

ENTERPRISE UX CASE STUDY

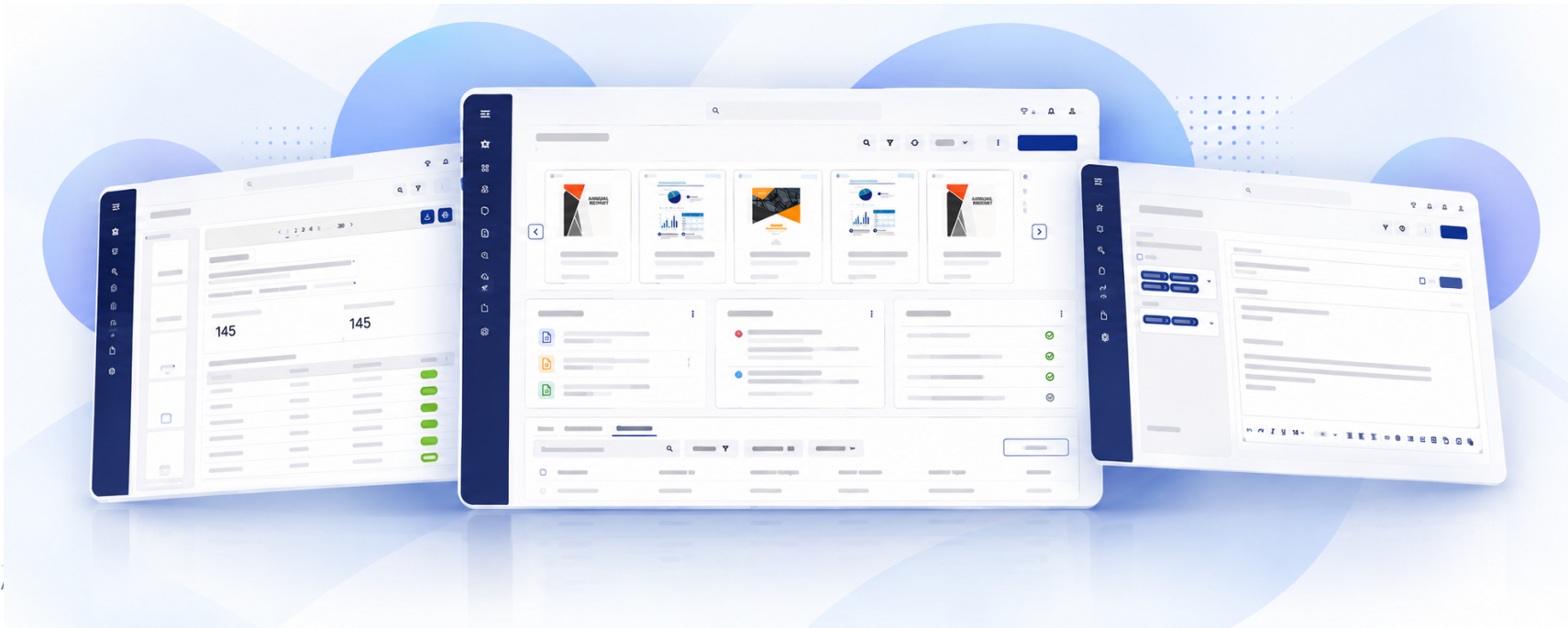
ASIO Report Builder

Redesigning a complex reporting workflow into a scalable, guided experience for MSPs and small-business users.

Role	Focus	Platform	Product
UX / Product Designer	Research, IA, user flows, interaction design	Enterprise SaaS / Web	ConnectWise ASIO

DESIGN OBJECTIVE

Preserve the flexibility of advanced reporting while making report creation easier to understand, faster to complete, and more useful throughout the reporting lifecycle.



Turn complex data into meaningful, actionable reports

The user goal of BrightGauge Report Builder is to empower users to create customized and visually appealing reports that provide valuable insights and analytics. Users should be able to gather data from different sources, define reporting requirements, select relevant information, and present it in a way that is easy to understand and act on.

Gather

Bring relevant data and metrics into one reporting workflow.

Customize

Start from a template or create a layout tailored to a specific client or business need.

Understand

Use visual reporting to track performance, identify trends and reveal important changes.

Act

Share or schedule reports so insights can support timely decisions.

The broader objective is to help users improve their understanding of key metrics, monitor performance and make informed decisions. Data visualization is not the end goal; it is the mechanism for improving business visibility, efficiency and decision-making.

PRIMARY EXPERIENCE PRINCIPLE

Give users a clear path from data selection to report design, validation and delivery - without exposing unnecessary complexity too early.

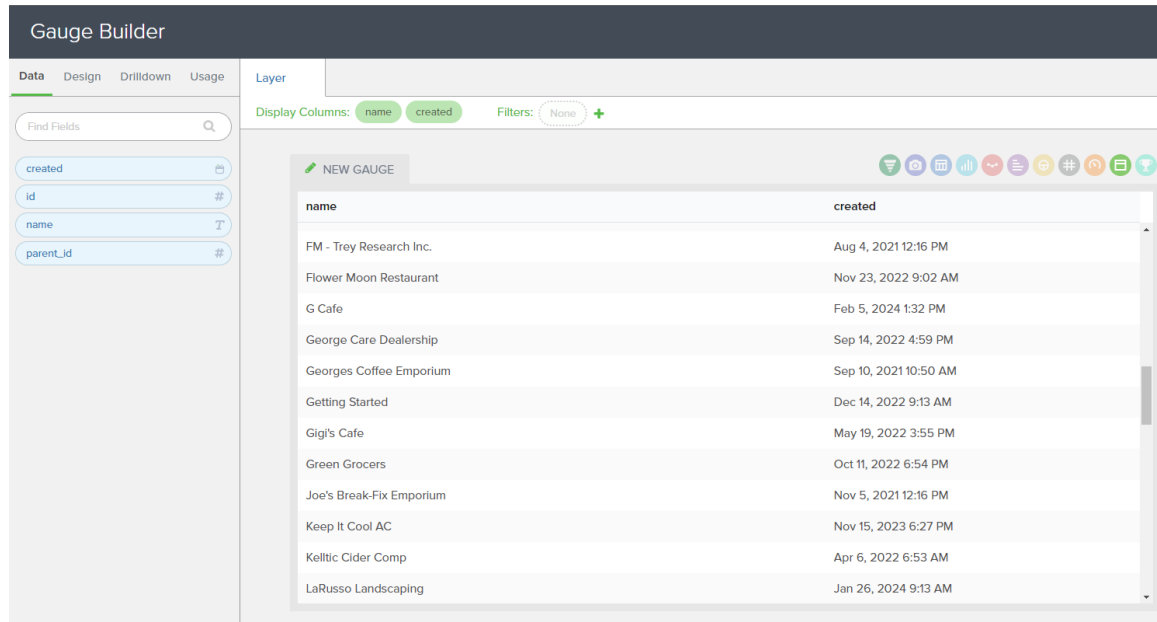
Designing for different levels of reporting expertise

The experience needed to support users who primarily consume insights as well as administrators who configure and automate complex reporting workflows.

Small Business Owner	MSP Administrator
Context Needs concise visibility into business or client performance without spending significant time configuring analytics.	Context Manages reporting across multiple customers, data sources and recurring operational workflows.
Goals • Understand important KPIs quickly • Receive clear, visually understandable reports • Identify trends and issues • Make informed business decisions	Goals • Build reusable reports efficiently • Configure filters and data sources • Maintain consistency across customers • Schedule recurring delivery
Pain points • Reporting terminology can feel technical • Too many controls increase cognitive load • Needs useful defaults and templates	Pain points • Repetitive setup across clients • Complex configuration and permissions • Needs advanced control without a cluttered UI
Design implication Prioritize templates, clear hierarchy and understandable outputs.	Design implication Support reusable structures, progressive controls, filtering and automation.

Existing Gauge Builder

The legacy builder was powerful, but the interface exposed a large amount of reporting functionality simultaneously. Flexibility came with a significant learning curve.



High cognitive load - many controls were visible at once.

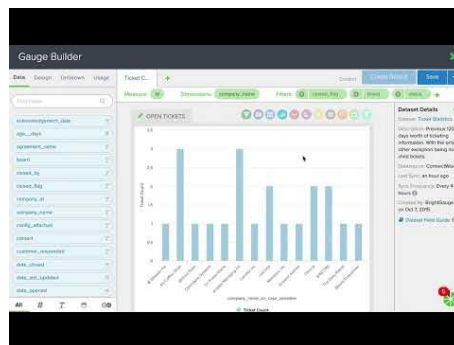
Weak hierarchy - primary and secondary actions competed for attention.

Fragmented workflow - Data, Design, Drilldown and Usage were separated.

Limited guidance - users needed prior knowledge of reporting concepts.

Icon-heavy controls - actions were harder to discover for new users.

Scalability risk - adding capability risked increasing interface density.

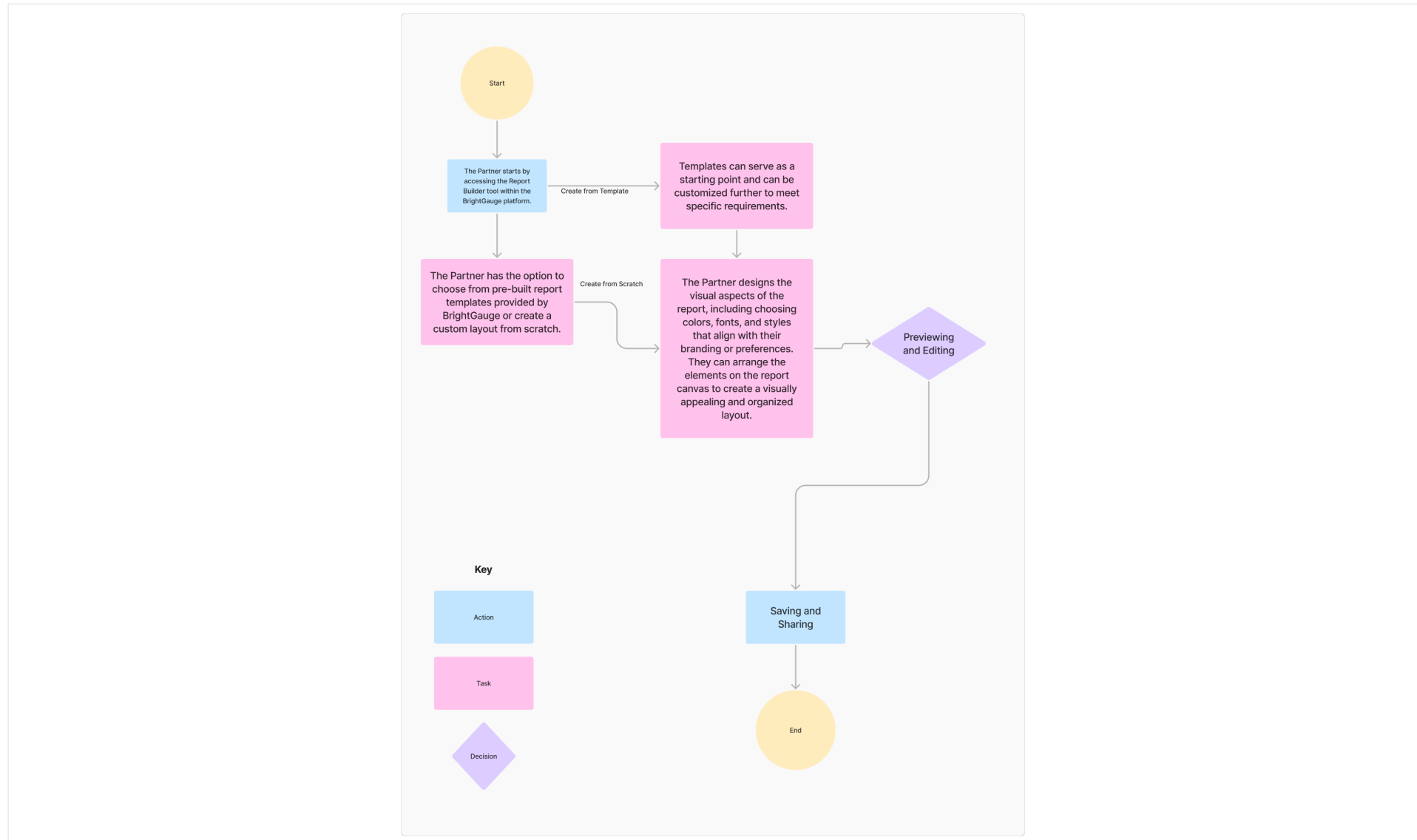


Design opportunity

How might we preserve the power of Gauge Builder while making report creation easier to understand, faster to complete and scalable for future reporting capabilities?

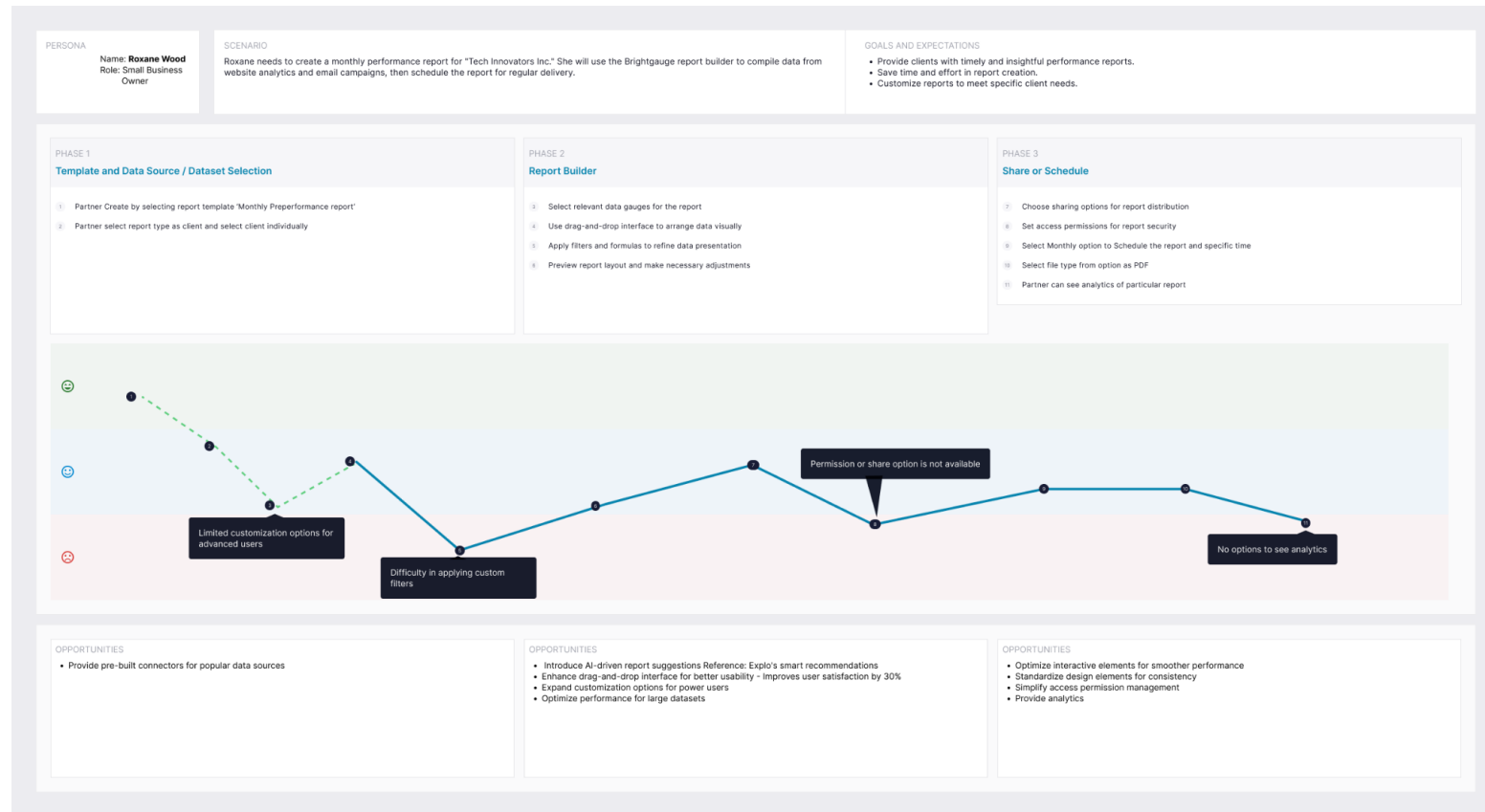
A clearer path from creation to delivery

The flow supports two entry paths - starting from a template or creating from scratch - before moving through report design, preview/editing, and finally saving or sharing.



Where the reporting experience breaks down

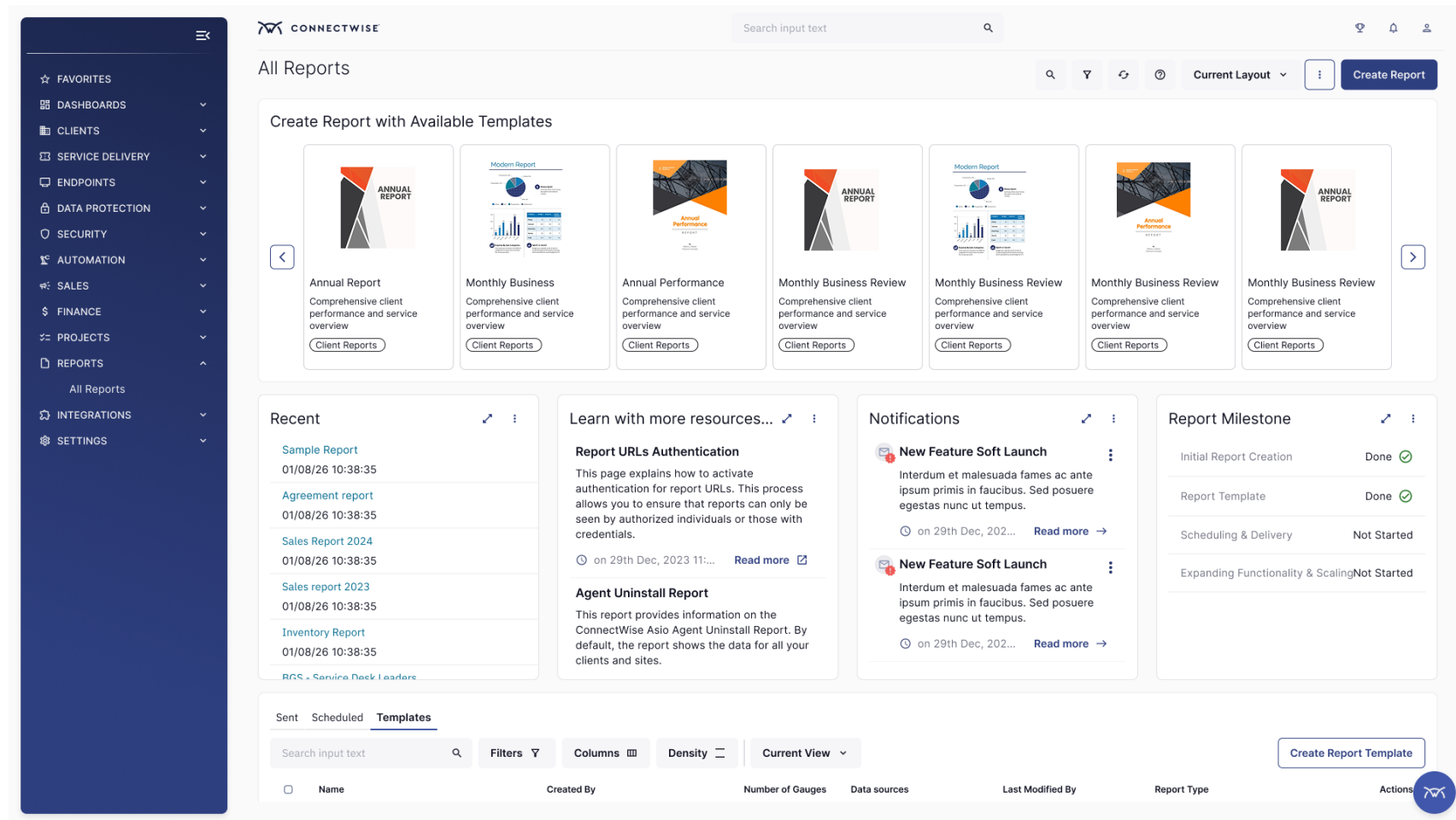
The journey map captures the end-to-end reporting process across template/data selection, report building, and sharing or scheduling. It highlights friction around customization, custom filters, permissions/sharing and analytics.



Key design takeaway: The most important opportunities were not isolated to a single screen. The redesign needed stronger continuity between selecting data, constructing the report, validating the output and distributing it.

From a simple report list to a reporting workspace

I expanded the report entry point from a basic list into a richer landing experience that helps users discover, create and manage reporting work before entering the builder.



Templates

Prominent starting points reduce setup effort for common reporting scenarios.

AI-assisted reporting

The information architecture allows a faster report-creation path using AI-assisted generation.

Learning resources

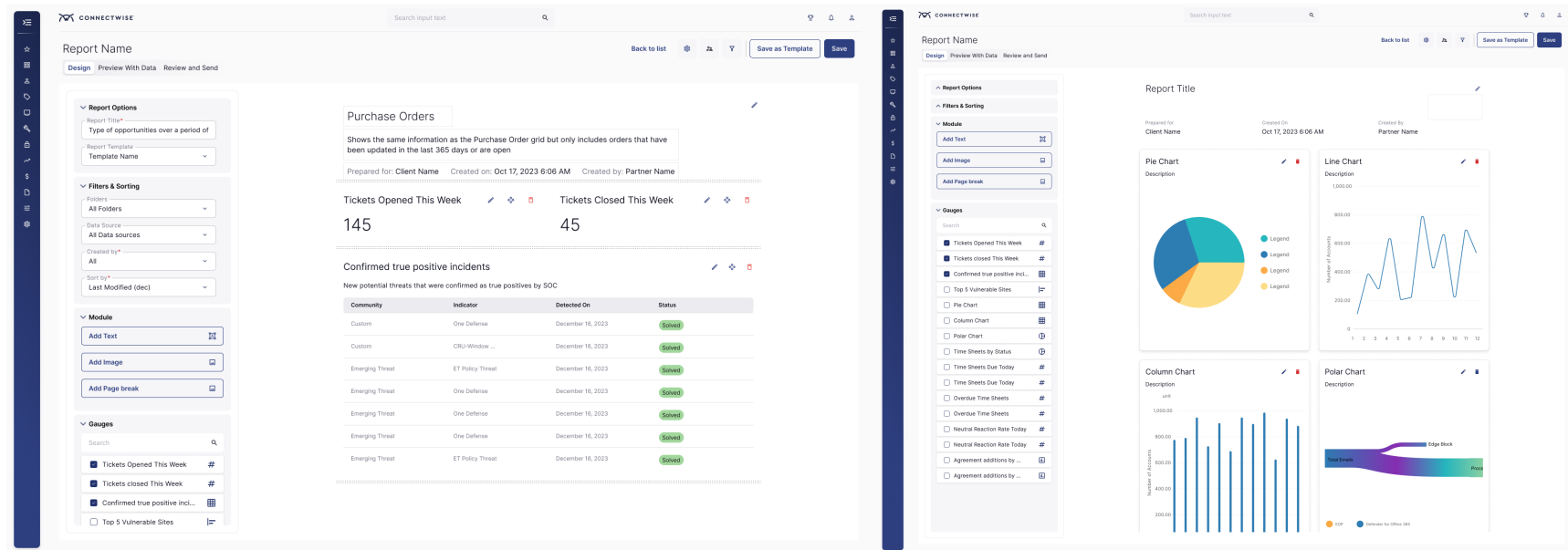
Contextual education helps users understand reporting features and workflows.

Milestones & notifications

Progress and product updates create visibility beyond the report list itself.

Progressive control without losing report context

The redesigned builder keeps report content central while moving options, filters, modules and gauges into a structured side panel. Users can work with KPI blocks, tables and visualizations while maintaining a clear relationship between configuration and output.



Contextual configuration

Controls are grouped in the left panel instead of competing with the report canvas.

Modular report composition

Text, images, page breaks, gauges and visualizations support flexible layouts.

Visible hierarchy

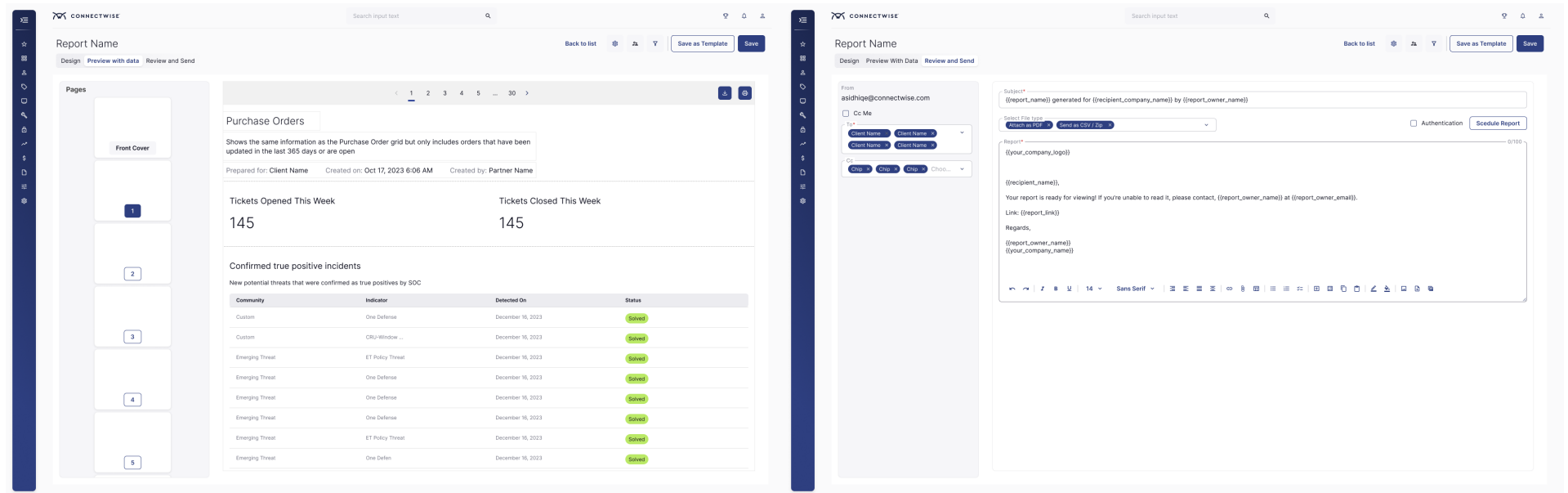
Report metadata, KPIs and data visualizations are clearly separated.

Reusable reporting

Templates and saved structures reduce repeated setup across clients.

Make validation and distribution part of the same workflow

Instead of treating export and delivery as separate tasks, the experience moves naturally from design to preview and then to review/send. This helps users validate the report before it reaches a customer.



Preview with data

Page navigation and realistic report content help users verify the final output before distribution.

Review and send

Recipients, file type, authentication, message content and scheduling are consolidated into the delivery stage.

A reporting experience designed to scale

- | | |
|----|---|
| 01 | Template-first entry
Provide useful starting structures while preserving a create-from-scratch path. |
| 02 | Progressive disclosure
Show common controls first and reveal advanced configuration when needed. |
| 03 | Contextual editing
Keep settings connected to the report content being changed. |
| 04 | Integrated preview
Allow users to validate real report content before distribution. |
| 05 | Delivery in the workflow
Connect recipients, formats, authentication and scheduling directly to the report lifecycle. |
| 06 | Richer landing page
Support discovery and management through templates, learning resources, notifications and milestones. |

Clearer workflow

Creation, preview and delivery are connected.

Lower visible complexity

Advanced controls do not dominate the default experience.

Better repeatability

Templates and reusable structures reduce repetitive work.

Stronger foundation

The architecture can accommodate future reporting capabilities.

KEY LEARNING

Simplifying enterprise software does not mean removing capability. The design challenge is deciding when functionality should appear, how it should be grouped, and how much information users need at each stage.

Outcome statements are intentionally qualitative; no unverified performance metrics have been introduced.