

IFRN

INFOWEB – PROJETO INTEGRADOR

GitHub – Guia de Uso 01

Prof. Gilbert Azevedo

Prof. George Azevedo

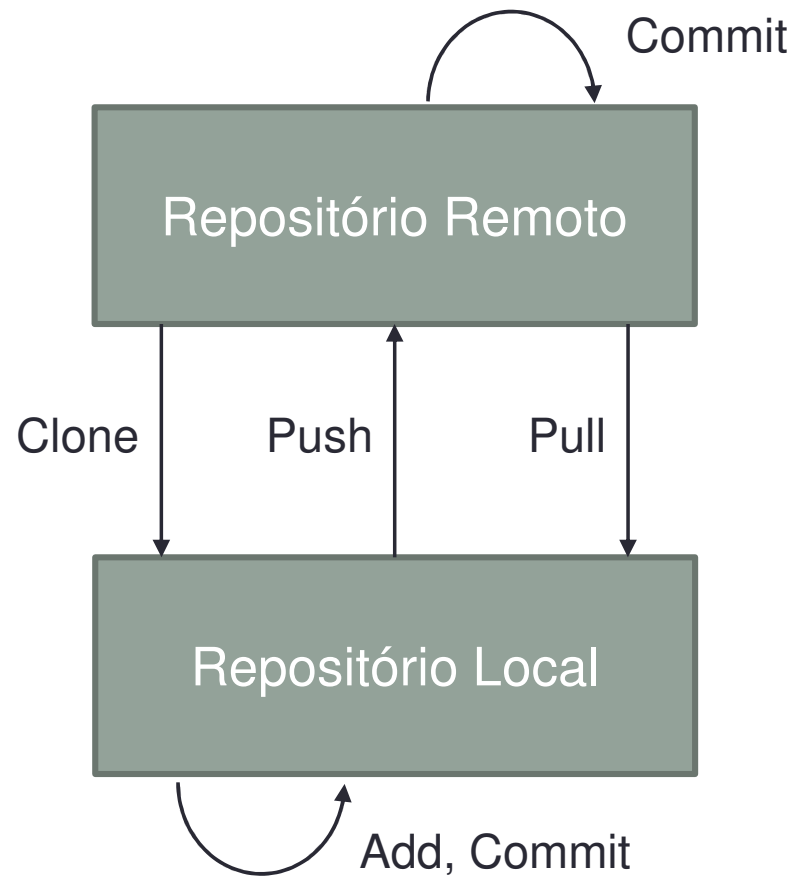
Conteúdo

- O que é GitHub
- Passo a passo: Utilização do GitHub no Terminal do Windows

GitHub

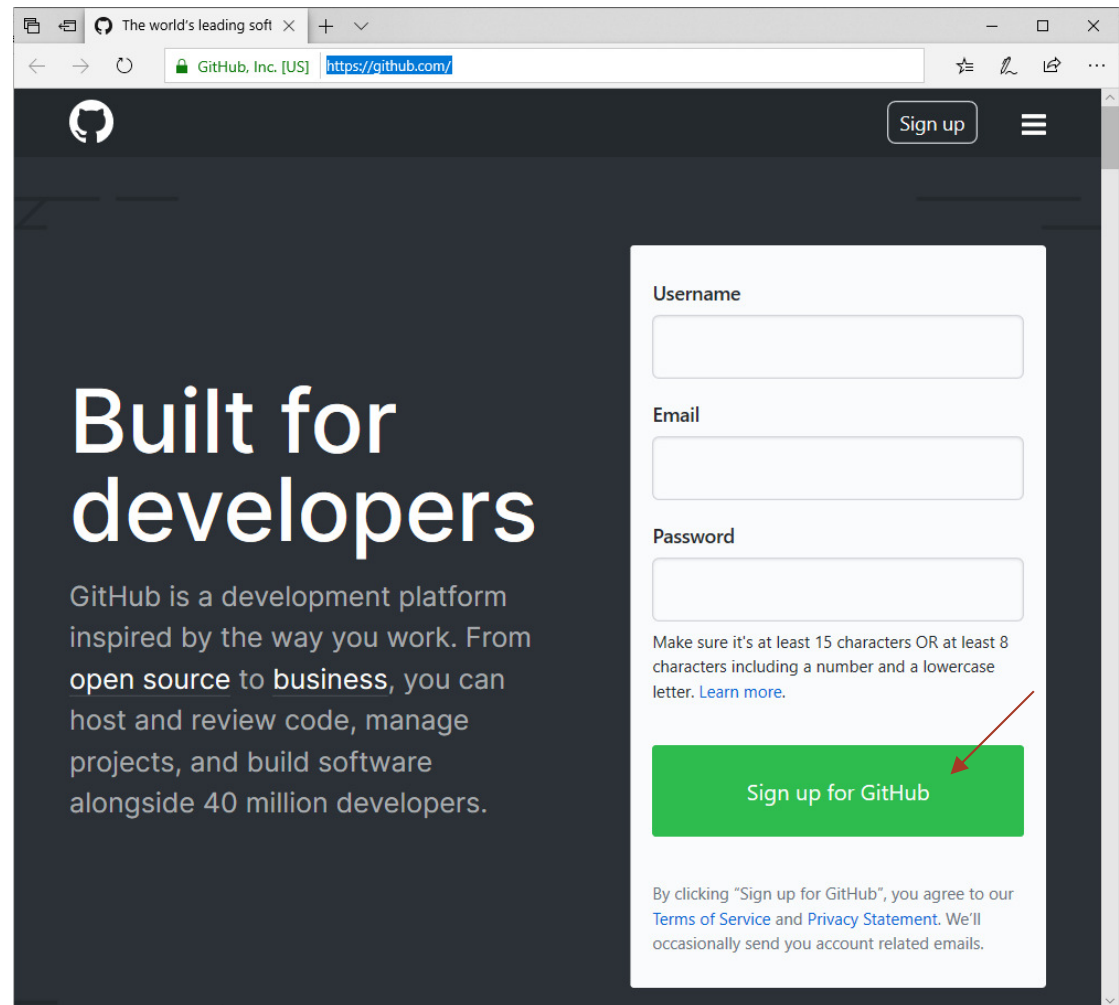
- O GitHub é uma plataforma de hospedagem utilizada para controle de versão e desenvolvimento colaborativo
 - Repositório de documentos e código-fonte
 - Controle de versão
 - Registro de projetos e tarefas
 - Fórum de discussão
 - Relatórios de atividades

GitHub – Repositórios



Registro no GitHub

- Registro no Site
 - <https://github.com/>



The screenshot shows the GitHub website in a web browser. The browser's address bar displays "https://github.com/". The page features a dark background with the GitHub logo in the top left and a "Sign up" button in the top right. The main heading reads "Built for developers". Below this, a paragraph describes GitHub as a development platform. On the right side, there is a white sign-up form with fields for "Username", "Email", and "Password". A red arrow points to the green "Sign up for GitHub" button. At the bottom of the form, there is a disclaimer about agreeing to the Terms of Service and Privacy Statement.

Username

Email

Password

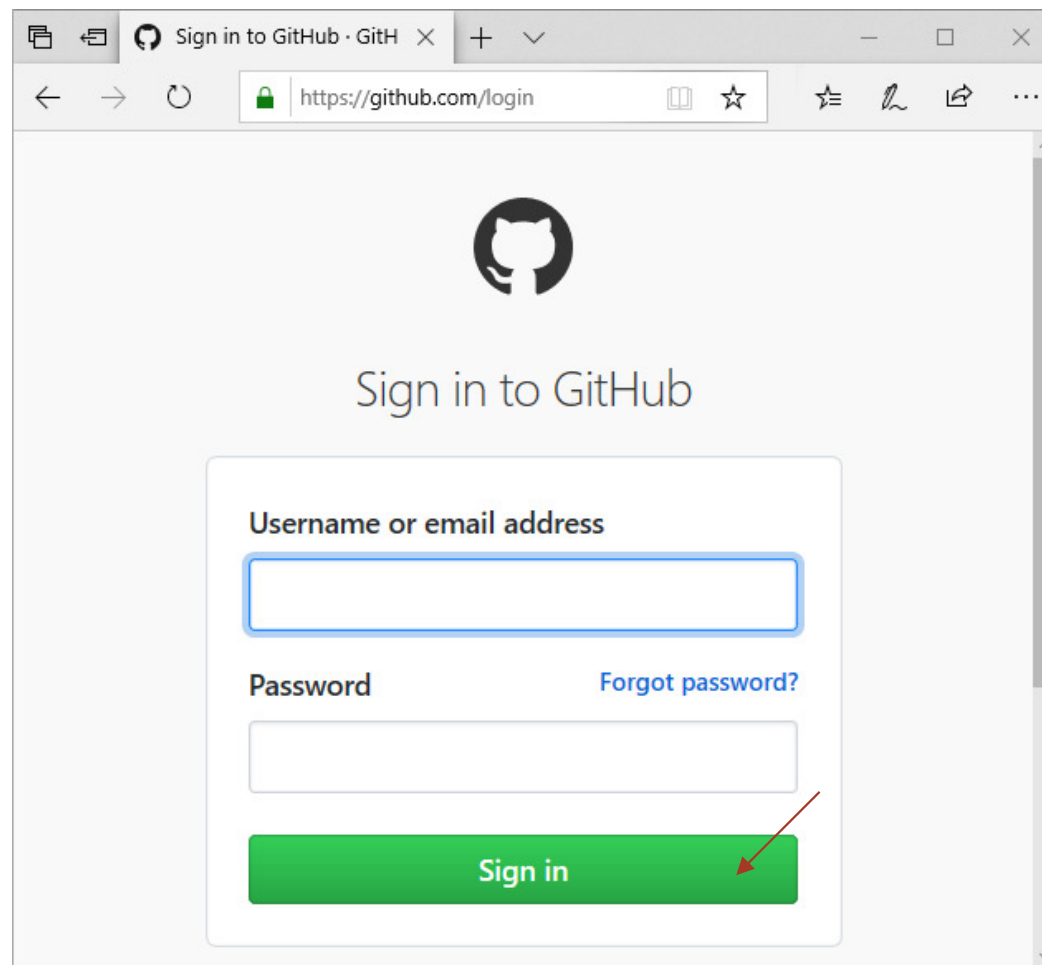
Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.

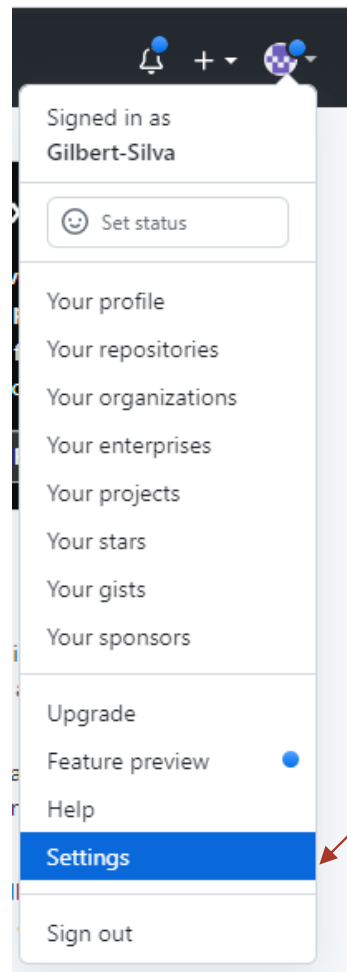
Login no GitHub

- Login – Utilize seu e-mail para entrar na plataforma



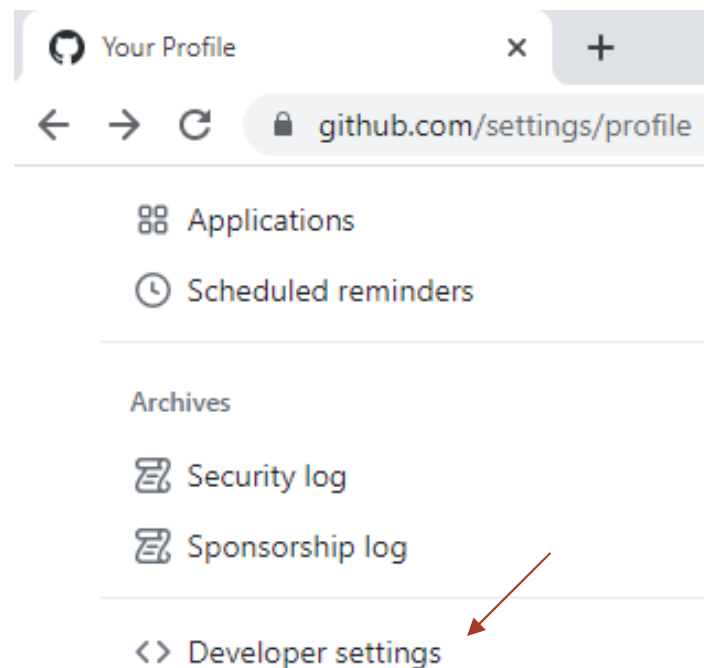
Criação do Token

- Após Logar, selecione *Settings* (Configurações), no menu do usuário



Criação do Token

- Acesse *Developer Settings* (configurações do desenvolvedor) no final da página



Criação do Token

- Expanda o item de menu *Personal access tokens*
- Selecione *Tokens (classic)*
- Selecione *Generate new token*
- Selecione *Generate new token (classic)*
- Informe sua senha para confirmar a geração do *token*

Settings / Developer settings

Personal access tokens (classic)

Tokens you have generated that can be used to access the Git

Token Name	Expiration	Actions
Teste — repo	Expires on Wed, Apr 26 2023.	Generate new token Revoke all
		Delete

Generate new token Beta
Fine-grained, repo-scoped

Generate new token (classic)
For general use

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Criação do Token

- Informe um nome para seu *token* e selecione a opção *repo*
- Selecione *Generate token* no final da página

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

MeuToken

What's this token for?

Expiration *

30 days

The token will expire on Thu, Apr 27 2023

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

- | | |
|-----------------------------------------------------|--------------------------------------|
| <input checked="" type="checkbox"/> repo | Full control of private repositories |
| <input checked="" type="checkbox"/> repo:status | Access commit status |
| <input checked="" type="checkbox"/> repo_deployment | Access deployment status |
| <input checked="" type="checkbox"/> public_repo | Access public repositories |
| <input checked="" type="checkbox"/> repo:invite | Access repository invitations |
| <input checked="" type="checkbox"/> security_events | Read and write security events |

Generate token

Cancel

Armazene o Token

- Armazene seu *token*, você vai utilizá-lo no lugar da senha

Personal access tokens (classic)

Generate new token ▾

Revoke all

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!



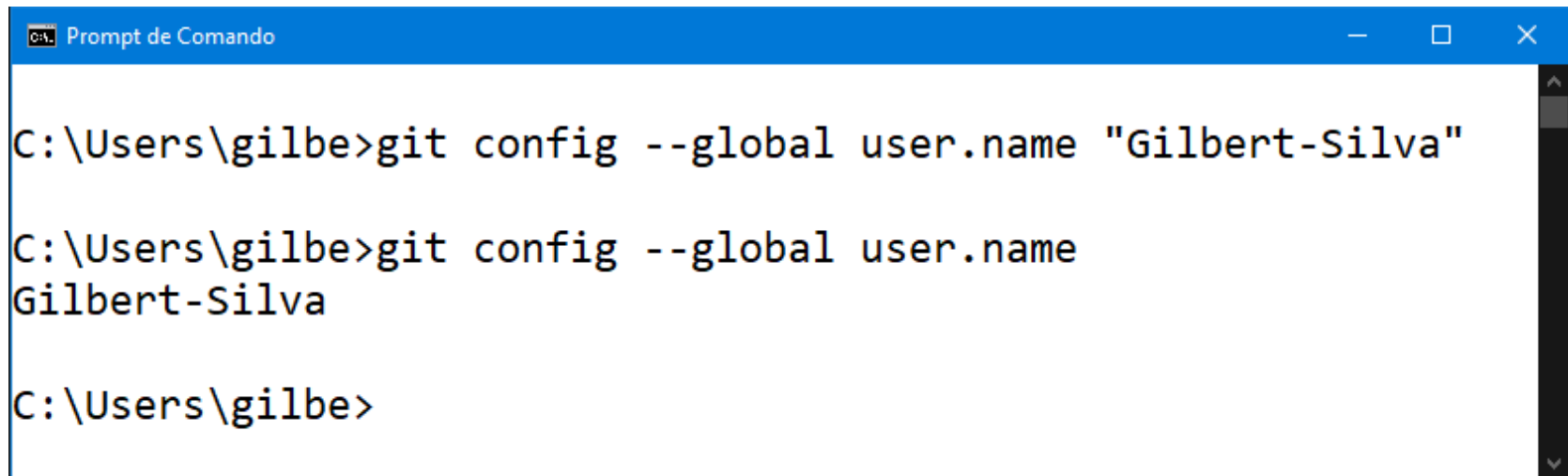
[Redacted token]



Delete

Configuração do Usuário

- Configure seu usuário no terminal do SO
- Abra o aplicativo *prompt de comando*
- Rode os comandos abaixo para configurar seu usuário e e-mail:
 - *git config --global user.name "nome do usuário"*
 - *git config --global user.email seuemail@email.com*



```

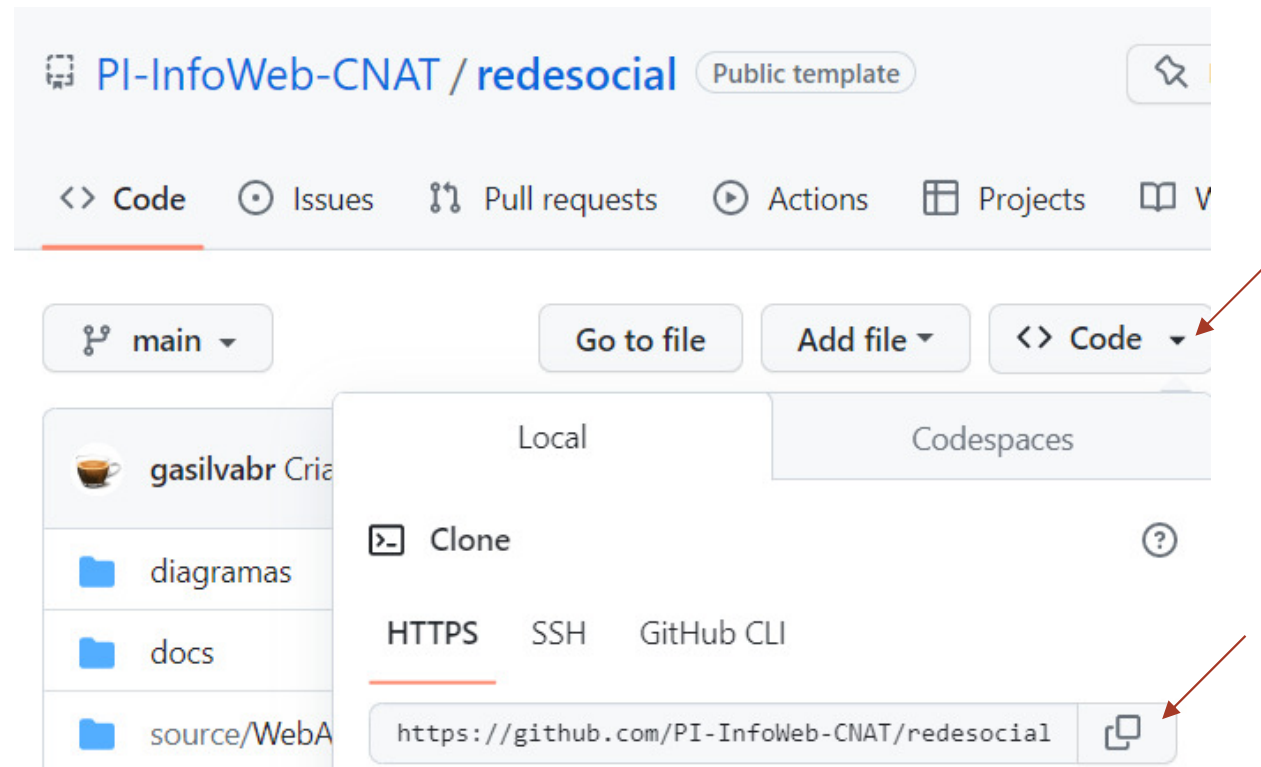
C:\Users\gilbe>git config --global user.name "Gilbert-Silva"

C:\Users\gilbe>git config --global user.name
Gilbert-Silva

C:\Users\gilbe>
```

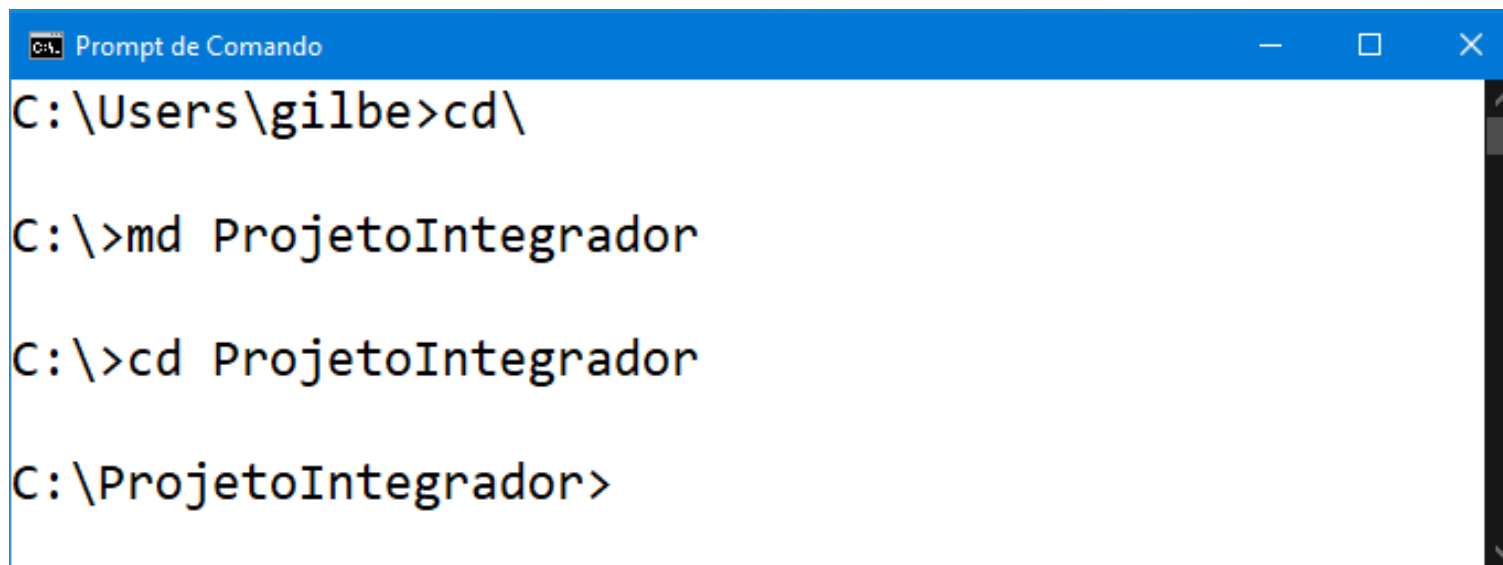
Clonagem do Repositório

- O repositório do projeto, com os documentos iniciais e a aplicação do VS, pode ser clonado com o comando *git clone*
- Copie o endereço *https* do repositório a ser clonado



Clonagem do Repositório

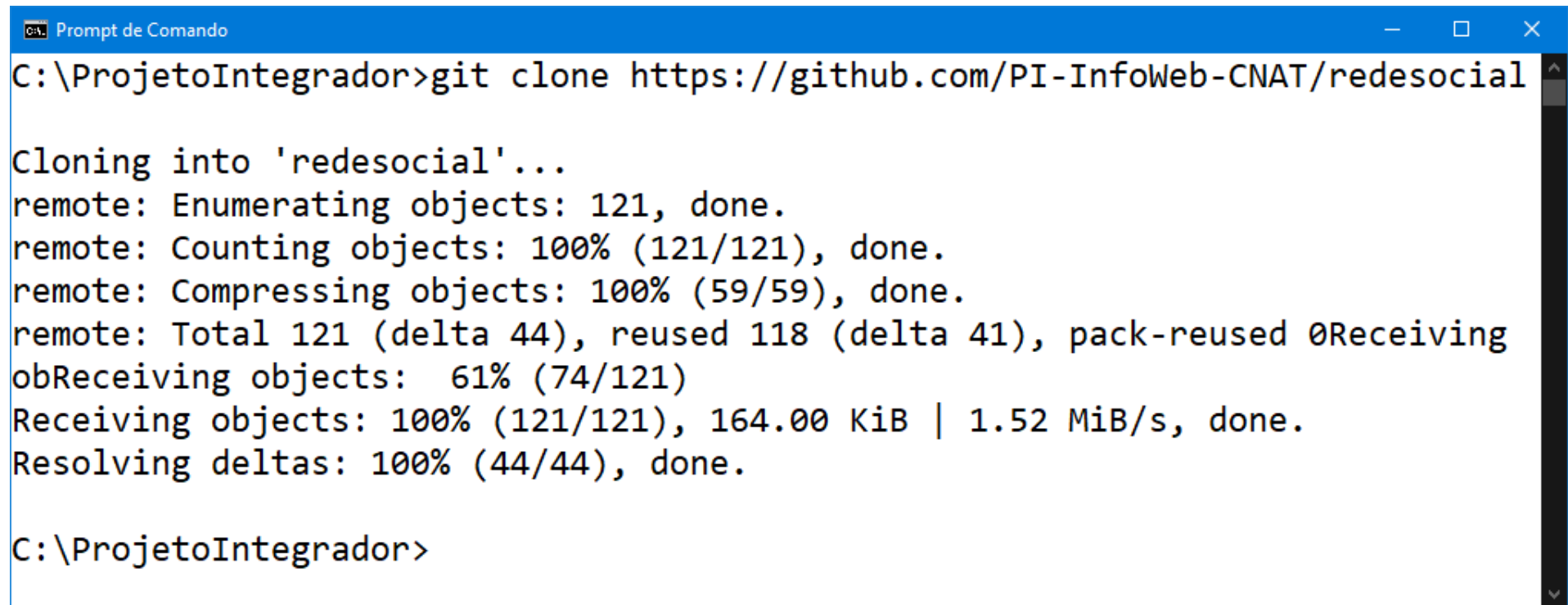
- No prompt de comando, execute as seguintes instruções:
- `cd\`
- `md ProjetoIntegrador`
- `cd ProjetoIntegrador`



```
C:\Users\gilbe>cd\  
  
C:\>md ProjetoIntegrador  
  
C:\>cd ProjetoIntegrador  
  
C:\ProjetoIntegrador>
```

Clonagem do Repositório

- Execute o comando *git clone* para clonar o repositório
- *git clone https://github.com/PI-InfoWeb-CNAT/<seu_repositório>*

A screenshot of a Windows Command Prompt window titled "Prompt de Comando". The window shows the execution of the command `git clone https://github.com/PI-InfoWeb-CNAT/redesocial` from the directory `C:\ProjetoIntegrador`. The output displays the progress of cloning the repository, including object enumeration, counting, compressing, and receiving. The final prompt is `C:\ProjetoIntegrador>`.

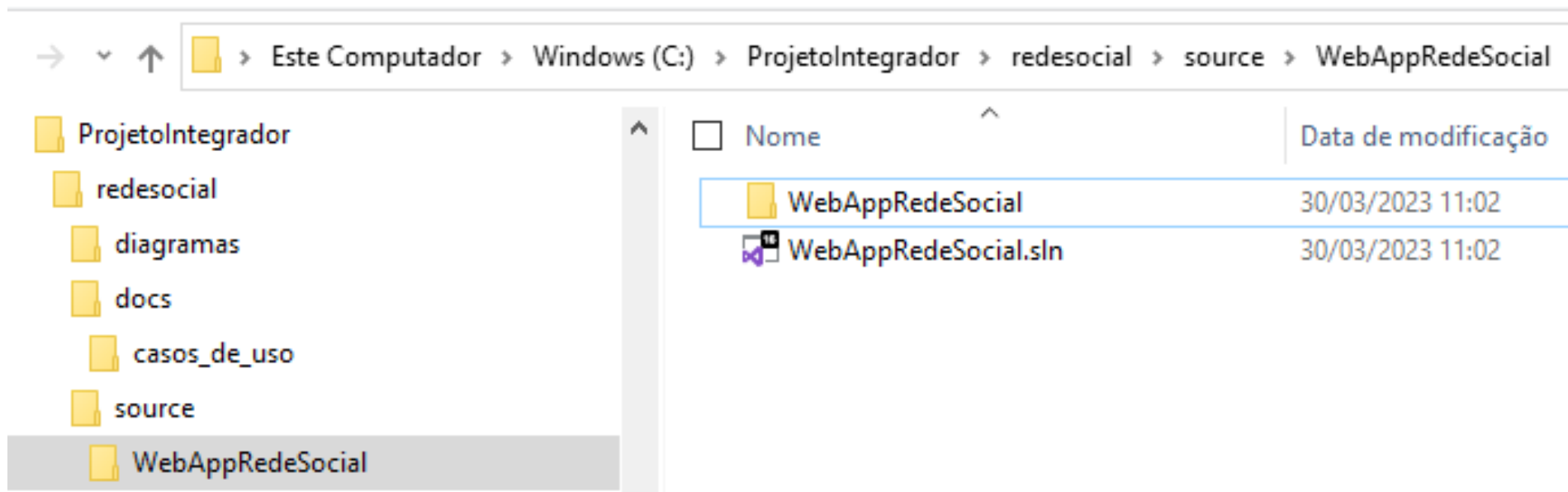
```
C:\ProjetoIntegrador>git clone https://github.com/PI-InfoWeb-CNAT/redesocial

Cloning into 'redesocial'...
remote: Enumerating objects: 121, done.
remote: Counting objects: 100% (121/121), done.
remote: Compressing objects: 100% (59/59), done.
remote: Total 121 (delta 44), reused 118 (delta 41), pack-reused 0Receiving
obReceiving objects: 61% (74/121)
Receiving objects: 100% (121/121), 164.00 KiB | 1.52 MiB/s, done.
Resolving deltas: 100% (44/44), done.

C:\ProjetoIntegrador>
```

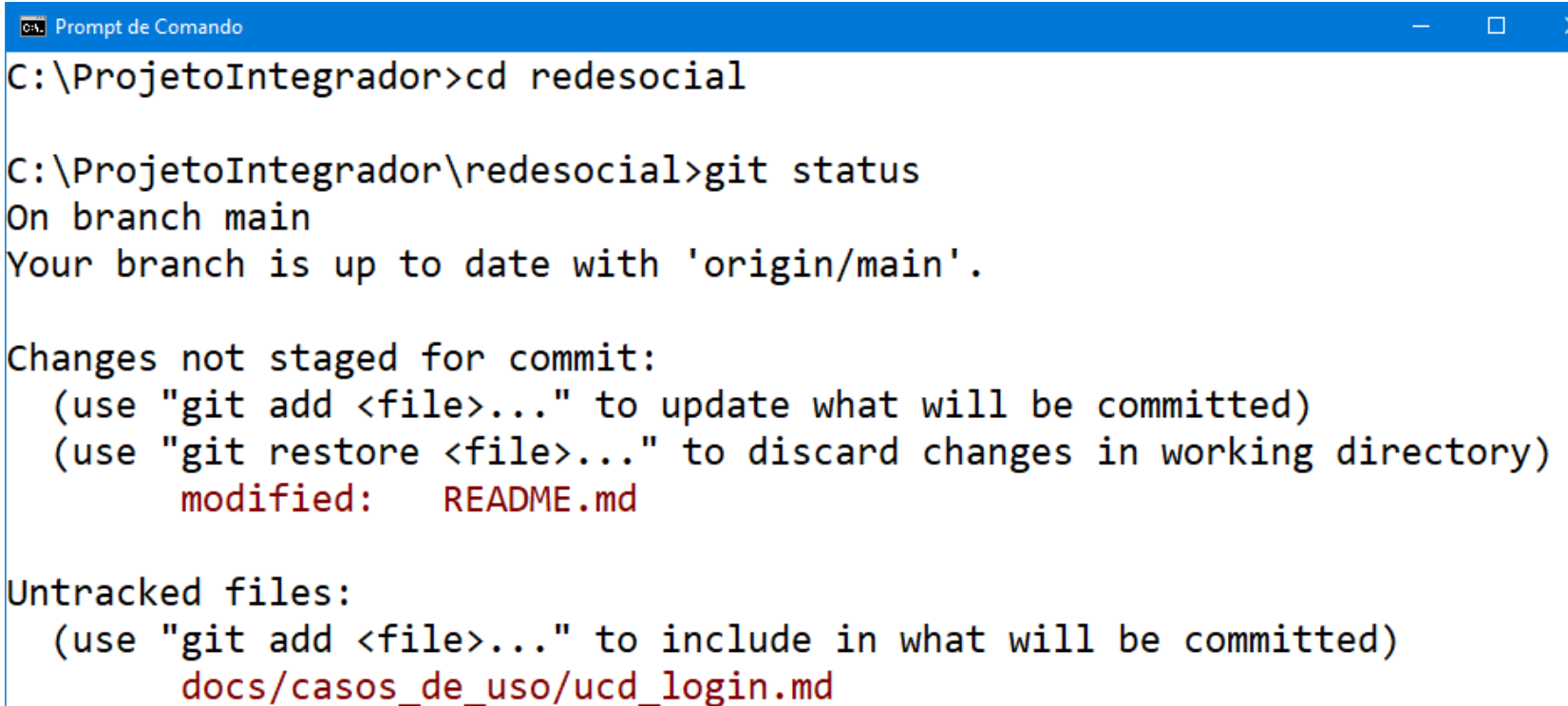
Arquivos do Projeto

- Documentos iniciais do projeto e aplicação do VS estarão na pasta ProjetoIntegrador (repositório local)
- Todas as alterações do projeto devem ser salvas na pasta do repositório clonado
- A aplicação do VS pode ser iniciada a partir do arquivo *sln*



Verificando Alterações

- As alterações realizadas no repositório local podem ser visualizadas com *git status* no *prompt*. Execute as seguintes instruções:
- *cd <seu_repositório>*
- *git status*



```
C:\ProjetoIntegrador>cd redesocial

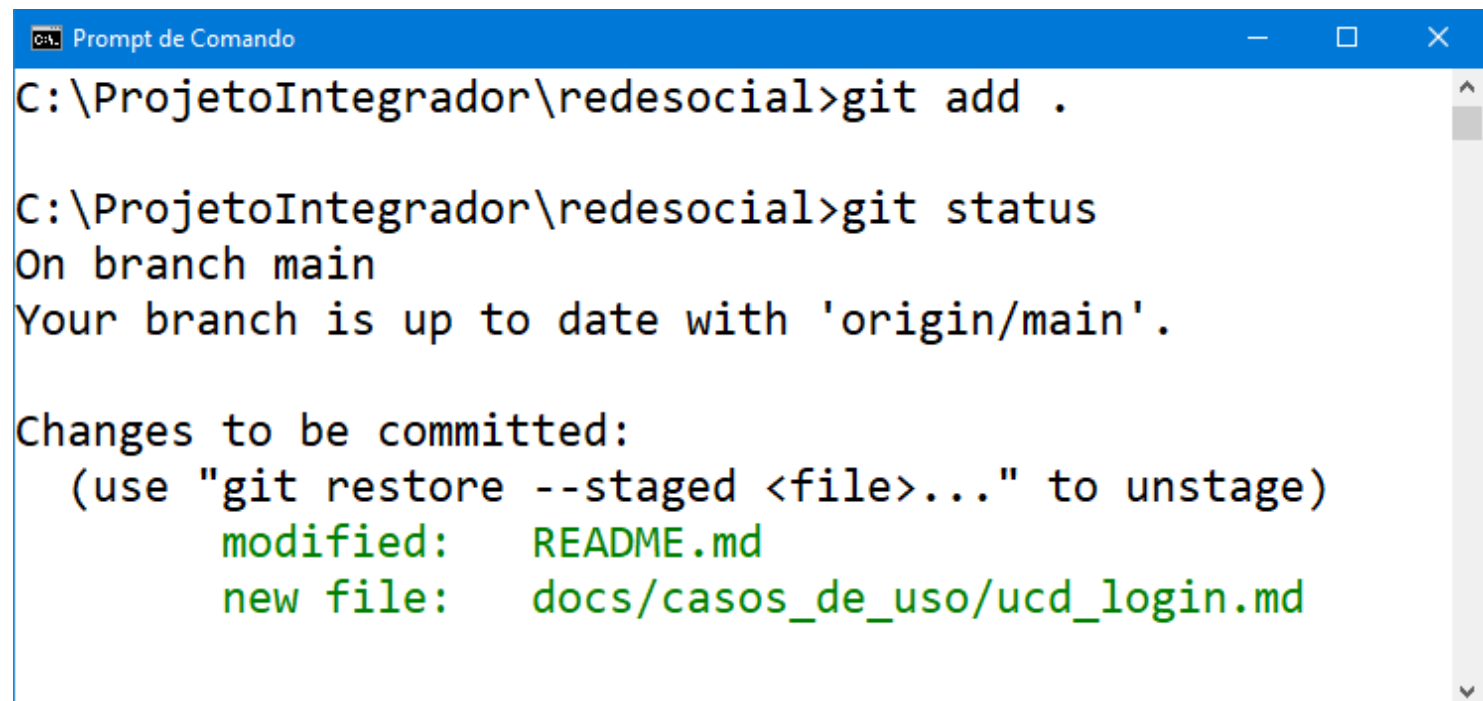
C:\ProjetoIntegrador\redesocial>git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        docs/casos_de_uso/ucd_login.md
```

Adicionando as Alterações

- As alterações precisam ser adicionadas no repositório local antes de serem enviadas para o GitHub. Execute as instruções abaixo para adicionar as modificações que serão confirmadas:
- *git add .*
- *git status*



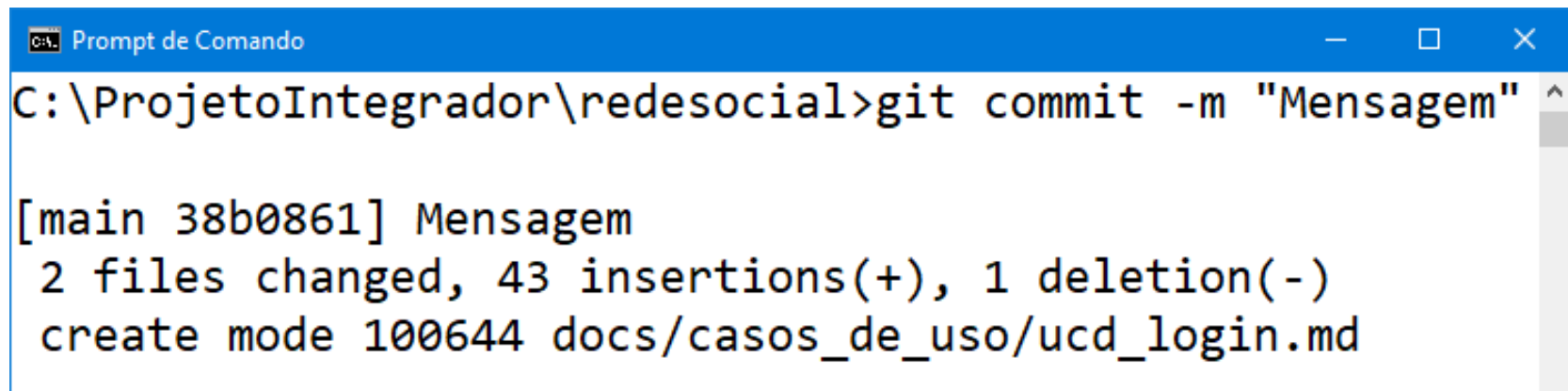
```
C:\ProjetoIntegrador\redesocial>git add .

C:\ProjetoIntegrador\redesocial>git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   README.md
        new file:   docs/casos_de_uso/ucd_login.md
```

Confirmando as Alterações

- As alterações precisam ainda ser confirmadas no repositório local antes do envio. Execute a instrução abaixo para confirmar as alterações, fornecendo uma mensagem para as alterações realizadas:
- *git commit -m "Mensagem"*

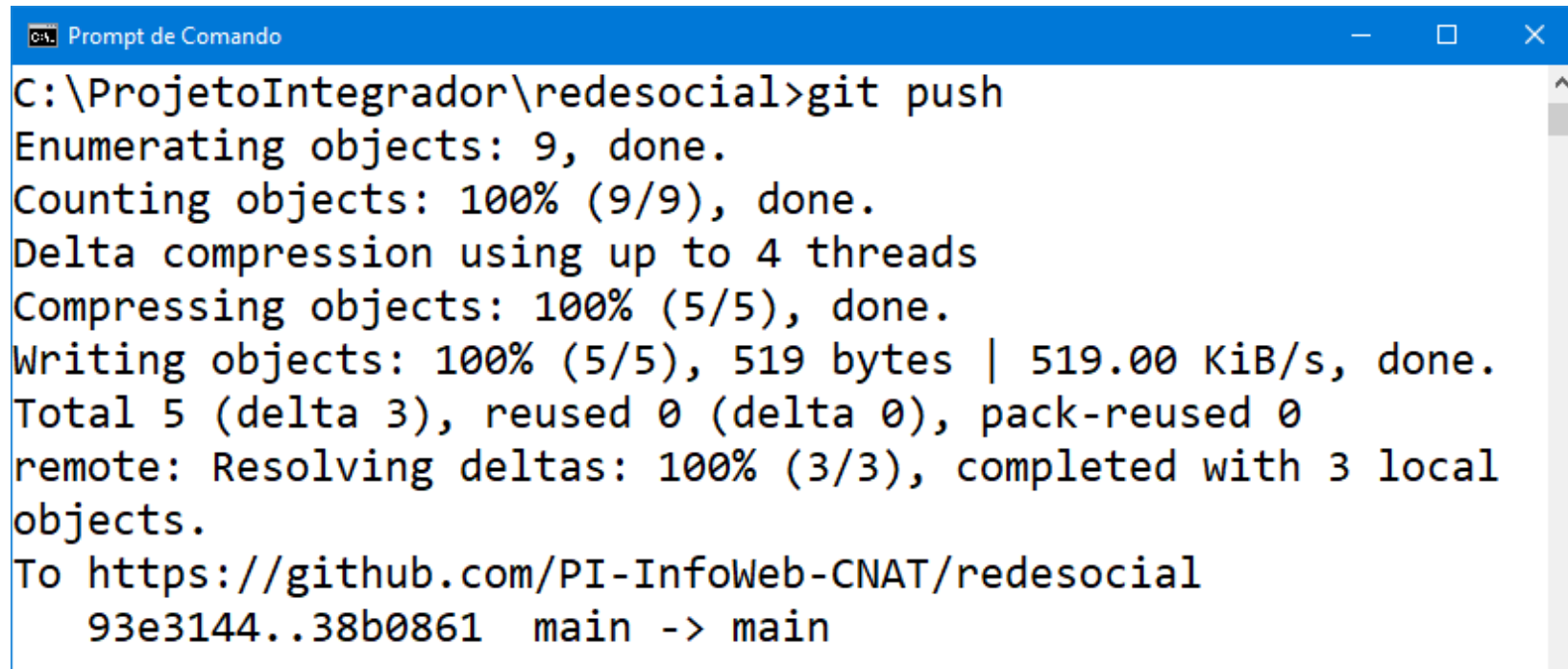


```
C:\ProjetoIntegrador\redesocial>git commit -m "Mensagem"

[main 38b0861] Mensagem
 2 files changed, 43 insertions(+), 1 deletion(-)
 create mode 100644 docs/casos_de_uso/ucd_login.md
```

Enviando as Alterações

- Para enviar as alterações confirmadas ao *GitHub*, execute:
- *git push*
- Importante: se for necessário autenticar, utilize o *token* ao invés da sua senha

A screenshot of a Windows Command Prompt window titled "Prompt de Comando". The window has a blue title bar with standard minimize, maximize, and close buttons. The command prompt shows the execution of the 'git push' command in the directory 'C:\ProjetoIntegrador\redesocial'. The output displays the progress of pushing 9 objects to the remote repository, including object enumeration, counting, compression, and writing. It also shows the resolution of deltas and the final commit hash '93e3144..38b0861' for the 'main' branch.

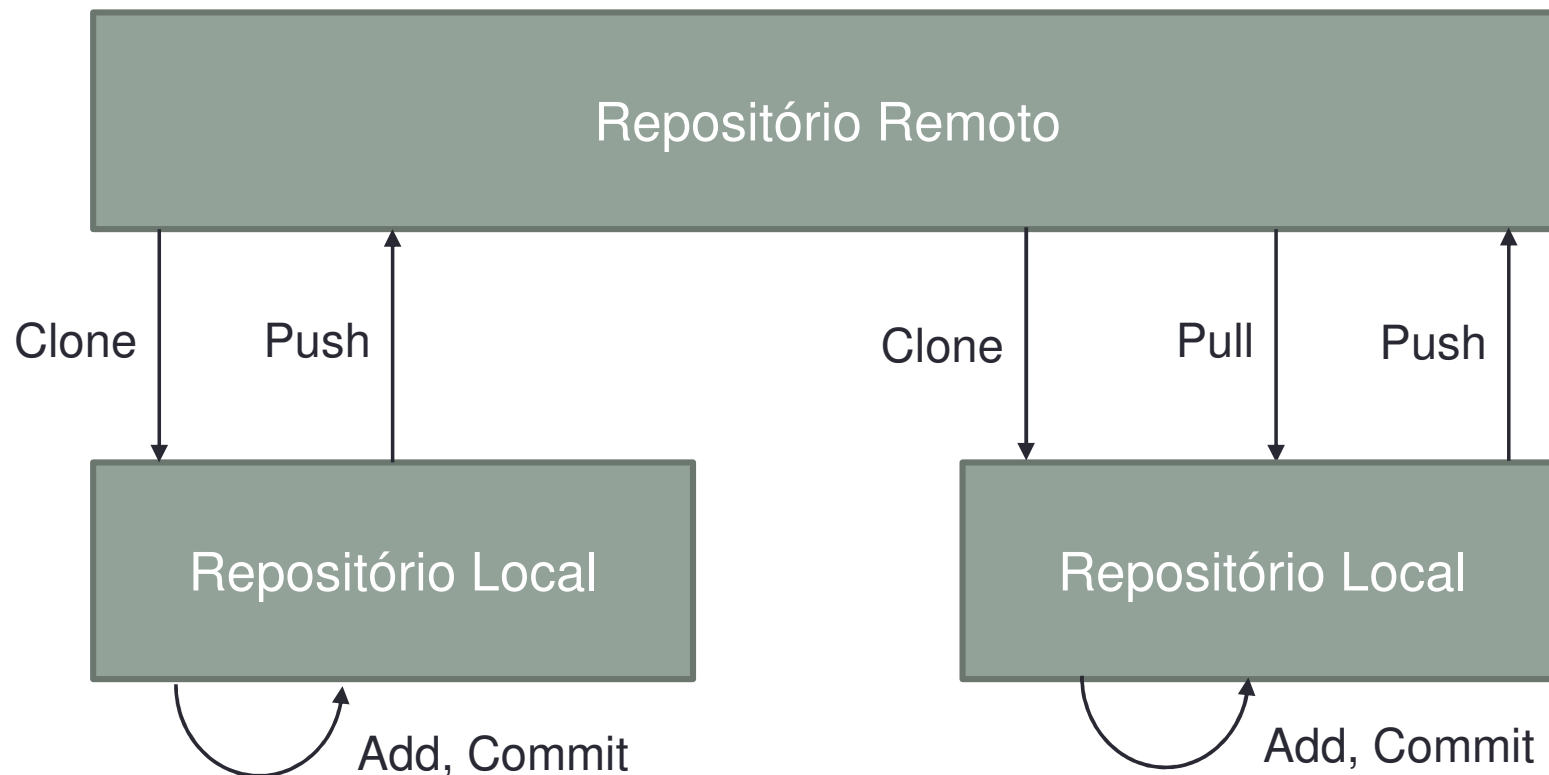
```
C:\ProjetoIntegrador\redesocial>git push
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 519 bytes | 519.00 KiB/s, done.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 3 local
objects.
To https://github.com/PI-InfoWeb-CNAT/redesocial
   93e3144..38b0861  main -> main
```

Fluxo de Trabalho Simplificado

- Esse é o fluxo de trabalho mais simples:
- *git clone*
- *git add .*
- *git commit*
- *git push*
- Não vai funcionar quando duas pessoas clonarem o repositório em máquinas diferentes e realizarem alterações.

Atualização do Repositório Local

- Após um *push*, os outros repositórios locais precisam ser atualizados com um *git pull*
- Conflitos serão abordados em outra aula



Referências

- GitHub
 - <https://github.com/>
- GitHub Guides
 - <https://guides.github.com/activities/hello-world/>
- GitHub Learning Lab
 - <https://lab.github.com/>