

Guia Completo: Subir Projeto para o GitHub

Pré-requisitos

- Conta no GitHub (crie em: <https://github.com/signup>)
- Git instalado no seu computador
- Acesso ao terminal/linha de comando

Método 1: Usar o Script Automático (MAIS FÁCIL)

Passo 1: Criar Repositório no GitHub

1. Acesse: <https://github.com>
2. Faça login
3. Clique no botão "+" (canto superior direito) → “**New repository**”
4. Configure:
 - **Nome:** sistema-salao-beleza
 - **Descrição:** “Sistema de gestão para salões de beleza”
 - **Visibilidade:** Private (recomendado)
 - **NÃO marque** nenhuma opção adicional
5. Clique em “**Create repository**”
6. **COPIE** a URL que aparece (exemplo: <https://github.com/seu-usuario/sistema-salao-beleza.git>)

Passo 2: Executar o Script

```
cd /home/ubuntu/sistema_salao_beleza
./upload-github.sh
```

Passo 3: Seguir as Instruções

O script vai pedir:

1. A URL do repositório (cole a que você copiou)
2. Suas credenciais do GitHub (usuário e senha ou token)

PRONTO! Seu código está no GitHub! 🎉

Método 2: Manual (Passo a Passo)

1. Criar Repositório no GitHub

(mesmos passos do Método 1)

2. Conectar o Repositório Local ao GitHub

```
cd /home/ubuntu/sistema_salao_beleza

# Adicionar o remote (substitua pela SUA URL)
git remote add origin https://github.com/SEU-USUARIO/sistema-salao-beleza.git

# Verificar se conectou
git remote -v
```

3. Enviar o Código

```
# Enviar todos os commits para o GitHub
git push -u origin master
```

4. Autenticar

Quando pedir credenciais:

- **Username:** seu usuário do GitHub
- **Password:**
- **NÃO** é sua senha normal!
- Use um **Personal Access Token** (veja próxima seção)



Como Criar Personal Access Token

O GitHub não aceita mais senha normal. Você precisa de um token:

Passos:

1. Acesse: <https://github.com/settings/tokens>
2. Clique em “**Generate new token**” → “**Generate new token (classic)**”
3. Configure:
 - **Note:** “Sistema Beleza - Deploy”
 - **Expiration:** 90 days (ou No expiration)
 - **Selecione:** repo (marque todas as opções de repo)
4. Clique em “**Generate token**”
5. **CÓPIE O TOKEN** (só aparece uma vez!)
6. Use este token como senha quando fizer o push

Exemplo:

```
Username: seu-usuario
Password: ghp_xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx (seu token)
```



Atualizações Futuras

Depois que o repositório estiver no GitHub, para enviar novas alterações:

```
cd /home/ubuntu/sistema_salao_beleza

# Adicionar arquivos modificados
git add .

# Fazer commit com mensagem
git commit -m "Descrição das mudanças"

# Enviar para o GitHub
git push
```

Verificar se Funcionou

1. Acesse seu repositório no GitHub
2. Você deve ver todos os arquivos do projeto
3. Verifique se aparece:
 -  README.md
 -  nextjs_space/
 -  scripts/
 -  Documentações (.md)
 -  .env (NÃO deve aparecer - arquivo protegido)
 -  node_modules (NÃO deve aparecer - ignorado)

Problemas Comuns

Erro: “failed to push some refs”

Solução:

```
git pull origin master --rebase
git push -u origin master
```

Erro: “Permission denied”

Solução: Verifique se o token tem permissões de “repo”

Erro: “fatal: remote origin already exists”

Solução:

```
git remote remove origin
git remote add origin https://github.com/SEU-USUARIO/sistema-salao-beleza.git
```

Erro: “Support for password authentication was removed”

Solução: Use um Personal Access Token (veja seção acima)

Resumo Rápido

```
# 1. Criar repositório no GitHub (pelo site)

# 2. Conectar e enviar (pelo terminal)
cd /home/ubuntu/sistema_salao_beleza
git remote add origin https://github.com/SEU-USUARIO/sistema-salao-beleza.git
git push -u origin master

# 3. Quando pedir senha, use o Personal Access Token
```

Próximos Passos

Depois de subir para o GitHub:

1. **Proteger a Branch Master:**

- Settings → Branches → Add rule
- Branch name: master
- Marque: “Require pull request reviews”

2. **Adicionar Colaboradores:**

- Settings → Collaborators → Add people

3. **Criar README Público:**

- Adicione prints do sistema
- Instruções de instalação

4. **Configurar GitHub Actions:**

- Para testes automáticos
- Deploy automático

Parabéns!

Seu projeto está agora versionado e protegido no GitHub!

Benefícios:

-  Histórico completo de mudanças
-  Backup seguro na nuvem
-  Colaboração facilitada
-  Controle de versões profissional

Acesse seu repositório:

<https://github.com/SEU-USUARIO/sistema-salao-beleza>

 **Dúvidas?** Execute o script automático: `./upload-github.sh`