

Integração com meio de pagamento Stripe, Planos e Assinaturas Queen Pitch

- Objetivo: implementar cobrança por assinaturas (principalmente planos Go e Premium) e compra avulsa de criações (criações extras), usando Stripe com checkout transparente via backend, garantindo persistência no banco e validações nas rotas (verificar plano e criações restantes).
- Decisão técnica principal: checkout “transparente” (tokenização/integração via backend), usar webhooks do Stripe para sincronizar status de pagamento/assinaturas; não depender diretamente do checkout hospedado do Stripe para fluxo principal.
- Divisão de trabalho/etapas acordadas:
 - 1) Integração Stripe + webhooks (cadastro de produtos/planos se necessário, roteamento de eventos).
 - 2) Modelagem e persistência no banco (campos para mapear assinaturas, produtos Stripe, criações/quantitativos, status).
 - 3) Ajuste das rotas e regras de autorização para checar planos e criações antes de liberar features.

Casos de Teste Principais (user journeys):

1. **Compra Mensal e Renovação Automática de Assinatura (Caminho Feliz):**
 - **O que testar:** Usuário sem plano compra um plano mensal ou anual.
 - Queremos ver se a cobrança é feita corretamente e se, após **um mês**, ele é cobrado novamente de forma automática e continua tendo acesso premium sem precisar fazer nada.
Queremos ver se a cobrança é feita corretamente e se, após **um ano**, ele é cobrado novamente de forma automática e continua tendo acesso premium sem precisar fazer nada.
 - Stripe (ou backend via agendamento) tenta cobrar cartão na data de renovação
→ se aprovado, `next_billing_date` atualiza e acesso continua sem interrupção.
 - Notificar usuário por e-mail: 10 dias antes (aviso), 1 dia antes (lembrete), e confirmação no dia da renovação.
 - Lidar com diferenças de dias/mês (Stripe gerencia prazos; documentar regra exata com Stripe).

- **Passos:**
 - Usuário (novo ou sem plano) compra um plano mensal (ex: Go Mensal).
 - Verificar se o pagamento é processado com sucesso.
 - Verificar se o usuário ganha acesso aos recursos premium.
 - Simular a passagem de 1 mês.
 - Verificar se uma nova cobrança é feita automaticamente no cartão do usuário.
 - Verificar se o usuário **continua** com acesso premium.
- **Resultado Esperado:**
 - Cobrança bem-sucedida; acesso liberado; registro persistido.
 - O usuário é cobrado mensalmente e mantém o acesso premium sem interrupções enquanto o pagamento for bem-sucedido.
 - O usuário é cobrado anualmente e mantém o acesso premium sem interrupções enquanto o pagamento for bem-sucedido.
 - Usuário é notificado por e-mail: 10 dias antes (aviso), 1 dia antes (lembrete), e confirmação no dia da renovação.
 - Cartão tenta cobrar na data de renovação e consegue, assinatura continua ativa e sendo renovada automaticamente.

2. Falha na Renovação Mensal (cobrança recorrente):

- **O que testar:** Um usuário com plano mensal tem uma falha no pagamento da renovação (ex: cartão expirado, ou sem limite). Queremos garantir que o acesso seja bloqueado após as tentativas de cobrança falharem.
 - Primeira tentativa falha: registrar falha, notificar usuário (e-mail), tentar novas tentativas (definir política — sugerido: 2-3 tentativas automáticas com janelas configuráveis).
 - Se continuar falhando: bloquear acesso após período definido e atualizar status no DB (past_due / canceled). **Possibilidade de permitir tentativas manuais pelo usuário (botão “Tentar cobrar novamente”).**
- **Passos:**
 - Usuário tem um plano mensal ativo.
 - Configurar o método de pagamento para falhar na próxima cobrança.
 - Simular a passagem de 1 mês (para acionar a renovação).
 - Verificar se o sistema tenta cobrar e registra a falha.
 - Verificar se o usuário **perde** o acesso aos recursos premium (após o período de retentativa definido, se houver).
- **Resultado Esperado:** O usuário perde o acesso premium quando a renovação automática falha. O sistema deve refletir que ele agora é um **usuário "free"**.

3. Compra Anual e Renovação Automática de Assinatura:

- **O que testar:** Um usuário compra o plano anual (pagando o valor total adiantado). Queremos garantir que ele tenha acesso premium por exatamente um ano e que, após esse período, seja cobrado novamente de forma automática e continue tendo acesso premium sem precisar fazer nada.
 - Usuário compra plano anual → cobrança única do valor anual → acesso por ~365 dias (considerar bissexto) → renovação automática no ano seguinte se cartão salvo.
 - Notificar usuário por e-mail: 10 dias antes (aviso), 1 dia antes (lembrete), e confirmação no dia da renovação.
- **Passos:**
 - Usuário compra um plano anual (ex: Plus Anual).
 - Verificar se o pagamento único é processado com sucesso.
 - Verificar se o usuário ganha acesso aos recursos premium.
 - Simular a passagem de 1 ano.
 - Verificar nova cobrança feita após 1 ano (renovação automática)..
 - Verificar se o usuário continua com acesso aos recursos premium após renovação.
- **Resultado Esperado:** O usuário é cobrado anualmente e mantém o acesso premium sem interrupções enquanto o pagamento for bem-sucedido.
- Cartão tenta cobrar na data de renovação e consegue, assinatura continua ativa e sendo renovada automaticamente.

4. Cancelamento de Assinatura - Plano Mensal OU Anual pelo Usuário:

- **O que testar:** Um usuário com plano ativo decide cancelar a renovação automática através do perfil dele. Queremos garantir que ele possa fazer isso e que mantenha o acesso até o fim do período que já pagou.
 - Usuário cancela assinatura durante período ativo → manter acesso até o fim do período já pago (ex.: até final do mês/ano) e então revogar; atualizar status no DB (cancel_at_period_end).
 - **UX:** confirmar motivo do cancelamento (pop-up) e oferecer reativação.
- **Passos:**
 - Usuário com plano mensal ativo clica em "Cancelar Assinatura" no perfil.
 - Verificar se o sistema confirma o agendamento do cancelamento (ex: "Sua assinatura será cancelada em [data]").
 - Verificar se o usuário **mantém** o acesso premium imediatamente após cancelar.
 - Simular a passagem do tempo até a data final do período pago.

- Verificar se nenhuma nova cobrança é feita.
 - Verificar se o usuário **perde** o acesso premium *após* essa data.
- **Resultado Esperado:** O usuário consegue cancelar a renovação automática, mantém o acesso pelo tempo já pago (período de carência) e depois volta a ser "free".
- Após o fim do período que já pagou, usuário perde acesso aos recursos premium.

5. Reativação de Assinatura - Plano Mensal/Anual Cancelado:

- **O que testar:** Um usuário cancelou a renovação do plano, mas muda de ideia antes do período que já pagou acabar. Queremos ver se ele consegue "desfazer" o cancelamento e continuar sendo cobrado normalmente nos meses seguintes.
 - Usuário cancelou mas deseja reativar antes do fim do período → permitir reativação via front (botão "Reativar assinatura") que aciona rota para reativar no backend/Stripe.
 - **UX:** botão Reativar assinatura só fica visível para usuários que estão com status no DB (cancel_at_period_end).
- **Passos:**
 - Usuário realizou os passos do teste 4 (cancelou, mas ainda tem acesso).
 - Usuário clica em "Reativar Assinatura" (ou botão similar) no perfil.
 - Verificar se o sistema confirma a reativação.
 - Simular a passagem do tempo até a data original de renovação/cancelamento.
 - Verificar se a cobrança de renovação é feita **normalmente**.
 - Verificar se o usuário **mantém** o acesso premium.
- **Resultado Esperado:** O usuário consegue reverter o cancelamento agendado e a assinatura continua ativa e sendo renovada automaticamente.

6. Compra avulsa de criações por usuário MASTER ADMIN (produtos únicos)

O que testar:

- Usuário compra N criações (produto não-recorrente) → backend processa compra via Stripe → creditar criações extras à organização com validade (sugerido: 30 dias por aquisição) → **decremento conforme uso**; registro separado de contagem de criações da assinatura.

- Observações: criações extras podem expirar (sugestão: validade de 30 dias); definir regras sobre acumulação entre criações de assinatura e extras.

Passos:

- Usuário

Resultado Esperado:

Usuário compra criações extras e vê o saldo de criações aumentar, somando a quantidade que ele adquiriu.

Usuário é notificado que essas criações têm prazo de 30 dias para serem utilizadas e após isso expiram.

As criações que usuário comprou ficam disponíveis para toda organização.

7. Troca/upgrade de plano: mensal para anual mesmo plano

O que testar:

Plano Go

- Usuário altera plano mensal GO para anual GO no meio do ciclo: oferecemos a opção de descontar o valor que ele já pagou para o plano mensal (ex. R\$285) do total do plano anual (ex. R\$2850); sendo debitado R\$ 2.565. A data de assinatura do plano anual deve contar a partir da data do pagamento, e deve ser renovada automaticamente no próximo ano.

Plano Plus

- Usuário altera plano mensal Plus para anual Plus no meio do ciclo: oferecemos a opção de descontar o valor que ele já pagou para o plano mensal (R\$485) do total do plano anual (R\$4850); sendo debitado R\$4365 da diferença. A data de assinatura do plano anual deve contar a partir da data do pagamento, e deve ser renovada automaticamente no próximo ano.

Passos:

- Usuário

Resultado Esperado: Usuário consegue alterar o plano com sucesso e é notificado que agora faz parte do plano anual xx, com a data de vencimento no dia xxxx e que será renovado automaticamente após 1 ano.

8. Troca de plano Anual para Mensal

O que testar:

- Usuário do plano anual **não tem a opção de voltar para o mensal** antes da data de vencimento do plano anual. Só poderá mudar de plano quando o ciclo anual finalizar.

Passos:

- Usuário

Resultado Esperado: Usuário **não** visualiza opções de downgrade de plano anual para mensal.

9. Troca/upgrade de plano Anual Go para Anual Plus

O que testar:

- Usuário do plano anual Go quer fazer upgrade para o plano anual Plus. Devemos calcular a diferença do que ele já pagou (R\$2850) perante o valor total do anual (R\$4850), a data final do plano não será alterada, **permanecendo a data anual do primeiro plano contratado**, apenas subtraindo a diferença de valor.

Passos:

- Usuário

Resultado Esperado: Usuário visualiza opções de upgrade de plano anual Go para Plus, e realiza a alteração, e é notificado que fez um upgrade de plano e agora faz parte do plano anual Plus, sendo que a data de vencimento permaneceu a mesma no dia xxxx e que será renovado automaticamente após 1 ano.

10. Troca/downgrade de plano Anual Plus para Anual Go

O que testar:

- Usuário do plano anual Plus, por já ter pago um valor superior nesse plano, **não tem a opção de voltar para o plano Go de valor inferior** antes da data de vencimento do plano atual dele (plus). Só poderá mudar de plano quando o ciclo anual finalizar.

Passos:

- Usuário

Resultado Esperado: Usuário não visualiza opções de downgrade de plano anual Plus para Anual Go.

Cancelar pagamento automático

Casos de Borda e Decisões pendentes:

6. Impedir Compra Dupla:

- **O que testar:** Um usuário que já possui *qualquer* plano ativo (mensal ou anual) tenta comprar um plano novamente.
- **Passos:**
 - Usuário com plano ativo acessa a página de planos.
 - Tentar iniciar a compra de qualquer plano.
- **Resultado Esperado:** O sistema deve impedir a compra, idealmente mostrando uma mensagem indicando que ele já possui um plano ativo (e talvez um link para gerenciar a assinatura existente).

7. - **Política de retries na falha de cobrança** (quantas tentativas? intervalos?).

8. - **Regra exata de contabilização do período de renovação** (Stripe trata, mas confirmar para documentação e testes).

9. - **Validade das criações extras (sugerido: 30 dias)**. Confirmar se criações extras compradas durante assinatura permanecem se a assinatura for cancelada/expirar.

10. - **Política de upgrade/downgrade intra-ciclo** (crédito proporcional imediato vs. começar no próximo ciclo).

11. - **Se produtos precisam estar criados previamente no Stripe ou serão enviados como compra pontual sem cadastro** (conforme conversa, não é obrigatório cadastrar produtos na Stripe para vendas pontuais; backend envia valores e Stripe processa).

12. Tipos de pagamento aceitos: crédito, débito e??

13. Contagem de projetos por plano

Regras de autorização/controle de acesso

- Antes de liberar recurso pago ou permitir criações: checar (a) se usuário pertence a organização paga OU (b) se usuário tem criações extras > 0 OU (c) se plano do usuário/organização permite o acesso.

- Tipos de permissões: proprietário (único que pode excluir projeto), administrador e demais roles (documentar). Proprietário imutável por conta.