UNIVERSIDADE FEDERAL DE MATO GROSSO
***CAMPUS* UNIVERSITÁRIO DE RONDONÓPOLIS – CUR**
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS – ICEN
SISTEMAS DA INFORMAÇÃO

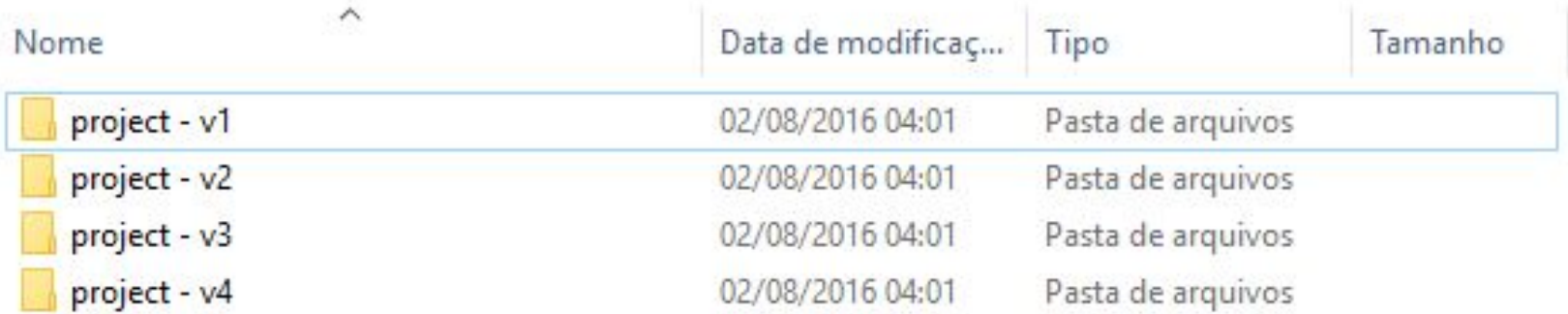
ALEXANDRE THEBALDI

CONTROLE DE VERSÃO

RONDONÓPOLIS
2016

Como muitas pessoas controlam suas versões?

Cópias dos arquivos em outros diretórios.

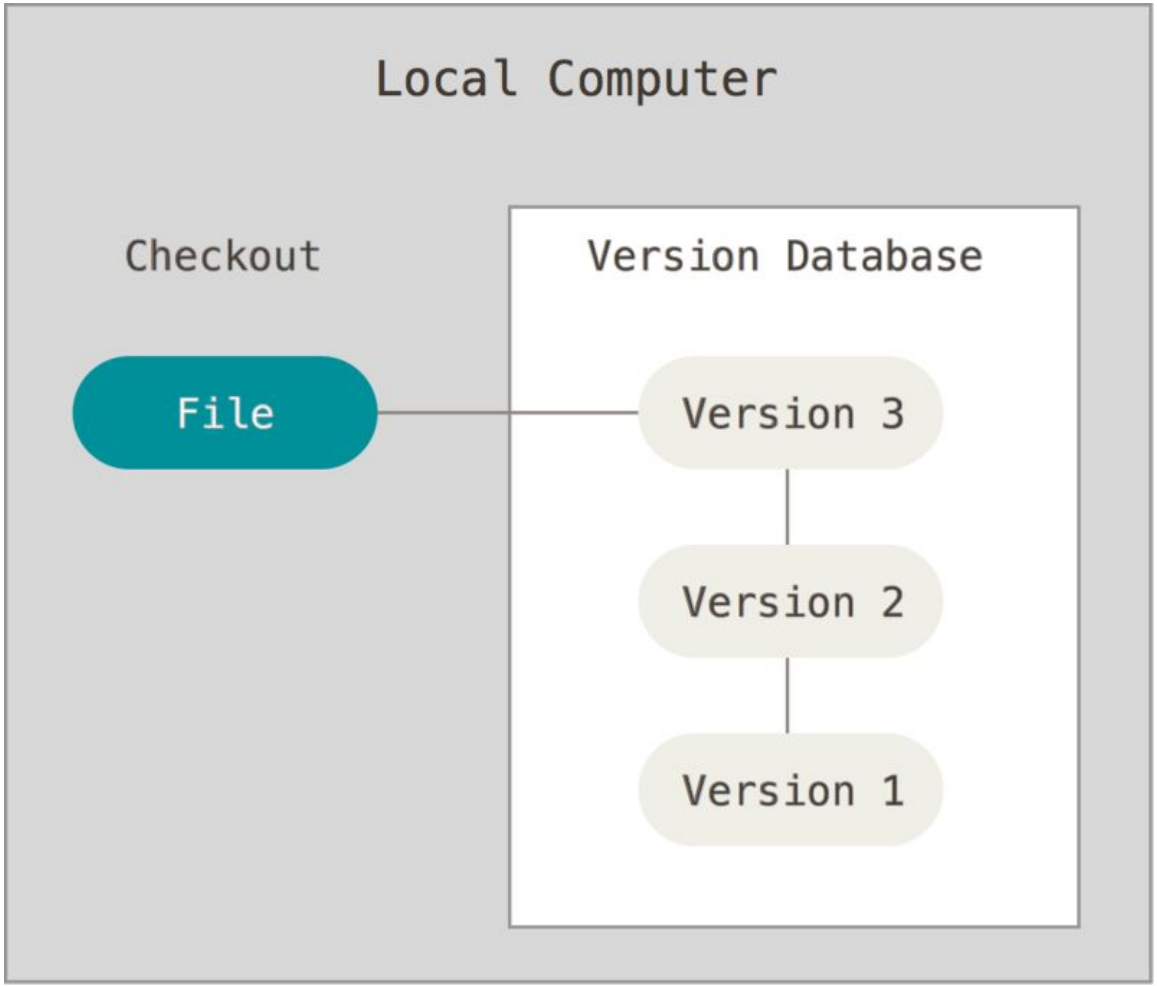


Sistemas para Controle de Versão

“*Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.*”

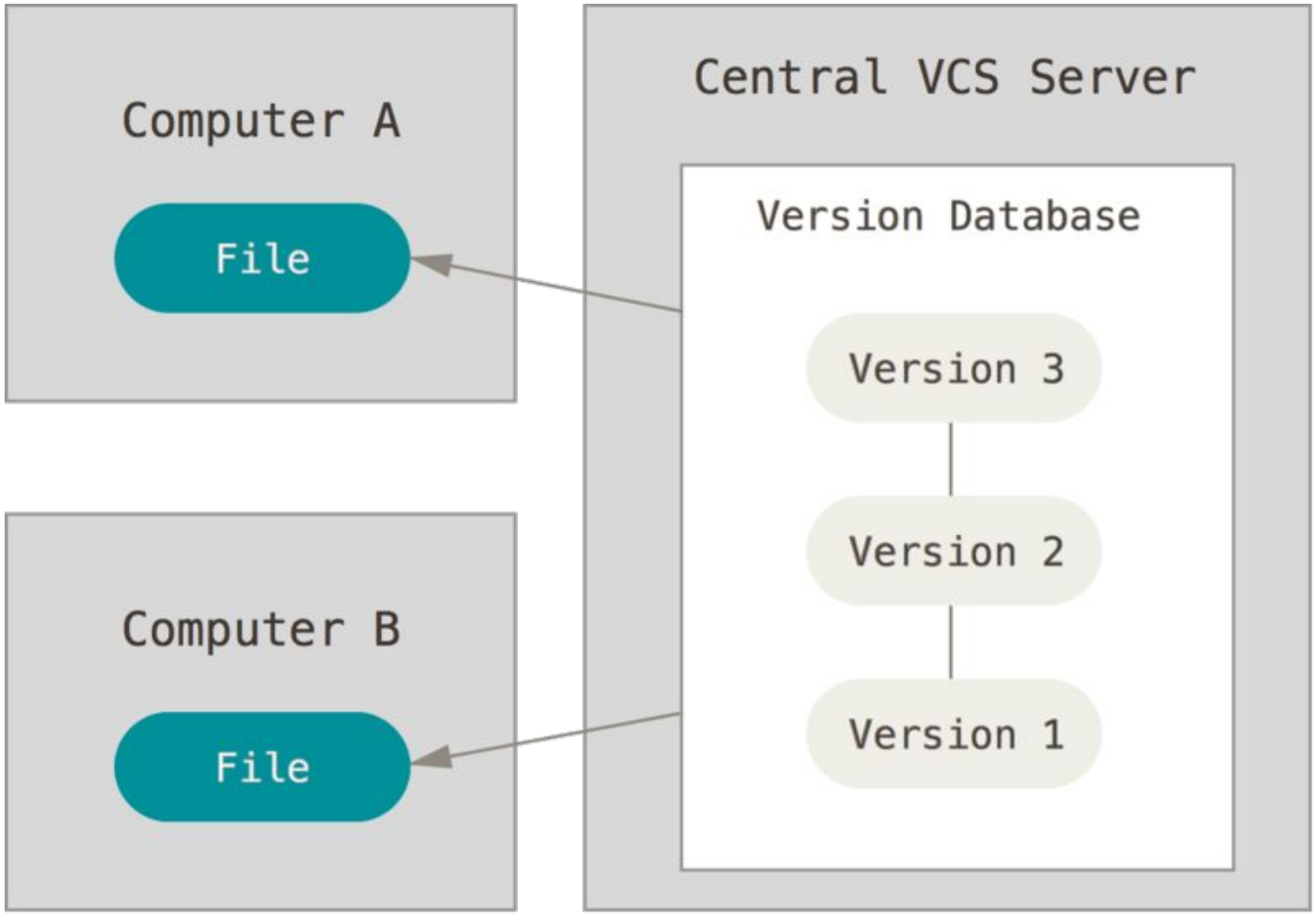
Git, About Version Control.

Sistemas de Controle de Versão Local


Local Computer
Checkout
File
Version Database
Version 3
Version 2
Version 1

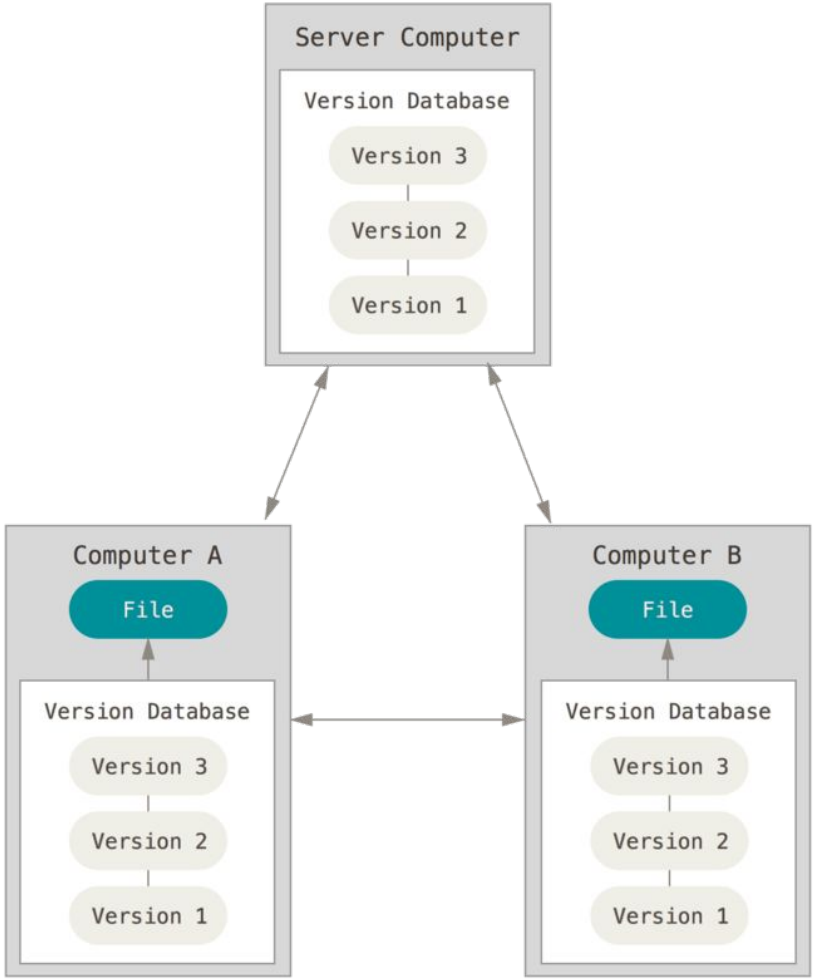
Ferramentas de CVS Local: *Revision Control System (RCS)*

Sistemas de Controle de Versão Centralizado


Computer A
File
Central VCS Server
Version Database
Version 3
Version 2
Version 1
Computer B
File

Ferramentas de CVS Centralizado*: Subversion (TortoiseSVN)* e *Perforce*.

Sistemas de Controle de Versão Distribuído


Server Computer
Version Database
Version 3
Version 2
Version 1
Computer A
File
Version Database
Version 3
Version 2
Version 1
Computer B
File
Version Database
Version 3
Version 2
Version 1

Ferramentas de CVS Distribuído: *Git*, *Mercurial*, *Bazaar* e *Darcs*.

Introdução ao Git

“Git is a version control system that is used for software development and other version control tasks. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows. Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development.”

Wikipedia, Git (Software).

1. Iniciando um repositório *Git*

1.1. Iniciar um repositório em um diretório existente

```
$ git init
```

```
alexandre@alexandre MINGW64 /c
$ mkdir git-basics

alexandre@alexandre MINGW64 /c
$ cd git-basics

alexandre@alexandre MINGW64 /c/git-basics
$ git init
Initialized empty Git repository in C:/git-basics/.git/

alexandre@alexandre MINGW64 /c/git-basics (master)
$ ls -a
./  ../  .git/

alexandre@alexandre MINGW64 /c/git-basics (master)
$ ls -r .git
refs/  objects/  info/  hooks/  HEAD  description  config

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

1.2. Clonar um repositório existente

```
$ git clone [url]
```

ahlechandre / git-basics

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

No description or website provided. — Edit

1 commit 1 branch 0 releases 1 contributor

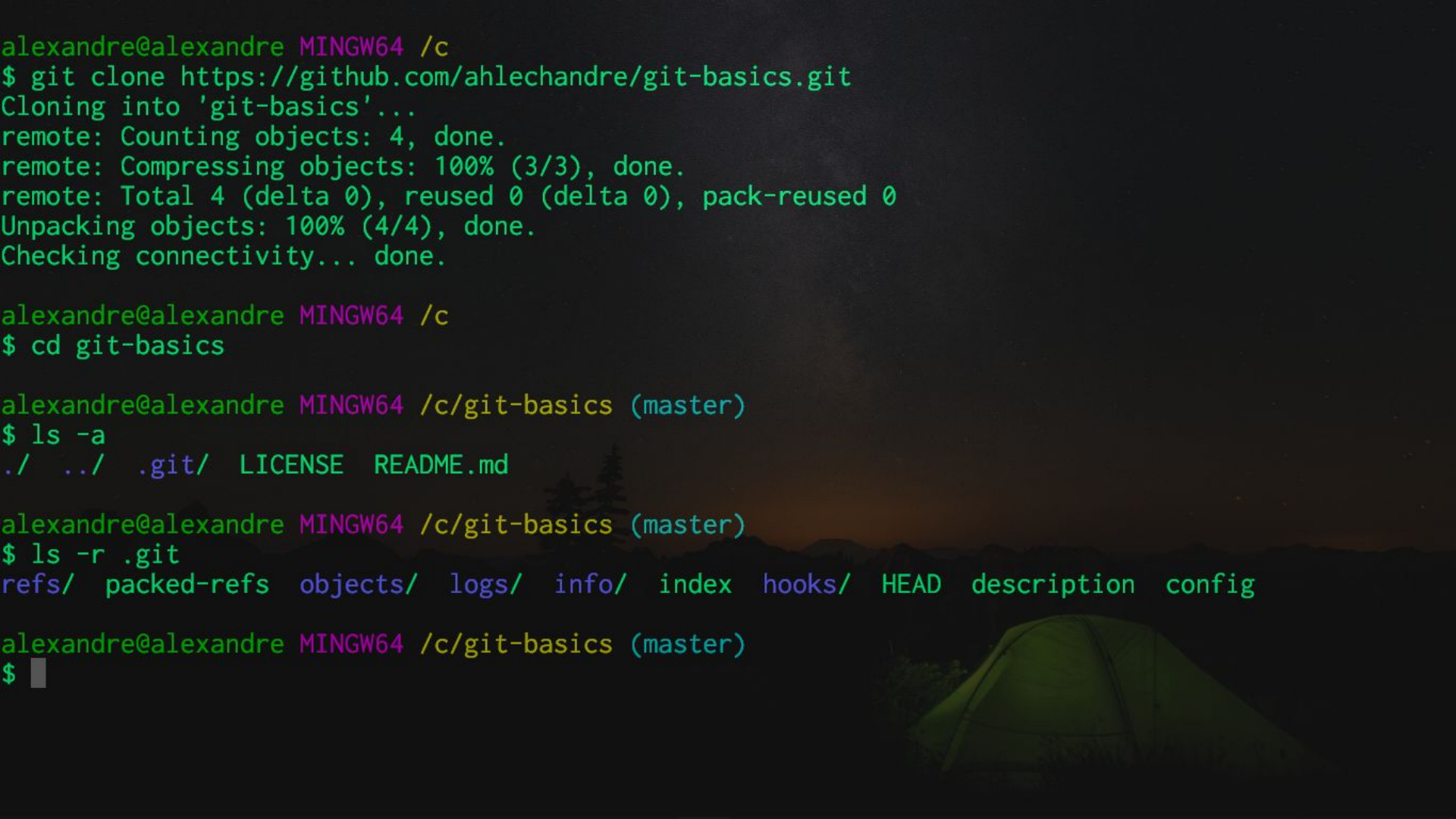
Branch: master New pull request Create new file Upload files Find file Clone or download

ahlechandre Initial commit Latest commit 5fcbdd3 14 minutes ago

LICENSE	Initial commit	14 minutes ago
README.md	Initial commit	14 minutes ago

README.md

git-basics


alexandre@alexandre MINGW64 /c
$ git clone https://github.com/ahlechandre/git-basics.git
Cloning into 'git-basics'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
Checking connectivity... done.
alexandre@alexandre MINGW64 /c
$ cd git-basics
alexandre@alexandre MINGW64 /c/git-basics (master)
$ ls -a
./ ../ .git/ LICENSE README.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$ ls -r .git
refs/ packed-refs objects/ logs/ info/ index hooks/ HEAD description config
alexandre@alexandre MINGW64 /c/git-basics (master)
$

2. Gravando modificações no repositório

Untracked vs. Tracked

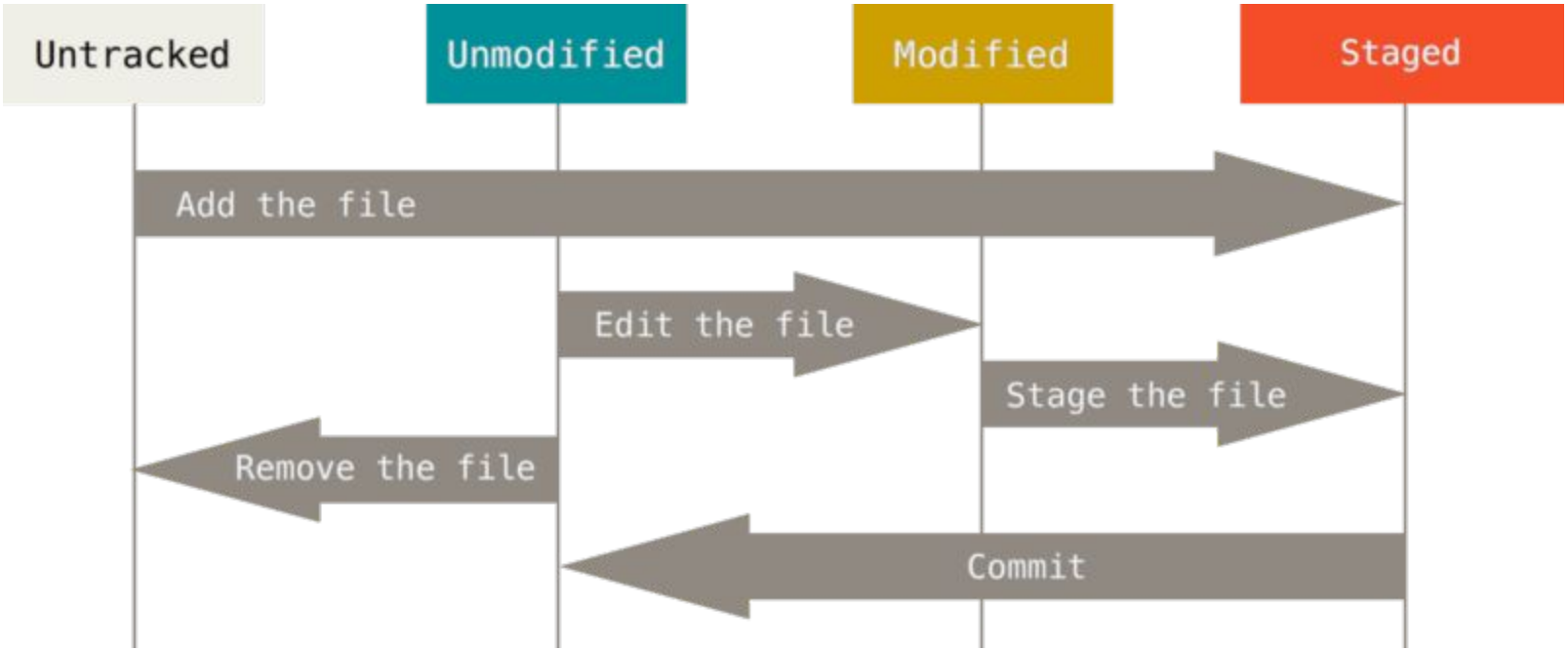
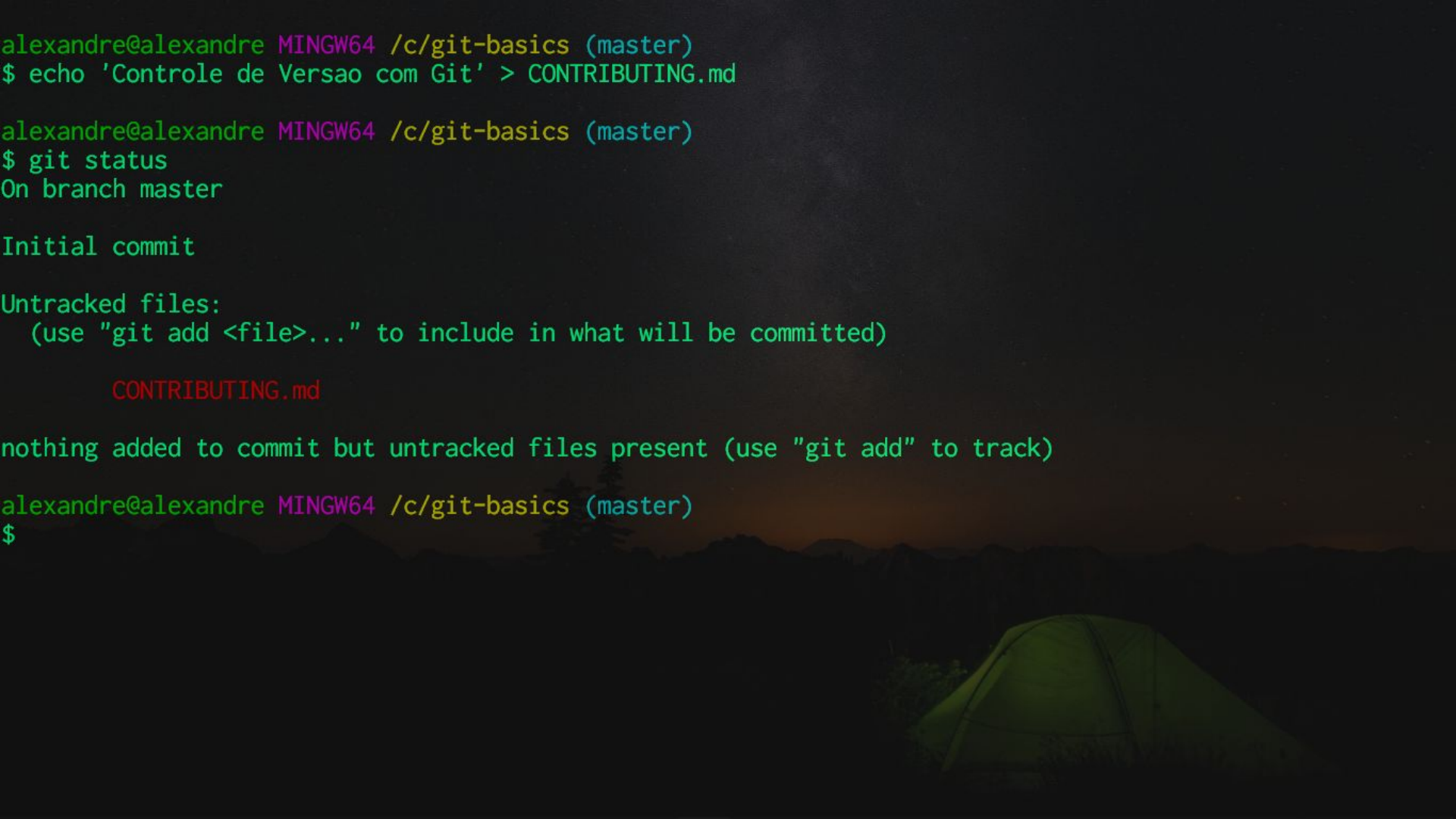


Figura 1. *The lifecycle of the status of your files*.

2.1. Verificar o *status* dos arquivos

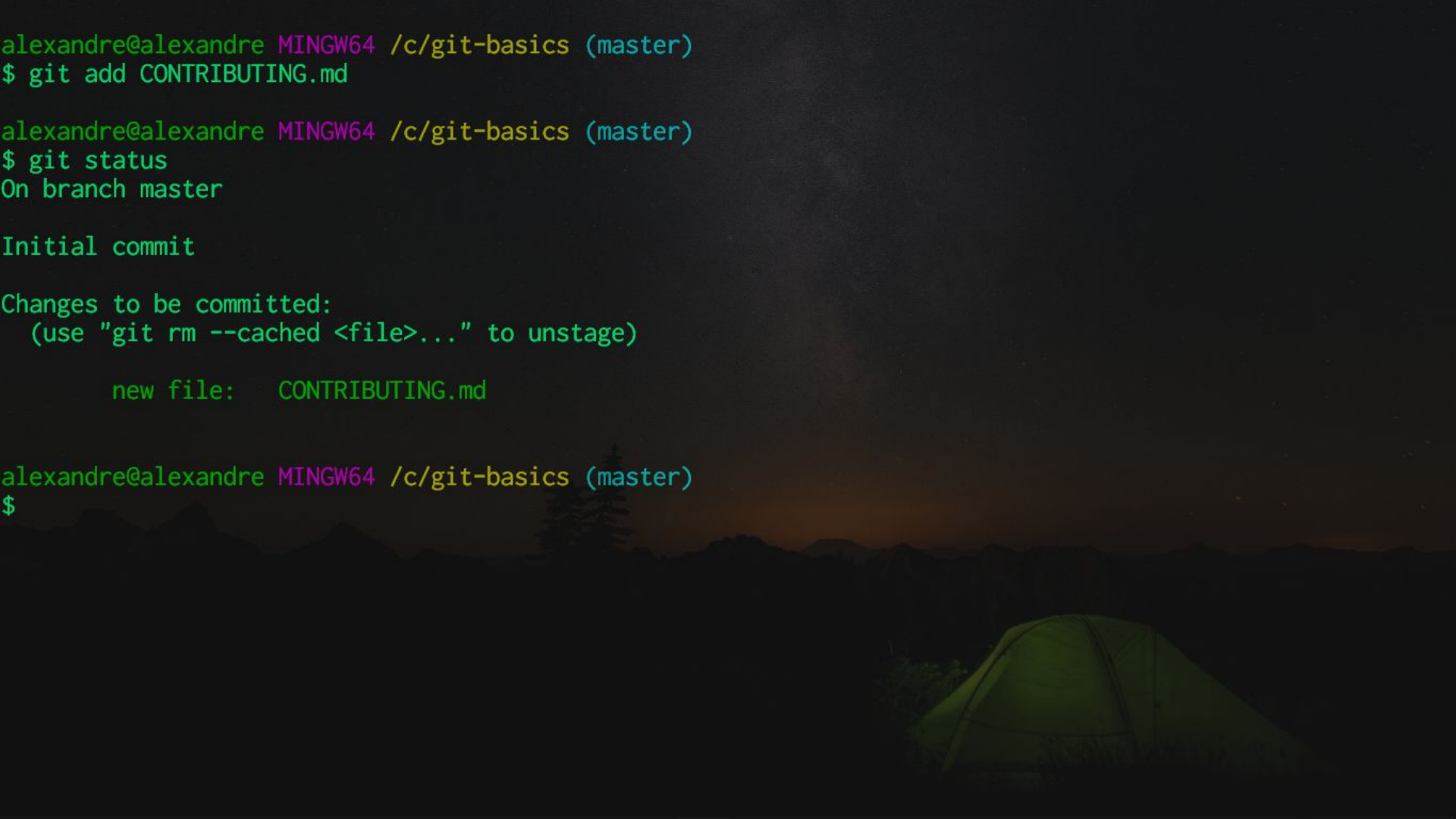
```
$ git status
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git' > CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        CONTRIBUTING.md

nothing added to commit but untracked files present (use "git add" to track)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```



2.2. *Tracking* (rastreamento) novos arquivos

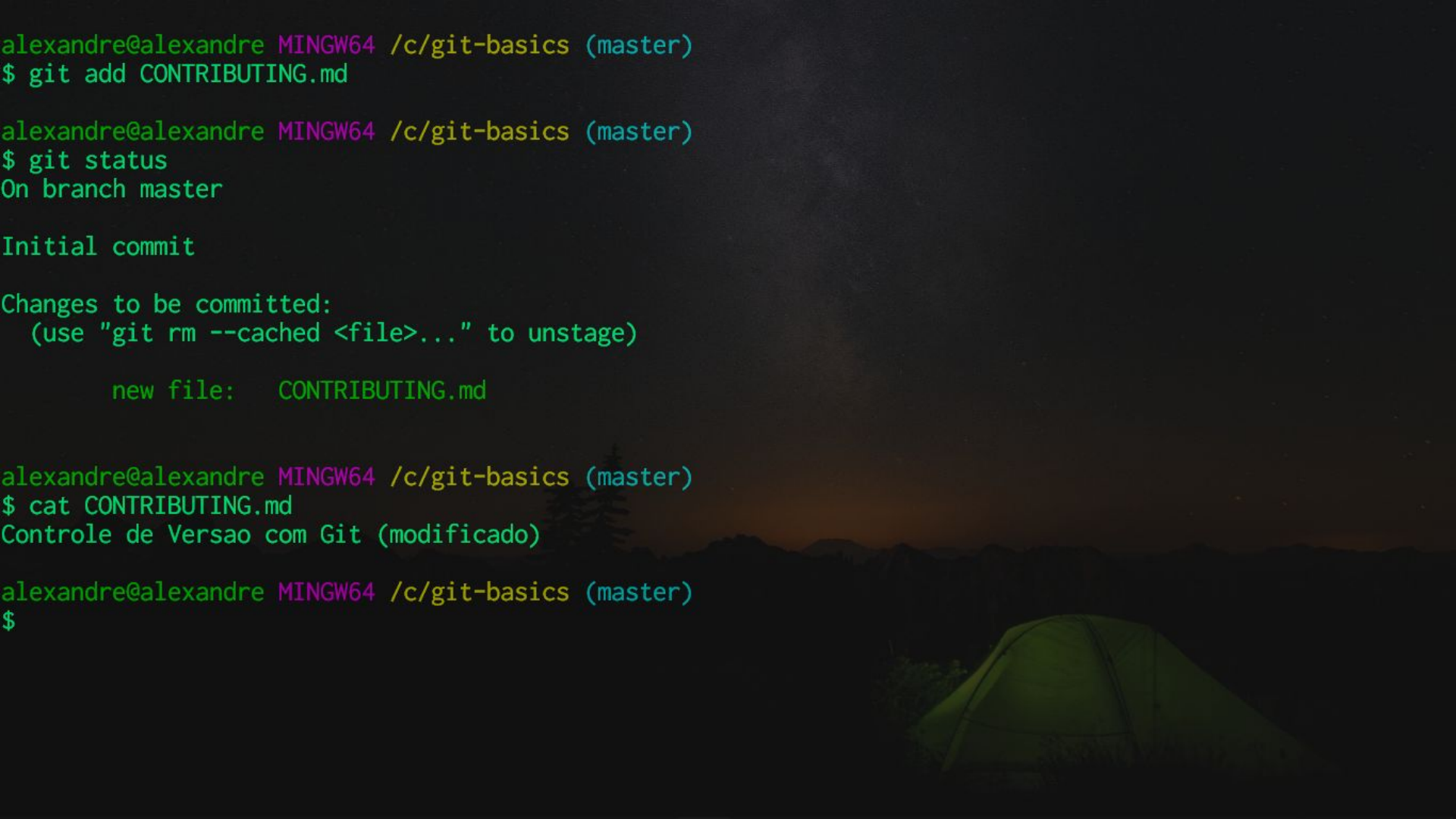
```
$ git add <file>
```


alexandre@alexandre MINGW64 /c/git-basics (master)
$ git add CONTRIBUTING.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master
Initial commit
Changes to be committed:
(use "git rm --cached <file>..." to unstage)
new file: CONTRIBUTING.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$

2.3. Modificar *staged* files

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git (modificado)' > CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   CONTRIBUTING.md

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

        modified:   CONTRIBUTING.md


alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git add CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   CONTRIBUTING.md


alexandre@alexandre MINGW64 /c/git-basics (master)
$ cat CONTRIBUTING.md
Controle de Versao com Git (modificado)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```



2.4. Mostrar modificações

```
$ git diff <file>
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git (nova modificacao)' > CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   CONTRIBUTING.md

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

        modified:   CONTRIBUTING.md


alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git diff CONTRIBUTING.md
diff --git a/CONTRIBUTING.md b/CONTRIBUTING.md
index 2c5f507..e112304 100644
--- a/CONTRIBUTING.md
+++ b/CONTRIBUTING.md
@@ -1 +1 @@
-Controle de Versao com Git (modificado)
+Controle de Versao com Git (nova modificacao)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

2.5. Fazer *commit* das modificações

```
$ git commit
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git add CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git commit -m 'versao 0.1'
[master (root-commit) 052f0e0] versao 0.1
 1 file changed, 1 insertion(+)
 create mode 100644 CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master
nothing to commit, working directory clean

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

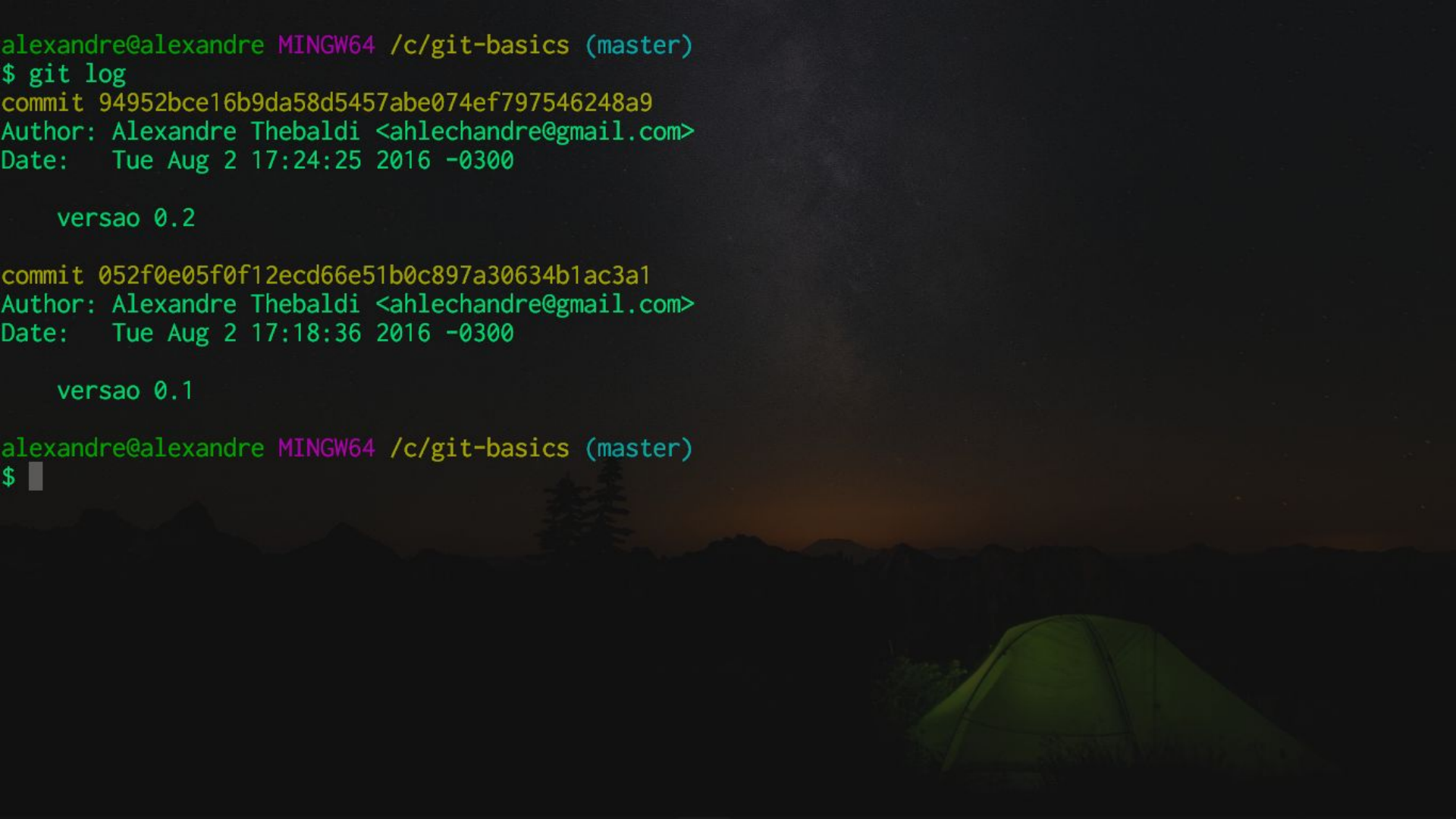
2.6. “Pulando” a *Staging area*

```
$ git commit -a
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git (versao 0.2)' > CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   CONTRIBUTING.md

no changes added to commit (use "git add" and/or "git commit -a")

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git commit -a -m 'versao 0.2'
[master 94952bc] versao 0.2
 1 file changed, 1 insertion(+), 1 deletion(-)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

3. Visualizando o histórico de *commits*

```
$ git log
```


alexandre@alexandre MINGW64 /c/git-basics (master)
$ git log
commit 94952bce16b9da58d5457abe074ef797546248a9
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date: Tue Aug 2 17:24:25 2016 -0300
versao 0.2
commit 052f0e05f0f12ecd66e51b0c897a30634b1ac3a1
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date: Tue Aug 2 17:18:36 2016 -0300
versao 0.1
alexandre@alexandre MINGW64 /c/git-basics (master)
$

4. Desfazendo coisas

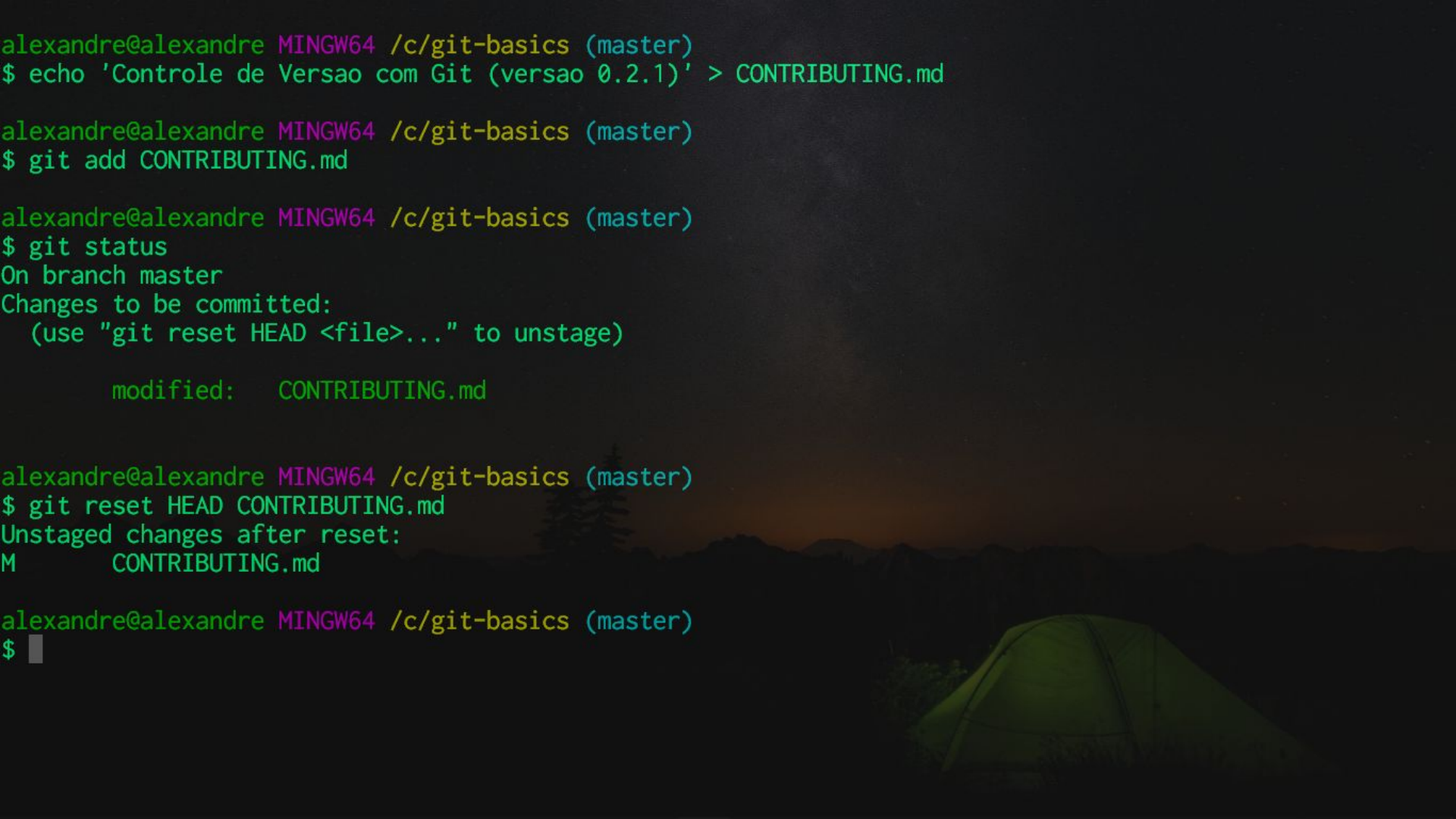
4.1. Desfazer modificação em arquivo modificado

```
$ git checkout -- <file>
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git (versao 0.2.1)' > CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   CONTRIBUTING.md

no changes added to commit (use "git add" and/or "git commit -a")

alexandre@alexandre MINGW64 /c/git-basics (master)
$ cat CONTRIBUTING.md
Controle de Versao com Git (versao 0.2.1)

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git checkout -- CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ cat CONTRIBUTING.md
Controle de Versao com Git (versao 0.2)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

4.2. Remover arquivos da *Staging area*

```
$ git reset HEAD <file>
```


alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git (versao 0.2.1)' > CONTRIBUTING.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git add CONTRIBUTING.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git status
On branch master
Changes to be committed:
(use "git reset HEAD <file>..." to unstage)
modified: CONTRIBUTING.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git reset HEAD CONTRIBUTING.md
Unstaged changes after reset:
M CONTRIBUTING.md
alexandre@alexandre MINGW64 /c/git-basics (master)
$

4.3. Desfazer o último *commit*

```
$ git reset [--hard] HEAD~1
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ echo 'Controle de Versao com Git (versao 0.3)' > CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git commit -a -m 'versao 0.3'
[master 4b3b177] versao 0.3
 1 file changed, 1 insertion(+), 1 deletion(-)

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git log
commit 4b3b177f8da7bdb2aacfef6f429df6f3f2226b32
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date:   Tue Aug 2 18:15:57 2016 -0300

    versao 0.3

commit 94952bce16b9da58d5457abe074ef797546248a9
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date:   Tue Aug 2 17:24:25 2016 -0300

    versao 0.2

commit 052f0e05f0f12ecd66e51b0c897a30634b1ac3a1
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date:   Tue Aug 2 17:18:36 2016 -0300

    versao 0.1
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git reset --hard HEAD~1
HEAD is now at 94952bc versao 0.2

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git log
commit 94952bce16b9da58d5457abe074ef797546248a9
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date:   Tue Aug 2 17:24:25 2016 -0300

    versao 0.2

commit 052f0e05f0f12ecd66e51b0c897a30634b1ac3a1
Author: Alexandre Thebaldi <ahlechandre@gmail.com>
Date:   Tue Aug 2 17:18:36 2016 -0300

    versao 0.1

alexandre@alexandre MINGW64 /c/git-basics (master)
$ cat CONTRIBUTING.md
Controle de Versao com Git (versao 0.2)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

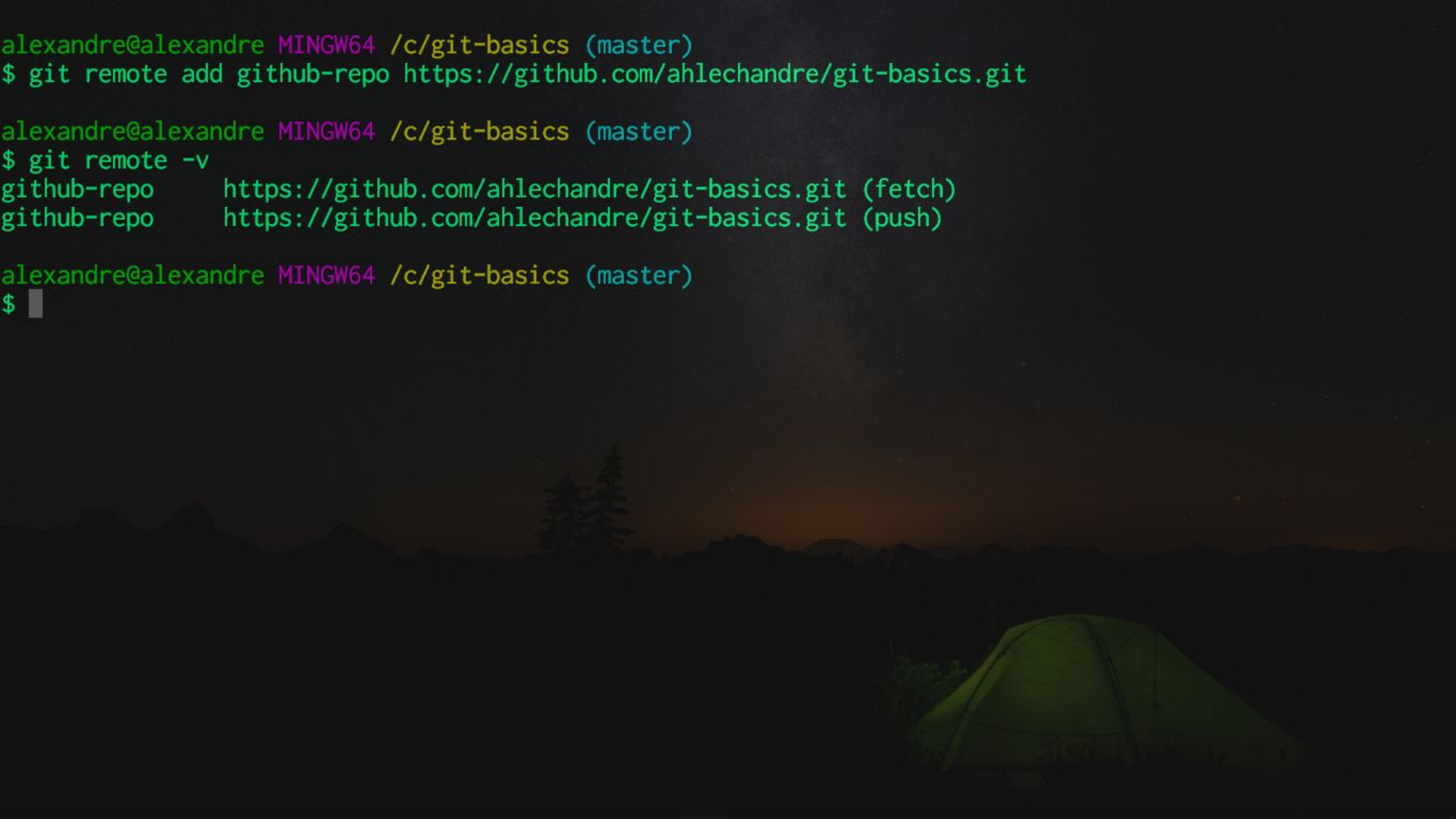
5. Trabalhando com remotos

```
$ git remote
```

5.1. Adicionando remotos

```
$ git remote add <remote>
```

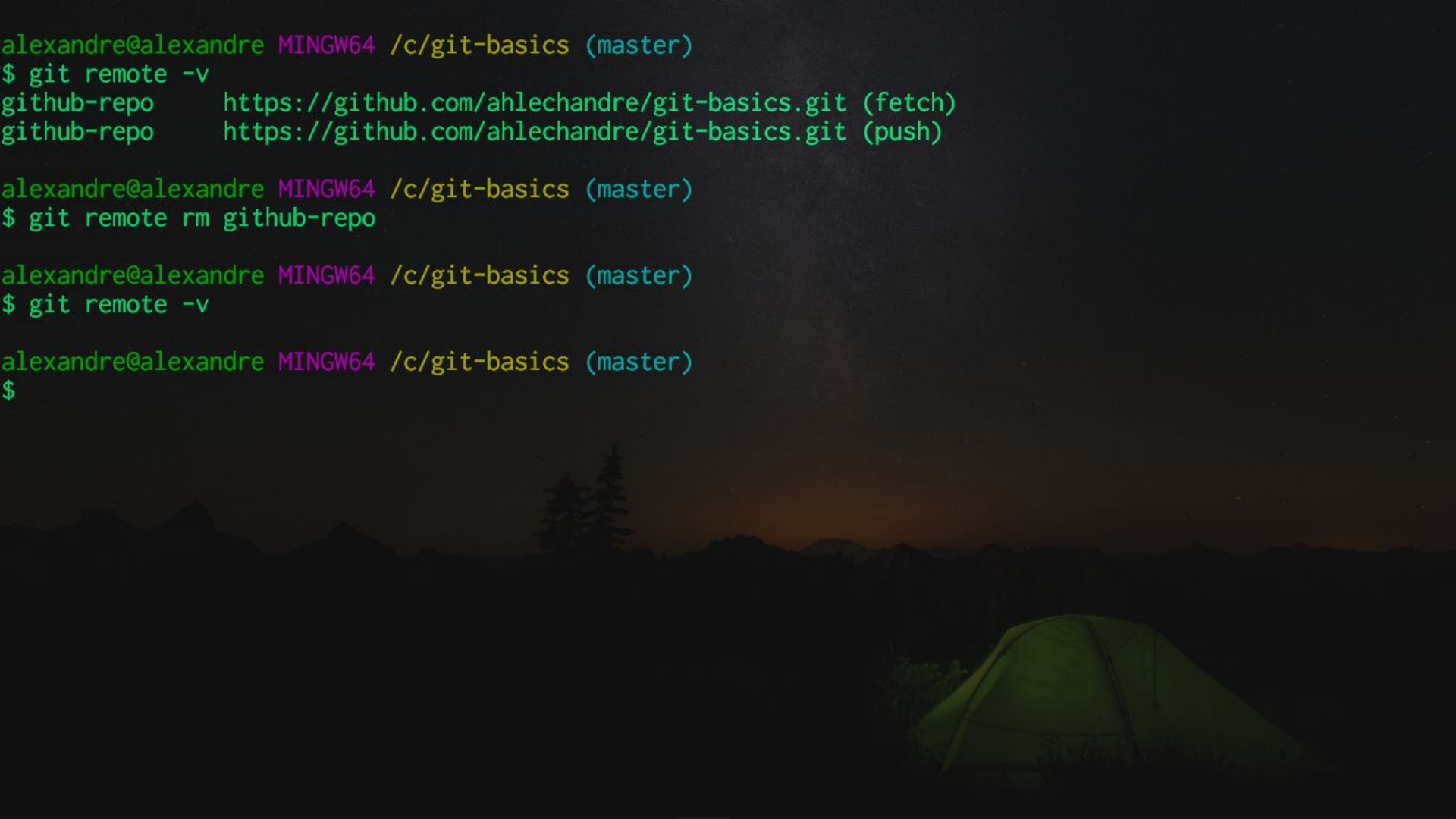
```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git remote add github-repo https://github.com/ahlechandre/git-basics.git

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git remote -v
github-repo     https://github.com/ahlechandre/git-basics.git (fetch)
github-repo     https://github.com/ahlechandre/git-basics.git (push)

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```



5.2. Removendo remotos

```
$ git remote rm <remote>
```

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ git remote -v
github-repo     https://github.com/ahlechandre/git-basics.git (fetch)
github-repo     https://github.com/ahlechandre/git-basics.git (push)

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git remote rm github-repo

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git remote -v

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```



5.3. “Puxando” os arquivos do repositório remoto

```
$ git pull <remote> <local>
```

ahlechandre / git-basics

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

No description or website provided. — Edit

1 commit 1 branch 0 releases 1 contributor

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

ahlechandre Initial commit Latest commit 5fcbdd3 14 hours ago

LICENSE	Initial commit	14 hours ago
README.md	Initial commit	14 hours ago

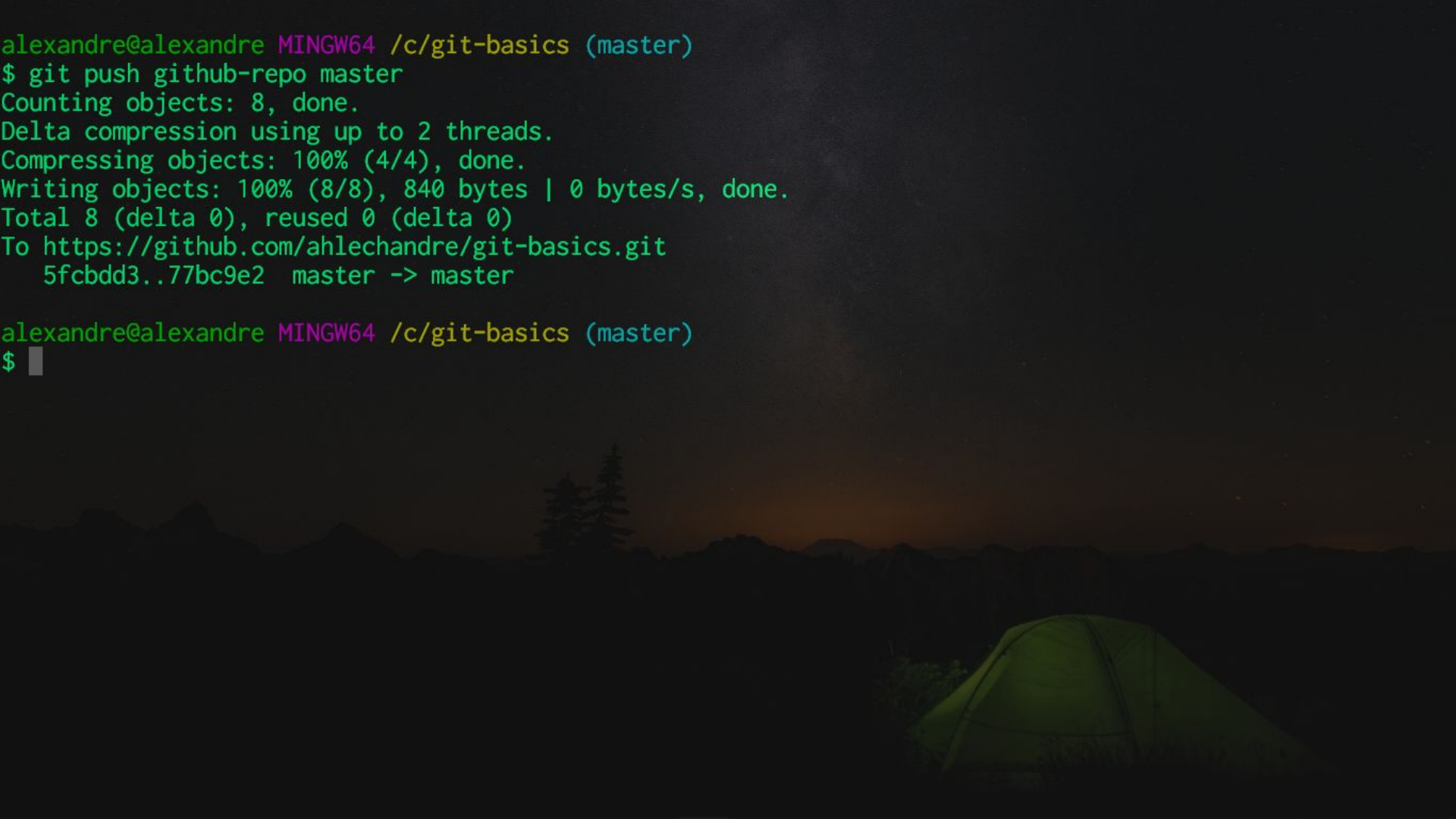
README.md

git-basics

```
alexandre@alexandre MINGW64 /c/git-basics (master)
$ ls
CONTRIBUTING.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ git pull github-repo master
warning: no common commits
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
From https://github.com/ahlechandre/git-basics
 * branch            master     -> FETCH_HEAD
 * [new branch]      master     -> github-repo/master
Merge made by the 'recursive' strategy.
 LICENSE   | 21 ++++++++++++++++++++
 README.md |  1 +
 2 files changed, 22 insertions(+)
 create mode 100644 LICENSE
 create mode 100644 README.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$ ls
CONTRIBUTING.md  LICENSE  README.md

alexandre@alexandre MINGW64 /c/git-basics (master)
$
```

5.4. “Empurrando” os arquivos para repositório remoto

```
$ git push <remote> <local>
```


alexandre@alexandre MINGW64 /c/git-basics (master)
$ git push github-repo master
Counting objects: 8, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (8/8), 840 bytes | 0 bytes/s, done.
Total 8 (delta 0), reused 0 (delta 0)
To https://github.com/ahlechandre/git-basics.git
5fcbdd3..77bc9e2 master -> master
alexandre@alexandre MINGW64 /c/git-basics (master)
$

ahlechandre / git-basics

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings

Branch: master git-basics / CONTRIBUTING.md Find file Copy path

ahlechandre versao 0.2 94952bc an hour ago

1 contributor

2 lines (1 sloc) 40 Bytes Raw Blame History

Controle de Versao com Git (versao 0.2)

6. *Deploy* com Git

6.1. VINT (grupo de pesquisa e extensão)

http://vint.roo.ufmt.br

VINT

- Página inicial
- Sobre
- Contato
- Membros
- Programas
- Projetos
- Publicações
- Pesquisar
- Entrar

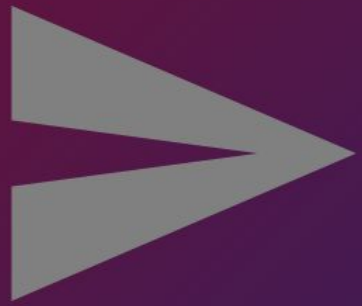
nputação Visual e Interativa

osso grupo de pesquisa e

e o VINT – Computação Visual e Interativa

Entre em contato

Fale conosco caso você tenha alguma dúvida ou esteja interessado por algo que conheceu aqui



s do VINT

Os programas

vint

ACTIONS
Clone
Create branch
Create pull request
Compare
Fork

NAVIGATION
Overview
Source
Commits
Branches
Pull requests
Downloads

Settings

Source

master | vint / | + New file

config			
plugins/vint			
themes/vint			
.gitignore	241 B	2016-06-06	pre-install configs
README.md	2.5 KB	2016-06-05	pre-install

Installation wizard for October

The wizard installation is a recommended way to install October. It is simpler than the command-line installation and doesn't require any special skills.

1. Prepare a directory on your server that is empty. It can be a sub-directory, domain root or a sub-domain.
2. Download the installer archive file.
3. Unpack the installer archive to the prepared directory.
4. Grant writing permissions on the installation directory and all its subdirectories and files.
5. Navigate to the install.php script in your web browser.
6. Follow the installation instructions.

Minimum System Requirements

October CMS has a few system requirements:

- PHP 5.4 or higher
- PDO PHP Extension
- cURL PHP Extension
- MCrypt PHP Extension
- ZipArchive PHP Library

Referências

Git, Book. Disponível em: <https://git-scm.com/book/en/v2>. Acesso em 2 de agosto de 2016.

Wikipedia, Git (software). Disponível em: <https://en.wikipedia.org/wiki/Git_(software)>. Acesso em 2 de agosto de 2016.