

**GIT, GITHUB,  
GITHUB ACTIONS**



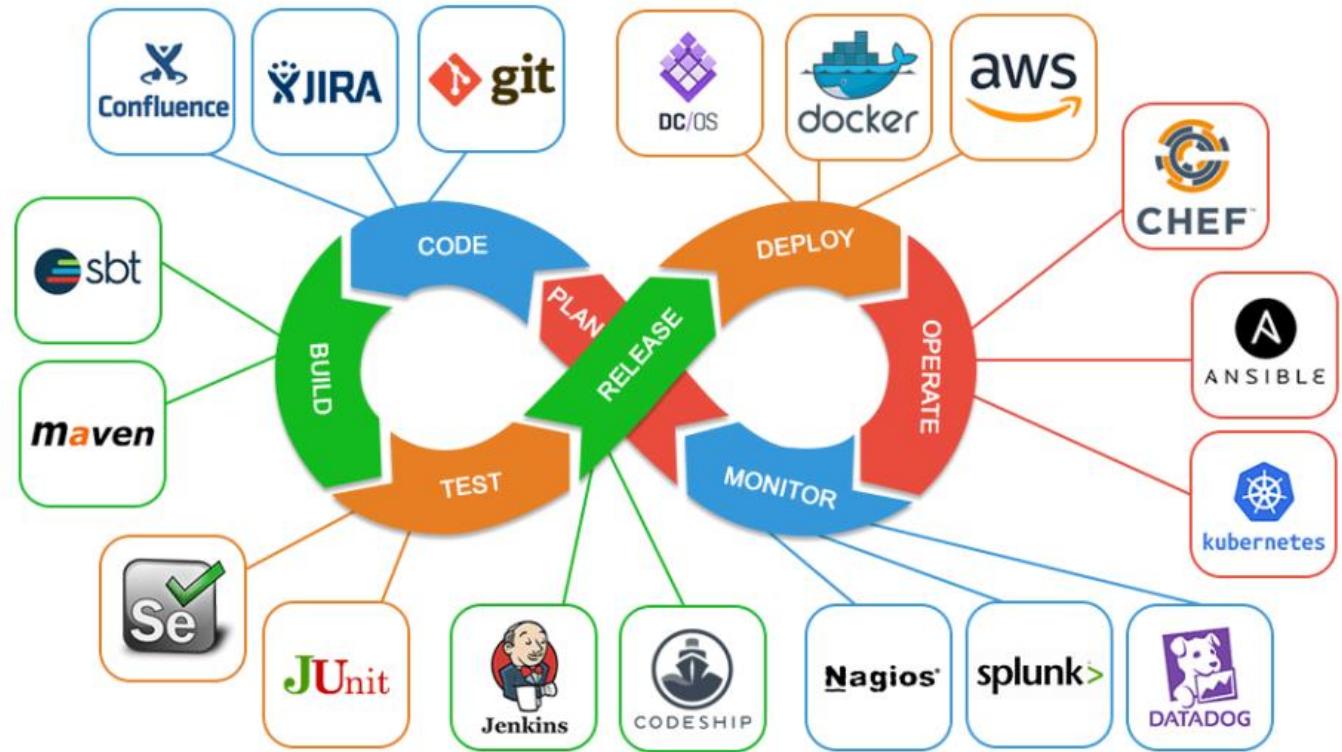
# ÍNDICE

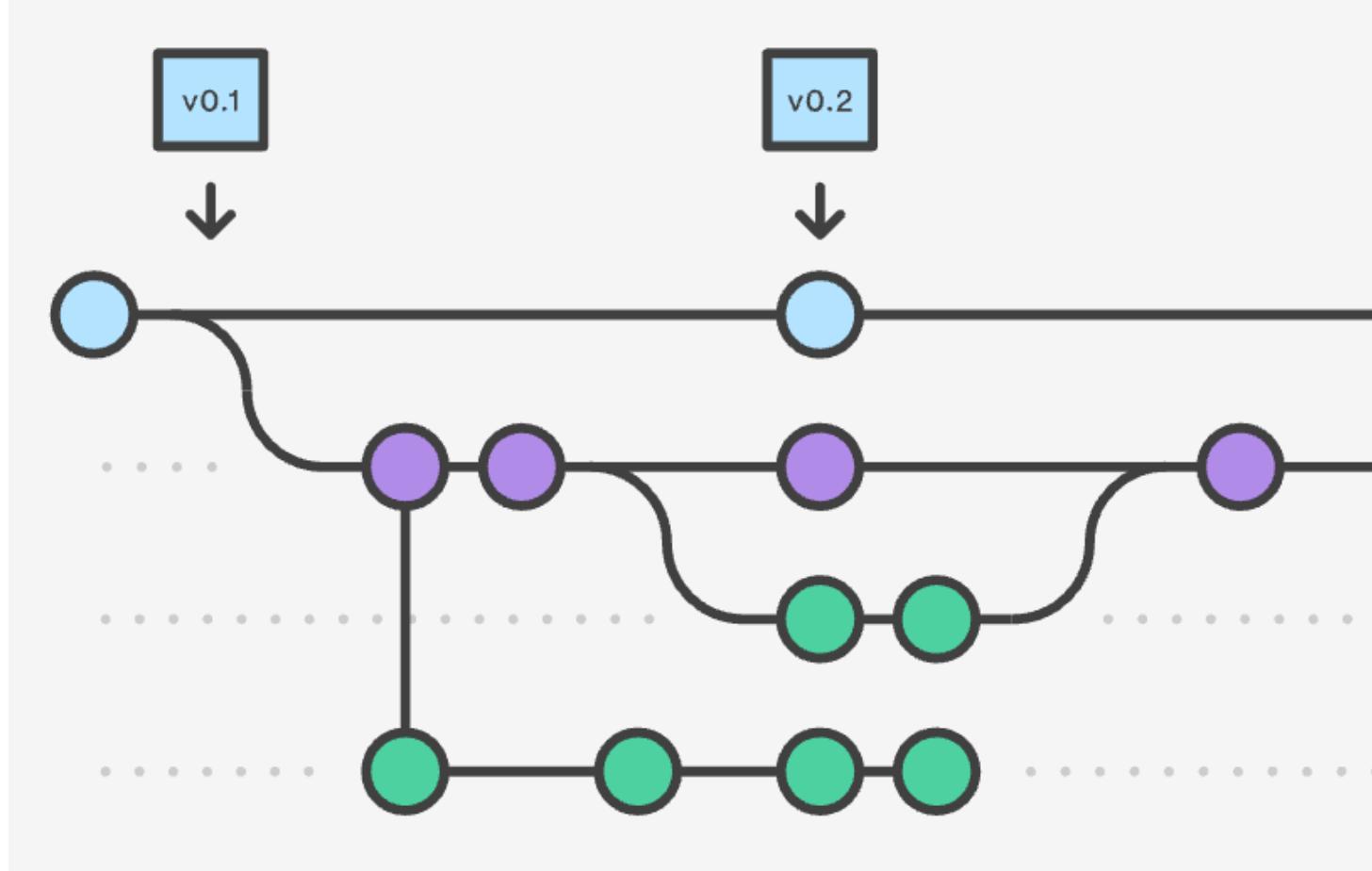
- Introducción
- Ramas
- Colaboración en Github
- Buenas prácticas
- GitFlow
- Github Actions
- Proyecto final Github

# CONTROL DE VERSIONES



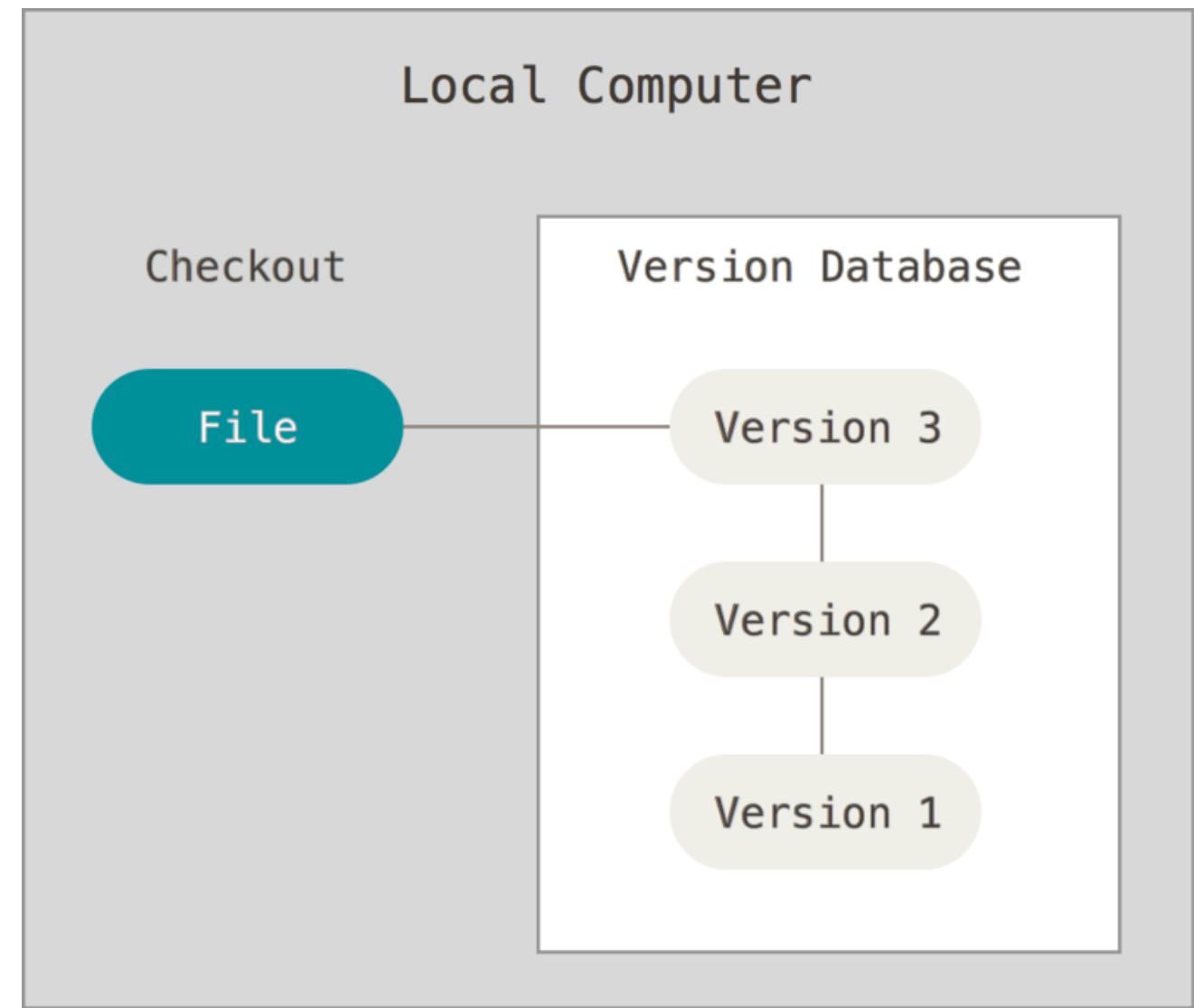
# DE V O P S



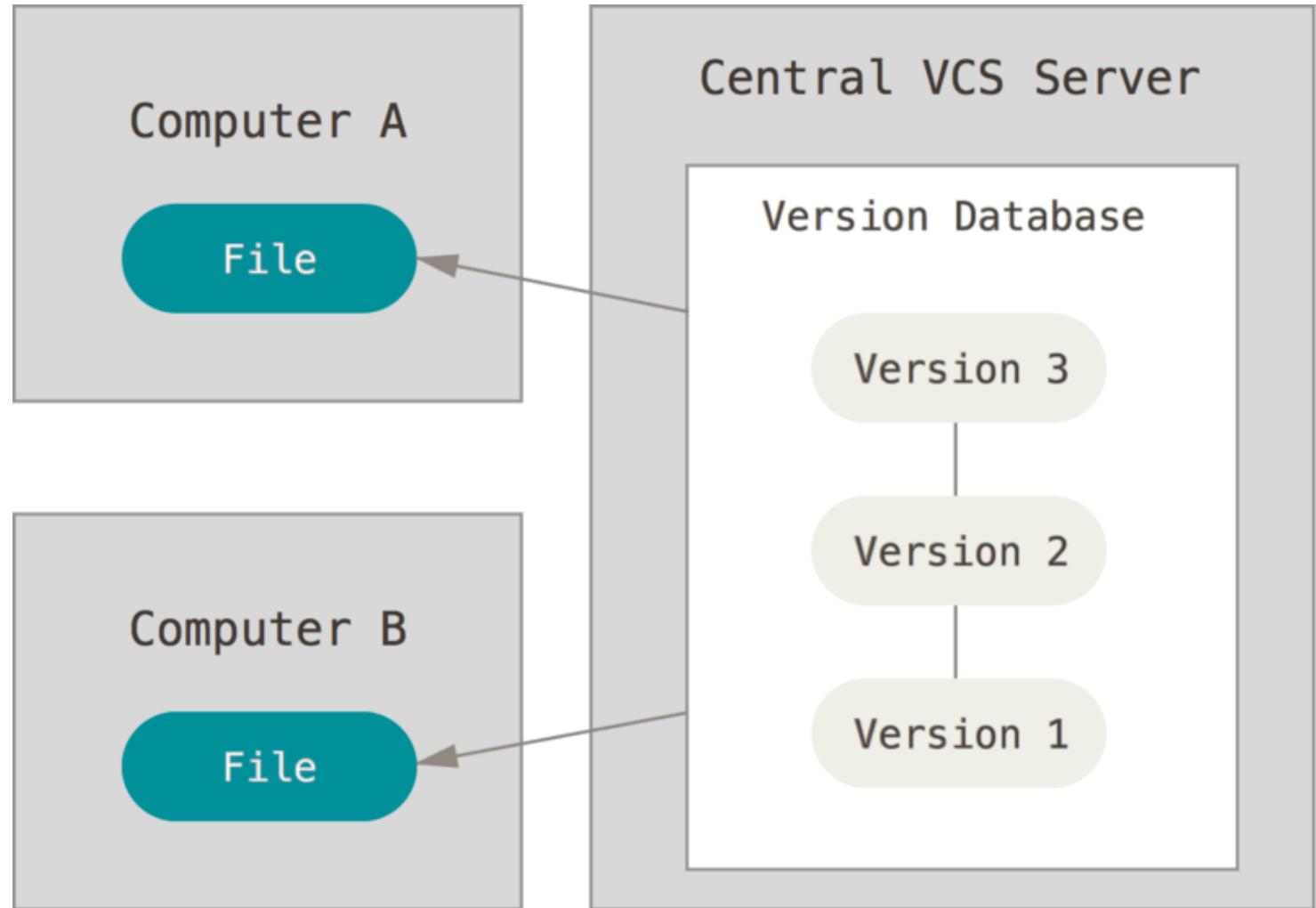


¿POR QUÉ USAR UN VCS?

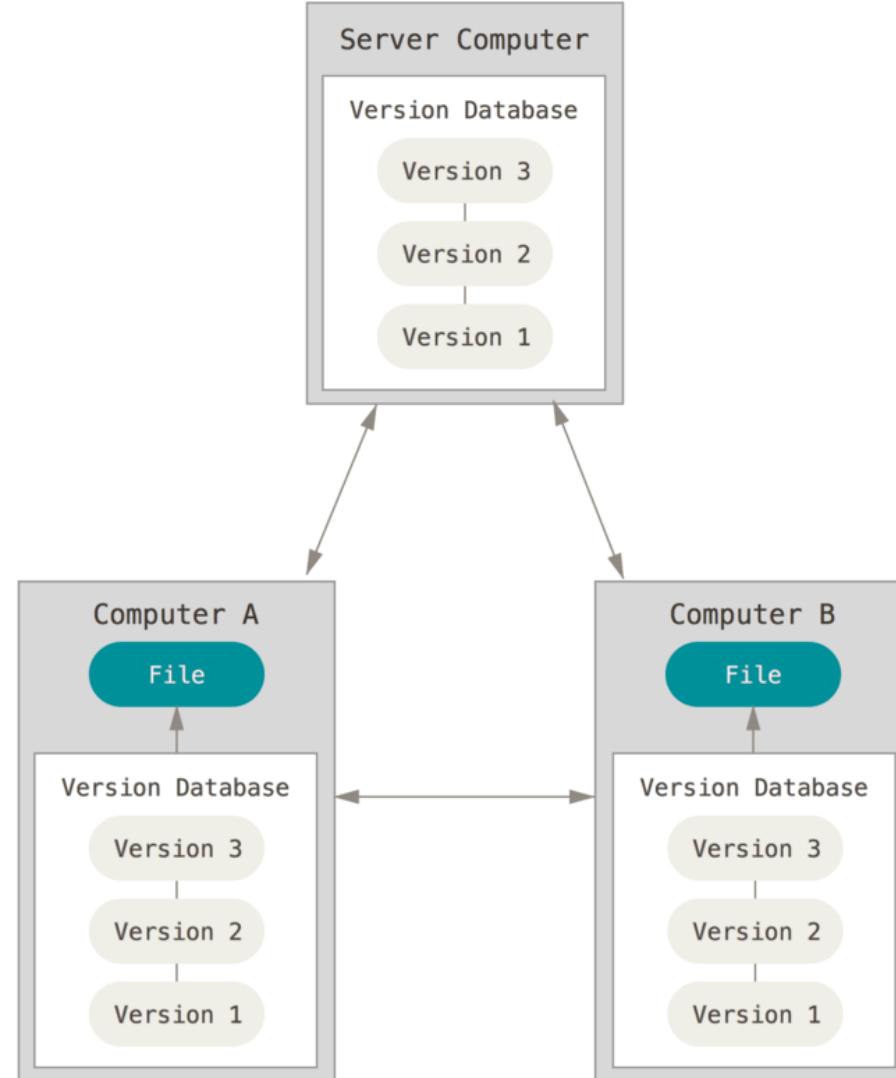
# SISTEMAS DE CONTROL DE VERSIONES LOCALES



# SISTEMAS DE CONTROL DE VERSIONES CENTRALIZA DOS



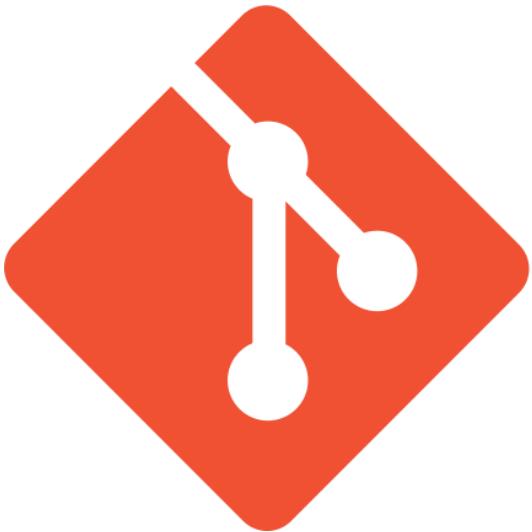
# SISTEMAS DE CONTROL DE VERSIONES DISTRIBUIDOS



# HISTORIA DE GIT

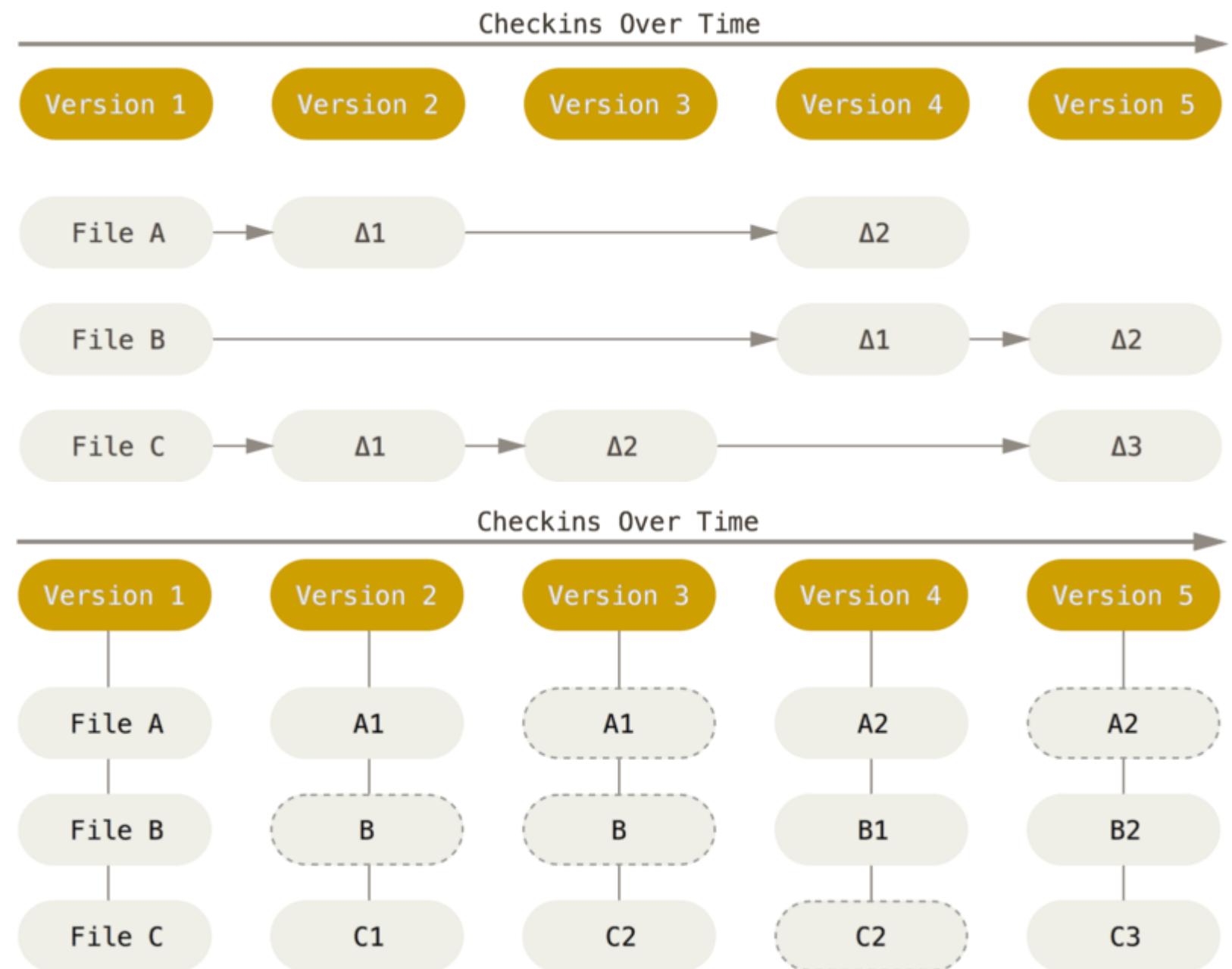


¿QUÉ ES  
GIT?

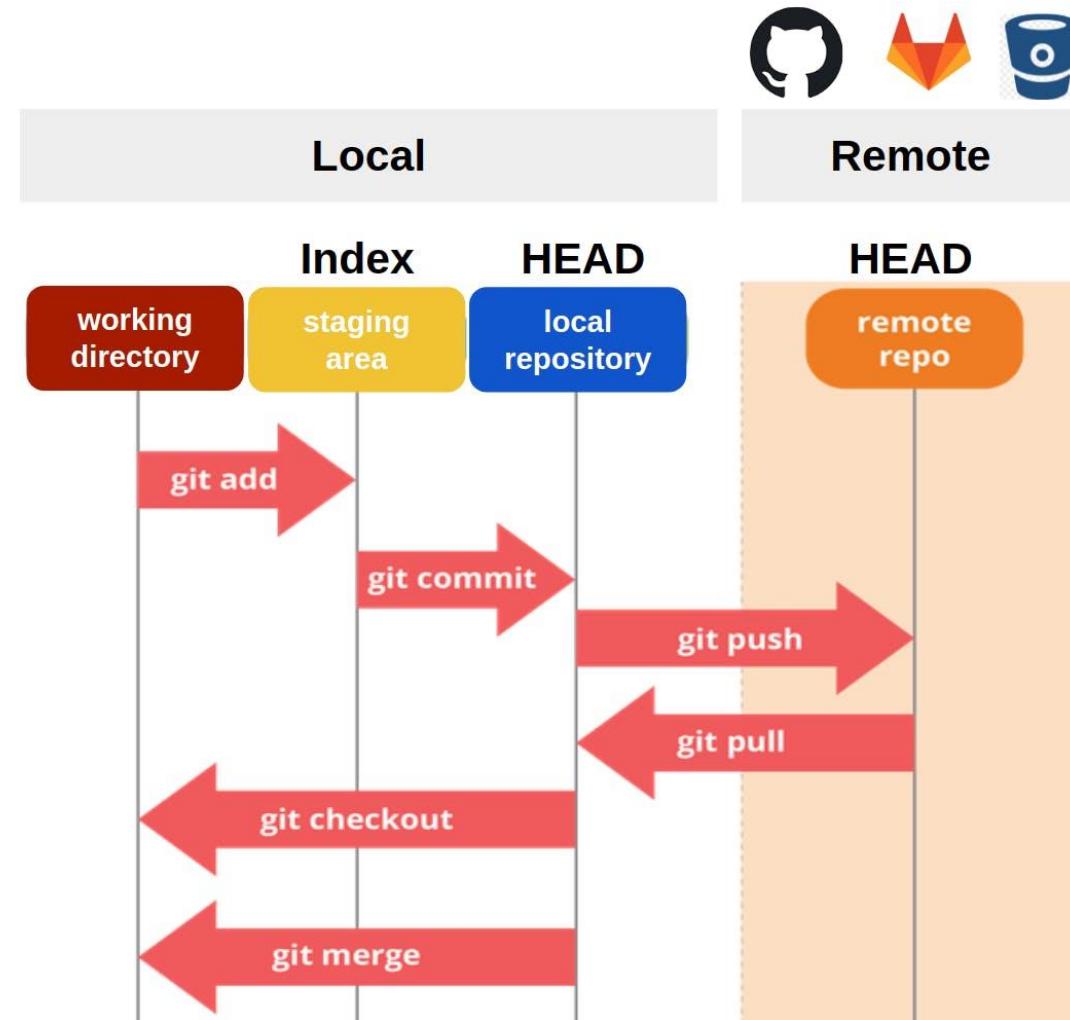


git

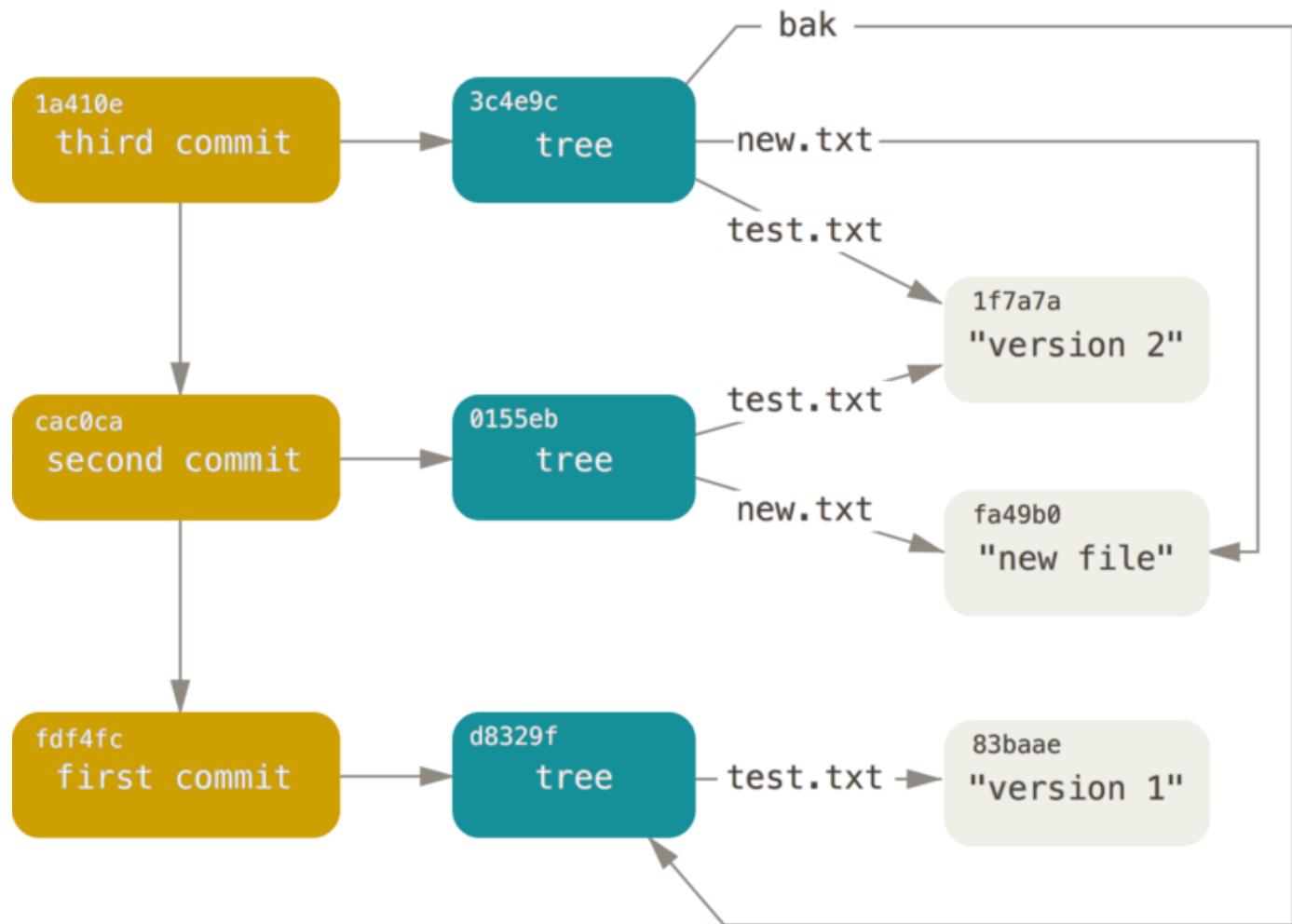
# C O P I A S I N S T A N T Á N E A S , N O D I F E R E N C I A S



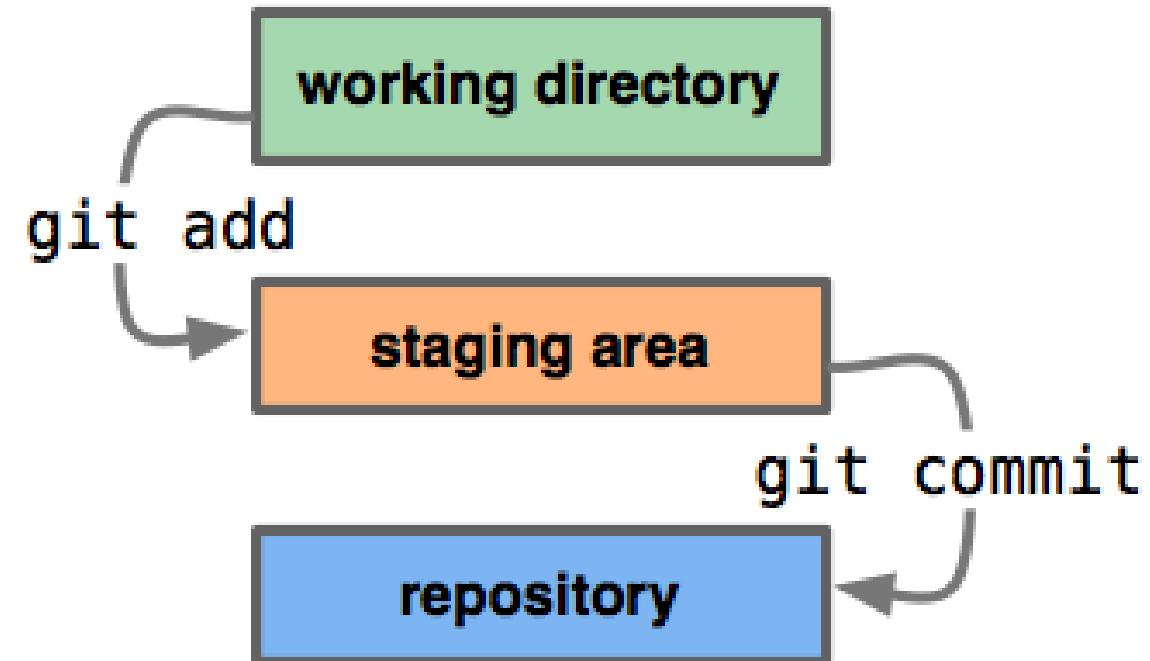
# OPERACIONES LOCALES



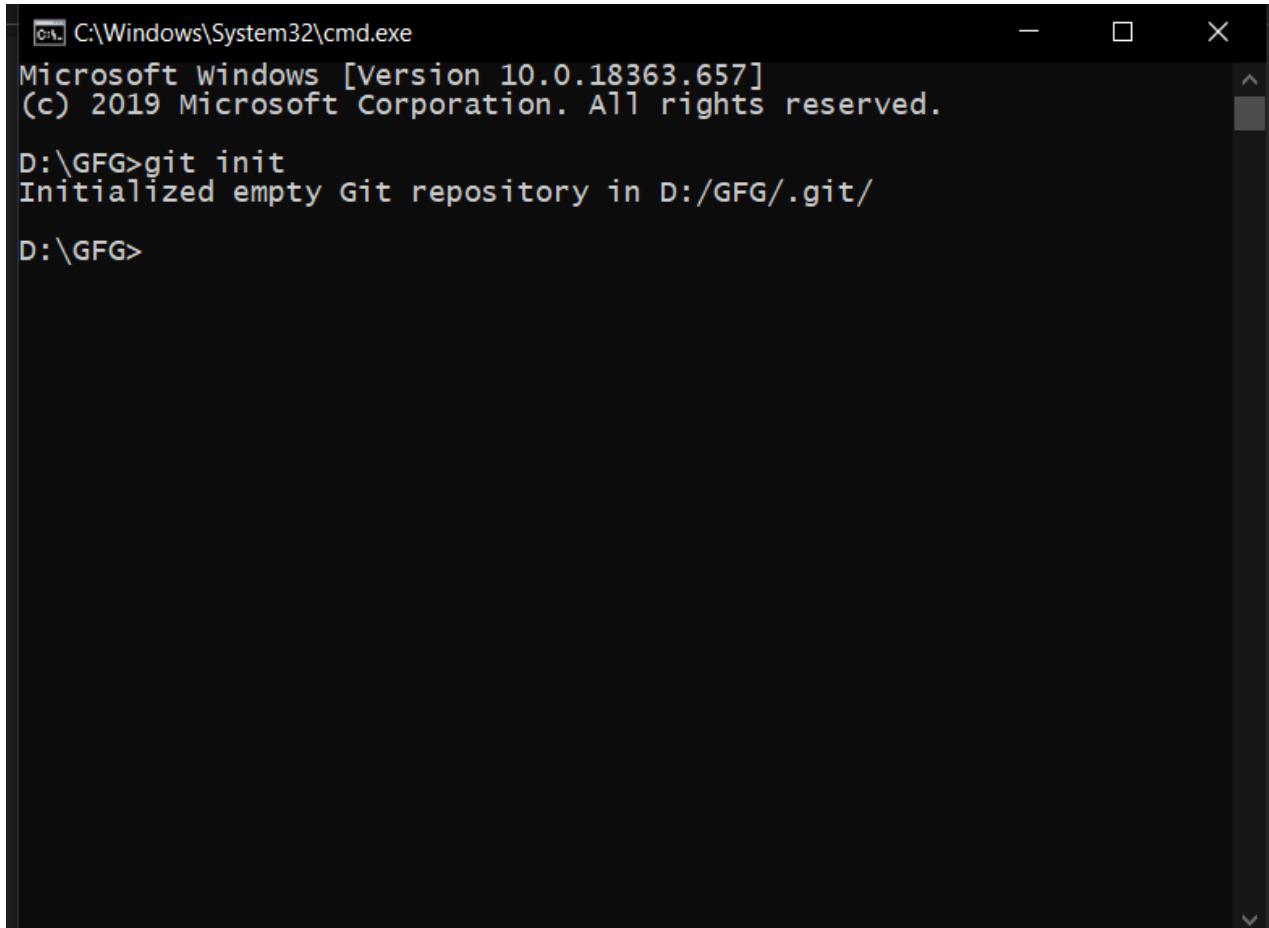
# INTEGRIDAD



# ESTADOS DE GIT

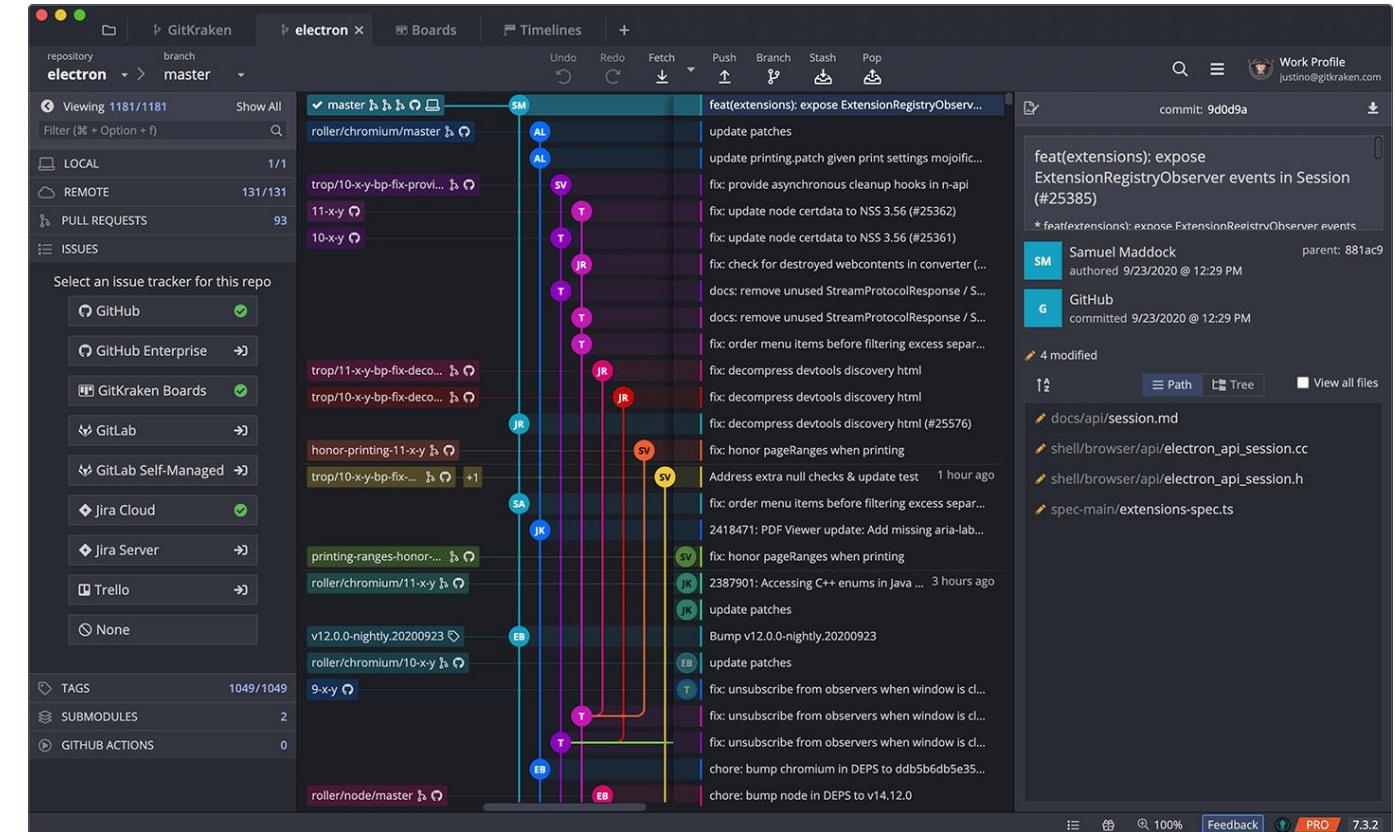


# LINEA DE COMANDOS

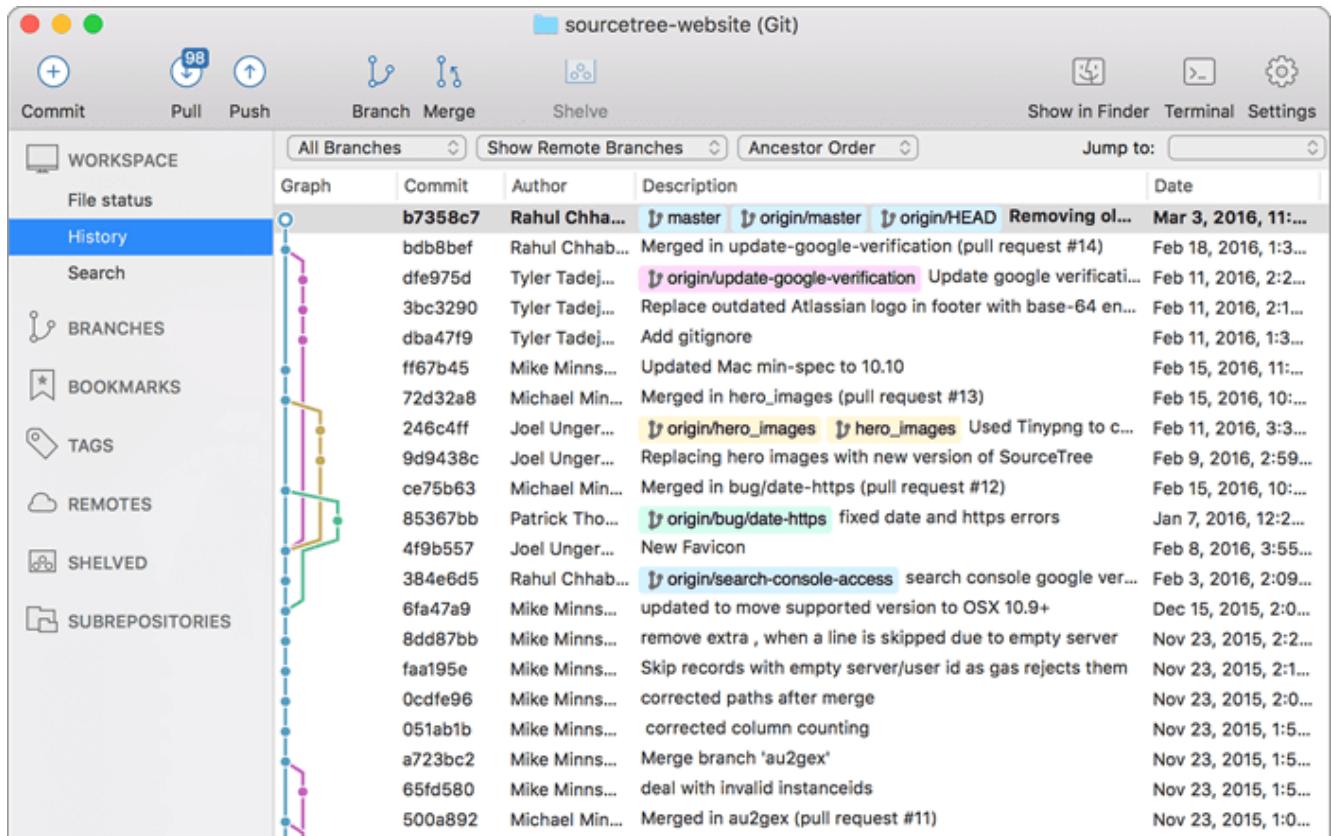


A screenshot of a Windows Command Prompt window titled "C:\Windows\System32\cmd.exe". The window shows the following text:  
Microsoft Windows [Version 10.0.18363.657]  
(c) 2019 Microsoft Corporation. All rights reserved.  
D:\GFG>git init  
Initialized empty Git repository in D:/GFG/.git/  
D:\GFG>

# GITKRAKEN



# SOURCETREE



# GITHUB DESKTOP

The screenshot shows the GitHub Desktop application interface. At the top, there's a header bar with tabs for 'File', 'Edit', 'View', 'Repository', 'Branch', and 'Help'. Below the header, the 'Current repository' dropdown is set to 'desktop', the 'Current branch' dropdown is set to 'esc-pr' (with PR #3972), and the 'Fetch origin' button indicates it was last fetched 2 minutes ago.

The main area has two tabs: 'Changes' (selected) and 'History'. The 'Changes' tab displays a list of commits from the 'esc-pr' branch:

- Add event handler to dropdown component (iAmWillShepherd and Markus Olsson committed a day ago)
- Add event handler to dropdown component (iAmWillShepherd and Markus Olsson co...)
- Move escape behavior to correct compo... (iAmWillShepherd and Markus Olsson co...)
- Remove event handler from the branches.. (iAmWillShepherd and Markus Olsson co...)
- Merge branch 'master' into esc-pr (iAmWillShepherd committed a day ago)
- Merge pull request #4044 from desktop/... (Neha Batra committed a day ago)
- Merge pull request #4070 from desktop/... (Brendan Forster committed 2 days ago)
- bump to beta3 (Brendan Forster committed 2 days ago)
- Merge pull request #4057 from desktop/... (Brendan Forster committed 2 days ago)
- Merge pull request #4067 from desktop/... (Brendan Forster committed 2 days ago)
- Release to 1.1.0-beta2 (Neha Batra committed 2 days ago)

The 'History' tab is also visible. On the right side, a detailed view of the commit 'Add event handler to dropdown component' is shown, comparing the code changes between commit 145 and 148. The code snippet is as follows:

```
@@ -145,6 +145,10 @@ export class ToolbarDropdown extends React.Component<
 145      this.state = { clientRect: null }
 146  }
 147
 148 + private get isOpen() {
 149 +   return this.props.dropdownState === 'open'
 150 + }
 151 +
 152     private dropdownIcon(state: DropdownState): OcticonSymbol {
 153       // @TODO: Remake triangle octicon in a 12px version,
 154       // right now it's scaled badly on normal dpi monitors.
 155
 156     @@ -249,6 +253,13 @@ export class ToolbarDropdown extends React.Component<
 249      }
 250  }
 251
 252 + private onFoldoutKeyDown = (event: React.KeyboardEvent<HTMLInputElement>) => {
 253 +   if (!event.defaultPrevented && this.isOpen && event.key === 'Escape') {
 254 +     event.preventDefault()
 255 +   }
 256 + }
```

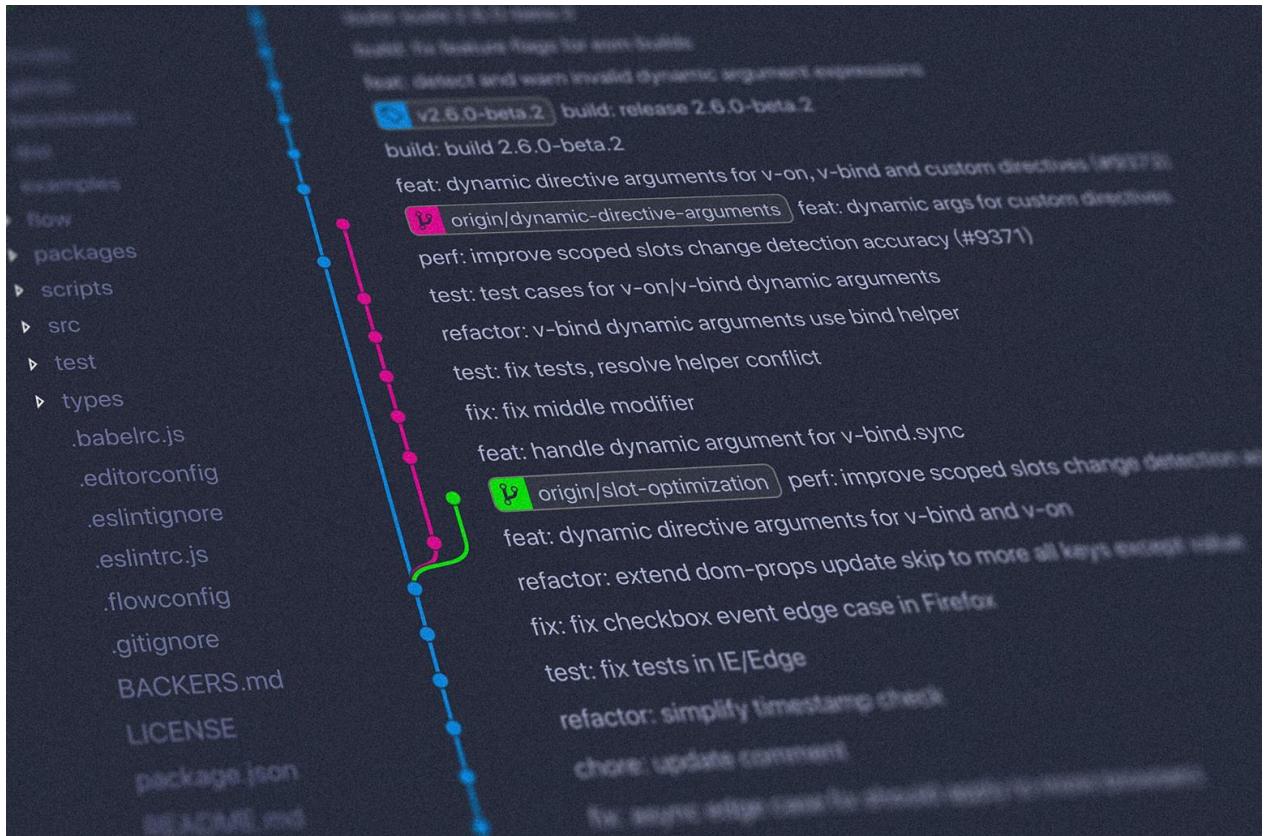
GITHUB



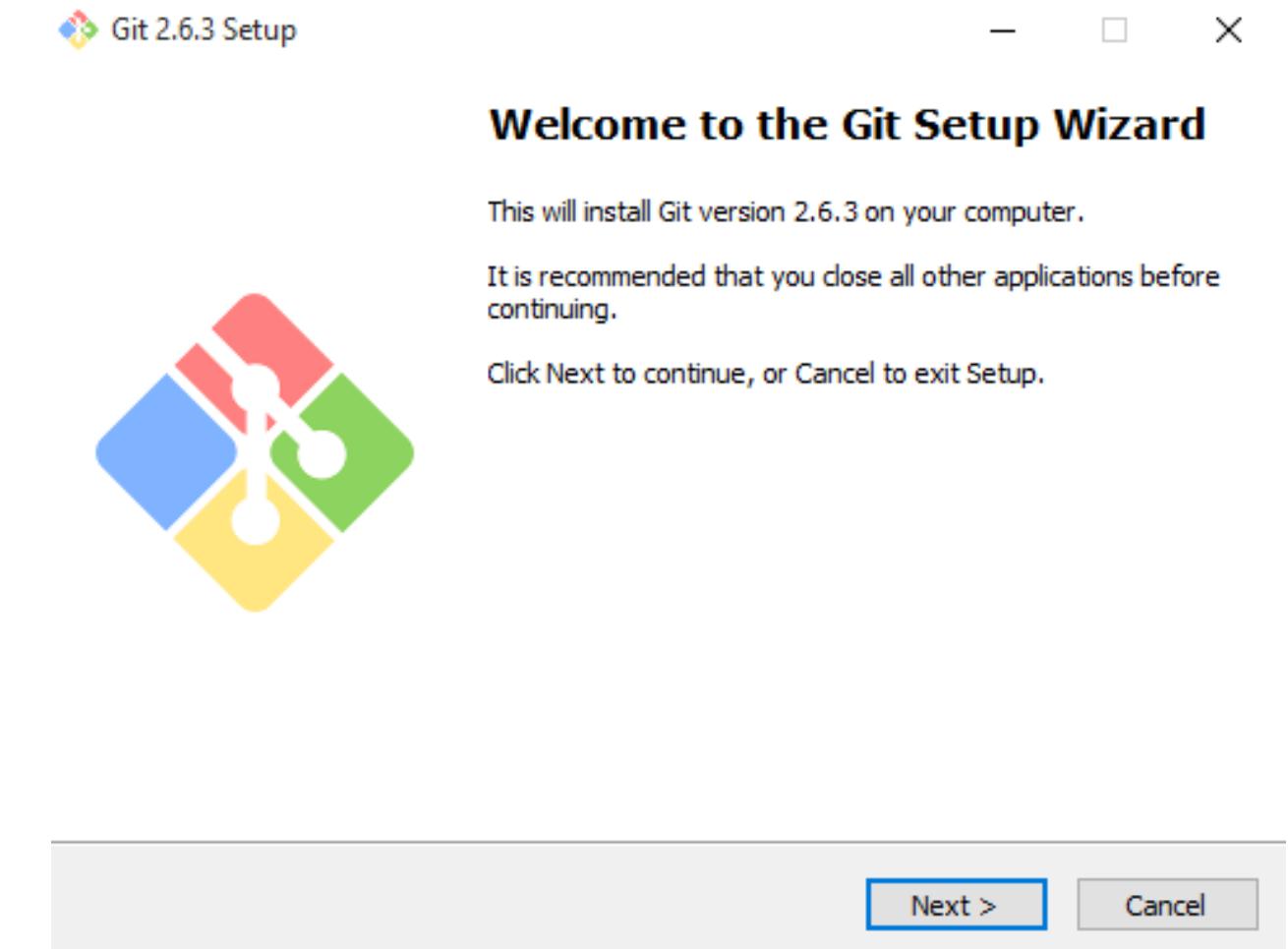
# G I T   V S G I T H U B

 git	 GitHub
1. It is a software	1. It is a service
2. It is installed locally on the system	2. It is hosted on Web
3. It is a command line tool	3. It provides a graphical interface
4. It is a tool to manage different versions of edits, made to files in a git repository	4. It is a space to upload a copy of the <b>Git</b> repository
5. It provides functionalities like Version Control System Source Code Management	5. It provides functionalities of Git like VCS, Source Code Management as well as adding few of its own features

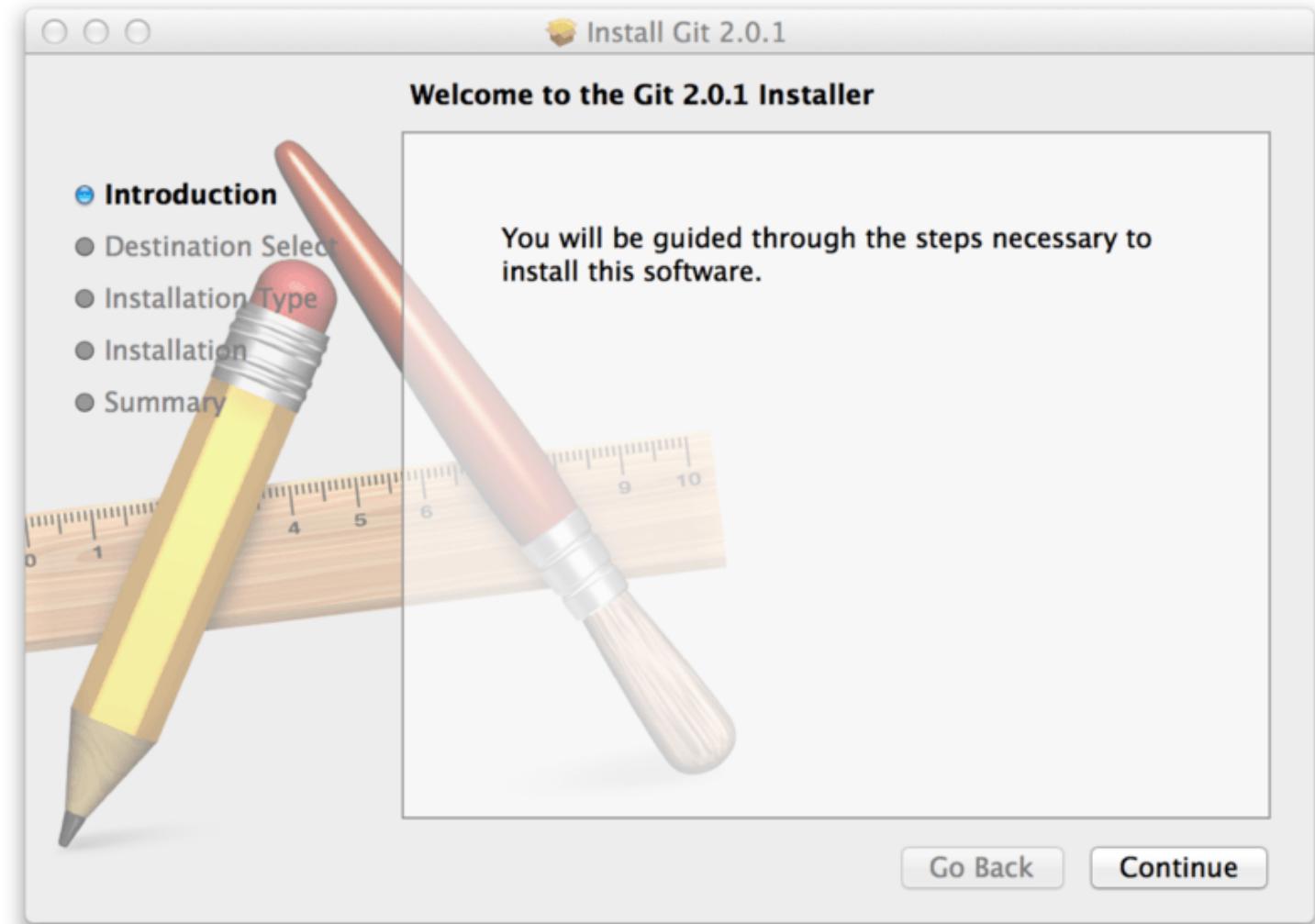
# LABORATORIO 1



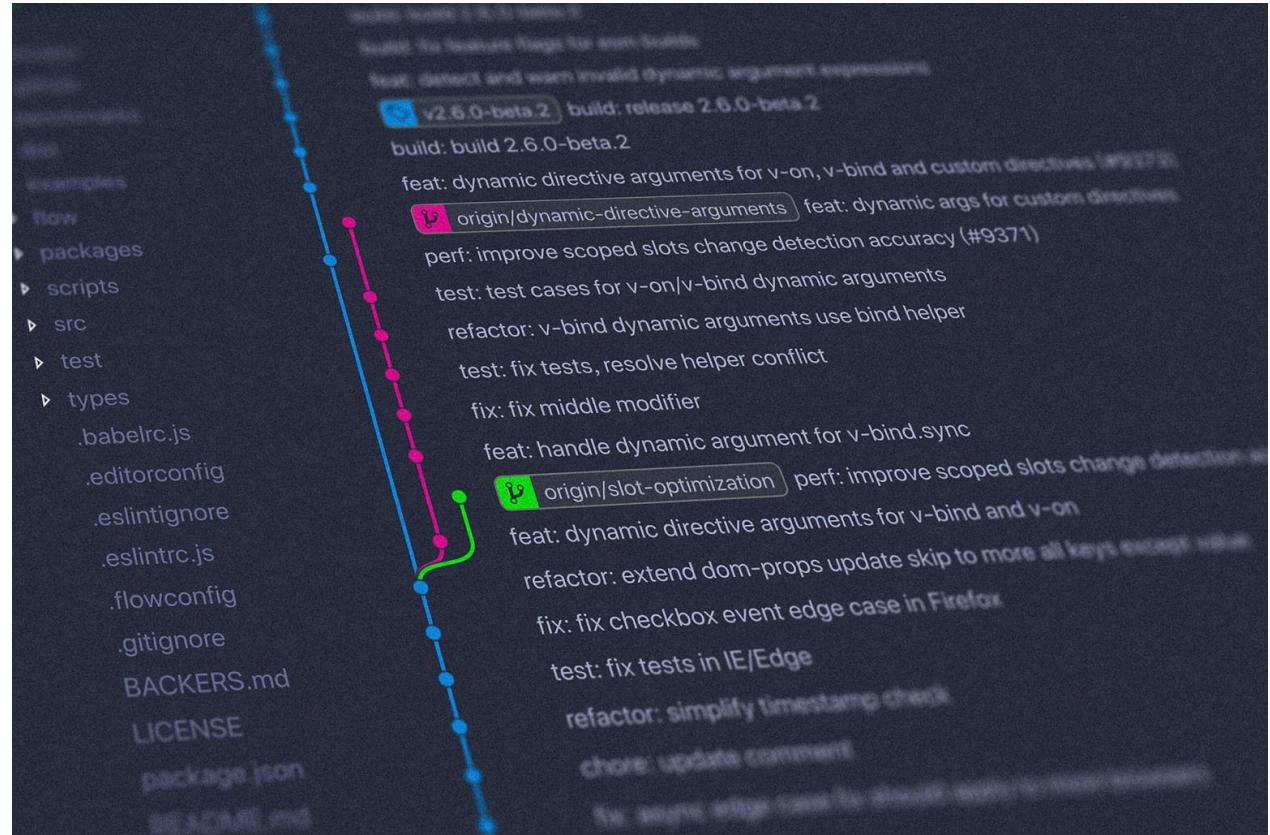
# INSTALACIÓN DE GIT EN WINDOWS



# INSTALACIÓN N DE GIT EN MAC



# LABORATORIO 2



# ATOM

The screenshot shows the Atom code editor interface. On the left, there is a file tree with the following structure:

- scipion
- gt
- config
- install
- pyworkflow

  - \_\_init\_\_.py
  - change\_rpath.py
  - clean.py
  - find\_deps.py
  - fix\_links.py
  - monitor.py
  - ploter.py
  - run\_apache.py
  - run\_tests.py
  - split\_stacks.py
  - sync\_data.py

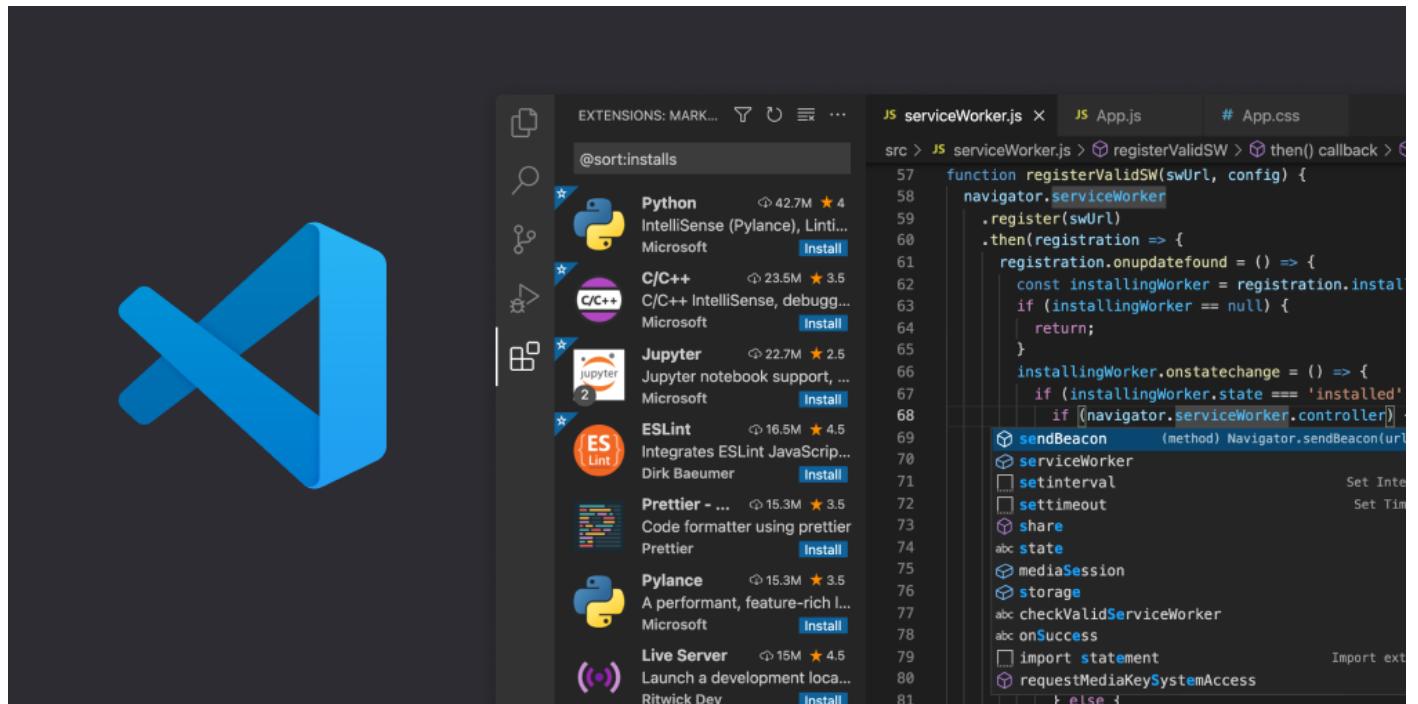
- software
- .gitignore
- README.md
- scipion
- SConstruct

The main editor area displays a Python script named `sync_data.py` with the following content:

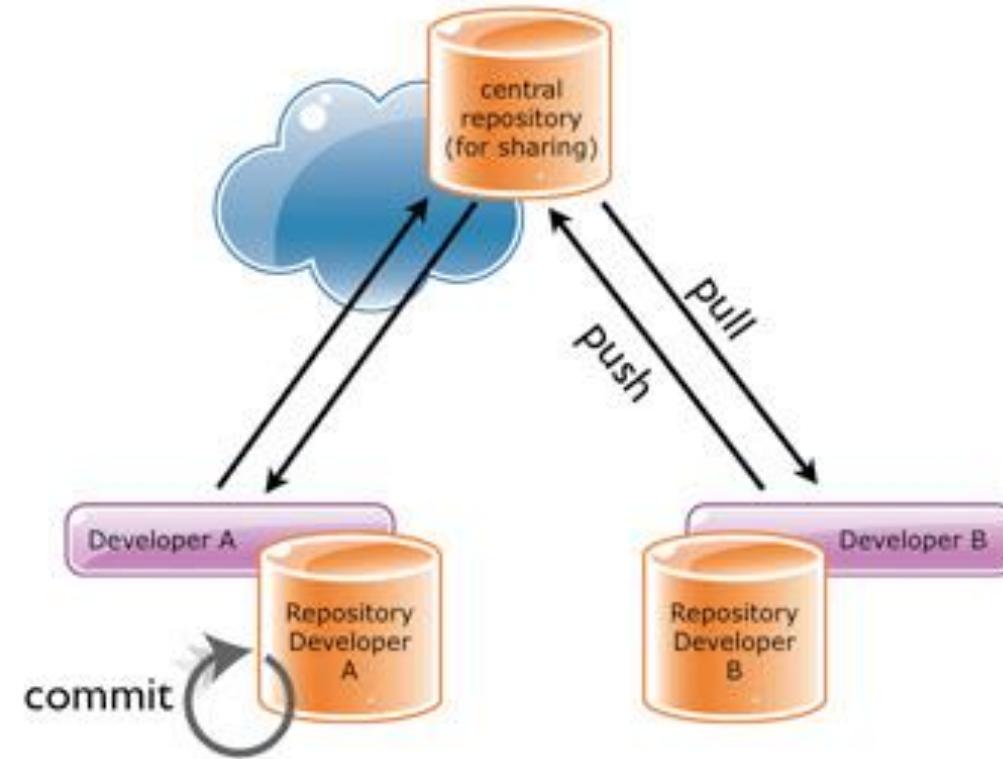
```
File Edit View Selection Find Packages Help
scripts\sync_data.py 1:1
177     add('--check-all', action='store_true',
178         help='See if there is any remote dataset not in sync with locals.')
179     add('--V', '--verbose', action='store_true', help='Print more details.')
180
181     return parser
182
183
184 def listDatasets(url):
185     """ Print a list of local and remote datasets """
186
187     tdir = os.environ['SCIPION_TESTS']
188     print "Local datasets in %s" % yellow(tdir)
189     for folder in sorted(os.listdir(tdir)):
190         if isdir(join(tdir, folder)):
191             if exists(join(tdir, folder, 'MANIFEST')):
192                 print " * %s" % folder
193             else:
194                 print " * %s (not in dataset format)" % folder
195
196     try:
197         print "\nRemote datasets in %s" % yellow(url)
198         for line in sorted(urlopen('%s/MANIFEST' % url)):
199             print " * %s" % line.strip('\r\n')
200     except Exception as e:
201         print "Error reading %s (%s)" % (url, e)
202
203
204 def check(dataset, url, verbose=False, updateMANIFEST=False):
205     """ See if our local copy of dataset is the same as the remote one.
206     Return True if it is (if all the checksums are equal), False if not.
207     """
208
209     def vlog(txt): sys.stdout.write(txt) if verbose else None # verbose log
210
211     vlog("Checking dataset %s ... %s dataset)" % (dataset))
212
213     if updateMANIFEST:
214         createMANIFEST(join(os.environ['SCIPION_TESTS'], dataset))
215     else:
216         vlog("(not updating local MANIFEST) ")
217
218     try:
219         md5sRemote = dict(x.split() for x in
220                           urlopen('%s/%s/MANIFEST' % (url, dataset)))
221
222         md5sLocal = dict(x.split() for x in
223                           open('%.MANIFEST' % dataset))
```

The status bar at the bottom right indicates: LF UTF-8 Python master.

V S C O D E



# REPOSITORIO DE GIT



# INICIALIZAR REPOSITORIO

```
> git init
```

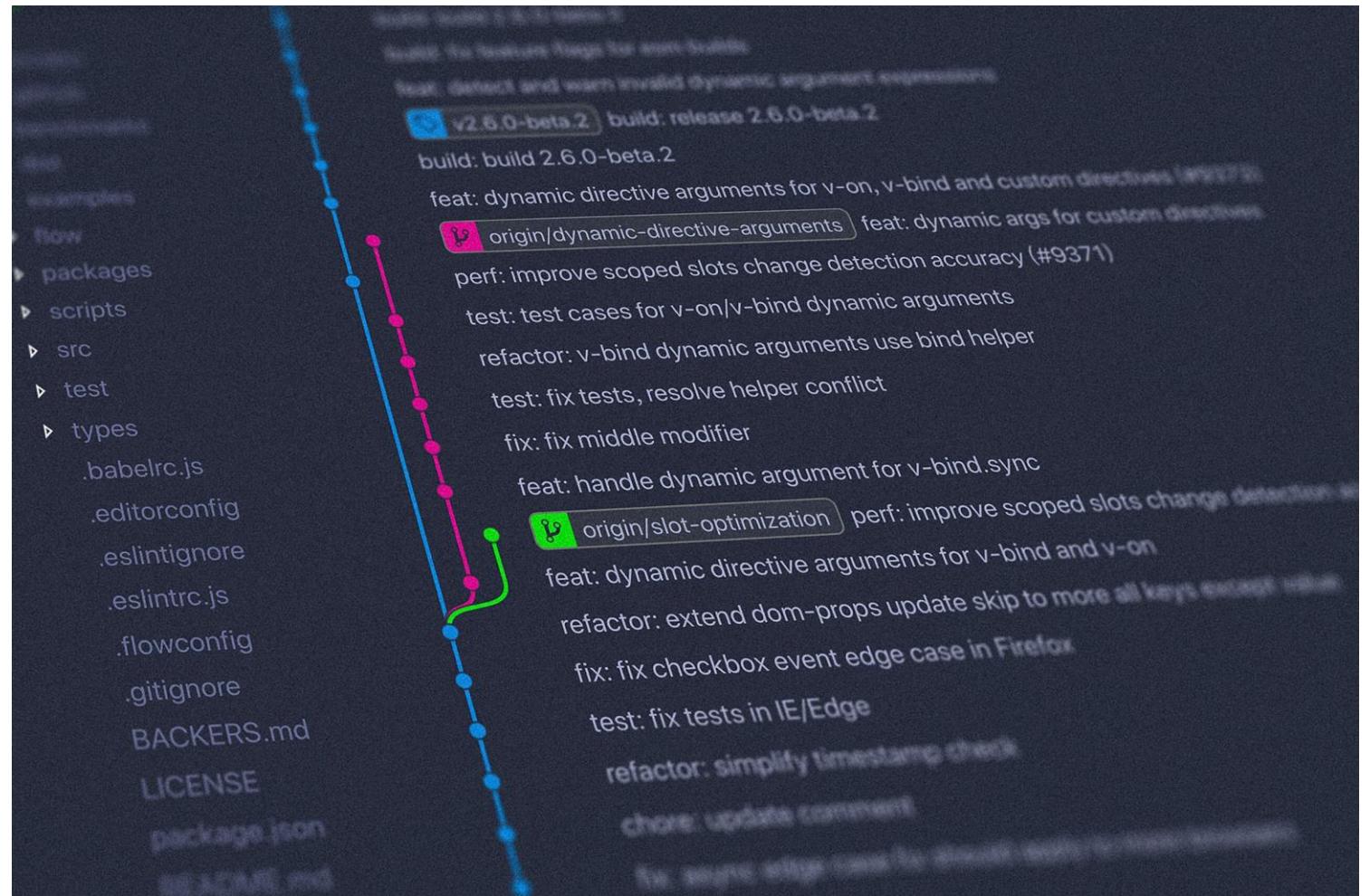
# CLONACIÓN

```
HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop (master)
$ cd "new folder"

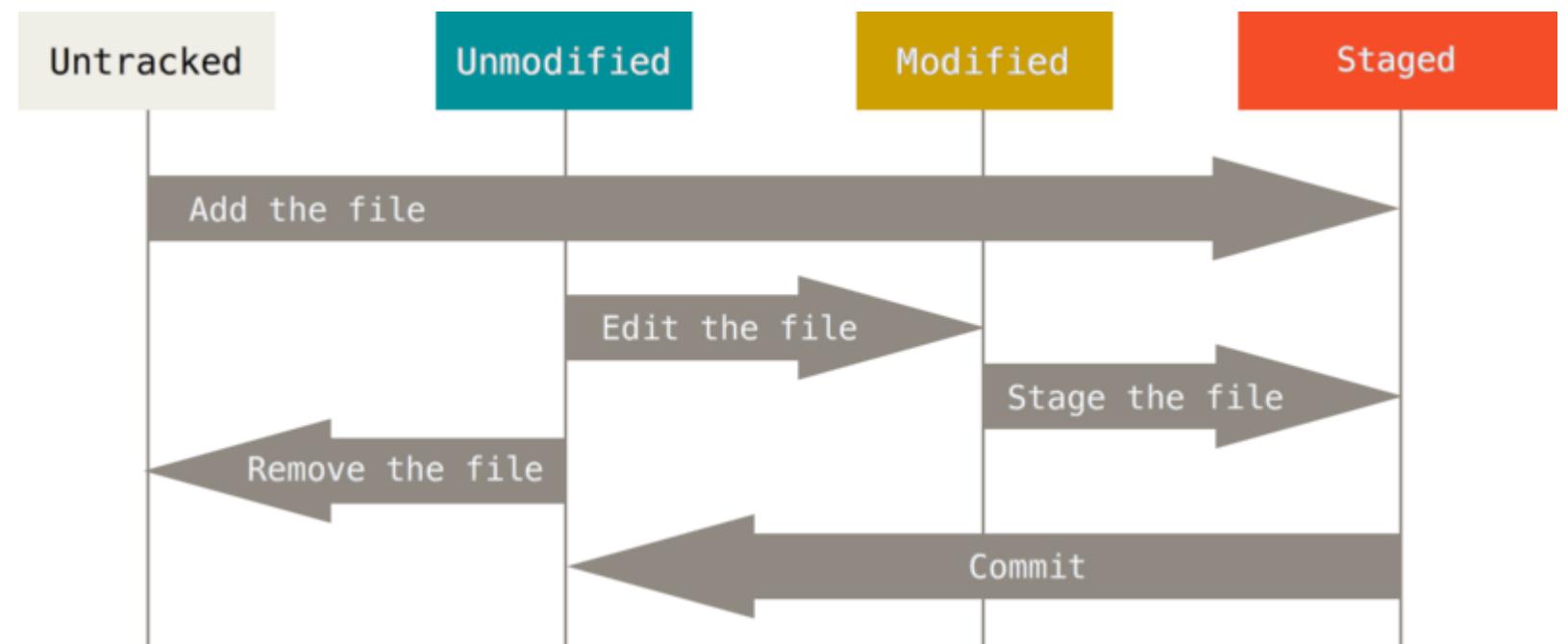
HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/new folder (master)
$ git clone https://github.com/IMDwivedi1/Git-Example.git
Cloning into 'Git-Example'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), done.

HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/new folder (master)
$
```

# LABORATORIO 3



# GUARDANDO CAMBIOS EN EL REPOSITORIO



# G I T   S T A T U S

```
HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/NewDirectory (master)
$ touch demofile

HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/NewDirectory (master)
$ git status
on branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    demofile

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

# G I T   A D D

```
HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/NewDirectory (master)
$ touch newfile.txt

HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/NewDirectory (master)
$ git status
on branch master

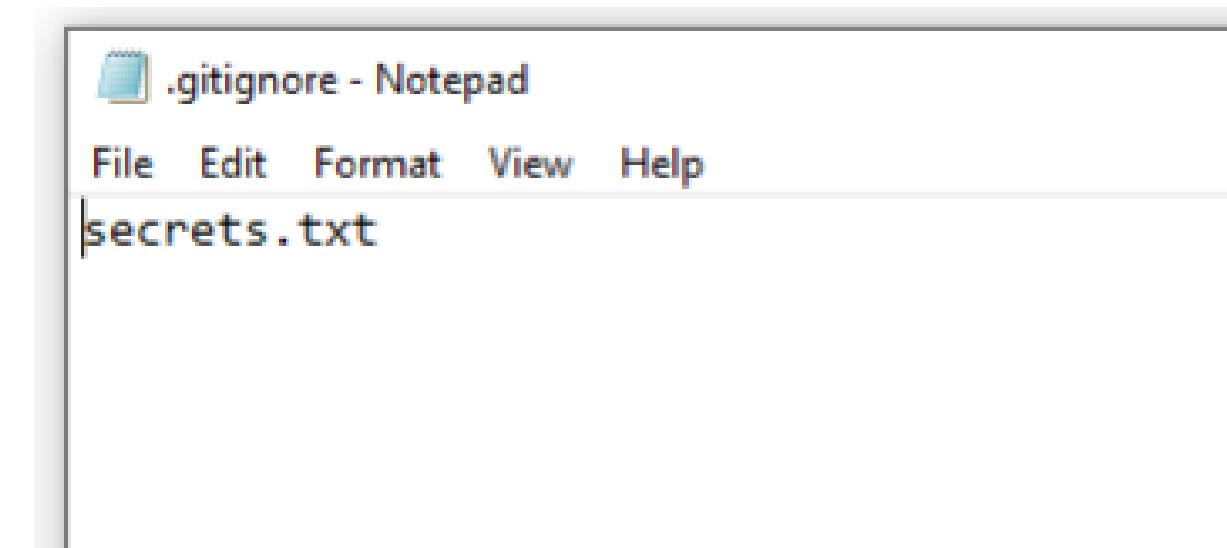
No commits yet

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

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

HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/NewDirectory (master)
$ git add newfile.txt
```

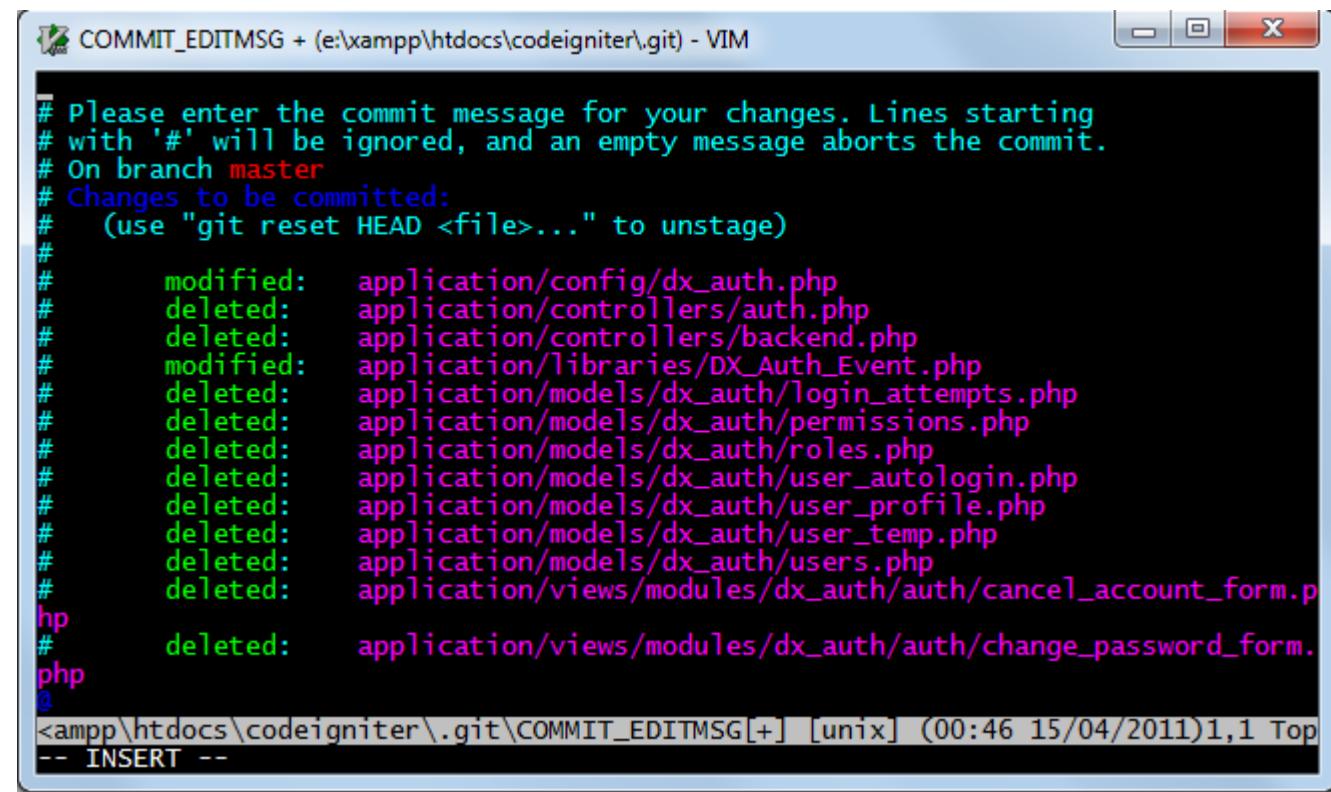
# G I T   I G N O R E



# G I T   D I F F

```
MINGW64:/e/Gitdiffnew
Lenovo@DESKTOP-LU5US00 MINGW64 /e/Gitdiffnew (master)
$ git diff 6eeda c87e1ed hello.txt
diff --git a/hello.txt b/hello.txt
index 01bafad..79cf701 100644
--- a/hello.txt
+++ b/hello.txt
@@ -1 +1,2 @@
-git is a powerful tool for vcs
+git is a powerful tool for vcs .
+Github is a pwerful tool for cvs .
```

# G I T   C O M M I T



The screenshot shows a VIM editor window titled "COMMIT\_EDITMSG + (e:\xampp\htdocs\codeigniter\.git) - VIM". The window displays a commit message template with the following content:

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   application/config/dx_auth.php
#       deleted:    application/controllers/auth.php
#       deleted:    application/controllers/backend.php
#       modified:   application/libraries/DX_Auth_Event.php
#       deleted:    application/models/dx_auth/login_attempts.php
#       deleted:    application/models/dx_auth/permissions.php
#       deleted:    application/models/dx_auth/roles.php
#       deleted:    application/models/dx_auth/user_autologin.php
#       deleted:    application/models/dx_auth/user_profile.php
#       deleted:    application/models/dx_auth/user_temp.php
#       deleted:    application/models/dx_auth/users.php
#       deleted:    application/views/modules/dx_auth/auth/cancel_account_form.p
hp
#       deleted:    application/views/modules/dx_auth/auth/change_password_form.
php
@
```

At the bottom of the window, the status bar shows the path "`e:\xampp\htdocs\codeigniter\.git\COMMIT_EDITMSG[+]`", the terminal type "[unix]", the time "(00:46 15/04/2011)", the line number "1,1", and the word "Top". It also displays the command "-- INSERT --".

# O P C I O N E S

Confirma la instantánea preparada

```
git commit
```

Confirma una instantánea de todos los cambios del directorio de trabajo.

```
git commit -a
```

Si se usa la opción **-m**, se omitirá la solicitud de editor de texto a favor de un mensaje insertado.

```
git commit -m "commit message"
```

Esta combinación crea inmediatamente una confirmación de todos los cambios preparados y aplica un mensaje de confirmación insertado.

```
git commit -am "commit message"
```

Al pasar esta opción, se modificará la última confirmación. En vez de crear una nueva confirmación, los cambios preparados se añadirán a la confirmación anterior

```
git commit --amend
```

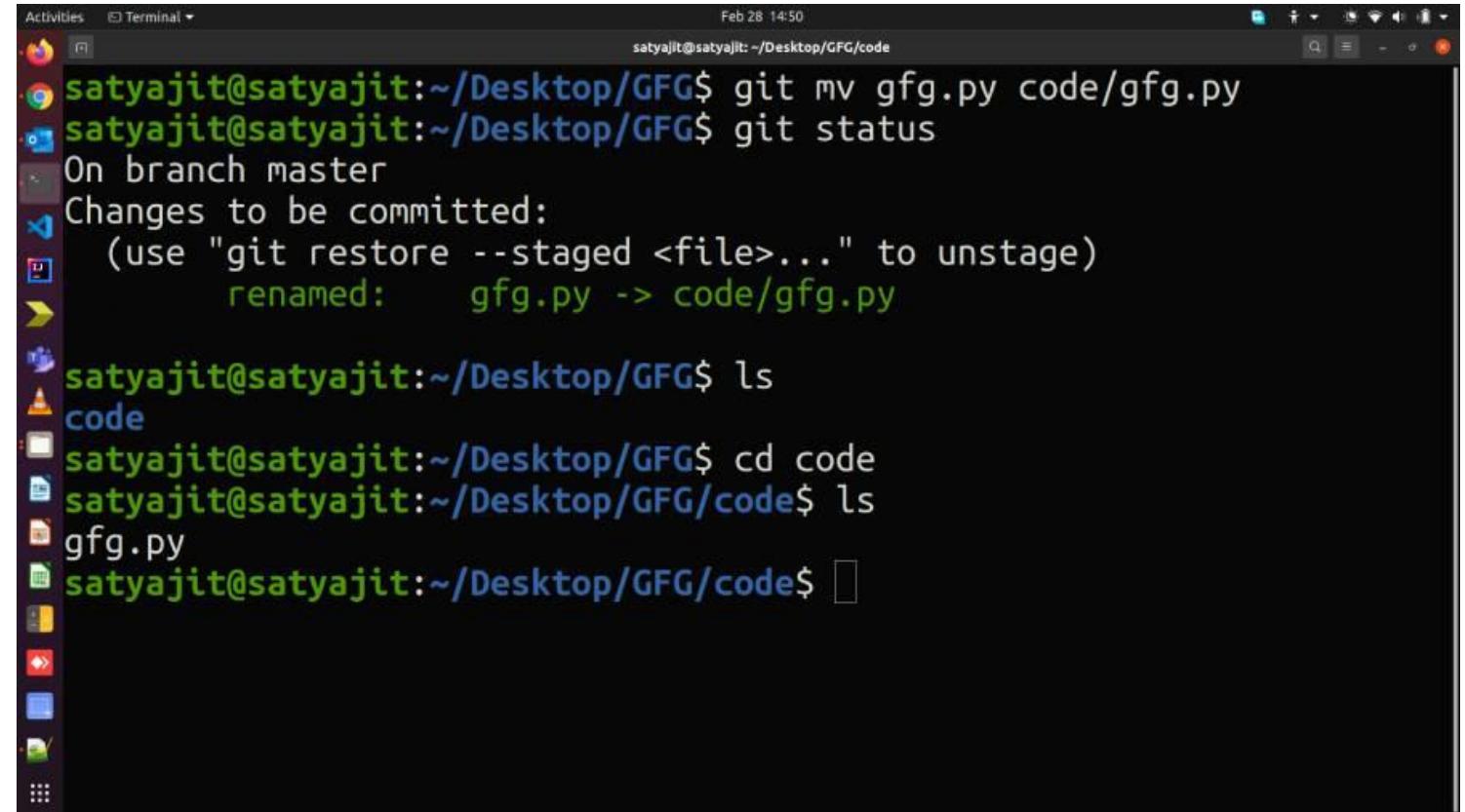
# G I T   R M

```
HIMANSHU@HIMANSHU-PC MINGW64 ~/Desktop/GitExample2 (master)
$ git rm --cached newfile1.txt
rm 'newfile1.txt'

HIMANSHU@HIMANSHU-PC MINGW64 ~/Desktop/GitExample2 (master)
$ git status
on branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    deleted:    newfile1.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    newfile1.txt
```

# G I T   M V

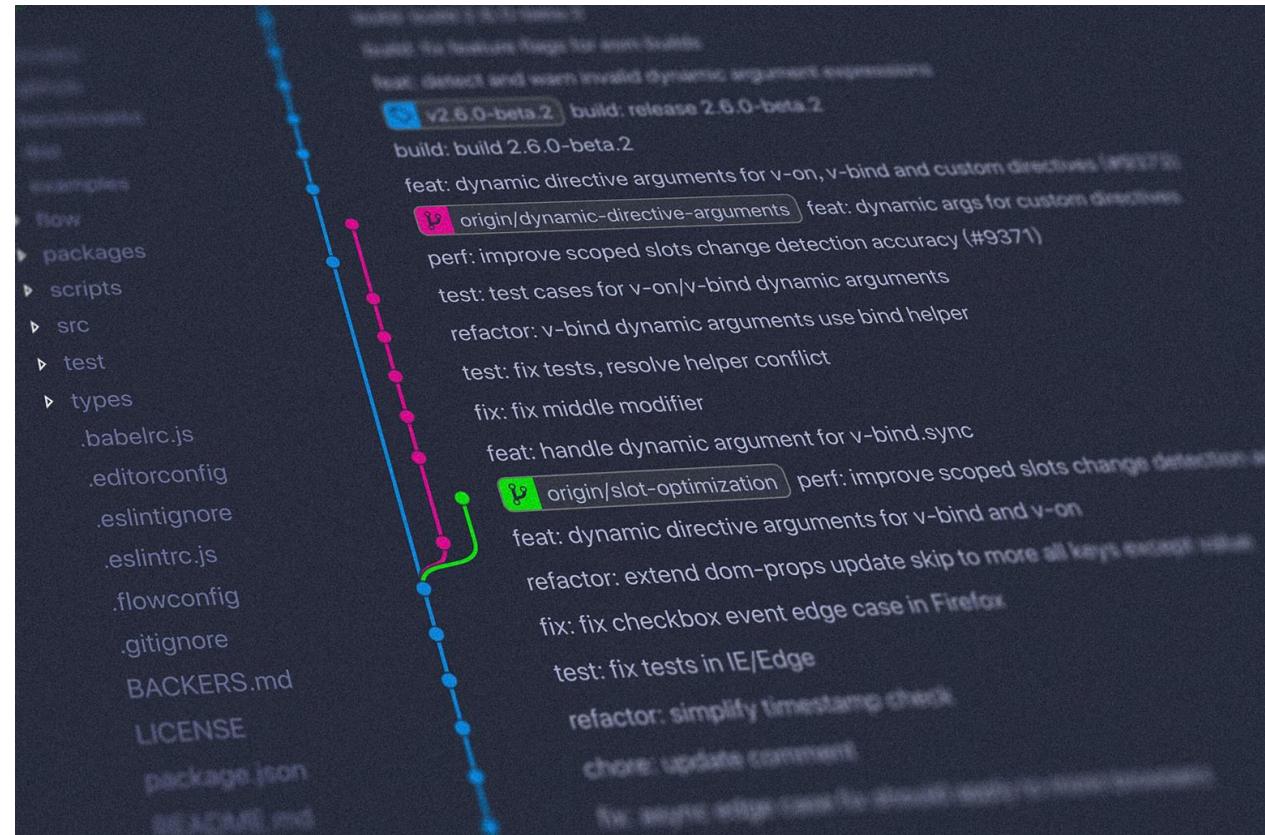


A screenshot of a Linux desktop environment showing a terminal window. The terminal window is titled "Terminal" and has the command line "Activities Terminal". The date and time "Feb 28 14:50" and the user "satyajit@satyajit: ~/Desktop/GFG/code" are displayed at the top right. The terminal content shows the following sequence of commands and output:

```
satyajit@satyajit:~/Desktop/GFG$ git mv gfg.py code/gfg.py
satyajit@satyajit:~/Desktop/GFG$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    renamed:   gfg.py -> code/gfg.py

satyajit@satyajit:~/Desktop/GFG$ ls
code
satyajit@satyajit:~/Desktop/GFG$ cd code
satyajit@satyajit:~/Desktop/GFG/code$ ls
gfg.py
satyajit@satyajit:~/Desktop/GFG/code$ █
```

# LABORATORIO 4



# G I T   L O G

```
$ git log --format=medium
commit 5bf53efc38e1d780b32daf7c1d9acb54a7e93936
Author: Will Anderson <will@itsananderson.com>
Date:   Mon Jul 21 08:27:50 2014 -0700

    Normalize indents to 4 spaces for all source files

commit bb82a7ab75f14cce5d8f8a60e97bfa8a2ceabbe
Author: Will Anderson <will@itsananderson.com>
Date:   Mon Jul 21 01:11:38 2014 -0700

    Bump version to 0.2.4

commit d9bb2ff54751797208c0e551004cab1c1adb01cc
Author: Will Anderson <will@itsananderson.com>
Date:   Mon Jul 21 01:11:18 2014 -0700

    Update .npmignore with new dev files

commit c75067e0f76c0e0adad58bb1ac663834f572f696
Author: Will Anderson <will@itsananderson.com>
Date:   Sun Jul 20 23:00:44 2014 -0700

    Add full coverage for application mixin

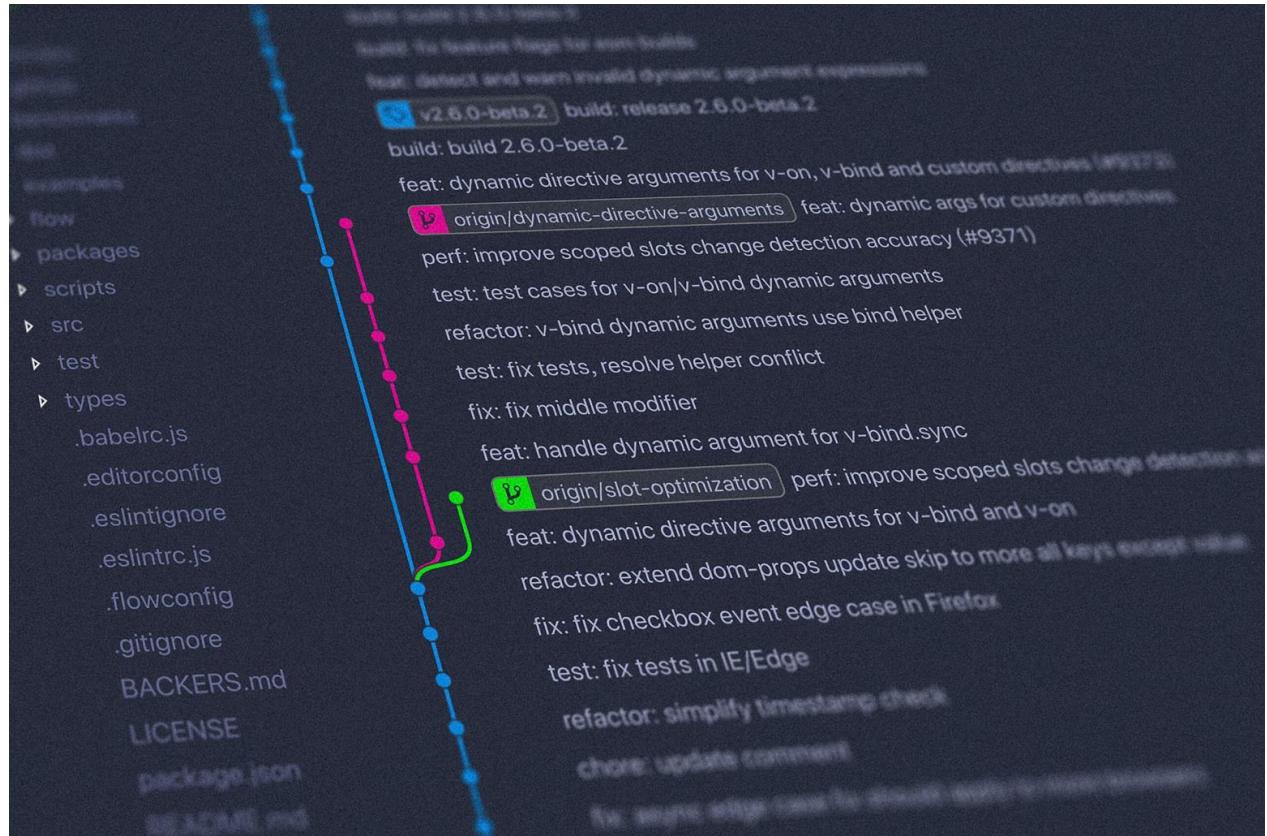
commit 98ee9cbbb4454912d352b8bdb6a3dcbcd64a38f9
Author: Will Anderson <will@itsananderson.com>
Date:   Sun Jul 20 14:34:18 2014 -0700

    Add full coverage for routeParams utility
```

# OPCIONES

Opción	Descripción de la salida
<code>%H</code>	Hash de la confirmación
<code>%h</code>	Hash de la confirmación abreviado
<code>%T</code>	Hash del árbol
<code>%t</code>	Hash del árbol abreviado
<code>%P</code>	Hashes de las confirmaciones padre
<code>%p</code>	Hashes de las confirmaciones padre abreviados
<code>%an</code>	Nombre del autor
<code>%ae</code>	Dirección de correo del autor
<code>%ad</code>	Fecha de autoría (el formato respeta la opción - <code>date</code> )
<code>%ar</code>	Fecha de autoría, relativa
<code>%cn</code>	Nombre del confirmador
<code>%ce</code>	Dirección de correo del confirmador
<code>%cd</code>	Fecha de confirmación
<code>%cr</code>	Fecha de confirmación, relativa
<code>%s</code>	Asunto

# LABORATORIO 5



# G I T   R E S E T

```
PC@PC-PC MINGW64 ~/Desktop/ghj/decodingdevops (master)
$ git status
On branch master
Your branch is based on 'origin/master', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   abc.txt

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

    decoding.txt

PC@PC-PC MINGW64 ~/Desktop/ghj/decodingdevops (master)
$
```

# G I T C H E C K O U T



A screenshot of a terminal window titled "MINGW64:/d/GeeksForGeeks-Branching and merging". The window shows a series of git commands being run:

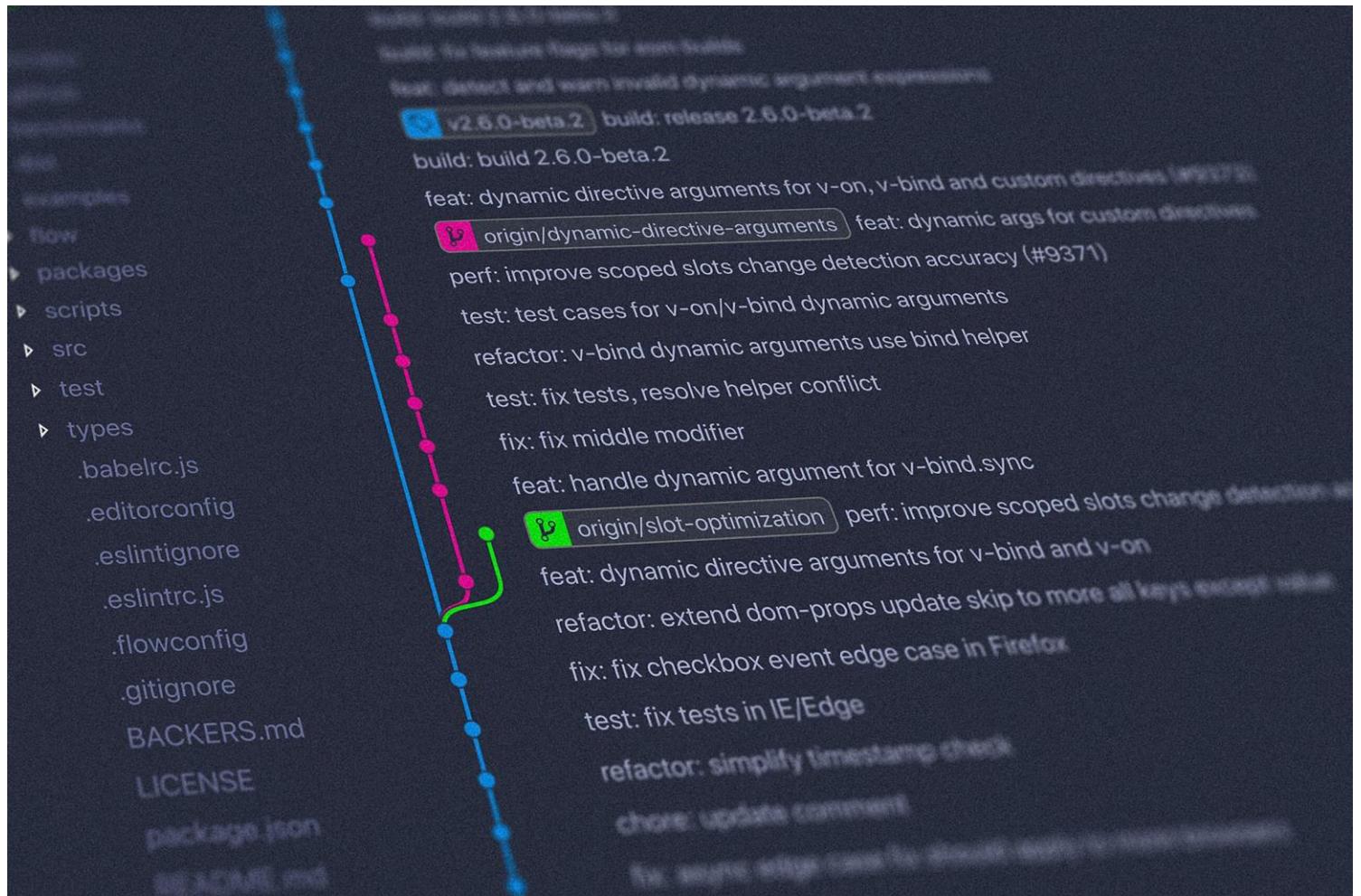
```
Asus@Asus-PC MINGW64 /d/GeeksForGeeks-Branching and merging (master)
$ git branch
* master
  next-five-even-odd

Asus@Asus-PC MINGW64 /d/GeeksForGeeks-Branching and merging (master)
$ git checkout next-five-even-odd
Switched to branch 'next-five-even-odd'

Asus@Asus-PC MINGW64 /d/GeeksForGeeks-Branching and merging (next-five-even-odd)
$ git branch
  master
* next-five-even-odd

Asus@Asus-PC MINGW64 /d/GeeksForGeeks-Branching and merging (next-five-even-odd)
$ :
```

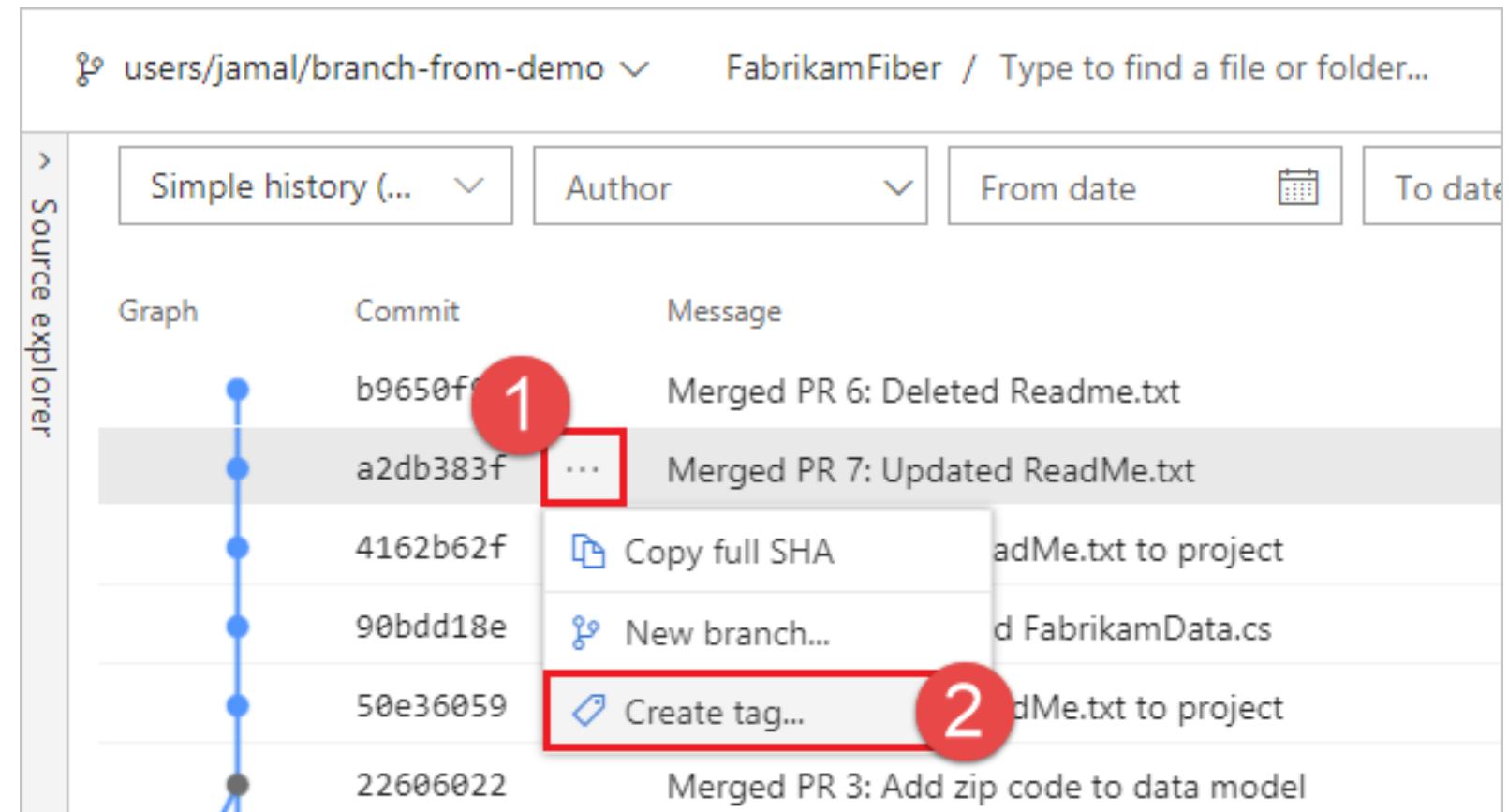
# LABORATORIO 6



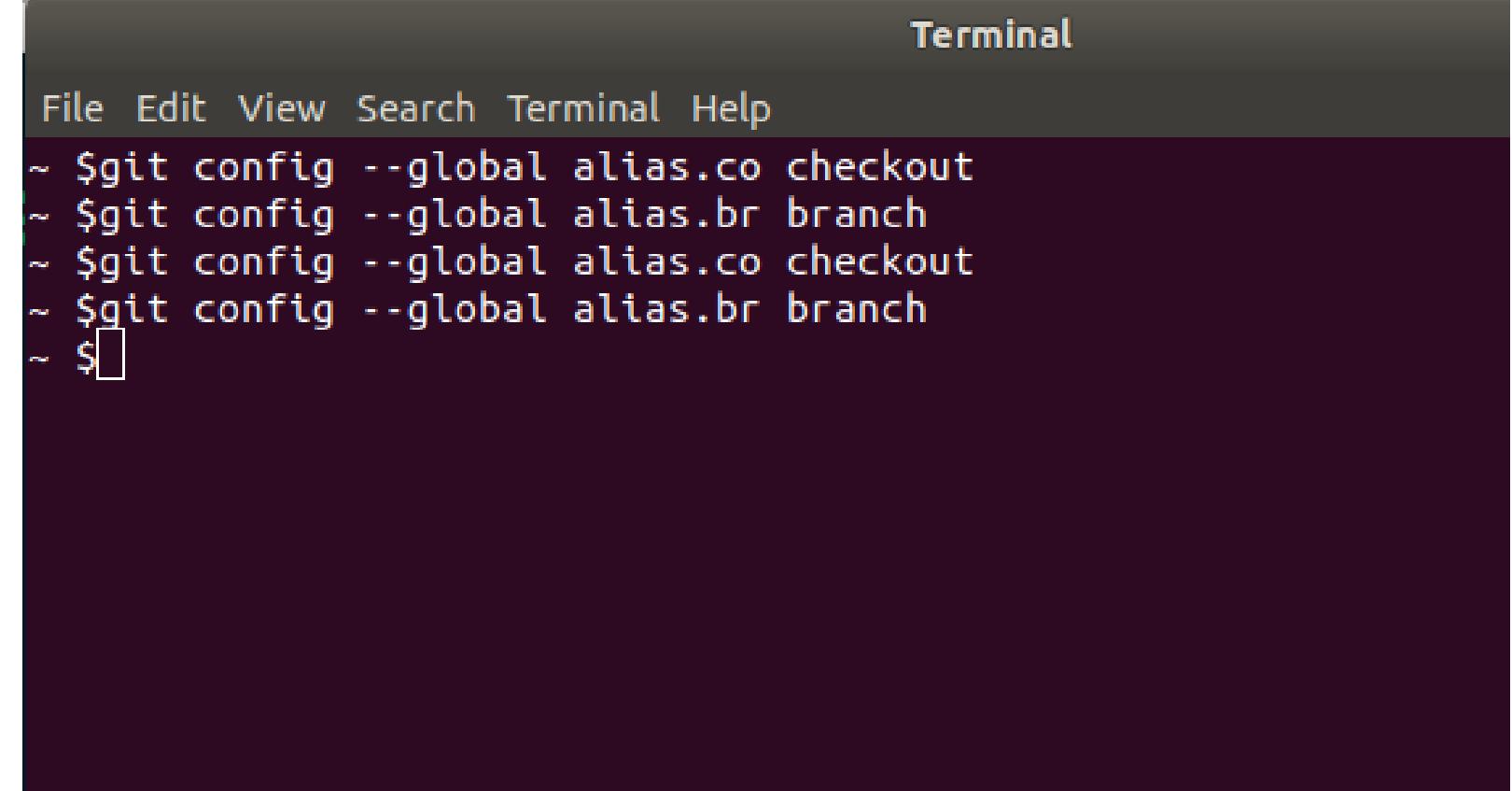
# G I T   T A G



# ETIQUETA ANOTADA Y LIGERA



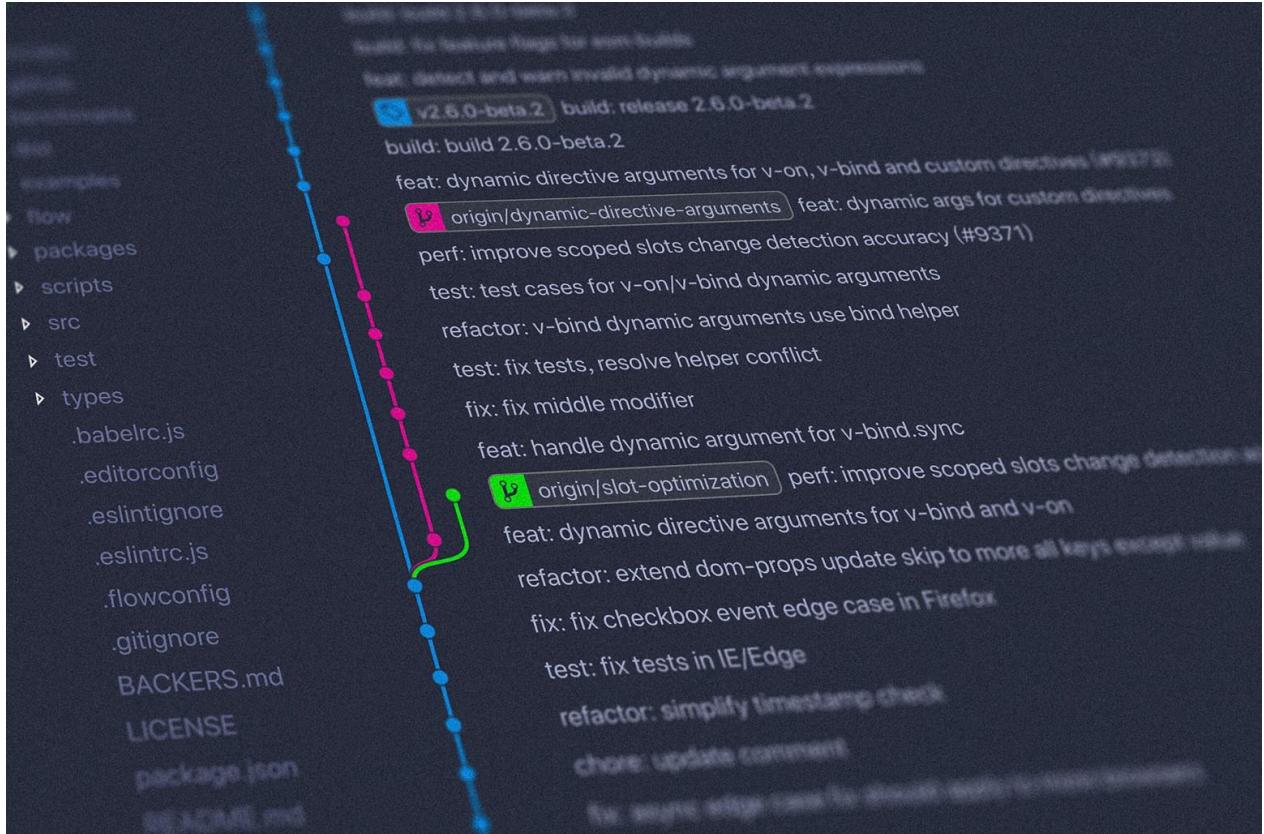
# ALIAS



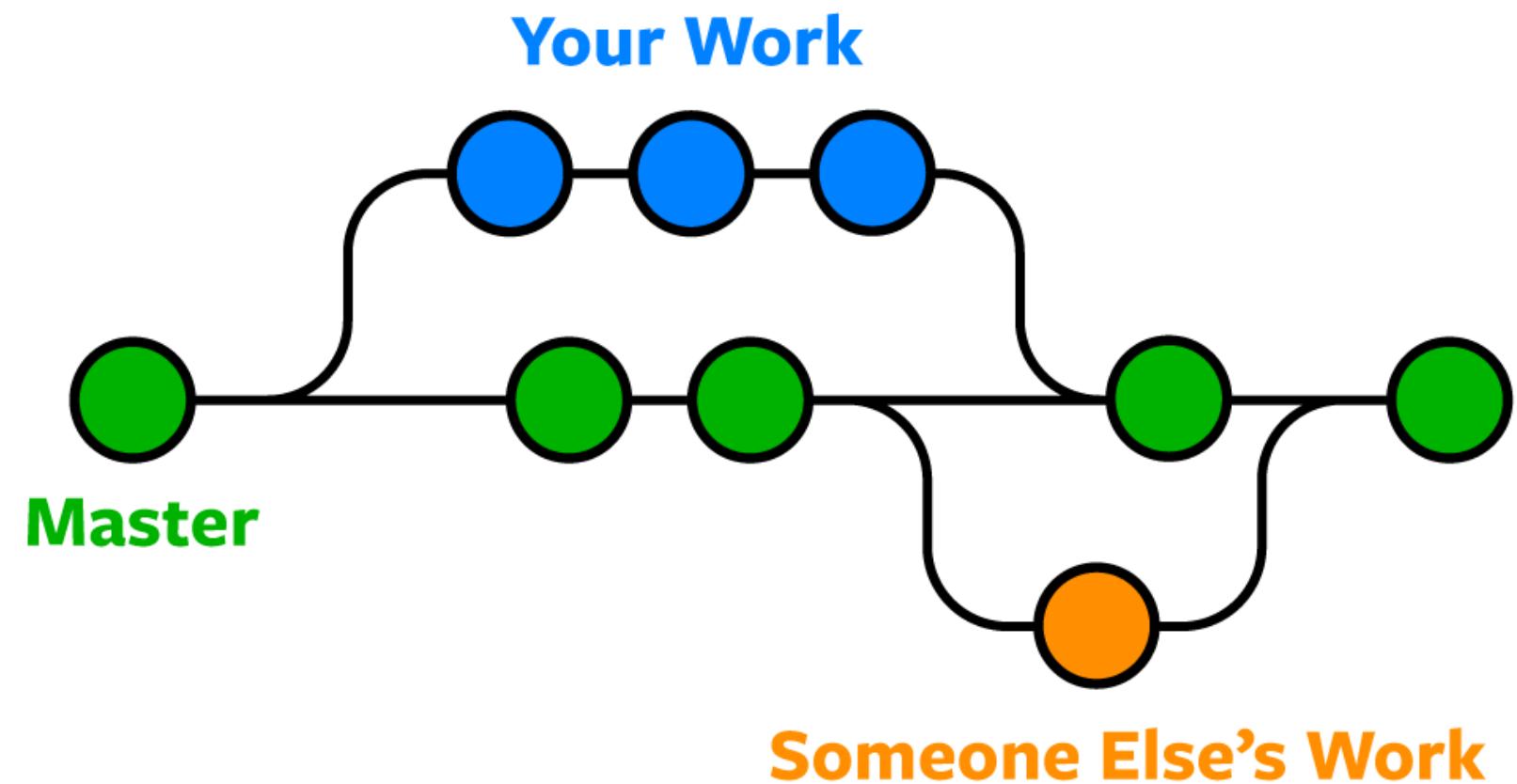
A screenshot of a terminal window titled "Terminal". The window has a dark background and a light-colored menu bar at the top. The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". Below the menu bar, there is a list of command-line entries. The entries are:

```
~ $git config --global alias.co checkout
~ $git config --global alias.br branch
~ $git config --global alias.co checkout
~ $git config --global alias.br branch
~ $
```

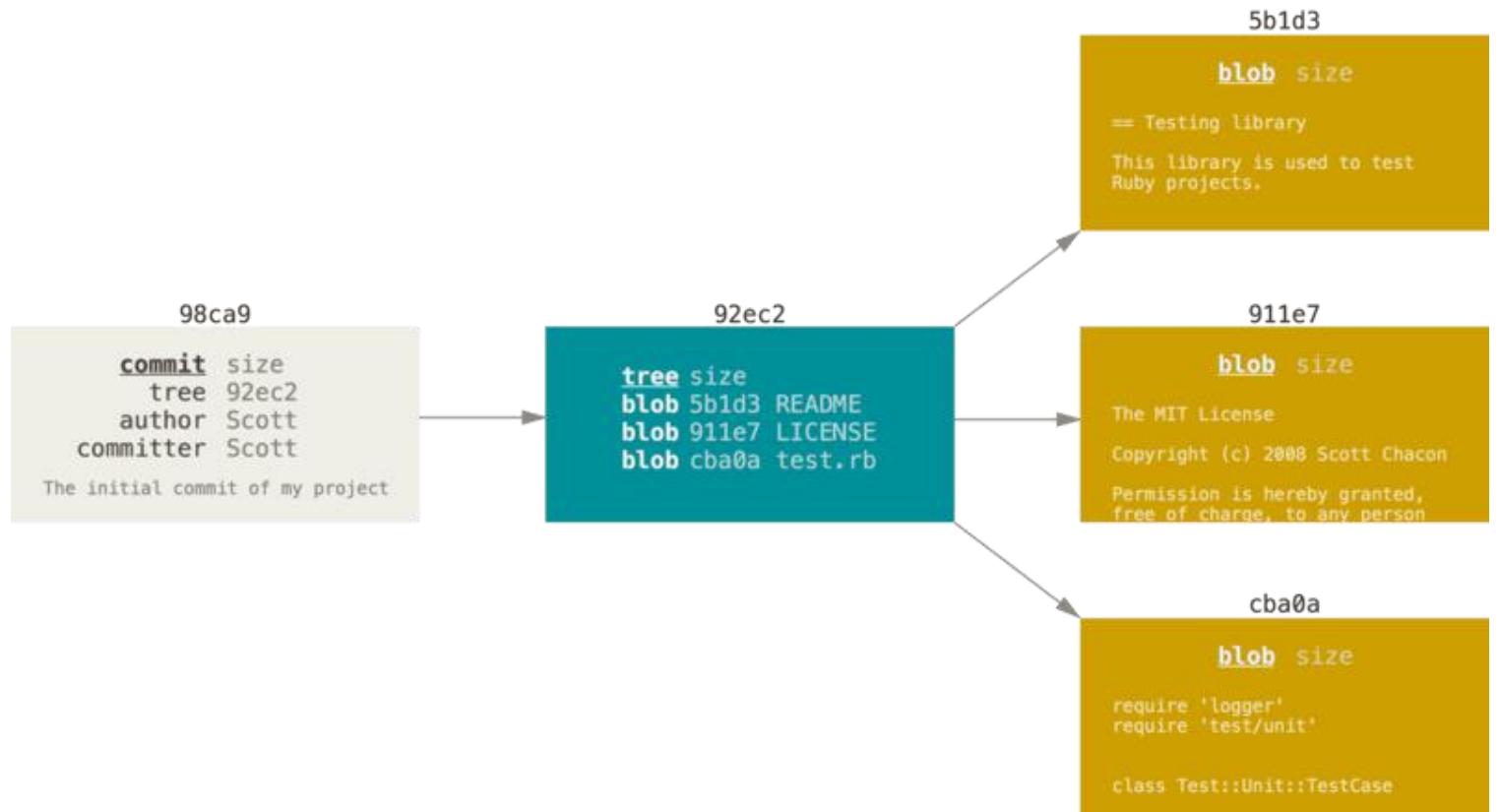
# EJERCICIO 1



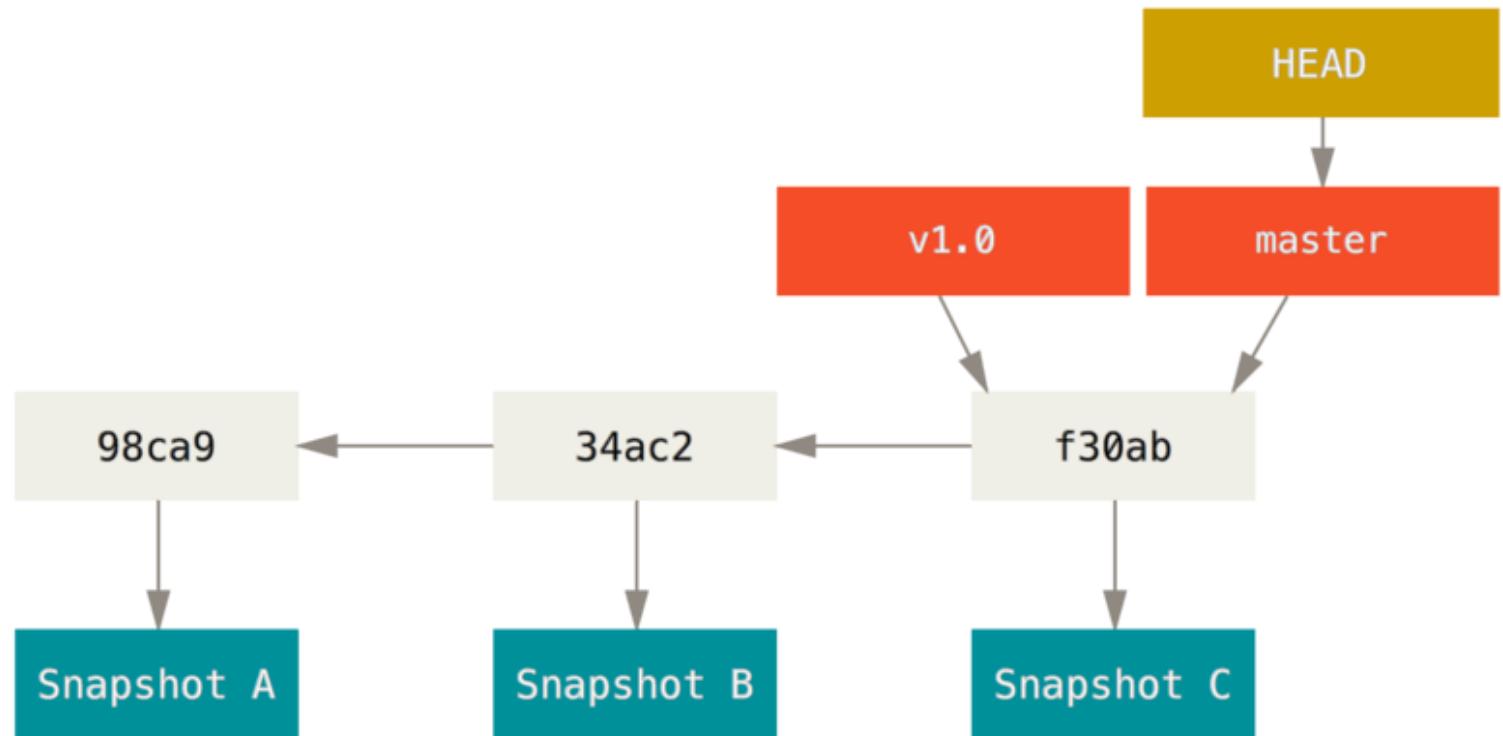
# RAMIFICACIONES



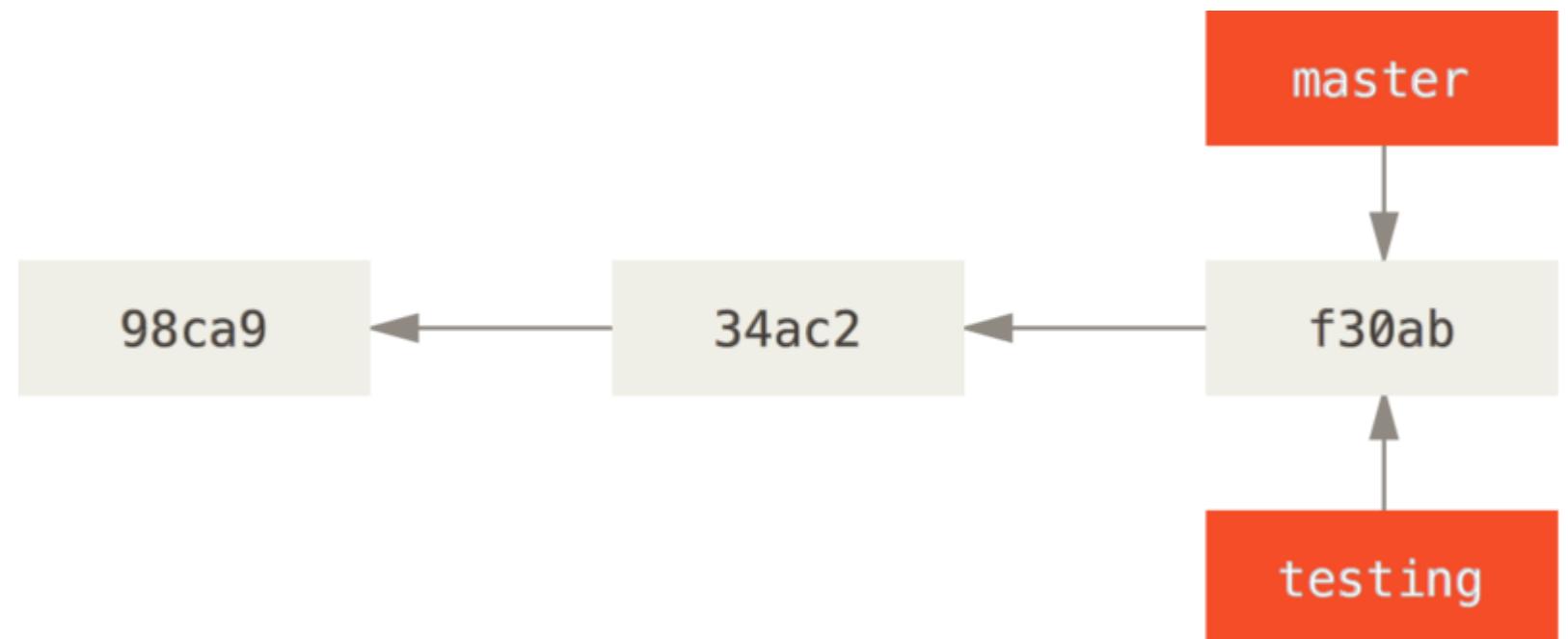
# ¿QUE ES UNA RAMA?



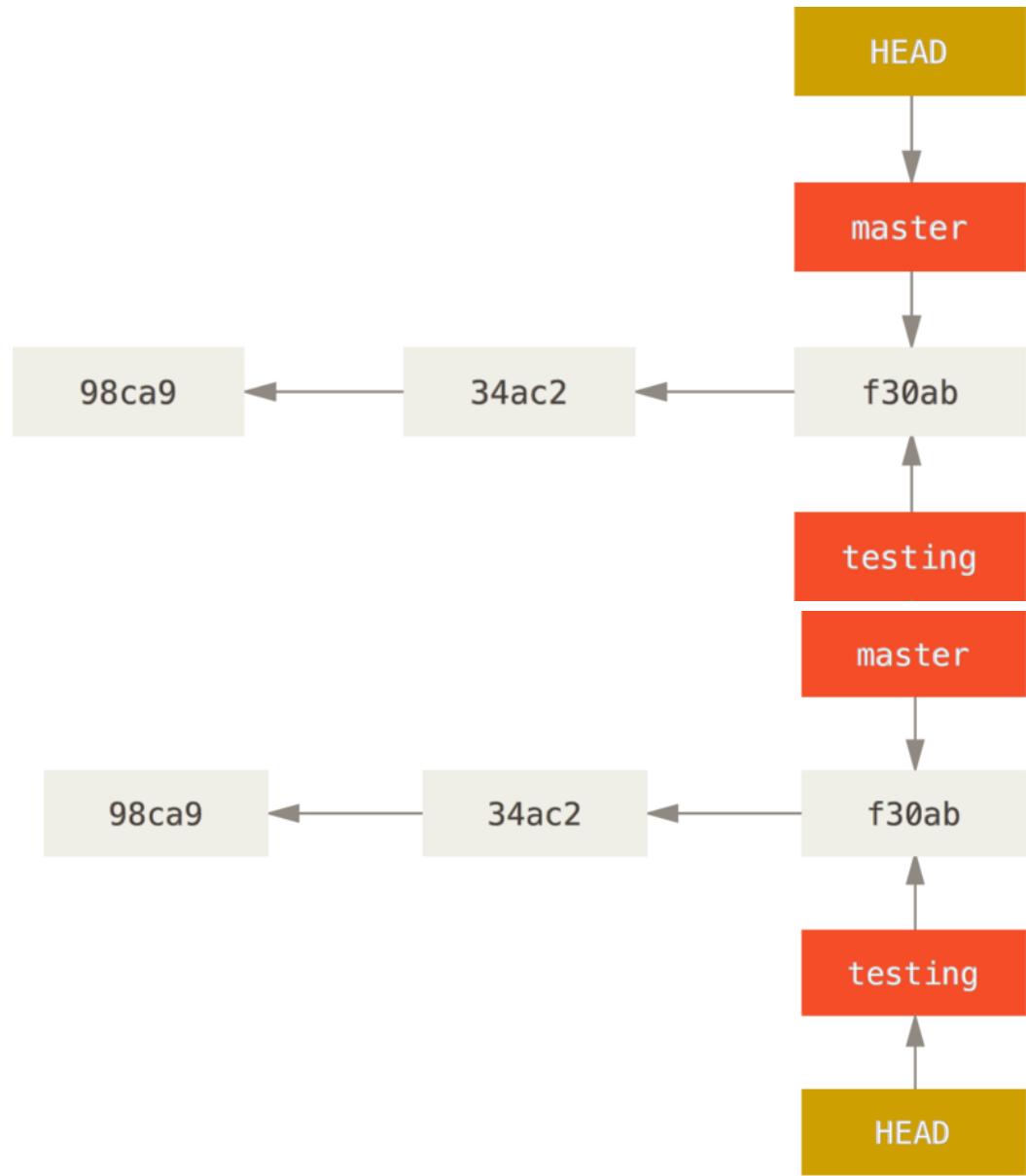
# A P U N T A D O R M O V I L



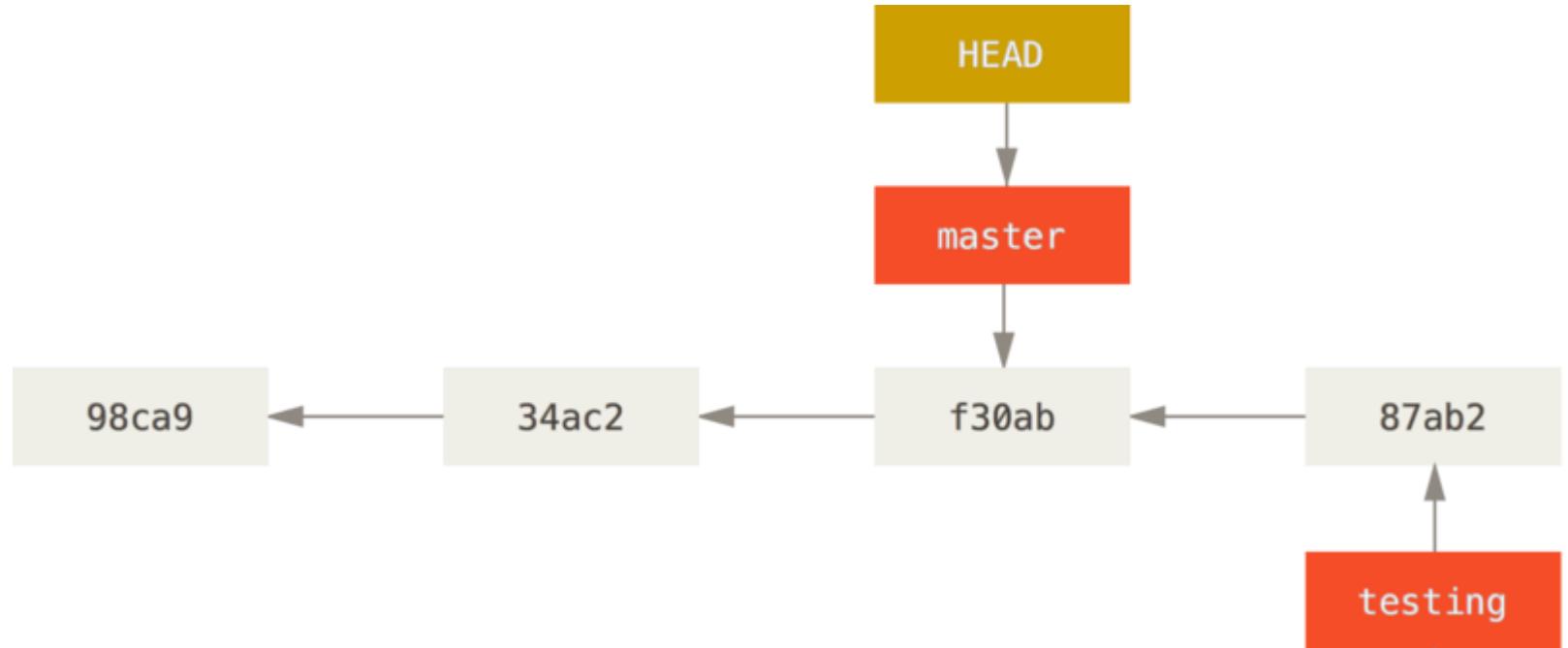
# NUEVAS RAMAS



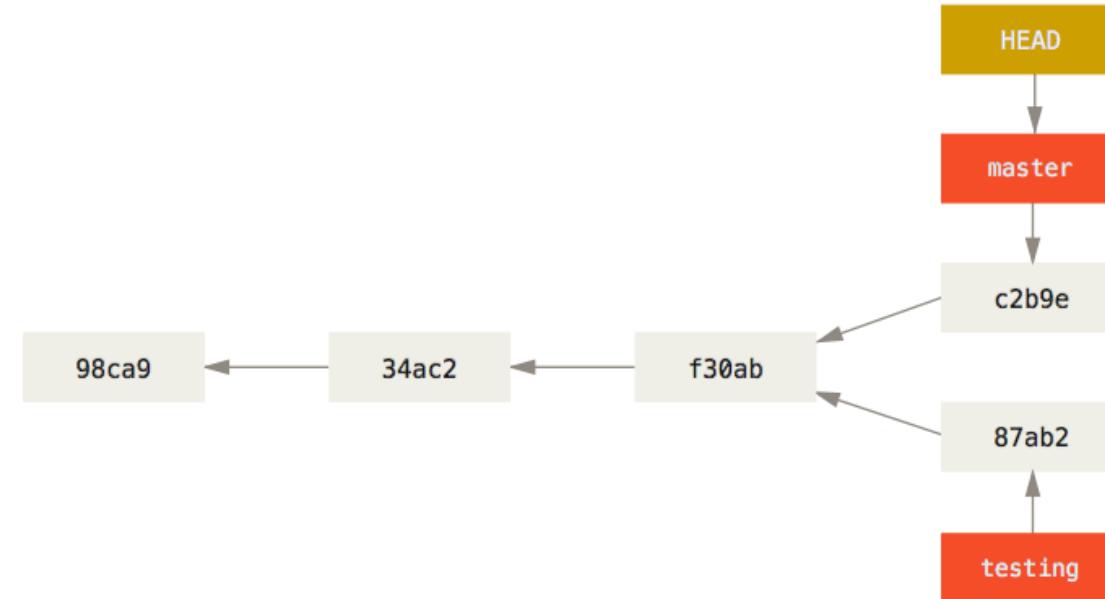
# MOVERNOS DE UNA RAMA A OTRA



# MOVERNOS DE UNA RAMA A OTRA



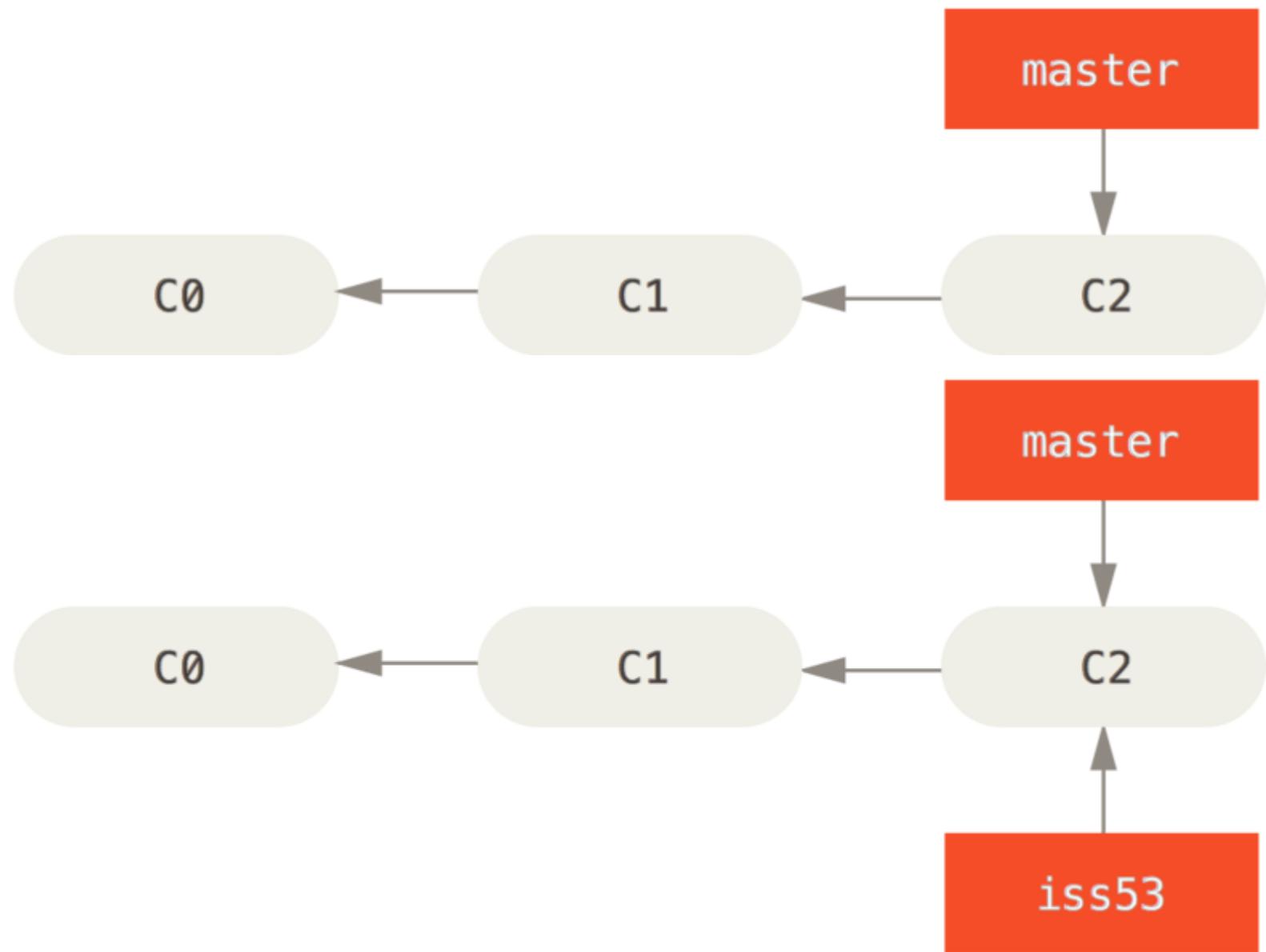
# ¿QUE NOS QUEDA FINALMENTE ?



## E J E M P L O

- 1. Trabajas en un sitio web.
- 2. Creas una rama para un nuevo tema sobre el que quieres trabajar.
- 3. Realizas algo de trabajo en esa rama.
- - En este momento, recibes una llamada avisándote de un problema crítico que has de resolver. Y sigues los siguientes pasos:
  - 1. Vuelves a la rama de producción original.
  - 2. Creas una nueva rama para el problema crítico y lo resuelves trabajando en ella.
  - 3. Tras las pertinentes pruebas, fusionas (merge) esa rama y la envías (push) a la rama de producción.
  - 4. Vuelves a la rama del tema en que andabas antes de la llamada y continuas tu trabajo

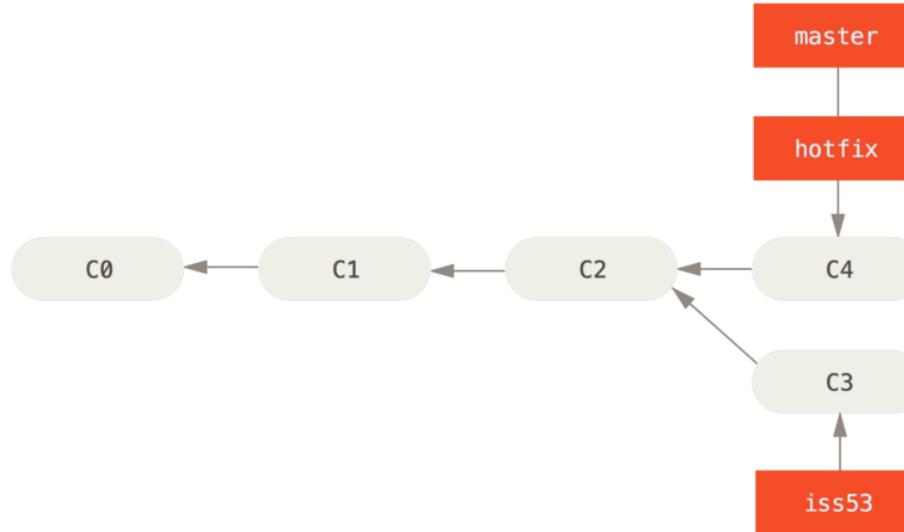
## PASO 1



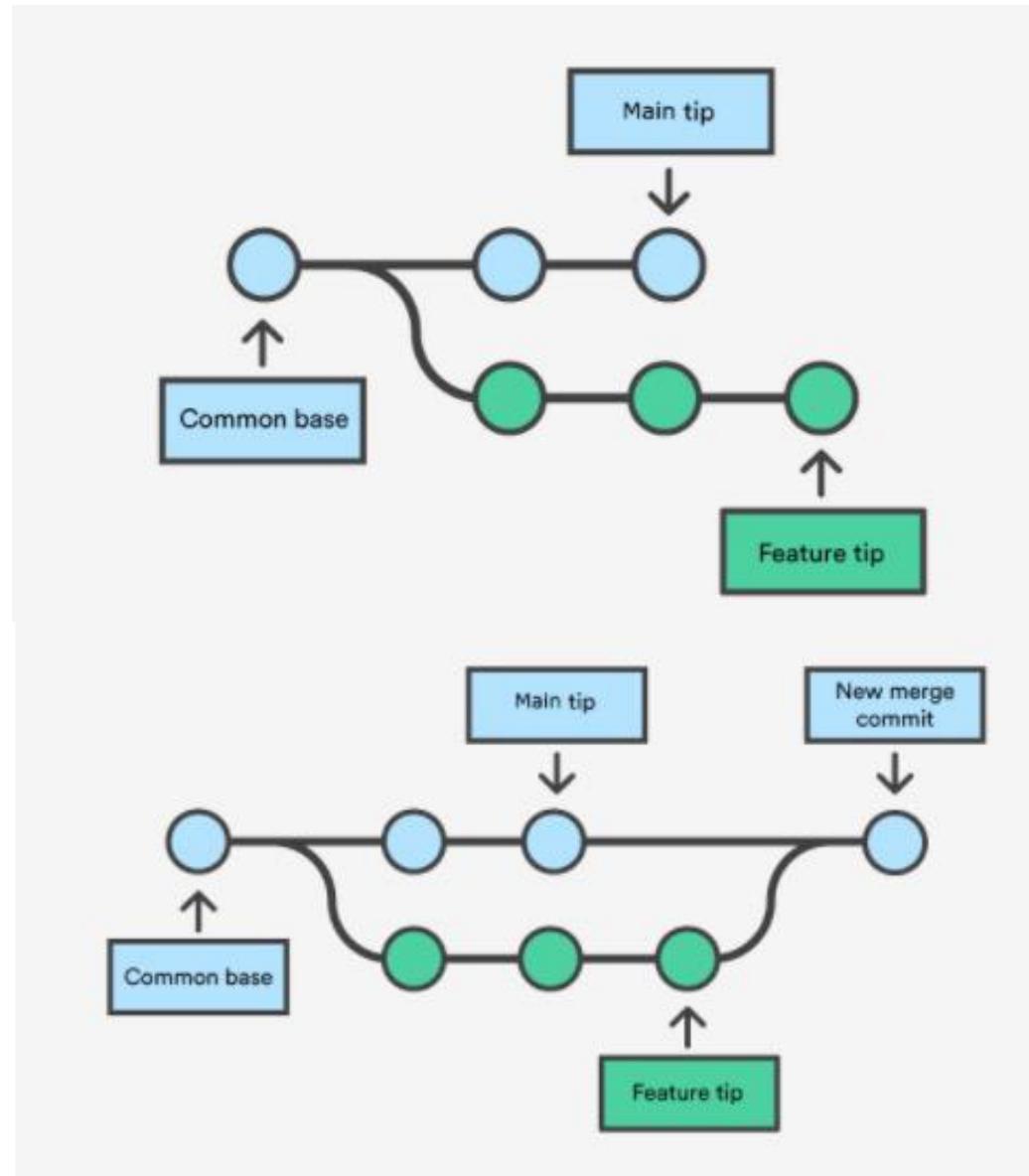
# RECIBES LA LLAMADA

```
$ git checkout master  
Switched to branch 'master'
```

```
$ git checkout -b hotfix  
Switched to a new branch 'hotfix'  
$ vim index.html  
$ git commit -a -m 'fixed the broken email address'  
[hotfix 1fb7853] fixed the broken email address  
1 file changed, 2 insertions(+)
```



# MERGE



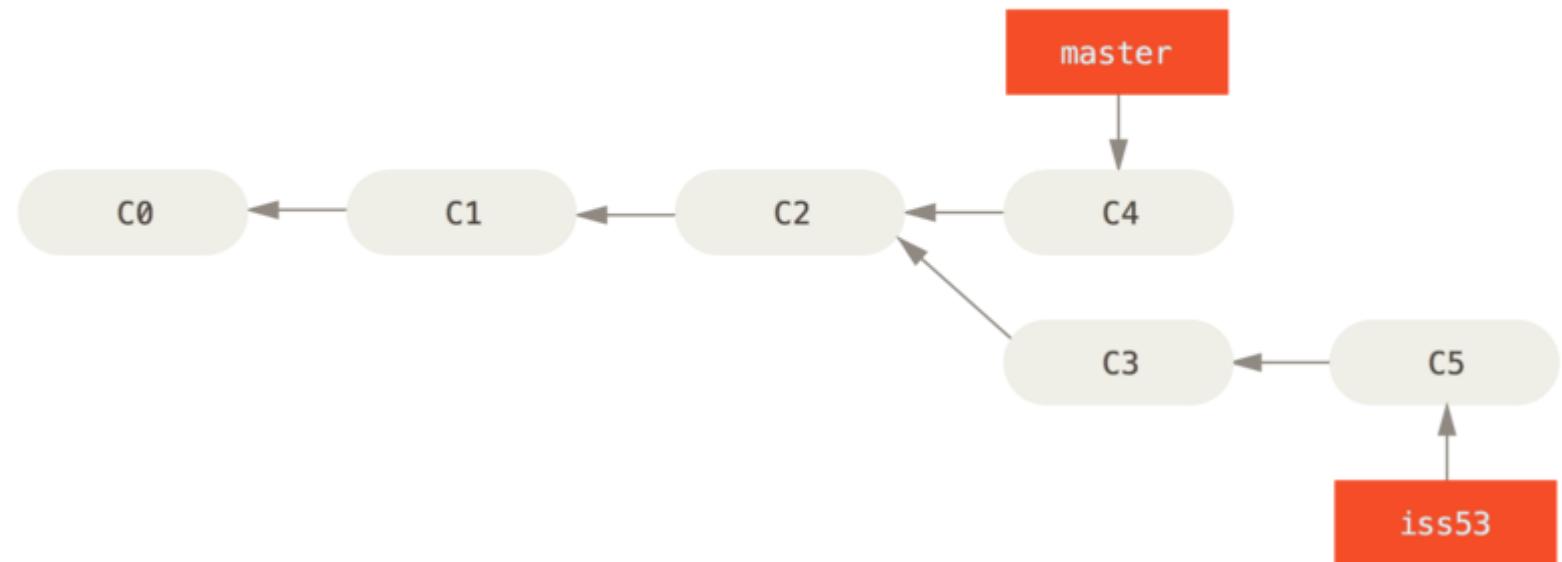
# SOLUCIONAMOS EL PROBLEMA URGENTE

```
$ git checkout master
$ git merge hotfix
Updating f42c576..3a0874c
Fast-forward
 index.html | 2 ++
 1 file changed, 2 insertions(+)
```

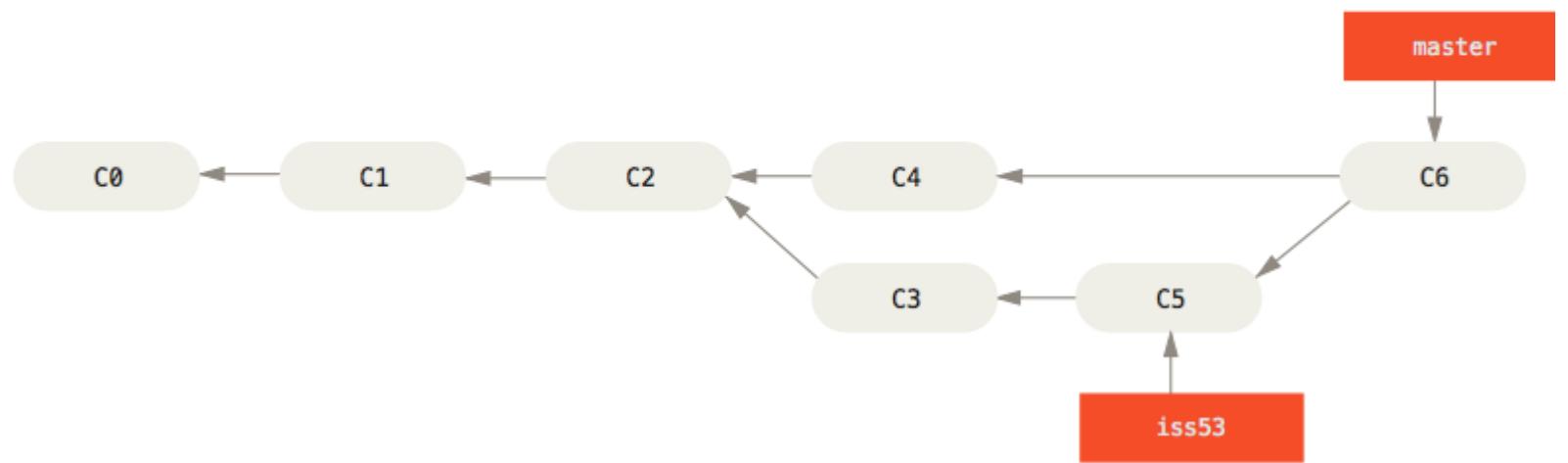
```
$ git branch -d hotfix
Deleted branch hotfix (3a0874c).
```

```
$ git checkout iss53
Switched to branch "iss53"
$ vim index.html
$ git commit -a -m 'finished the new footer [issue 53]'
[iss53 ad82d7a] finished the new footer [issue 53]
 1 file changed, 1 insertion(+)
```

# SITUACIÓN ACTUAL



# RESOLVIENDO EL PROBLEMA INICIAL



# DEMO

v2.6.0-beta.2 build: release 2.6.0-beta.2

build: build 2.6.0-beta.2

feat: dynamic directive arguments for v-on, v-bind and custom directives (https://github.com/vuejs/vue/pull/9370) | feat: dynamic args for custom directives (https://github.com/vuejs/vue/pull/9371)

perf: improve scoped slots change detection accuracy (#9371)

test: test cases for v-on/v-bind dynamic arguments

refactor: v-bind dynamic arguments use bind helper

test: fix tests, resolve helper conflict

fix: fix middle modifier

feat: handle dynamic argument for v-bind.sync

origin/slot-optimization | perf: improve scoped slots change detection in IE/Edge

feat: dynamic directive arguments for v-bind and v-on

refactor: extend dom-props update skip to more all keys except value

fix: fix checkbox event edge case in Firefox

test: fix tests in IE/Edge

refactor: simplify timestamp check

chore: update comment

fix: move merge pull to start of history

... (truncated)

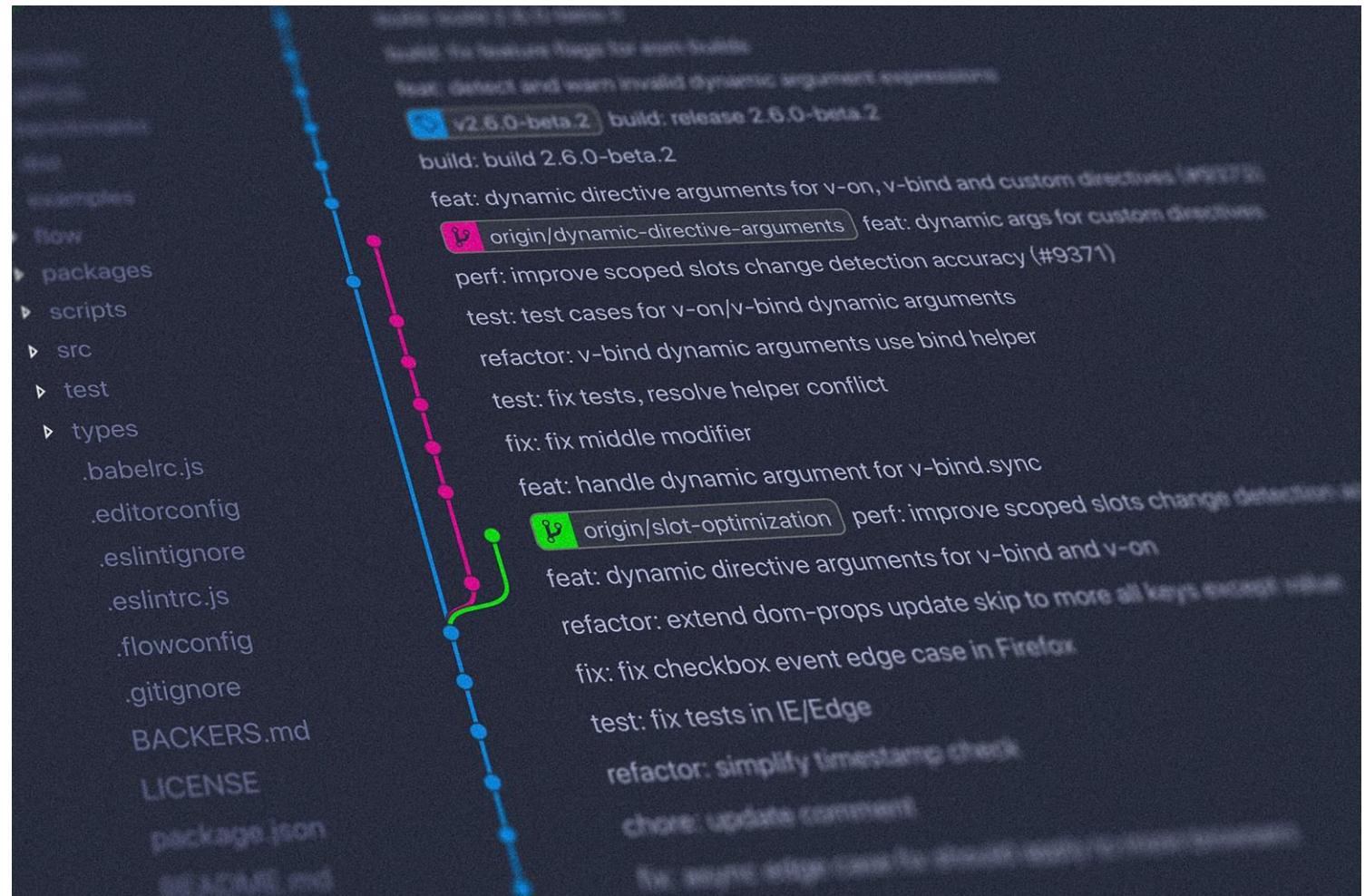
BACKERS.md

LICENSE

package.json

README.md

## E J E R C I C I O 2



# CONFLICTOS

```
killmer@hp MINGW64 ~/Desktop/pruebagit (prod)
$ git merge home
Auto-merging home.html
CONFLICT (content): Merge conflict in home.html
Automatic merge failed; fix conflicts and then commit the result.
```

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes  
|||||| HEAD (Current Change)  
holo  
=====

que???

>>>>> home (Incoming Change)

|

# G I T   B R A N C H

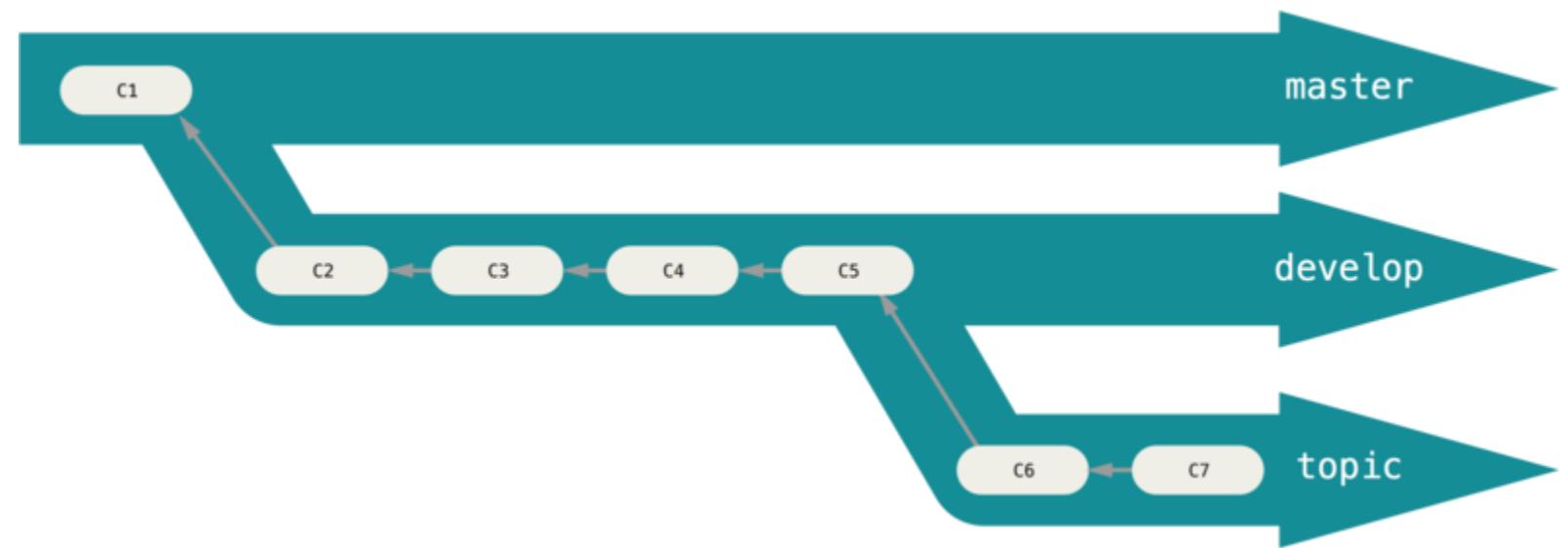
```
$ git branch  
iss53  
* master  
testing
```

```
$ git branch --merged  
iss53  
* master
```

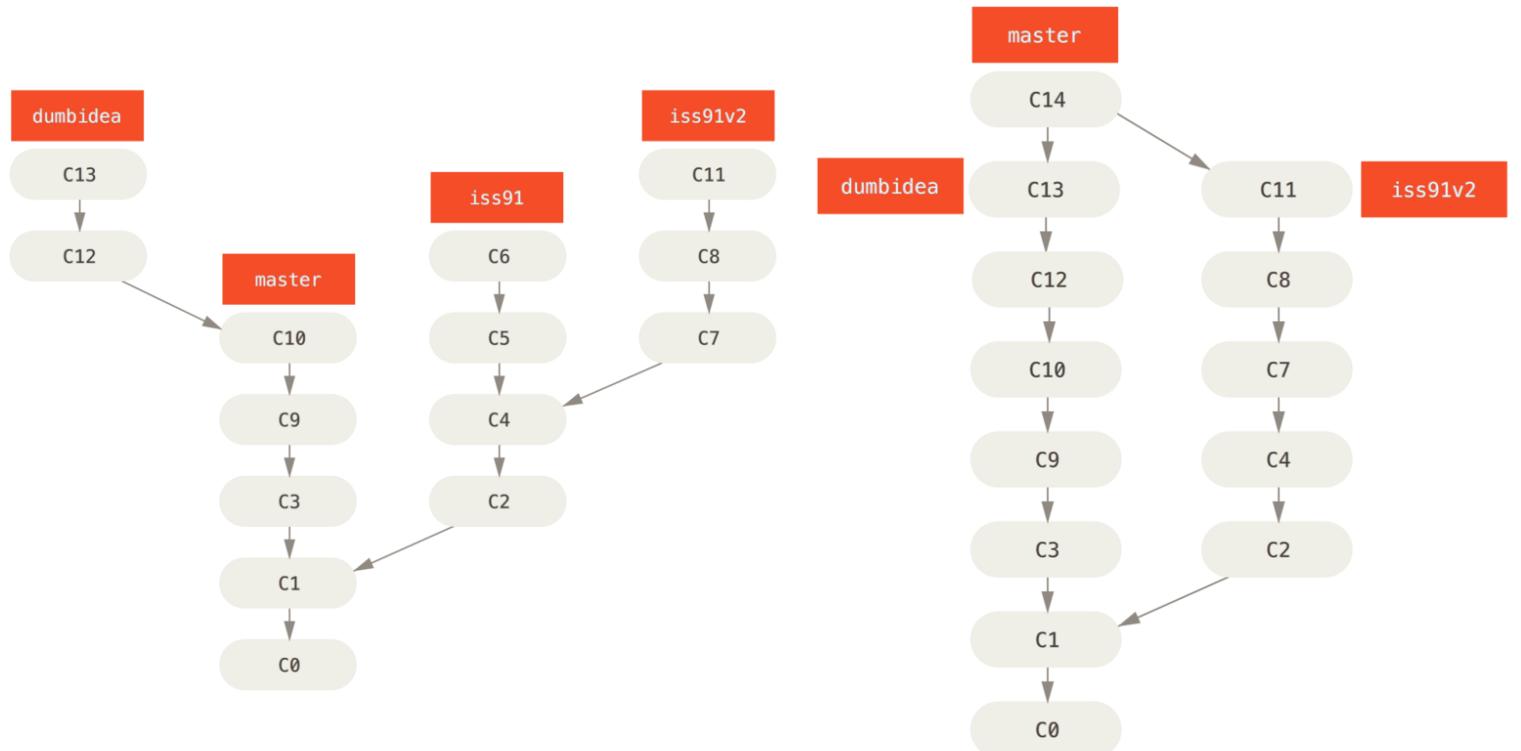
```
$ git branch -v  
iss53 93b412c fix javascript issue  
* master 7a98805 Merge branch 'iss53'  
testing 782fd34 add scott to the author list in the readmes
```

```
$ git branch -d testing  
error: The branch 'testing' is not fully merged.  
If you are sure you want to delete it, run 'git branch -D testing'.
```

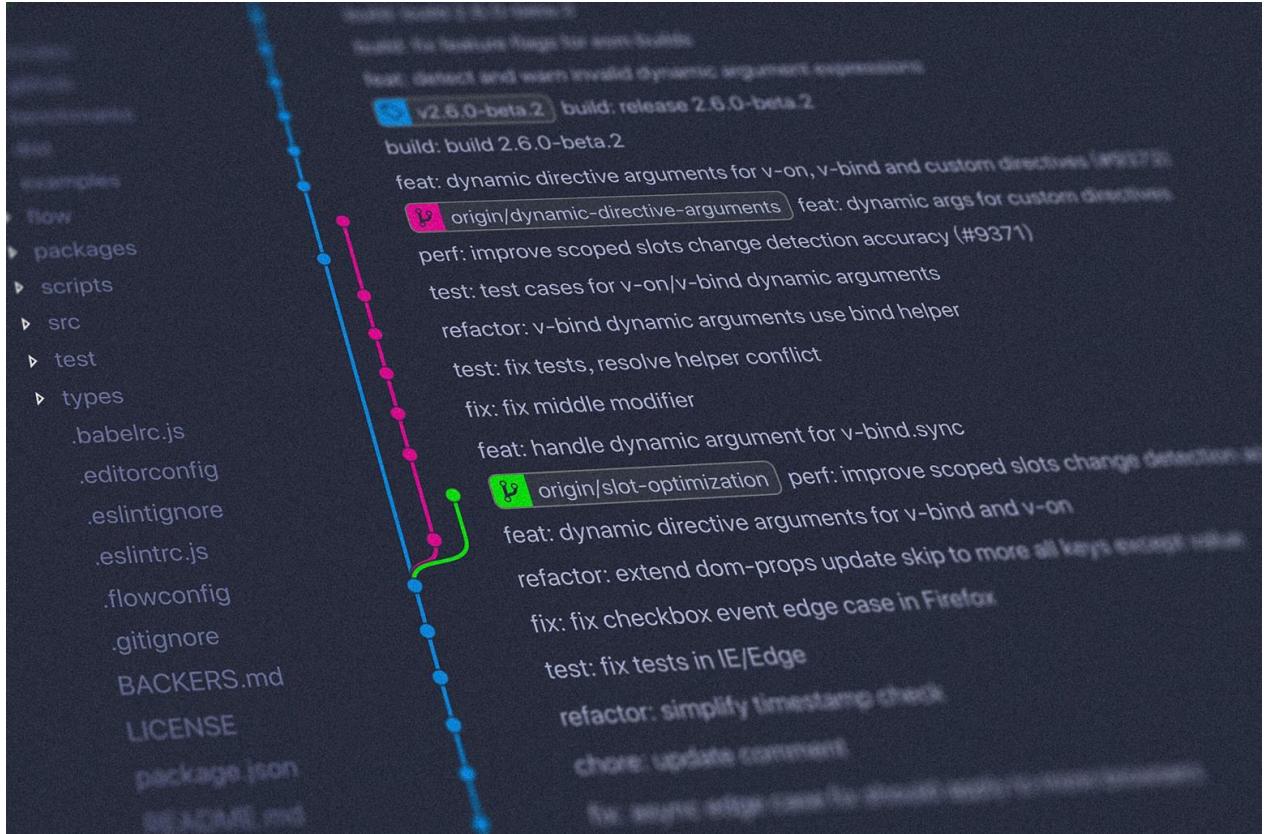
# RAMAS DE LARGO RECORRIDO



# RAMAS PUNTUALES



# EJERCICIO 3





REMOTOS













