

Semana 1 - 2





Cuántos cambios
hemos hecho ayer
en lo avanzado?

Han sido muchos...

The screenshot shows an IDE workspace with a sidebar on the left and a main editor area on the right. The sidebar contains icons and labels for WORKSPACE, File status, History, Search, BRANCHES, TAGS, REMOTES, origin, STASHES, SUBMODULES, and SUBTREES. The main editor area is divided into two panes. The top pane, titled 'Pending files, sorted by filename', shows a table of staged files. The bottom pane shows a diff view of the selected file, .classpath.

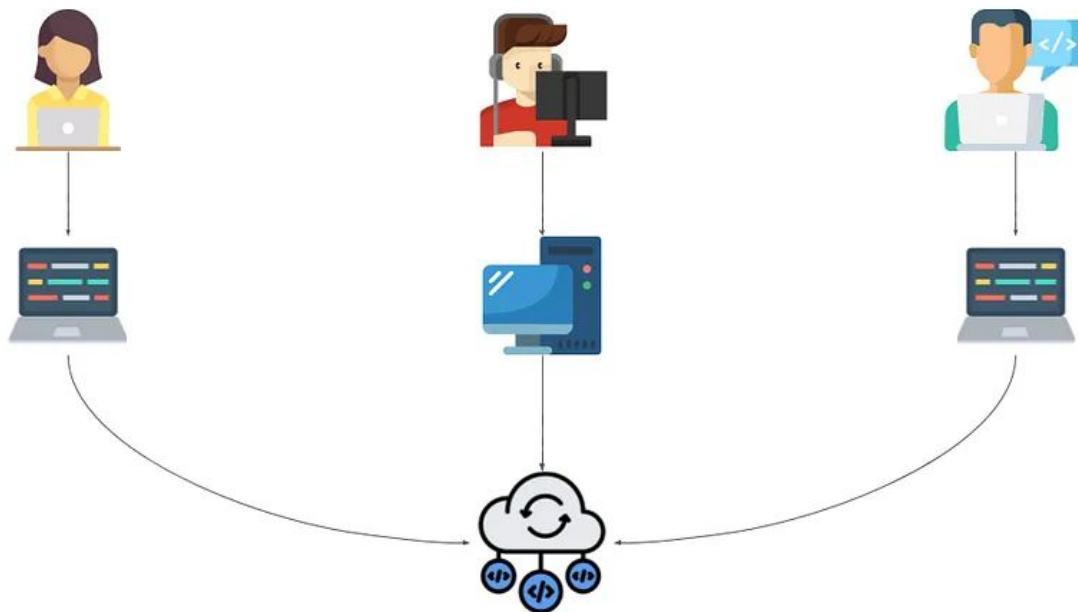
Staged files	
Filename	Path
✓ .classpath	

Unstaged files	
Filename	Path
.project	
log4j2.xml	src/main/resources
pom.xml	

Diff view of .classpath (Hunk 1: Lines 1-45):

```
1 1 <?xml version="1.0" encoding="UTF-8"?>
2 2 <classpath>
3 3 - <classpathentry excluding="**/*.java" kind="src" path="src/main/api"/>
4 4 - <classpathentry excluding="mule-app.properties|**/*.java" kind="src" path="src/main/app"/>
5 5 - <classpathentry kind="src" path="src/main/java"/>
6 6 - <classpathentry excluding="**/*.java" kind="src" path="src/main/resources"/>
7 7 - <classpathentry kind="src" output="target/test-classes" path="src/test/java"/>
8 8 - <classpathentry excluding="**/*.java" kind="src" output="target/test-classes" path="src/test/munit"/>
9 9 - <classpathentry excluding="**/*.java" kind="src" output="target/test-classes" path="src/test/resources"/>
10 10 - <classpathentry kind="con" path="MULE_RUNTIME"/>
11 11 - <classpathentry kind="con" path="org.eclipse.jdt.launching.JRE_CONTAINER"/>
12 3 + <classpathentry excluding="**" kind="src" output="target/classes" path="src/main/api">
13 4 +   <attributes>
14 5 +     <attribute name="maven.pomderived" value="true"/>
15 6 +   </attributes>
16 7 + </classpathentry>
17 8 + <classpathentry excluding="**" kind="src" output="target/classes" path="src/main/app">
18 9 +   <attributes>
19 10 +     <attribute name="maven.pomderived" value="true"/>
20 11 +   </attributes>
21 12 + </classpathentry>
```

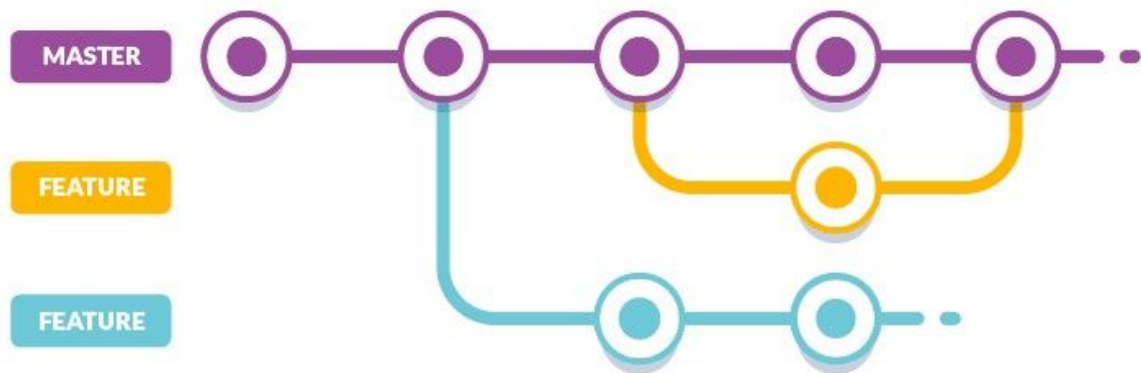
Se suele colaborar en equipo...



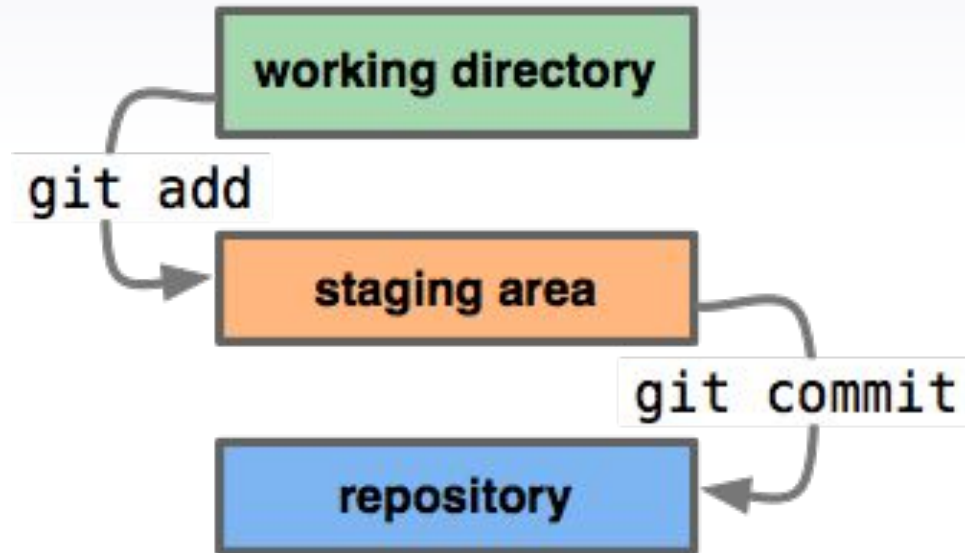
Git



git



► Flujo de trabajo Local



Flujo de trabajo Local

//Me sirve para ver mi conf. global actual

git config --list

//configurando...

git config --global user.name "Nombre Apellidos"

git config --global user.email "micorreo@gmail.com"

// para inicializar git dentro de una carpeta - se inicializa 01 vez por proyecto

git init

//ver estado de mi repositorio

git status

//agrega los cambios a una etapa de stage (bandeja)

git add <carpeta o directorio a añadir>

git add .

//para hacer el commit y guardar los cambios

git commit -m "mensaje descriptivo del cambio"

Ejercicio

Creación de un repositorio local:

Crea una nueva carpeta para trabajar.

Abre tu terminal (en Windows, puedes usar Git Bash o la terminal de tu elección).

Asegurate de que la terminar esta en la carpeta donde vas a trabajar.

Ejecuta el comando **git init** para inicializar un repositorio Git vacío en ese directorio.

Creación de un archivo HTML:

Utiliza Visual Studio Code para crear un archivo HTML en el directorio del repositorio.
Por ejemplo, puedes crear un archivo llamado index.html.
Agregando contenido al archivo HTML:

Abre el archivo index.html en tu editor de texto.
Agrega el siguiente contenido básico:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Título de tu página</title>  
</head>  
<body>  
  <h1>Mi primer título</h1>  
  <p>Este es mi primer párrafo.</p>  
    
</body>  
</html>
```

Guarda el archivo.

Añadiendo el archivo al área de preparación:

En tu terminal, ejecuta el comando **git add .** para agregar el archivo HTML al área de preparación de Git.

Haciendo commit de los cambios:

Ejecuta el comando **git commit -m "Agregado archivo HTML con título"** para hacer commit de los cambios. Asegúrate de proporcionar un mensaje descriptivo para el commit.
Añadiendo un párrafo y una imagen:

Regresa a tu archivo index.html en el editor de texto.

Agrega un nuevo párrafo o modifica el existente según lo necesites.
Añade una etiqueta para insertar una imagen. Asegúrate de especificar la ruta correcta de la imagen y una descripción adecuada.

Añadiendo y haciendo commit de los cambios adicionales:

Vuelve a ejecutar **git add .** para agregar los cambios al área de preparación.
Luego, ejecuta **git commit -m "Añadido párrafo e imagen"** para hacer otro commit con estos cambios.

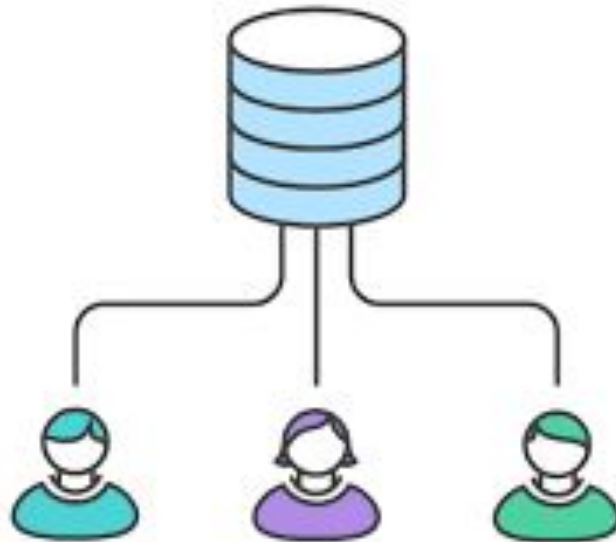
Verifica tus cambios:

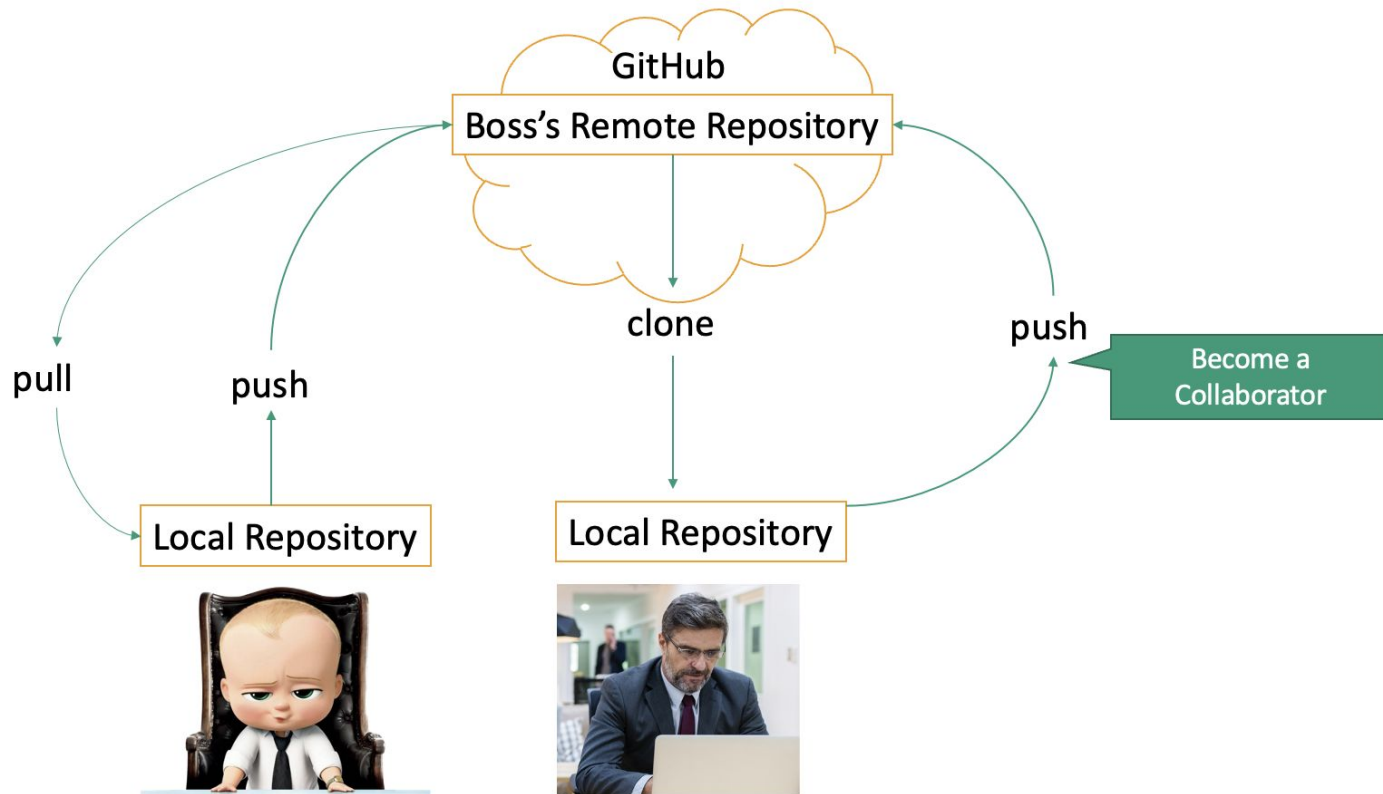
Ejecuta ***git log*** en tu terminal para mostrar una lista de los commits realizados, junto con detalles como el autor, la fecha y el mensaje del commit.

Esto te permitirá tener una visión general de la historia de cambios en tu repositorio y verificar que tus cambios se han registrado correctamente.

► Ok y como
hacemos para
trabajar juntos

► Flujo colaborativo





Recursos



Curso de Git: <https://www.atlassian.com/es/git>

Git en una hora:

<https://www.youtube.com/watch?v=VdGzPZ31ts8&t=1s&pp=ygUDZ2l0>

<https://docs.github.com/es/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

<https://learngitbranching.js.org/>

