



Git Git-Flow

Cómo usar git flow en tus proyectos

Publicado por [Mohit Jain](#) en ⌚ 5 de abril de 2013.

Git es una herramienta de administración de versiones y soy un gran admirador de ella. Espero que hayas venido aquí por Git Flow. Entonces tienes conceptos básicos de git en tus manos. De nuevo en la misma nota de buenas prácticas, creo que todos deberían trabajar en diferentes ramas y luego seguir fusionando esas ramas con la rama principal. Pero eso se vuelve desordenado porque generalmente nos olvidamos de crear una rama. Así que aquí hay una herramienta para ti.

Nota: estoy hablando de ramas predeterminadas aquí. Puede cambiar esa convención de nomenclatura, pero sugeriré que no lo haga.

Volviendo al tema del flujo de git.

#Instalación

Instalación en Mac OS X

```
brew install git-flow
```

Instalación en Ubuntu



Iniciando el flujo de Git:

```
git flow init
```

Le indicará algo como esto:

```
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
```

Mantenga presionada la tecla Intro para usar los valores predeterminados. o simplemente hacer

```
git flow init -d
```

por aceptar todos los valores predeterminados.

¿Cuál es todo el proceso:



- Para cada tarea del proyecto, necesitamos crear una rama y luego fusionar esa rama en la rama de desarrollo.
- Una vez que el código esté listo para la producción. Necesitamos crear una versión y probar el código y hacer las correcciones apropiadas.
- Combine el código fijo en la rama de desarrollo y en la rama de producción.
- Implemente el código y comience a trabajar en nuevas tareas de los proyectos.
- Mientras tanto, puede haber un error en el servidor de producción. Necesitamos corregir el código mediante una revisión y luego fusionar el código fijo en la rama de desarrollo y la rama de producción.
- Ahora retroceda en la tarea que estábamos haciendo antes de la revisión. ;)

Entonces ahora tenemos git flow listo en nuestro repositorio git. Hagamos las cosas una por una.

Crear una nueva rama de características:

Para cada tarea que estamos haciendo, crearemos una característica usando git flow. Supongamos que queremos implementar la tarea de inicio de sesión.

```
git flow feature start login
```

Se creará una nueva sucursal con un espacio de nombres de funciones: función / inicio de sesión



```
git flow feature finish login
```

Se fusionará la rama de función / inicio de sesión para desarrollar la rama y eliminar esta característica / rama de inicio de sesión.

Sigue creando una nueva rama de características. Sigue trabajando en ello. Una vez hecho esto, finalice esa característica y se fusionará en la rama de desarrollo. Empuje esa rama desarrollada para que otras personas puedan tirar de ella.

```
git push origin develop  
git pull origin develop
```

Empujar / tirar de una función.

Puede haber casos en los que necesite presionar la función sin fusionarla para desarrollar una rama.

```
git flow feature publish  
name_of_the_feature_like_we_had_login
```

si quieres sacar alguna característica

```
git flow feature pull origin  
name_of_the_feature_like_we_had_login
```



Ahora tenemos codificación suficiente de codificación. Queremos finalmente probar el código y luego debemos enviarlo a la rama de producción. Creemos primero una versión para que podamos probar una vez y hacer los cambios apropiados si es necesario.

```
git flow release start version_1
```

Ahora pruebe el código y realice los cambios que necesite hacer. Cometer esos cambios. Una vez hecho.

```
git flow release finish version_1
```

Esto fusionará el código en la rama de desarrollo + rama maestra y ahora podemos implementar el código desde la rama maestra y luego volver a desarrollar la rama para implementar nuevas tareas.

Arreglos calientes

Ahora estamos trabajando en una nueva tarea interesante y estamos en una función, es decir, `my_awesome_task` y, mientras tanto, descubrimos que hay un error en la producción en el título de la página. Ahora necesitamos hacer una revisión.

```
git flow hotfix start typo_bug_in_title
```

```
git flow hotfix finish typo_bug_in_title
```

Esto fusionará el código en la rama maestra y en la rama de desarrollo también. Ahora puede implementar el código utilizando la rama maestra y volver a su función `my_awesome_task` y hacer la tarea. Puede consultar su rama de características en la que desea trabajar

```
git flow feature check out my_awesome_task
```

Espero que esto tenga algún sentido. Avísame si tienes dudas o algo.



Gorjeo



Facebook

**Mohit Jain**<http://codebeerstartups.com>

Desarrollador de rieles





[I7s Tws auriculares bluetooth inalámbricos Bluetooth Auriculares auriculares manos libres en](#)

2,64 € - es.aliexpress.com

[La transformación física de Elsa Pataky por sus 43 años](#)

Enfemenino

[Haz clic aquí para conocer a mujeres solteras en la región de Port De Pollença](#)

Meetic

[Miriam Sánchez muestra cómo es su espectacular cambio físico sin nada de ropa](#)

Look

[Descubre la gama de vehículos electrificados que te ofrece KIA #electrificate](#)

KIA

[British Expat in Spain? £70k-£1m In UK Pensions - Understand Your Options](#)

Abbey Wealth

0 Comments

codebeerstartups

Javier Martín Alon... ▾

Recommend

Tweet

Share

Sort by Newest ▾



Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your siteAdd DisqusAdd





[Haz clic aquí para conocer a mujeres solteras en la región de Port De Pollença](#)

Meetic

[Miriam Sánchez muestra cómo es su espectacular cambio físico sin nada de ropa](#)

Look

[Play this Game for 1 Minute and see why everyone is addicted](#)

Desert Order

[Plan for a Better Retirement in Spain or the UK in Under 20 Minutes](#)

Copyright © [Codebeerstartups](#) . 2014 • Todos los derechos reservados.

Orgullosamente publicado con [Jekyll](#) .