



# Guia de Deploy na Vercel - Gabi Cleaning

Este guia irá ajudá-lo a fazer o deploy do site da Gabi Cleaning na Vercel.



## Pré-requisitos

Antes de começar, você precisará:

1. Uma conta na [Vercel](https://vercel.com) (<https://vercel.com>) (pode fazer login com sua conta GitHub)
2. O repositório GitHub já configurado: <https://github.com/Beny74/Gabicleaning>
3. Um banco de dados PostgreSQL (recomendamos [Vercel Postgres](https://vercel.com/docs/storage/vercel-postgres) (<https://vercel.com/docs/storage/vercel-postgres>) ou [Supabase](https://supabase.com) (<https://supabase.com>))
4. Uma conta Gmail com senha de aplicativo configurada (para envio de e-mails)



## Passo a Passo

### 1. Criar Conta na Vercel

- Acesse [vercel.com](https://vercel.com) (<https://vercel.com>)
- Clique em “Sign Up” ou “Login”
- Faça login com sua conta GitHub (recomendado)

### 2. Importar o Projeto

1. No dashboard da Vercel, clique em “Add New...” → “Project”
2. Selecione “Import Git Repository”
3. Escolha o repositório **Beny74/Gabicleaning**
4. Clique em “Import”

### 3. Configurar o Projeto

Na tela de configuração:

#### Framework Preset

- A Vercel detectará automaticamente que é um projeto **Next.js**
- Mantenha as configurações padrão

#### Root Directory

- Defina como: `nextjs_space`
- Isso é **MUITO IMPORTANTE** pois o código está dentro desta pasta

#### Build Settings

- **Build Command:** `yarn build` (ou deixe padrão)
- **Output Directory:** `.next` (padrão)
- **Install Command:** `yarn install` (padrão)

### 4. Configurar Variáveis de Ambiente

Clique em “**Environment Variables**” e adicione as seguintes variáveis:

## Banco de Dados

DATABASE\_URL

**Valor:** Sua string de conexão PostgreSQL

**Exemplo:** postgresql://user:password@host:5432/database?connect\_timeout=15

**Dica:** Se não tiver um banco de dados ainda, você pode criar um gratuitamente:

- [Vercel Postgres](https://vercel.com/docs/storage/vercel-postgres) (<https://vercel.com/docs/storage/vercel-postgres>) (integrado)
- [Supabase](https://supabase.com) (<https://supabase.com>) (grátis até 500MB)
- [Neon](https://neon.tech) (<https://neon.tech>) (grátis até 10GB)

## Configuração de E-mail

EMAIL\_USER

**Valor:** gabifcleaning@gmail.com

EMAIL\_PASSWORD

**Valor:** Sua senha de aplicativo do Gmail

**Como criar uma senha de aplicativo no Gmail:**

1. Acesse [myaccount.google.com/security](https://myaccount.google.com/security) (<https://myaccount.google.com/security>)
2. Ative a **verificação em duas etapas** (se ainda não estiver ativa)
3. Procure por “**Senhas de app**” na busca
4. Selecione “**E-mail**” e “**Outro (nome personalizado)**”
5. Digite “Gabi Cleaning Website” como nome
6. Clique em “**Gerar**”
7. Copie a senha de 16 caracteres gerada
8. Use essa senha na variável `EMAIL_PASSWORD`

## 5. Deploy

1. Após configurar as variáveis de ambiente, clique em “**Deploy**”
2. Aguarde o processo de build (geralmente leva 2-5 minutos)
3. Quando concluído, você verá uma mensagem de sucesso com o link do site

## 6. Configurar Domínio Personalizado (Opcional)

Se você quiser usar o domínio **gabifcleaner.com**:

1. No dashboard do projeto na Vercel, vá em “**Settings**” → “**Domains**”
2. Clique em “**Add Domain**”
3. Digite `gabifcleaner.com` e clique em “**Add**”
4. A Vercel fornecerá as configurações DNS necessárias
5. Acesse o painel do seu provedor de domínio (onde comprou o domínio)
6. Adicione os registros DNS conforme instruído pela Vercel:
  - Tipo `A` ou `CNAME` apontando para os servidores da Vercel
7. Aguarde a propagação DNS (pode levar até 48 horas, mas geralmente é rápido)

## 7. Configurar o Banco de Dados

Após o primeiro deploy, você precisará executar as migrations do Prisma:

### Opção 1: Usando Vercel CLI (Recomendado)

```
# Instalar Vercel CLI
npm i -g vercel

# Fazer login
vercel login

# Ir para a pasta do projeto
cd nextjs_space

# Executar comando no ambiente de produção
vercel env pull .env.production
prisma migrate deploy
```

### Opção 2: Manualmente via Prisma Studio ou SQL

1. Acesse seu banco de dados
2. Execute o schema SQL do arquivo `prisma/schema.prisma`



### Pronto!

Seu site agora está no ar! A Vercel fornecerá:

- URL de produção (ex: `gabicleaning.vercel.app`)
- URLs de preview para cada commit
- HTTPS automático
- CDN global
- Deploy automático a cada push no GitHub



### Atualizações Automáticas

A partir de agora, **toda vez que você fizer um push** para o repositório GitHub:

1. A Vercel detectará automaticamente as mudanças
2. Fará o build do projeto
3. Atualizará o site automaticamente

Você não precisa fazer nada manualmente! 🎉



### Links Úteis

- **Dashboard Vercel:** [vercel.com/dashboard](https://vercel.com/dashboard) (<https://vercel.com/dashboard>)
- **Documentação Vercel:** [vercel.com/docs](https://vercel.com/docs) (<https://vercel.com/docs>)
- **Status do Deploy:** Você receberá notificações por e-mail
- **Logs:** Disponíveis no dashboard da Vercel

## SOS Problemas Comuns

---

### Build falha com erro de Prisma

**Solução:** Verifique se a variável `DATABASE_URL` está configurada corretamente

### E-mails não estão sendo enviados

**Solução:**

1. Verifique se `EMAIL_USER` e `EMAIL_PASSWORD` estão configurados
2. Confirme que você está usando uma senha de aplicativo, não sua senha normal
3. Verifique se a autenticação em duas etapas está ativa no Gmail

### Erro 404 ao acessar o site

**Solução:** Certifique-se de que o **Root Directory** está configurado como `nextjs_space`

## Suporte

---

Se encontrar algum problema, você pode:

1. Verificar os logs de build no dashboard da Vercel
2. Consultar a documentação da Vercel
3. Entrar em contato com o suporte da Vercel

---

Desenvolvido com ❤️ para Gabi Cleaning