

Git

Emilio Silva

Quantil S.A.S.

Grupo de Lectura, noviembre 5 de 2015

Agenda

- 1 Qué es Git
 - Control de Versiones Local
 - Control de Versiones Centralizado
 - Control de Versiones Distribuido
- 2 Cómo usar Git
 - Cómo hacer cambios
 - Cómo obtener los cambios de otros
 - Cómo trabajar en equipo

Agenda

- 1 Qué es Git
 - Control de Versiones Local
 - Control de Versiones Centralizado
 - Control de Versiones Distribuido
- 2 Cómo usar Git
 - Cómo hacer cambios
 - Cómo obtener los cambios de otros
 - Cómo trabajar en equipo

Control de Versiones Local

- Recordar los cambios de unos archivos
- Volver a versiones anteriores
- Mirar en qué momento se introdujo un error
- Todo tipo de archivos de texto

Agenda

1 Qué es Git

- Control de Versiones Local
- Control de Versiones Centralizado
- Control de Versiones Distribuido

2 Cómo usar Git

- Cómo hacer cambios
- Cómo obtener los cambios de otros
- Cómo trabajar en equipo

Control de Versiones Centralizado

- Varias personas modifican un mismo archivo
- Saber quién hizo qué
- Entender en qué consiste cada cambio
- Un servidor central; todos publican ahí

Agenda

1 Qué es Git

- Control de Versiones Local
- Control de Versiones Centralizado
- Control de Versiones Distribuido

2 Cómo usar Git

- Cómo hacer cambios
- Cómo obtener los cambios de otros
- Cómo trabajar en equipo

Control de Versiones Distribuido

- ¿Qué pasa si el servidor no está disponible?
- ¿Qué pasa si el servidor se daña?
- En un sistema distribuido, *cada colaborador tiene toda la historia*
- Si se daña el servidor central, se puede recuperar con cualquier copia local

Agenda

- 1 Qué es Git
 - Control de Versiones Local
 - Control de Versiones Centralizado
 - Control de Versiones Distribuido
- 2 Cómo usar Git
 - Cómo hacer cambios
 - Cómo obtener los cambios de otros
 - Cómo trabajar en equipo

Pasos

Para hacer un cambio, en Git hay que:

- 1 Hacer el cambio
- 2 Revisarlo y aceptarlo
- 3 Escribir de qué se trata y por qué lo hice
- 4 Publicarlo

Nombres de los pasos

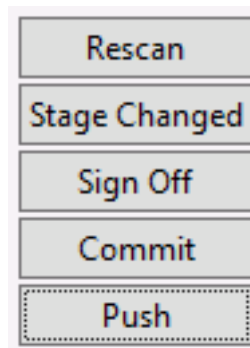
Para hacer un cambio, en Git hay que:

- 1 Hacer el cambio
- 2 Revisarlo y aceptarlo - **STAGE**
- 3 Escribir de qué se trata y por qué lo hice - **COMMIT**
- 4 Publicarlo - **PUSH**

Nombres de los pasos

Para hacer un cambio, en Git hay que:

- 1 Hacer el cambio
- 2 Revisarlo y aceptarlo - **STAGE**
- 3 Escribir de qué se trata y por qué lo hice - **COMMIT**
- 4 Publicarlo - **PUSH**



Los 4 Estados

Los 4 estados de los cambios son:

- 1 **UNSTAGED** - No he revisado el cambio
- 2 **STAGED** - Revisé y confirmé este cambio
- 3 **COMMITTED** - Describí en qué consiste y por qué es importante este cambio
- 4 **PUSHED** - Publiqué este cambio en internet

Granularidad

Granularidad:

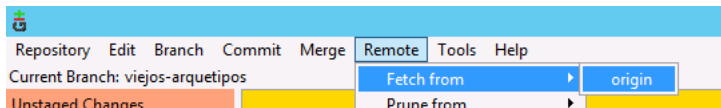
- 1 **UNSTAGED** - Archivo
- 2 **STAGED** - Archivo
- 3 **COMMITTED** - Grupo de archivos
- 4 **PUSHED** - Grupo de archivos

Agenda

- 1 Qué es Git
 - Control de Versiones Local
 - Control de Versiones Centralizado
 - Control de Versiones Distribuido
- 2 Cómo usar Git
 - Cómo hacer cambios
 - **Cómo obtener los cambios de otros**
 - Cómo trabajar en equipo

Fetch

- Antes de hacer cambios, es importante tener los cambios que otros ya han hecho
- Esto se hace mediante el comando **FETCH**



Agenda

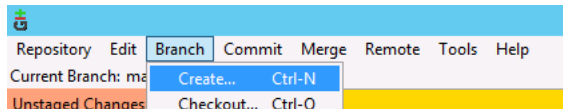
- 1 Qué es Git
 - Control de Versiones Local
 - Control de Versiones Centralizado
 - Control de Versiones Distribuido
- 2 Cómo usar Git
 - Cómo hacer cambios
 - Cómo obtener los cambios de otros
 - Cómo trabajar en equipo

Pull Requests

- Una única persona debe ser responsable de los cambios de un proyecto
- Todos los demás publican cambios, pero sólo son integrados dentro del proyecto cuando esta persona los aprueba
- Las solicitudes de cambio se llaman **Pull Requests**

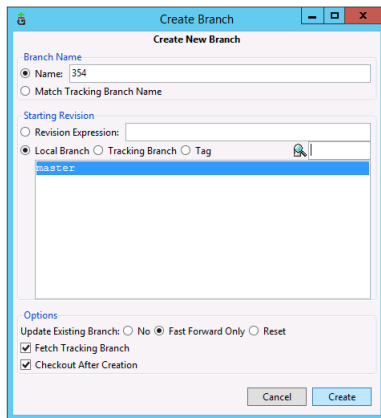
Cómo crear una Pull Request

1. Cree una nueva Branch



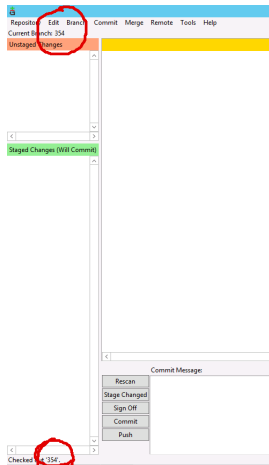
Cómo crear una Pull Request

2. Póngale un nombre significativo (Task Management Tool)



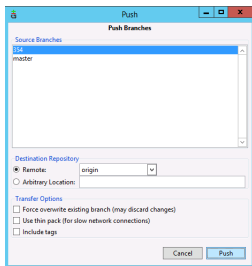
Cómo crear una Pull Request

3. Verifique que está en la branch correcta



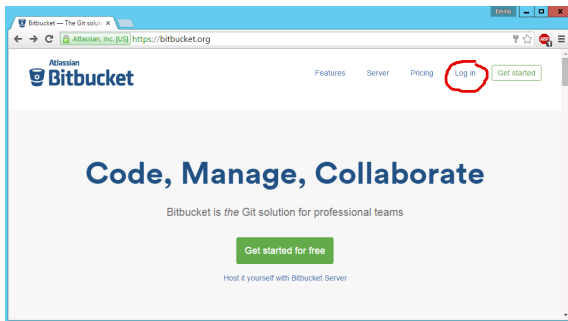
Cómo crear una Pull Request

4. Haga PUSH



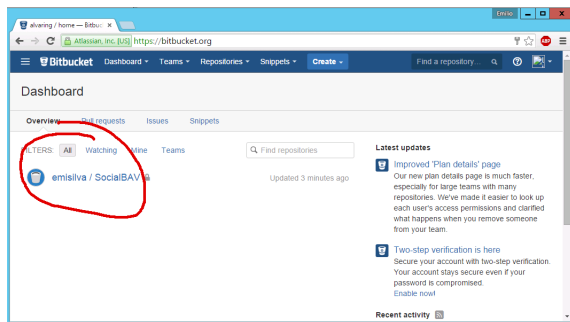
Cómo crear una Pull Request

5. Inicie sesión en BitBucket



Cómo crear una Pull Request

6. Seleccione el repositorio en cuestión



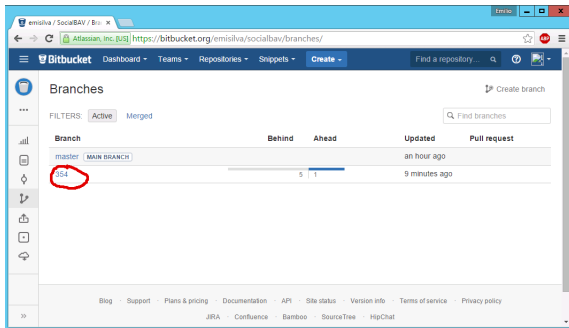
Cómo crear una Pull Request

7. Vaya a Branches

The screenshot shows the Bitbucket web interface for a repository named 'emiliva / SocialBAV'. The page includes a navigation bar with 'Dashboard', 'Teams', 'Repositories', and 'Snippets'. The main content area is titled 'Overview' and displays repository statistics: 'Last updated 7 minutes ago', 'Language R', 'Access level Write (revoke)', '2 Branches', '0 Tags', '0 Forks', and '1 Watcher'. The '2 Branches' value is circled in red. On the left sidebar, the 'Branches' icon is also circled in red. Below the statistics, there is a message: 'THERE ISN'T A README YET' with a subtext 'A README file tells people where to start and how to contribute.' The right sidebar shows 'Recent activity' with a list of commits and their authors.

Cómo crear una Pull Request

8. Seleccione la Branch en cuestión



Cómo crear una Pull Request

9. Cree la Pull Request

