

GUÍAS DE ESTILO PARA GITHUB

Edilberto Alonso Taba A01283973
José Andrés Lozano A01284569
Guillermo Rodríguez Juárez A01067172
Rodrigo Rodriguez de Luna A01384318
Guillermo Emmanuel Ayala A01197833
Alan de la Cruz A01280638

```
$ git init
Initialized empty Git repository in /tmp/tmp.IMBYSY7R8Y/.git/
$ cat > README << 'EOF'
> Git is a distributed revision control system.
> EOF
$ git add README
$ git commit
[master (root-commit) e4dcc69] You can edit locally and push
to any remote.
1 file changed, 1 insertion(+)
create mode 100644 README
$ git remote add origin git@github.com:cdown/thats.git
$ git push -u origin master
```

MAKE SMALL COMMITS

Es recomendable hacer commits a pequeñas partes de un código, para así tener un orden de registros que este organizado, esto hará que los desarrolladores puedan leer con mayor facilidad y mantener el código base.

COMMIT COMPLETE AND WELL TESTED CODE

Para lograr tener un código bien probado es necesario no hacer commit de un código incompleto. Cuando estés trabajando con un trabajo muy grande trata de descomponerlo en secciones mas pequeñas y asegúrate de completarlas. Por Ultimo trata siempre de probar tu código antes de hacer commit.

```
makat@LAPTOP-HNN157C5 MINGW64 ~/Desktop/GitExample2 (newbranch)
$ git log
commit e43483b4cddedd1b28ac358857ca75d23376e1f9 (HEAD -> newbranch)
Author: ImDwivedi1 <himanshudubey481@gmail.com>
Date: Mon Sep 16 12:21:55 2019 +0530

    commit for the project's main branch

commit 4f7cc7458f79f8c21fa3563dbecce03381dc9645
Author: ImDwivedi1 <himanshudubey481@gmail.com>
Date: Mon Sep 16 12:02:23 2019 +0530

    new changes proposes for master branch

commit 4a6693a71151323623c04dd75cb0d44c1c4dbadf (origin/master, origin/HEAD)
Merge: 30193f3 78c5fbd
Author: ImDwivedi1 <52317024@users.noreply.github.com>
Date: Mon Sep 9 15:24:13 2019 +0530

    Merge pull request #1 from ImDwivedi1/branch2

    Create merge the branch
```

```
~/Development/serious-project master ✓
$ git commit -m "Add codes for the purpose of the project"
write-good:git: Commit message warnings!

Add codes for the purpose of the project
AAAAAAAAAAAAAAAAAAAA
*for the purpose of* is wordy or unneeded on line 1 at column 10

Duster (root-commit) dfa1b5f Add codes for the purpose of the project
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 codes

~/Development/serious-project master ✓
$
```

WRITE GOOD COMMIT MESSAGES

Es bueno describir el proceso que realizaste en el commit para que recuerdes lo que se realizó. El mensaje en cada commit debe ser corto, en presente y explicar porqué se realizó el cambio.

USE THE IMPERATIVE STYLE IN THE SUBJECT LINE

En el asunto referente a un commit se tiende a mandar un mensaje que no siempre puede ser útil, como lo es decir se arreglo un bug o cambie algunas cosas. Es mejor utilizar mensajes donde se expresen ordenes claras.

- Imperative programs
 - Description of *WHAT* is to be computed is *inter-twined* with *HOW* it is to be computed
 - The latter involves organization of data and the sequencing of instructions.

```
11410523623c04dd75cb0d44c1c4dbadf /d/google Drive/web/maapserver64/www/desarro/1oun
advirtual-php?com (modificaciones)
$ git log --oneline
97a77d33 (HEAD -> modificaciones, origin/modificaciones) resolviendo conflicto
on vendor merge branch modificaciones #d https://github.com/alexustor/desarro
10unidadvirtual-php?com into modificaciones
e67214c1 Trabajando casa de padres traendo dela.com de huanboy carpeta vendor
e672159d iconos en svg
e47931d6 poner un comentario
e32c6f81 arreglando iconos poniendolo a svg carrito e compras
97981726 vendor movido a raíz
ec040685 Agregao composer
e680dec6 instalando composer
e689f5d0 (origin/modificaspadre, origin/main, origin/HEAD, modicasapadre, main)
hemos creado un archivo
e680dec6 La primera copia seguridad de la base de datos
e7812390 hemos eliminado un archivo
e4067c51 iniciamos subiendo todos los archivos por primera vez
02e891f9 hemos creado un archivo.
04210aed hemos creado un archivo.
f0a4b21d Initial commit
```

USE YOUR CODE EDITOR FOR LONG COMMIT MESSAGES

Es confiable usar -m cuando el mensaje de tu commit solo es un asunto, sin embargo, es mucho más fácil de usar un editor de código cuando tienes que escribir un cuerpo de texto. Asegurate de separar tu asunto y el cuerpo de tu texto con una línea en blanco.

READ DE PRO GIT BOOK

Para poder mejorar tu manejo de git, el uso del libro pro git te ayudara a aprender sobre diferentes comandos y herramientas que te ayudaran en el manejo de git.

LEARN TO USE GIT WITH COMMAND LINE FIRST

Si estas considerando usar un Git GUI (Interfaz gráfica de usuario, que permite usar git sin usar linea de comandos) como tu primera herramienta de control de versiones, entonces deberías tomarte el tiempo de aprender a usar git desde la línea de comandos y después cambiarte a la interfaz gráfica, de lo contrario no te sentirás como un profesional.

REFERENCIAS

Nawar Yossef. (28 de septiembre del 2017). nawar commit best practices. 29 de septiembre del 2021, de Medium Sitio web: <https://medium.com/@nawarpianist/git-commit-best-practices-dab8d722de99>

