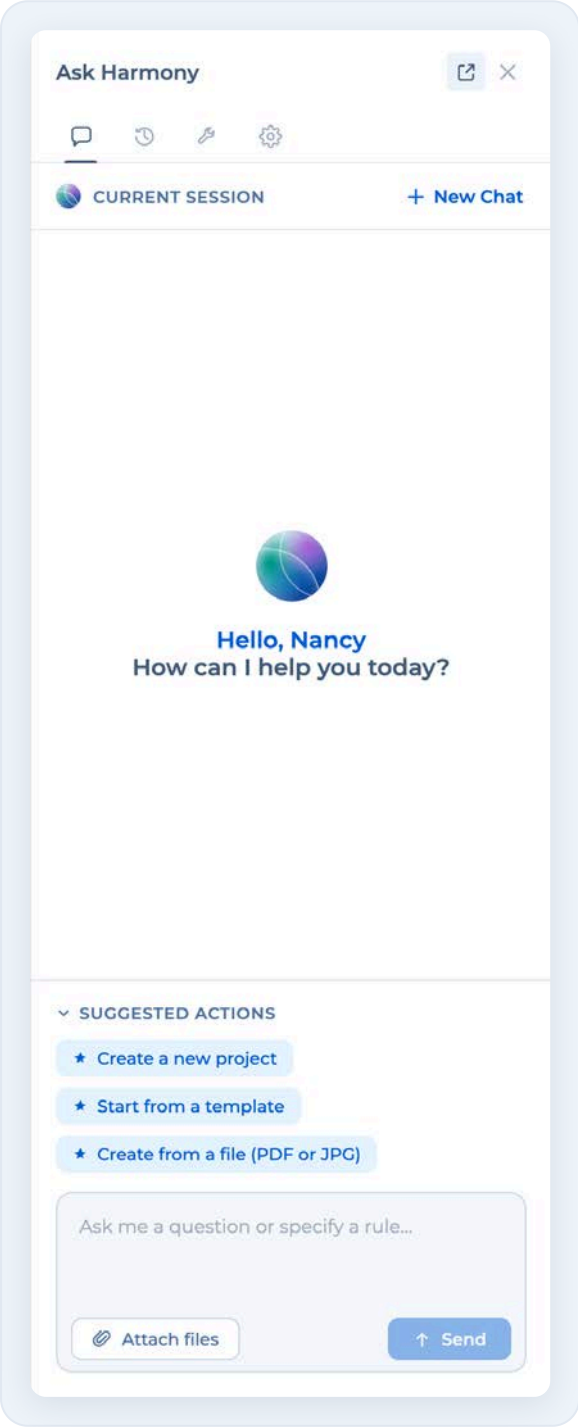
Preset spectrum

WHY IT TRAVELS

The chosen theme is part of the form's definition - not a builder-only skin. It renders identically for the End User and inside the Processor work item, so brand and accessibility carry across the platform boundary automatically.

Designing the AI panel

The assistant is a first-class surface, not a bolt-on. Its panel is styled in the same Bluestone system as the rest of Chorus - so when it later appears for the End User and the Processor, it already feels native. Calm empty state, on-brand gradient orb, and suggested actions that teach the user what to ask.



1 **Gradient identity orb**
The blue-violet-teal orb is the assistant's signature mark - the same gradient family as the report cover, tying brand to product.

2 **Warm, personal empty state**
"Hello, Nancy - how can I help you today?" greets by name in Montserrat, keeping the blank canvas inviting rather than empty.

3 **Suggested actions**
Starred chips ("Create a new project," "Start from a template") lower the blank-page barrier and hint at what the assistant can do.

4 **Session & tools tabs**
Chat, history, tools, and settings live in a quiet icon row - power kept one tap away without cluttering the conversation.

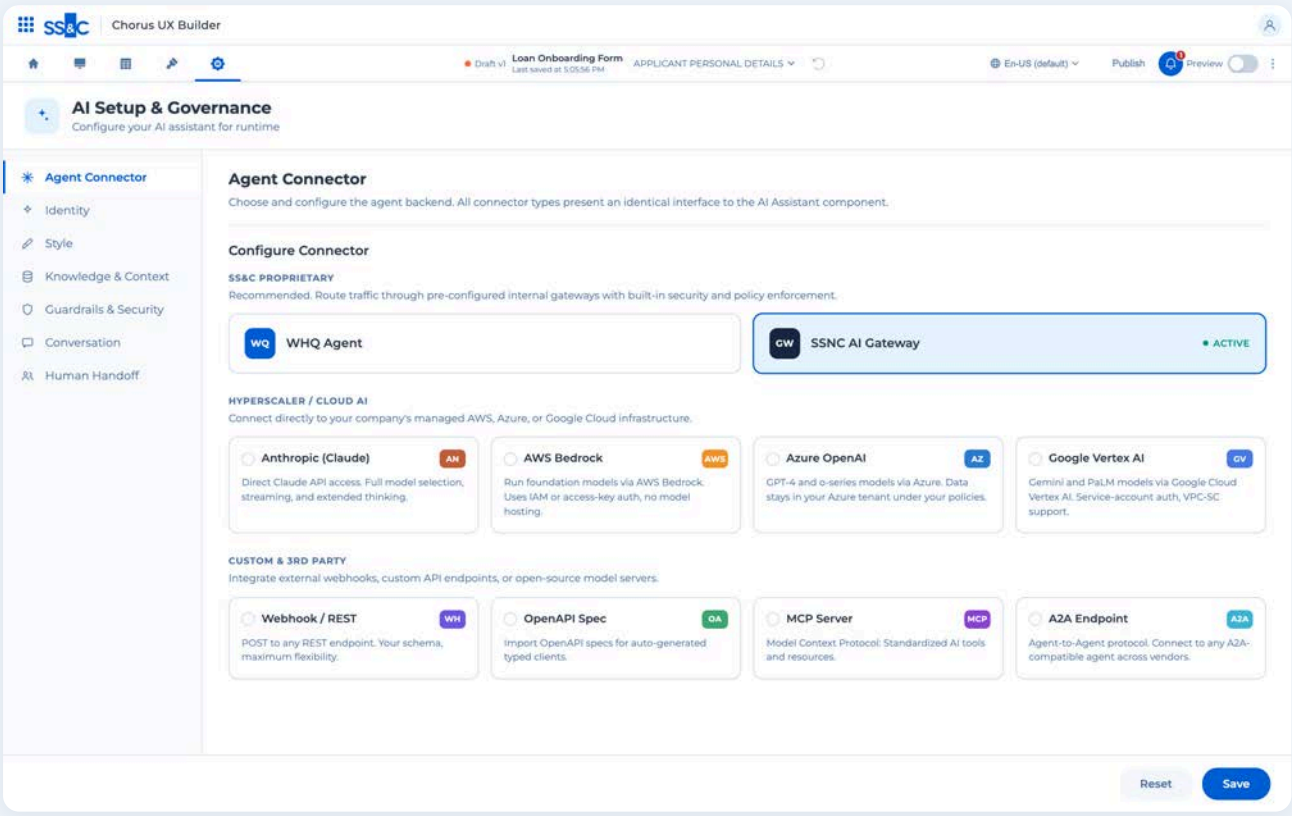
5 **Composer with attachments**
A soft-bordered input, "Attach files," and a primary Send button - consistent with every other Chorus control.

Primary #0760D5 **Chip #E5F4FF**

Orb gradient

One governed assistant, everywhere

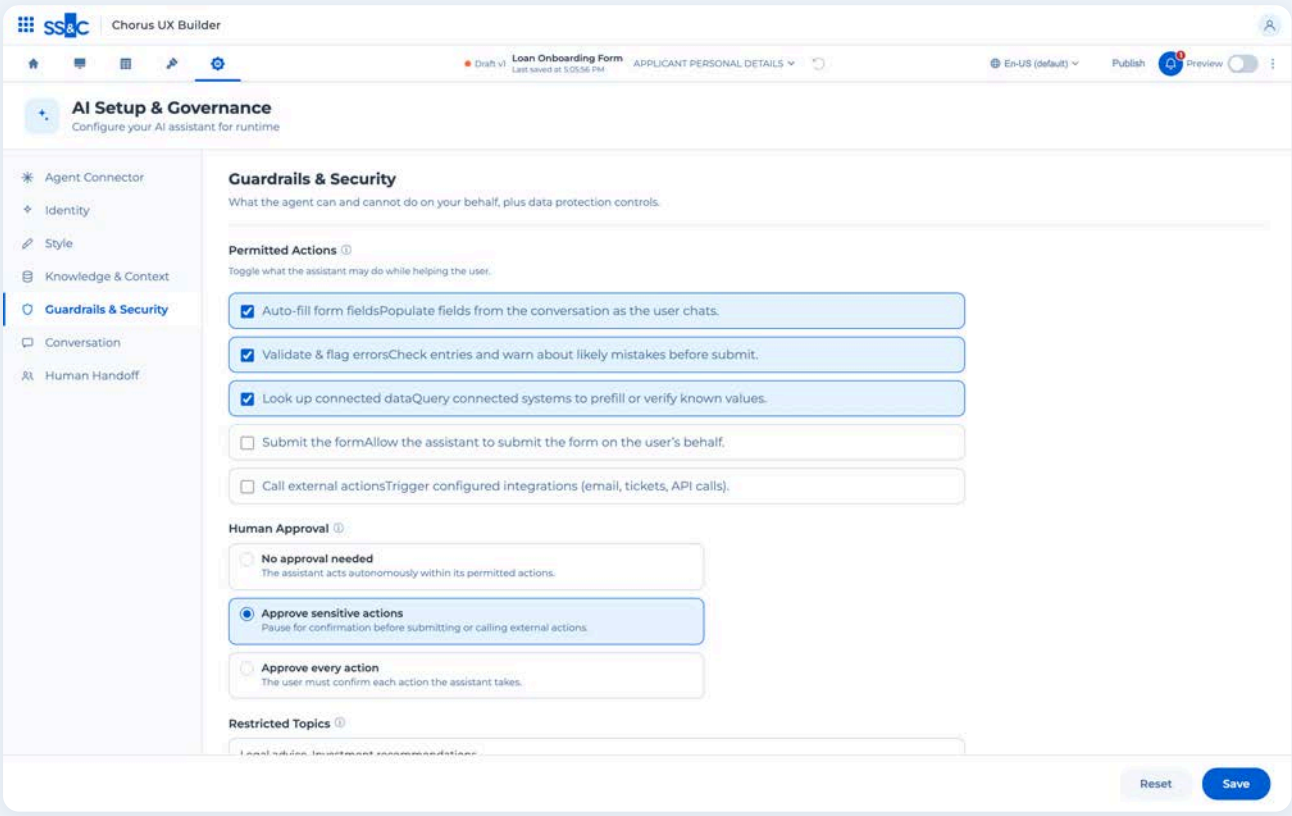
Across seven governed steps, the Designer connects an AI backend, gives it an identity and style, grounds it in approved knowledge, and constrains what it can do. Those guardrails travel with the form - the same assistant appears for the End User *and* the Processor, always inside the same limits.



Step 1 - Agent Connector. SS&C-proprietary gateways are recommended; hyperscaler (AWS, Azure, Google) and custom endpoints (Webhook, OpenAPI, MCP, A2A) are available for flexibility.

Constraining what the agent can do

Guardrails are the step that protects everyone downstream. The Designer toggles permitted actions, sets a human-approval threshold, and lists restricted topics - the exact configuration that keeps the End User's assistant helpful and the Processor's assistant read-only.



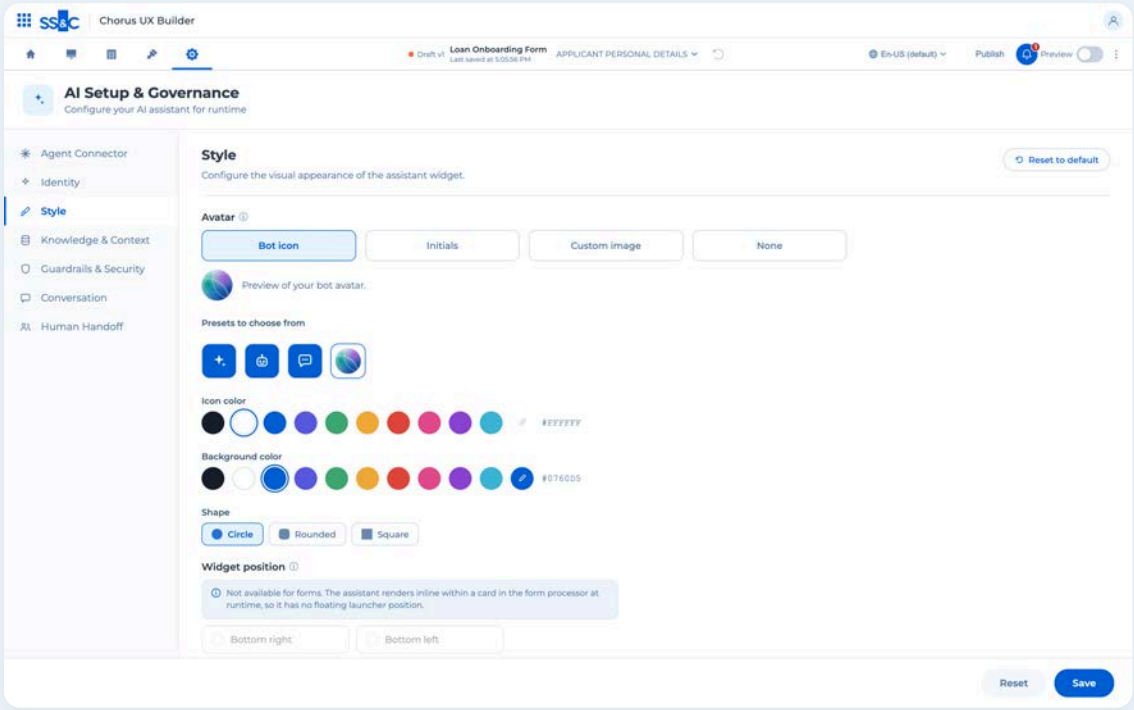
Step 5 - Guardrails & Security. Permitted actions, human-approval level, and restricted topics define what the agent can and cannot do on the user's behalf.

WHY IT TRAVELS

Because guardrails are attached to the form's AI configuration - not to a single screen - they follow the artifact into the published form and the Processor work item automatically.

Making the assistant look like yours

Governance decides what the agent *does*; the Style step decides what it *looks like*. From one panel the Designer sets the avatar, colors, and shape of the assistant widget - so the same Bluestone-native assistant carries its identity into the published form and the Processor work item, no re-styling downstream.



Step 3 - Style. Avatar, icon & background color, and widget shape define how the assistant presents itself - configured once, rendered everywhere.

Flexibility

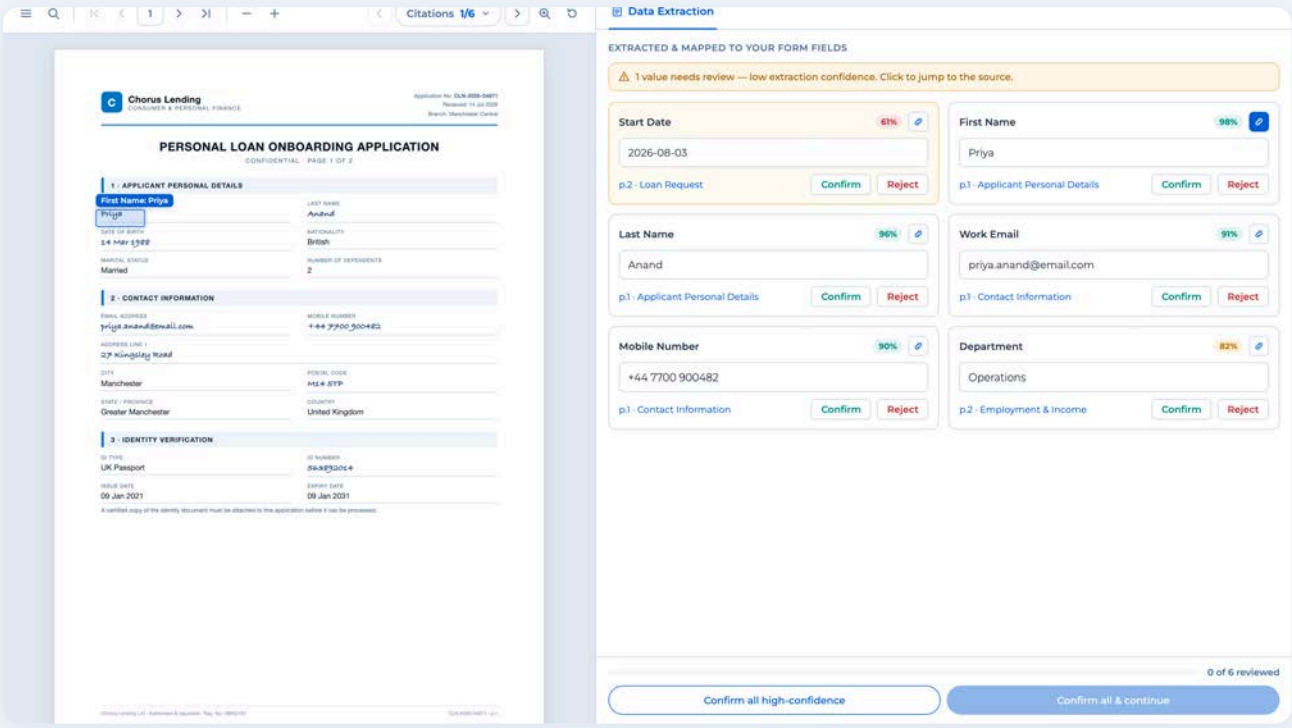
Pick an avatar, set icon and background from swatches or an exact hex, and choose Circle / Rounded / Square - all without code. The panel is context-aware, too: widget position is disabled for forms, since it renders inline in the Processor card.

The advantage

One visual definition, three surfaces. Because Style lives in the form's AI configuration, the look travels with the artifact - Designer, End User, and Processor all meet the same assistant, on-brand across the platform boundary.

Extracting a form from a document

Rather than re-typing everything, the End User uploads a source PDF - here, a consumer loan-onboarding application. The governed AI extracts each value, maps it to the right form field, and shows a **confidence score** with a link back to the exact spot in the document. The human stays in control: every value is theirs to confirm or reject.



Data Extraction - the uploaded PDF (left) with the active citation highlighted; extracted values (right) mapped to form fields, each with a confidence score, source link, and Confirm / Reject.

96%+

High confidence. Confirm in one click, or "Confirm all high-confidence."

82–90%

Worth a glance. Likely right; the human skims before confirming.

<70%

Needs review. Flagged; click the citation to check the source.

Trust through transparency, not automation

The design goal was never "let AI fill the form." It was to make the AI's work *inspectable* - so a person can trust it in seconds. Three interaction principles do the heavy lifting.

Confidence, surfaced

Every value wears its extraction confidence as a percentage and a color. The user's attention goes straight to what's uncertain - the 61% "Start Date," not the 98% "First Name."

Cited to the source

Each field links to its origin ("p.2 · Loan Request"). One click jumps the PDF viewer to that spot and highlights the exact text - verification without hunting.

Human holds the pen

Nothing is written to the form until the user acts. Confirm, Reject, or "Confirm all high-confidence" to clear the easy ones and focus on the rest.

1

END USER

Upload the document

Drag in the signed PDF. The governed AI (the same one configured at design time) runs extraction.

2

AI

Extract, map & score

Values are pulled, matched to form fields, and each is assigned a confidence score and a citation.

3

END USER

Review & confirm

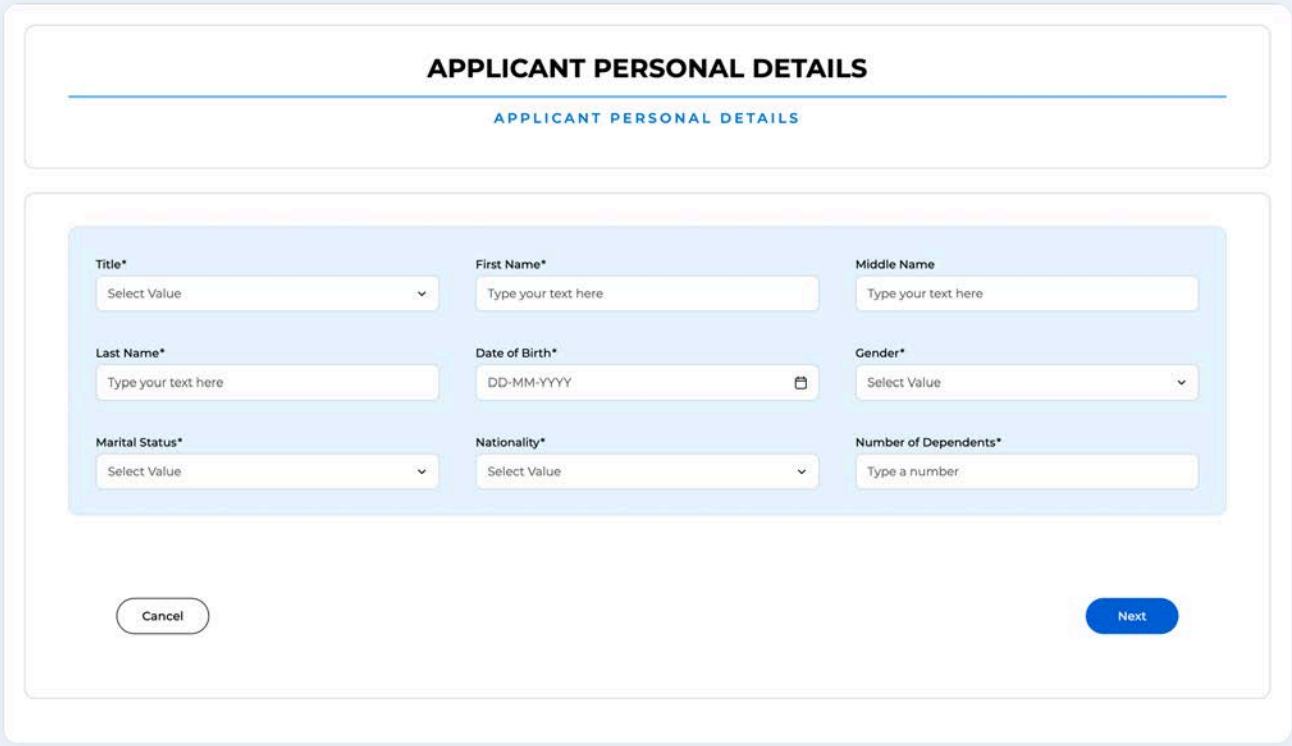
Low-confidence items are flagged first; the human confirms or corrects, then continues with a fully populated form.

THE COLLABORATION

AI does the tedious reading; the human does the judgment. Confidence scores and citations are what make that division of labor safe - the person is never asked to trust a black box.

What the End User receives

Whether typed by hand or populated from a document, the result is the same: a calm, branded, accessible form. No panels, no palette - just the task at hand. Required fields are marked, the layout breathes, and the governed assistant is one tap away. The applicant reviews and submits.



The published Loan Onboarding Form as an applicant sees it - the builder's chrome is gone; only the experience remains.

Clarity by default

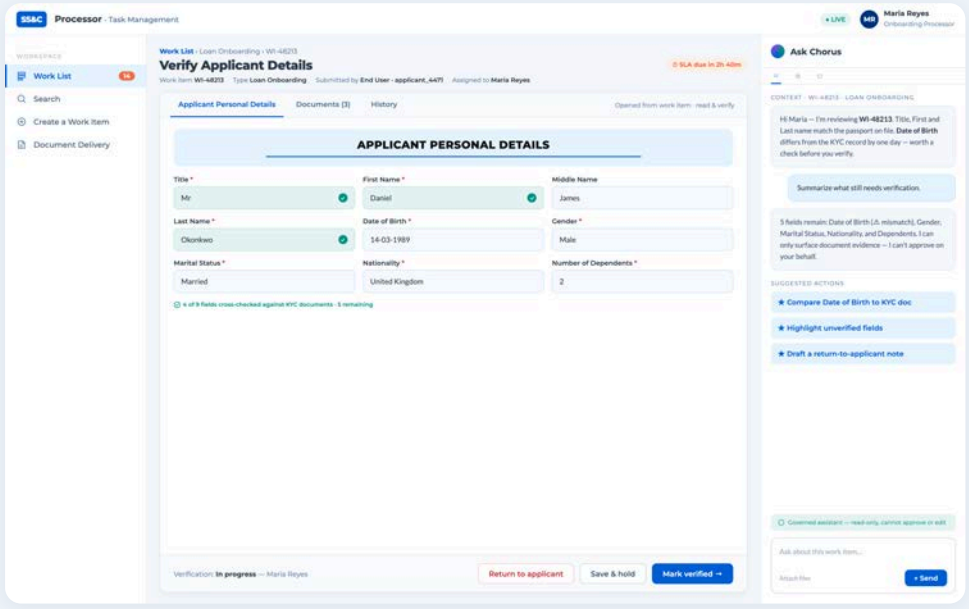
Consistent field cards, clear required markers, and a single obvious next action. Nothing competes for attention.

Then it becomes a work item

On submit, the completed form is routed into the Processor task platform as a work item - carrying its data, theme, and AI panel intact.

The same form, now a work item to verify

The submitted form opens inside **Processor**, SS&C's task-management (BPM) platform, as a card launched from the reviewer's work list. Because Processor runs the same Chorus design system, the form renders identically - and the same governed AI panel sits on the right, scoped to the work item and locked read-only.



Processor - the onboarding form opened as work item WI-48213 from the Work List (left: Work List · Search · Create a Work Item · Document Delivery; right: governed "Ask Chorus," read-only).

A task, not a tab

An assigned work item with an SLA, opened as a self-contained card - no re-keying, no context switch.

Verified in place

The applicant's answers appear in the exact form they filled; the reviewer cross-checks against KYC docs.

Same AI, tighter leash

The assistant surfaces evidence and flags mismatches but - per the design-time guardrails - cannot approve or edit.

WHY THIS MATTERS

The payoff of a shared design system: the artifact built once in UX Builder is trustworthy in a back-office reviewer's hands - governance and branding carried across the platform boundary automatically.

07 - THE HANDOFF

How one form crosses two platforms

The artifact - a single form definition and its governed AI configuration - is authored once and reused at every stage. Each persona receives the version fit for their job, on the platform they already work in.

DESIGNER

UX Builder

Designs the form, configures & governs the AI assistant, publishes v1.

→

END USER

Published form

Uploads a doc for AI extraction, reviews the values, and submits.

→

PROCESSOR

Processor (BPM)

Opens the submission as a work-item card, verifies it against documents, signs off.

What the form carries

The layout & theme - rendered identically in all three contexts. **The field data** - captured once, never re-keyed. **The AI configuration** - the same connector, identity, and guardrails everywhere.

What changes per stage

The chrome - builder panels, then nothing, then the Processor work list. **The AI's permissions** - editable at design time, extract-and-assist for the end user, read-only for the verifier. **The intent** - build → complete → verify.

THE DESIGN PRINCIPLE

Author once, render everywhere, govern consistently. A shared design system isn't cosmetic - it's what lets a work item feel native in a BPM tool it was never explicitly built for.

Testing across all three personas

To validate the shared-artifact thesis, we ran a moderated study against the incumbent legacy stack. Designers built the loan-onboarding flow, end users completed the published result (including AI document extraction), and processors verified the submissions. The numbers below are illustrative of the study design.

PARTICIPANTS	METHOD	TASKS	DURATION
30 total · 10 Designers · 10 End Users · 10 Processors	**Moderated remote · Within-subjects A/B**	**4 build · 3 completion · 3 verification**	**45-min sessions · Over 4 weeks**

84/100
SUS - Designers (UX Builder)
▲ vs 58 legacy

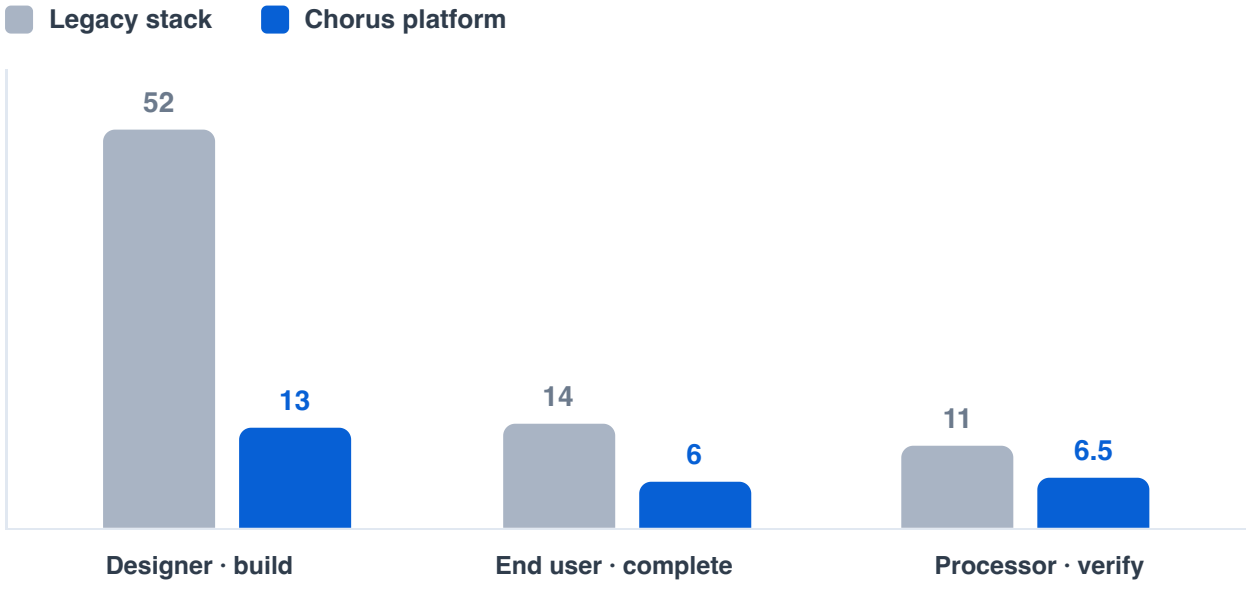
73%
Less manual data entry via AI extraction
▼ keystrokes

41%
Faster median verification (Processor)
▼ from 11m

0
Fields re-keyed by the Processor
▼ from ~9

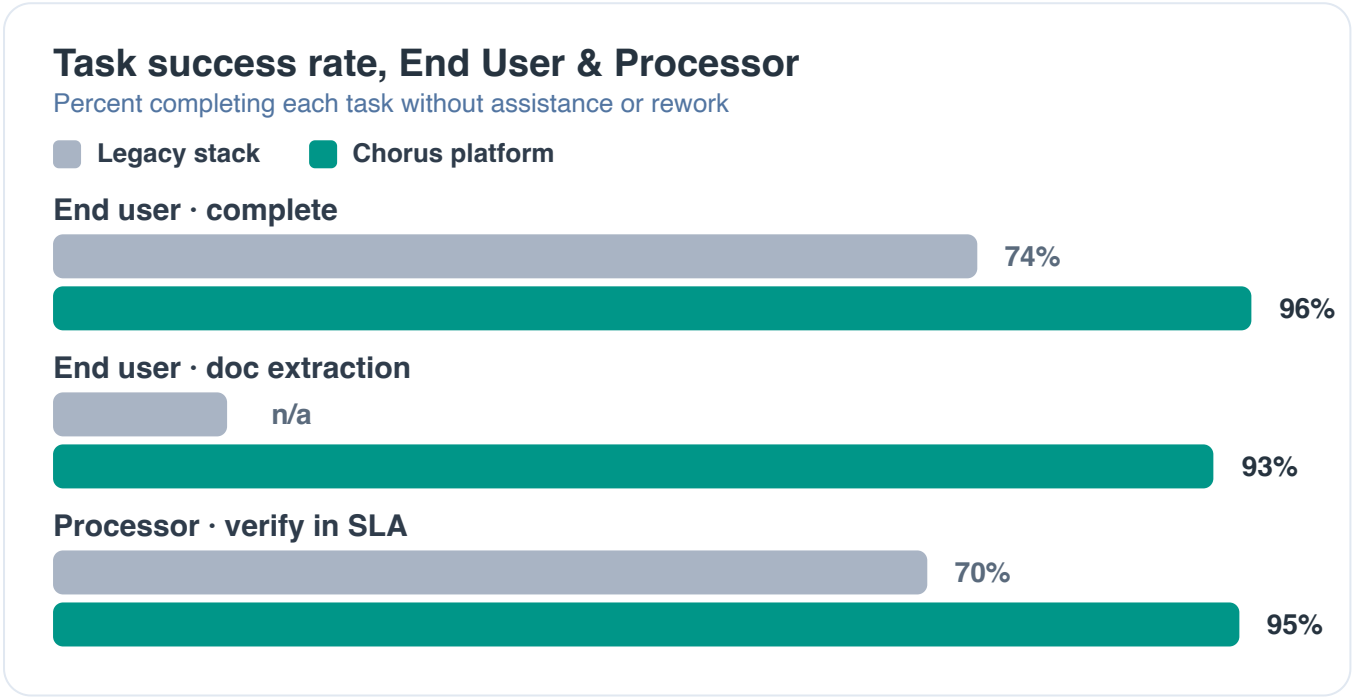
Median time on task, by persona

Minutes · lower is better



What the data told us

The gains compounded down the chain. Designers shipped faster; end users succeeded more often and typed far less thanks to AI extraction; and because the submission arrived pre-populated, processors verified without re-keying a single field.



Designer

Self-served AI setup replaced developer tickets. SUS 58 → 84 ("excellent").

End User

AI extraction cut manual entry ~73%; confidence scores + citations kept people in control.

Processor

Zero re-keying and a familiar UI cut verification time 41%; SLA hit-rate 70% → 95%.

> *"I uploaded the PDF and it filled the form - but I could see exactly where each value came from and how sure it was. I only had to check the one it flagged."*
>
> **- End User participant, P22 (illustrative)**

READING THIS DATA

Figures on these two pages are fictional and illustrative - included to demonstrate the study design and how results would be reported, not measured outcomes.

09 - END TO END

The full journey, all three personas

A single unbroken thread runs from the Designer's first drag, through the End User's AI-assisted submit, to the Processor's sign-off - across two platforms, without the artifact ever being rebuilt.

1

DESIGNER · UX BUILDER

Create, compose & govern

Drag sections and fields onto the canvas; walk the 7-step AI Setup to connect and constrain the assistant; publish.

2

END USER · PUBLISHED FORM

Upload & extract

Uploads a source PDF; the governed AI extracts, maps, and scores each value with a citation.

3

END USER · PUBLISHED FORM

Review & submit

Confirms high-confidence values, checks the flagged one against the source, and submits a complete form.

4

PROCESSOR (BPM)

Open as a work item

The submission lands in the Work List; the Processor opens it as a card - same form, pre-populated.

5

PROCESSOR (BPM)

Verify against documents

Cross-checks fields against KYC docs with the read-only assistant surfacing evidence and flagging mismatches.

6

PROCESSOR (BPM)

Sign off within SLA

Marks the work item verified - closing the loop the Designer opened, with governance intact throughout.

THE PAYOFF

The Designer never wrote code. The End User barely typed. The Processor never re-keyed a field. One governed artifact served all three - shipped and processed in hours, not sprints.

My Design Process

Six phases carried Chorus UX Builder from first conversations to a governed, cross-platform artifact - grounded in research, tested with every persona, and refined before a line of code was written.

1 Discovery

- Partnered with Product, Engineering, and SMEs.
- Reviewed existing workflow pain points.
- Identified developer bottlenecks and disconnected runtime experiences.

2 Research

- Interviewed designers, processors, and business analysts.
- Observed existing onboarding workflows.
- Identified friction around manual configuration, inconsistent branding, and lack of AI trust.

3 Problem Definition

- Reduce engineering dependency.
- Maintain one artifact across platforms.
- Build trust into AI interactions.

4 Ideation

- Explored multiple IA approaches.
- Evaluated different AI interaction models.
- Prototyped several layout concepts.

5 Validation

- Walked prototypes with users and stakeholders.
- Refined navigation, hierarchy, and AI workflows.
- Incorporated feedback before development.

6 Outcome

- Final design system.
- Shared runtime.
- Governed AI experience.