

## Sesión # 4 Componente Práctico

### Control de versiones con GIT

Teniendo en cuenta lo aprendido en clase, vamos a crear un nuevo proyecto de Git al que llamaremos 'web-mintic', agrega a tu equipo de trabajo como colabores y según el enunciado de la sesión anterior implementa de manera muy sencilla los siguientes archivos en html (luego les añadiremos estilos y demás) puedes usar como template los archivos de la carpeta [recursos](#). Ten muy presente las ramas en las que debes crearlas.

1. Configura Github en tu equipo. Puedes seguir las siguientes instrucciones para [instalar la consola](#).
2. Crea un repositorio para tu solución y añade a tu equipo como colaboradores
3. Crea las siguientes vistas en la rama indicada:
  - Home (creala en la rama master)
  - Noticias (creala en una rama diferente a master, llamada news utiliza el comando `git checkout branchName`)  
Ejemplo: pdte  

```
git branch  
-master  
git branch news  
git checkout news  
git branch  
-news
```
  - Inicio de sesión (creala en la rama diferente a master, llamada authentication. Realiza commit con estos cambios)
  - Registro (creala en la rama diferente a master, llamada authentication. Realiza un commit con estos cambios)

*Nota\*\*:* Si deseas, puedes ver los archivos creados arrastrandolos al navegador.

4. Contesta las siguientes preguntas:
  - Utiliza el comando `git log --all --decorate --oneline --graph` en tu consola de Git, ¿que puedes observar?
  - Realiza un pull request de la rama news a la rama master y apruebalo. Esto generará un merge en tu master, utiliza nuevamente el comando del inciso anterior y observa ¿cómo queda tu árbol de commits luego de esto? ¿Qué diferencias puedes notar?

- Realiza rebase de la rama authentication en la rama master. Observa nuevamente el árbol de logs, ¿que diferencias puedes notar entre git rebase y git merge?
- Imagina que quieres revertir un cambio realizado en tu rama de authentication. Utiliza uno de los siguientes comandos para revertir el último commit.  
`git revert hashCommit`  
`git revert HEAD~1`

**Material de apoyo:**

Revertir un commit: <https://www.youtube.com/watch?v=4SwaV29Splc>