# **MAKE IT REAL PREPARACIÓN:** Git y Github

# **Bienvenido al módulo de Git y Github**

En este módulo vas a aprender lo siguiente:

* Qué es git.
* Cómo crear un repositorio de git.
* Qué son y cómo crear commits.
* Cómo consultar el estado de un repositorio de git.
* Cómo consultar el historial de commits.
* Qué son y cómo trabajar con las ramas de git.
* Para qué sirve y cómo usar el archivo .gitignore
* Al final te recomendamos volver a esta sección y verificar que tengas claros estos conceptos.

Aún si ya has trabajado con git, revisar el contenido de esta sección puede ayudarte a cubrir vacíos de conocimiento o a apropiar mejores prácticas. Empecemos!

# **Primeros pasos**

Git es un sistema de control de versiones distribuido creado por Linus Torvalds.

Un sistema de control de versiones nos permite guardar el historial de cambios de nuestros proyectos y trabajar con otros colaboradores en un mismo proyecto.

Git es una aplicación para la línea de comandos. Para instalarla y configurarla sigue las instrucciones que se encuentran en este enlace.

El concepto más importante de git es el repositorio. Un repositorio es una carpeta que va a almacenar todo el historial de cambios de un proyecto.

Para crear un repositorio ejecuta el comando git init sobre la carpeta de tu proyecto. Eso va a crear una carpeta oculta llamada .git.

El historial de cambios se crea a partir de commits. Un commit es una fotografía de tu proyecto en un momento determinado.

Creando un commit

Para crear un commit debes ejecutar dos comandos:

git add .

git commit -m 'Acá va el mensaje describiendo los cambios'

l primer comando, git add, le dice a git que incluya todos los cambios en el siguiente comit.

El segundo comando, git commit, es el que crea el commit. Cada commit tiene la siguiente información:

* Un identificador (una cadena larga de caracteres).
* Un autor (nombre y correo electrónico).
* Una fecha
* Un mensaje que describe el commit (lo escribe uno cuando crea el commit).
* Los cambios desde el último commit.

El estado del repositorio

Para ver el estado del repositorio utiliza el comando git status.

Cuando no hay nuevos cambios vas a ver algo así:

$ git status

On branch master

nothing to commit, working tree clean

Si hay cambios git status muestra tres secciones:

1. Archivos que se van a incluir en el siguiente commit.
2. Archivos modificados o eliminados que no serían incluidos.
3. Nuevos archivos.

El historial de commits

Para ver el historial de commits utiliza el comando git log:

$ git log

commit 7844a6552c2f838db7b7bed81f7be61e4b51ac84

Author: Pedro Perez <pedro@gmail.com>

Date: Sun Jan 19 15:33:49 2020 -0500

Modifica el landing principal

commit fc3b5612f0e23158263ad1a47d91d6c8f84f1a1f

Author: Pedro Perez <pedro@gmail.com>

Date: Sun Jan 19 14:43:21 2020 -0500

Crea el landing principal

commit a4fcb6a1c87958a64e305c302757b591822b7075

Author: Pedro Perez <pedro@gmail.com>

Date: Sun Jan 19 14:23:20 2020 -0500

Primer commit

Este comando muestra los commits en orden descendente (primero los más recientes)

Puedes utilizar la opción --oneline para ver una versión resumida que muestra los primeros 7 caracteres del identificador y el mensaje de cada commit:

$ git log --oneline

7844a65 Modifica el landing principal

fc3b561 Crea el landing principal

# a4fcb6a Primer commit

# **El comando git status**

# **Trabajando con ramas**

# **Estado de archivos y .gitignore**

# **Github**

# **Resumen Git**