

Tutorial: Como Fazer Upload de Projetos para o GitHub (iOS)


Para quem usa iPhone ou iPad, o processo de enviar um projeto para o GitHub é um pouco diferente, pois não temos o Termux. A alternativa mais recomendada e poderosa é o aplicativo **Working Copy**, um cliente Git completo com uma interface gráfica fácil de usar.

Aqui está um passo a passo completo:


Pré-requisito: Salvar seu Projeto

Antes de começar, garanta que a pasta do seu projeto (contendo `src/`, `README.md`, etc.) esteja salva no aplicativo **Arquivos** (Files) do seu iPhone/iPad. Você pode criar os arquivos em qualquer editor de texto (como o Textastic, Runestone ou até mesmo online) e salvá-los em uma pasta no "Meu iPhone" ou "iCloud Drive".

Passo 1: Instalar e Configurar o Working Copy

1. Baixe o **Working Copy** na App Store. O aplicativo é gratuito para clonar, editar e enviar para repositórios públicos e privados.
2. Abra o aplicativo. Na primeira vez, ele pode pedir algumas permissões.
3. Toque no ícone de engrenagem () no canto superior esquerdo para abrir as **Configurações**.
4. Vá para a seção **Git** e configure seu **Nome de Usuário** e **E-mail**. Use o mesmo nome e e-mail que você configurou na sua conta do GitHub.

Passo 2: Importar seu Projeto Local

1. Na tela principal do Working Copy, toque no ícone de mais () no canto superior esquerdo.
2. Selecione a opção **"Setup a synced directory"** (Configurar um diretório sincronizado).
3. Navegue pelo aplicativo **Arquivos** até encontrar e selecionar a pasta principal do seu projeto (a pasta que contém a `src`).
4. O Working Copy irá analisar a pasta e mostrar os arquivos. Ele vai reconhecer que isso ainda não é um repositório Git e mostrará um botão **"Create Repository"**. Toque nele.

Passo 3: Fazer o Primeiro Commit

1. Após criar o repositório, o app mostrará todos os arquivos do seu projeto como "untracked" (não rastreados).
2. No topo, escreva uma mensagem de commit, por exemplo:
`Primeiro commit: Estrutura inicial do projeto .`
3. Marque a caixa **"Stage All"** (ou selecione os arquivos individualmente) para adicionar todos os arquivos ao commit.
4. Toque no botão **"Commit"** no canto superior direito.

Passo 4: Conectar ao GitHub e Enviar (Push)

1. **Garanta que você já tenha um repositório criado no GitHub.** Se ainda não criou, crie um repositório **vazio** (sem `README.md` ou outros arquivos) pelo navegador do celular.
2. **Copie a URL do seu repositório.** Será algo como
`https://github.com/seu-usuario/seu-repositorio.git` .
3. De volta ao Working Copy, na tela do seu repositório, toque no ícone de nuvem com uma seta para cima (☁ ↑) no canto superior direito. Isso abrirá a tela de "Remote".
4. Toque em **"Add Remote"**.
5. Cole a URL do seu repositório do GitHub no campo **"Remote URL"** e dê um nome ao remote (o padrão `origin` é perfeito). Toque em **"Save"**.
6. Agora, na tela de "Remote", você verá o `origin` listado. Toque no botão **"Push"**.
7. Na primeira vez, o Working Copy pedirá para você fazer login no GitHub para se autenticar. Siga os passos na tela.
8. Após a autenticação, o envio será concluído. Pronto! Seu código está no GitHub.

Para Atualizações Futuras

O processo fica muito mais simples para as próximas vezes:

1. Faça as alterações nos seus arquivos de código (usando um editor de texto que salve no app Arquivos).
2. Abra o Working Copy. Ele detectará as mudanças automaticamente.
3. Escreva uma nova mensagem de commit (ex:
`feat: Adiciona cálculo de média na Atividade 1`).
4. Toque em **"Commit"**.
5. Toque no ícone de nuvem (☁ ↑) e depois em **"Push"**.