

---

# Introducción a alojamientos para Git

PGPI  
E.T.S.I. Informática  
Universidad de Sevilla

Jorge Juan <jjchico@dte.us.es> 2014-2018

Usted es libre de copiar, distribuir y comunicar públicamente la obra y de hacer obras derivadas siempre que se cite la fuente y se respeten las condiciones de la licencia Attribution-Share alike de Creative Commons.

Puede consultar el texto completo de la licencia en <http://creativecommons.org/licenses/by-sa/3.0/>



---

*Departamento de Tecnología Electrónica - Universidad de Sevilla*

## Contenidos

---

- Servicios de alojamiento Git
- Repositorios públicos y privados
- Crear un repositorio
- Contribuir a un proyecto
- Flujos de trabajo



---

*Departamento de Tecnología Electrónica - Universidad de Sevilla*

# Servicios de alojamiento

---

- Proporcionan repositorios remotos
- Servicios adicionales
  - Gestión vía web
  - Control de permisos
  - Wiki/web del proyecto
  - Descarga de archivos/paquetes
- Ejemplos: Github, Bitbucket, Gitlab, etc.

# Servicios de alojamiento

---

- Github
  - Gratis: infinitos repos públicos, seguimiento de errores
  - De pago: repos privados, servicios adicionales
- Bitbucket
  - Gratis: infinitos repositorios públicos o privados, grupos de hasta 5 miembros.
  - De pago: gestión de grupos, servicios adicionales
- Gitlab
  - Gratis: infinitos repositorios públicos o privados, infinitos colaboradores.
  - De pago: servicios adicionales
  - Basado en software libre

## Servicios de alojamiento (planes gratuitos en 2016)

Free Plans	Public Repos	Private Repos	Collaborators	Storage Space	Hosting	Support
GitHub Public	Unlimited	0	Unlimited	N / A	Cloud	Email / Forum
Bitbucket Small teams	Unlimited	Unlimited (1Gb /project)	5	N / A	Cloud	Email / Forum
GitLab Cloud Hosted	Unlimited	Unlimited (10Gb / Project)	Unlimited	Unlimited	Cloud	Forum
GitLab Community Edition	Unlimited	Unlimited	Unlimited	N / A	Self-hosted	Forum
Coding Free Plan	Unlimited	Unlimited	10	1GB	Cloud	Email /Forum



Fuente: <https://medium.com/flow-ci/github-vs-bitbucket-vs-gitlab-vs-coding-7cf2b43888a1#.6e3db8cp1>

Departamento de Tecnología Electrónica - Universidad de Sevilla

## Servicios de alojamiento

Paid Plans	Users	Private Repository	Price [USD]	Payment Frequency	Hosting	Support
GitHub Personal	1	Unlimited	7	Monthly	Cloud	Email / Forum
GitHub Organisation	5	Unlimited	25	Monthly	Cloud	Email / Forum
	200 + 5		25 + 1800			
Bitbucket Growing Team ( min10 - unlimited)	10	Unlimited	10	Monthly	Cloud	Email / Forum
	Unlimited		200			
Coding Developer Plan	20	Unlimited (5GB)	7.3	Month	Cloud	Email / Forum
			36.67	6 Month		
			66	Year		
Coding Advanced Plan	50	Unlimited (10GB)	30	Month	Cloud	Email / Forum / Chat / Phone
			149	6 Month		
			268	Year		



Fuente: <https://medium.com/flow-ci/github-vs-bitbucket-vs-gitlab-vs-coding-7cf2b43888a1#.6e3db8cp1>

Departamento de Tecnología Electrónica - Universidad de Sevilla

# Servicios de alojamiento

---

“An exciting active GitHub profile could help you landing a great job. In more and more cases recruiters favor candidates with an active GitHub profile.”

# Repositorios públicos y privados

---

- Es muy frecuente manejar repositorios públicos:
  - La mayoría de los proyectos alojados en los servidores Git son proyectos de software libre.
  - Cualquier proyecto, incluso privado, suele emplear recursos de proyectos libres:
    - Bibliotecas de programación
    - Elementos de diseño, gráficos, iconos, etc.
    - No sólo software: diseños de PCB, huellas de componentes, código Verilog, etc.
- Repositorios privados
  - Acceso restringido a grupos de desarrolladores
  - Tendencia a emplear los mismos flujos de trabajo que los repositorios públicos: influencia del software libre en el software privativo

## Requisitos de repositorios públicos (también mayoría de privados)

---

- Identificar claramente al autor o autores.
- Documentación mínima: descripción general del proyecto
  - README.txt, README.md
- Establecer o informar claramente de la licencia aplicada al proyecto.
- Establecer claramente las reglas para contribuir al proyecto (si es que se aceptan contribuciones).
  - Ej: en README.md o CONTRIBUTING.md

## Crear un nuevo proyecto

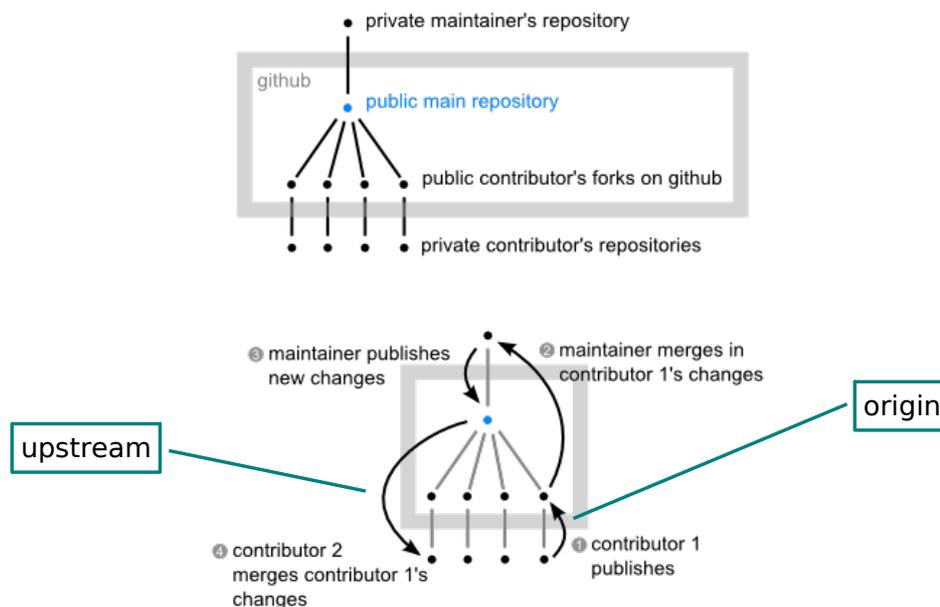
---

- Proyecto totalmente nuevo
  - Crear en servicio de repositorios
  - Clonar localmente
- Proyecto local existente
  - Crear repositorio remoto
  - Añadir remoto al proyecto local
  - Enviar cambios (push)
- Los proveedores de repositorios dan instrucciones detalladas a la hora de crear/subir nuevos proyectos.

# Contribuir a un proyecto: flujos de trabajo

- Los alojamientos Git facilitan el desarrollo de proyectos colaborativos al alojar en un sólo lugar los repositorios remotos de todos los desarrolladores.
- El servicio del proveedor (web) facilita muchas tareas:
  - Visualizar cambios de otros proyectos.
  - Copiar otros repositorios (*fork*).
  - Aportar cambios a otros proyectos.
  - Revisar y comentar los cambios propuestos.
  - Hacer seguimiento de errores, etc.

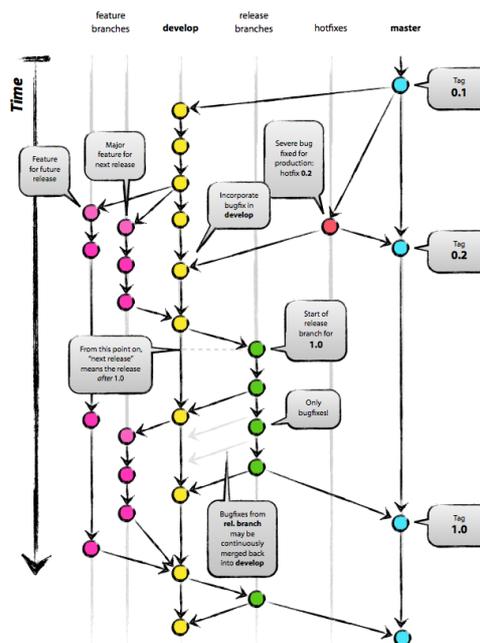
# Contribuir a un proyecto. Mecanismo básico



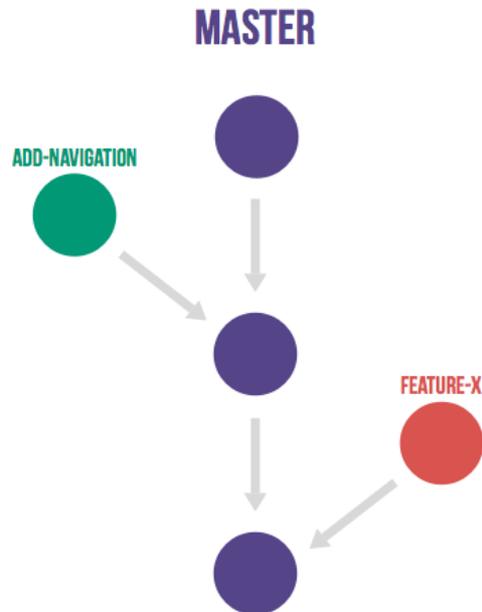
# Flujos de trabajo

- Cada proyecto colaborativo debe establecer un flujo de trabajo claro:
  - Facilitar la colaboración.
  - Evitar ramas obsoletas.
  - Mantener una base de código estable, etc.
- Algunos flujos de trabajo:
  - **Gitflow**
    - Completo pero complejo.
  - **GitHub flow**
    - Simple y efectivo para proyectos no muy grandes.
  - **GitLab flow**
    - Similar a GitHub flow, con ramas específicas para despliegue y publicación de versiones.

## Git flow



# GitHub Flow



# GitLab Flow

