

FIAP GRADUAÇÃO

**DISCIPLINA: PROJETO DE SISTEMAS APLICADO AS MELHORES PRÁTICAS EM
QUALIDADE DE SOFTWARE E GOVERNANÇA DE TI**

AULA:

**4 – EXEMPLIFICAÇÃO DE COMO FERRAMENTAS E PROCESSOS IMPACTAM A
QUALIDADE E GOVERNANÇA – GIT HUB**

PROFESSOR:

RENATO JARDIM PARDOCCI

PROFRENATO.PARDOCCI@FIAP.COM.BR

AGENDA DA AULA

- ✓ Exemplificação do impacto de processos de trabalho e ferramentas sobre a Governança e a Qualidade
- ✓ Controle de conteúdo e versões



VAMOS REFORÇAR OS CONCEITOS DA GOVERNANÇA COM UM JOGO!

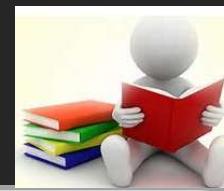
Kahoot!

Introdução à Governança

by RenatoJP 13 seconds ago

Quiz 13 questions Private

ESTUDO DE CASO SIMULADO



Para atender a Governança de TI na empresa de D e melhorar a condução dos projetos de software, você escolheu criar um processo de controle de versões de documentos e fontes de projetos e selecionou a ferramenta (recurso) GIT para implementar essa mudança.

Lembrando o nosso problema... a GD (Gerência de Desenvolvimento) da empresa de D está sofrendo bastante com o controle de fontes de aplicação e de documentação de projeto.

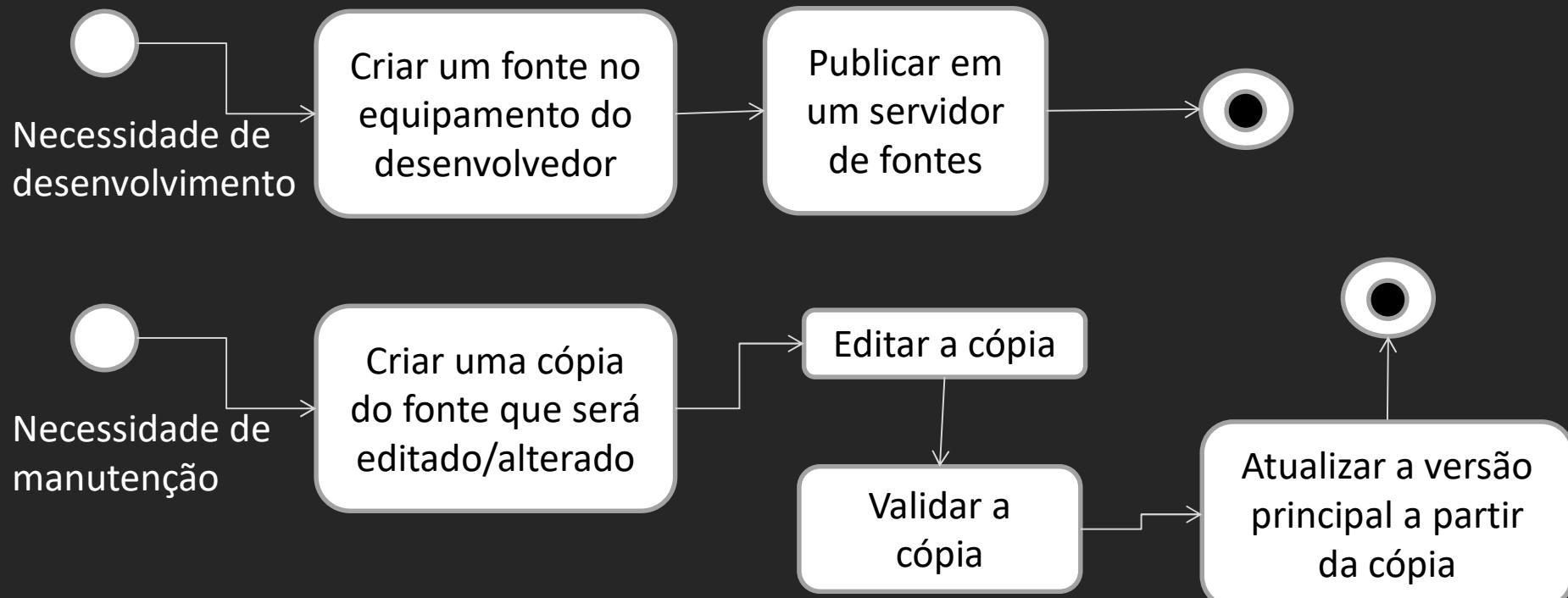
Cada grupo de projeto acaba definindo uma forma de trabalhar é não é incomum a sobreposição de fontes de forma errada, ou atualização de fonte em versão que não a última, gerando problemas em builds e releases.

D gostaria que a sua empresa tivesse uma solução única para controlar fonte de programas e documentos, que permitisse a colaboração de desenvolvedores sobre um mesmo material, sem perder controle sobre a ótima versa oficial.

Um processo foi desenhado por C e ele preparou um treinamento sobre GIT que você vai realizar agora!

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Vamos trabalhar com a seguinte proposta de processo de gestão de mudança no desenvolvimento de software:



ASSISTA OS VÍDEOS NO CANAL DO PROFESSOR



<https://youtu.be/t1E-cbB4gFY>

A screenshot of a Windows desktop. On the left, a file explorer window shows a folder structure under 'RunMIDIA > Google Drive'. The contents include 'Metodologia' (modified 02/02/2016), 'Docencia (I)' (modified 14/03/2016), and 'Recibos 2015' (modified 11/07/2016). To the right of the file explorer is a video player window displaying a video of a man with a shaved head and a dark shirt, wearing headphones, speaking. The video player has standard controls at the bottom.

Conteúdo didático complementar - Controle de Versão de Documentos



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Em seguida, vamos vemos adotar uma solução na nuvem como ferramenta para administrar os fontes – o **GIT** e **GIT HUB**:



Existem outras soluções, hoje menos populares que o GIT mas que funcionam bem na gestão de fontes e versões...



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Através da ferramenta de controle de versões de códigos e/ou documentos de software, é possível evitar os problemas de se trabalhar com a fonte errada em um determinado ponto do projeto, facilitando a colaboração no projeto.



Repositório de Fontes

```
socket.error: [Errno 35] Resource temporarily unavailable
socket.error: [Errno 35] Resource temporarily unavailable
print "ncfiles: socket error (%s) for host %s (%s) & port %s\n"
for h3 in page.findAll("h3"):
    value = (h3.contents[0])
    if value != "Afdeeling":
        print >> txt, value
import codecs
f = codecs.open("alle.txt", "r", encoding="utf-8")
text = f.read()
f.close()
# open the file again for writing
f = codecs.open("alle.txt", "w", encoding="utf-8")
f.write(value+"\n")
```

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Trabalhando com GIT, os arquivos fontes serão organizados em:

Cópia Master

Contém os arquivos na versão estável, que podem ser usados por outros desenvolvedores na integração de componentes ou com outros sistemas, ou podem ser usados para gerar um pacote de versão final do produto.

Cópia Branch

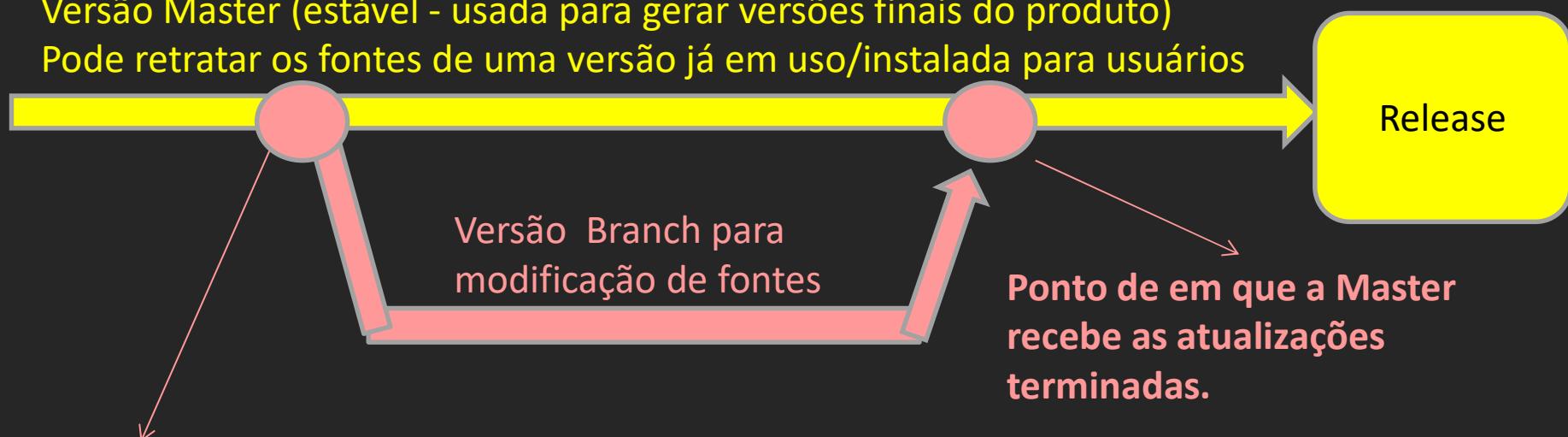
Contém os arquivos em uma versão de manutenção/atualização que não estão estáveis e não podem ser usados para gerar uma versão final do produto.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

O funcionamento da Branch.

Versão Master (estável - usada para gerar versões finais do produto)

Pode retratar os fontes de uma versão já em uso/installada para usuários

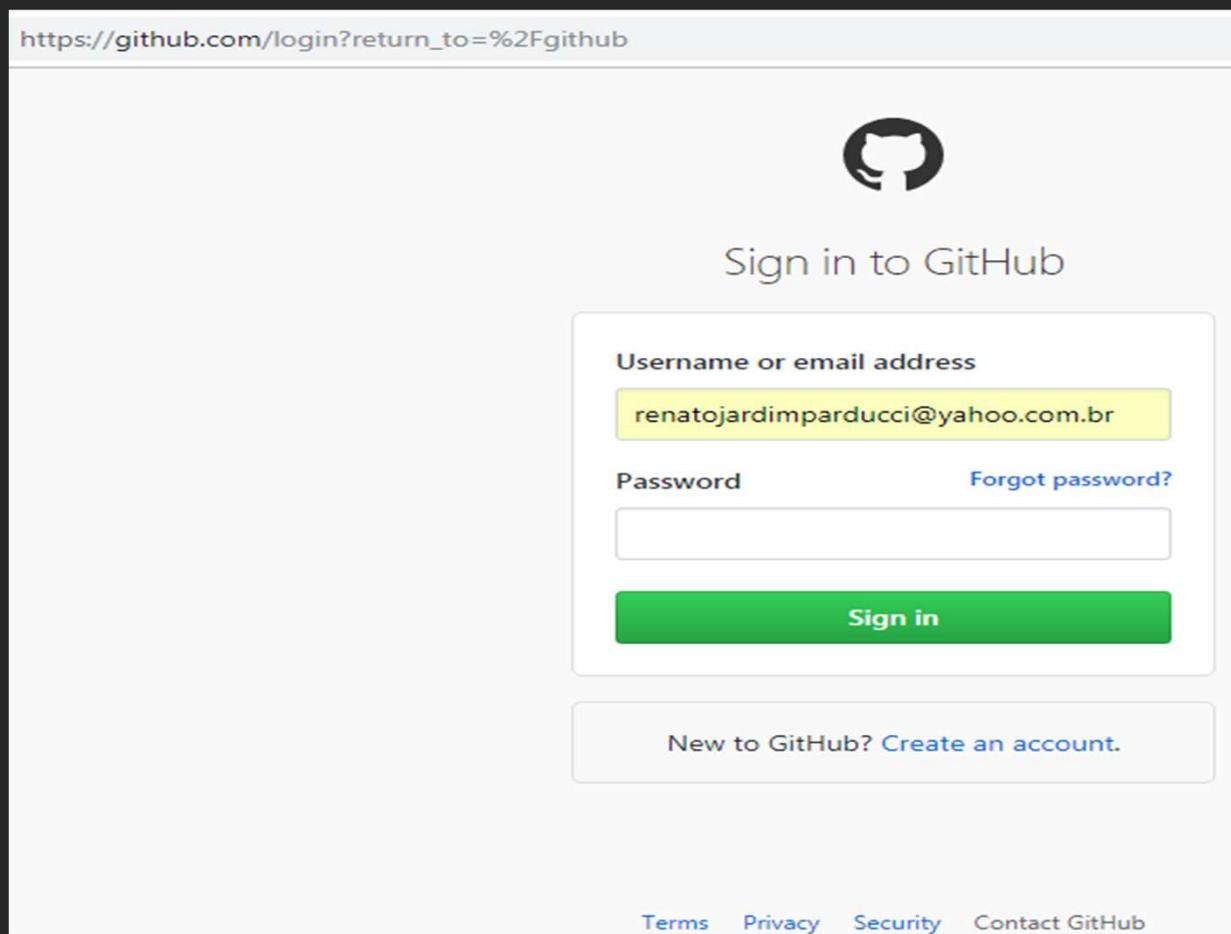


Ponto de necessidade de manutenção para:

- Corrigir o software/eliminar BUGs (manutenção CORRETIVA);
- Adaptar o software para novas regras de negócio (manutenção ADAPTATIVA);
- Prevenir contra possíveis problemas futuros (manutenção PREVENTIVA);
- Alcançar a perfeição na experiência do usuário (manutenção PREFECTIVA)

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Crie sua conta no GIT HUB.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Faça o login para ver se está tudo Ok.

The screenshot shows a GitHub profile page with the following details:

- Profile Picture:** A black cat icon.
- Name:** GitHub
- Slogan:** How people build software.
- Location:** San Francisco, CA
- Website:** <https://github.com/about>
- Contact:** support@github.com
- Verification:** Verified
- Repositories:** 289
- People:** 206
- Projects:** 0
- Pinned repositories:**
 - fetch**: A window.fetch JavaScript polyfill. (JavaScript, 20.8k stars, 1.9k forks)
 - hub**: hub helps you win at git. (Go, 14.1k stars, 1.4k forks)
 - training-kit**: Open source on demand courses and cheat sheets for Git and GitHub (HTML, 1.9k stars, 1.9k forks)
 - choosealicense.com**: A site to provide non-judgmental guidance on choosing a license for your open source project. (Ruby, 1.4k stars, 457 forks)
 - scientist**: A Ruby library for carefully refactoring critical paths. (Ruby, 4.7k stars, 200 forks)
 - gh-ost**: GitHub's Online Schema Migrations for MySQL. (Go, 5.9k stars, 453 forks)

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Logado no GITHUB, acesse a área de repositórios.

The screenshot shows the GitHub user interface. At the top, there's a navigation bar with links for Pull requests, Issues, Marketplace, and Explore. On the right side, a user dropdown menu is open, showing options like Signed in as, Your profile, and Your repositories. A red arrow points to the 'Your repositories' option, which is highlighted with a red box. Below the dropdown, the main GitHub page displays the user's profile picture, name (RenatoJardimParducci), location (San Francisco, CA), website (https://github.com/about), email (support@github.com), and a verified badge. It also shows statistics for Repositories (289), People (206), and Projects (0). The 'Repositories' section lists three pinned repositories: 'fetch', 'hub', and 'training-kit'. The bottom of the page has a pink footer bar with the URL www.fiap.com.br and the name Prof. Renato Jardim Parducci.

Signed in as
RenatoJardimParducci

Your profile

Your repositories

Your stars

Your gists

Help

Settings

Sign out

GitHub

How people build software.

San Francisco, CA https://github.com/about support@github.com Verified

Repositories 289 People 206 Projects 0

Pinned repositories

fetch hub training-kit

A window.fetch JavaScript polyfill. hub helps you win at git. Open source on demand courses and cheat sheets

www.fiap.com.br Prof. Renato Jardim Parducci

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Seus repositórios de projetos de software aparecerão, caso você já tenha catalogado algum.

The screenshot shows a GitHub profile interface. At the top, there is a blue banner with the text "ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you." and a green "Edit profile" button. Below the banner, the profile navigation bar includes "Overview", "Repositories 12", "Stars 0", "Followers 0", and "Following 0". There are also search and filter options: "Find a repository...", "Type: All ▾", "Language: All ▾", and a green "New" button. The main content area displays three repository cards: "1TDSJ" (updated on 26 Apr 2017), "RepExemplo" (updated on 17 Jan 2017), and "ExemploTDSS". Each card has a horizontal progress bar below it.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Seus repositórios de projetos de software aparecerão, caso você já tenha catalogado algum.

The screenshot shows a GitHub profile interface. At the top, there is a blue banner with the text "ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you." and a green "Edit profile" button. Below the banner, the profile navigation bar includes "Overview", "Repositories 12", "Stars 0", "Followers 0", and "Following 0". There are also search and filter options: "Find a repository...", "Type: All ▾", "Language: All ▾", and a green "New" button. The main content area displays three repository cards: "1TDSJ" (updated on 26 Apr 2017), "RepExemplo" (updated on 17 Jan 2017), and "ExemploTDSS". Each card has a horizontal progress bar below it.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Crie um repositório.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner / Repository name 

Great repository names are short and memorable. Need inspiration? How about sturdy-spoon.

Description (optional)

 Public
Anyone can see this repository. You choose who can commit.

 Private
You choose who can see and commit to this repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None | 

Create repository

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Entenda a tela do GitHub.

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The top navigation bar includes 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below the bar are links for 'Code', 'Issues (0)', 'Pull requests (0)', 'Projects (0)', 'Wiki', 'Insights', and 'Settings'. A note says 'No description, website, or topics provided.' with an 'Edit' button. The repository summary shows '1 commit', '1 branch', '0 releases', and '1 contributor'. Annotations with red circles and arrows point to specific elements:

- Annotation 1 points to the 'Branch: master ▾' dropdown menu.
- Annotation 2 points to the 'README.md' file listed in the repository files.
- Annotation 3 points to the 'Upload files' button in the main repository actions area.

Below the summary, there's a list of files:

- RenatoJardimParducci Initial commit (latest commit)
- README.md (Initial commit, a minute ago)
- README.md (a minute ago)

At the bottom left, it says 'TesteGITHUB'.

Annotations with numbers 1, 2, and 3 are located at the bottom right of the screenshot, corresponding to the numbered callouts below:

- 1 Seleciona a área/cópia de fontes para trabalho
- 2 Nomes dos arquivos que constam na área
- 3 Usado para carregar arquivos fonte no GIT

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode criar pastas para separar tipos de arquivos

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The page includes navigation links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. A red callout box with the text 'Crie um nome para a pasta/nome de um arquivo Readme que será criado' points to the 'Create new file' button in the main repository actions area. The repository summary shows 1 commit, 1 branch, 0 releases, and 1 contributor. Below the summary, there are sections for README.md files and a commit history. The commit history lists an 'Initial commit' by RenatoJardimParducci. At the bottom of the page, the repository name 'TesteGITHUB' is displayed.

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

No description, website, or topics provided.

Manage topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request

Create new file Upload files Find file Clone or download

RenatoJardimParducci Initial commit Latest commit 7296d6c a minute ago

README.md Initial commit a minute ago

README.md

Crie um nome para a pasta/nome de um arquivo Readme que será criado

TesteGITHUB

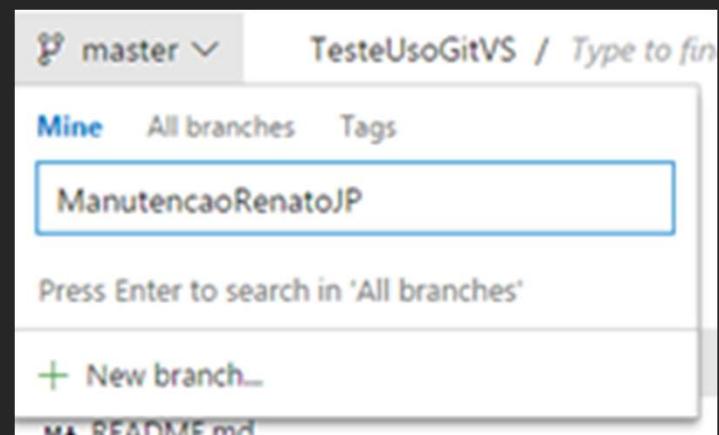
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Para incluir novos fontes ou alterar fontes publicados no GIT, crie uma Branch (cópia de desvio) da área Master.

A Master deve conter apenas os códigos fonte estáveis, que podem ser usados por outros desenvolvedores.

A Branch é uma réplica da Master para que um ou mais programadores façam alterações nos programas e depois republiquem os arquivos atualizados na Master.

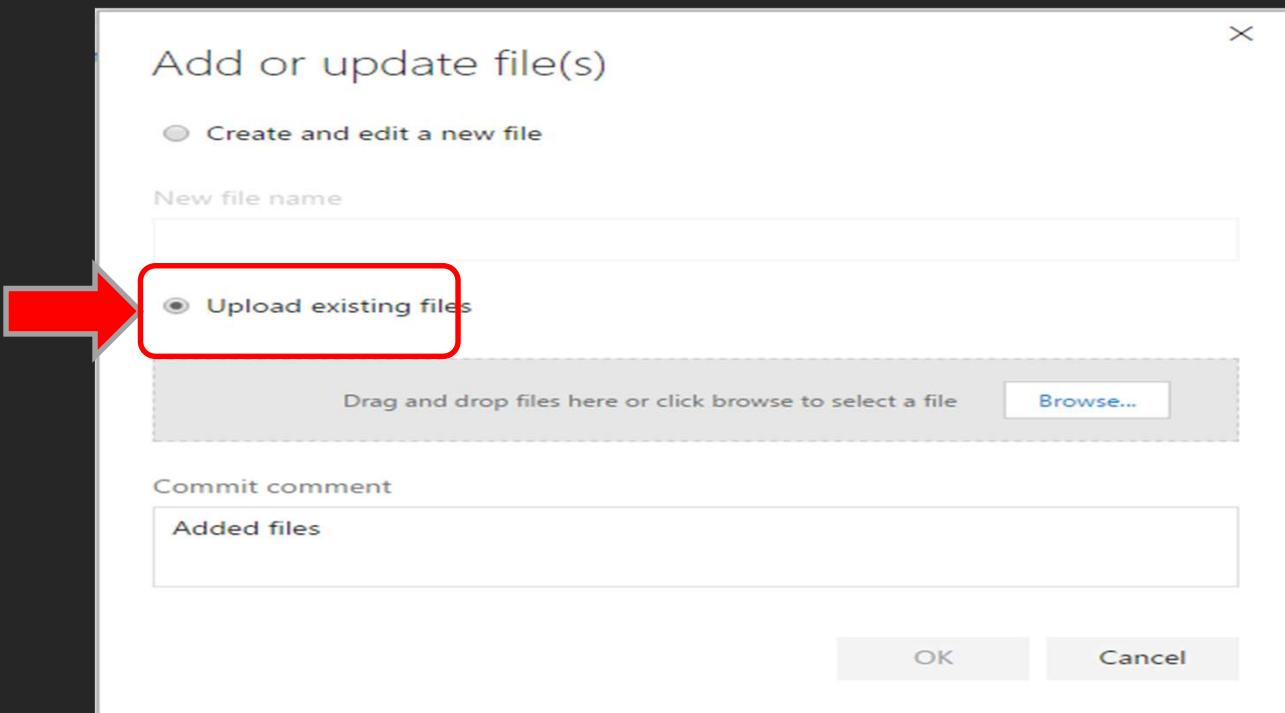
Crie uma Branch a partir da Master com o nome que desejar.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Acesse o link de Upload para subir para o GitHub um arquivo do seu computador.

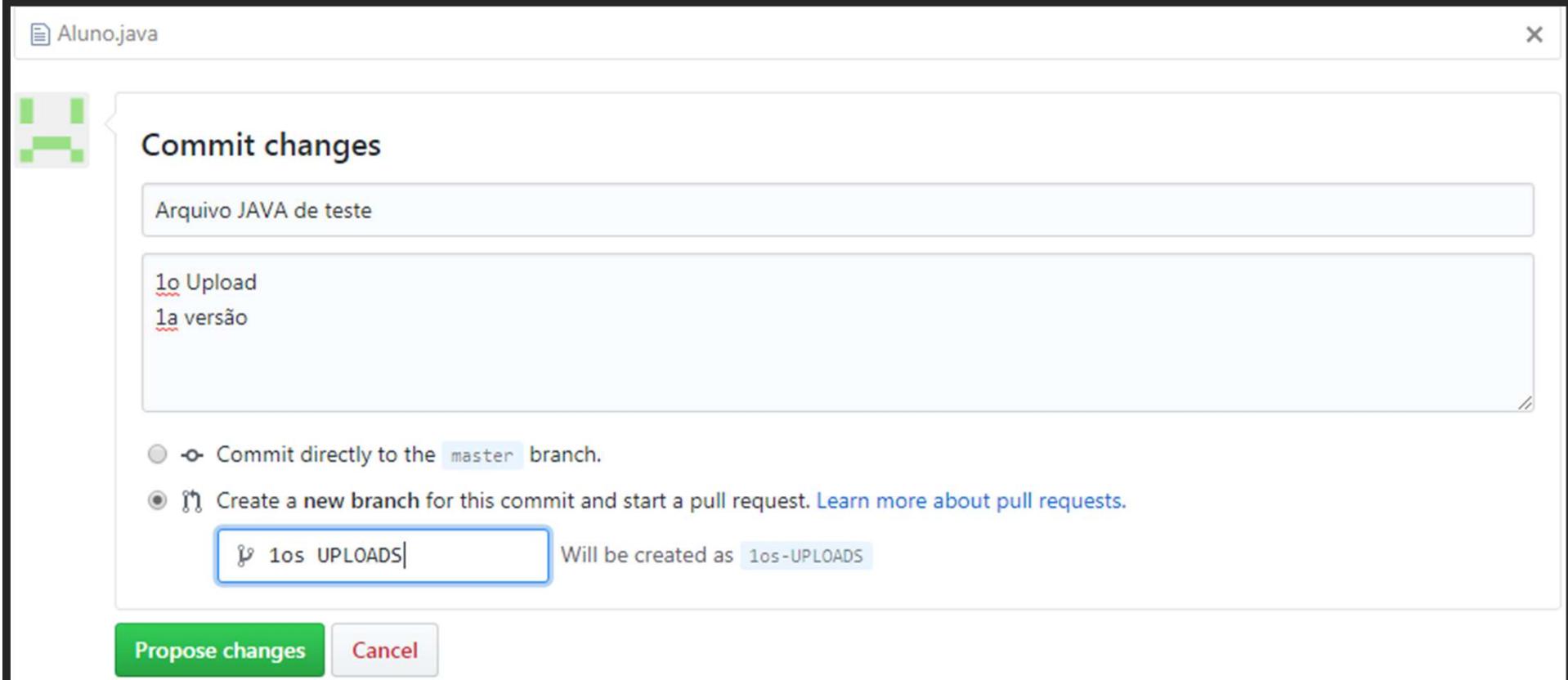
Suba um arquivo .JAVA ou .SQL para experimentar!



Como alternativa, você pode abrir a pasta com o seu arquivo no Windows Explorer e arrastá-lo para a página do GITHUB.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Cada inclusão ou alteração de arquivo pode ser comentada ao ser salva, facilitando a interpretação das versões.!



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Quando algo é modificado na área de trabalho/cópia, fica habilitada a possibilidade de criar uma Pull request para atualizar a Master a partir da Branch.!

A screenshot of a GitHub interface. At the top, it says "Your recently pushed branches:" followed by a list containing "1 los-UPLOADS (1 minute ago)". To the right of this list is a green button with a white icon and the text "Compare & pull request". A large red arrow points downwards from the text "fazer pull request" in the previous slide to this green button. Below the branch list, there are several buttons: "Branch: master ▾", "New pull request", "Create new file", "Upload files", "Find file", and "Clone or download ▾". At the bottom of the interface, it shows a commit history with "RenatoJardimParducci Initial commit" on the left and "Latest commit 7296d6c 34 minutes ago" on the right.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Depois que estiver certo que a mudança está completa e correta, publique a modificação na cópia Master, tornando-a disponível para todos os desenvolvedores usarem!

The screenshot shows a GitHub interface for a repository named 'TesteUsoGitVS'. The 'Pull Requests' tab is selected. A pull request titled 'Updated CalHoras.js' is displayed. The 'Description' section contains the note '- Updated CalHoras.js'. Below the description, it says 'Markdown supported. Use # to mention a work item or @ to mention a person.' and lists a bullet point: '• Updated CalHoras.js'. The 'Reviewers' section shows '[ProjetoExemplo-2TBA-2016]\ProjetoExemplo-2TBA-2016 Team' selected in a dropdown. The 'Work Items' section has a search bar with the placeholder 'Search work items by ID or title'. At the bottom, there is a blue button labeled 'New pull request' with a red arrow pointing to it from the left.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Experimente criar uma nova Branch para fazer alterações.!

Faça as modificações e execute o Commit.

Crie uma Pull request e execute a atualização da Master (Merge).

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

A Master terá o histórico de todas as modificações feitas.

The screenshot shows a GitHub commit page for the file `Aluno.java` in the `master` branch. The commit was made by `RenatoJardimParducci` and updated `Aluno.java`. It has `1 contributor`. The commit message is `Update Aluno.java`. The commit hash is `e92ad94` and it was made `9 minutes ago`. Below the commit details, there is a code snippet of `Aluno1.java` with 10 lines (5 sloc) and 121 Bytes. At the bottom right, there are buttons for `Raw`, `Blame`, `History` (which is highlighted with a red box and a red arrow pointing to it), and icons for desktop, edit, and delete. The URL of the commit is [https://github.com/TesteGITHUB/Aluno.java/commit/e92ad94](#).

```
public class Aluno1 extends Pessoa {  
    private Matricula matricula;  
    public void SolicitarMatricula() {  
    }  
}
```

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

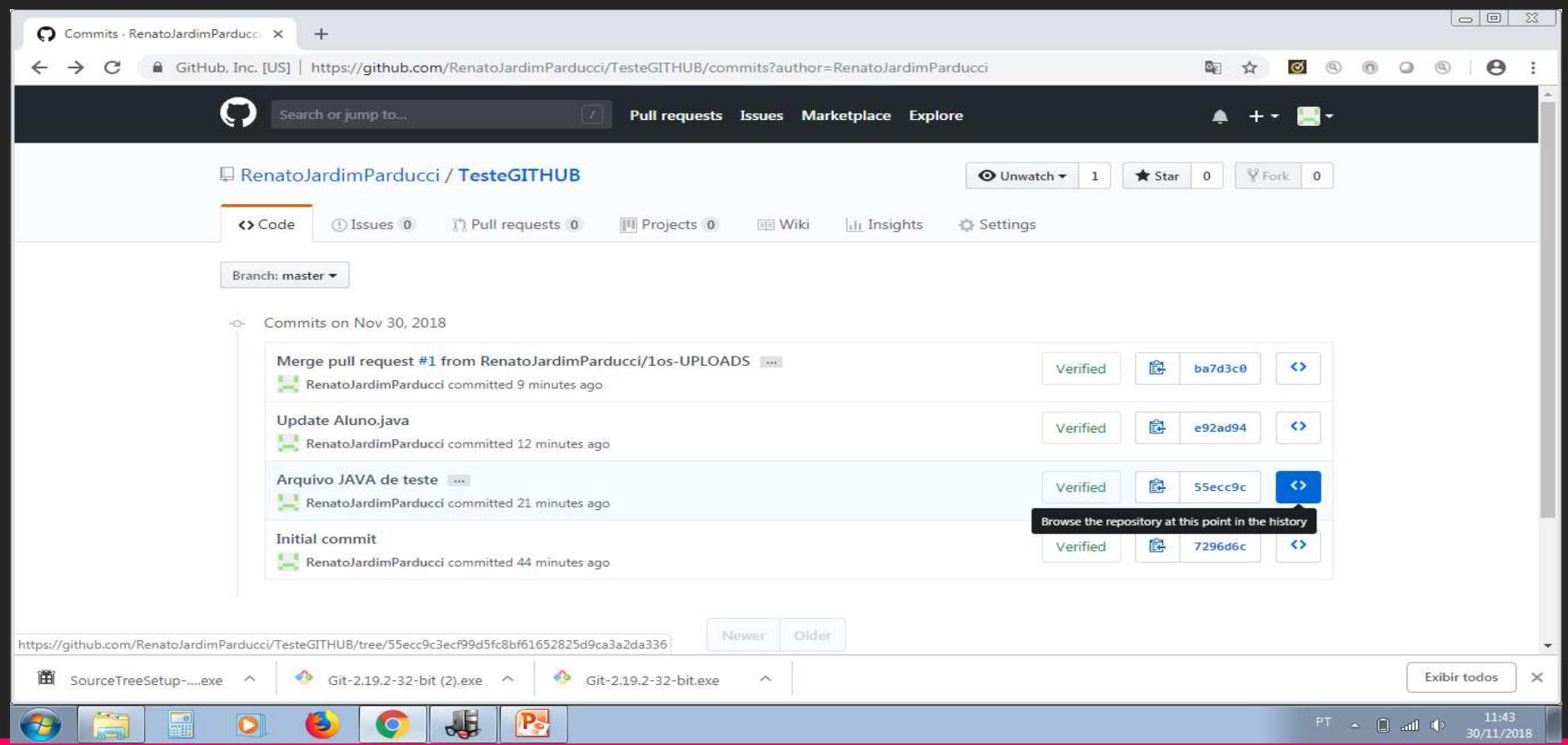
A Master terá o histórico de todas as modificações feitas.

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The 'Code' tab is selected. A dropdown menu indicates the 'Branch: master'. Below, a section titled 'Commits on Nov 30, 2018' lists four commits:

- Merge pull request #1 from RenatoJardimParducci/1os-UPLOADS ...
RenatoJardimParducci committed 10 minutes ago
Verified | ba7d3c0 | ↗
- Update Aluno.java
RenatoJardimParducci committed 14 minutes ago
Verified | e92ad94 | ↗
- Arquivo JAVA de teste ...
RenatoJardimParducci committed 23 minutes ago
Verified | 55ecc9c | ↗
- Initial commit
RenatoJardimParducci committed an hour ago
Verified | 7296d6c | ↗

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Caso você precise recuperar uma versão anterior, basta selecioná-la.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Quando selecionada uma versão do fonte, o GIT mostra o que foi alterado para você ter certeza de qual versão está vendo.

The screenshot shows a GitHub commit history for the file `Aluno.java`. The commit was made by RenatoJardimParducci 12 minutes ago and is verified. It has one parent commit `55ecc9c` and a commit hash `e92ad942ca6b8d544ad4282d95dfcde8d1121567`. The commit message is not visible. The interface shows a unified diff view with 1 changed file, 1 addition, and 1 deletion. The code change is:

```
@@ -1,4 +1,4 @@
- public class Aluno extends Pessoa {
+ public class Aluno1 extends Pessoa {
```

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Se precisar deletar uma Branch que ficou pra trás (não deletada no momento do MERGE), acesse na página principal do projeto, o número de branches e na janela que aparecer, clique na lata de lixo para eliminar a branch correspondente.

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The main header has tabs for Code, Issues (0), Pull requests (0), Projects (0), and Wiki. Below the header, it says 'No description, website, or topics provided.' and has a 'Manage topics' link. On the left, there's a summary card with 8 commits. Below that, a dropdown says 'Branch: master' and a button says 'New pull request'. A list of files includes 'Aluno.java' and 'README.md'. In the center, a large red box highlights the text '2 branches' next to a trash icon. A red arrow points from this red box to a detailed list of branches on the right. The right side shows the 'Overview' tab selected, with other tabs for Yours, Active, Stale, and All branches. It lists two branches: 'master' (the default branch) and 'SUPERBRANCH-01'. The 'master' branch was updated a minute ago by RenatoJardimParducci. The 'SUPERBRANCH-01' branch was updated 2 minutes ago by RenatoJardimParducci and has a status of '#3 Merged'. There are also sections for 'Your branches' and 'Active branches'.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você acabou de estudar um processo com atividades, definição de responsabilidades e ferramentas para gerenciar fontes de programas de aplicação em suas versões.

Seu 1º passo para garantir Qualidade em projetos de software e a Governança, através da garantia da continuidade dos negócios.

ASSISTA OS VÍDEOS NO CANAL DO PROFESSOR



<https://youtu.be/MYhIM0bk9aQ>

A screenshot of a video player displaying a GitHub tutorial. The video frame shows a man with a shaved head, wearing a dark t-shirt, looking at the camera. The video content is a GitHub tutorial titled "Learn Git and GitHub without any code!". It features a "Read the guide" button and a "Start a project" button. Below the video frame, there is a news feed interface with a play button, volume icon, and a timestamp of 0:30 / 13:49. A tooltip message says "Visualize your project's community". The video player has a red border and a play button icon in the bottom right corner.



Conteúdo didático complementar - GitHub

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode ainda, baixar o aplicativo GIT para Windows ou outra plataforma em <https://git-scm.com/download/win>, para fazer comandos da sua máquina.

Downloading Git

Your download is starting...

You are downloading the latest (2.19.2) 32-bit version of Git for Windows. This is the most recent [maintained build](#). It was released **9 days ago**, on 2018-11-21.

If your download hasn't started, [click here to download manually](#).

Other Git for Windows downloads

[Git for Windows Setup](#)

[32-bit Git for Windows Setup](#) (highlighted)

[64-bit Git for Windows Setup](#).

Now What?

Now that you have downloaded Git, it's time to start using it.

Read the Book
Dive into the Pro Git book and learn at your own pace.

Download a GUI
Several free and commercial GUI tools are available for the Windows platform.

Get Involved
A knowledgeable Git community is available to answer your questions.

A large red arrow points from the bottom left towards the "32-bit Git for Windows Setup" link. A second red arrow points from the bottom right towards the "Download a GUI" section.

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Pode também selecionar um gerenciador de interface gráfica (Client – GIT) como o SourceTree para não ter que executar comandos GIT via linha de comando.

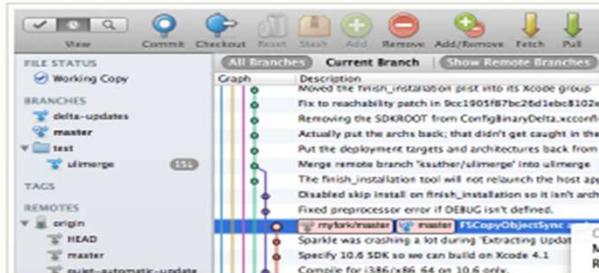
GUI Clients

Git comes with built-in GUI tools for committing ([git-gui](#)) and browsing ([gitk](#)), but there are several third-party tools for users looking for platform-specific experience.

If you want to add another GUI tool to this list, just [follow the instructions](#).

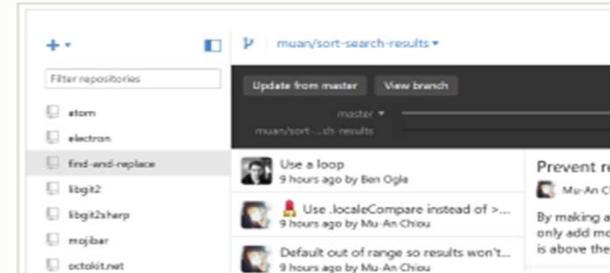
[All](#)[Windows](#)[Mac](#)[Linux](#)[Android](#)[iOS](#)

22 Windows GUIs are shown below ↓



SourceTree

Platforms: Mac, Windows
Price: Free
License: Proprietary

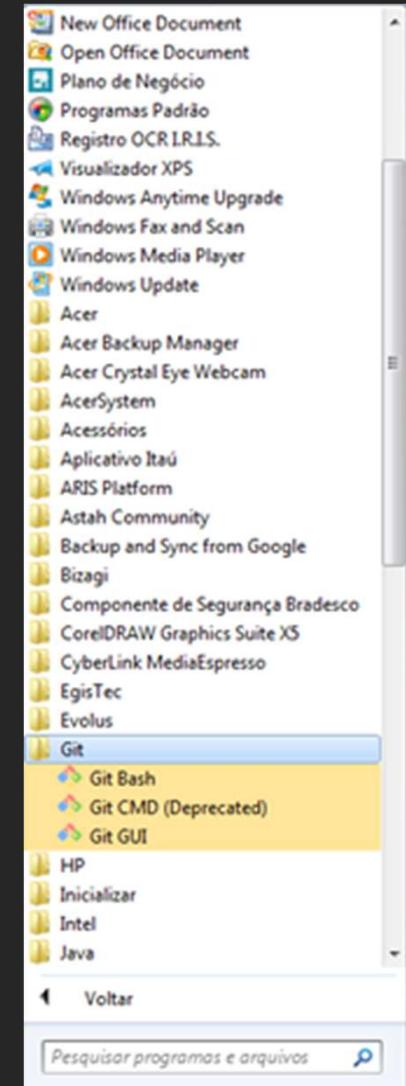


GitHub Desktop

Platforms: Mac, Windows
Price: Free
License: MIT

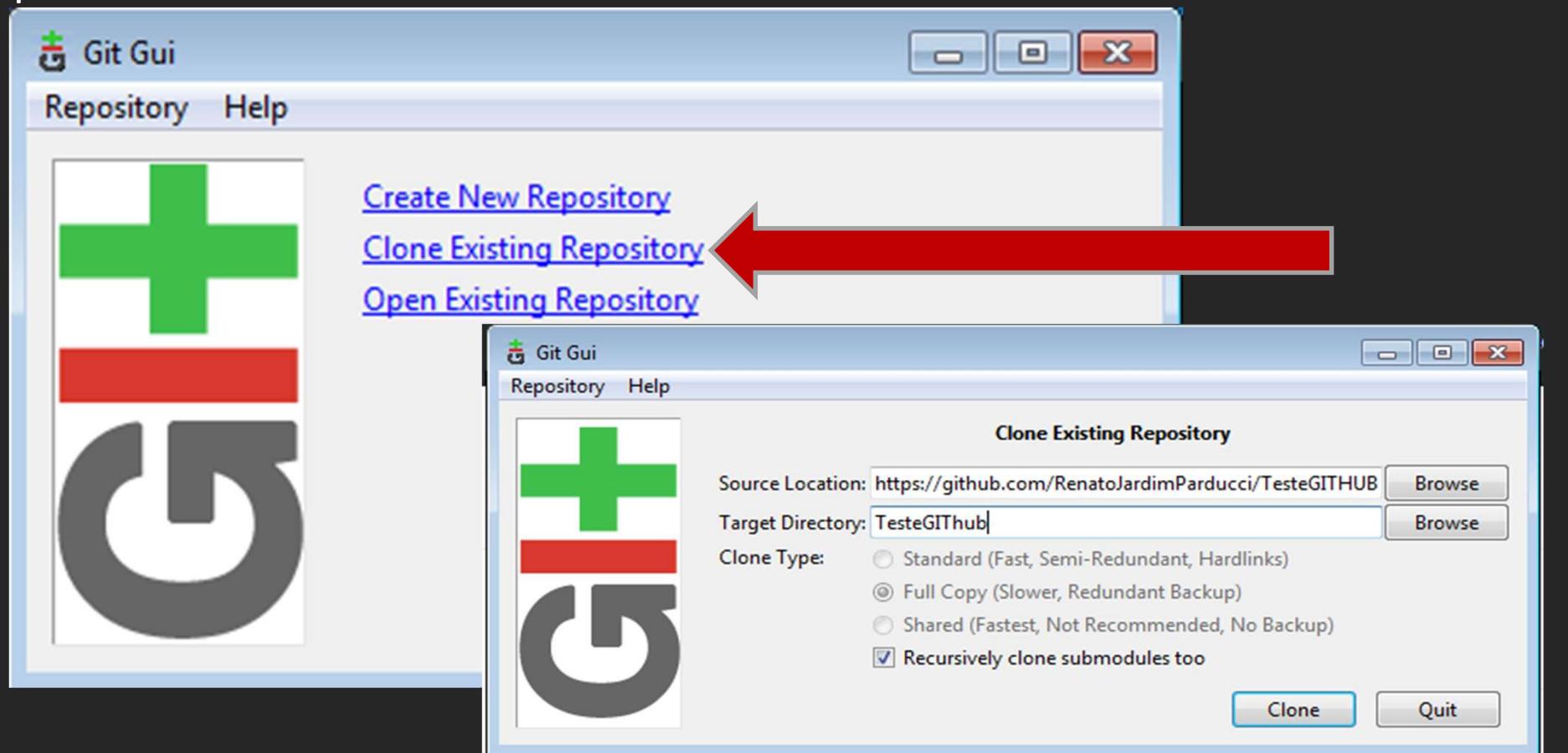
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Esses aplicativos adicionais permitem que você publique e puxe arquivos para manutenção usando somente o seu computador pessoal, sem necessidade de usar o navegador, como nós fizemos no nosso exemplo.



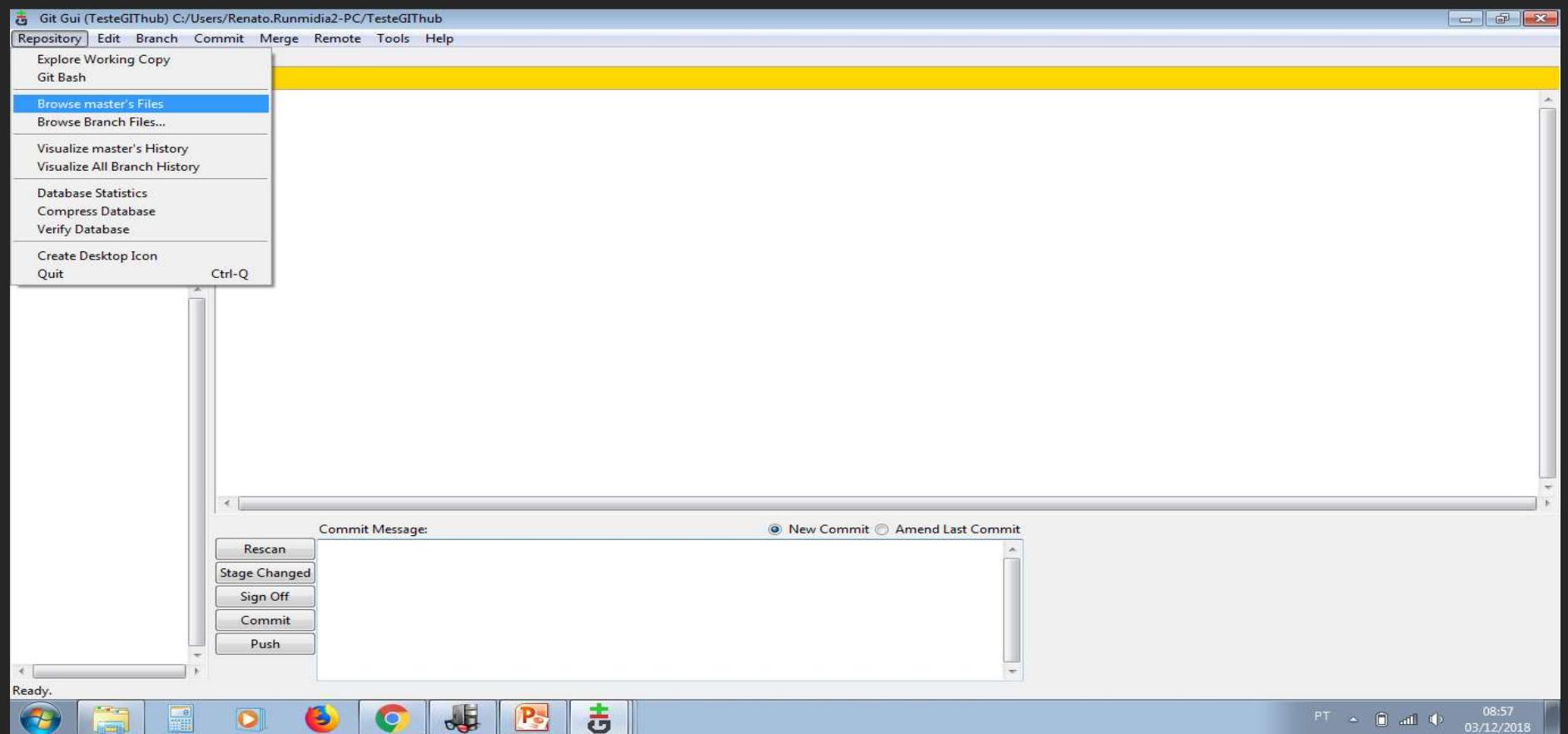
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode sincronizar os repositórios em nuvem com o computador pessoal.



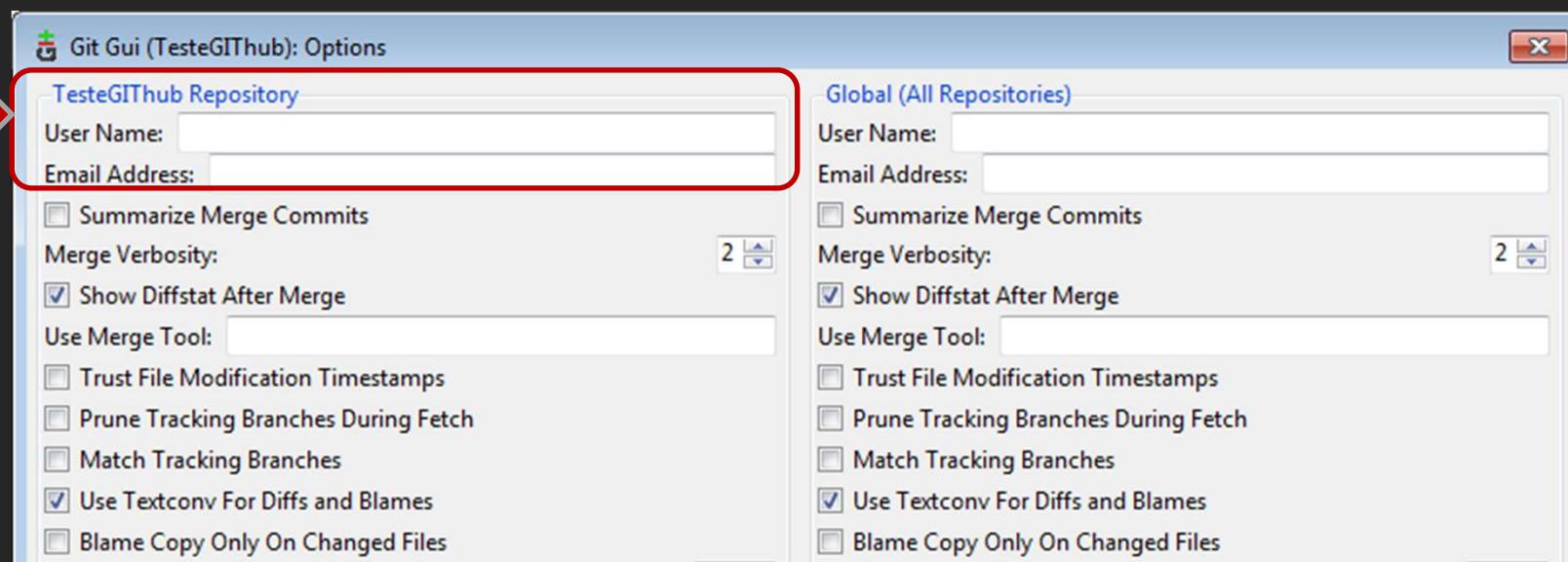
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Com o GitGUI, por exemplo, você pode acessar as áreas Master, Branches e pastas pelos menus o seu PC..



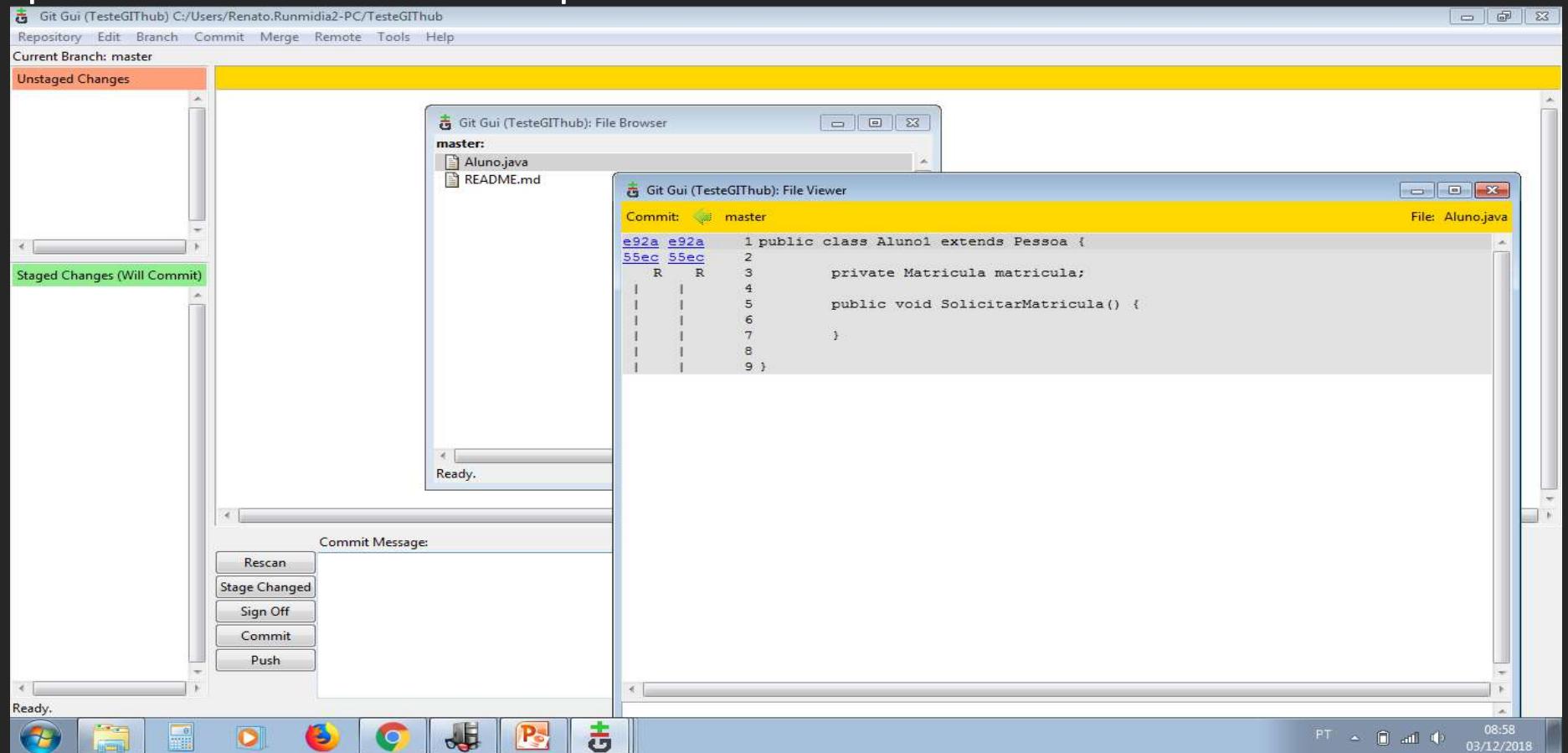
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Ele pode sincronizar com o GitHub.com se você configurar o Submenu Edit/Options com os dados do seu login na WEB.



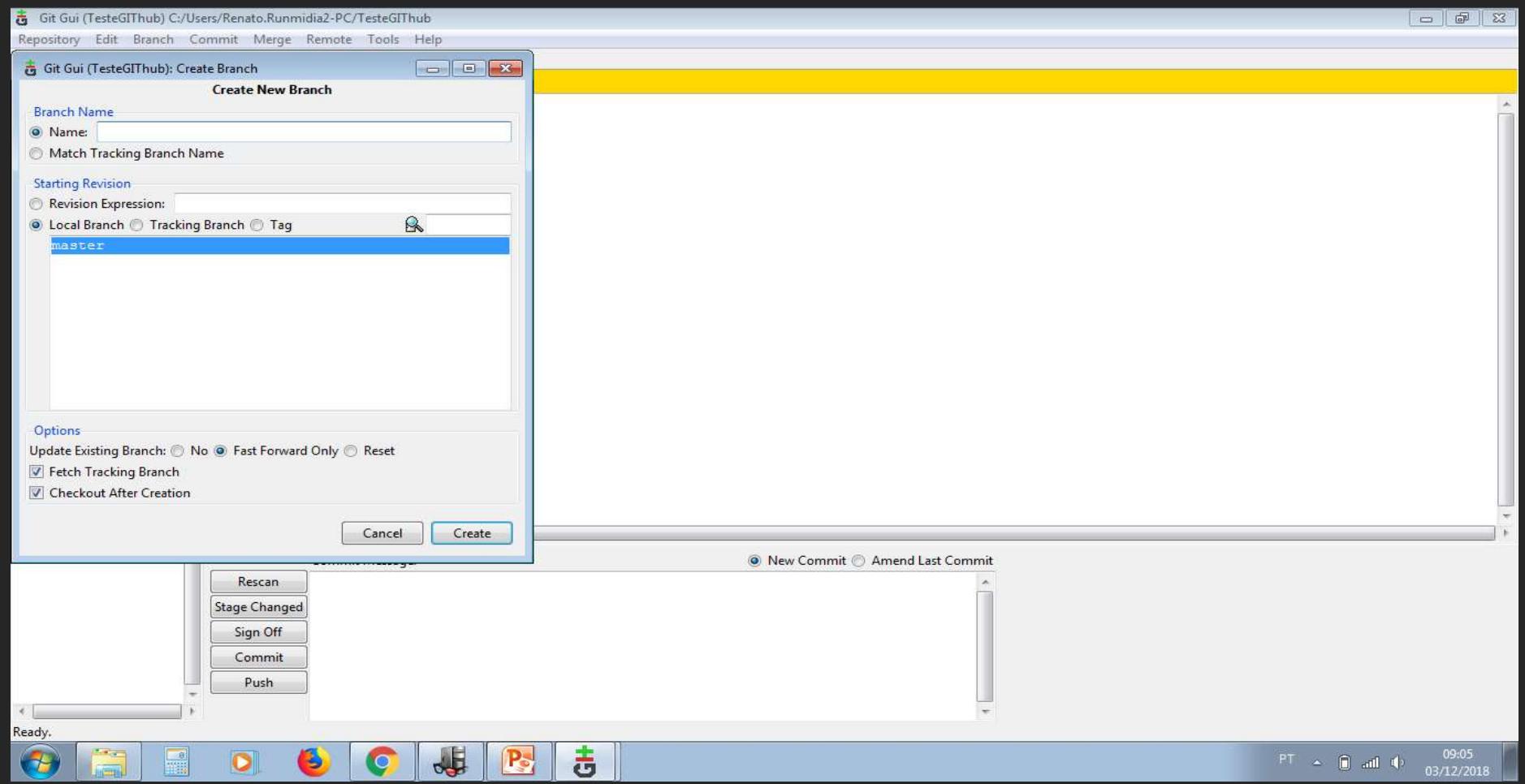
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode sincronizar os repositórios em nuvem com o computador pessoal e visualizar seus repositórios.



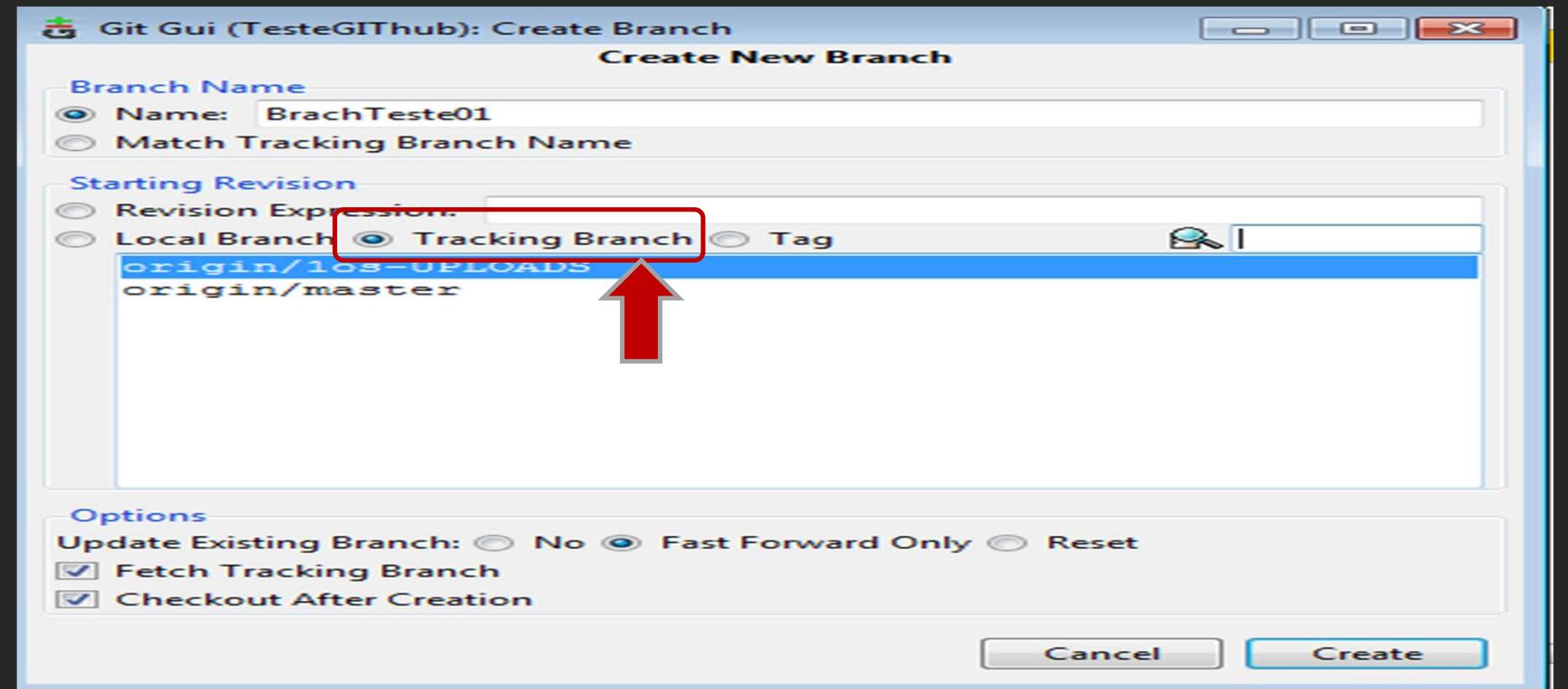
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Pode ainda criar branch...



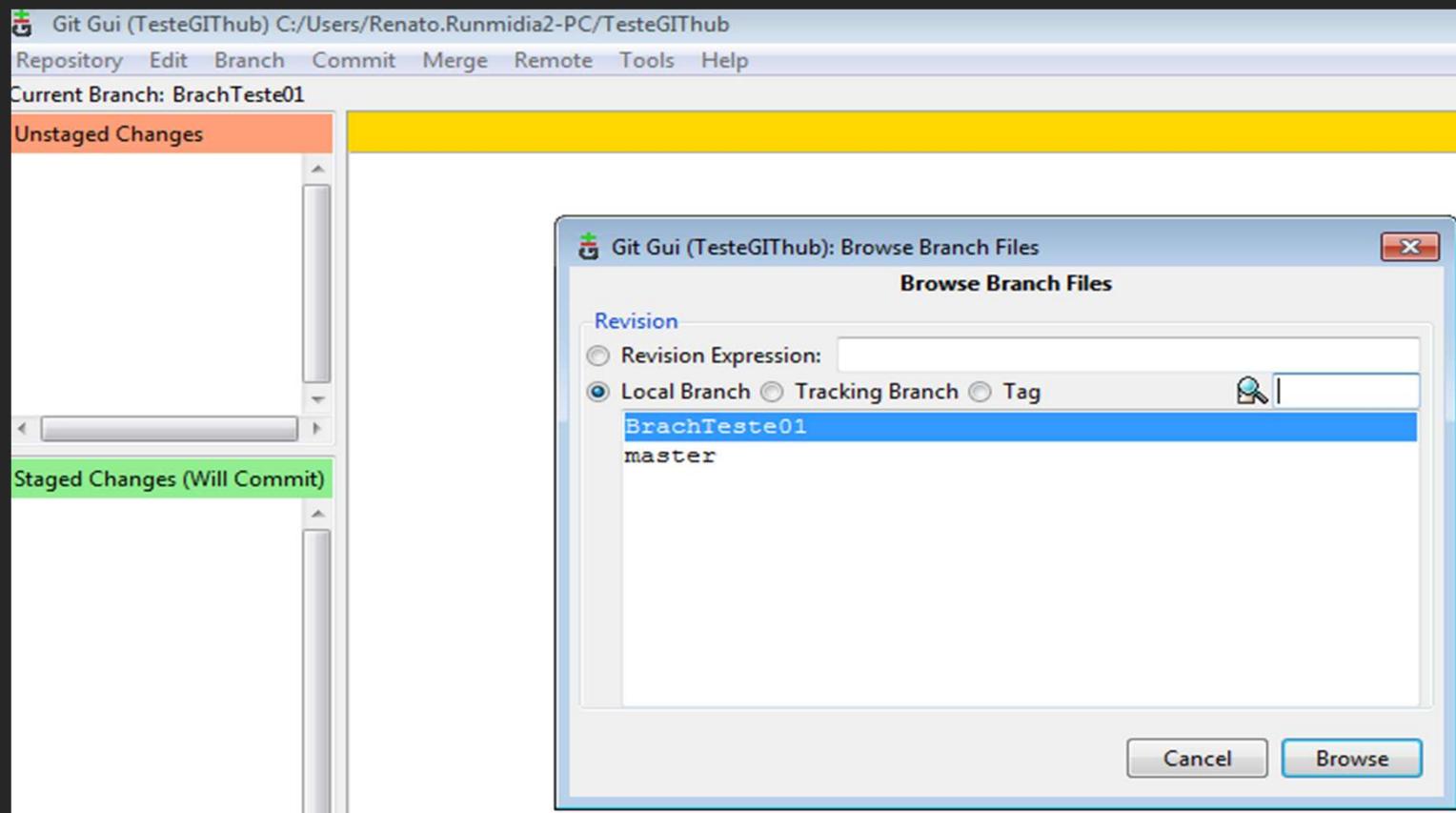
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

E essa Branch pode ser local (só no seu computador) ou rastreada (track), ligada com o repositório remoto na Internet (associada ao GitHub).



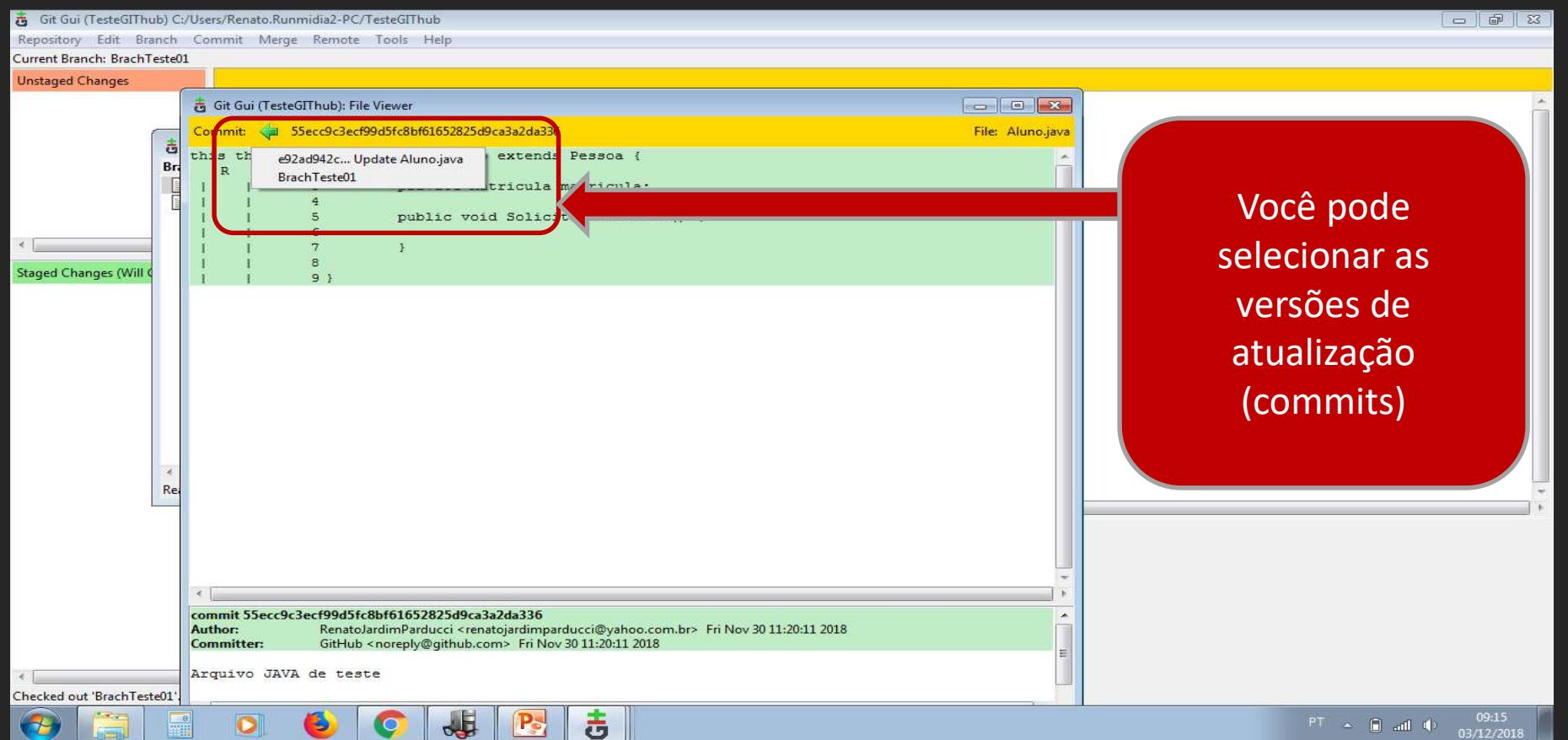
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Após criada a Branch, a navegação de acesso aos fontes é via menu da aplicação GUI.



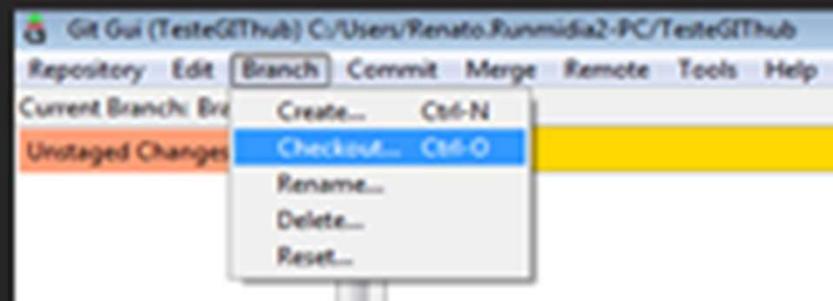
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Após criada a Branch, a navegação de acesso aos fontes é via menu da aplicação GUI.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

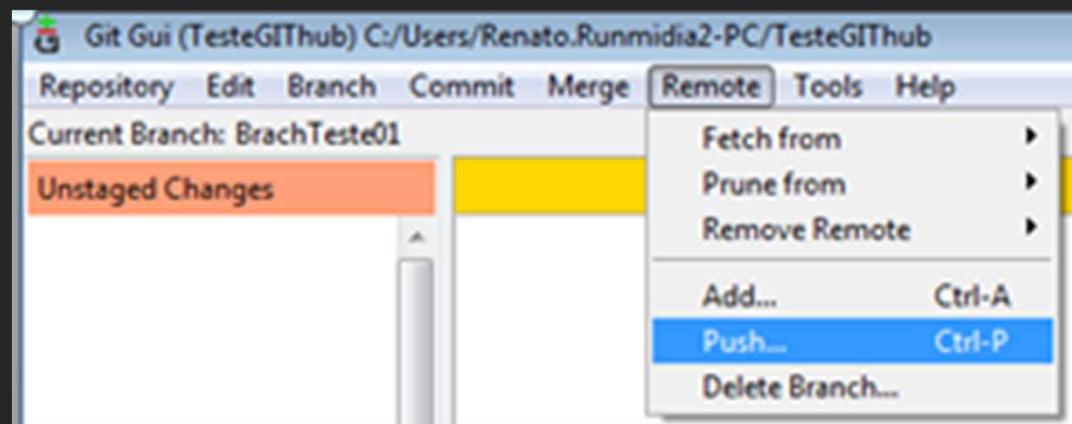
Para editar arquivos na Branch, faça um Checkout!



Use a opção REPOSITORY/EXPLORE WORKING COPY para acessar os documentos no repositório local e editá-los!

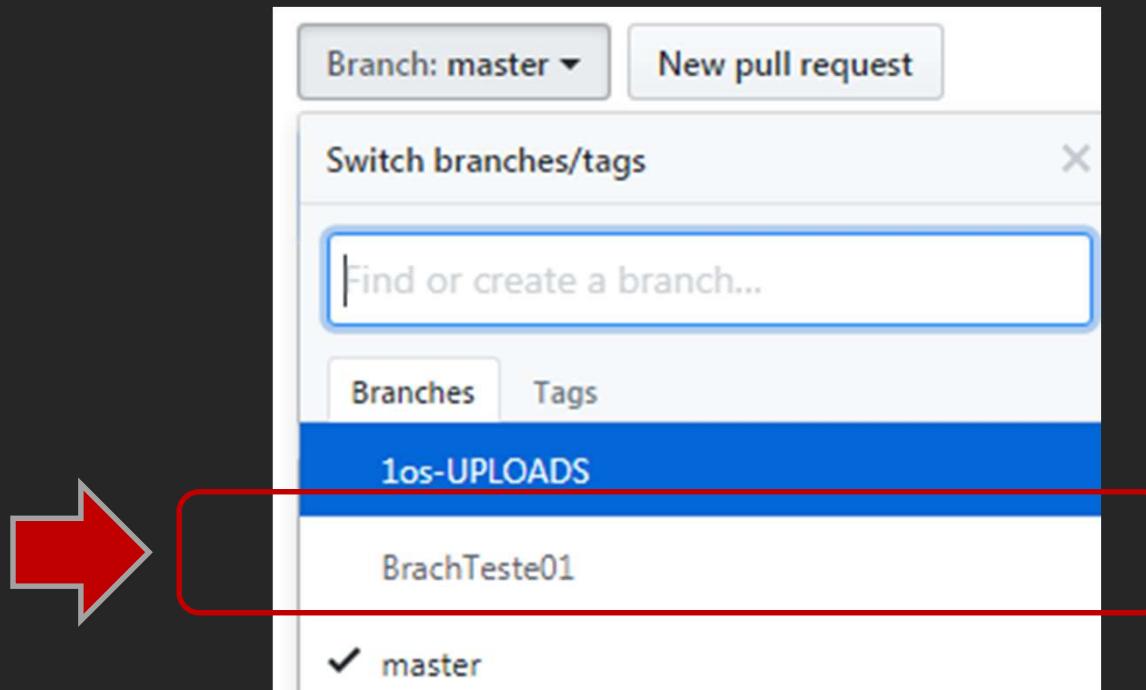
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Se você desejar publicar a Branch criada no seu PC no GitHub.com, faça um PUSH no submenu Remote (será exigido o Login do GitHub para essa operação)!



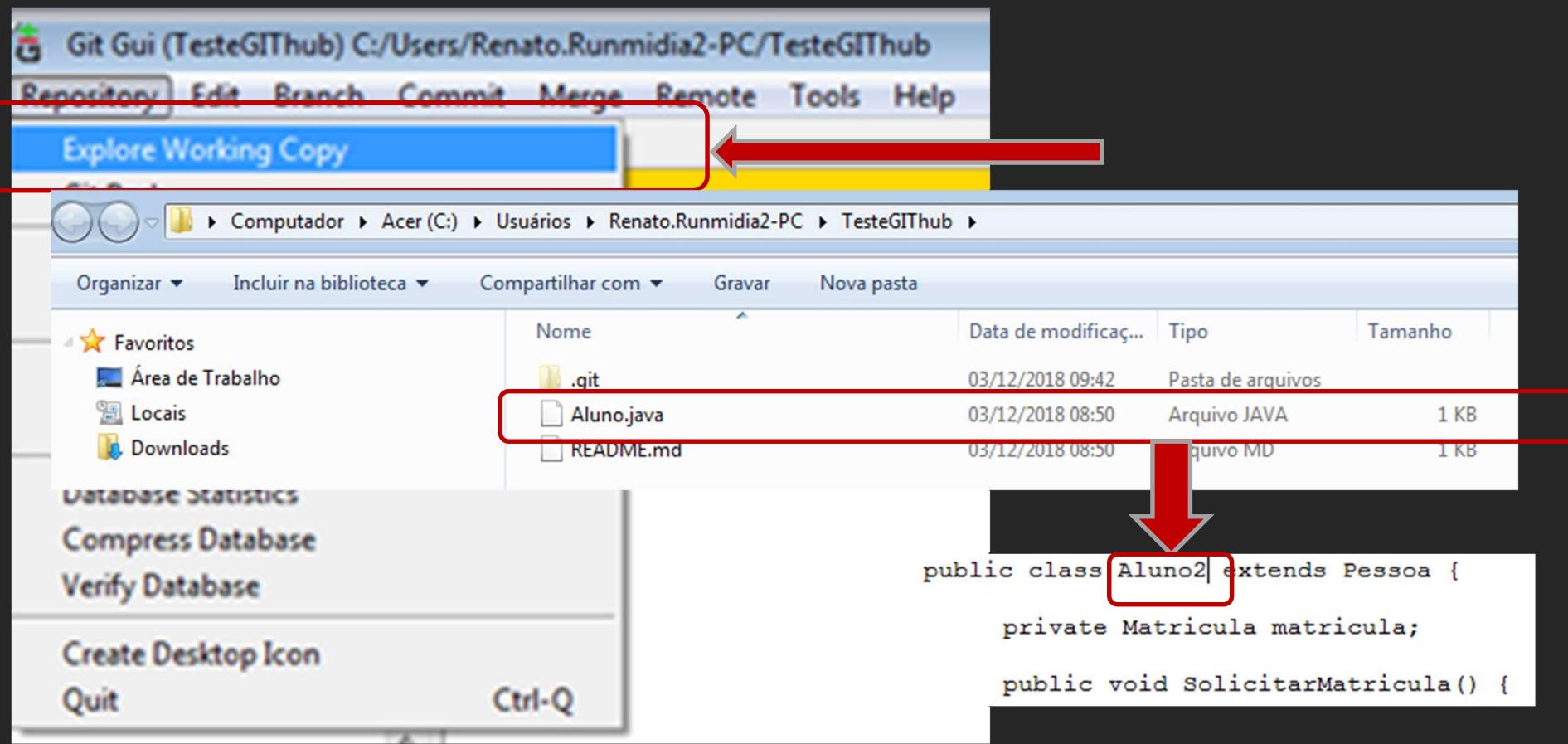
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Após o Push, você poderá visualizar e operar a Branch e seus arquivos na WEB:



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

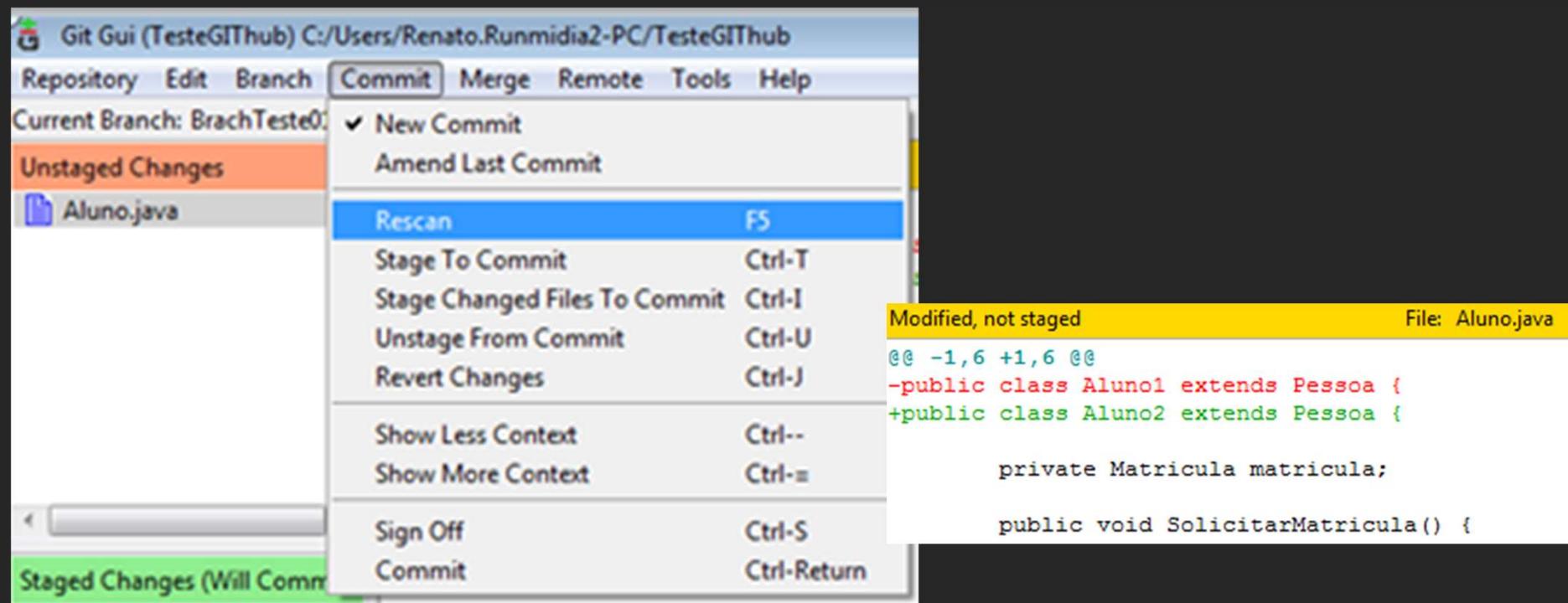
Você pode editar os arquivos no seu PC:



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

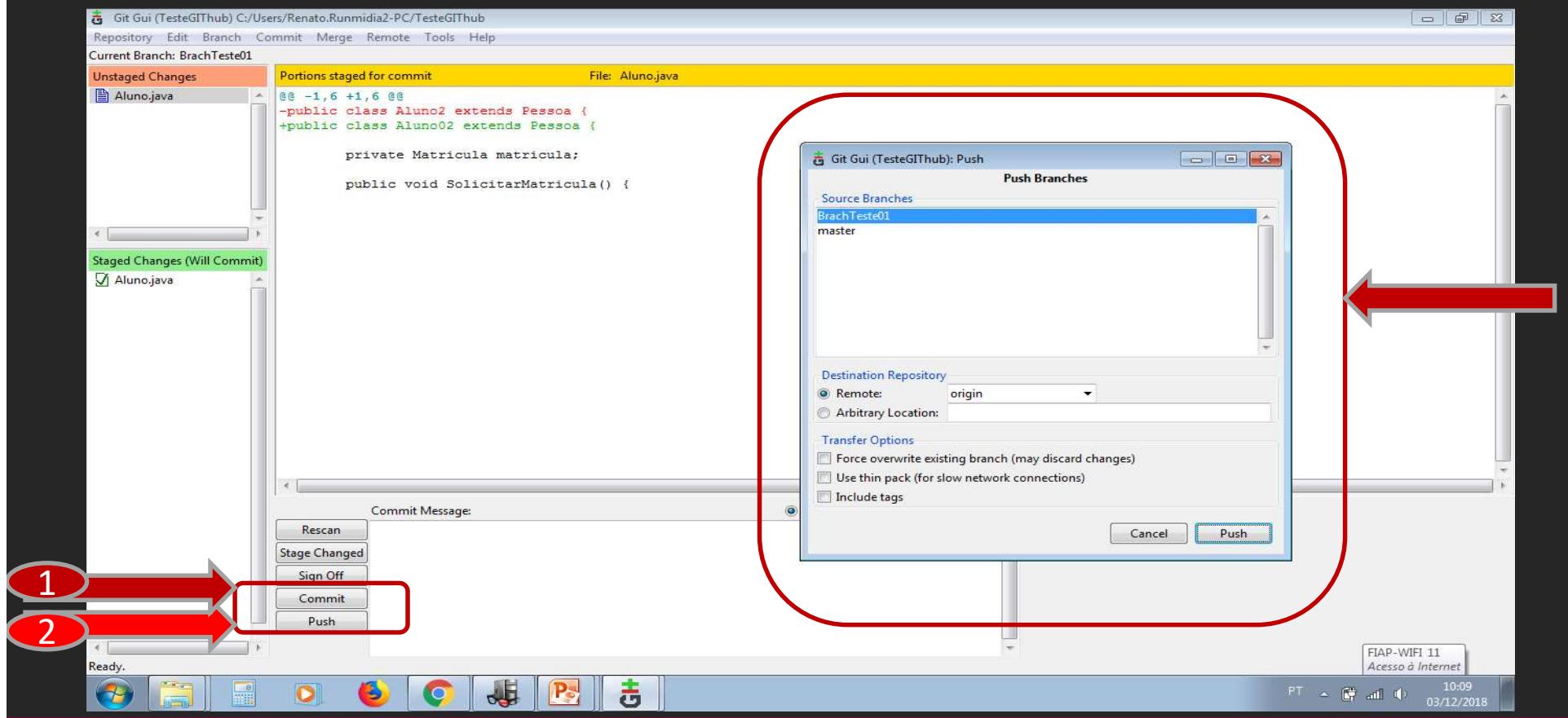
As edições são refletidas no GitGui e podem ser atualizadas por Push no GitHub!

Vá no submenu Commit/Rescan para descobrir as mudanças feitas, antes de fazer o Push!



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Executando o Push via Menu Commit: execute o Commit e depois o Push!!



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Com esses procedimentos, você consegue fazer Check out e Check in (Push) , mantendo sincronia entre o ambiente remoto dos fontes e o seu ambiente local de trabalho.

Após os Pushes, confira sempre se o GitHub.com foi atualizado corretamente.

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The repository has 1 commit, 0 stars, and 0 forks. A recent push to the 'BrachTeste01' branch is highlighted. The commit message is 'Mudança via Cliente GUI' and it was made 5 minutes ago by 'Aluno.java'. There are buttons for creating a pull request, cloning, and comparing branches.

Search or jump to... / Pull requests Issues Marketplace Explore

Unwatch 1 Star 0 Fork 0

Your recently pushed branches:

BrachTeste01 (less than a minute ago) Compare & pull request

Branch: BrachTeste01 New pull request Create new file Upload files Find file Clone or download

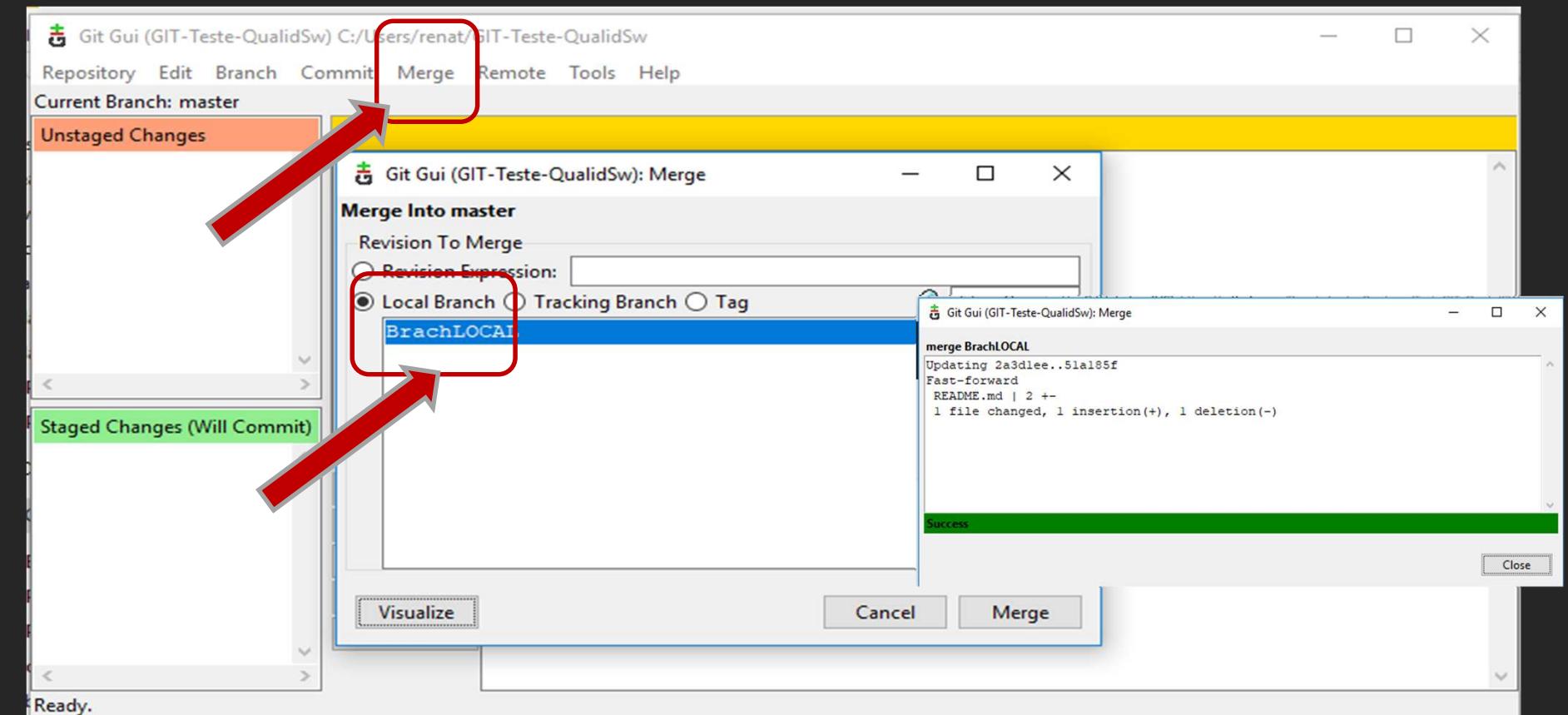
This branch is 1 commit ahead of master. Pull request Compare

RenatoJardimParducci Mudança via Cliente GUI Latest commit b700069 5 minutes ago

Aluno.java Mudança via Cliente GUI 5 minutes ago

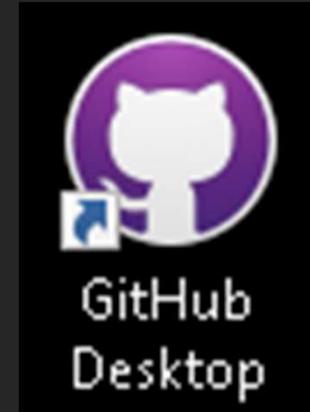
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

IMPORTANTE: o fato de você ter atualizado a Branch e Master no GITHUB não implica que a Master Local tenha sido atualizada.
Para atualizar a Master Local com as modificações, use o menu MERGE.



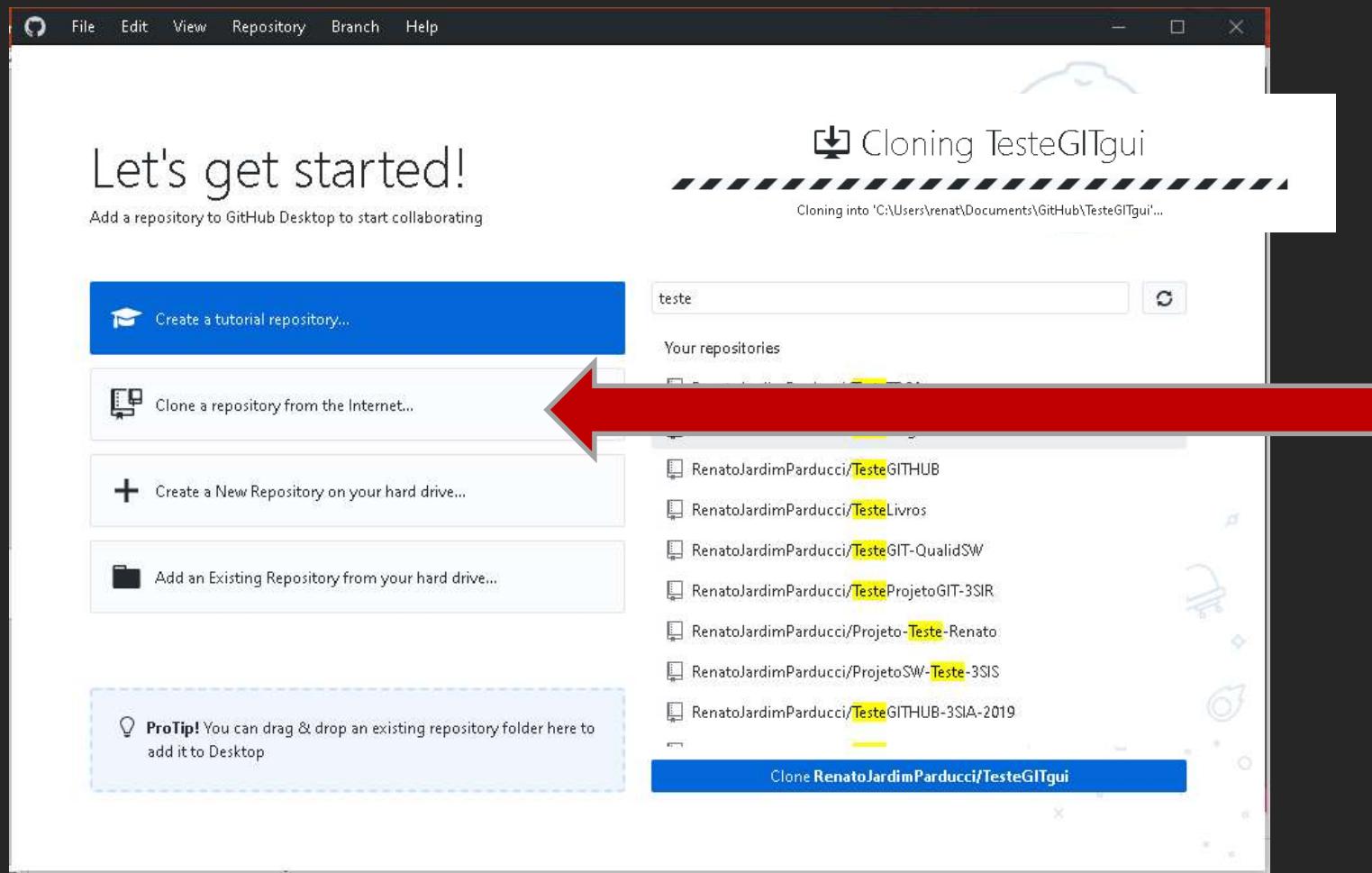
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Outro aplicativo muito interessante é o GitHub Desktop que pode ser obtido no próprio portal do GITHUB.



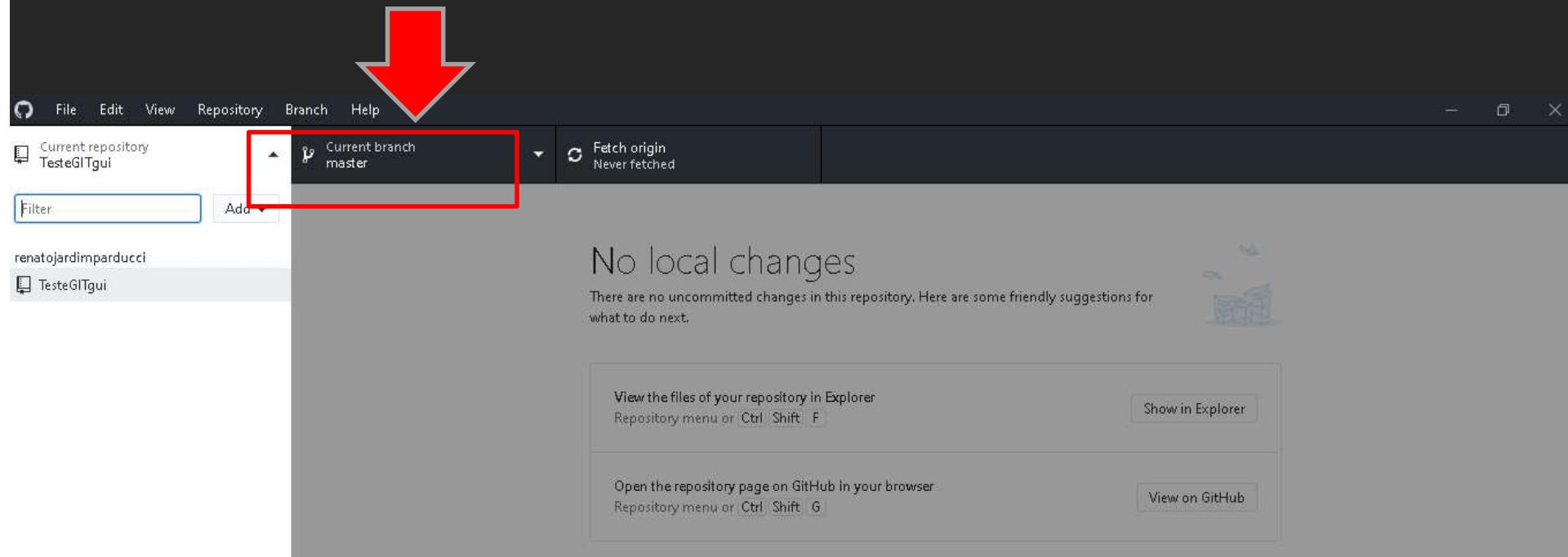
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode sincronizar os repositórios em nuvem com o computador pessoal.



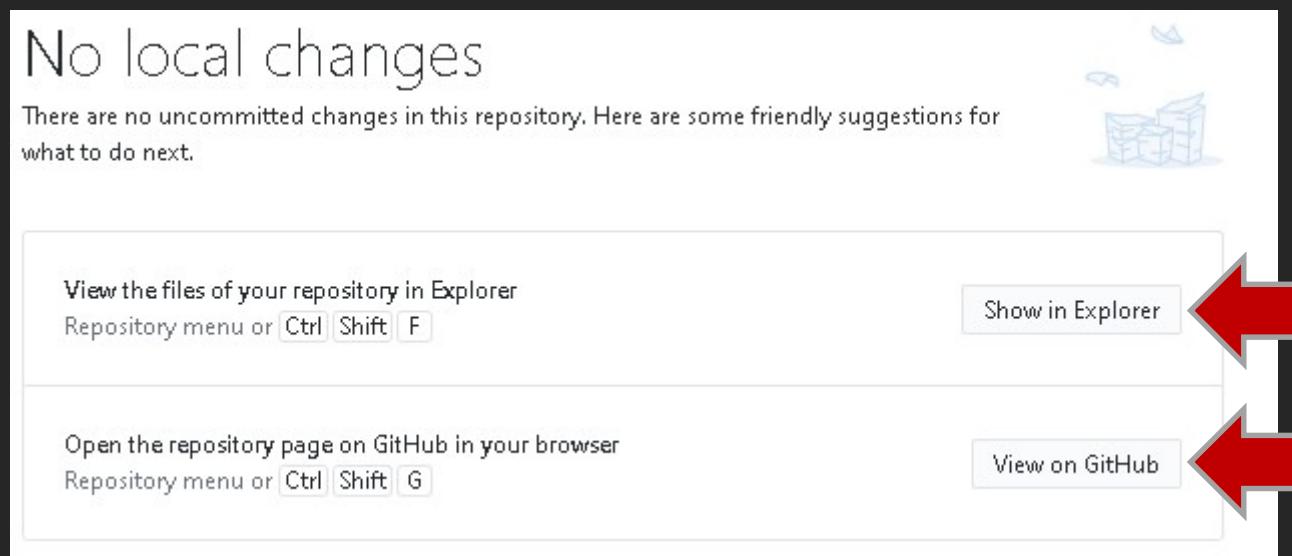
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Com o GitGUI, por exemplo, você pode acessar as áreas Master, Branches e pastas pelos menus o seu PC..



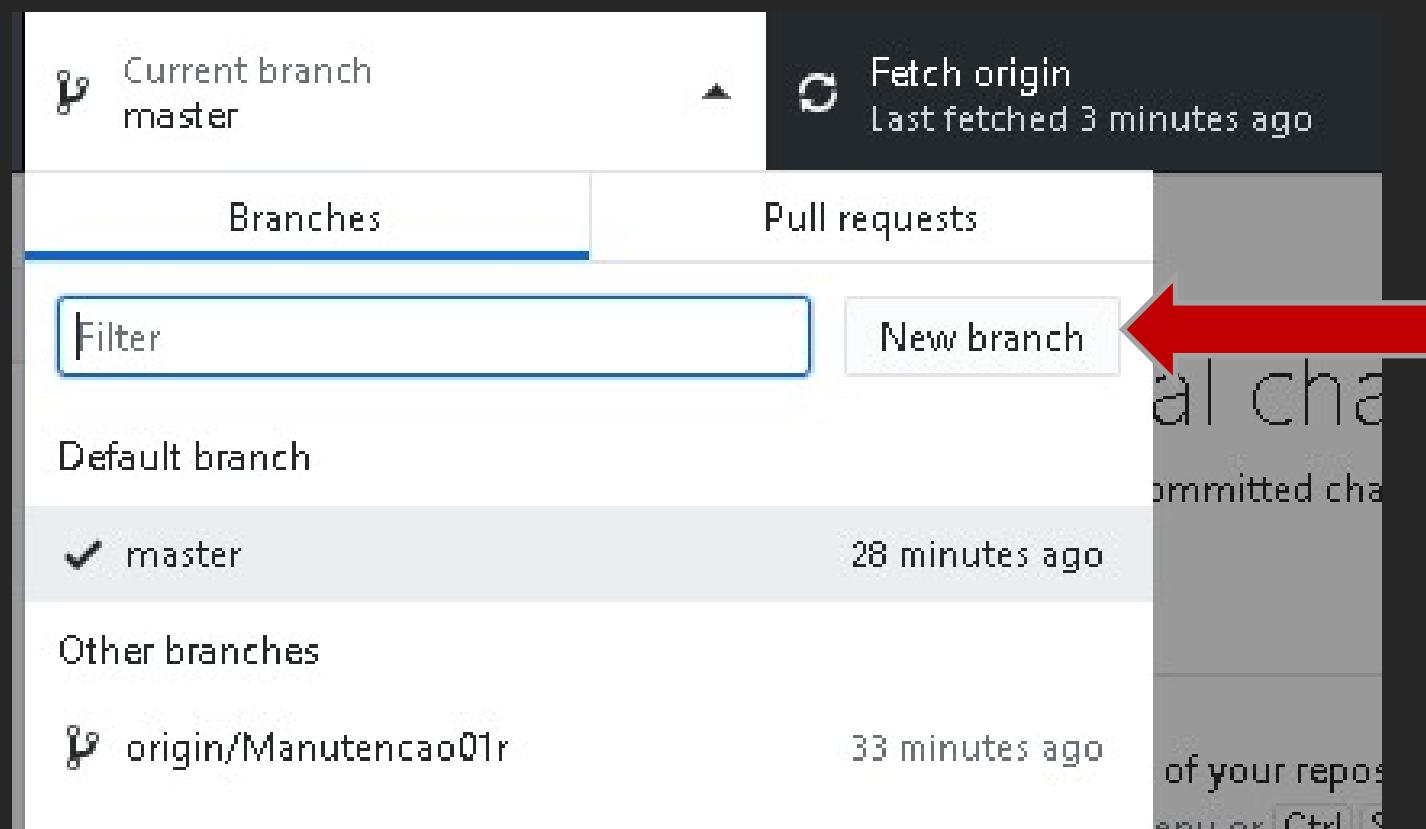
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode acessar o clone local ou a pasta remota do GITHUB por esse software cliente.



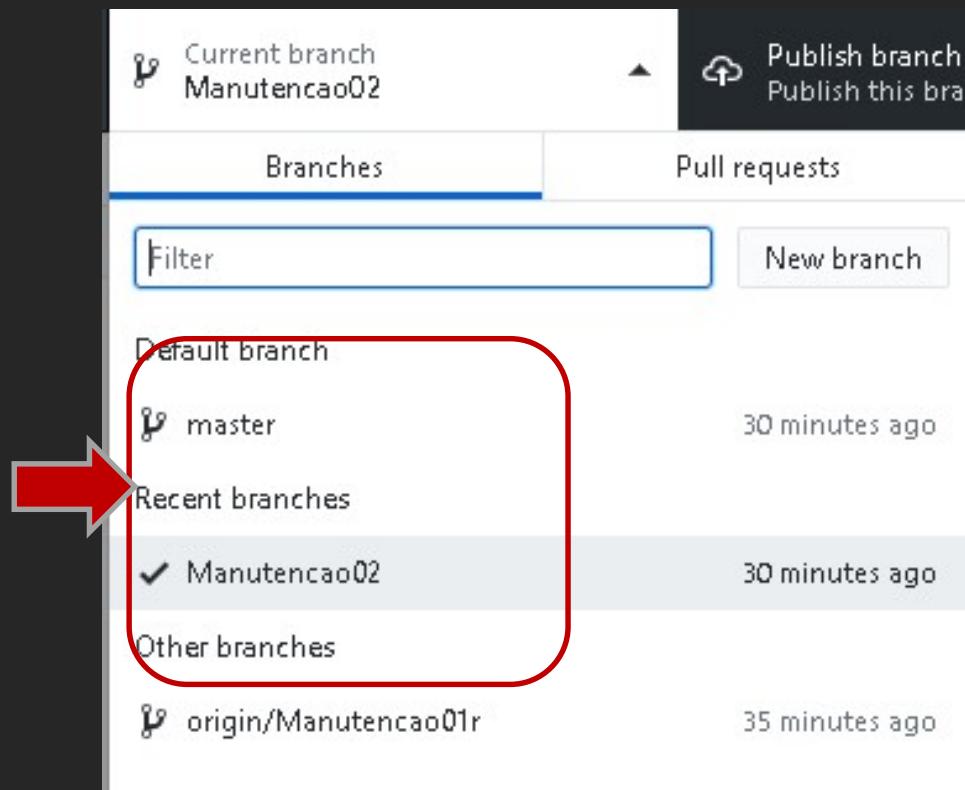
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Pode ainda criar branch...



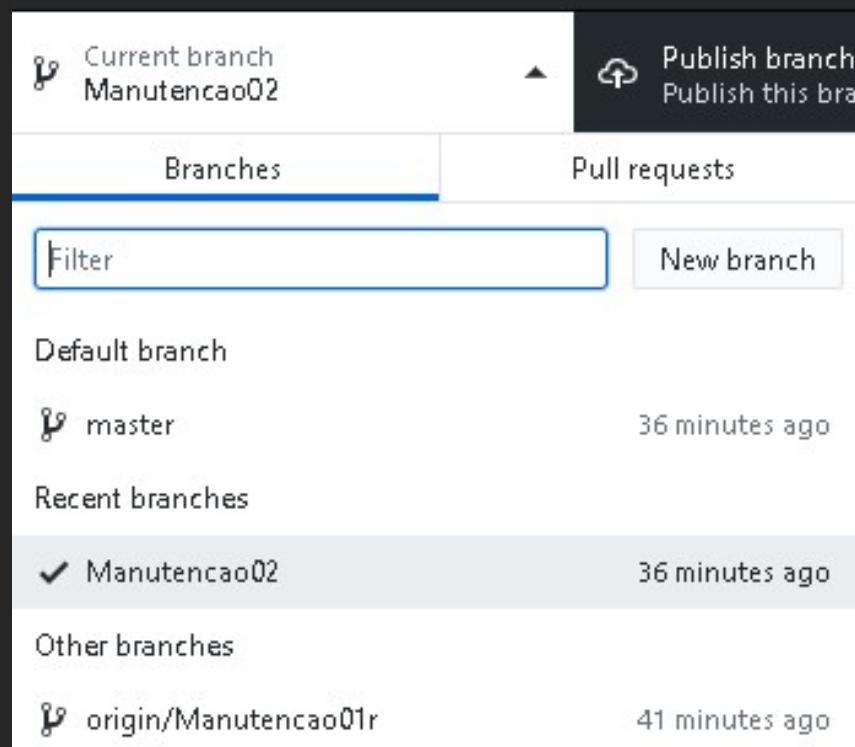
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

E essa Branch quando criada localmente, pode ser visualizada no GitHub client.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

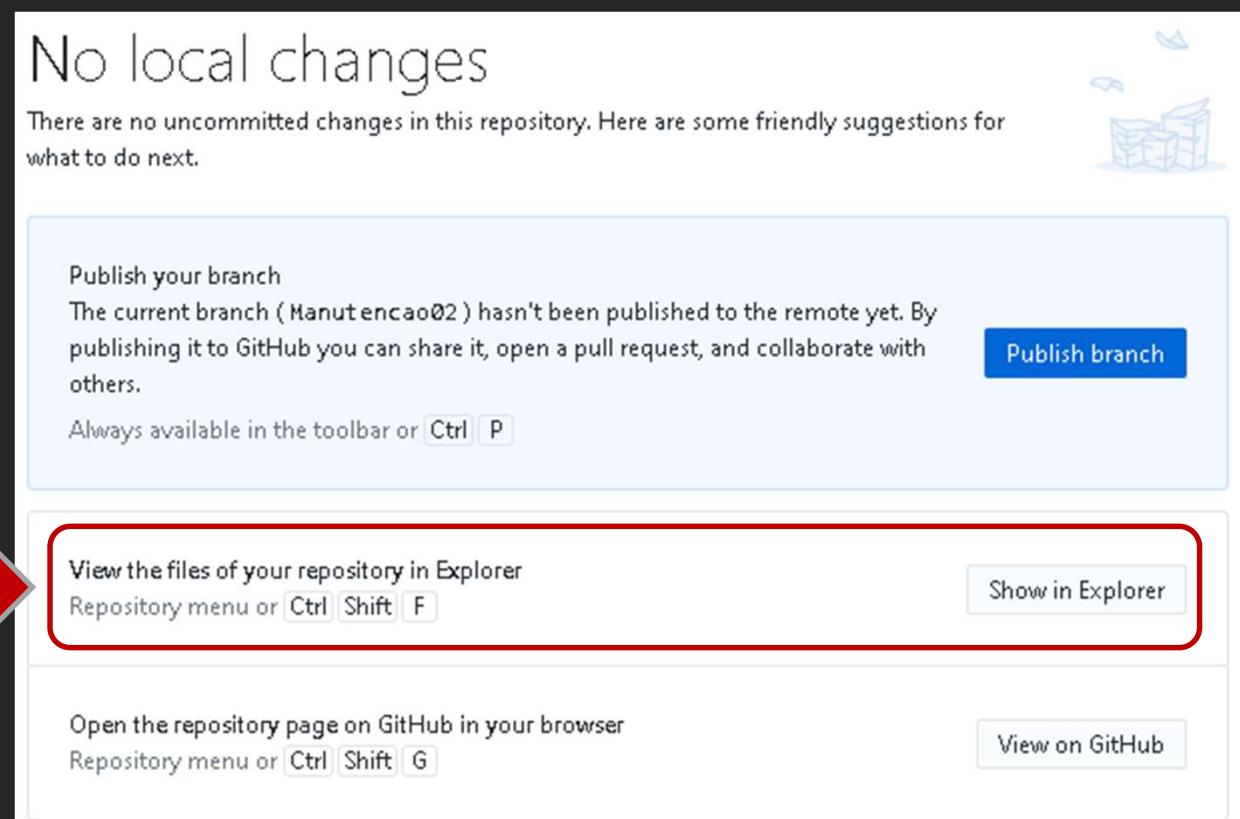
Para editar arquivos na Branch, faça um Checkout!



Selecione a branch no menu de navegação da ferramenta!

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Após criada a Branch, a navegação de acesso aos fontes é via menu da aplicação GUI.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

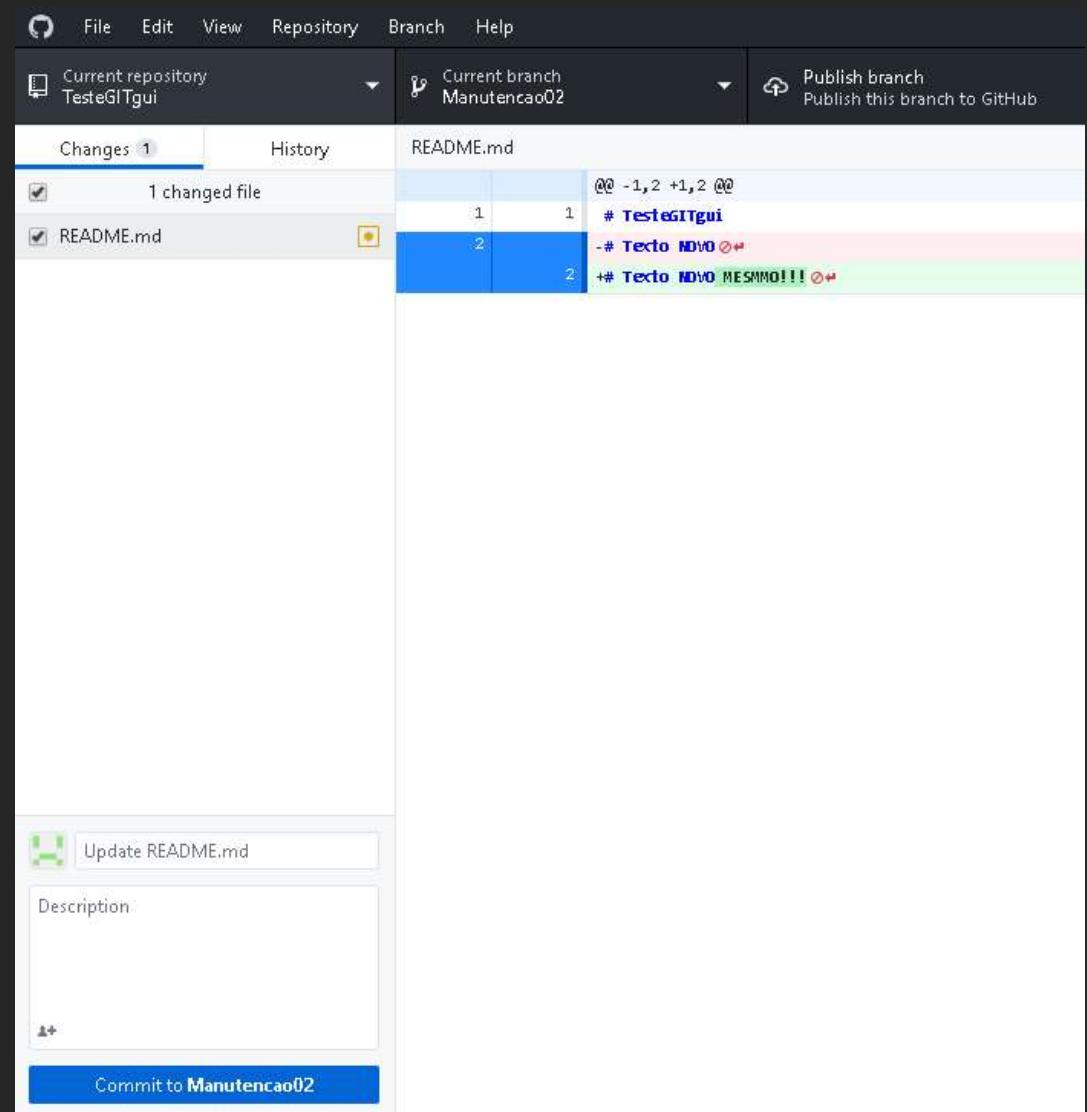
Assim que alterado o arquivo ou criado um arquivo novo na pasta do GIT local, o GitHub Client vai mostrar as diferenças entre a Master e a Branch.



The screenshot shows the GitHub Desktop application interface. At the top, there's a menu bar with File, Edit, View, Repository, Branch, and Help. Below the menu, it displays the current repository as "TesteGITgui" and the current branch as "Manutencao02". There's also a button to "Publish branch" to GitHub. The main area is divided into two sections: "Changes" (which shows 1 changed file) and "History". The "Changes" section is currently selected and displays a diff view for the file "README.md". The diff shows three lines of text: the first line is a comment "# TesteGITgui", the second line is "-# Texto NOVO", and the third line is "+# Texto NOVO MESMO!!!". The lines are color-coded: the first line is blue, the second is red, and the third is green.

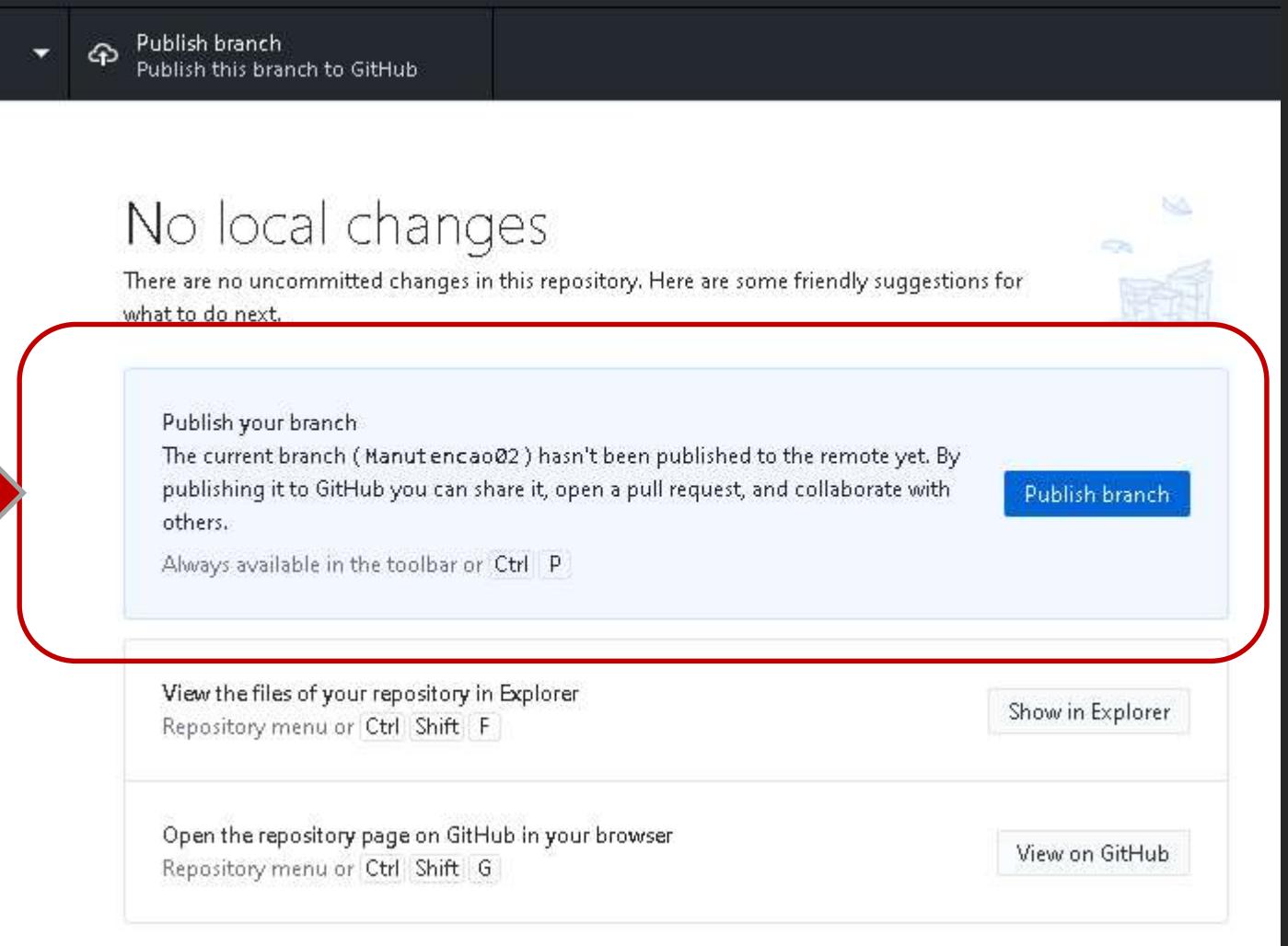
GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Se você desejar publicar a Branch criada no seu PC no GitHub.com, primeiro, faça um Commit no Git local



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Faça o PUSH no
GITHUB remoto
(WEB).



Publish branch
Publish this branch to GitHub

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Publish your branch
The current branch (Manutencao02) hasn't been published to the remote yet. By publishing it to GitHub you can share it, open a pull request, and collaborate with others.

Always available in the toolbar or **Ctrl P**

Publish branch

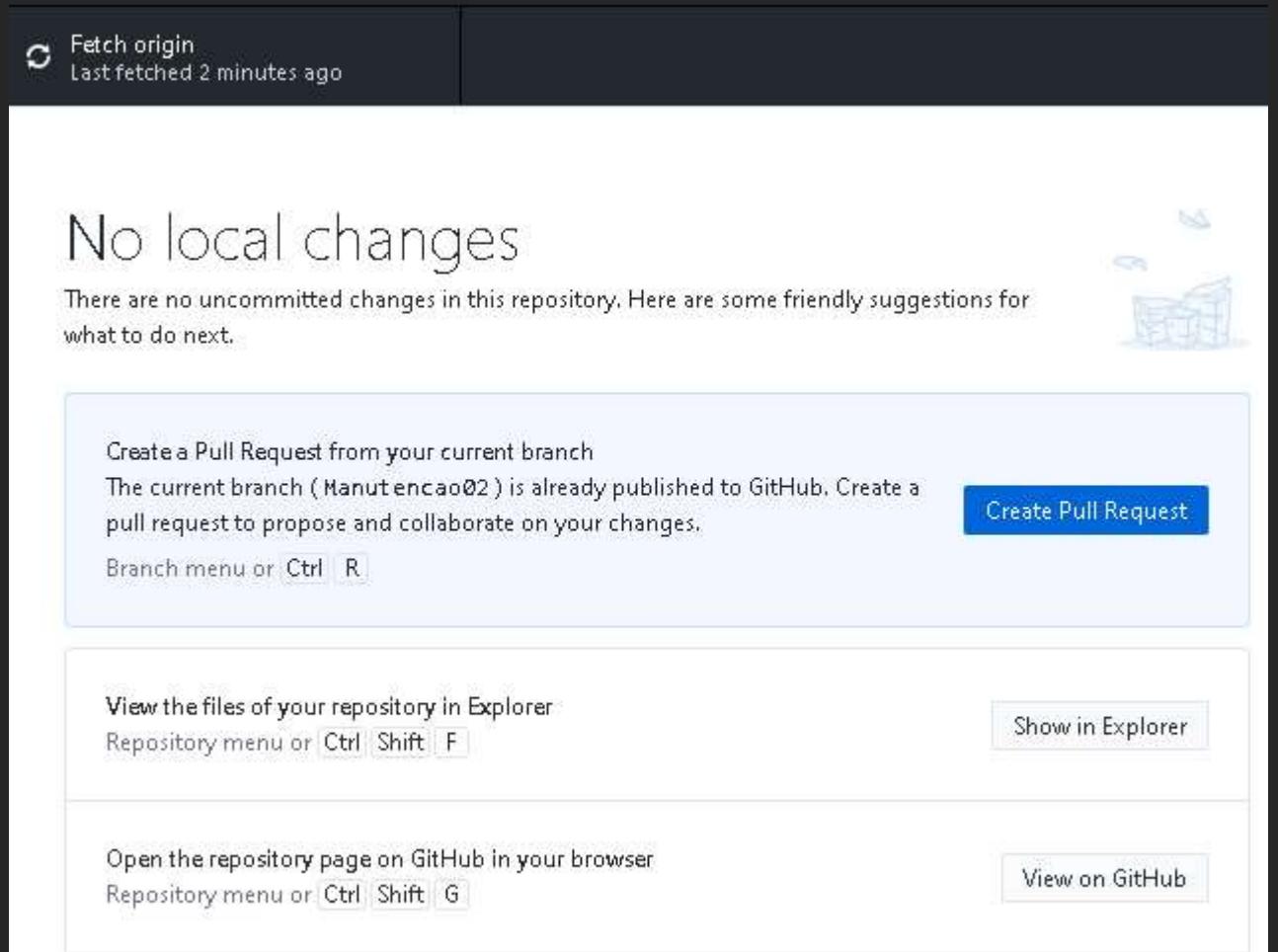
Show in Explorer

View on GitHub

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

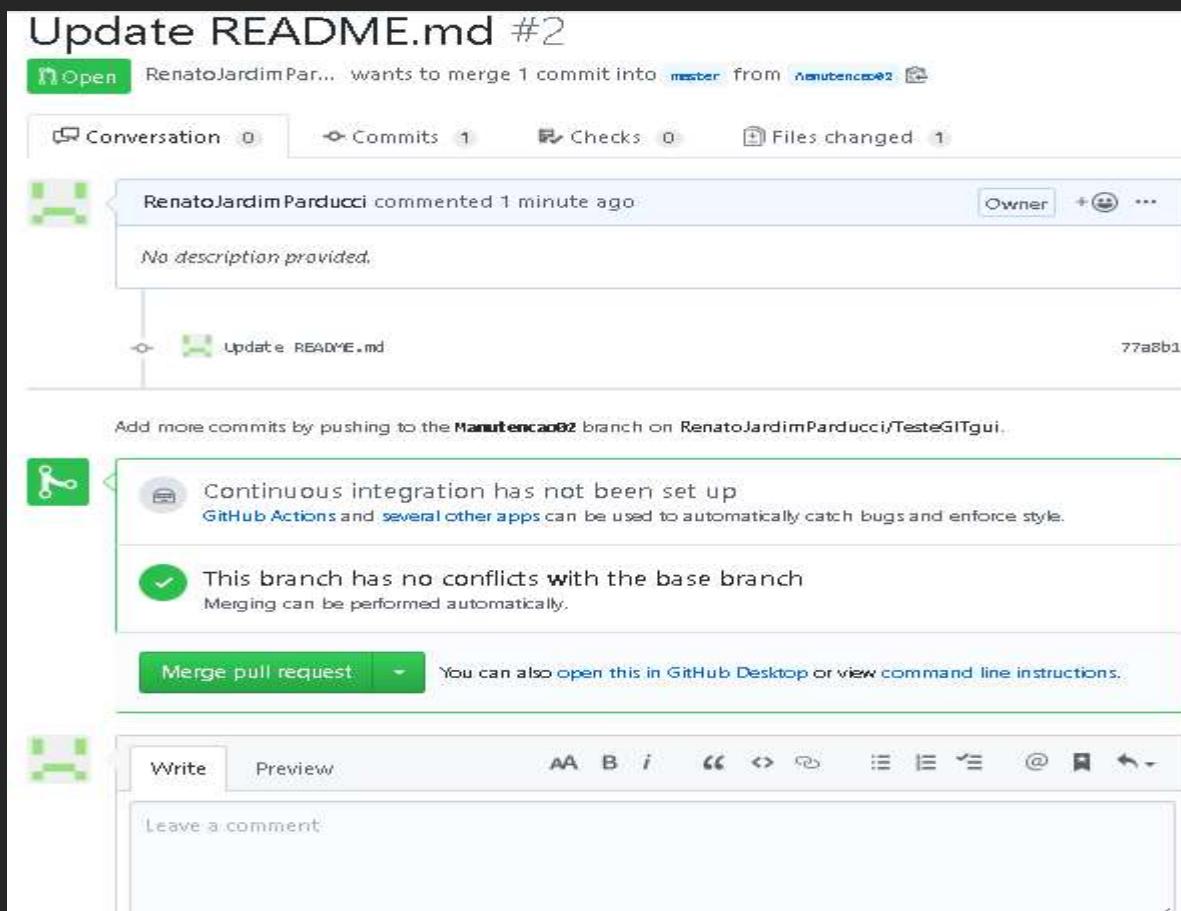
As edições são refletidas do GitGui para o GitHub!

Crie a Pull Request remota via GITHUB Client para atualizar a Master remota.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Ao solicitar a Pull Request, o Git Hub é acionado. Você fará Login e confirmará o Merge/Pull Request!



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Com esses procedimentos, você consegue fazer Check out e Check in (Push) , mantendo sincronia entre o ambiente remoto dos fontes e o seu ambiente local de trabalho.

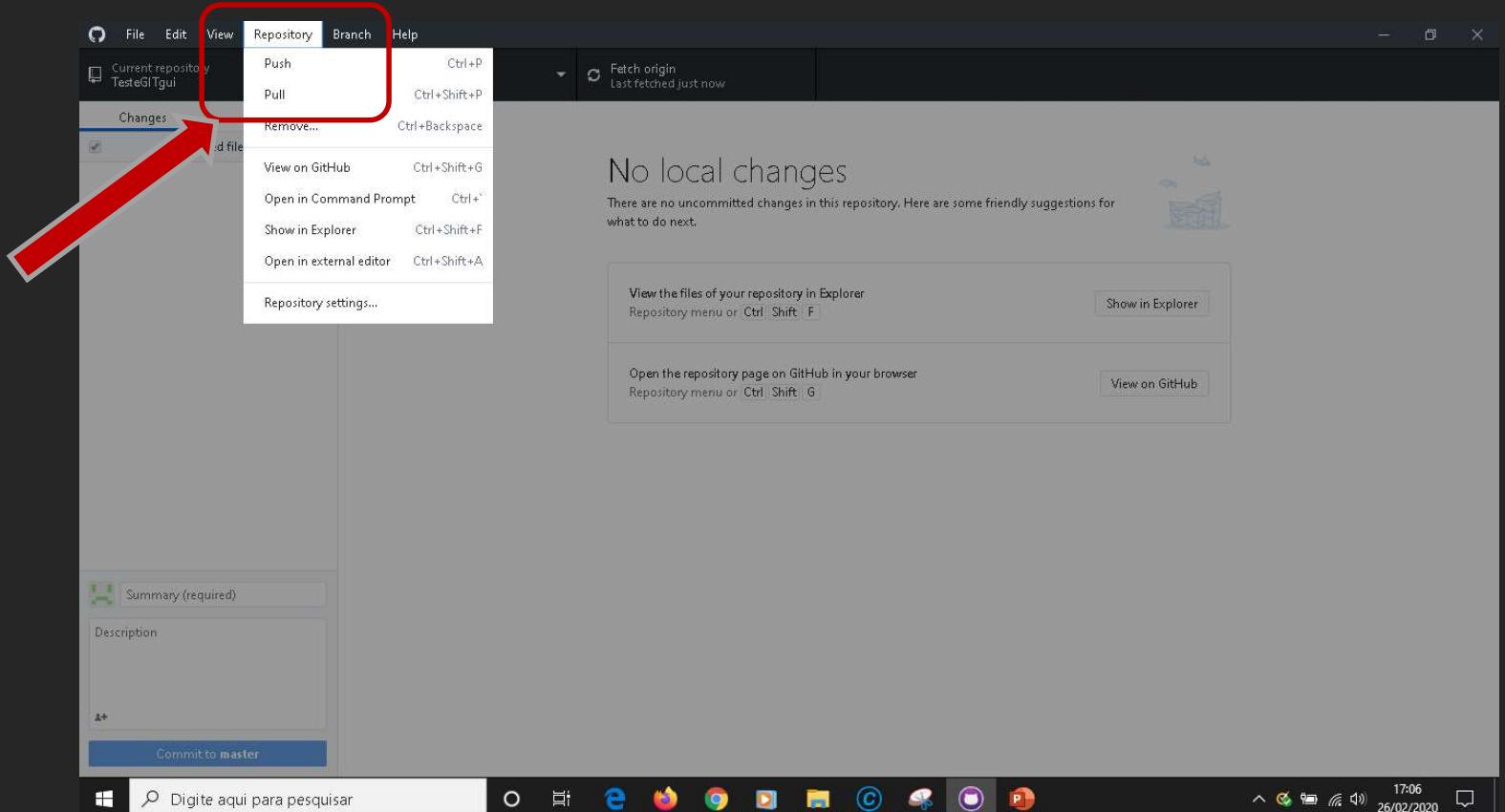
Após os Pushes, confira sempre se o GitHub.com foi atualizado corretamente.

The screenshot shows a GitHub repository interface. At the top, there are statistics: 5 commits, 3 branches, 0 packages, 0 releases, and 1 contributor. Below this, a navigation bar includes 'Branch: master' with a dropdown, 'New pull request', and buttons for 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. A list of recent activity shows a merge pull request from 'RenatoJardimParducci' to 'RenatoJardimParducci/Manutencao02' and a commit 'Update README.md' by 'RenatoJardimParducci' made 1 minute ago. The main content area displays the 'README.md' file with the following text:

```
TesteGITgui
Texto NOVO MESMMO!!!
```

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

IMPORTANTE: o fato de você ter atualizado a Branch e Master no GITHUB não implica que a Master Local tenha sido atualizada.
Para atualizar a Master Local, faça um Pull do Repositório.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Outra forma de usar o GIT é na linha de prompt do sistema operacional (opção que muitas vezes é a favorita de alguns desenvolvedores).

Para isso, instale o GIT CMD na sua máquina e ao rodá-lo, a tela de prompt do sistema operacional aparecerá e você poderá executar comandos nesse prompt.

A lista de comandos que podem ser usados via prompt estão relacionados a seguir.



GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Lista de comandos do GIT CMD (considerando que o repositório já existe no GIT HUB):

Setar usuário

```
git config --global user.name "Leonardo Comelli"
```

Criando um novo branch

```
git branch bug-123
```

Setar email

```
git config --global user.email leonardo@software-ltda.com.br
```

Trocando para um branch existente

```
git checkout bug-123
```

Clonar um repositório remoto já existente

```
git clone git@github.com:leocomelli/curso-git.git
```

Apagando um branch

```
git branch -d bug-123
```

Verificar estado dos arquivos/diretórios

```
git status
```

Listar branches

Listar branches

```
git branch
```

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Lista de comandos do GIT CMD (considerando que o repositório já existe no GIT HUB):

Atualizar os arquivos no branch atual

```
git pull
```

Comitar arquivo/diretório

Comitar um arquivo

```
git commit meu_arquivo.txt
```

Adicionar arquivo/diretório (staged area)

Adicionar um arquivo em específico

```
git add meu_arquivo.txt
```

Comitar vários arquivos

```
git commit meu_arquivo.txt meu_outro_arquivo.txt
```

Adicionar um diretório em específico

```
git add meu_diretorio
```

Comitar informando mensagem

```
git commit meu_arquivo.txt -m "minha mensagem de commit"
```

Adicionar todos os arquivos/diretórios

```
git add .
```

Exibir histórico modificação de um arquivo

```
git log --diff-filter=M -- <caminho_do_arquivo>
```

GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Lista de comandos do GIT CMD (considerando que o repositório já existe no GIT HUB):

Remover arquivo/diretório

Desfazendo alteração local (staging area)

Remover arquivo

Este comando deve ser utilizado quando o arquivo já foi adicionado na **staged area**.

```
git rm meu_arquivo.txt
```

```
git reset HEAD meu_arquivo.txt
```

Remover diretório

Se o resultado abaixo for exibido, o comando reset *não* alterou o diretório de trabalho.

```
git rm -r diretorio
```

```
Unstaged changes after reset:  
M     meu_arquivo.txt
```

Visualizar histórico

Enviar arquivos/diretórios para o repositório remoto

Exibir histórico

O primeiro **push** de um repositório deve conter o nome do repositório remoto e o branch.

```
git log
```

```
git push -u origin master
```

Exibir histórico de um arquivo específico

Os demais **pushes** não precisam dessa informação

```
git log -- <caminho_do_arquivo>
```

```
git push
```

ASSISTA OS VÍDEOS NO CANAL DO PROFESSOR

https://youtu.be/aSpdQ_82S9k

**Conteúdo didático complementar - GIT-CMD**



D Ú V I D A S

Boa diversão!

Agora você conhece um jeito inteligente de controlar suas versões de arquivos com total segurança!

Aproveite esse ambiente em todas as suas disciplinas de programação!

**EXEMPLIFICAÇÃO DE COMO FERRAMENTAS E PROCESSOS IMPACTAM A
QUALIDADE E GOVERNANÇA – GIT HUB**

FIM

**PROFESSOR:
RENATO JARDIM PARDUCCI**

PROFRENATO.PARDUCCI@FIAP.COM.BR