Backend Engineering Roadmap

2025-2026

Building a Scalable Foundation for Studio

Transform our backend from technical debt to competitive advantage

Koyo Isono / Backend Engineer

Studio, Inc.

Current Challenges & Risks

1. Technical Debt

- Stripe API version stuck at 2017
- 7+ years behind current version
- Missing modern payment features

2. Development Efficiency

- Flat directory structure
- Growing file count reducing visibility
- Interface and Implementation scattered

3. Security & Scalability

- No unified authentication system
- Ad-hoc API key authentication
- Difficult cross-service data sharing

Technical Strategy

1. Stripe API Modernization

7+ years of technical debt → Modern payment infrastructure

→ **Implement Adapter Pattern**

Isolate Stripe API calls in dedicated adapter classes for safe migration

→ **Centralize API Interactions**

Move from scattered calls across services/repositories to unified location

→ **Comprehensive Test Coverage**

Write tests for each adapter to ensure safe, gradual migration

2. Domain-Driven Structure

Flat file chaos → Organized, maintainable codebase

→ **Feature-Based Organization**

Group related code by domain instead of technical layers

→ **Co-locate Interface & Implementation**

Keep contracts and their implementations together for better navigation

→ **Improve Developer Experience**

Faster file discovery, clearer relationships, easier onboarding

3. OAuth2 Authentication Platform

Ad-hoc auth chaos → Unified, secure authentication

→ **Build Unified Auth Service**

Single source of truth for user authentication across all services

→ **Enable Secure Data Sharing**

Share Studio user data with new services without custom API keys

→ **Accelerate New Products**

Launch new services faster with ready-to-use authentication

Implementation Roadmap

2025 Second Half (Jul-Dec)

Foundation Phase - Build the groundwork

✓ Complete Directory Restructuring

Migrate from flat structure to domain-driven organization

Impact: Improved code navigation and team productivity

→ Stripe API Migration Progress

Implement Adapter pattern and start gradual migration

Target: 60% of Stripe calls migrated by year-end

→ Start OAuth2 Platform

Design architecture and begin core implementation

Impact: Setting stage for unified authentication

2026 First Half (Jan-Jun)

Delivery Phase - Ship critical infrastructure

✓ Complete Stripe API Migration

Finish migrating all Stripe calls to new API version

Result: Access to 9 years of payment innovations

✓ Complete OAuth2 Implementation

Finish building and deploy unified authentication service

Key features: SSO, token management, user federation

→ Begin Service Migration

Start migrating existing services to new auth platform

Priority: Low-risk internal tools first

2026 Second Half (Jul-Dec)

Scale Phase - Realize the full potential

✓ All Services on Unified Auth

Complete migration of all services to OAuth2 platform

Result: Single sign-on across entire ecosystem

✓ Launch New Services

Deploy new products leveraging shared authentication

Speed: 10x faster service launch with built-in auth

→ Measure & Optimize

Analyze metrics and optimize performance

Focus: Response time, reliability, developer satisfaction

Resources & Gaps

What We Have ✓

- Shared awareness & strong team trust
- Test culture & deep codebase knowledge
- Clear business needs with new service plans
- AI tools & Laravel/Lumen expertise

What We Need →

- Full Stripe API changes analysis (2017→2025)
- OAuth2 production experience & reliability design
- Migration strategy without productivity loss
- Optimal directory structure decision

Personal Motivation

Why this matters to me as an engineer

Technical Excellence

Building systems that are not just functional, but elegant and maintainable

Developer Empowerment

Creating an environment where engineers can focus on innovation, not fighting legacy code

Business Impact

Directly enabling faster product launches and better user experiences

Personal Growth

Mastering system design and large-scale refactoring challenges

"Transform technical debt into competitive advantage"

Building for the Future

From: Fragmented, legacy systems

To: Unified, modern, scalable platform

Enabling

- Faster product development cycles
- Better security posture
- Foundation for Studio's next phase