

APOSTILA DE UTILIZAÇÃO: GITHUB E GITHUB DESKTOP



Nome: Kaio Gomes do Nascimento Mazza

I.A utilizada: ChatGPT

1. O QUE É O GITHUB?

GitHub é uma plataforma online que utiliza o sistema de controle de versões Git para gerenciar e armazenar o código-fonte de projetos. Ele facilita o compartilhamento, colaboração e versionamento de código, sendo amplamente usado por desenvolvedores e equipes ao redor do mundo.

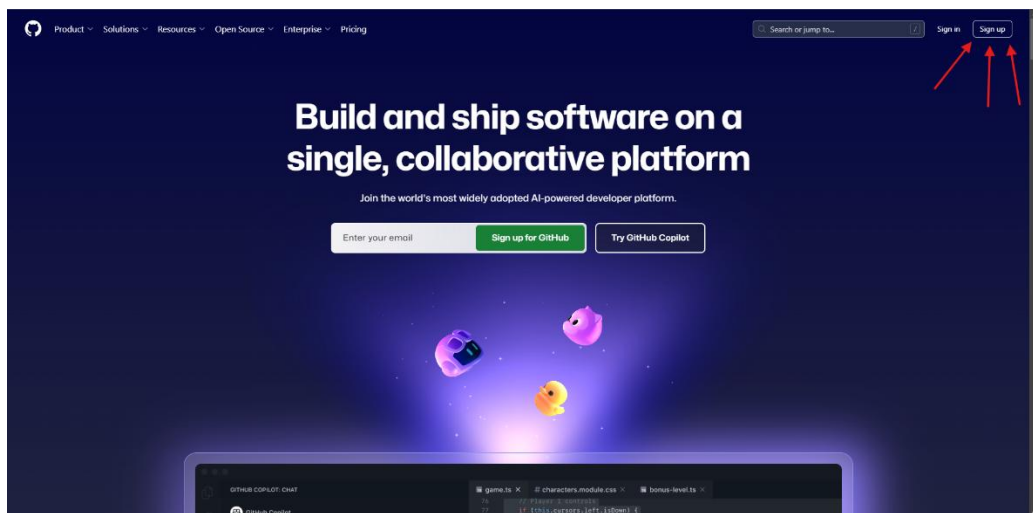
Principais características do GitHub:

- Repositórios Remotos: Armazena seu código na nuvem.
- Controle de Versão: Registra todas as alterações feitas no código, permitindo reverter versões anteriores.
- Colaboração: Permite que várias pessoas trabalhem juntas no mesmo projeto.
- Pull Requests: Facilita a revisão e integração de mudanças feitas por diferentes desenvolvedores.
- Issues e Wiki: Recursos para organizar tarefas e documentar projetos.

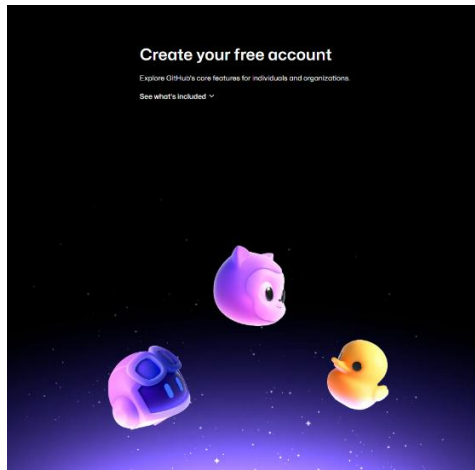
2. CRIANDO UMA CONTA NO GITHUB:

Passos para criar sua conta:

1. Acesse o site oficial: <https://github.com>.
2. Clique em "Sign Up" no canto superior direito da página.



3. Preencha os dados necessários: Insira seu nome de usuário, e-mail e senha.



Already have an account? [Sign in](#)

Sign up to GitHub

Email

Password

Username

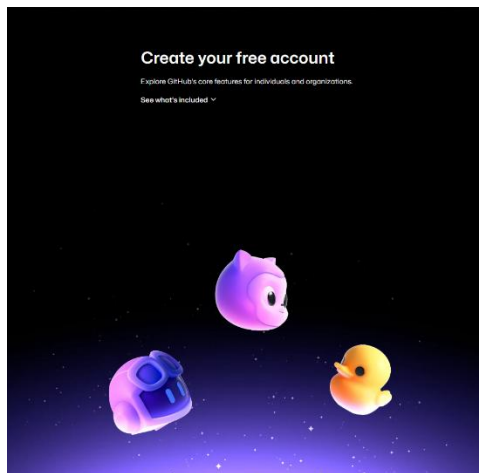
Password should be at least 16 characters. GitHub also requires a number and a lowercase letter.

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

[Continue](#)

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account related emails.

- Verificação de e-mail: O GitHub enviará um e-mail de confirmação. Abra o e-mail e clique no link para validar sua conta.



Already have an account? [Sign in](#)

Confirm your email address

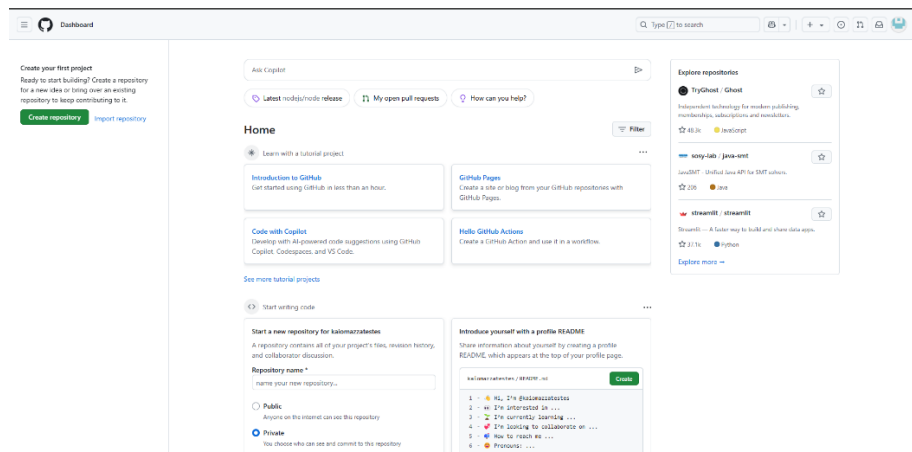
We have sent a code to [fipalicio@gmail.com](#)

Enter code

[Continue](#)

Didn't get your email? [Resend the code](#) or [update your email address](#)

- Escolha o plano: O GitHub oferece planos gratuitos e pagos. Para a maioria dos usuários, o plano gratuito é suficiente.
- Configurações iniciais: Após o login, você pode personalizar seu perfil e definir preferências de notificações.



Simples, não é? Então agora, continue no foco aprendendo sobre o nosso querido Github!

3. O QUE É O GITHUB DESKTOP?



O GitHub Desktop é uma aplicação gráfica que facilita o uso do GitHub, permitindo que você gerencie seus repositórios localmente no seu computador, sem a necessidade de usar a linha de comando. Ele é projetado para simplificar a interação com repositórios do GitHub de forma visual e intuitiva.

Funcionalidades principais:

- Gerenciar Repositórios Locais e Remotos: Crie, clone e gerencie repositórios diretamente no GitHub Desktop.
- Commit, Push e Pull: Realize operações comuns como commits, push e pull de forma gráfica.
- Gerenciamento de Branches: Crie, mude e gerencie branches sem usar o terminal.
- Facilidade de Uso: Interface simples e fácil de navegar, ideal para iniciantes e desenvolvedores que preferem uma GUI.

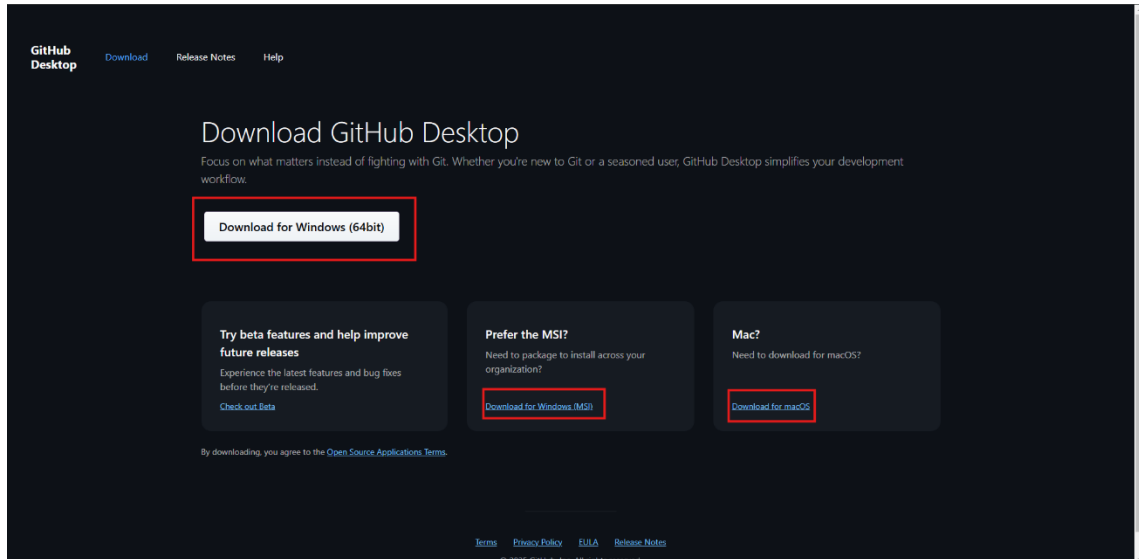
Realmente, o Github Desktop traz uma facilidade imensa para aqueles que utilizam Git e Github! Por isso, é necessário nos atualizarmos e não ficar somente nos terminais.

4. INSTALANDO O GITHUB DESKTOP:

Passos para instalar o GitHub Desktop:

1. Baixe o GitHub Desktop:

- Acesse o site oficial: <https://desktop.github.com>.
- Clique no botão "Download for Windows" (ou para MacOS, caso esteja usando este sistema).



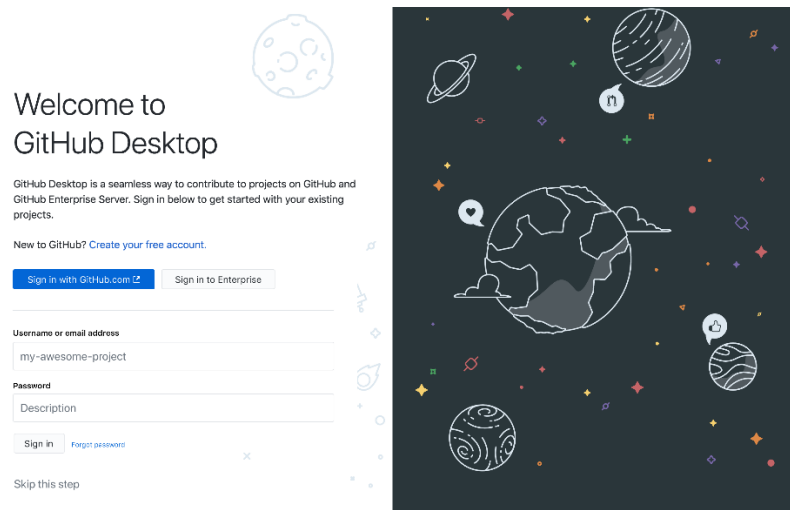
2. Instale o GitHub Desktop:

- Abra o arquivo baixado e siga as instruções do instalador (não há opções complicadas, basta clicar em "Next" até o fim).

3. Abrindo o GitHub Desktop:

- Após a instalação, abra o aplicativo GitHub Desktop.

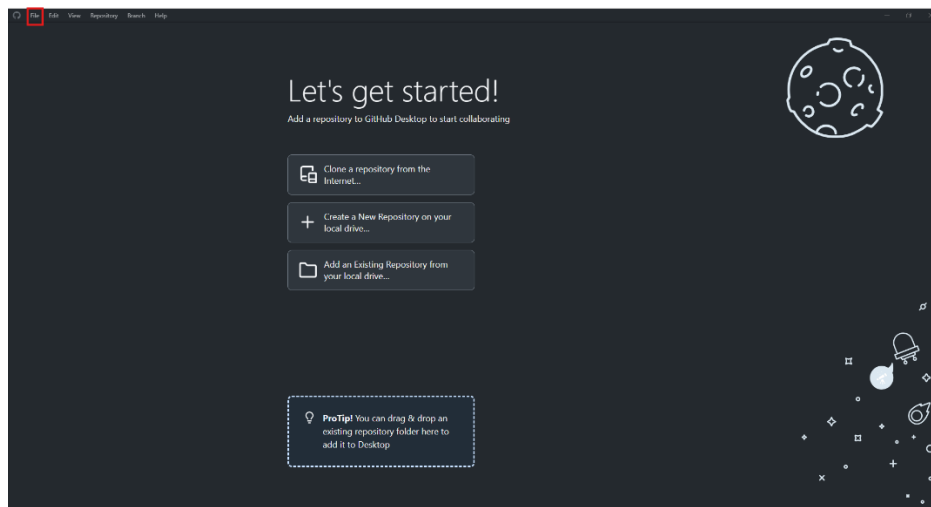
4. Conectando a conta se estiver abrindo o Github Desktop pela primeira vez:



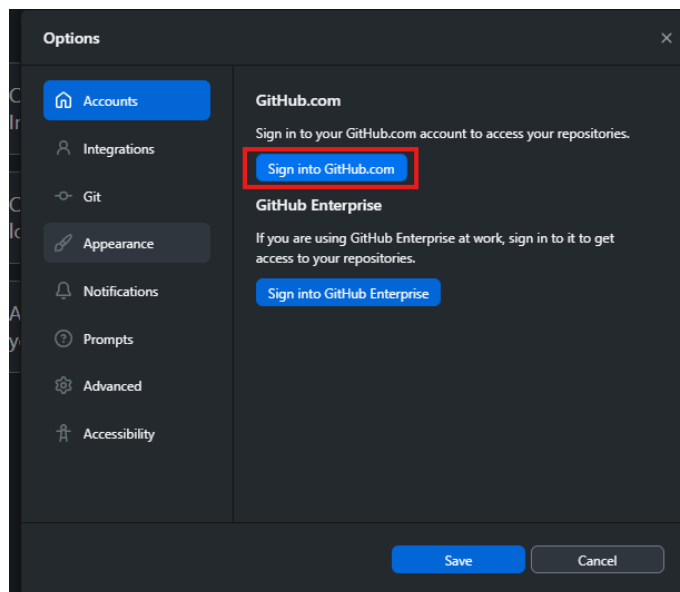
Caso você esteja abrindo o Github Desktop pela primeira vez, clique no botão do menu inicial: "Sign in Github.com". Então, ele automaticamente o redirecionará você para navegador principal de seu computador, então você irá fazer login com a conta criada anteriormente!

5. Conectando a conta no menu inicial do APP:

- No menu superior, clique na opção “File”;



- Após isso, clique na opção “Options”;
- E agora clique no botão “Sign into GitHub.com” e realize o processo de login.

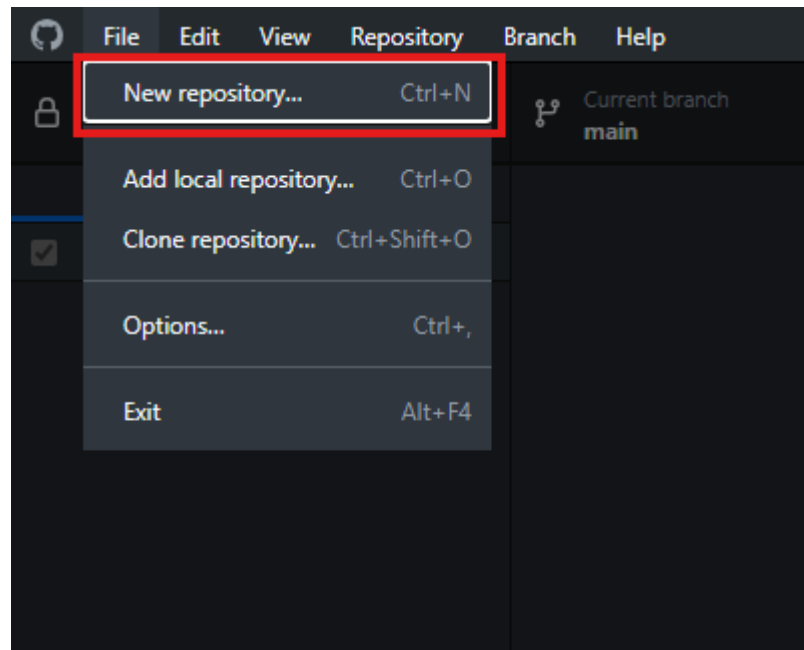


E fácil assim, você concluiu a instalação e o APP já está conectado com sua conta do Github!

5. CRIANDO E CONFIGURANDO UM REPOSITÓRIO LOCAL NO GITHUB DESKTOP

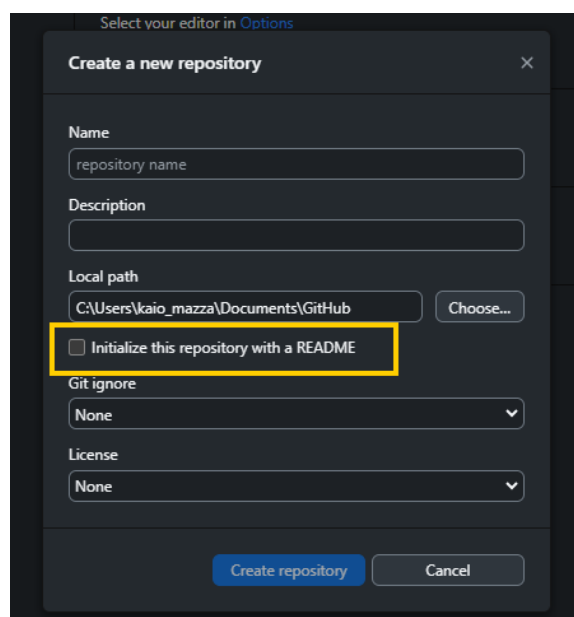
Passos para Criar um Novo Repositório Local:

1. Abra o GitHub Desktop.
2. Clique em "File" (no menu superior) e selecione "New Repository...".

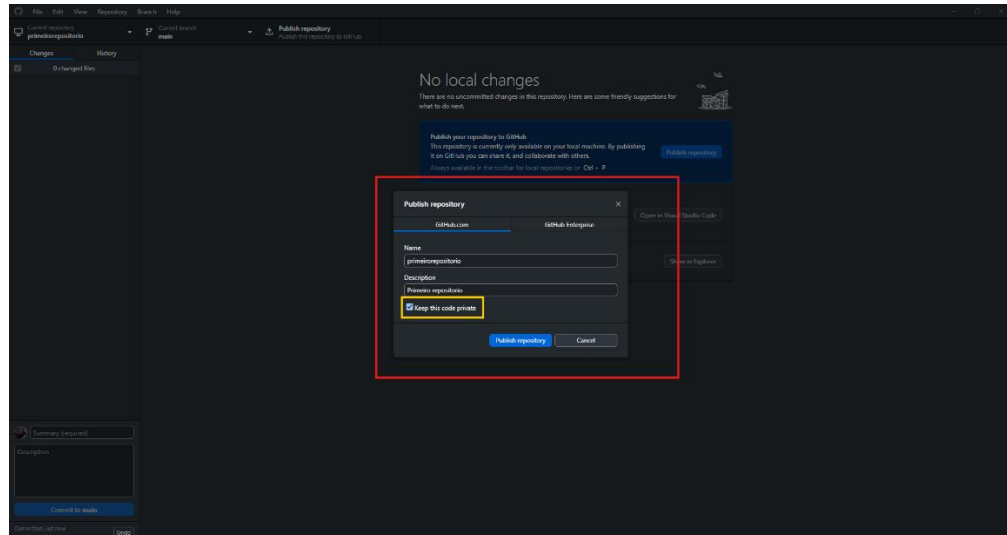


3. Configuração do Repositório:

- Nome: Escolha o nome do seu repositório.
- Descrição: Opcional, adicione uma descrição breve sobre o projeto.
- Local Path: Escolha a pasta onde o repositório será armazenado no seu computador.
- Inicialização do Repositório: O GitHub Desktop oferece a opção de inicializar o repositório com um arquivo README.md ou .gitignore. Deixe estas opções ativadas se desejar.



4. Clique em "Create Repository": Isso cria um repositório local no seu computador.
5. Após a criação do seu repositório, irá aparecer no menu "Changes": Publish repository. Esse comando envia o seu repositório do espaço local (seu computador) para a nuvem, na sua conta do Github que você criou e conectou no Github Desktop.



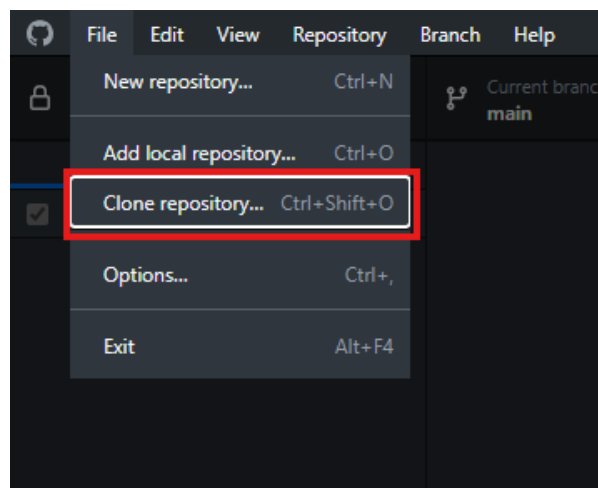
Agora, você já tem o seu primeiro repositório e ele já está pronto para uso.

6. CLONANDO UM REPOSITÓRIO DO GITHUB NO GITHUB DESKTOP

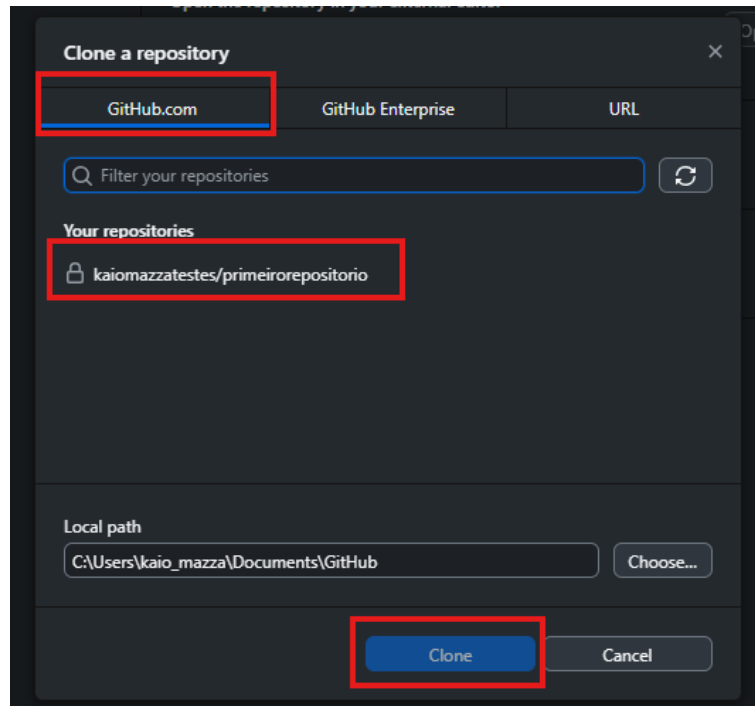
Clonar um repositório significa criar uma cópia local de um repositório que está hospedado no GitHub.

Passos para Clonar um Repositório:

1. Abra o GitHub Desktop.
2. Clique em "File" > "Clone Repository...".



3. Selecione "GitHub.com" na aba que abrirá.
4. Escolha o Repositório: Todos os repositórios que você possui no GitHub estarão listados. Selecione o repositório desejado.
5. Escolha o Local para Clonar: Selecione o diretório onde deseja armazenar o repositório no seu computador.

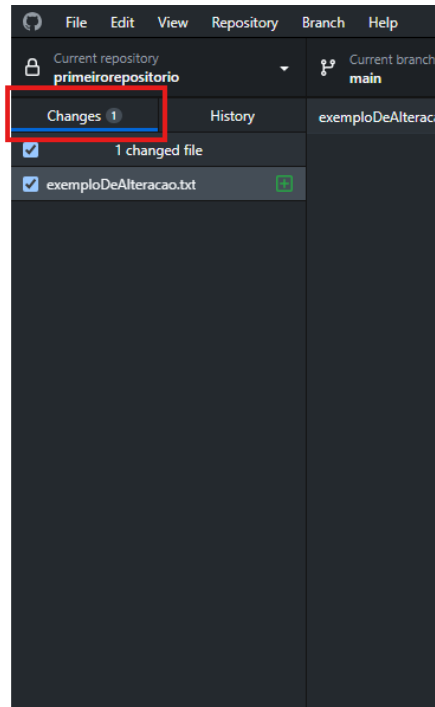


6. Clique em "Clone": O repositório será copiado para o seu computador.

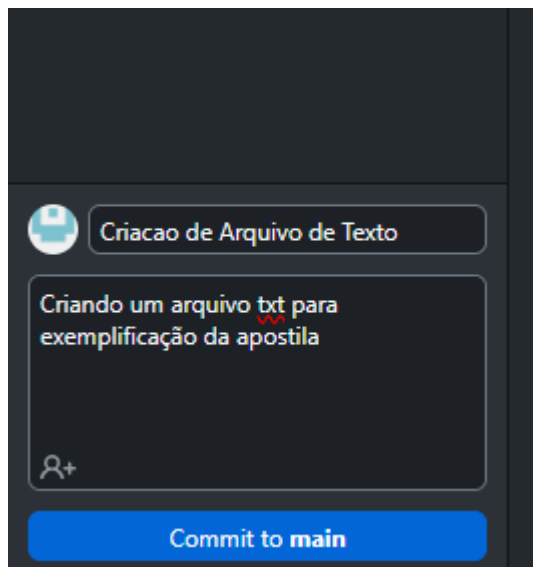
7. FAZENDO ALTERAÇÕES NO REPOSITÓRIO E COMMITANDO (GITHUB DESKTOP)

Passos para Editar e Commitar:

1. Abra o repositório no GitHub Desktop.
2. Faça alterações nos arquivos: Abra o repositório em um editor de código (como VSCode ou Sublime Text) e edite os arquivos.
3. Volte ao GitHub Desktop: No GitHub Desktop, você verá os arquivos modificados listados na seção "Changes".



4. Adicione uma Mensagem de Commit: Na parte inferior da tela, escreva uma mensagem breve e clara descrevendo as mudanças que você fez.
5. Clique em "Commit to main": Isso irá salvar suas alterações localmente no repositório.

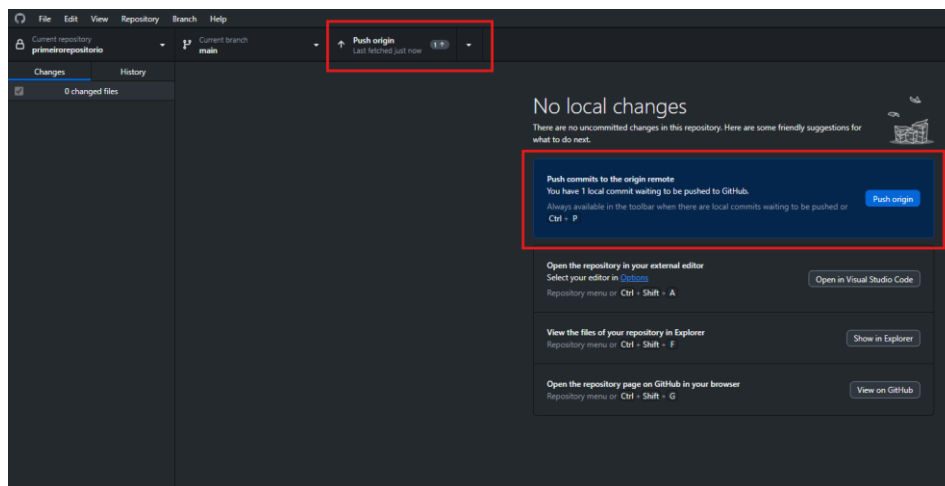


8. ENVIANDO ALTERAÇÕES PARA O GITHUB (PUSH)

Após realizar o commit, você pode enviar suas alterações para o repositório remoto no GitHub.

Passos para Fazer Push:

1. Após o commit, clique no botão "Push origin" no topo da tela.
2. Isso enviará as alterações para o repositório remoto no GitHub.

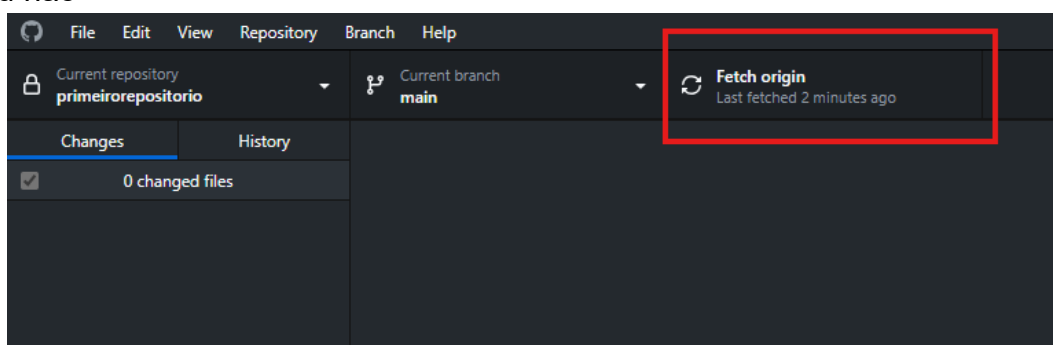


9. SINCRONIZANDO SEU REPOSITÓRIO LOCAL COM O REMOTO (PULL)

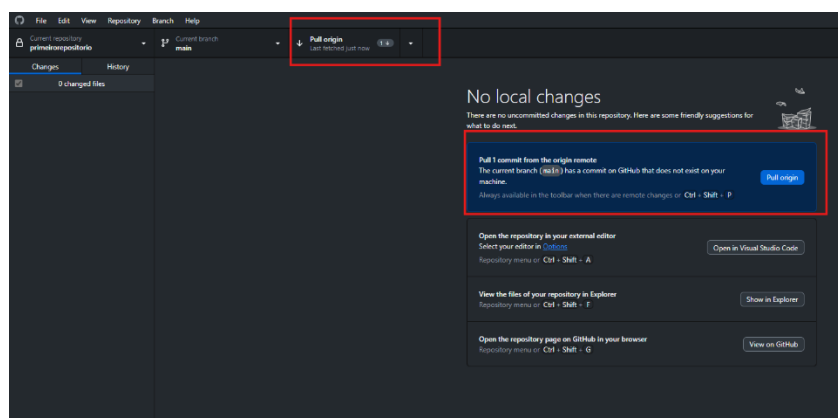
Se outras pessoas estiverem trabalhando no repositório, você deve sincronizar as alterações que foram feitas no repositório remoto.

Passos para Fazer Pull:

1. Abra o GitHub Desktop.
2. Clique em "Fetch" para atualizar e trazer informações se há algo novo na nuvem ou não



3. Clique em "Pull": Isso irá baixar as alterações mais recentes feitas no repositório remoto e integrá-las ao seu repositório local.



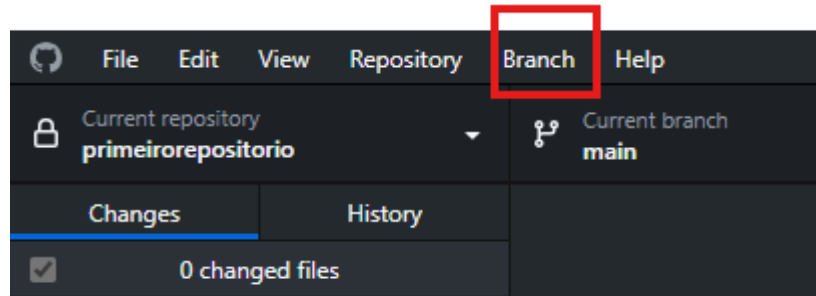
10. TRABALHANDO COM BRANCHES NO GITHUB DESKTOP

O que são Branches?

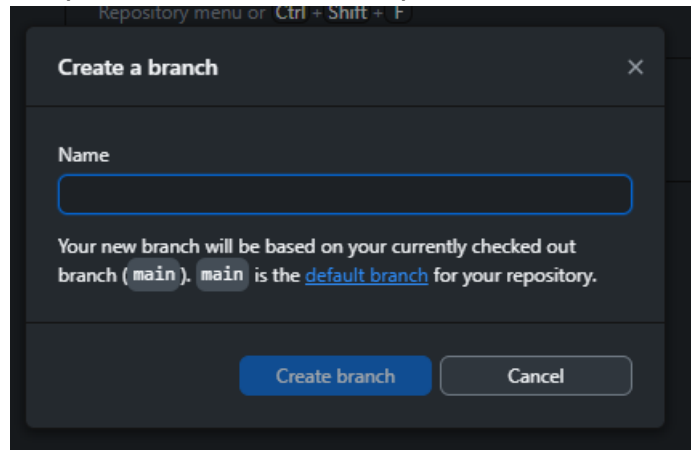
Branches permitem que você trabalhe em diferentes versões do projeto sem afetar o código principal (geralmente na branch main).

Passos para Criar um Novo Branch:

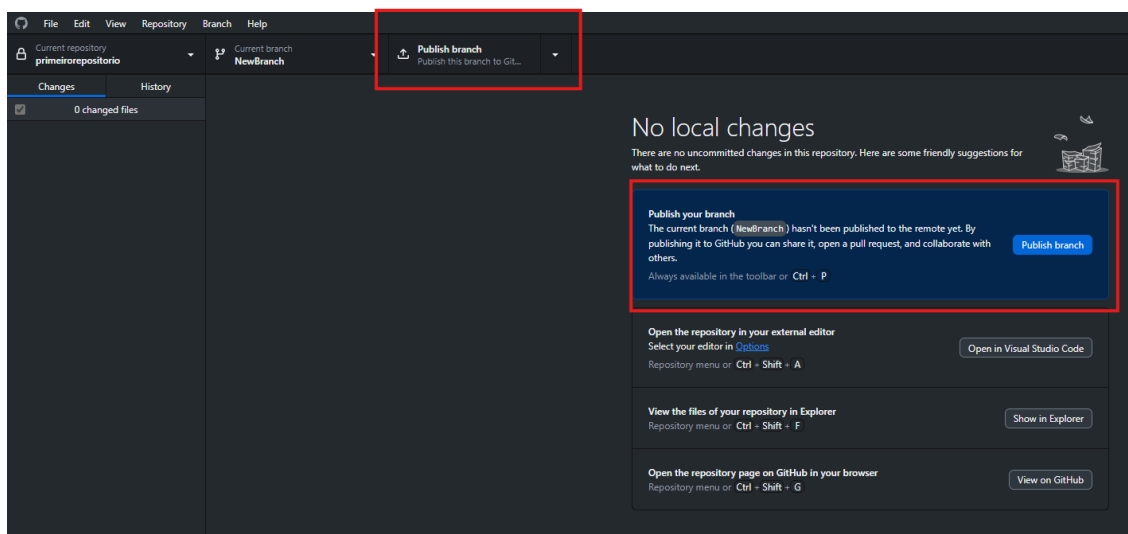
1. Abra o GitHub Desktop.
2. Clique em "Branch" no menu superior e selecione "New Branch...".



3. Escolha um nome para o seu branch e clique em "Create Branch".



4. Agora faça as alterações necessárias e publique sua Branch.



Passos para Mudar de Branch:

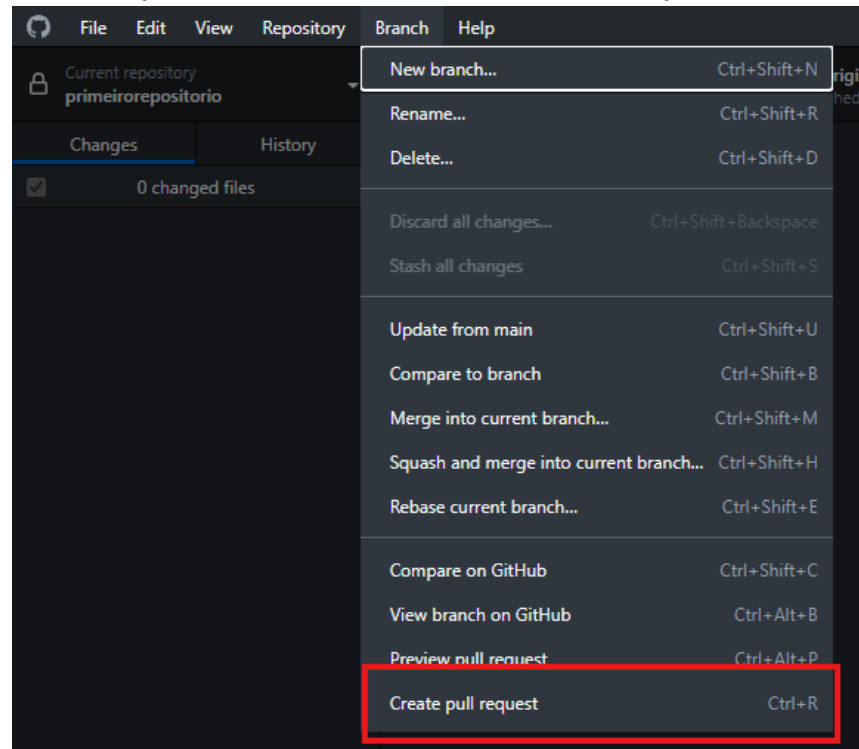
1. Clique no nome do branch atual no topo da tela.
2. Selecione o branch que deseja acessar.

11. COMO FAZER UM PULL REQUEST (PR)

Um Pull Request (PR) permite que você envie suas mudanças para o repositório principal, para revisão e aprovação.

Passos para Criar um Pull Request:

1. Faça todas as alterações e commits no seu branch.
2. No GitHub Desktop, vá até "Branch" > "Create Pull Request".



3. O GitHub Desktop abrirá a página do repositório no GitHub, onde você poderá revisar suas mudanças.
4. Preencha a descrição do que foi alterado e envie o Pull Request.

A screenshot of the GitHub web interface for creating a pull request. The 'Add a title' field contains the text 'Create testeBranch'. Below it, the 'Add a description' section has a large text area with the placeholder 'Add your description here...'. At the bottom right, a green button labeled 'Create pull request' is highlighted with a red rectangular box. The interface also shows a 'Write' tab and a 'Preview' tab, along with various formatting icons.

12. DICAS E BOAS PRÁTICAS NO GITHUB DESKTOP

- **Commits Frequentes:** Faça commits com frequência e com mensagens claras sobre o que foi alterado.
- **Use Branches:** Sempre que estiver implementando novas funcionalidades, crie um branch para não afetar o código principal.
- **Sincronize Regularmente:** Antes de começar a trabalhar, faça um pull para garantir que você esteja trabalhando na versão mais recente.
- **Faça Pull Requests:** Ao colaborar em equipe, utilize pull requests para revisar e integrar as alterações de outras pessoas.

13. FINALIZAÇÃO:

Com o GitHub Desktop, você tem todas as ferramentas necessárias para gerenciar repositórios de forma simples e eficiente, sem precisar usar a linha de comando. Ao seguir os passos descritos nesta apostila, você será capaz de criar, clonar, editar e colaborar em projetos no GitHub com facilidade.

Agora utilize com gosto o Github e o Github Desktop!!!