

ÍNDICE

Comandos git	2
Spring Batch	4
Spring MVC	5

Comandos git

Pull Request

El pull Request son solicitudes para combinar el proyecto main con las ramas que han solicitado hacer merge, estos solo serán aceptados por el encargado del proyecto quienes se encargaran de verificar que estos cambios tengan conflictos.

Fork

El fork nos permite tener una instancia de un proyecto como nuestro pero un repositorio propio permitiendo hacer cambios a este sin afectar al original y permitiendo a ambos proyectos evolucionar de formas diferentes, esto va de la mano con el pull request ya que si deseamos hacer un cambios al proyecto original podremos solicitar hacer un merge al original.

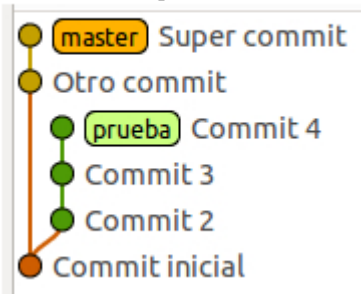
Stash

El comando git stash nos permite guardar los cambios que hayamos hecho y regresa nuestro código a como estaba antes de los cambios. para volver a ver estos cambios aplicados simplemente deberemos hacer un git stash pop.

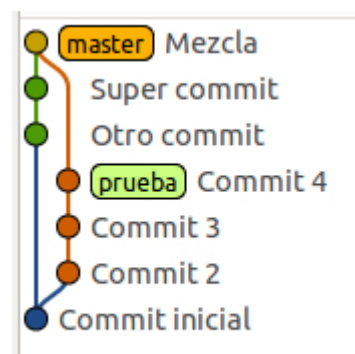
Rebase

El git rebase es un comando que nos permite hacer merge entre ramas pero simplificando el historial de commits de manera más simple el merge nos genera un historial ramificado mientras que con el base nos genera un historial lineal.

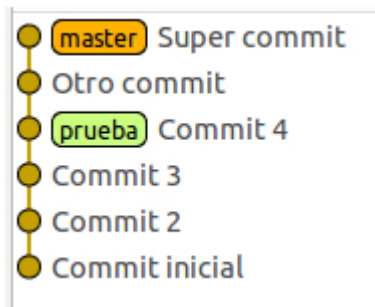
Punto de partida



Merge



Rebase

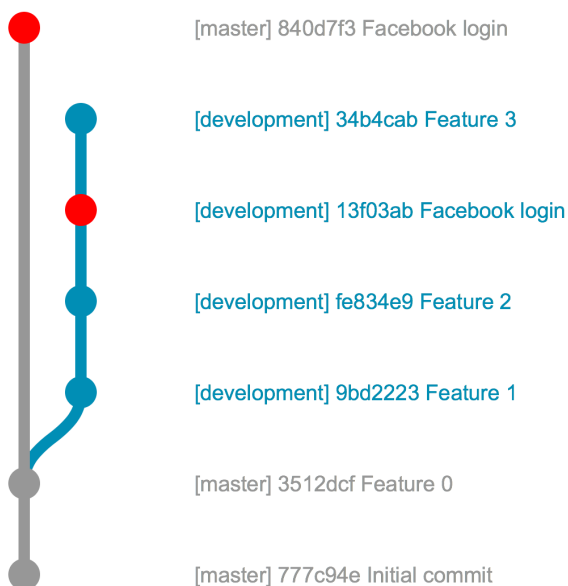


Clean

El comando `git clean` funciona de forma similar al `git reset` o `git checkout` pero a diferencia de estos que trabajan sobre archivos modificados `git clean` trabaja con los archivos versionados es decir elimina todos aquellos que no están en el repositorio o aquellos que no sean ignorados por el `gitignore`.

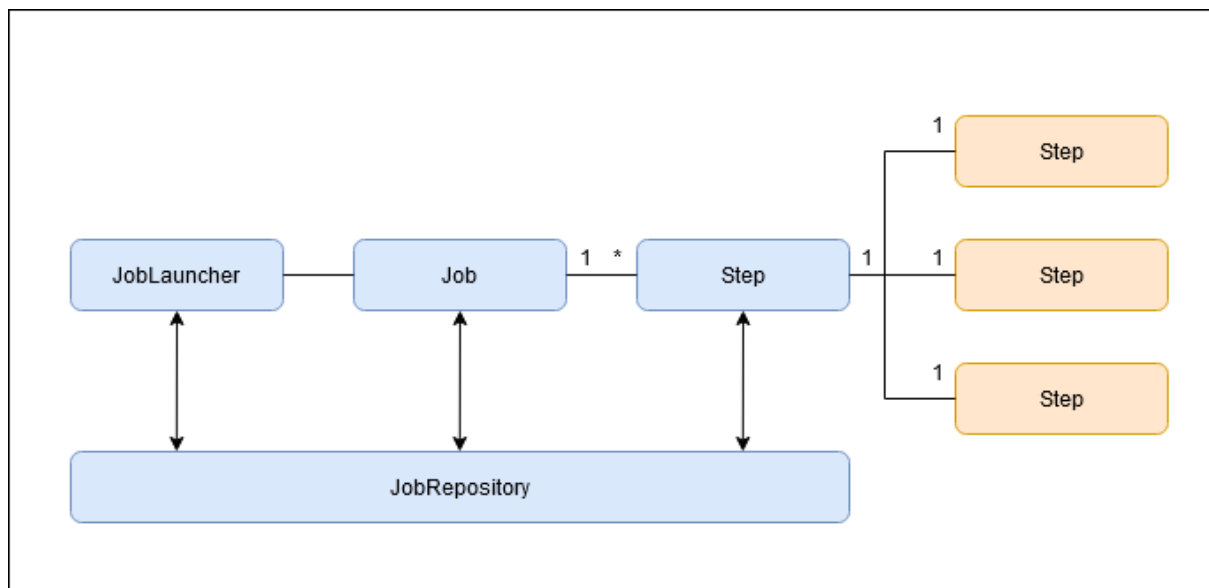
Cherry Pick

El comando `cherry pick` funciona de cierto modo como el `git merge` es decir combinamos una rama con otra pero a diferencia de este nos permite seleccionar un commit específico y así evitamos hacer todo el merge entre ramas.



Spring Batch

Es un framework enfocado a la creación de procesos los cuales se ejecutan solos y serán los encargados de procesar grandes cantidades de datos. este se divide en varias secciones, JobLauncher el cual se encarga de iniciar el proceso, JobRepository en este se encuentra toda la información que será necesaria para el job, el job es el bloque o sección de trabajo el cual se conforma de varios steps estos son los pasos que necesita el job para ejecutar su tarea los cuales son ItemReader el cual se encarga de leer los datos, ItemProcessor se encarga de procesar los datos y transformarlos y por ultimo el ItemWriter este se encarga de guardar los datos en archivos o dbs.



Spring MVC

