

Guia de Deploy - CRSET Finance Backend

1. Deploy no Railway (Recomendado)

Preparação do Repositório

```
# 1. Criar repositório no GitHub
git init
git add .
git commit -m "Initial commit - CRSET Finance Backend"
git remote add origin https://github.com/seu-usuario/crset-finance-backend.git
git push -u origin main
```

Deploy no Railway

1. Acesse railway.app (https://railway.app)
2. Conecte sua conta GitHub
3. Clique em “New Project” → “Deploy from GitHub repo”
4. Selecione o repositório `crset-finance-backend`
5. Railway detectará automaticamente o Dockerfile

Configuração de Variáveis

No Railway Dashboard → Settings → Variables:

```
STRIPE_SECRET_KEY=sk_live_51RbH0XFVbYwjbgXtLKG9hy9meibU27laik66jsGfhUDU6udH61hgNsfiB8hIoP2EoxhVw50eir
STRIPE_WEBHOOK_SECRET=whsec_seu_webhook_secret
FRONTEND_URL=https://crsetsolutions.com
PORT=8000
JWT_SECRET=FdBvJRJ2U+jS8q7YVxJiY2TtYdY1nnep29b+1NUBLETUFiLPNHIdBHo5/46C0g03qHYNK2Fb1y+B
allqkkAvrg==
```

Gerar Domínio Público

1. Vá para Settings → Networking
2. Clique em “Generate Domain”
3. Anote a URL gerada (ex: `crset-backend-production-abc123.up.railway.app`)

2. Configuração do Stripe Dashboard

Webhook Configuration

1. Acesse [Stripe Dashboard](https://dashboard.stripe.com) (https://dashboard.stripe.com)
2. Vá para Developers → Webhooks
3. Clique em “Add endpoint”
4. URL: `https://sua-url-railway.up.railway.app/webhook`
5. Eventos para escutar:
 - `checkout.session.completed`
 - `payment_intent.succeeded`
 - `payment_intent.payment_failed`

6. Copie o “Signing secret” e adicione como `STRIPE_WEBHOOK_SECRET`

Produtos e Preços

1. Vá para Products no Stripe Dashboard
2. Crie produtos para seus serviços:
 - Consultoria Financeira CRSET
 - Análise de Investimentos
 - etc.

3. Integração Frontend

Exemplo de Código JavaScript

```
async function createCheckout(productName, priceAmount) {
  const response = await fetch('https://sua-url-backend/create-checkout-session', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
    },
    body: JSON.stringify({
      product_name: productName,
      price_amount: priceAmount,
      currency: 'brl',
      quantity: 1
    })
  });

  const data = await response.json();

  if (response.ok) {
    // Redirecionar para checkout
    window.location.href = data.checkout_url;
  } else {
    console.error('Erro:', data.detail);
  }
}
```

Atualizar crsetsolutions.com

No seu frontend atual, adicione:

```
// Configuração da API
const API_BASE = 'https://sua-url-railway.up.railway.app';

// Função para criar checkout
function iniciarPagamento(servico, valor) {
  createCheckout(servico, valor * 100); // Converter para centavos
}
```

4. Testes de Produção

Verificar Endpoints

```
curl https://sua-url-railway.up.railway.app/health
curl https://sua-url-railway.up.railway.app/products
```

Testar Checkout

```
curl -X POST https://sua-url-railway.up.railway.app/create-checkout-session \
-H "Content-Type: application/json" \
-d '{
  "product_name": "Teste",
  "price_amount": 1000,
  "currency": "brl"
}'
```

5. Monitoramento

Logs do Railway

- Acesse Deployments → View Logs
- Monitore erros e performance

Stripe Dashboard

- Monitore pagamentos em Payments
- Verifique webhooks em Developers → Webhooks

6. Segurança

Checklist de Segurança

- [] HTTPS habilitado (automático no Railway)
- [] Webhook secret configurado
- [] Variáveis de ambiente protegidas
- [] CORS configurado corretamente
- [] Logs não expõem dados sensíveis

Backup

- Código versionado no GitHub
- Configurações documentadas
- Chaves de backup seguras

7. Próximos Passos

1. **Deploy no Railway** com o código fornecido
2. **Configurar webhook** no Stripe Dashboard
3. **Testar integração** completa
4. **Atualizar frontend** em crsetsolutions.com
5. **Monitorar** primeiros pagamentos

6. **Implementar** funcionalidades adicionais conforme necessário

Suporte

Para dúvidas ou problemas:

1. Verificar logs no Railway Dashboard
2. Consultar documentação do Stripe
3. Testar endpoints com curl/Postman
4. Verificar configuração de variáveis de ambiente