**📚 Manual Completo de Implantação**

**Sistema de Agendamento Online**

**🎯 Objetivo:** Implantar um sistema completo de agendamento online com painel administrativo, integração Google Calendar e backup automático.

**📋 Pré-requisitos**

Antes de começar, você precisará criar contas nas seguintes plataformas:

* ✅ **Conta Google** (Gmail)
* ✅ **Conta GitHub**
* ✅ **Conta Streamlit Cloud** (gratuita)

**⏱️ Tempo estimado:** 45-60 minutos  
**💰 Custo:** Gratuito (todas as ferramentas usadas são gratuitas)

**🏗️ Etapa 1: Configuração do GitHub**

**1.1 Criar Repositório**

1. **Acesse:** https://github.com
2. **Faça login** na sua conta
3. **Clique em "New repository"** (botão verde no canto superior direito)
4. **Configure:**
   * **Repository name:** sistema-agendamento
   * **Description:** Sistema de Agendamento Online
   * **Visibilidade:** ✅ **Public** (deixe público)
   * **Initialize:** ✅ Marque "Add a README file"
5. **Clique em "Create repository"**

**1.2 Fazer Upload dos Arquivos**

1. **No seu repositório criado**, clique em "Add file" → "Upload files"
2. **Arraste o arquivo app.py** para a área de upload
3. **Crie o arquivo requirements.txt:**
   * Clique em "Add file" → "Create new file"
   * **Nome:** requirements.txt
   * **Conteúdo:**
4. streamlit>=1.28.0
5. pandas>=1.5.0
6. google-auth>=2.17.0
7. google-auth-oauthlib>=1.0.0
8. google-auth-httplib2>=0.1.0
9. google-api-python-client>=2.88.0
10. requests>=2.28.0
11. **Commit as mudanças:**
    * **Commit message:** Upload inicial do sistema
    * Clique em "Commit changes"

**1.3 Obter Token do GitHub**

1. **Clique na sua foto** (canto superior direito) → "Settings"
2. **No menu esquerdo:** "Developer settings" (último item)
3. **Clique em:** "Personal access tokens" → "Tokens (classic)"
4. **Clique em:** "Generate new token" → "Generate new token (classic)"
5. **Configure:**
   * **Note:** Sistema Agendamento
   * **Expiration:** No expiration
   * **Scopes:** ✅ Marque apenas repo (acesso completo aos repositórios)
6. **Clique em "Generate token"**
7. **⚠️ IMPORTANTE:** Copie o token e guarde em local seguro (você só verá uma vez!)

**🔐 Etapa 2: Configuração do Google Cloud**

**2.1 Criar Projeto no Google Cloud**

1. **Acesse:** https://console.cloud.google.com
2. **Faça login** com sua conta Google
3. **Clique em "Select a project"** (canto superior esquerdo)
4. **Clique em "NEW PROJECT"**
5. **Configure:**
   * **Project name:** Sistema Agendamento
   * **Organization:** Deixe como está
6. **Clique em "CREATE"**
7. **Aguarde criação** e selecione o projeto criado

**2.2 Habilitar APIs Necessárias**

1. **No menu esquerdo:** "APIs & Services" → "Library"
2. **Busque e habilite as seguintes APIs:**

**📅 Google Calendar API:**

* + Digite "Calendar API" na busca
  + Clique em "Google Calendar API"
  + Clique em "ENABLE"

**📧 Gmail API (opcional - para emails):**

* + Digite "Gmail API" na busca
  + Clique em "Gmail API"
  + Clique em "ENABLE"

**2.3 Configurar Tela de Consentimento OAuth**

1. **Vá para:** "APIs & Services" → "OAuth consent screen"
2. **Selecione:** ⚪ External
3. **Clique em "CREATE"**
4. **Preencha os campos obrigatórios:**
   * **App name:** Sistema de Agendamento
   * **User support email:** seu-email@gmail.com
   * **Developer contact information:** seu-email@gmail.com
5. **Clique em "SAVE AND CONTINUE"**
6. **Na tela "Scopes":** Clique em "SAVE AND CONTINUE" (sem adicionar nada)
7. **Na tela "Test users":**
   * Clique em "ADD USERS"
   * Adicione seu próprio email
   * Clique em "SAVE AND CONTINUE"
8. **Revise e clique em "BACK TO DASHBOARD"**

**2.4 Criar Credenciais OAuth**

1. **Vá para:** "APIs & Services" → "Credentials"
2. **Clique em "+ CREATE CREDENTIALS"** → "OAuth client ID"
3. **Configure:**
   * **Application type:** Web application
   * **Name:** Sistema Agendamento Web
   * **Authorized redirect URIs:**
     + Clique em "ADD URI"
     + Adicione: http://localhost:8080/callback
     + Clique em "ADD URI" novamente
     + Adicione: https://developers.google.com/oauthplayground
4. **Clique em "CREATE"**
5. **⚠️ IMPORTANTE:** Anote o **Client ID** e **Client Secret**

**2.5 Obter Refresh Token**

1. **Acesse:** https://developers.google.com/oauthplayground
2. **No canto direito superior:** Clique na engrenagem ⚙️
3. **Marque:** ✅ "Use your own OAuth credentials"
4. **Preencha:**
   * **OAuth Client ID:** (cole o Client ID da etapa anterior)
   * **OAuth Client secret:** (cole o Client Secret da etapa anterior)
5. **Feche a configuração**
6. **No lado esquerdo:**
   * Na caixa de busca, digite: https://www.googleapis.com/auth/calendar
   * Clique no resultado
   * Clique em "Authorize APIs"
7. **Faça login** com sua conta Google
8. **Aceite as permissões**
9. **Clique em "Exchange authorization code for tokens"**
10. **⚠️ IMPORTANTE:** Copie o **Refresh token** que aparecer

**☁️ Etapa 3: Configuração do Streamlit Cloud**

**3.1 Criar Conta no Streamlit Cloud**

1. **Acesse:** https://share.streamlit.io
2. **Clique em "Sign up"**
3. **Faça login com GitHub** (recomendado)
4. **Autorize o Streamlit** a acessar sua conta GitHub

**3.2 Fazer Deploy da Aplicação**

1. **No Streamlit Cloud**, clique em "New app"
2. **Configure:**
   * **Repository:** seu-usuario/sistema-agendamento
   * **Branch:** main
   * **Main file path:** app.py
   * **App URL:** sistema-agendamento-[seu-nome] (será gerado automaticamente)
3. **Clique em "Deploy!"**
4. **⚠️ A aplicação irá falhar inicialmente** - isso é normal!

**3.3 Configurar Secrets**

1. **Na página da sua app**, clique em "⚙️ Settings" (canto inferior direito)
2. **Clique em "Secrets"**
3. **Cole o seguinte conteúdo** (substitua os valores pelos seus):

# Configurações do Sistema

ADMIN\_PASSWORD = "sua\_senha\_admin\_aqui"

ADMIN\_URL\_KEY = "chave\_secreta\_admin"

# GitHub Backup

GITHUB\_TOKEN = "seu\_token\_github\_aqui"

GITHUB\_REPO = "seu-usuario/sistema-agendamento"

GITHUB\_BRANCH = "main"

CONFIG\_FILE = "configuracoes.json"

# Google Calendar

GOOGLE\_CLIENT\_ID = "seu\_client\_id\_aqui.apps.googleusercontent.com"

GOOGLE\_CLIENT\_SECRET = "seu\_client\_secret\_aqui"

GOOGLE\_REFRESH\_TOKEN = "seu\_refresh\_token\_aqui"

GOOGLE\_CALENDAR\_ID = "primary"

1. **Substitua os valores:**
   * **sua\_senha\_admin\_aqui**: Crie uma senha forte para o painel admin
   * **chave\_secreta\_admin**: Crie uma chave secreta (ex: admin123456)
   * **seu\_token\_github\_aqui**: Cole o token do GitHub (Etapa 1.3)
   * **seu-usuario**: Seu nome de usuário do GitHub
   * **seu\_client\_id\_aqui**: Client ID do Google (Etapa 2.4)
   * **seu\_client\_secret\_aqui**: Client Secret do Google (Etapa 2.4)
   * **seu\_refresh\_token\_aqui**: Refresh Token do Google (Etapa 2.5)
2. **Clique em "Save"**

**3.4 Reiniciar a Aplicação**

1. **Clique em "Reboot app"** (botão no canto inferior direito)
2. **Aguarde alguns minutos** para o deploy completar
3. **Sua aplicação estará funcionando!**

**🔧 Etapa 4: Configuração Inicial do Sistema**

**4.1 Acessar o Painel Administrativo**

1. **Acesse sua aplicação:** https://seu-app.streamlit.app
2. **Para acessar o admin, adicione no final da URL:**
3. ?admin=chave\_secreta\_admin

**Exemplo:** https://seu-app.streamlit.app?admin=admin123456

1. **Digite a senha** que você configurou nos secrets
2. **Você estará no painel administrativo!**

**4.2 Configurar Informações Básicas**

1. **Vá para "⚙️ Configurações Gerais"**
2. **Na aba "📞 Contato & Local"**, preencha:
   * Nome do profissional
   * Especialidade
   * Nome da clínica
   * Telefone e WhatsApp
   * Endereço completo
3. **Na aba "📅 Agendamento"**, configure:
   * Horários de funcionamento
   * Duração das consultas
   * Antecedência mínima
   * Dias disponíveis no futuro
4. **Clique em "💾 Salvar Todas as Configurações"**

**4.3 Configurar Dias de Funcionamento**

1. **Vá para "📅 Configurar Agenda"**
2. **Selecione os dias** da semana que você atende
3. **Clique em "💾 Salvar Dias"**

**4.4 Testar o Sistema**

1. **Abra uma aba anônima** no navegador
2. **Acesse sua aplicação** (sem o ?admin=)
3. **Teste fazer um agendamento**
4. **Volte ao painel admin** e verifique se o agendamento apareceu

**📧 Etapa 5: Configuração de Email (Opcional)**

**5.1 Configurar Gmail para Envio Automático**

1. **No painel admin**, vá para "⚙️ Configurações Gerais" → aba "📧 Email"
2. **Marque:** ✅ "Ativar envio automático de emails"
3. **Configure:**
   * **Email do sistema:** seu-email@gmail.com
   * **Servidor SMTP:** smtp.gmail.com
   * **Porta SMTP:** 587

**5.2 Criar Senha de App no Gmail**

1. **Acesse:** https://myaccount.google.com
2. **Vá para:** "Security" → "2-Step Verification"
3. **Configure a verificação em 2 etapas** (se ainda não tiver)
4. **Depois, vá para:** "Security" → "App passwords"
5. **Crie uma senha de app:**
   * **App:** Mail
   * **Device:** Other (custom name)
   * **Nome:** Sistema Agendamento
6. **Use essa senha** no campo "Senha do email" no sistema

**5.3 Testar Email**

1. **No painel admin**, na configuração de email
2. **Digite seu email** no campo "Email para teste"
3. **Clique em "📧 Enviar Email Teste"**
4. **Verifique sua caixa de entrada**

**🎨 Etapa 6: Personalização e Ajustes Finais**

**6.1 Personalizar Mensagens de Email**

1. **No painel admin:** "⚙️ Configurações" → "📧 Email"
2. **Edite o template de confirmação** com sua mensagem personalizada
3. **Use as variáveis:** {nome}, {data}, {horario}, {local}

**6.2 Configurar Bloqueios**

1. **Vá para "🗓️ Gerenciar Bloqueios"**
2. **Configure:**
   * **Dias específicos:** Feriados, faltas pontuais
   * **Períodos:** Férias, viagens
   * **Horários permanentes:** Almoço, intervalos

**6.3 Configurar Backup Automático**

1. **No painel admin:** "⚙️ Configurações" → "📧 Email"
2. **Ative o backup no GitHub**
3. **Configure backup por email** (opcional)

**✅ Checklist Final**

**🔍 Verificações Obrigatórias**

* [ ] Aplicação carrega sem erros
* [ ] Consegue acessar painel admin com ?admin=sua\_chave
* [ ] Agendamento funciona na interface pública
* [ ] Agendamentos aparecem no painel admin
* [ ] Google Calendar sincroniza (se configurado)
* [ ] Emails são enviados (se configurado)
* [ ] Backup GitHub funciona

**🎯 URLs Importantes**

**Anote essas URLs importantes:**

* **Sistema público:** https://seu-app.streamlit.app
* **Painel admin:** https://seu-app.streamlit.app?admin=sua\_chave
* **GitHub:** https://github.com/seu-usuario/sistema-agendamento
* **Streamlit Cloud:** https://share.streamlit.io

**🆘 Solução de Problemas**

**❌ App não carrega / Erro de dependências**

**Solução:**

1. Verifique se o requirements.txt está correto
2. No Streamlit Cloud: Settings → Reboot app
3. Aguarde 2-3 minutos para nova build

**❌ Erro "Google Calendar API not enabled"**

**Solução:**

1. Acesse Google Cloud Console
2. APIs & Services → Library
3. Busque "Google Calendar API" e habilite

**❌ Erro de autenticação Google**

**Solução:**

1. Verifique se Client ID e Client Secret estão corretos nos secrets
2. Verifique se o Refresh Token foi copiado corretamente
3. Teste novamente no OAuth Playground

**❌ Emails não são enviados**

**Solução:**

1. Verifique se a senha de app foi criada corretamente
2. Teste com outro email para verificar se não está indo para spam
3. Use a função "Testar Email" no painel admin

**❌ Backup GitHub não funciona**

**Solução:**

1. Verifique se o token GitHub tem permissão repo
2. Verifique se o nome do repositório está correto nos secrets
3. Teste fazer backup manual no painel admin

**🎉 Parabéns!**

Seu sistema de agendamento está funcionando!

**📚 Próximos Passos**

1. **Compartilhe o link** com seus clientes
2. **Configure bloqueios** conforme sua agenda
3. **Monitore agendamentos** pelo painel admin
4. **Faça backups regulares** dos dados

**🔧 Manutenção**

* **Acesse o painel admin** semanalmente
* **Verifique backups** mensalmente
* **Mantenha tokens atualizados** se necessário

**📞 Dúvidas?** Revise cada etapa cuidadosamente. Todos os links e configurações foram testados e funcionam corretamente quando seguidos na ordem indicada.

**✨ Boa sorte com seu sistema de agendamento!**