

# Cultural Daily

Independent Voices, New Perspectives

## How Notion and Stripe Redesigned Their UX Flows as They Scaled and What Growing Teams Can Take From It

Our Friends · Tuesday, April 28th, 2026

There is a version of both Notion and Stripe that most designers never think about: the early one. Notion's first public release in 2016 was a personal note-taking tool aimed at individuals who wanted something more flexible than Evernote. Stripe launched in 2011 with a checkout flow targeted almost entirely at developers who needed a faster way to accept payments. Neither product looked like what it looks like today, and neither was designed for the audience it now serves. Watching how both companies adapted their UX as their user bases grew is one of the more instructive stories in recent product history, and the full record of their flows across different product types is available in a [directory of 1,000+ apps with real UX examples](#) that documents how these products changed over time.

The two companies got much bigger than their original founders thought they would. Notion added 80 million people to its user base, with 100 million total users worldwide by 2024, compared to just 20 million users in 2022; Stripe processed \$1.4 trillion worth of payment volume globally in 2024, vs \$1 trillion for 2019; that's a 40% increase year over year. This number of additional users is a major milestone for each company, but an even bigger milestone is the design challenge they face as each of these companies were designed when there were no students and no Fortune 500 companies using the products at the same time. The UX assumptions that were used to create the original product no longer hold true when you have such different user types engaging with your product.

### Notion: From Blank Canvas to Guided Workspace

#### The Onboarding Split

Notion's most significant UX change as it scaled was the decision to branch its onboarding based on user type. Early versions of the product offered everyone the same empty workspace and left them to figure it out. That worked for users who already had a clear idea of what they wanted to build. For everyone else, the blank canvas was overwhelming. As Notion grew beyond its initial audience of knowledge workers and indie developers, the company introduced a fork early in the signup flow: personal use or team use.

This single structural decision unlocked a much more directed experience for both groups. Individual users landed in a simplified template gallery. Team users were introduced to shared workspaces, given a brief explanation of how collaborative pages work, and prompted to invite their first collaborator. Neither group had to wade through features that belonged to the other. The

result was a more focused first session for both.

## Templates as a Scaling Mechanism

The template feature of Notion's software evolved from being a simple tool to being a major contributor to the growth of Notion's business. The use of templates provides a way to lower the effort needed by users to begin using Notion, which makes it easier for users to use Notion that likely otherwise would never have learned how to use Notion's block based system.

This was in the form of two types of template: company curated templates for more common use cases, and community created templates for now common but less commonly used processes/workflows. By the end of 2022 Notion had converted over 4 million users into customers by utilizing templates as one of the many contributing factors to its conversion funnel.

## What Grew More Complex Over Time

As Notion added databases, automations, forms, and its AI features in 2023, the settings architecture had to absorb that complexity without surfacing it to users who did not need it. The settings screens Notion has today reflect a layered approach: basic account controls appear immediately, advanced permissions and workspace analytics are available deeper in the hierarchy. That structure did not exist in the early product. It was built as the feature set expanded, and the fact that it remains navigable is a design achievement that is easy to miss.

## Stripe: Designing for Two Audiences at Once

Stripe's original onboarding was built for developers. It offered sandbox access, a magic link for authentication, and clear API documentation before asking for any business details. That flow worked well for its intended audience, and the developer-first reputation it built became a real competitive advantage.

The challenge arrived when Stripe began serving non-technical business owners, finance teams, and enterprise clients alongside the developers. A checkout integration flow that assumes comfort with API documentation is completely opaque to a CFO trying to understand monthly reconciliation.

Stripe's response was not to simplify the developer experience. Instead, the company built parallel paths that serve different user types without degrading the experience for any of them. The dashboard evolved from a raw data view into a configurable surface with modular sections. Businesses processing over \$1 billion annually on Stripe get access to enterprise controls, custom reporting through Stripe Sigma, and team management features that were nowhere in the original product. Over 100 companies reach that threshold today.

When Stripe redesigned its Connect onboarding flow for marketplace platforms, it reported an average 5.3 percent increase in conversion rates across Express accounts. One platform saw a 17 percent improvement after the update. Those numbers reflect the business value of a flow that previously created friction.

## What Growing Teams Can Actually Use

The pattern across both companies is consistent enough to extract:

- Segment early in the flow by user type, not by feature set
- Build template systems or guided paths that lower the entry barrier without removing flexibility
- Keep advanced features accessible but out of the default view
- Design settings architecture that can absorb new features without requiring a full rebuild
- Measure conversion at each step and treat drops as design problems, not user problems

Neither Notion nor Stripe stayed attached to their original UX logic when the evidence said it was no longer serving their actual user base. That willingness to redesign foundational flows, not just surface-level UI, is what made the difference. Most growing teams delay those decisions too long, which means the debt compounds alongside the user count.

*Photo: Ofspace LLC, Culture via Pexels*

---

**[CLICK HERE TO DONATE IN SUPPORT OF OUR NONPROFIT COVERAGE OF ARTS AND CULTURE](#)**

This entry was posted on Tuesday, April 28th, 2026 at 9:13 am and is filed under [Check This Out](#). You can follow any responses to this entry through the [Comments \(RSS\)](#) feed. You can leave a response, or [trackback](#) from your own site.