



Teste prático — Sistema de Pagamentos Multi-Provider – Back End

Objetivo

Avaliar capacidade do candidato de projetar uma **arquitetura** escalável e resiliente para integrar múltiplos provedores de pagamento (ex.: Stripe, PayPal, PagSeguro, MercadoPago), e ao final **entregar uma API simples de autenticação** e um esqueleto que demonstre o fluxo de pagamento (integração fake / mock).

Entregáveis esperados

1. **Documento de arquitetura** (máx. 2–3 páginas) contendo:
 - Diagrama de alto nível (components/services).
 - Fluxo de pagamento (do checkout até a confirmação e webhook).
 - Como tratar falhas / retries / idempotência / conciliação.
 - Estratégia para rotear transações entre providers (balance, custo, disponibilidade).
 - Segurança (armazenamento de credenciais, PCI scope minimizado, tokenização).
 - Observabilidade (logs, métricas, tracing) e testes.
2. **Repositório com código** contendo:
 - **API de autenticação** simples (registro/login/token + rota protegida). — *Implementação entregue abaixo em Laravel.*
 - **Esqueleto do serviço de pagamentos:**
 - Interface/abstração (ex.: `PaymentProviderInterface`).
 - 2 providers mock (ex.: `ProviderA`, `ProviderB`) que simulam respostas distintas.
 - Endpoint que cria uma tentativa de pagamento e escolhe provider (pode ser baseado em parâmetro).
 - Endpoint que simula webhook do provider (idempotência).
3. **README** com instruções para rodar localmente, endpoints contidos e como testar.