

SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL
TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS

CAIO VINICIUS AIRES DA SILVA

MANUAL PRÁTICO DO GITHUB: UTILIZAÇÃO ONLINE E DESKTOP
IA Utilizada: Copilot

JOINVILLE

2025

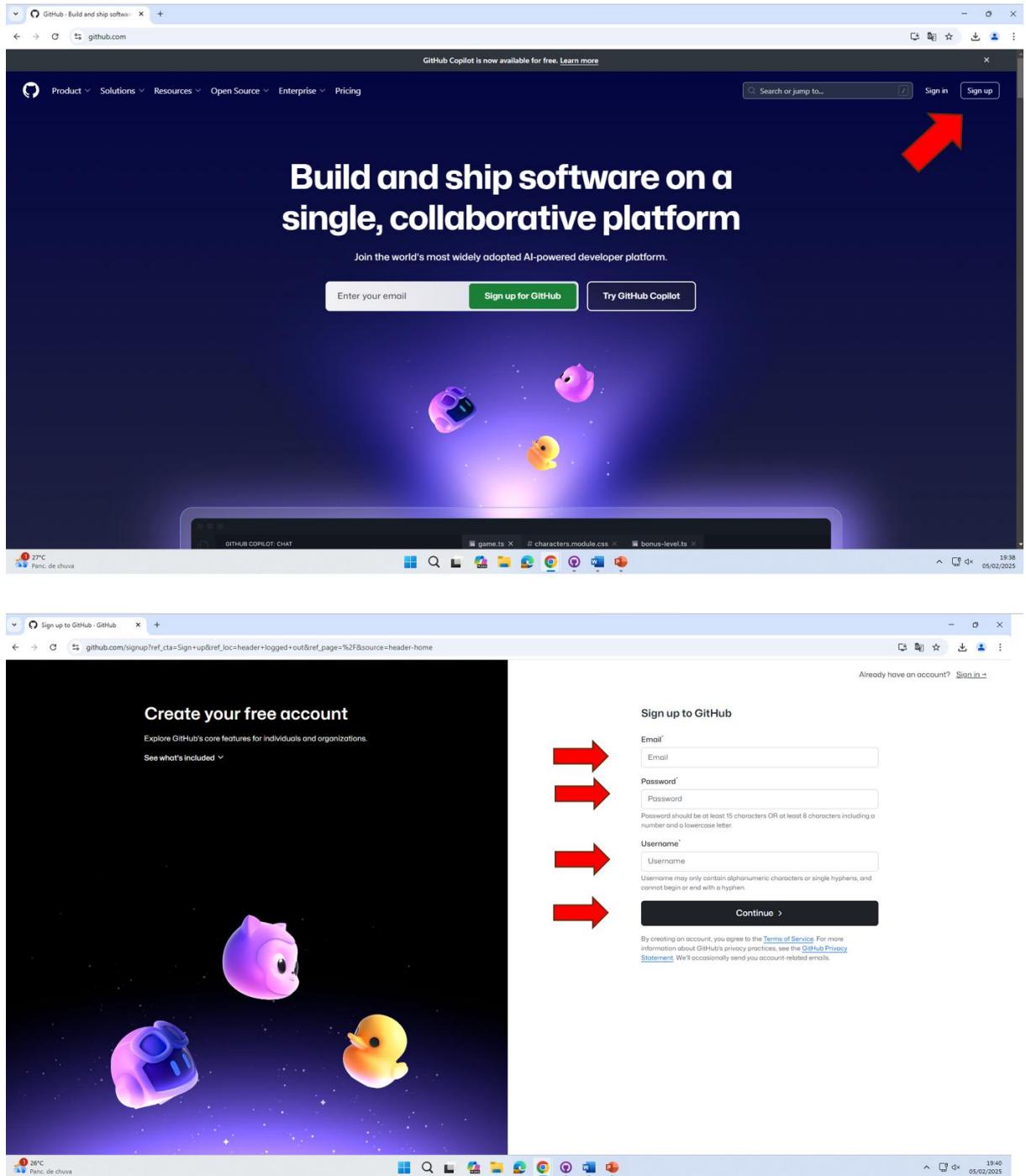
1. O QUE É O GITHUB?

O GitHub é uma plataforma para hospedagem de código-fonte e controle de versão usando o Git. Ele permite que você e outros trabalhem juntos em projetos de qualquer lugar.



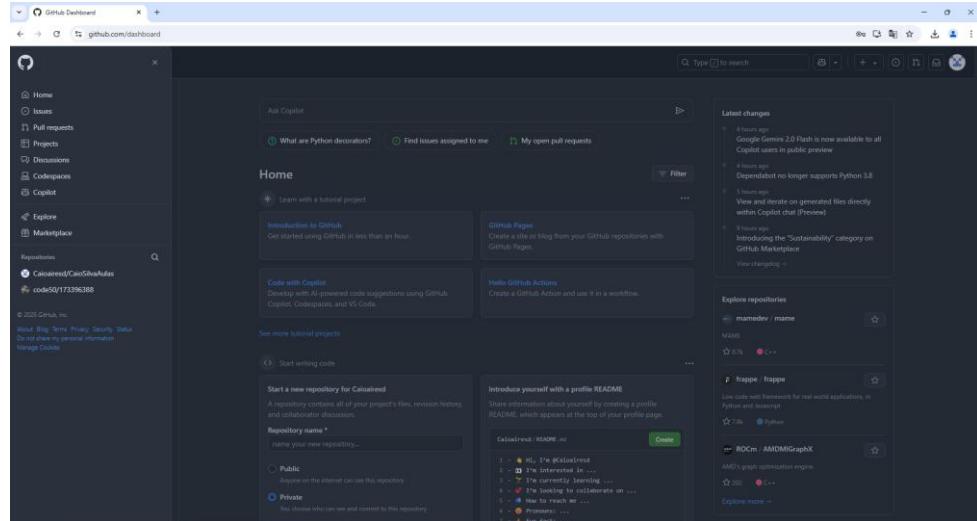
2. CRIANDO UMA CONTA NO GITHUB

1. Acesse github.com.
2. Clique em Sign Up.
3. Preencha os campos com suas informações e siga as instruções.



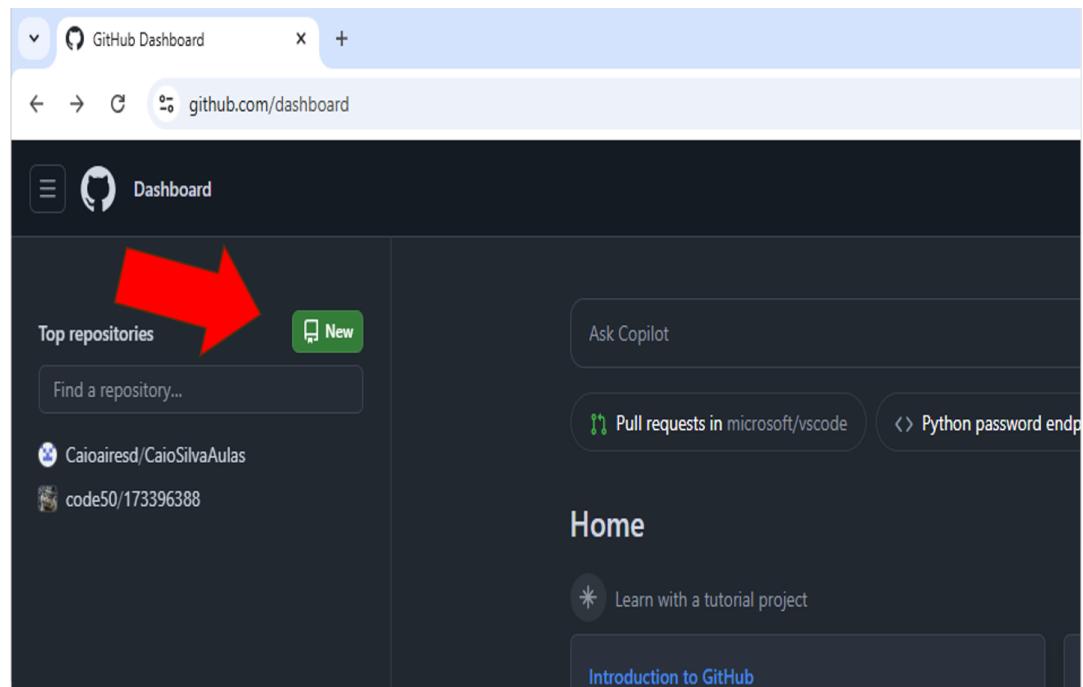
3. NAVEGANDO PELO GITHUB ONLINE

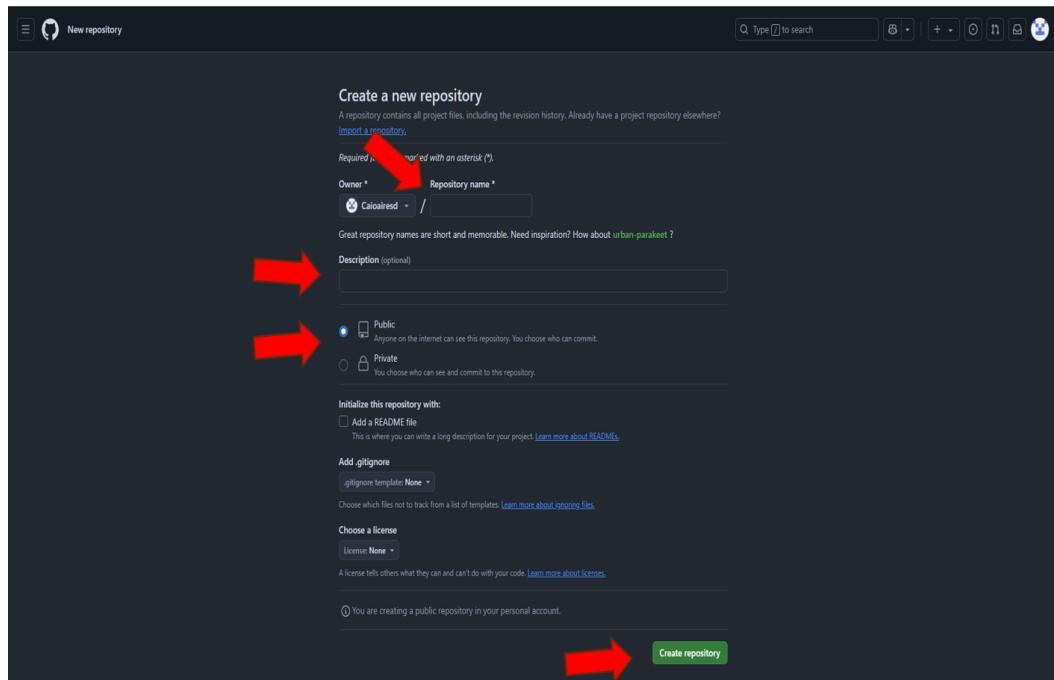
1. Após criar a conta, faça login.
2. No painel principal, você verá opções como Repositórios, Pull Requests, Issues, etc.



4. CRIANDO UM REPOSITÓRIO ONLINE

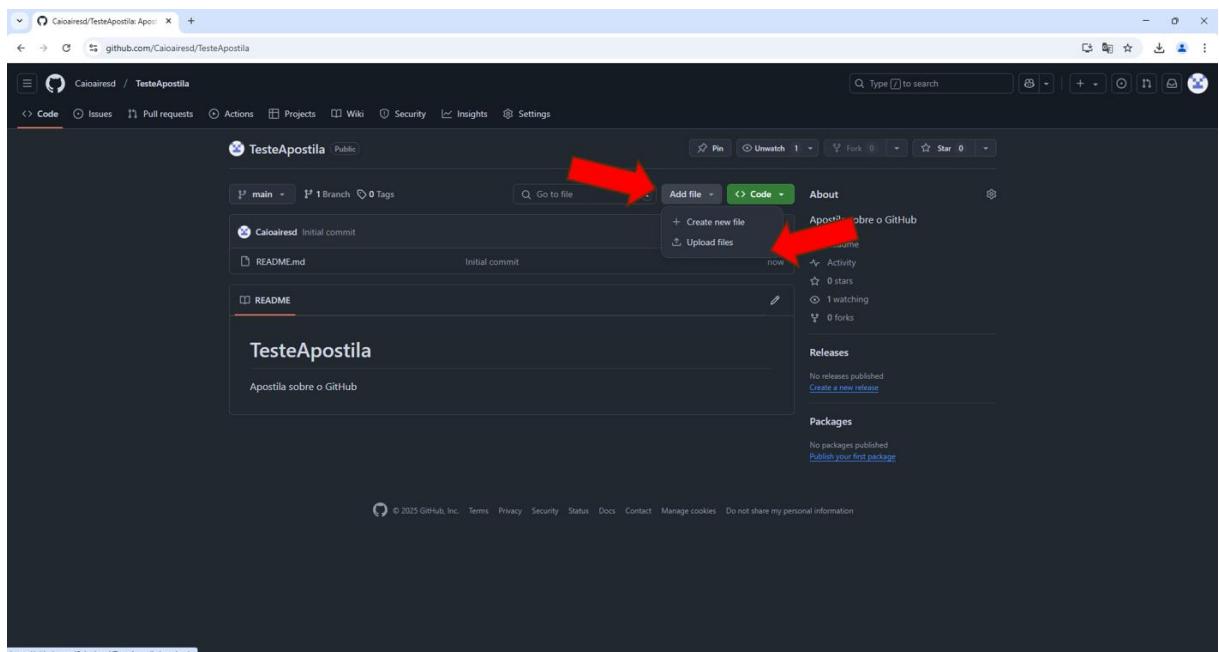
1. Clique no botão New no canto superior direito.
2. Dê um nome ao repositório e adicione uma descrição.
3. Escolha se o repositório será público ou privado.
4. Clique em Create Repository.





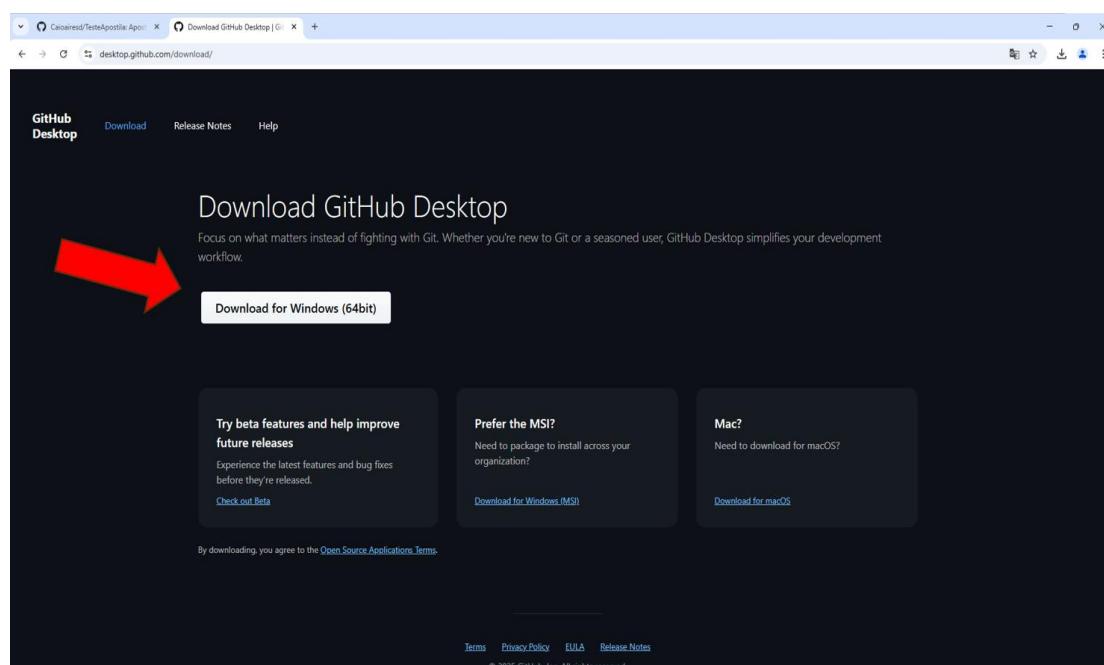
5. FAZENDO UPLOAD DE ARQUIVOS

1. Acesse o repositório que você criou.
2. Clique em Add file e depois em Upload files.
3. Selecione os arquivos do seu computador e clique em Commit changes.



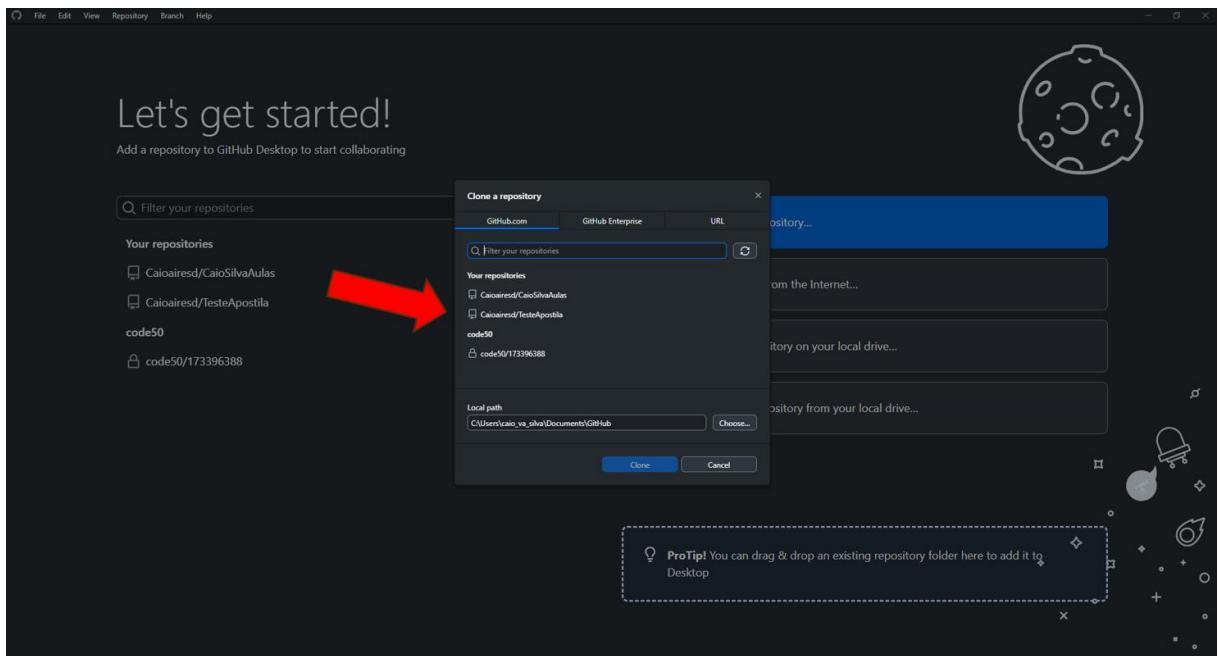
6. INTRODUÇÃO AO GITHUB DESKTOP

1. Baixe e instale o GitHub Desktop.
2. Abra o aplicativo e faça login com sua conta GitHub.



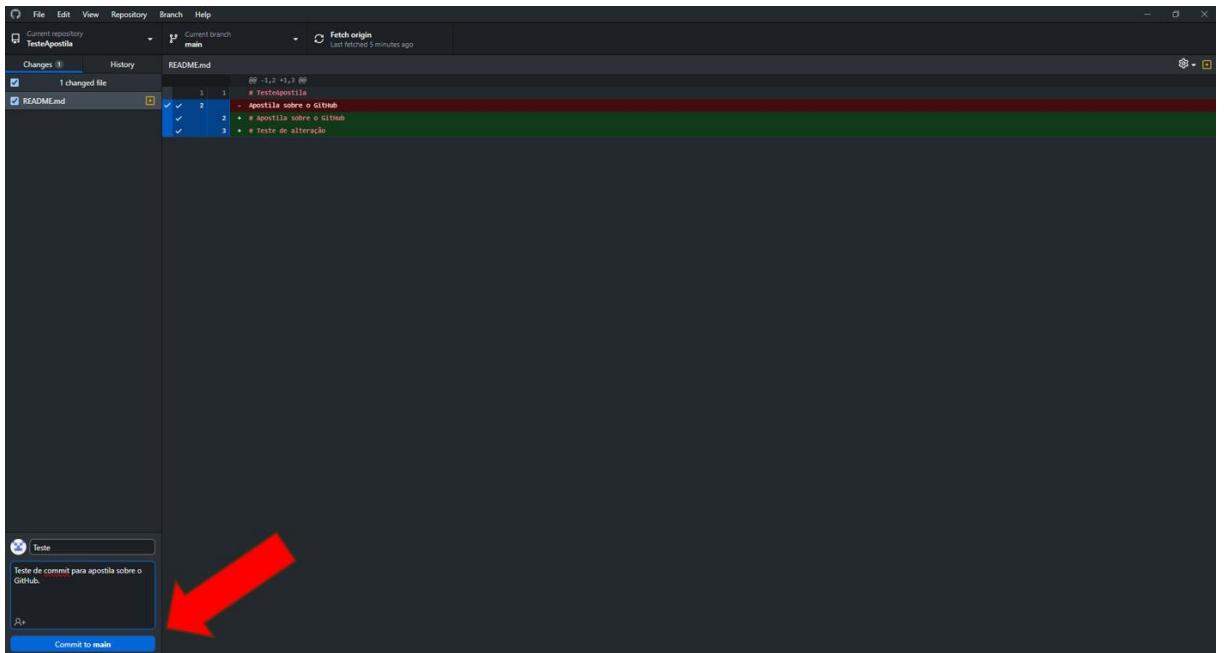
7. CLONANDO UM REPOSITÓRIO COM GITHUB DESKTOP

1. No GitHub Desktop, clique em File > Clone Repository.
2. Selecione o repositório que você deseja clonar.



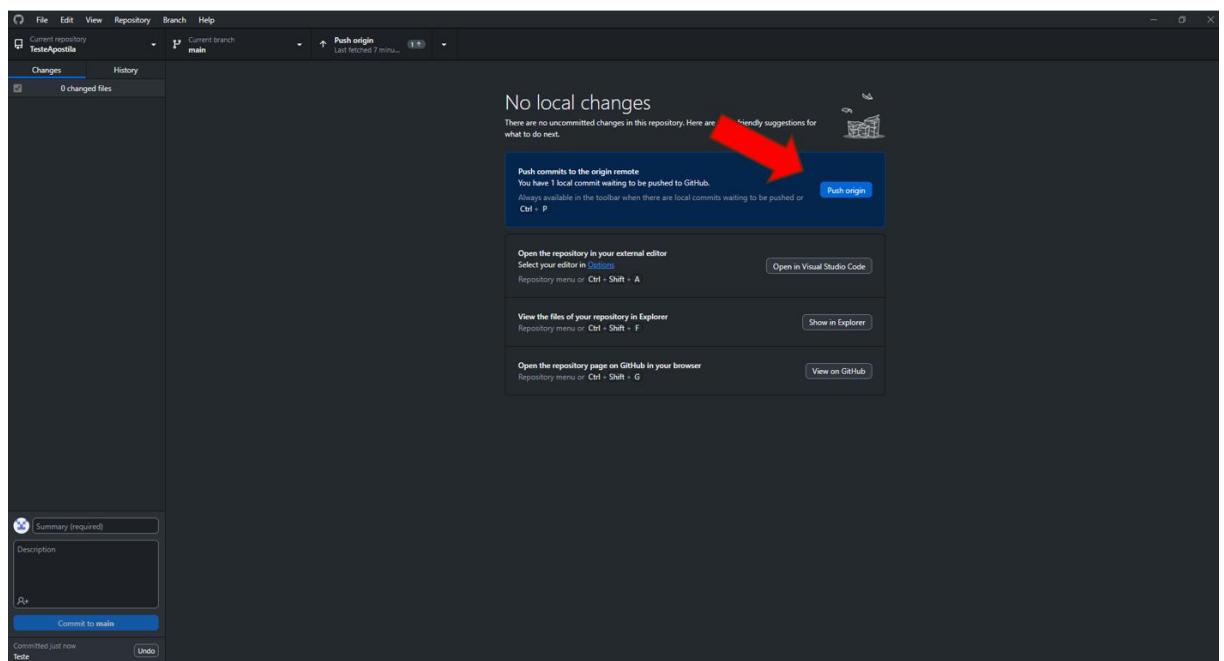
8. FAZENDO COMMIT DE MUDANÇAS COM GITHUB DESKTOP

1. Abra o repositório clonado no GitHub Desktop.
2. Faça alterações nos arquivos locais.
3. No GitHub Desktop, veja as mudanças, Adicione o título e a descrição das alterações realizadas e clique em Commit to main.



9. FAZENDO PUSH DE ALTERAÇÕES COM GITHUB DESKTOP

1. Após fazer o commit, clique em Push origin no GitHub Desktop para enviar as mudanças para o repositório online.

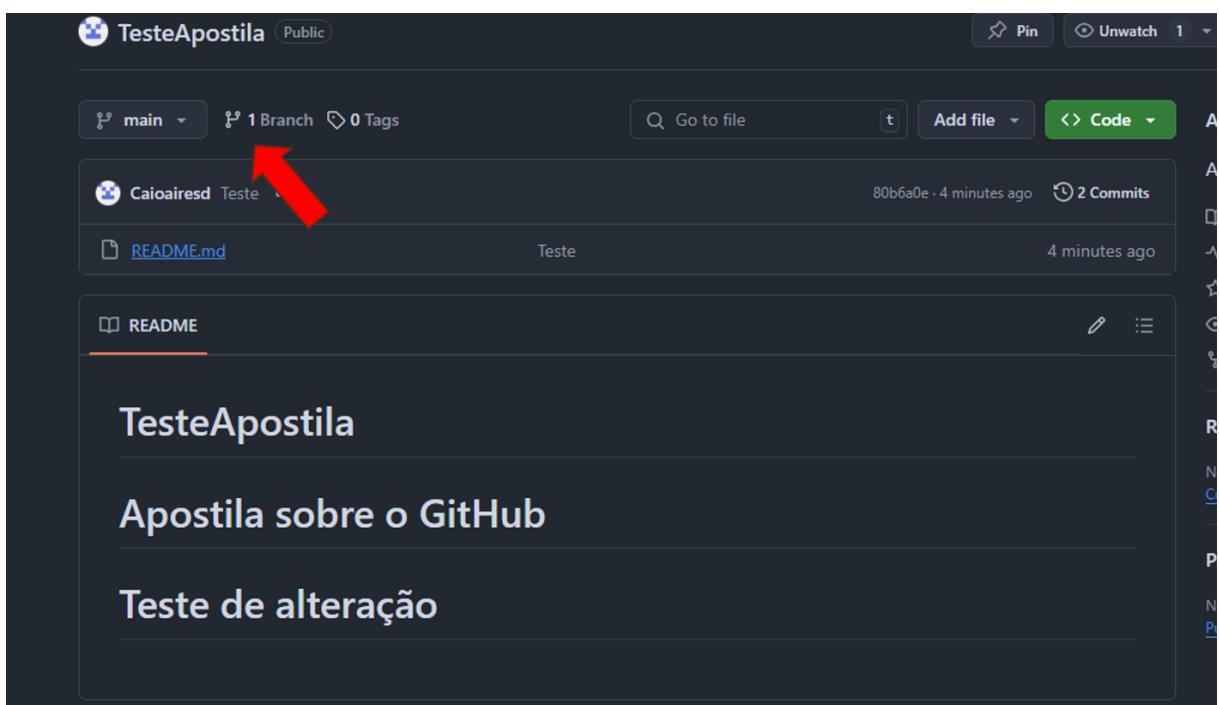


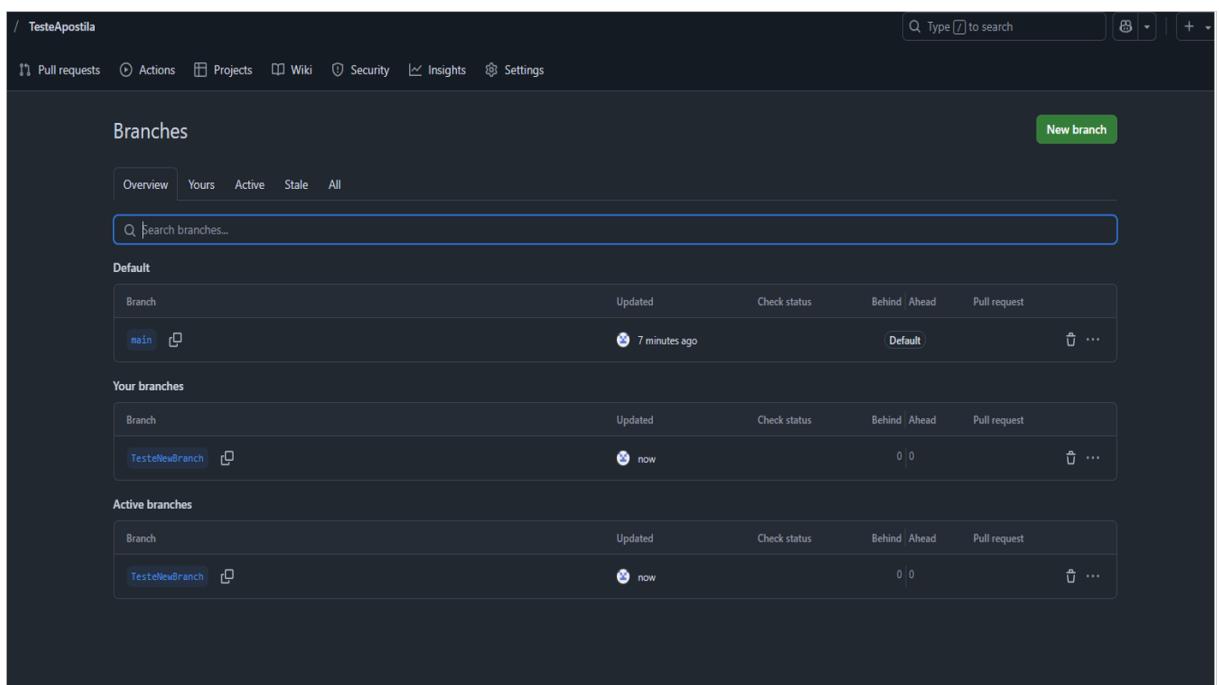
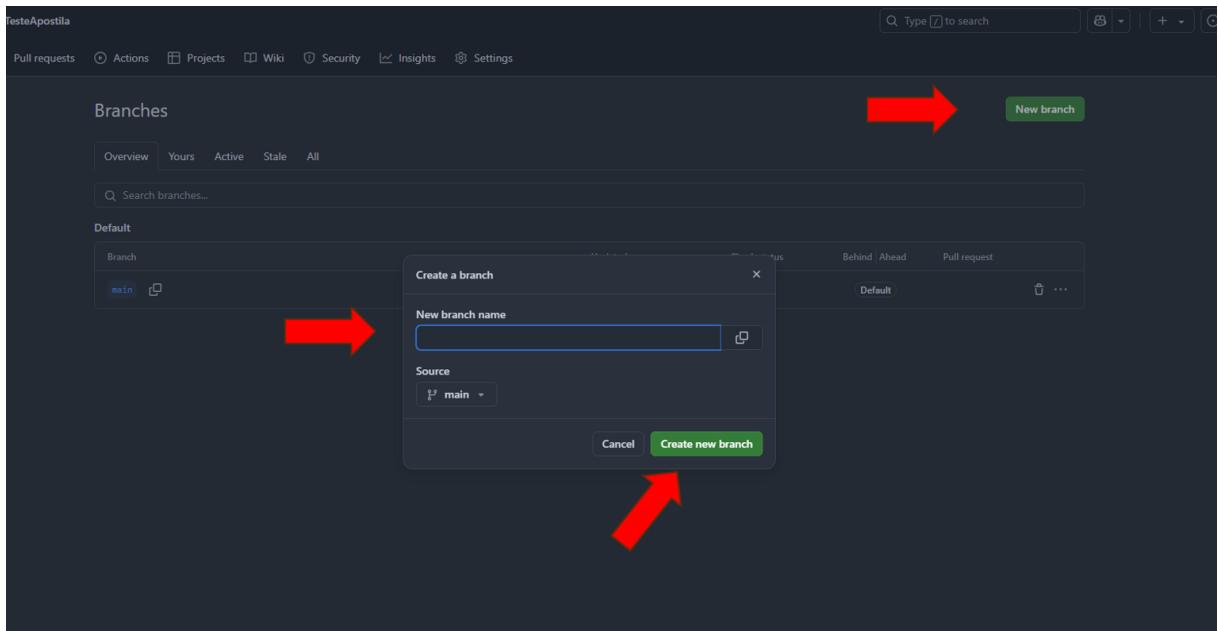
10. CRIANDO UMA BRANCH

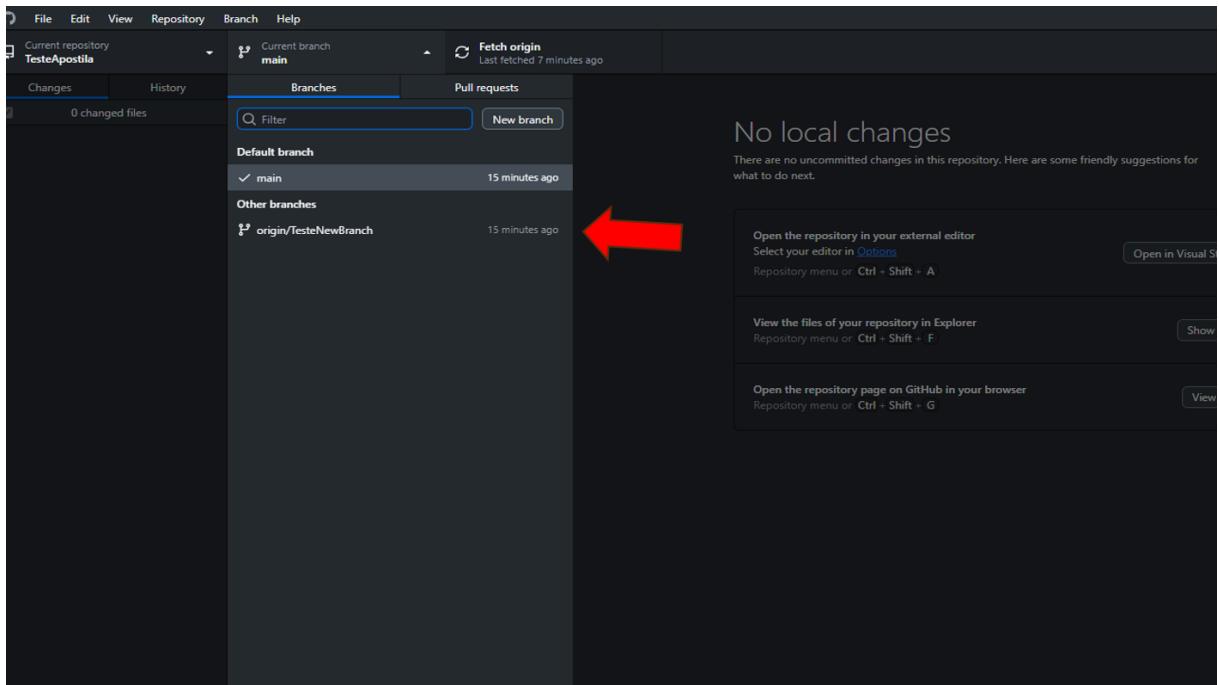
1. O que é uma Branch? Uma Branch (ou ramificação) é uma versão paralela do seu projeto que permite que você trabalhe em novas funcionalidades,

correções de bugs ou experimentos de forma isolada da Branch principal (geralmente chamada de "main" ou "master"). Ao finalizar as alterações, você pode mesclá-las de volta na Branch principal sem interferir no código já existente.

2. No repositório online, clique em Branch: main e depois em New branch.
3. Dê um nome para a nova Branch e clique em Create branch.
4. No GitHub Desktop escolha a Branch desejada, e realize as implementações desejadas em seu código.

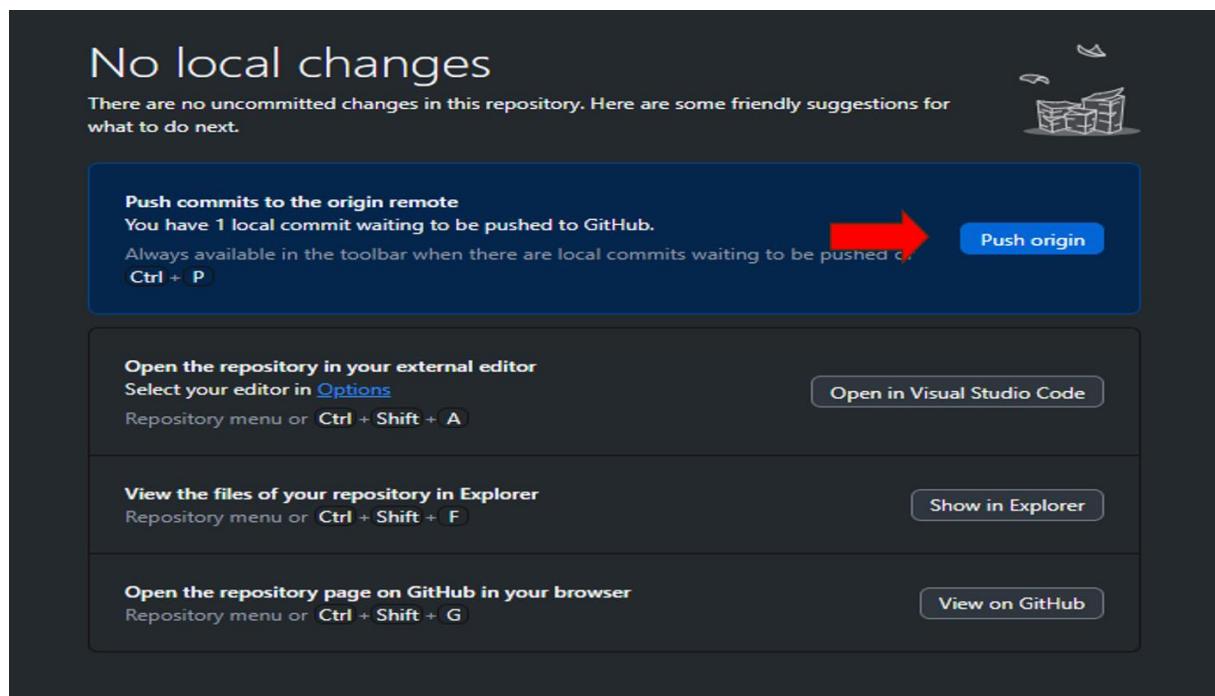


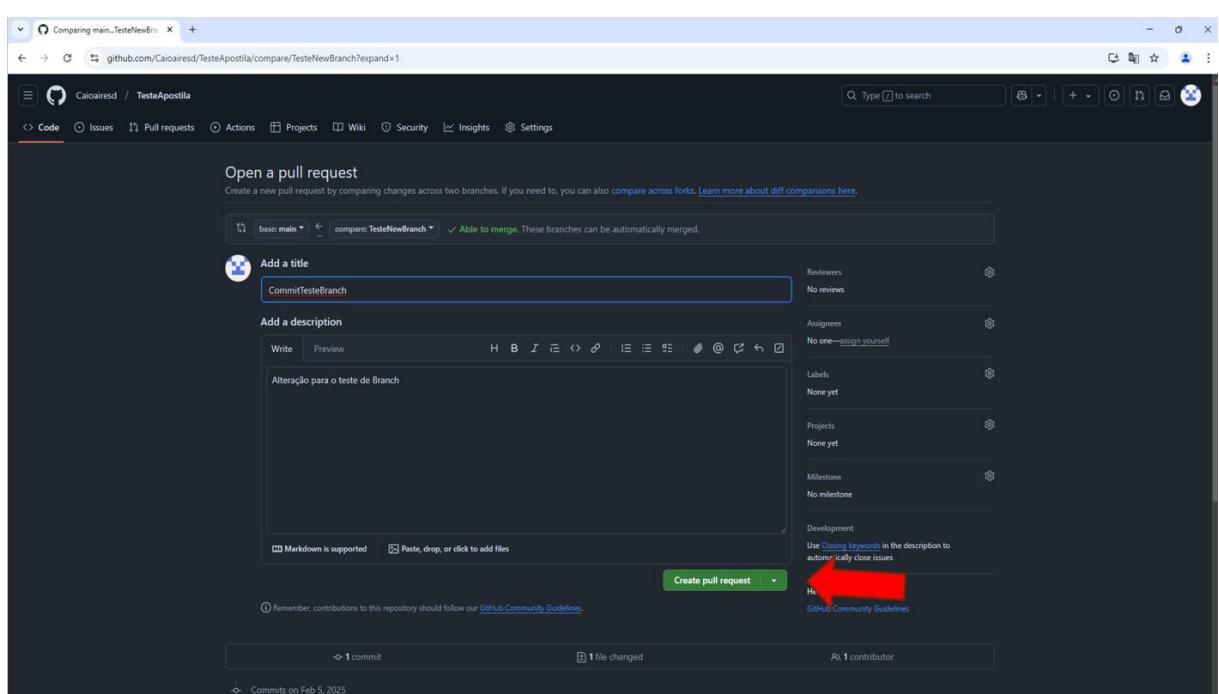
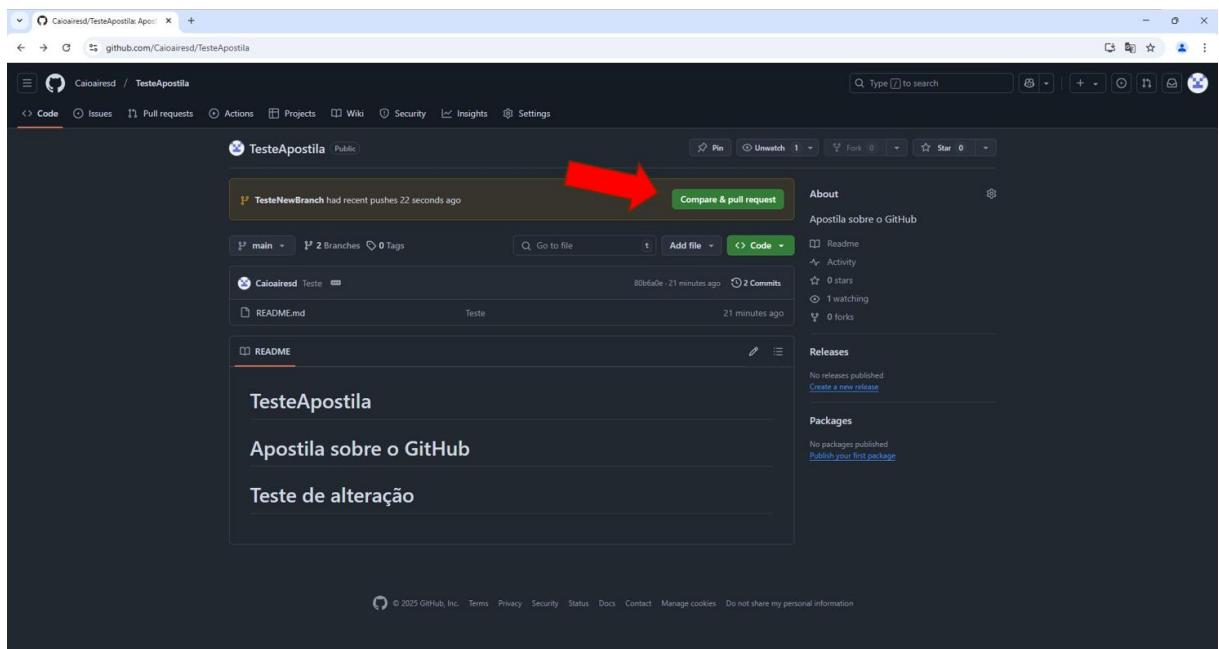




11. FAZENDO UM PULL REQUEST

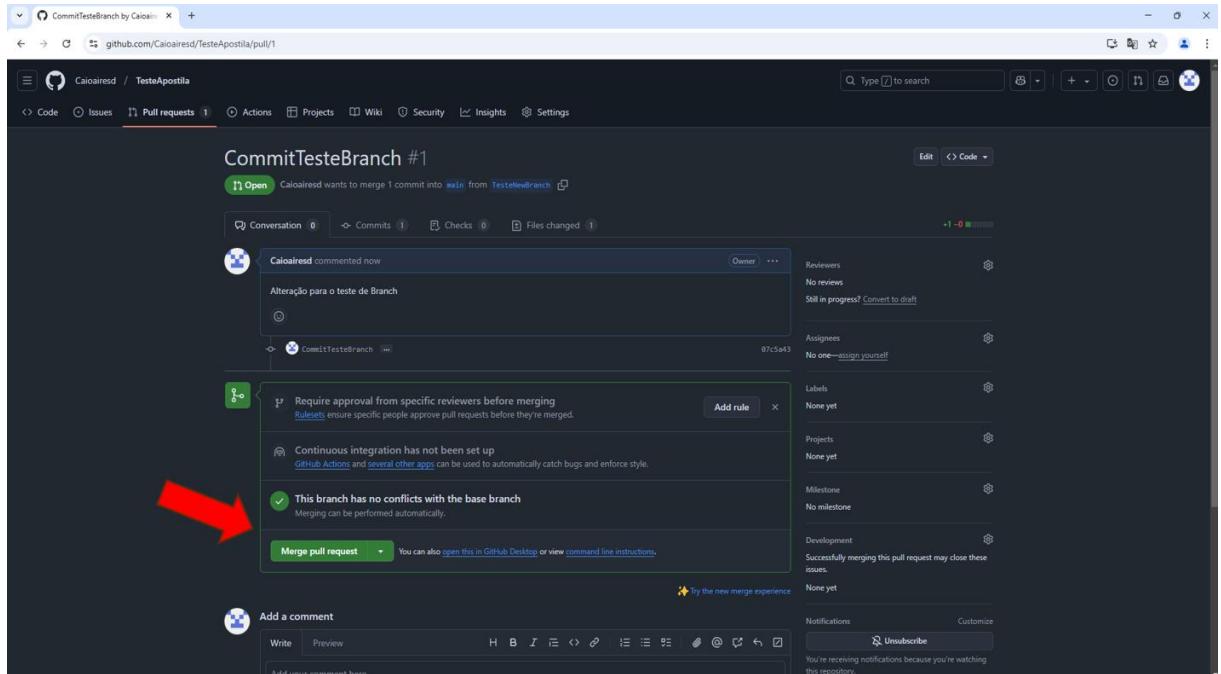
1. Depois de fazer mudanças na nova Branch, realizar o commit relacionado a essa mudança, e realize o Push Origin no GitHub Desktop.
2. No GitHub Online clique em New pull request, compare as mudanças e clique em Create pull request.





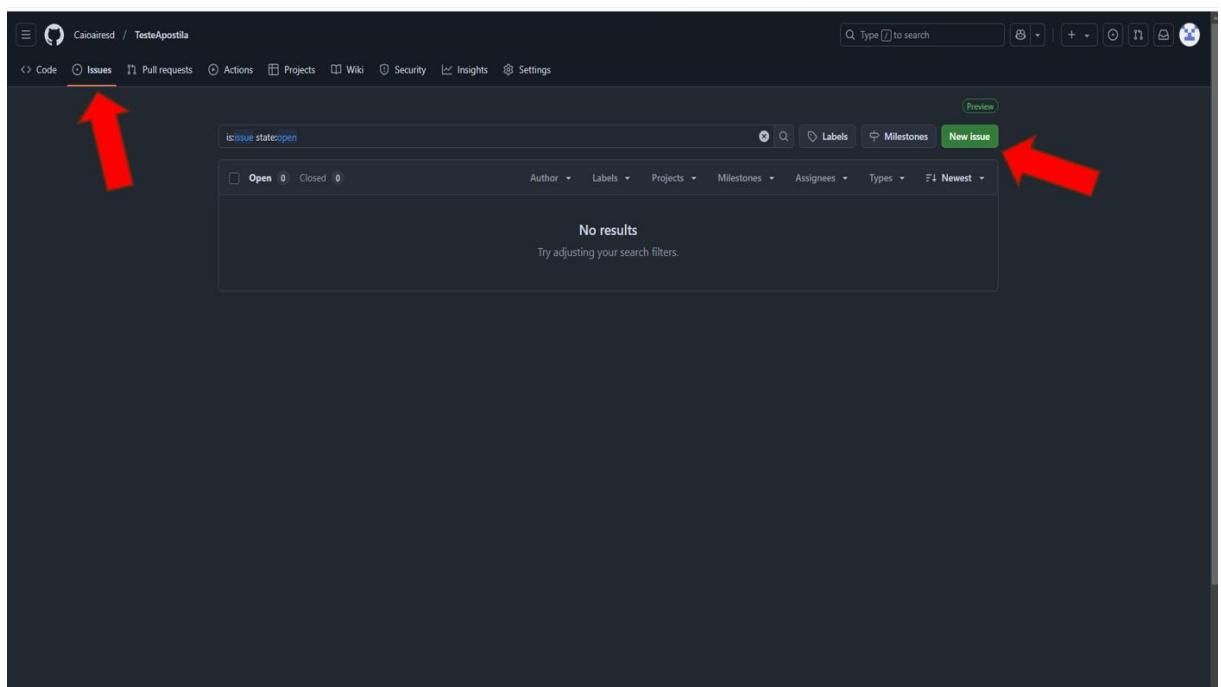
12. REVIEW E MERGE DE PULL REQUESTS

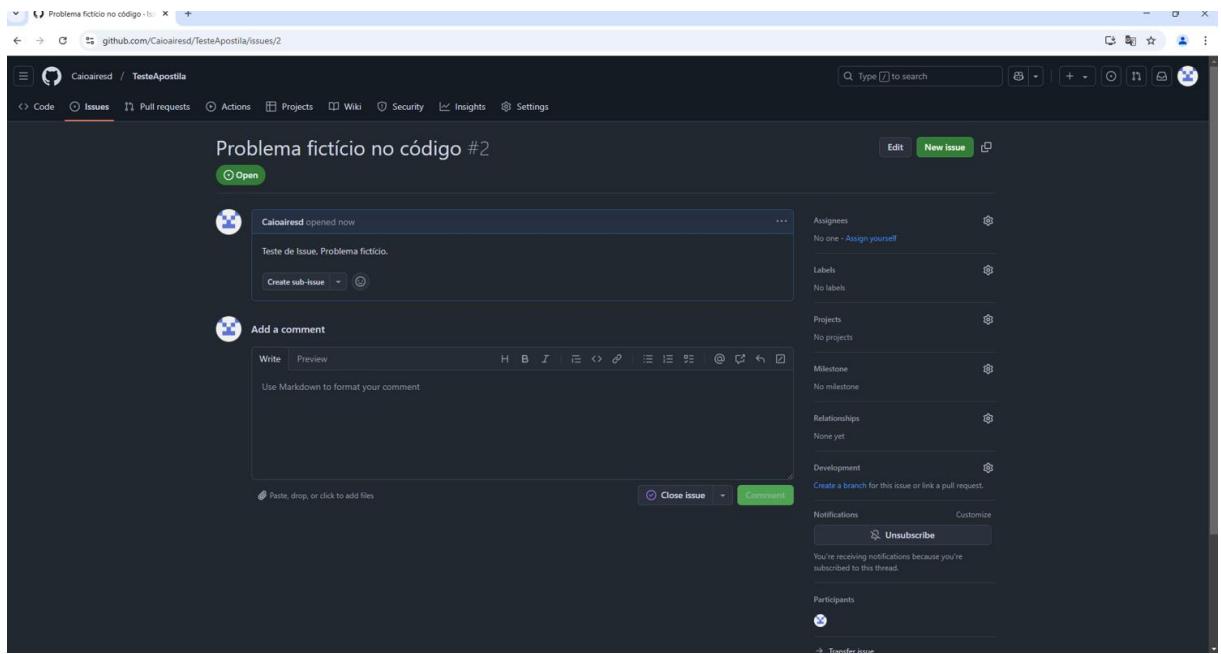
1. Os colaboradores podem revisar o pull request.
2. Após a revisão, clique em Merge pull request para integrar as mudanças à branch principal.



13. CONFIGURANDO ISSUES

1. No repositório online, vá para a aba Issues.
2. Clique em New issue para criar um novo problema ou sugestão.





16. COLABORANDO EM PROJETOS

1. Adicione colaboradores no repositório, indo para a aba Settings e depois Manage access.
2. Convide os usuários através do nome de usuário ou e-mail.

The screenshot shows the 'General' settings page for a repository named 'TesteApostila'. The sidebar on the left lists various settings categories: General, Access, Collaborators (which is highlighted with a red arrow), Moderation options, Code and automation, Branches, Tags, Rules, Actions, Webhooks, Environments, Codespaces, Pages, Security, Code security, Deploy keys, Secrets and variables, Integrations, GitHub Apps, and Email notifications. The main content area displays the repository name 'TesteApostila', a 'Template repository' checkbox, a 'Require contributors to sign off on web-based commits' checkbox, the 'Default branch' set to 'main', a 'Social preview' section, and a 'Features' section with a 'Wikis' checkbox. A red arrow also points to the 'General' tab at the top of the main content area.

The screenshot shows the 'Who has access' section of a GitHub repository's settings. On the left, a sidebar lists various repository management options like General, Access, Collaborators, and Code and automation. The 'Collaborators' option is currently selected. The main area displays two sections: 'PUBLIC REPOSITORY' and 'DIRECT ACCESS'. Under 'PUBLIC REPOSITORY', it says 'This repository is public and visible to anyone.' and 'Manage'. Under 'DIRECT ACCESS', it says '0 collaborators have access to this repository. Only you can contribute to this repository.' and 'Manage'. Below this, a large box titled 'Manage access' contains a message 'You haven't invited any collaborators yet' with a red arrow pointing to a green 'Add people' button.