

Ministério da Educação

Centro Federal de Educação Tecnológica Celso Suckow da Fonseca

UNED Nova Friburgo

Curso Técnico em Informática Integrado ao Ensino Médio

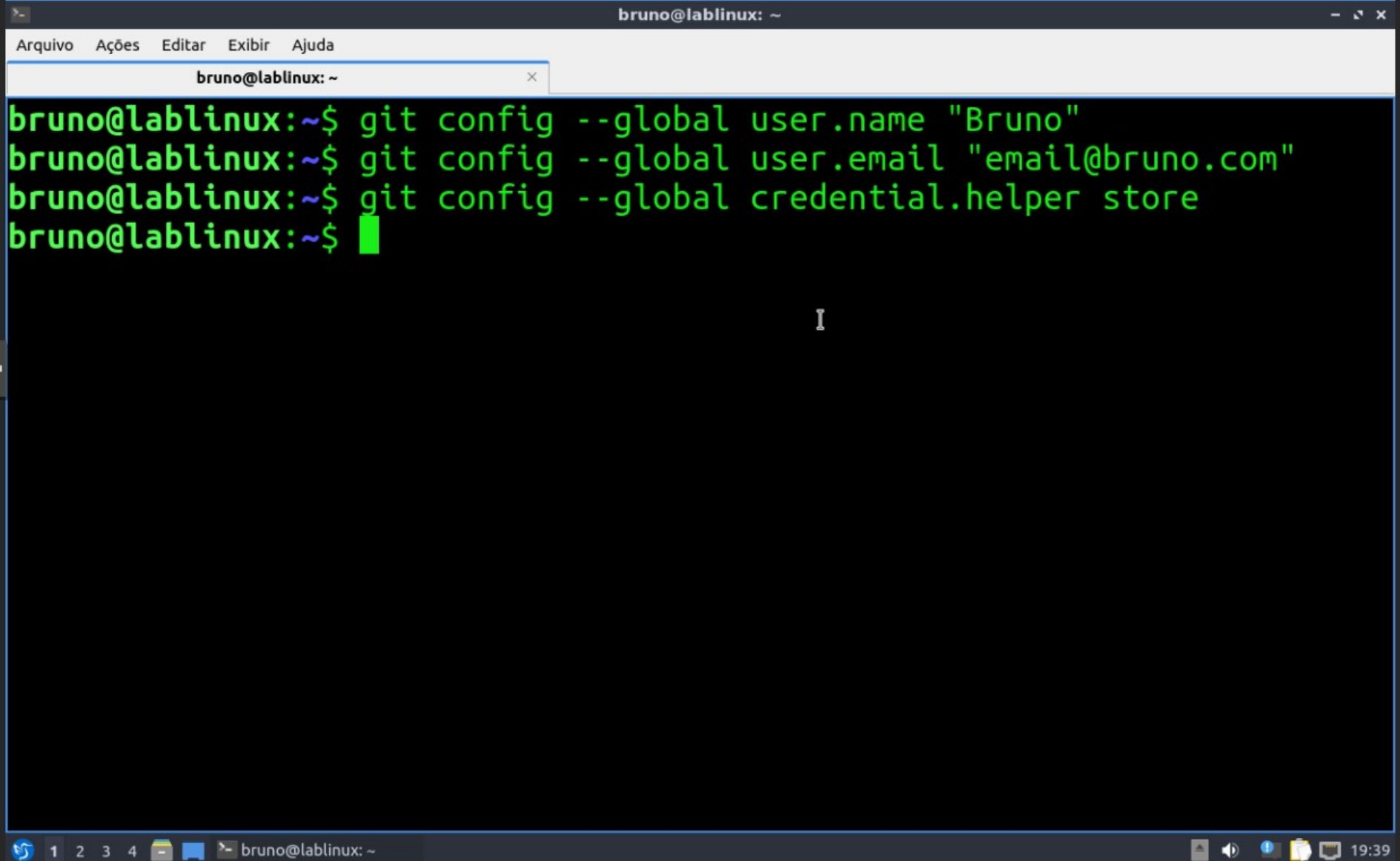
Tutorial: autenticação automática para desenvolvimento de projetos com GIT no github.com



Prof. Bruno Policarpo Toledo Freitas

bruno.freitas@cefet-rj.br

1) Configuração do git pelo terminal

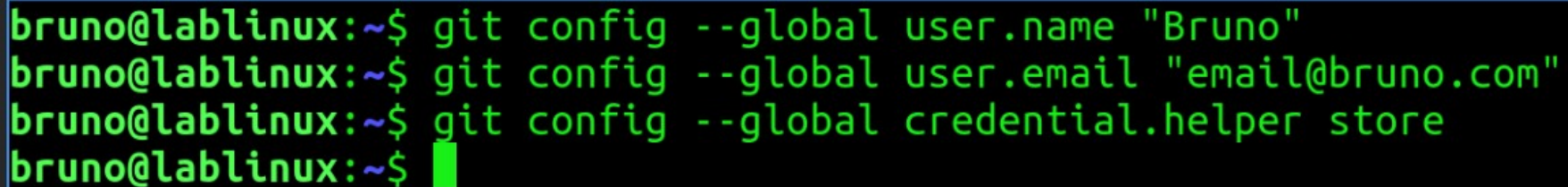


The image shows a terminal window titled "bruno@lablinux: ~". The window has a menu bar with "Arquivo", "Ações", "Editar", "Exibir", and "Ajuda". The terminal content shows the following commands and their prompts:

```
bruno@lablinux:~$ git config --global user.name "Bruno"
bruno@lablinux:~$ git config --global user.email "email@bruno.com"
bruno@lablinux:~$ git config --global credential.helper store
bruno@lablinux:~$
```

The terminal window is part of a VNC viewer, as indicated by the "VNC" logo in the top-left corner of the viewer's interface. The viewer's interface also includes a vertical toolbar on the left with icons for a text editor, a terminal, settings, and a power button. At the bottom of the viewer, there is a taskbar with a blue bar containing icons for a terminal, a file manager, and a network icon, followed by the text "bruno@lablinux: ~". On the right side of the taskbar, there are icons for a window manager, a volume icon, a network icon, and a clock showing "19:39".

1) Configuração do git pelo terminal



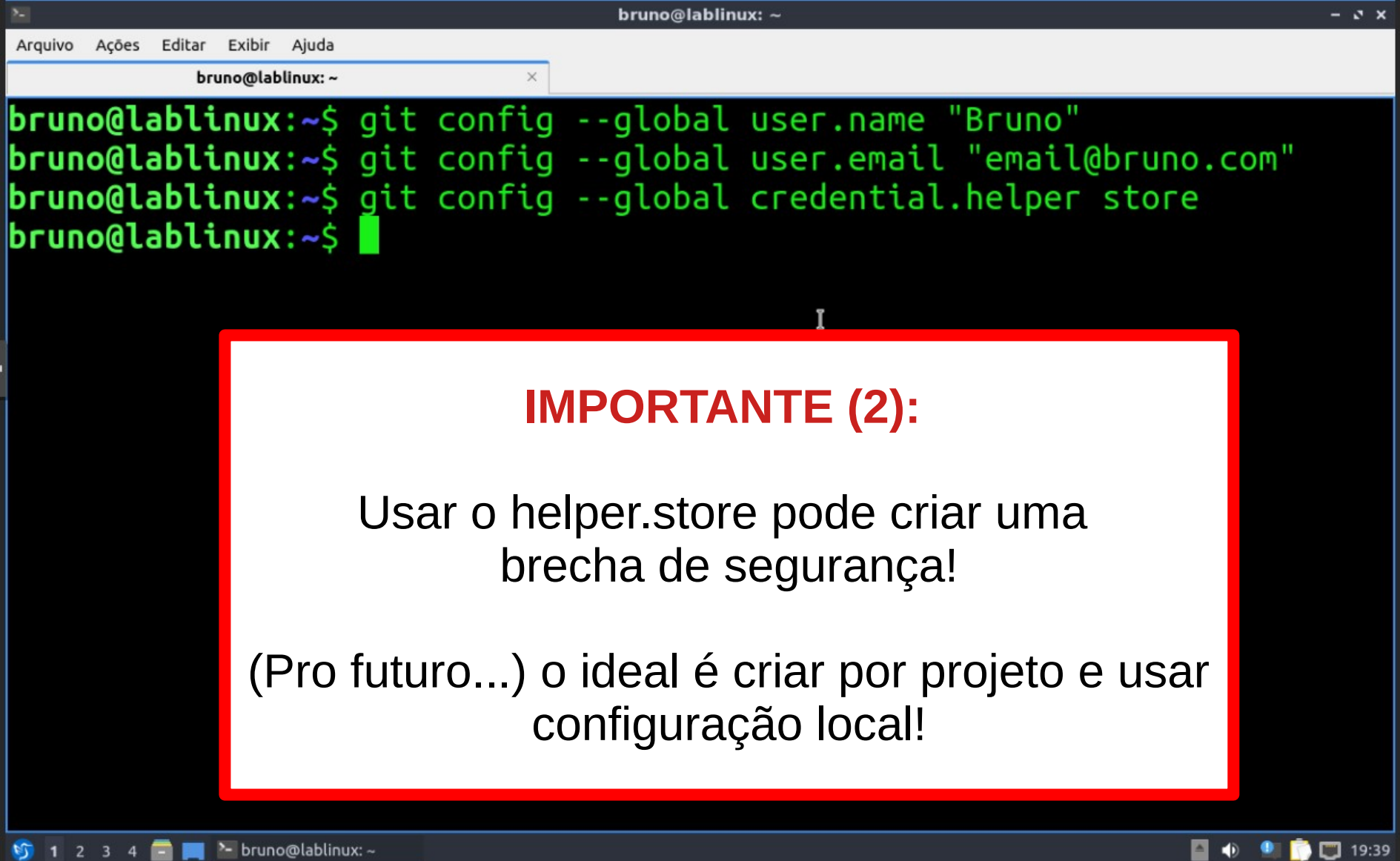
A screenshot of a terminal window titled 'bruno@lablinux: ~'. The window has a menu bar with 'Arquivo', 'Ações', 'Editar', 'Exibir', and 'Ajuda'. The terminal shows four lines of green text representing commands and their prompts:

```
bruno@lablinux:~$ git config --global user.name "Bruno"
bruno@lablinux:~$ git config --global user.email "email@bruno.com"
bruno@lablinux:~$ git config --global credential.helper store
bruno@lablinux:~$
```

IMPORTANTE (1):

Você pode fazer configuração local por projeto omitindo o `--global` e executando o *git config* dentro da pasta o repositório recém-clonado

1) Configuração do git pelo terminal



The image shows a terminal window titled 'bruno@lablinux: ~' with a menu bar containing 'Arquivo', 'Ações', 'Editar', 'Exibir', and 'Ajuda'. The terminal displays the following commands and their output:

```
bruno@lablinux:~$ git config --global user.name "Bruno"
bruno@lablinux:~$ git config --global user.email "email@bruno.com"
bruno@lablinux:~$ git config --global credential.helper store
bruno@lablinux:~$
```

Below the terminal output, a red-bordered box contains the following text:

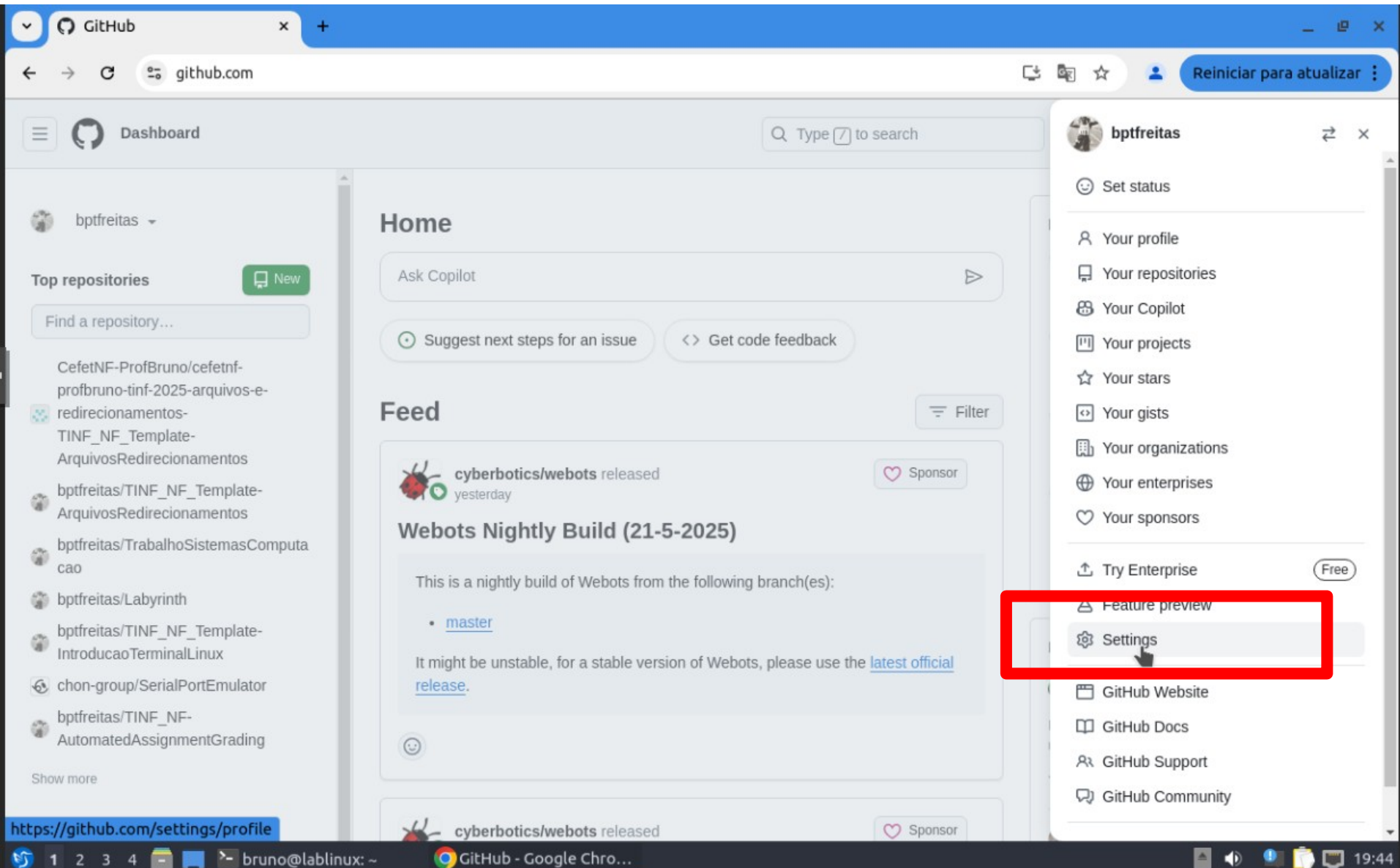
IMPORTANTE (2):

Usar o helper.store pode criar uma brecha de segurança!

(Pro futuro...) o ideal é criar por projeto e usar configuração local!

The terminal window is part of a VNC session, as indicated by the 'VNC' label on the left side of the screen. The bottom of the screen shows a taskbar with icons for a terminal, a file manager, and a clock displaying '19:39'.

2) Fazer login no github.com e ir em “Settings”



The screenshot shows the GitHub website interface. The user is logged in as 'bptfreitas'. The top navigation bar includes the GitHub logo, a search bar, and a button to 'Reiniciar para atualizar'. The left sidebar shows the user's profile and a list of repositories. The main content area displays the 'Home' page with a search bar and a feed of repository updates. The right sidebar shows the user's profile dropdown menu, which is open. The 'Settings' option is highlighted with a red box.

GitHub Dashboard

Search: Type / to search

Home

Ask Copilot

Suggest next steps for an issue | Get code feedback

Feed

cyberbotics/webots released yesterday

Sponsor

Webots Nightly Build (21-5-2025)

This is a nightly build of Webots from the following branch(es):

- [master](#)

It might be unstable, for a stable version of Webots, please use the [latest official release](#).

Settings

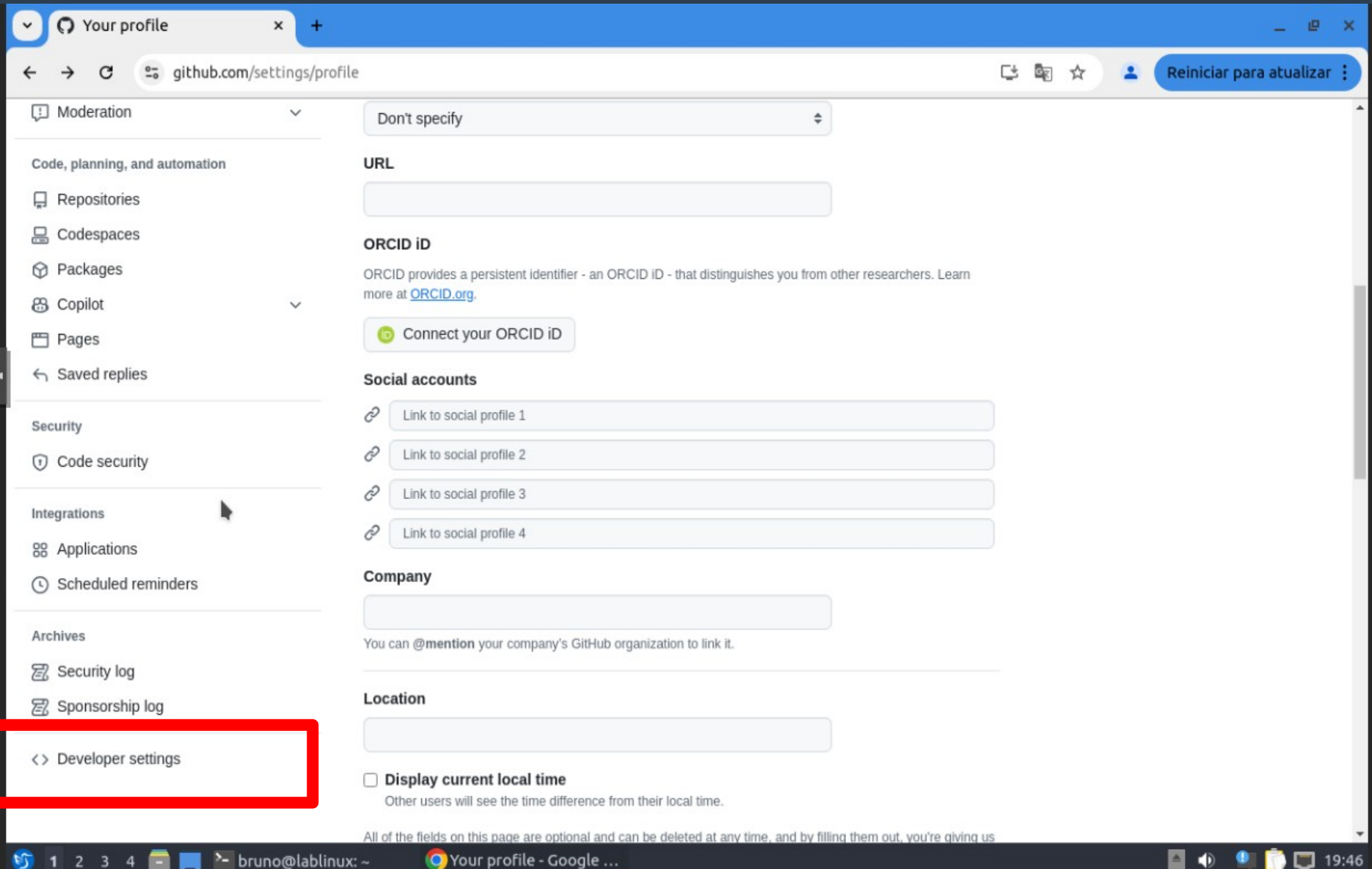
GitHub Website

GitHub Docs

GitHub Support

GitHub Community

3) Ir em “Developer Settings”



The screenshot shows the GitHub 'Your profile' settings page. The left sidebar contains a list of settings categories. The 'Developer settings' option, located at the bottom of the sidebar, is highlighted with a red rectangular box. The main content area on the right is currently displaying the 'Moderation' settings, which include a dropdown menu set to 'Don't specify', a 'URL' input field, an 'ORCID ID' section with a 'Connect your ORCID iD' button, a 'Social accounts' section with four 'Link to social profile' buttons, a 'Company' section with an input field, and a 'Location' section with an input field and a 'Display current local time' checkbox. The browser's address bar shows 'github.com/settings/profile'. The system tray at the bottom of the screen shows the terminal window 'bruno@lablinux: ~' and the time '19:46'.

Your profile x +

github.com/settings/profile

Reiniciar para atualizar

Moderation Don't specify

Code, planning, and automation

Repositories

Codespaces

Packages

Copilot

Pages

← Saved replies

Security

Code security

Integrations

Applications

Scheduled reminders

Archives

Security log

Sponsorship log

< > Developer settings

URL

ORCID ID

ORCID provides a persistent identifier - an ORCID ID - that distinguishes you from other researchers. Learn more at [ORCID.org](https://orcid.org).

Connect your ORCID iD

Social accounts

Link to social profile 1

Link to social profile 2

Link to social profile 3

Link to social profile 4

Company

You can @mention your company's GitHub organization to link it.

Location

☐ Display current local time

Other users will see the time difference from their local time.

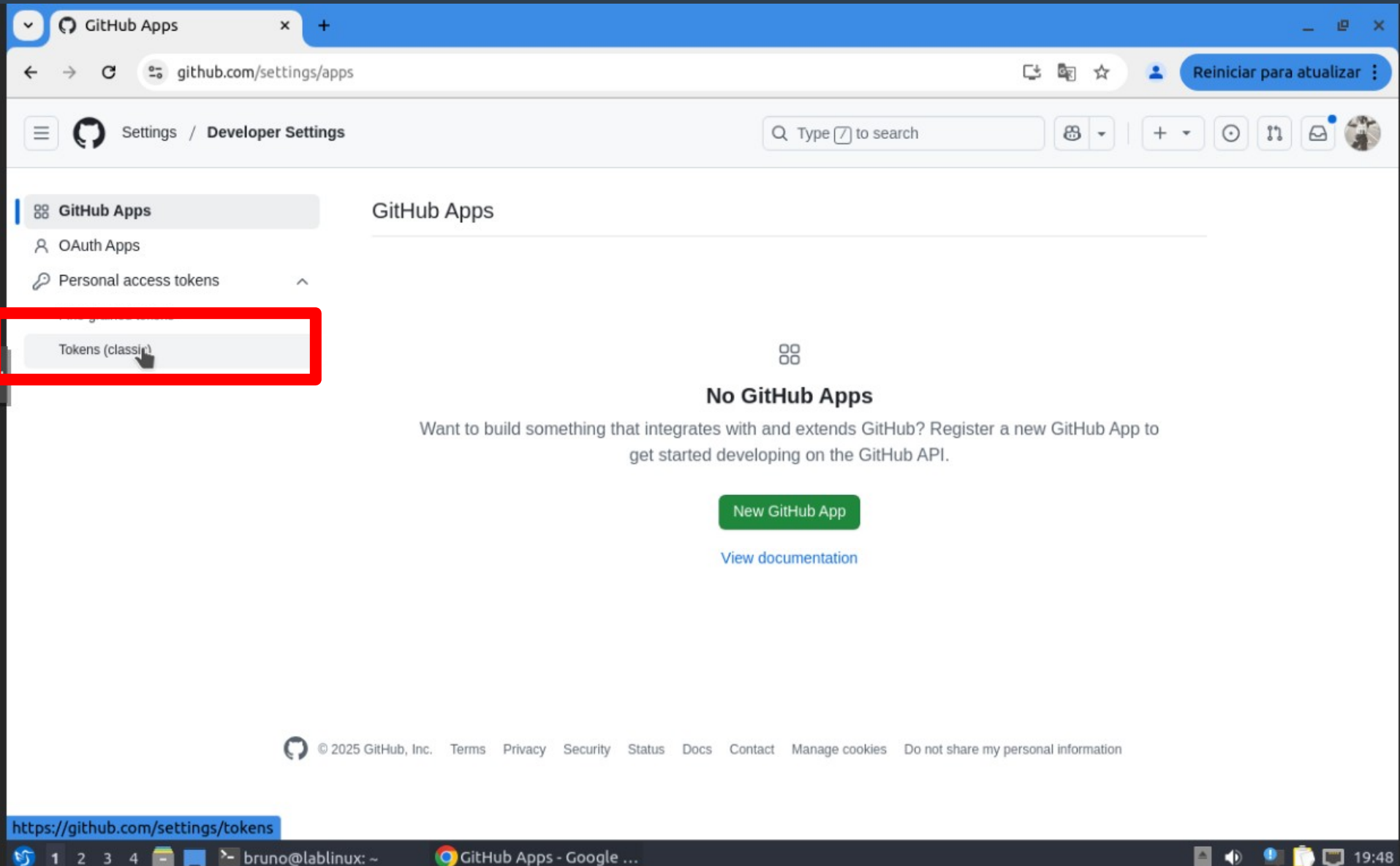
All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us

bruno@lablinux: ~

Your profile - Google ...

19:46

4) Personal access tokens → Tokens (classic)



5) Generate new token (classic)

The screenshot displays the GitHub 'Personal Access Tokens (classic)' settings page. The left sidebar shows the navigation menu with 'Personal access tokens' selected. The main content area lists existing tokens with their scopes and last used dates. A red box highlights the 'Generate new token (classic)' button, which is used for general use. The page also includes a 'Generate new token' button for fine-grained, repo-scoped tokens. The footer shows the GitHub copyright notice and various links.

Personal Access Tokens (classic)

Generate new token

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

Generate new token
Fine-grained, repo-scoped

Generate new token (classic)
For general use

GH auth — repo
⚠ This token has no expiration date.

Ubuntu on Linux — delete:packages, project, repo, workflow, write:packages
⚠ This token has no expiration date. Last used within the last 2 weeks [Delete](#)

Windows WSL — repo
⚠ This token has no expiration date. Last used within the last 6 months [Delete](#)

Ubuntu em casa — repo
⚠ This token has no expiration date. Last used within the last 6 months [Delete](#)

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](#).

© 2025 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)

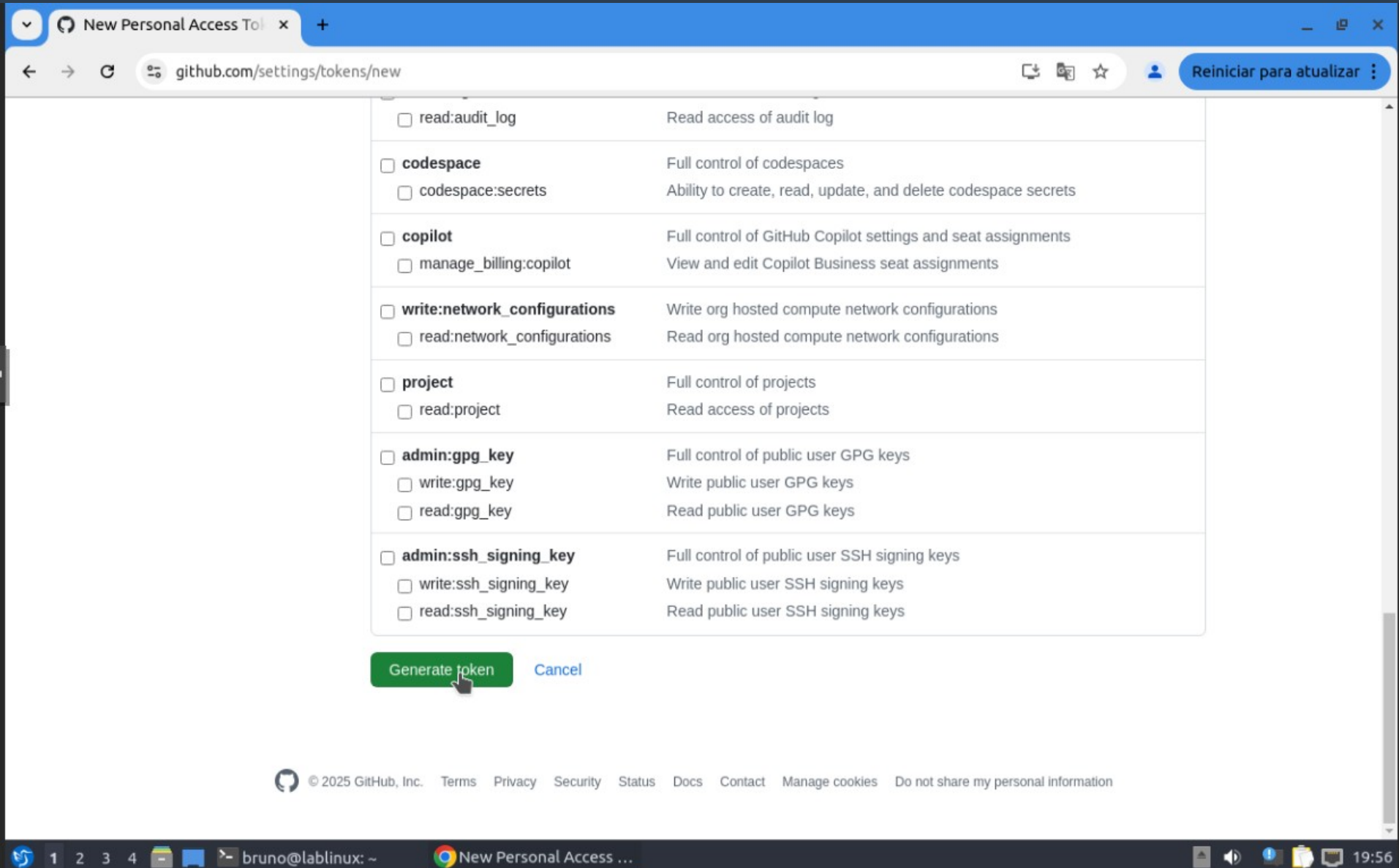
<https://github.com/settings/tokens/new>

6) Habilitar só o “repo”

The screenshot shows the GitHub 'New personal access token (classic)' page. The left sidebar shows the navigation menu with 'Personal access tokens' selected. The main content area is titled 'New personal access token (classic)' and includes a description of personal access tokens, a 'Note' section with a token example, an 'Expiration' dropdown set to 'No expiration', and a 'Select scopes' section. The 'Select scopes' section is highlighted with a red box, showing the following selected scopes:

- ☒ **repo** Full control of private repositories
 - ☒ repo:status Access commit status
 - ☒ repo_deployment Access deployment status
 - ☒ public_repo Access public repositories
 - ☒ repo:invite Access repository invitations
 - ☒ security_events Read and write security events
- ☐ **workflow** Update GitHub Action workflows

7) Generate token ...



... NÃO fechar esta janela !!!

Personal Access Tokens (classic)

Generate new token

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!

Token	Scope	Last used	Action
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			Delete
GH auth — repo		Never used	Delete
⚠ This token has no expiration date.			
Ubuntu on Linux — delete:packages, project, repo, workflow, write:packages		Last used within the last 2 weeks	Delete
⚠ This token has no expiration date.			
Windows WSL — repo		Last used within the last 6 months	Delete
⚠ This token has no expiration date.			
Ubuntu em casa — repo		Last used within the last 6 months	Delete
⚠ This token has no expiration date.			


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](#).

... NÃO fechar esta janela !!!

Personal Access Tokens (classic)

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!

Token Name	Scopes	Last Used	Action
ghp_3XUYvfUi1c9RNKP59E8I2IHIfQMXJY30h...			 Delete
GH auth — repo		Never used	Delete
Ubuntu on Linux — delete:packages, project, repo, workflow, write:packages		Last used within the last 2 weeks	Delete
Windows WSL — repo		Last used within the last 6 months	Delete
Ubuntu em casa — repo			

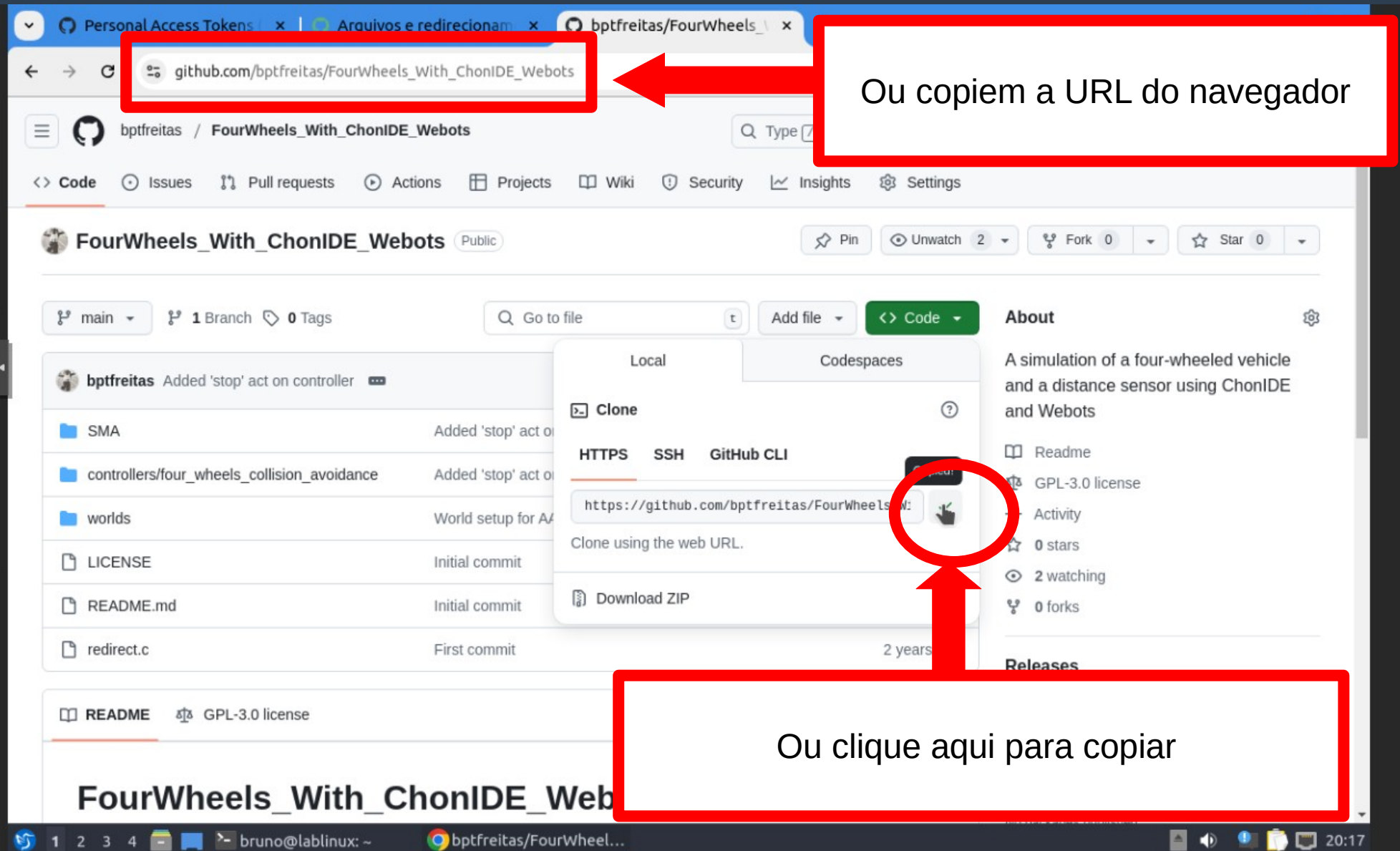
Personal access tokens (classic) can be used to [authenticate to the API over Basic Authentication](#).

Cliquem nesse botão para copiar a chave na "Área de transferência"

8) Abrir/criar um repositório público seu e clonem em sua máquina

The screenshot shows a web browser window displaying the GitHub repository page for `bptfreitas/FourWheels_With_ChonIDE_Webots`. The repository is public and has 0 stars, 2 watchers, and 0 forks. The 'Clone' dropdown menu is open, showing options for cloning via HTTPS, SSH, or GitHub CLI. The HTTPS URL is `https://github.com/bptfreitas/FourWheels_With_ChonIDE_Webots.git` and is marked as 'Copied!'. There is also a 'Download ZIP' option. The repository's file list includes `SMA`, `controllers/four_wheels_collision_avoidance`, `worlds`, `LICENSE`, `README.md`, and `redirect.c`. The repository description is 'A simulation of a four-wheeled vehicle and a distance sensor using ChonIDE and Webots'. The page also shows the 'About' section with a README link and the 'Releases' section with a 'Create a new release' link.

8) Abrir/criar um repositório público seu e clonem em sua máquina



The screenshot shows a web browser displaying the GitHub repository page for `bptfreitas/FourWheels_With_ChonIDE_Webots`. The browser's address bar is highlighted with a red box, and a red arrow points from a text box to it. The text box contains the instruction "Ou copiem a URL do navegador". Below the repository name, the "Code" button is highlighted with a red box, and a red arrow points from another text box to it. The text box contains the instruction "Ou clique aqui para copiar". The repository page shows the file structure, commit history, and repository statistics.

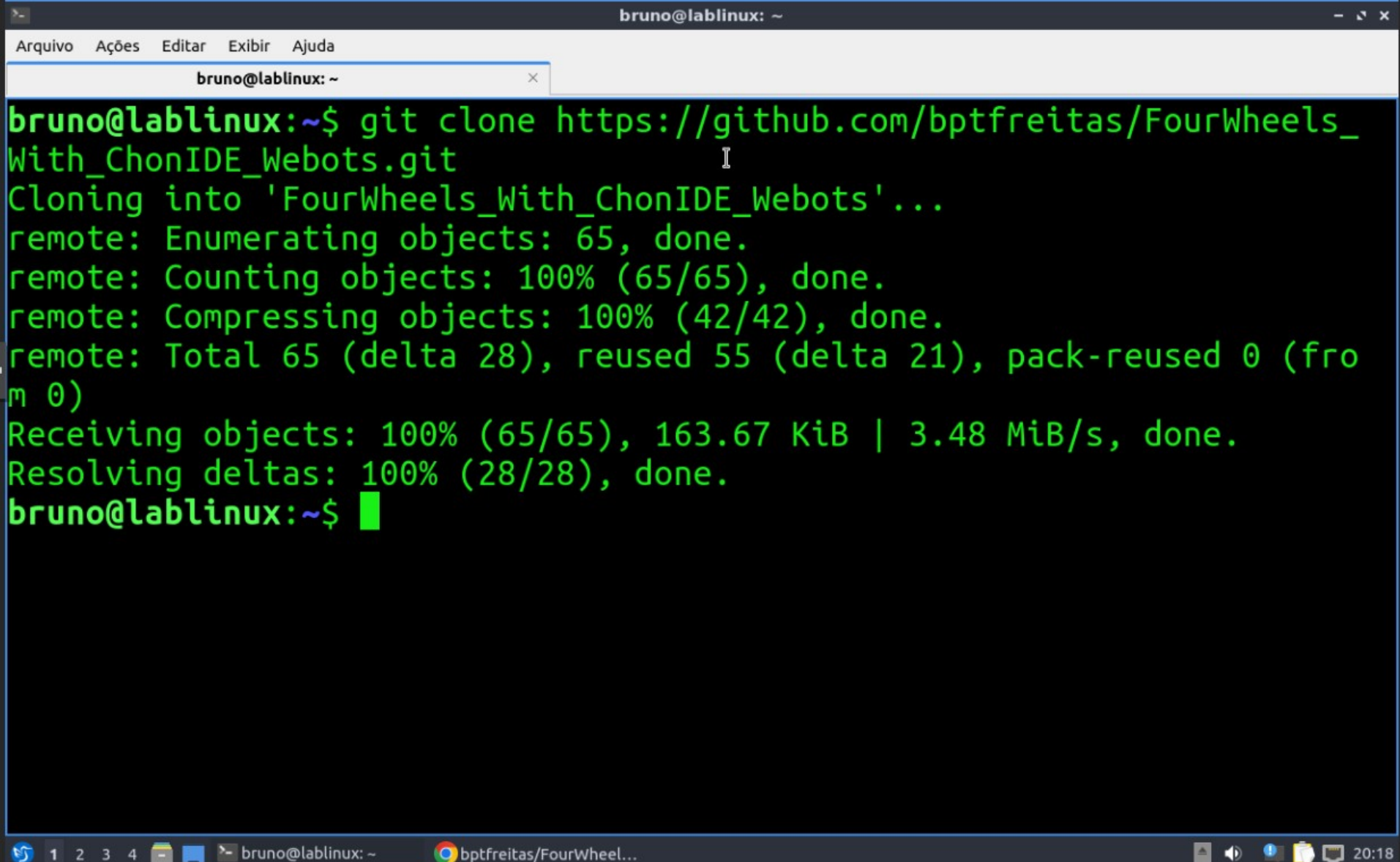
github.com/bptfreitas/FourWheels_With_ChonIDE_Webots

Ou copiem a URL do navegador

Code

Ou clique aqui para copiar

8) Abrir/criar um repositório público seu e clonem em sua máquina

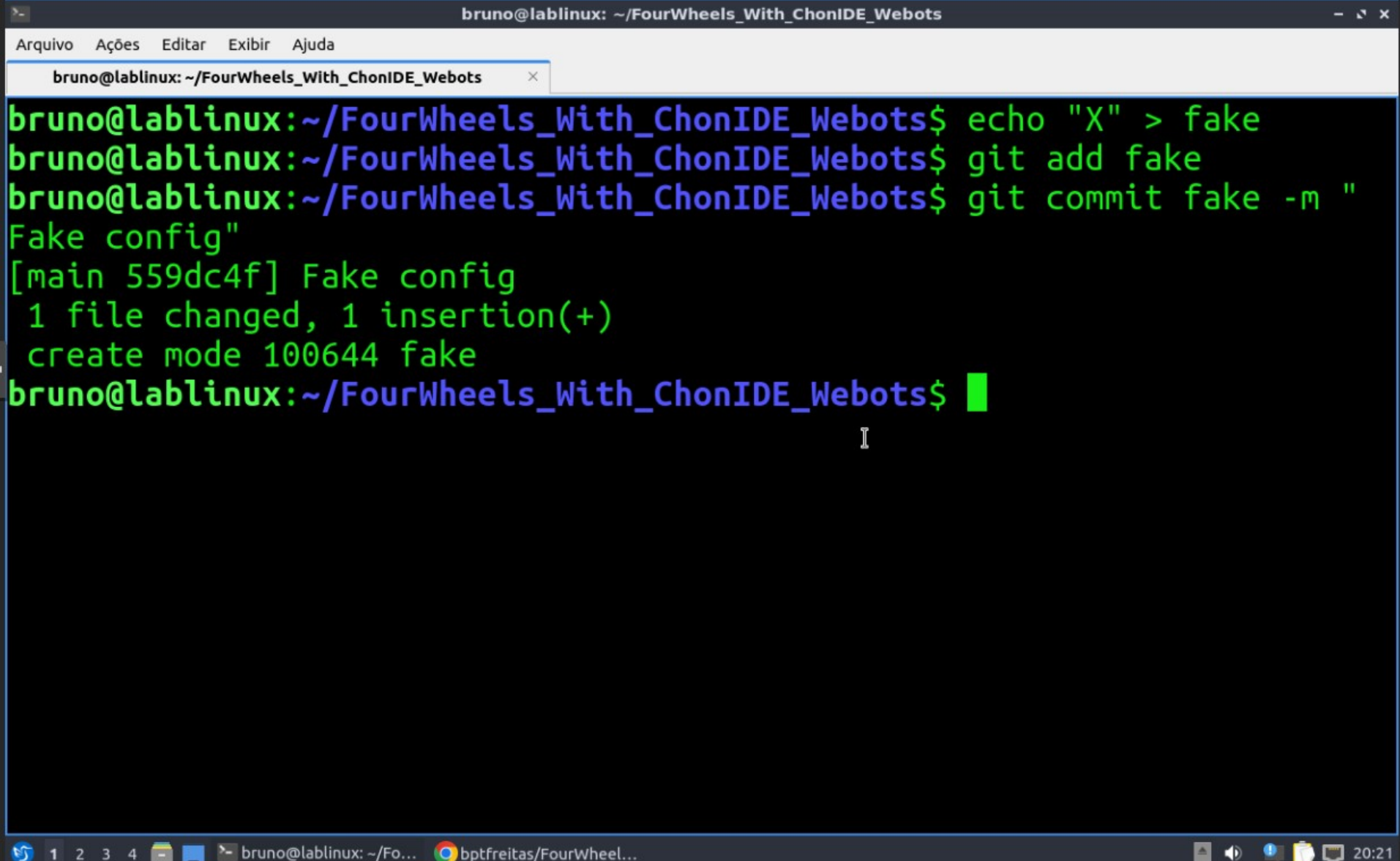


The screenshot shows a terminal window titled 'bruno@lablinux: ~'. The terminal output is as follows:

```
bruno@lablinux:~$ git clone https://github.com/bptfreitas/FourWheels_With_ChonIDE_Webots.git
Cloning into 'FourWheels_With_ChonIDE_Webots'...
remote: Enumerating objects: 65, done.
remote: Counting objects: 100% (65/65), done.
remote: Compressing objects: 100% (42/42), done.
remote: Total 65 (delta 28), reused 55 (delta 21), pack-reused 0 (from 0)
Receiving objects: 100% (65/65), 163.67 KiB | 3.48 MiB/s, done.
Resolving deltas: 100% (28/28), done.
bruno@lablinux:~$
```

The terminal window has a menu bar with 'Arquivo', 'Ações', 'Editar', 'Exibir', and 'Ajuda'. The left sidebar shows a VNC icon and other system icons. The bottom status bar shows the current directory 'bruno@lablinux: ~' and the repository name 'bptfreitas/FourWheel...'.

9) Façam uma alteração (mínima) no projeto e dê um commit

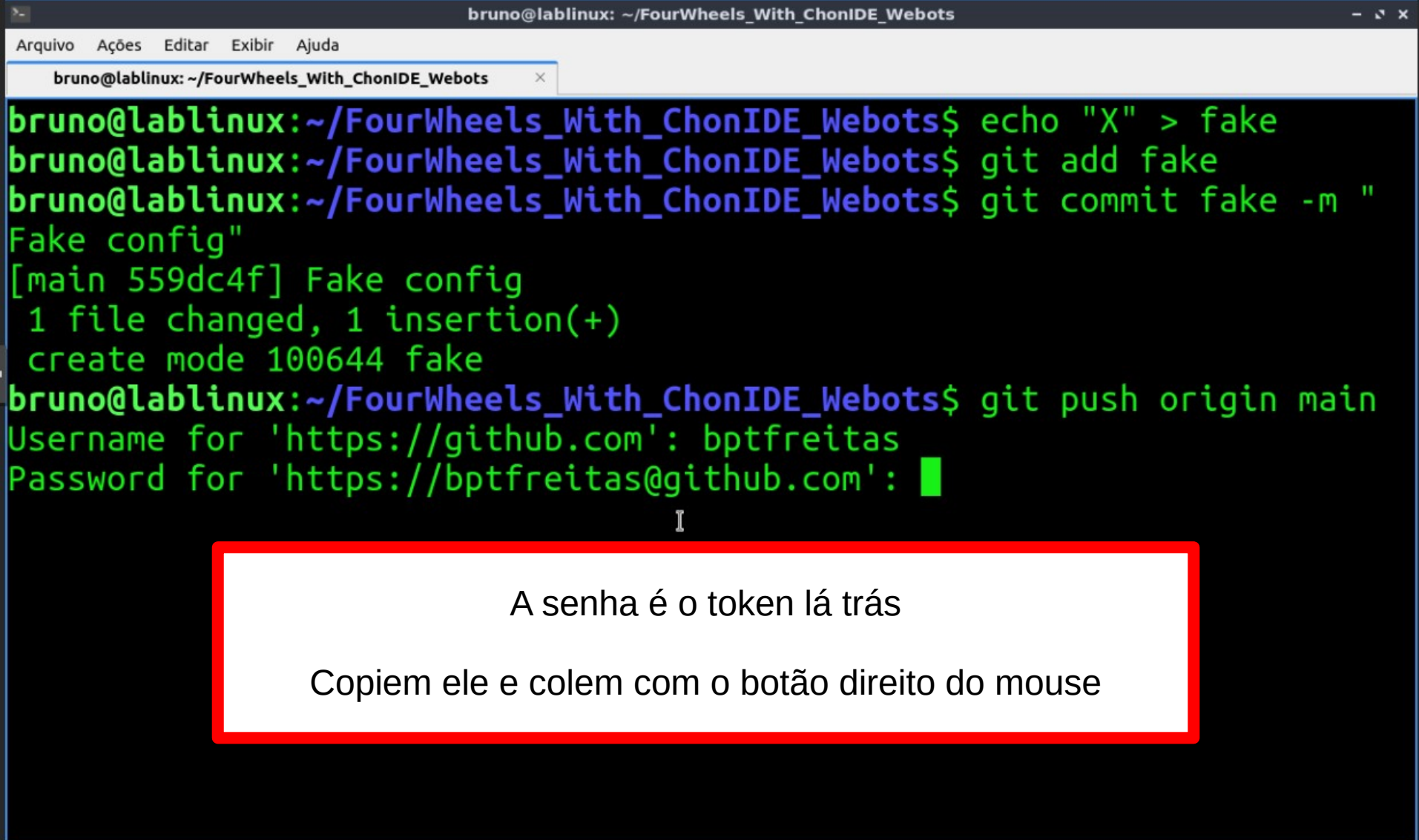


The screenshot shows a terminal window titled "bruno@lablinux: ~/FourWheels_With_ChonIDE_Webots". The terminal displays the following commands and output:

```
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ echo "X" > fake
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ git add fake
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ git commit fake -m "
Fake config"
[main 559dc4f] Fake config
1 file changed, 1 insertion(+)
create mode 100644 fake
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$
```

The terminal window has a menu bar with "Arquivo", "Ações", "Editar", "Exibir", and "Ajuda". On the left side, there is a VNC toolbar with icons for a keyboard, mouse, settings, and power. The bottom of the screen shows a taskbar with several open applications and a system tray with icons for network, volume, and time (20:21).

10) Faça um *git push origin main*



The screenshot shows a terminal window titled "bruno@lablinux: ~/FourWheels_With_ChonIDE_Webots". The terminal output is as follows:

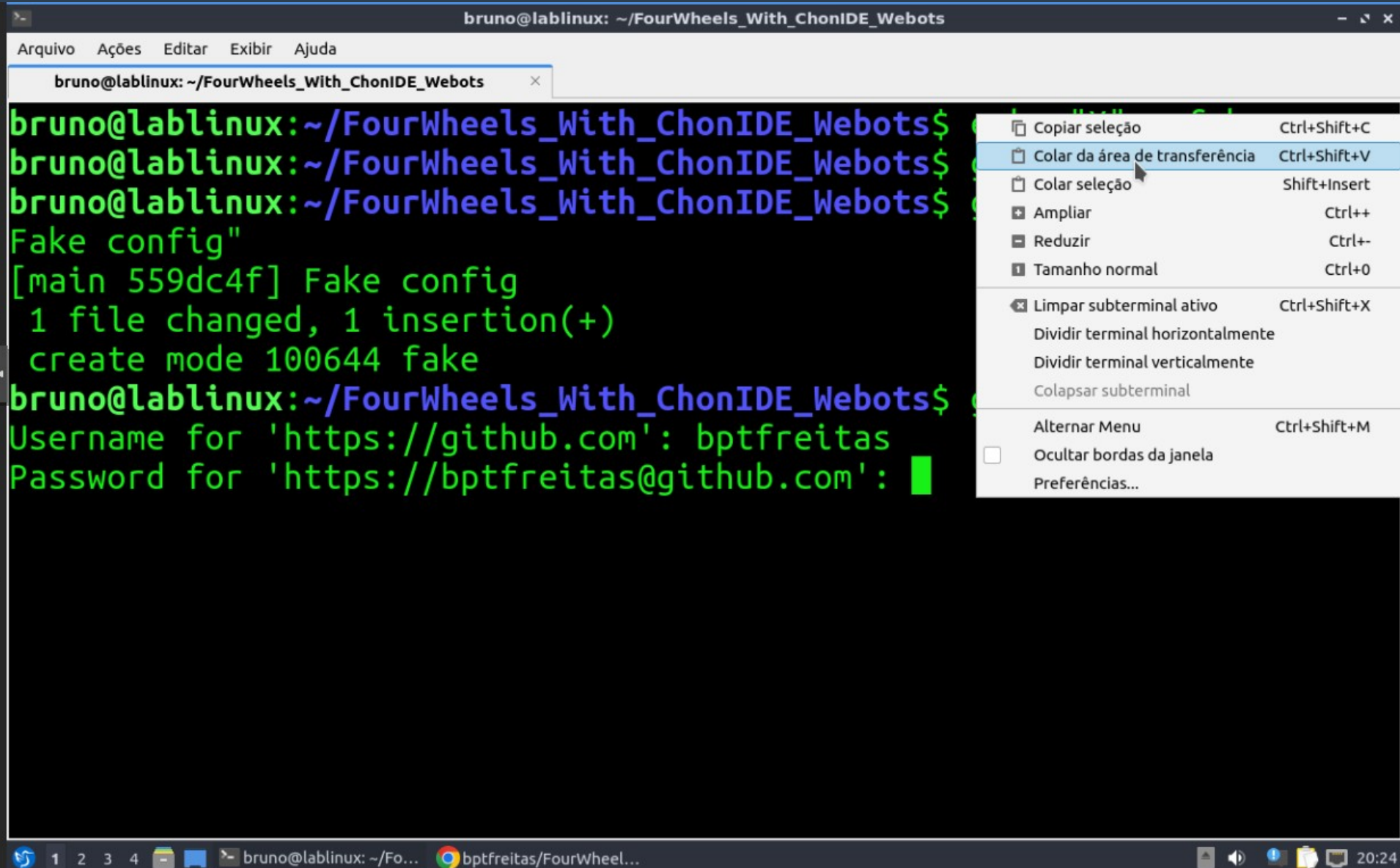
```
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ echo "X" > fake
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ git add fake
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ git commit fake -m "
Fake config"
[main 559dc4f] Fake config
1 file changed, 1 insertion(+)
create mode 100644 fake
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$ git push origin main
Username for 'https://github.com': bptfreitas
Password for 'https://bptfreitas@github.com':
```

A red rectangular box highlights the password prompt area, containing the text:

A senha é o token lá trás

Copiem ele e colem com o botão direito do mouse

10) Faça um *git push origin main*



The screenshot shows a terminal window titled "bruno@lablinux: ~/FourWheels_With_ChonIDE_Webots". The terminal output is as follows:

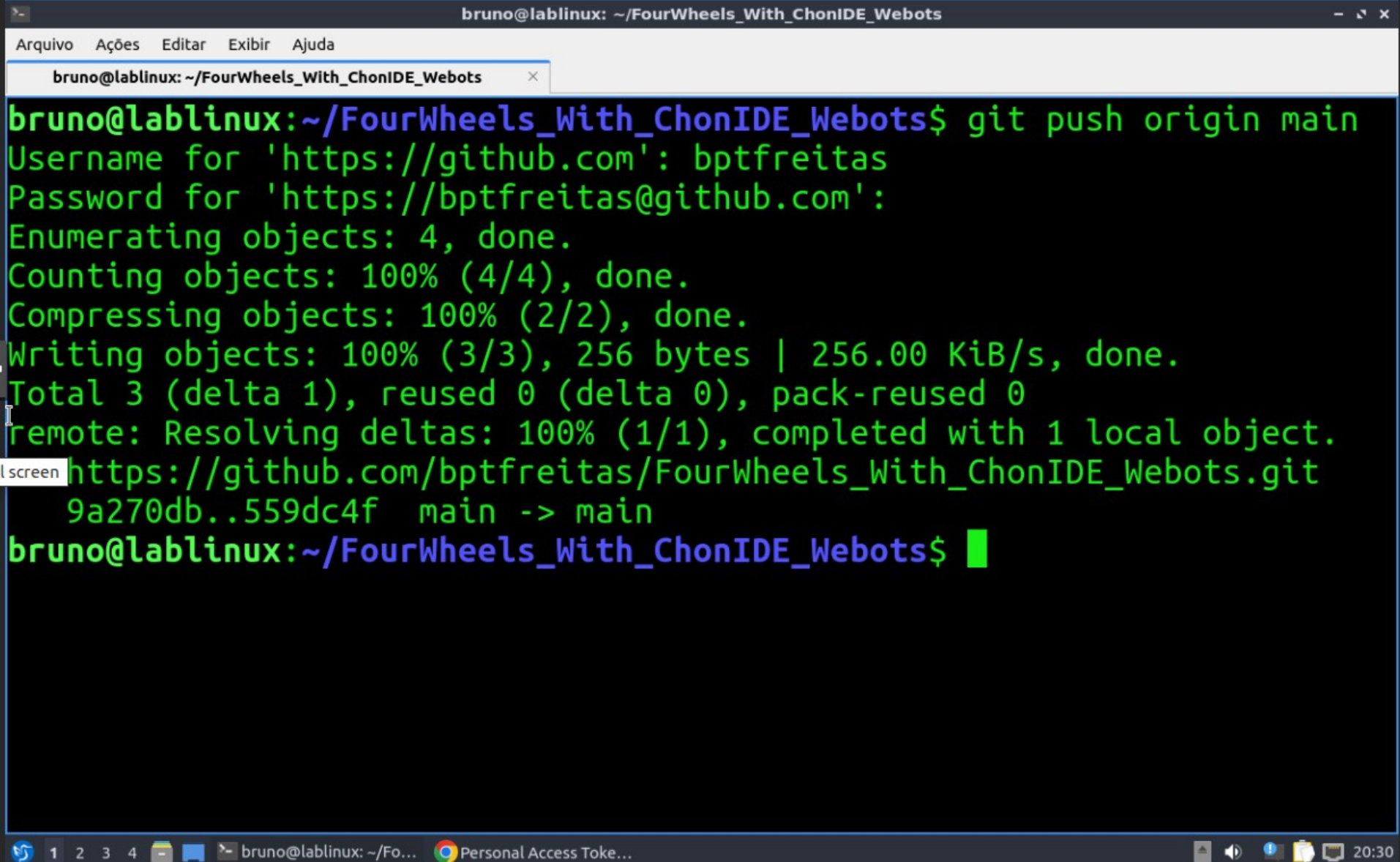
```
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$  
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$  
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$  
Fake config"  
[main 559dc4f] Fake config  
1 file changed, 1 insertion(+)  
create mode 100644 fake  
bruno@lablinux:~/FourWheels_With_ChonIDE_Webots$  
Username for 'https://github.com': bptfreitas  
Password for 'https://bptfreitas@github.com':
```

A context menu is open on the right side of the terminal, listing various actions and their keyboard shortcuts:

- Copiar seleção (Ctrl+Shift+C)
- Colar da área de transferência (Ctrl+Shift+V)
- Colar seleção (Shift+Insert)
- Ampliar (Ctrl++)
- Reduzir (Ctrl+-)
- Tamanho normal (Ctrl+0)
- Limpar subterminal ativo (Ctrl+Shift+X)
- Dividir terminal horizontalmente
- Dividir terminal verticalmente
- Colapsar subterminal
- Alternar Menu (Ctrl+Shift+M)
- Ocultar bordas da janela
- Preferências...

The terminal window has a menu bar with "Arquivo", "Ações", "Editar", "Exibir", and "Ajuda". The bottom status bar shows the current directory and the user's name.

Se tudo deu certo ...



The screenshot shows a terminal window titled "bruno@lablinux: ~/FourWheels_With_ChonIDE_Webots". The terminal output shows the execution of the command "git push origin main". The output includes prompts for username and password, progress bars for enumerating, counting, and compressing objects, and a final confirmation of the push to the remote repository. The terminal window is part of a VNC session, as indicated by the "VNC" logo in the top left corner of the desktop environment. The desktop environment includes a taskbar at the bottom with icons for a web browser, a file manager, and a terminal window. The system clock in the bottom right corner shows the time as 20:30.

```
bruno@lablinux: ~/FourWheels_With_ChonIDE_Webots$ git push origin main
Username for 'https://github.com': bptfreitas
Password for 'https://bptfreitas@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 256 bytes | 256.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
https://github.com/bptfreitas/FourWheels_With_ChonIDE_Webots.git
9a270db..559dc4f  main -> main
bruno@lablinux: ~/FourWheels_With_ChonIDE_Webots$
```

- **Pronto! Agora toda vez que vocês forem fazer um push em repositório com autenticação, ele não irá perguntar senha!**

- **Nota: No mundo real, o ideal é criar um token por projeto**
 - O token que criamos aqui vale para todos os seus repositórios
 - Se alguém descobrir o token ... ferrou!
- **E tem um agravante: o token é salvo em texto claro!**
 - ~/.git-credentials