

Comandos Git

- **git stash**: Descartar temporalmente los cambios

Cuando se está trabajando en una rama y se desea cambiar a otra, no se puede sin guardar los cambios por lo que se hace un commit, sin embargo se pueden descartar los cambios temporalmente con: git stash.

Posteriormente para recuperar los cambios se utiliza el comando

- **git stash pop**

-**git bisect**: Encontrar errores.

Este es el comando para encontrar cuando exactamente se introdujo un error en los archivos. Confirmando pasó a paso por los cambios hechos si es bueno o malo, para encontrar el error.

-**git remote**: Mostrará los nombres de cada uno de los remotos que tienes especificados. Si has clonado tu repositorio, deberías ver al menos origin (origen, en inglés) - este es el nombre que por defecto Git le da al servidor del que has clonado.

-**git remote -v**: Muestra las URLs que Git ha asociado al nombre y que serán usadas al leer y escribir en ese remoto. Si tienes más de un remoto, el comando los listará todos.

-**git remote add**: Añadir un remoto nuevo y asociarlo a un nombre que puedas referenciar fácilmente,
se ejecuta git remote add [nombre] [url]

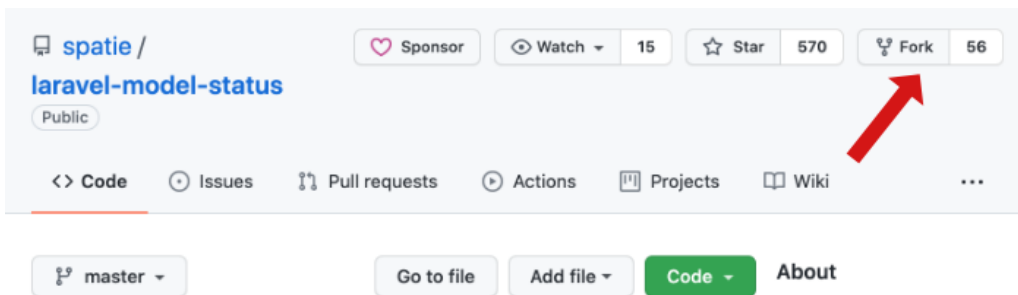
- **git fetch [remote-name]**: irá al proyecto remoto y se traerá todos los datos que aún no tienes de dicho remoto.

Luego de hacer esto, tendrás referencias a todas las ramas del remoto, las cuales puedes combinar e inspeccionar cuando quieras.

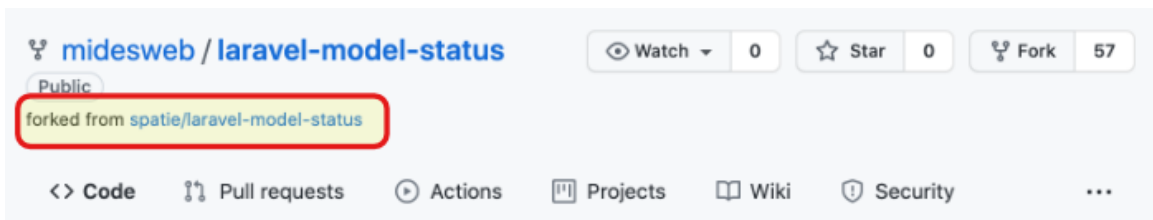
Git Fork

Un fork sirve para crear una copia de un repositorio en nuestra cuenta de usuario. el repositorio se puede tomar prácticamente como un clon pero que al estar en un espacio diferente cambiara de manera distinta. es por así decirlo una rama externa de un repositorio, así puedes modificarlo sin afectar el repositorio original.

Simplemente accedes al repositorio que deseas forkear, y pulsas el botón "Fork".



Una vez realizado el fork **se creará un nuevo repositorio en tu cuenta, con una copia del repo original**. En la página de GitHub se mostrará además que este repositorio es un fork de otro repositorio, el original.



Ahora podrás perfectamente hacer el clon de este fork en local, para descargarlo en tu ordenador de desarrollo. Podrás realizar cambios en el proyecto y luego hacer el commit, para seguidamente subir los cambios a GitHub.

Como los cambios los estás subiendo en un repositorio de tu propiedad, ya que el fork está realizado en tu propia cuenta, GitHub te permitirá publicar las modificaciones realizadas.