



# Desenvolvimento WEB

## GIT e GITHub

### Aula 5



# Sumário

Github Desktop

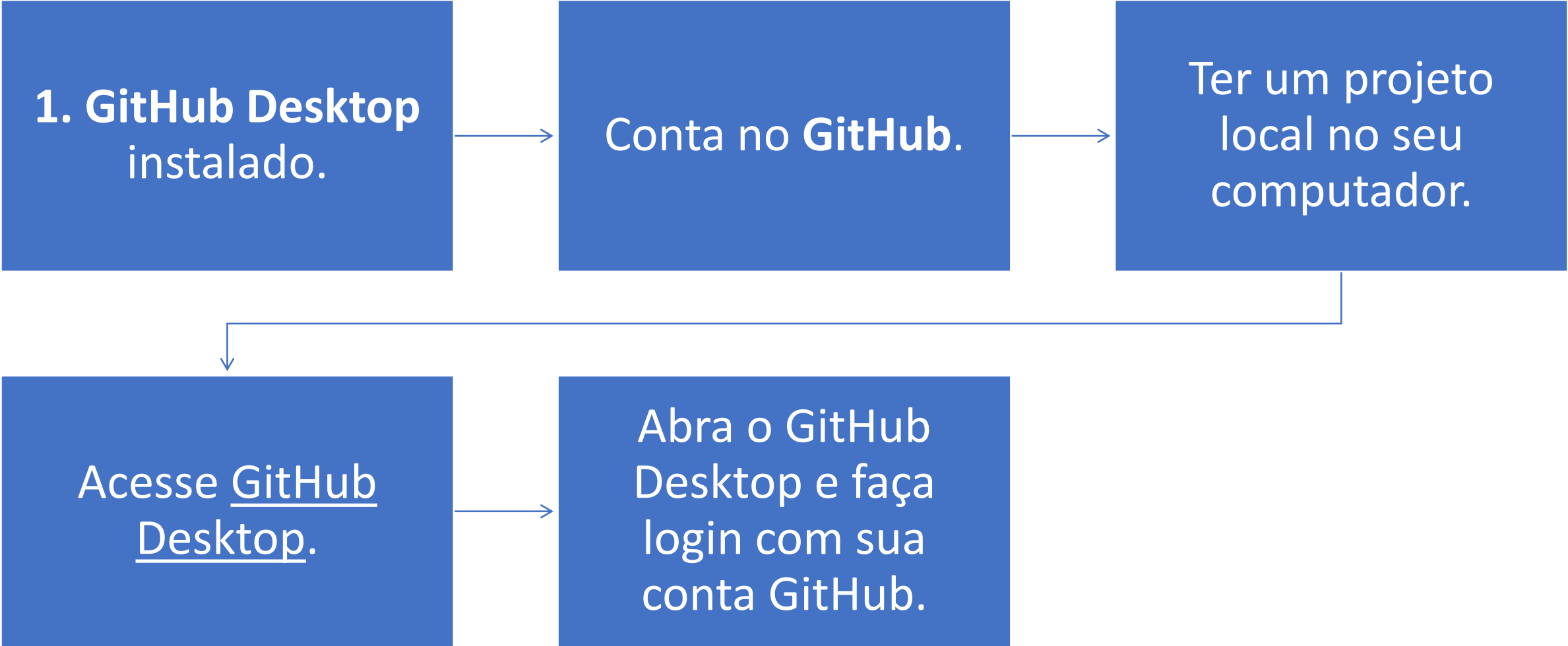
Github VS Code

Github Terminal Bash

Github Pages

- **Objetivo:** Aprender a usar o Git e GutHub





## Passo 3 - Criar um Novo Repositório no GitHub Desktop

1. Abra o GitHub Desktop.
2. Clique em **File > New repository**.
3. Defina o nome do repositório e o caminho local (Local Path).
4. Se desejar, marque a opção "**Initialize this repository with a README**".
5. Clique em **Create repository**.

## Passo 4 - Subir o Repositório para o GitHub

- 1.No GitHub Desktop, clique em **Repository > Publish repository**.
- 2.Preencha o nome do repositório e uma breve descrição.
- 3.Clique em **Publish repository** para fazer o upload para o GitHub.

## Passo 5 - Confirmar no GitHub

1. Acesse o GitHub no seu navegador.
2. Veja seu repositório recém-criado com os arquivos enviados.
3. Agora, seu projeto está disponível publicamente ou privado, conforme sua escolha.

## Passo 6 - Fazer Alterações e Sincronizar

1. Faça alterações no seu projeto local (por exemplo, edite ou adicione arquivos).
2. No GitHub Desktop, veja as mudanças na aba **Changes**.
3. Clique em **Commit to main** para registrar as alterações.
4. Clique em **Push origin** para enviar as alterações para o GitHub.

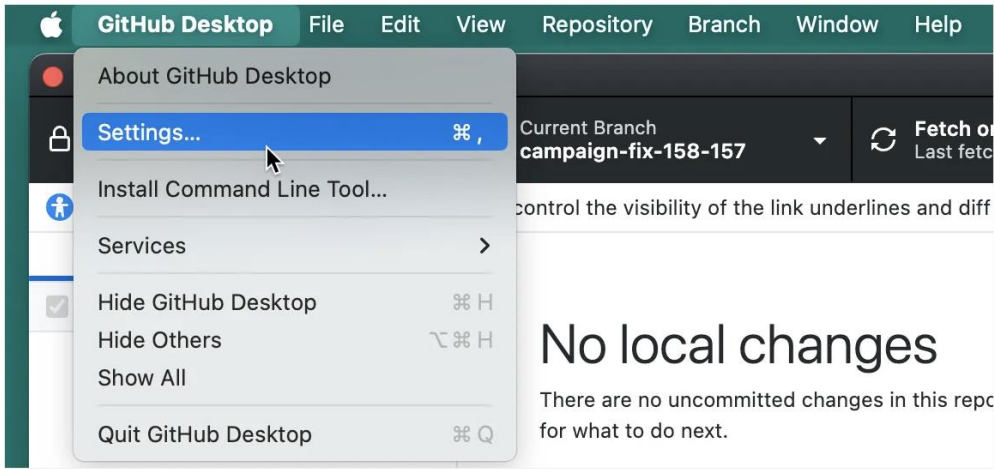


**GitHub Desktop** facilita o uso do Git para gerenciamento de código.

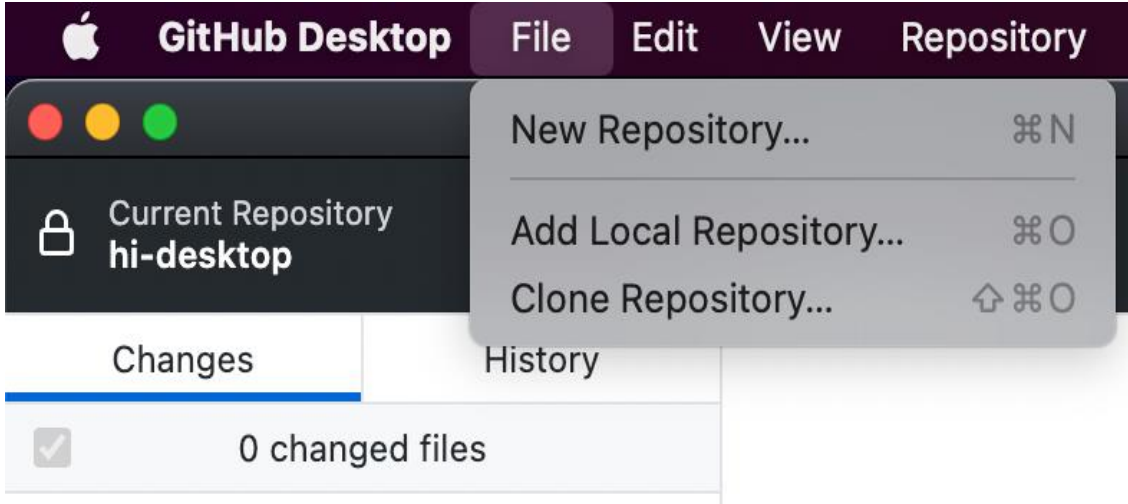
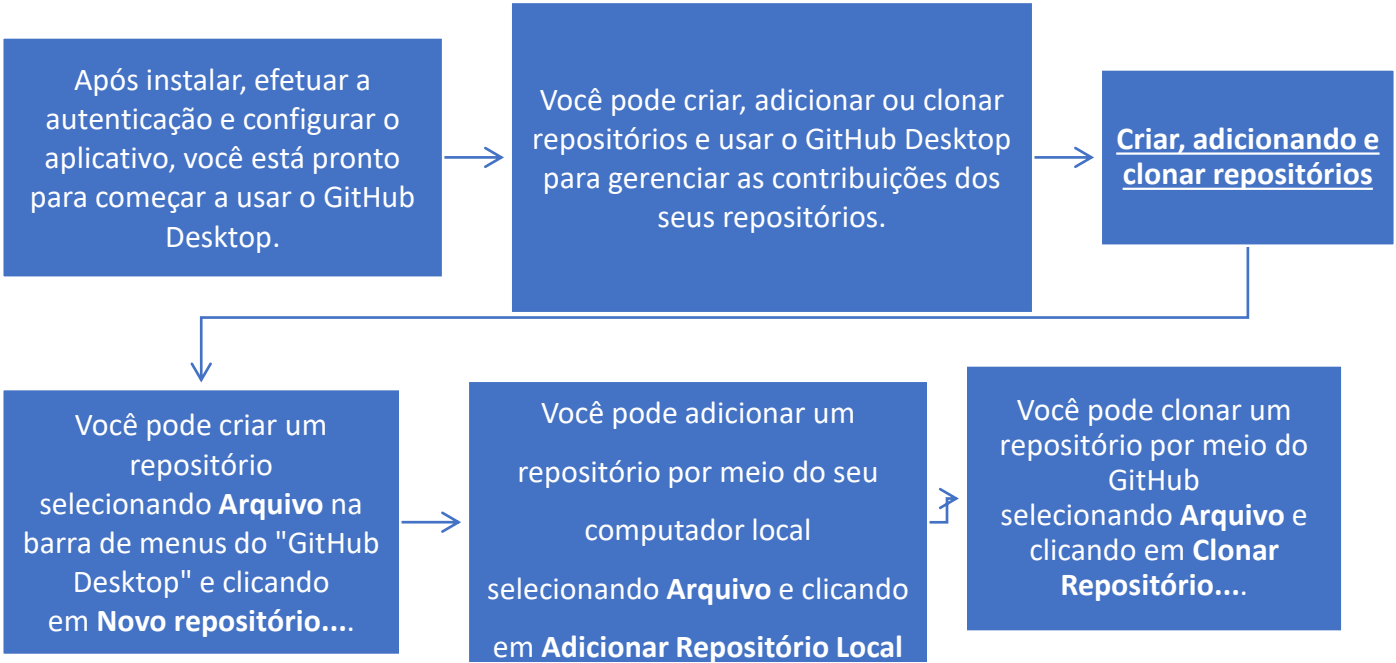
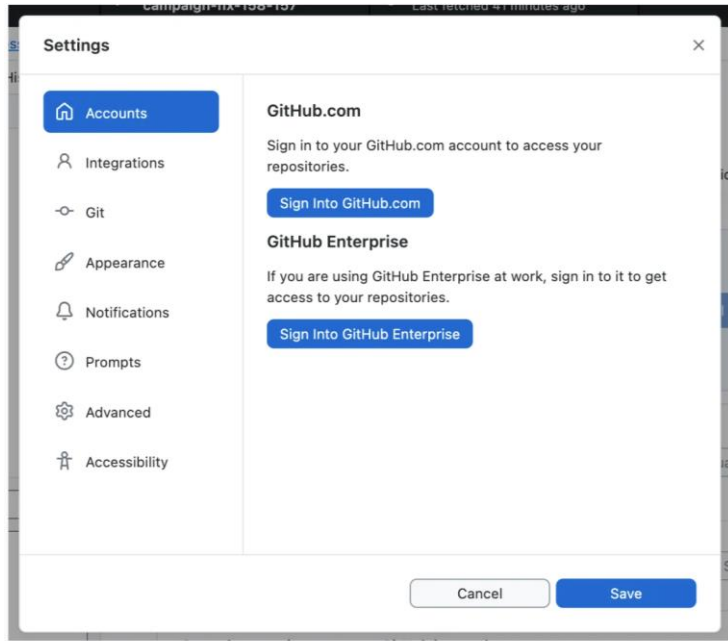
**Subir projeto** envolve:

- Criar repositório local.
- Publicar para o GitHub.
- Sincronizar alterações futuras.

Na barra de menus, selecione **GitHub Desktop** e selecione **Configurações**.



Na janela "Configurações", no painel **Contas**, selecione o botão "Entrar" apropriado. Use **Entrar no GitHub Enterprise** para entrar no GitHub Enterprise Server ou no GitHub Enterprise Cloud com residência de dados.



# Referências de Estudo:



✓ [https://www.youtube.com/watch?v=y\\_3hJcw0dns&t=582s](https://www.youtube.com/watch?v=y_3hJcw0dns&t=582s)

✓ <https://youtu.be/nAHVEzDBVeo?si=Byy9PpOmMPMW-vi5>

A solid dark grey vertical bar on the left side of the slide.

Github Terminal Bash

Github VS Code

# COMANDOS BÁSICOS DE GIT

## GIT INIT

Crea un repositorio vacío o reinicia uno existente.

## GIT ADD

Agrega archivos al staged (staging area - index).

## GIT COMMIT

Registra los cambios en el repositorio.

## GIT STATUS

Muestra el estado actual del repositorio.

## GIT CLONE

Clona un repositorio en un nuevo directorio.

## GIT PUSH

Actualiza las referencias remotas junto con los objetos asociados

## GIT PULL

Actualiza las referencias locales junto con los objetos asociados

## GIT CHECKOUT

Cambia de rama o restaura archivos del espacio de trabajo.

GIT es un sistema de control de versiones

 /developaul





|  |   |  |
|--|---|--|
| 1. Instalar o Git                          | Instalar o Git no seu computador para gerenciar repositórios locais e remotos.                            | Baixe e instale o Git a partir de <a href="https://git-scm.com">git-scm.com</a> . Depois, verifique a instalação com <code>git --version</code> .  |
| 2. Criar um Repositório no GitHub          | Crie um repositório no GitHub onde seu projeto será armazenado online.                                    | Acesse <a href="https://github.com">GitHub</a> , crie um repositório novo e anote a URL do repositório criado.   |
| 3. Inicializar o Repositório Local         | Crie um repositório Git local dentro da pasta do seu projeto.   | Abra o terminal na pasta do projeto e execute <code>git init</code> para inicializar o repositório Git local.  |
| 4. Adicionar Arquivos ao Repositório Local | Adicione os arquivos do seu projeto ao repositório Git local para prepará-los para o commit.              | Execute <code>git add .</code> para adicionar todos os arquivos ou <code>git add &lt;arquivo&gt;</code> para adicionar um arquivo específico.  |
| 5. Fazer o Commit Inicial                  | Registre as alterações feitas (adicionar arquivos) com uma mensagem explicativa.                          | Execute <code>git commit -m "Commit inicial do projeto"</code> para salvar as mudanças no repositório local.   |
| 6. Conectar o Repositório Local ao GitHub  | Conecte o repositório local ao repositório remoto do GitHub para poder enviá-lo para a nuvem.             | Execute <code>git remote add origin &lt;URL_do_repositorio&gt;</code> (substitua <code>&lt;URL_do_repositorio&gt;</code> pela URL do repositório no GitHub).   |
| 7. Subir o Projeto para o GitHub           | Envie o conteúdo do repositório local para o GitHub pela primeira vez.                                    | Execute <code>git push -u origin master</code> para enviar os arquivos locais ao GitHub. <b>Autenticação pode ser necessária.</b>  |
| 8. Fazer Alterações e Sincronizar          | Após o primeiro envio, modifique seu projeto, adicione novas mudanças, faça commit e envie para o GitHub. | <ul style="list-style-type: none"> <li>- Modifique seus arquivos.</li> <li>- Use <code>git add .</code> para adicionar as modificações.</li> <li>- Execute <code>git commit -m "Descrição das mudanças"</code>.</li> <li>- Envie com <code>git push origin master</code>.</li> </ul> |

1. `git init`
2. `git add .`
3. `git commit -m "nome do commit"`
4. `git remote add origin url do repositorio`
5. `git push origin master`

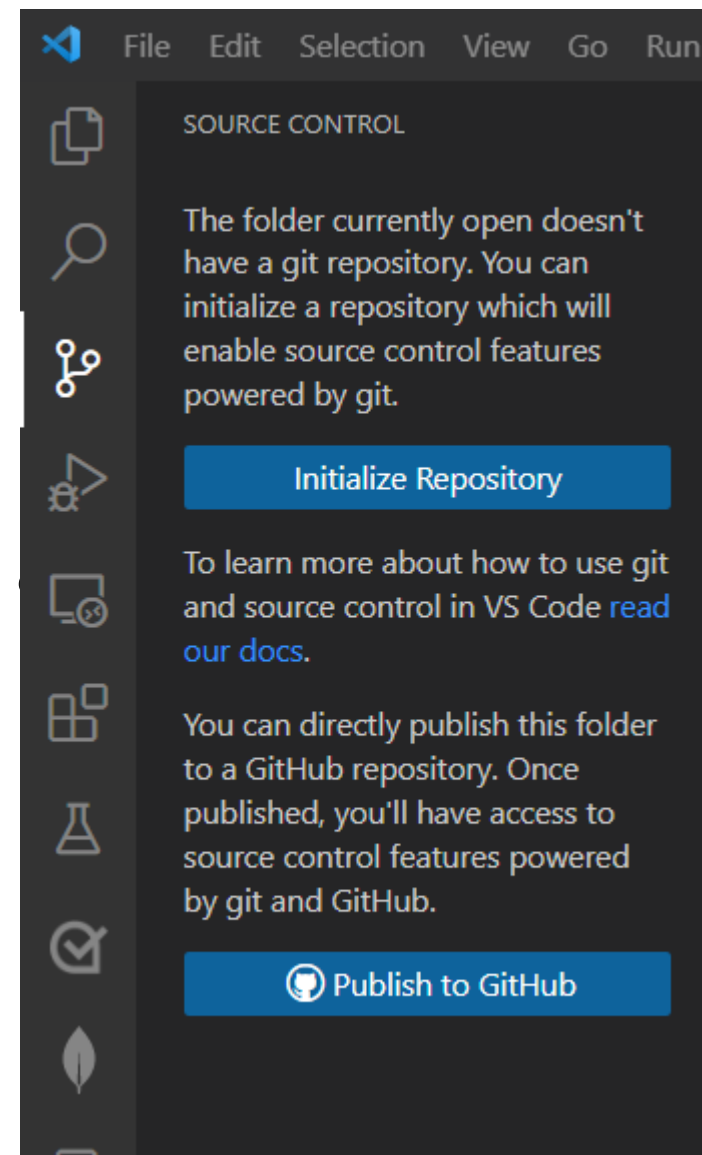




# Visual Studio Code

1. Selecione o controle do código-fonte na barra de use a combinação de teclas de `Ctrl + Shift + G`.

2. Selecione **Inicializar repositório**.



Depois que o repositório for inicializado, [crie o repositório no GitHub](#).

Em seguida, adicione esse repositório como um repositório remoto ao seu projeto local na Paleta de comandos: procure por Git: Add remote.

## Criar um branch para alterações

Crie um novo branch para capturar alterações e isolar do branch principal ou padrão.

1. Selecione o controle do código-fonte na barra de atividades.
2. Selecione reticências (...) ao lado de **Controle do Código-Fonte**.
3. Selecione **Ramificação** -> **Criar Ramificação**.

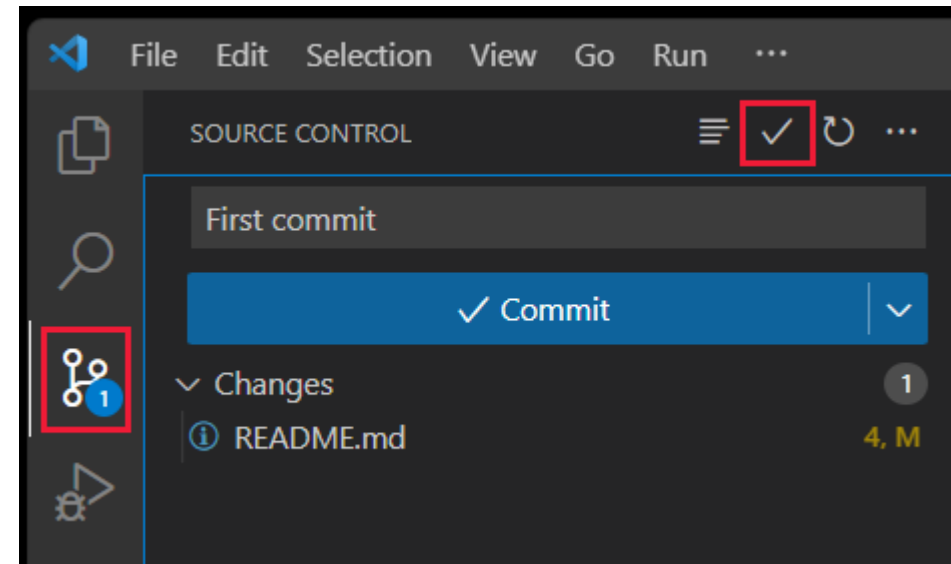


## Confirmar alterações localmente

Depois de fazer alterações em seu branch, confirme as alterações.

1. Selecione o controle do código-fonte na barra de atividades.

2. Insira sua mensagem de confirmação e selecione **Confirmar**.



# Enviar um branch local para o GitHub

Selecione o ícone Controle do Código-Fonte na barra de atividade.

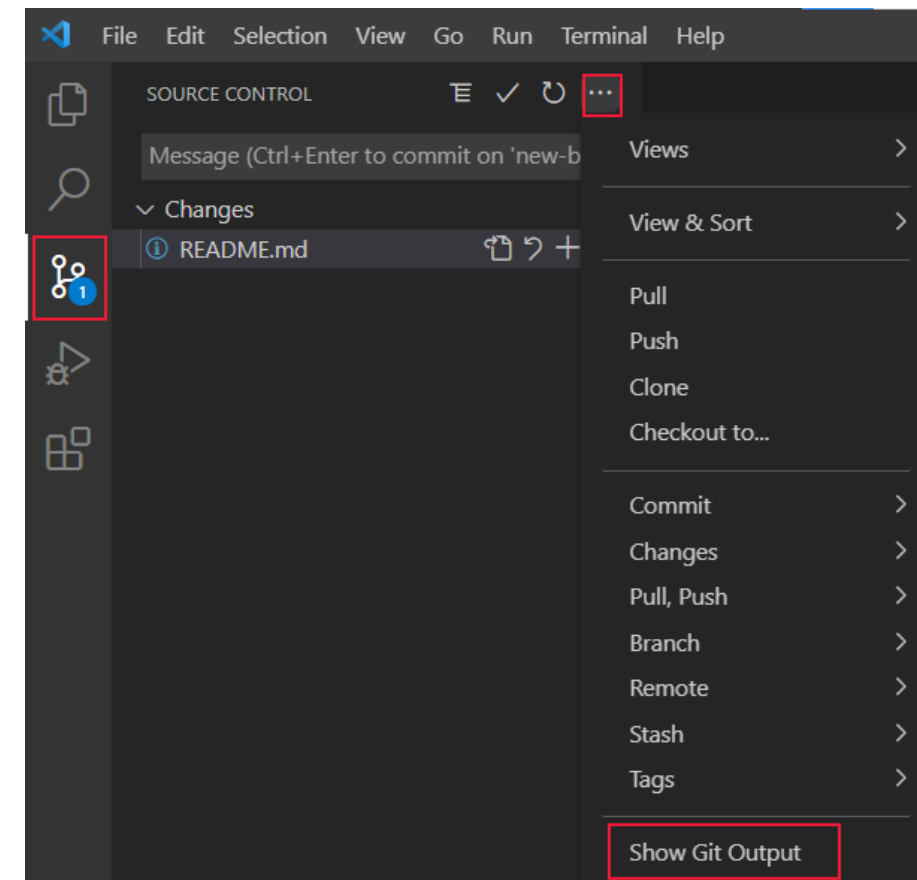
Selecione **Publicar Ramificação**. Se o repositório não existir no GitHub, ele criará o repositório para você.

## Exibir saída do Git

Você pode exibir os comandos do Git executados ao usar a extensão de Controle do código-fonte. Isso ajuda a depurar quando um comando falha.

Selecione o ícone Controle do Código-Fonte na barra de atividade.

Selecione as reticências (...) e **Mostrar Saída do Git**.



<https://code.visualstudio.com/docs/sourcecontrol/github>

<https://learn.microsoft.com/pt-br/azure/developer/javascript/how-to/with-visual-studio-code/clone-github-repository?tabs=activity-bar>

# Como usar a integração do Git no Visual Studio Code

## Familiarizando-se com a guia de controle de código-fonte e inicializando um repositório Git.

A integração do Git no VS Code começa com a inicialização de um projeto como um repositório Git, seja via terminal ou pela guia de controle de código-fonte. Após inicializar, os arquivos novos ou modificados aparecem na guia Source Control, onde podem ser rastreados, adicionados e confirmados.

## Compreensão dos indicadores de medianiz para alterações no código.

A medianiz no VS Code, localizada à direita dos números das linhas, fornece indicações visuais sobre as alterações no código: uma marca azul para linhas alteradas, um triângulo vermelho para linhas excluídas e uma barra verde para linhas adicionadas. Isso facilita a identificação rápida das modificações no código.

## Comparação de arquivos e gerenciamento de ramificações no VS Code.

O VS Code permite comparar arquivos diretamente na interface, exibindo as diferenças entre as versões. Além disso, é possível criar e alternar entre ramificações (branches) usando a barra inferior, facilitando o trabalho colaborativo e o desenvolvimento de recursos sem afetar a base de código principal.

## Utilização de repositórios remotos e instalação de extensões úteis para Git no VS Code.

Através da guia de controle de código-fonte, é possível interagir com repositórios remotos, utilizando comandos como pull, sync e publish. Além disso, extensões como Git Blame, Git History e GitLens aprimoram a experiência, fornecendo informações sobre a autoria do código, histórico de alterações e funcionalidades de comparação.

# Referência de Estudo:

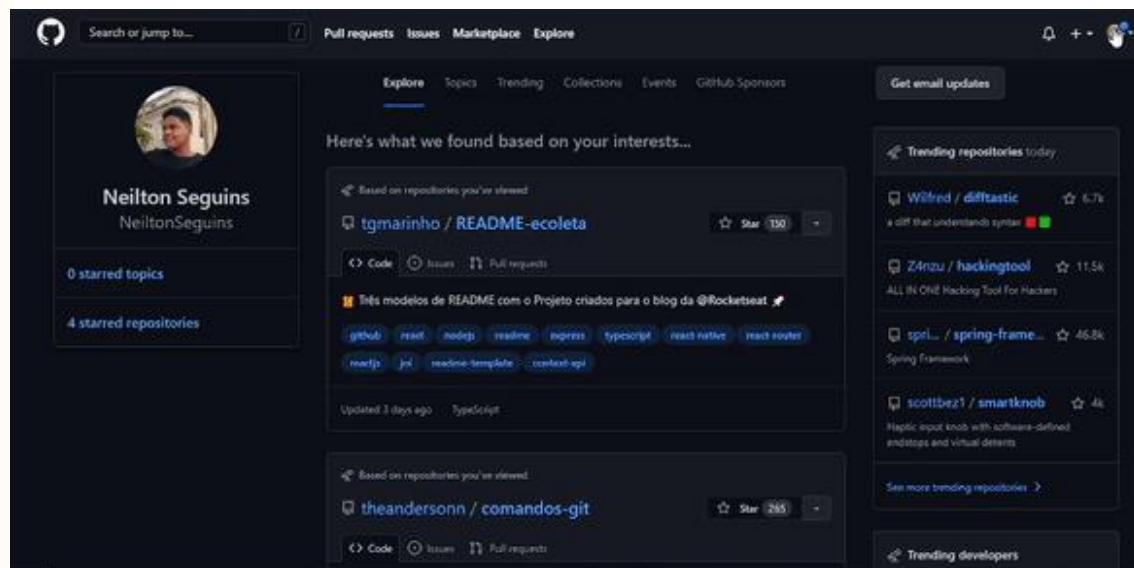


- ✓ <https://www.youtube.com/watch?v=Zwv9qRyVeU4>
- ✓ <https://www.youtube.com/watch?v=3UmHejdKes0>



# O que é hospedagem e sua necessidade

Hospedagem é um serviço online que permite a publicação de sites e aplicações, armazenando os arquivos necessários em um servidor para que sejam acessíveis na internet. Ao contrário do desenvolvimento local, a hospedagem disponibiliza o site para o mundo.



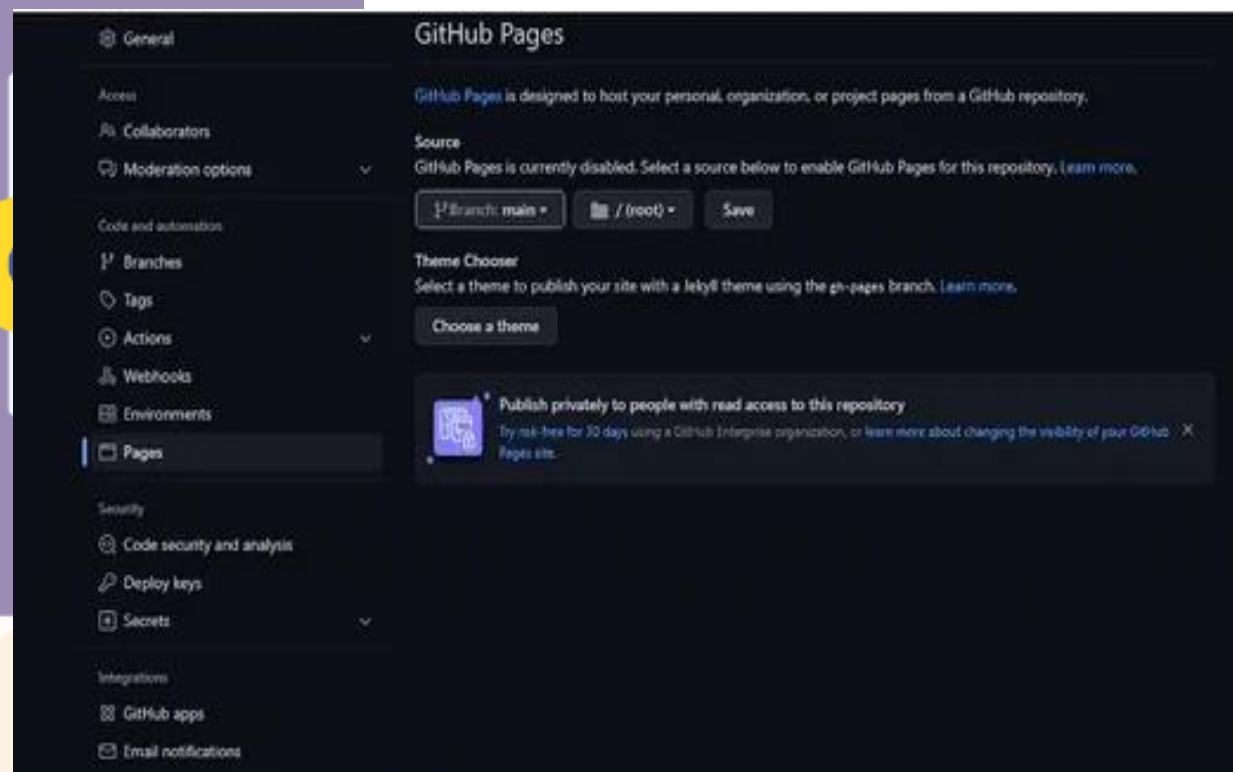
## Github Pages: Uma solução de hospedagem gratuita



Github Pages é um serviço de hospedagem de sites estáticos do Github, que utiliza arquivos HTML, CSS e Javascript diretamente de um repositório para publicar um site. Ele permite hospedar sites de projetos pessoais, empresas ou organizações, utilizando um domínio github.io ou um domínio personalizado.

# Como colocar seu projeto no ar usando o Github Pages

Para hospedar seu projeto, crie um repositório no Github e acesse as configurações do repositório. Na seção 'Pages', selecione a branch de publicação e, opcionalmente, uma pasta específica. O Github Pages irá gerar um URL com o domínio github.io para você acessar o seu site.



## Requisitos e dicas importantes



Certifique-se de ter um repositório no Github. Se usar o Github Free, o repositório deve ser público. Após a configuração, aguarde o GitHub Pages concluir o processo e verifique o URL gerado.

# Referências de Estudo:



✓ <https://www.youtube.com/watch?v=TSTk1xjeWro>

✓ <https://www.youtube.com/watch?v=jI3782DGDc>





# Exercícios

- ✓ Projetos das atividades anterior no github
- ✓ Deploy dos projetos.