

Collaboration Design Group Assignment

Presented to

Mr. Beau Giannini

Teamwork and Collaboration

MGT-6054

By

Yunpeg (Jasper) Ge , Vito Grassano, Alexandre Ly, Vinicius Muniz, Analiza Pelovello, Ali Amaan Rizvi

22/01/26

Part 1: Collaboration Context

In the world of e-commerce, Stripe acts as facilitator for payments, tax assistance and fraud detection in the form of banking-like features through simple APIs for businesses (Stripe). It allows payments and money flow into one unified location on an app or a website. Stripe differentiates themselves amongst Shopify, Amazon Pay and Squarespace by providing flexible payment and financial infrastructure that businesses can deeply adapt to their own products via APIs. While tools like Shopify bundle storefront and ecommerce features, Stripe focuses on being the payment “bridge,” handling payments, fraud prevention, and tax logic between customers and businesses behind the scenes. In brief, Stripe built its reputation as one of the most efficient payment processors since Paypal with almost 1.4 trillion dollars in total payment processed in 2024, expanding tremendously year over year roughly 25-40% since 2023 (Lee). The reason of their success remains in the structure of their teams that focuses on payments processing lifecycles and stages. For example, the first stage of the payment lifecycle in the initiation of a payment where a client approaches the store physically or virtually for checkout. Indeed, this stage would be in the ownership and responsibility of the product design team, but collaboration with engineering is essential to ensure the proposal are technically feasible. Aligning highly specialized groups around core payment processing needs minimizes silos and people working unilaterally maximizes speed and efficiency. This goes beyond just helping the clients need to process payments, but it also allows room for further improvement, an aspect even the client is not aware that they will need. In the past example of the payment initiation stage, Stripe have anticipated the potential need of an AI assistant for fraud detection into production systems that actually reduce fraud and increase authorization rates, rather than just producing dashboards (Khan). Their collaboration structure helps nurture opportunities to oversee a project through multiple dimensions with cross functional teams resulting in a more well-rounded solution.

Part 2: Diagnosis (Where Collaboration Breaks)

The first pressure point that Stripe faces reside in the trade-off between prioritizing speed or safety in their product iteration. As Facebook or now META would refer to their motto: “Move fast and break things”, this culture cannot be replicated in the financial infrastructure building industry (Strategy Breakdowns). The decision to either 'jump in' immediately or cautiously assess the depth of the pool represents a challenge that most fintech companies face. The concept of “compliance debt” introduces the continuous process of reiterating their controls and partner programs as regulations tighten, makes it an obstacle for Stripe’s team to work more rapidly than planned (Stripe). Not only this causes friction for the clients using their product and programs, but also for the teams at Stripe having to continuously monitor collaboratively. In recent news, an example showcasing this problem arise with Flipcause, where teams at Stripe prioritized swiftness over compliance where multiple violations where flagged post-launch where the teams did not catch the initial legal gaps until their financial partners raised

alarms (Shabazz). As a result, a fine of $137 500 was attributed to Stripe by Mastercard, proving this dilemma between speed and safety still represent an ongoing struggle.

Furthermore, constraints like global versus local fiscal requirement do affect team structures from the Stripe offices in New York to Paris. One structure does not fit all teams around the world anymore due to local regulatory demands, an adaptation that Stripe is having difficulty where the regional squads are left retrofitting compliance that pushes the process of building the product to restart again to fit the Brazilian market for example. Indeed, the Brazilian Central Bank function with an instant payment system that relies on QR codes also known as Chave Pix for anyone to send money, where this infrastructure functions fundamentally differently from US-centric money flows, it requires bespoke engineering solutions. And so, the demand for market expert and a restructuration of the personnel is needed to be able to respond to Brazilian regulations and market-specific needs.

Moreover, Stripe's shift to PaymentIntents and PaymentMethods as unified abstractions exposed collaboration weaknesses by revealing how siloed product teams had independently evolved inconsistent APIs (Railsdrop). Specifically, groups like Payments and Billing worked on separate roadmaps with conflicting incentives (growth vs. retention), leading them to build incompatible software. This disconnect meant that critical conflicts were missed until they became emergencies, forcing leadership to intervene and mandate a massive, top-down redesign to fix the drift caused by prioritizing speed over coordination.

Part 3: Design Principles

In an environment where innovation needs to be fostered and refined, the companies need to assert their value and alignment of their culture. One of the principles can be rooted in building structures pre-mortem where cross-functional teams can quickly collaborate and experiment with their initiatives creating rapport of ownership with their ideas and concretely executing it while keeping their platform still functional. For example, Stripe iterates a product that is always subject to a changing environment, the financial behaviours of their client are always in need of a specific function or feature. Indeed, their collaboration team should adapt their method of work to be able to anticipate which feature or action they can implement in order to better provide our clients with the tools they need to make their businesses work. When enabling and structuring these conditions to collaborate and experiment, legal team for example can catch reliability or compliance issues when developing a new feature for Stripe’s product with the product team when they are building the feature.

Which leads to the second principle, their collaboration must enable a shared global core with explicit structures adapted for most regions to then be adapted to regional squads attending the matter of cash flows, compliance, and engineering patterns early rather than restarting the build when Brazil’s requirement finally surface. Concretely, encouraged behaviours such as shared design docs and

reviews are introduced to programmers for information to be unified and standardized. These little changes may seem banal, but their impact allow a more efficient alignment and reduces asymmetries of information across all teams. Once standards are set, global APIs are set and agreed upon for the regional teams in Hong Kong or Paris to adapt the main platform to accommodate their local requirements pushing better development and user experience as innovation outcomes. Indeed, Faster rollout of new payment methods and flows across markets becomes possible when teams share consistent, well-understood primitives like unified APIs.

Lastly, the root cause of siloed teams is embedded in the conflicting incentives attributed to the different teams like Payments and Billing. Joint decision forums and leader reframing would have aligned the enterprise goals upfront by evaluating both growth and risk metrics rather than prioritizing speed over safety or vice-versa. Having teams with contradicting incentives and goals creates confusion and conflict in isolated teams. To remediate the situation, regular joint decision forums tied to launches encourage the global teams and local teams to share their view of risk and growth of a project or product development, avoiding having hardened separate roadmaps across teams like Payments and Billing mentioned in the past example. By documenting clearly the concerns or the gaps and committing to shared enterprise outcomes, management and technical teams can prevent quiet resistance or local-win bonuses, delivering durable innovation without emergency redesigns.

Part 4: Proposed Collaboration Mechanisms

To address these challenges, our team came up with a set of solutions that can be implemented at scale. Because the coordination failures at Stripe usually involve not only multiple teams but also multiple locations across the globe, it is important that the solutions are simple and efficient. The company has enough complexity to deal with when it comes to their core product so adding more complexities coming from process would be suboptimal.

One issue that is often at the center of collaboration issues is the lack of information. As a general principle, we suggest the information is readily available and easy to access across different teams. Goals, results, progress and even failure to hit targets should be public information across different teams (excluding sensitive information about customers, for example). This helps giving the employees a general sense of where the company is right now, and how likely it is to hit its targets. From that shared understanding of the current situations, teams can understand each other’s difficulties and work together addressing the most pressing issues.

Some practical recommendations we came up with are:

Shared Design Dossier: Stripe could benefit to have a shared document detailing core payment infrastructure, consolidating technical architecture. The document should be updated as often as

there are changes and publicised through the company so it can become the go to document for understanding the process. It can also be used as instrument for training new hires and as a guide for decision making.

Ownership Maps: Each critical component of Stripe's core business should have a clear global owner. Someone the team can refer to in case there are disagreements or questions. There should also be local owners, who report to the global owner. These local owners should be able to help with everyday question and be the ones who escalate issues when they are not addressable locally or when the lessons learned from a local situation can be shared globally and help format the workflow for the whole company.

Risk Escalation Process: Lastly, for some critical categories, such as regulatory and financial risk, it is important to have a clear escalation line through a cross functional chain of owners. This should allow for clear and quick decision making on critical issues when timing is important. This would reduce ad-hoc decision making and ensure decisions set up precedents and start becoming the norm on how to handle situations that tend to repeat themselves.

Part 5: Trade-offs and Limits

Our collaboration design does not eliminate strategic disagreement between teams with fundamentally different incentives, prevent external shocks, such as regulatory changes or reputational crises, from overwhelming coordination capacity, and guarantee alignment when organizational strategy or priorities are themselves ambiguous

These failures lie outside the scope of collaboration design. Authority, process, and structure can support coordination, but they cannot replace strategic clarity or eliminate uncertainty in Stripe's external environment.

No collaboration design can fully eliminate friction in an organization as fast-growing, widely distributed, and technically complex as Stripe. The proposed collaboration mechanisms intentionally prioritize system reliability, coordination clarity, and regulatory resilience. These priorities create unavoidable trade-offs that shape where collaboration may still fail and why those limits are acceptable within Stripe's operating context.

Firstly, the design emphasizes clear ownership, bounded decision rights, and structured escalation to prevent coordination paralysis in critical payment infrastructure. While this supports fast execution, it limits broad participation in certain decisions. Some teams may experience reduced voice in upstream design discussions, particularly when decisions are resolved within defined cross-functional forums rather than open consensus processes. This trade-off is acceptable because Stripe operates in a high-stakes environment where delayed decisions can expose the organization to financial, legal, and reputational risk. In payment systems, the cost of slow coordination often exceeds the cost of limited

inclusion. Therefore, the design accepts selective participation as a necessary constraint to preserve execution speed and operational safety.

Moreover, Stripe's collaboration design preserves strong local autonomy for product, engineering, and regional teams while introducing shared interfaces and coordination. This balance enables innovation but does not fully prevent variation in how teams interpret priorities or implement collaboration norms. As a result, collaboration quality may remain uneven across teams, particularly during periods of rapid hiring or market expansion. This inconsistency reflects a deliberate choice: enforcing uniform collaboration behavior would undermine Stripe's culture of ownership and technical creativity. The design accepts localized variation as a cost of maintaining decentralized problem-solving at scale.

Eventually, our design relies heavily on transparency through shared documentation, decision records, and cross-team visibility into trade-offs. While the design reduces hidden misalignment, it increases the volume of information teams must process. In practice, some teams may disengage from shared artifacts or selectively attend to information, weakening the intended coordination benefits. This limitation is unavoidable in a large organization. Centralized filtering of information would reintroduce gatekeeping and delay. Instead, the design assumes teams will develop informal filtering norms over time. Occasional information overload is treated as an acceptable cost relative to the risks of opaque decision-making.

Conclusion:

This analysis demonstrates that collaboration at Stripe is not constrained by a lack of authority but by the complexity of coordinating speed, regulation, global scale, and interdependent product systems. Stripe's collaboration challenges emerge where incentives diverge, dependencies are obscured, and constraints are surfaced too late in the development process. The proposed collaboration design addresses these failures by shifting coordination upstream, clarifying ownership and interfaces, and embedding cross-functional and regional expertise into critical decision points.

Meanwhile, the design acknowledges inherent limits. It is difficult to eliminate risks without undermining the company's core strengths from trade-offs between speed and safety, autonomy and consistency, and transparency and attention. Rather than attempting to remove friction entirely, the design aims to make collaboration failures earlier, more visible, and less costly. By doing so, it reframes collaboration not as consensus or control, but as a deliberately engineered system that supports reliable performance in a high-stakes, rapidly evolving financial infrastructure environment.

References

Debnath, Riten. “Stripe in 2026: Usage, Revenue, Valuation & Growth Statistics.” *Fueler*, 18 Oct. 2025, fueler.io/blog/stripe-usage-revenue-valuation-growth-statistics. Accessed 17 Jan. 2026.

Edmondson, A. C. (2012). Teamwork on the fly: How to master the new art of teaming. Harvard Business Review. Retrieved from https://hbr.org/2012/04/teamwork-on-the-fly

Galbraith, J. R. (2014). Designing organizations: Strategy, structure, and process at the business unit and enterprise levels (3rd ed.). Wiley.

Galbraith, J. R. (n.d.). The Star Model™. Retrieved from https://www.studiodojo.com/wp-content/uploads/2021/12/Star-Model-Jay-Galbraith.pdf

Harvard Business School Working Knowledge. (2012). The importance of teaming. Retrieved from https://www.library.hbs.edu/working-knowledge/the-importance-of-teaming

Khan, Faryal. “Stripe Metrics: 16 Top Stripe KPIs to Track - AgencyAnalytics.” *AgencyAnalytics*, 2024, agencyanalytics.com/blog/stripe-metrics. Accessed 18 Jan. 2026.

Lee, Robert A. “Stripe Statistics 2025: Key Insights behind Its Trillion-Dollar Surge • SQ Magazine.” *SQ Magazine*, 23 Oct. 2025, sqmagazine.co.uk/stripe-statistics/.

Railsdrop. “The Evolution of Stripe’s Payment APIs: From Charges to Payment Intents.” *The Rails Drop* , 3 Nov. 2025, railsdrop.com/2025/11/03/the-evolution-of-stripes-payment-apis-from-charges-to-payment-intents/. Accessed 19 Jan. 2026.

Shabazz, Rasheed. “Why Stripe Stopped Processing Payments for Flipcause.” *Oakland Voices*, 10 Jan. 2026, oaklandvoices.us/2026/01/10/stripe-terminated-services-flipcause-bankruptcy-filings-reveal/. Accessed 19 Jan. 2026.

Strategy Breakdowns. “🎯 Move Fast and Break Things.” *Strategy Breakdowns*, 3 Dec. 2025, strategybreakdowns.com/p/move-fast-and-break-things. Accessed 19 Jan. 2026.

Stripe. “Embedded Finance for Software Platforms | Stripe.” *Stripe.com*, 2025, stripe.com/guides/introduction-to-embedded-finance.

---. “What’s Happening in Fintech—Banking-As-a-Service, Compliance, and New Revenue Models.” *Stripe.com*, 2023, stripe.com/sessions/2023/whats-happening-in-fintech. Accessed 19 Jan. 2026.