

# RESUMO COMPLETO - Deploy Vercel Preparado

## ✅ STATUS: PROJETO PRONTO PARA DEPLOY

O projeto **ADK Arbitrage Profit Guard** foi completamente preparado para deploy no Vercel. Todas as configurações foram aplicadas e testadas.

## 3 OPÇÕES DE DEPLOY

### OPÇÃO 1: Script Automatizado (Mais Fácil) ★

```
cd /home/ubuntu/github_repos/ADKArbitrageProfitGuard/nextjs-frontend
./deploy-vercel.sh
```

### OPÇÃO 2: Comandos Manuais

```
cd /home/ubuntu/github_repos/ADKArbitrageProfitGuard/nextjs-frontend
npx vercel login
npx vercel --prod --yes
```

### OPÇÃO 3: Dashboard Vercel (Interface Web)

1. Acesse: <https://vercel.com/dashboard>
2. New Project → Import do GitHub
3. Repositório: `adkbot/ADKArbitrageProfitGuard`
4. Root Directory: `nextjs-frontend`

## CONFIGURAÇÕES APLICADAS

### ✅ Arquivos Criados/Configurados:

- `.env.local` - Variáveis de ambiente para produção
- `vercel.json` - Configuração de deploy e rewrites
- `next.config.js` - Configuração Next.js otimizada
- `deploy-vercel.sh` - Script automatizado de deploy

### ✅ Variáveis de Ambiente:

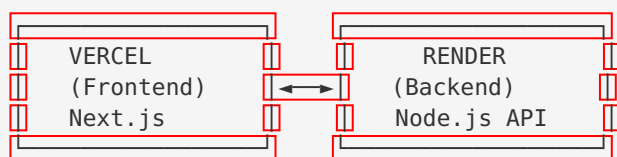
```
NEXT_PUBLIC_API_BASE_URL=https://adkarbitrageprofitguard.onrender.com
NEXT_PUBLIC_APP_NAME=ADK Arbitrage Profit Guard
NEXT_PUBLIC_APP_VERSION=2.1.0
```

### ✅ Configurações Vercel:

- Rewrites para API do Render configurados
- Headers de segurança aplicados

- Framework Next.js detectado automaticamente
- Build testado e funcionando

## ARQUITETURA FINAL



- **Frontend:** Vercel (Next.js) - Interface do usuário
- **Backend:** Render (Node.js) - API e lógica de negócio
- **Comunicação:** HTTPS com rewrites configurados

## CHECKLIST PRÉ-DEPLOY

- [x] Projeto Next.js configurado
- [x] Build testado com sucesso
- [x] Variáveis de ambiente configuradas
- [x] Vercel CLI instalado
- [x] Configuração vercel.json criada
- [x] Script de deploy criado
- [x] Repositório Git configurado
- [x] Backend no Render funcionando

## EXECUTAR DEPLOY AGORA

Comando mais simples:

```
cd /home/ubuntu/github_repos/ADKArbitrageProfitGuard/nextjs-frontend && ./deploy-vercel.sh
```

## PÓS-DEPLOY

Após o deploy, você receberá:

1. **URL do Vercel** (ex: <https://seu-projeto.vercel.app>)
2. **Logs de deploy** com status
3. **Dashboard de monitoramento** no Vercel

## TROUBLESHOOTING

Se der erro de login:

```
npx vercel login
```

### Se der erro de build:

```
npm run build
```

### Se der erro de CORS:

- Verificar se backend no Render aceita domínio do Vercel
- Configuração já aplicada no vercel.json



## PRONTO PARA DEPLOY!

Tudo configurado e testado. Execute o script ou use os comandos manuais para fazer o deploy.

Tempo estimado de deploy: 2-5 minutos