**DAW1**

M01: Sistemes informàtics

UF02

Act04: Acronis

*22/2/2023*

*Marc Arcedo Alvaro*



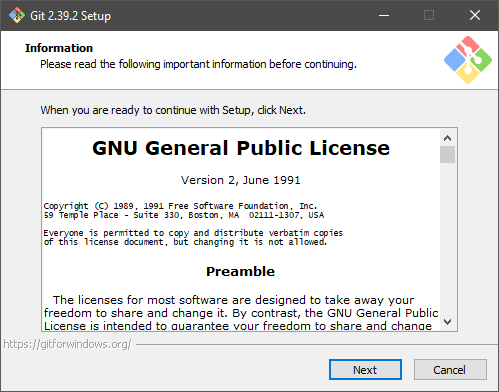
Índex

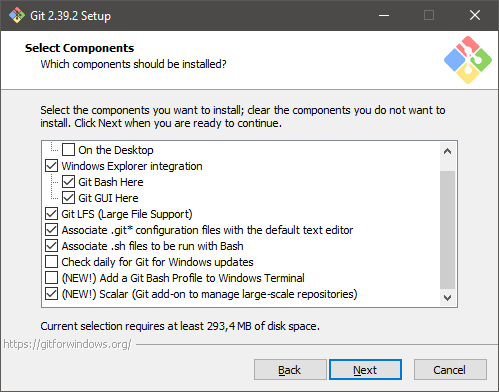
[Ex1 3](#_Toc126685837)

[Ex2 8](#_Toc126685838)

## Ex1

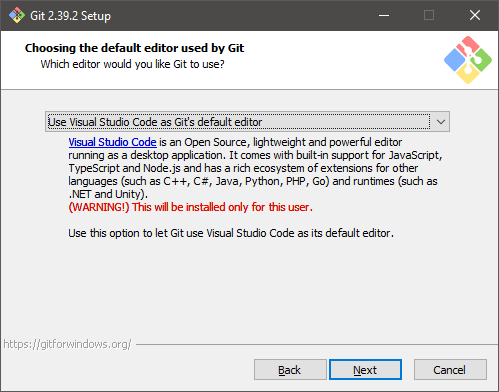
Creació de la imatge del sistema:

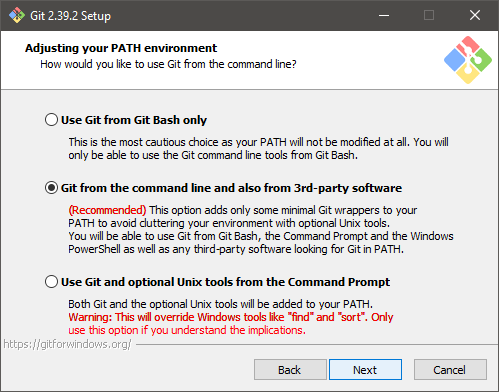


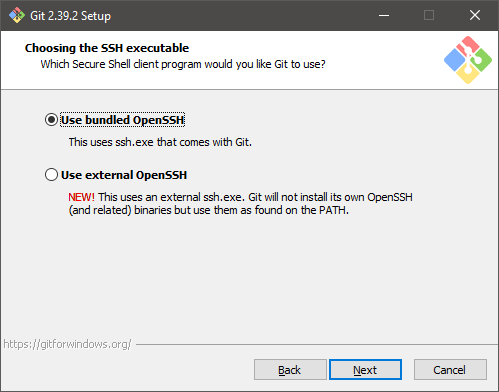


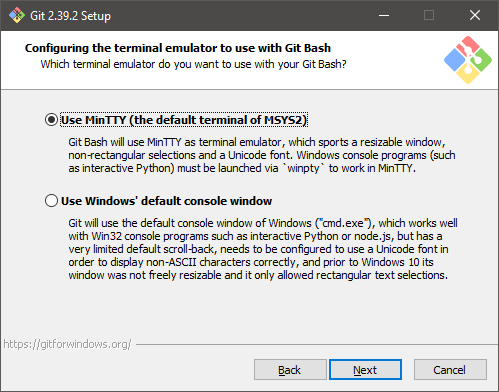
Docker

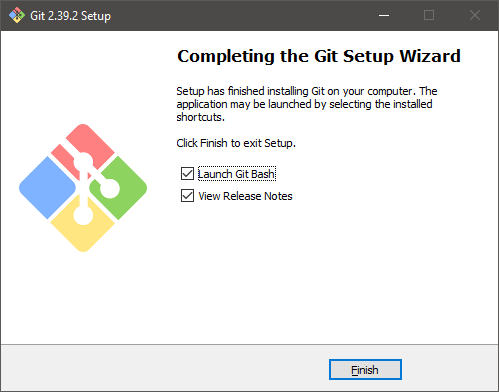
Cubernetes

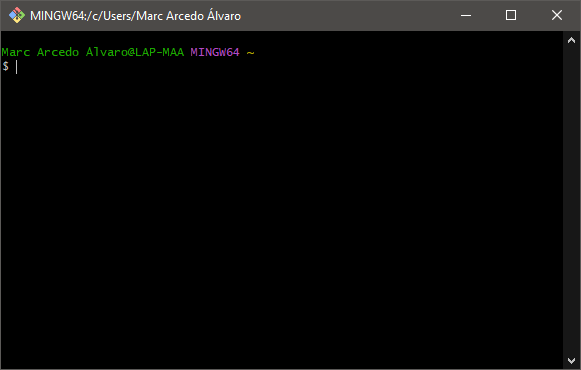


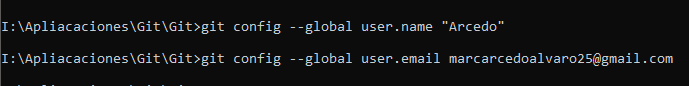


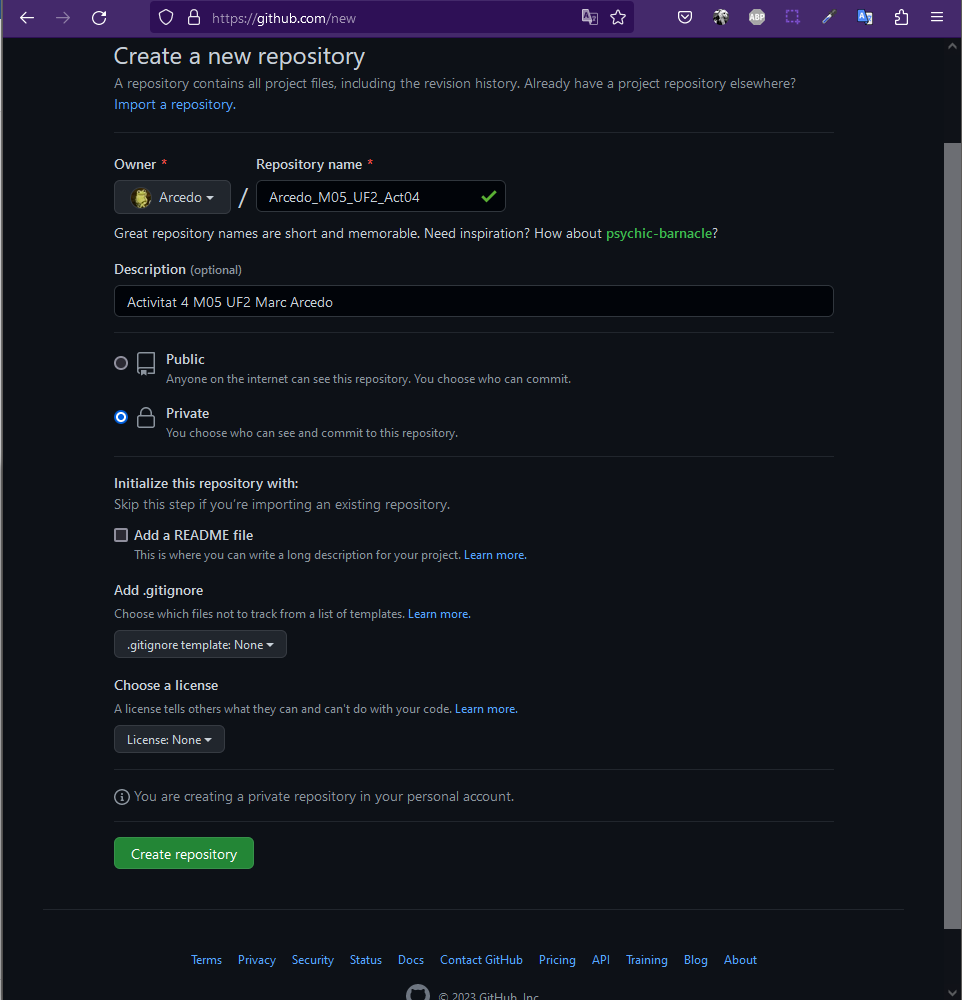


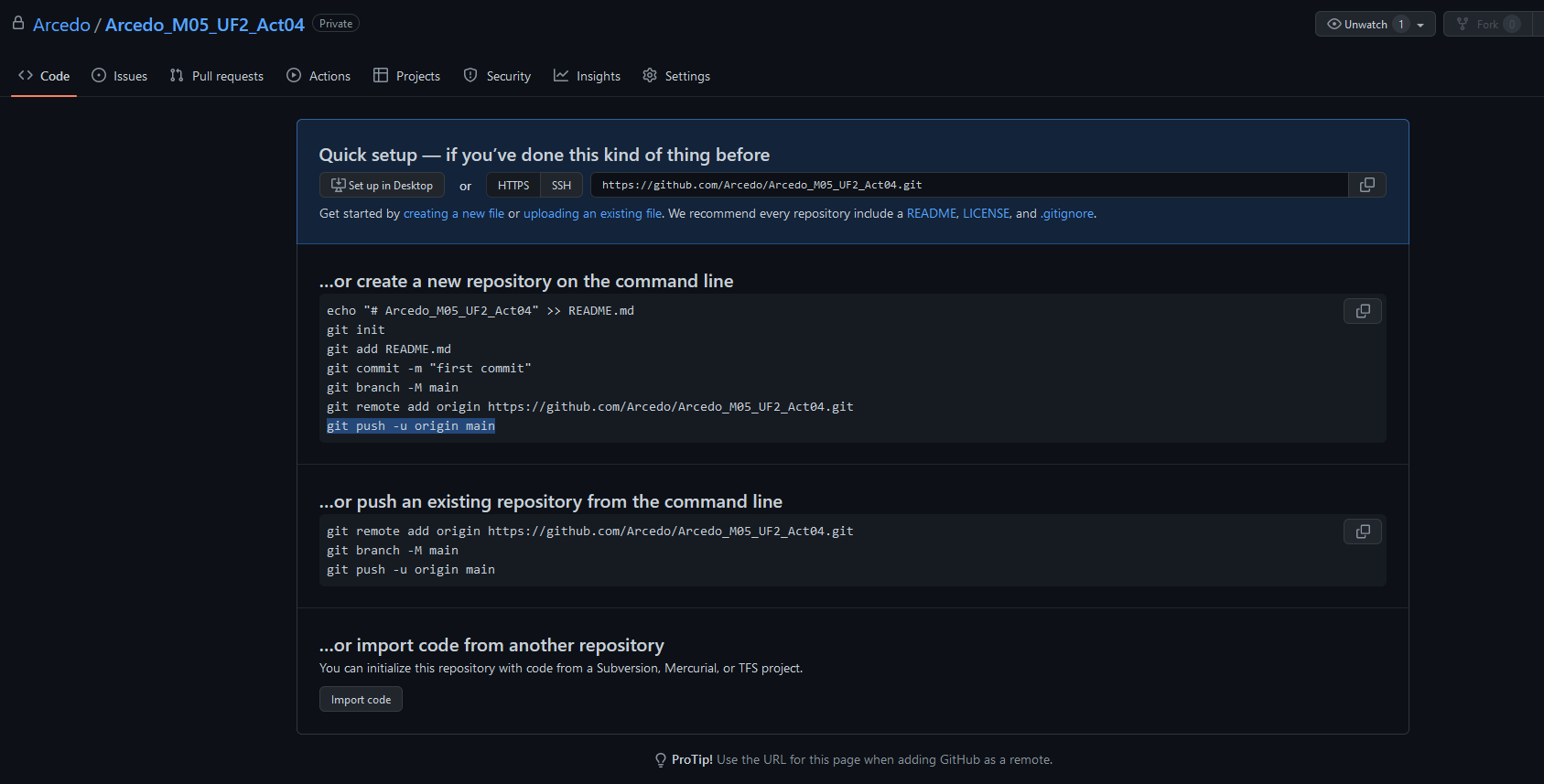


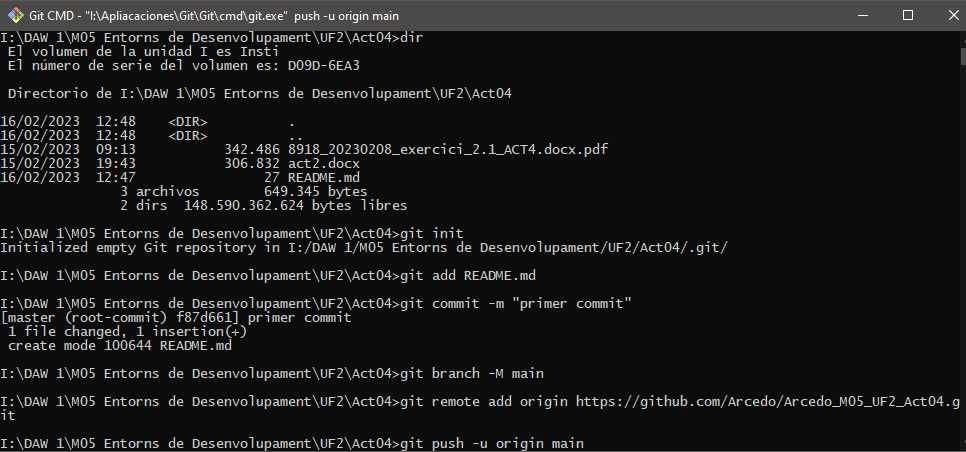


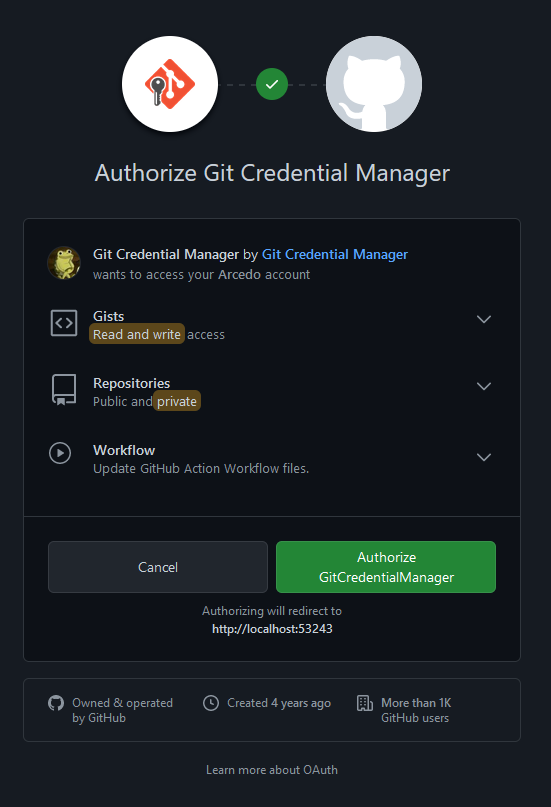


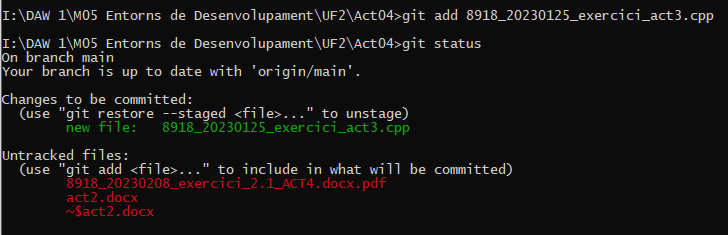


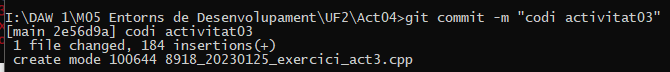






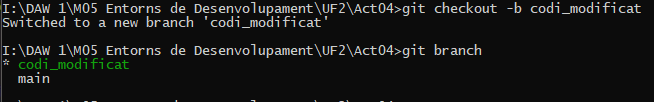




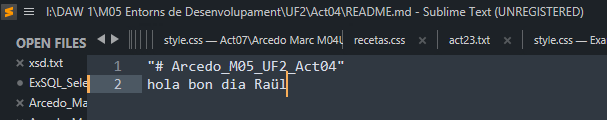


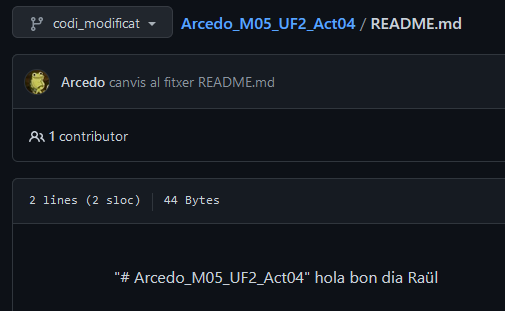
Una branca de git serveix per si hi ha diferents projectes en curs amb més persones cadascú tingui la seva part separada. Això permet que sense voler algú editi algun fitxer o que es solapin els treballs i estiguin separats de la branca main. Quan el codi estigui correcte es pujarà a la branca main.

Per fer la comprovació he creat una nova branca que es diu codi\_modificat on faré canvis al fitxer README.md i ho actualitzare i pujaré a aquesta branca i posteriorment fusionare les branques

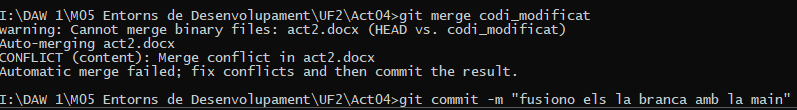


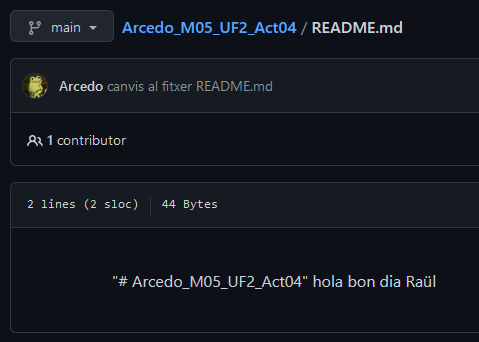
A aquesta branca he afegit tots els fitxers.



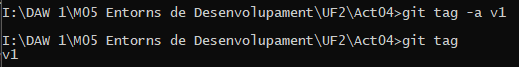


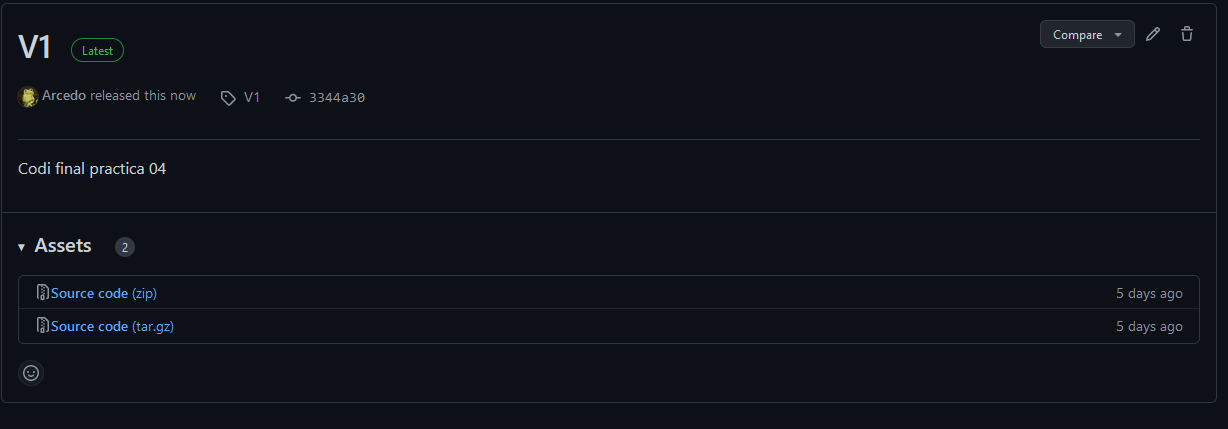


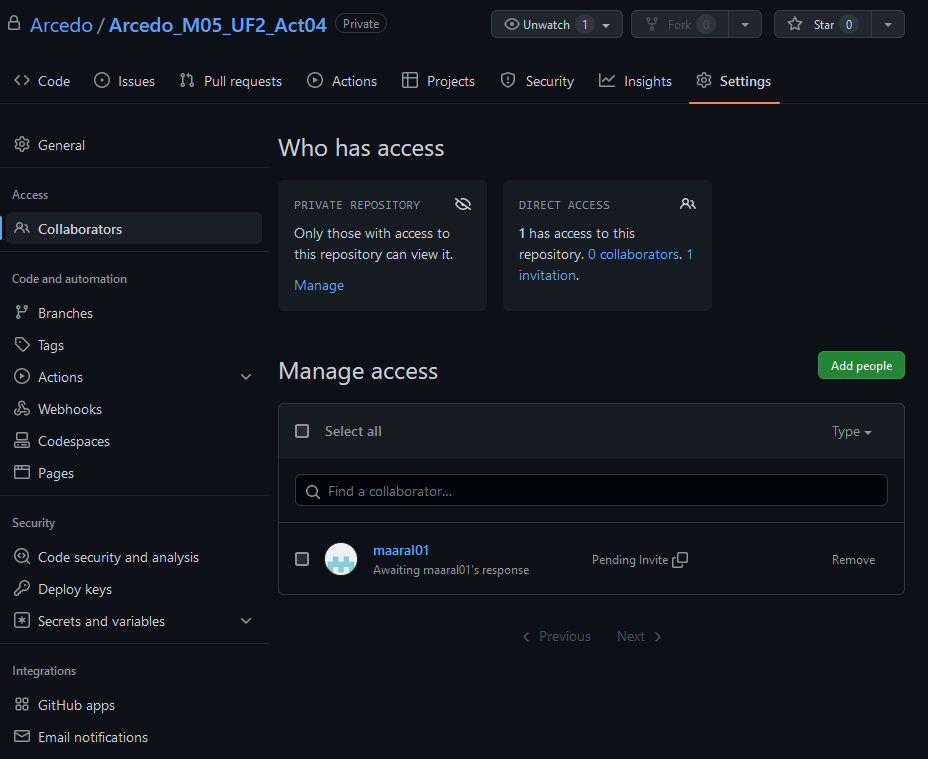


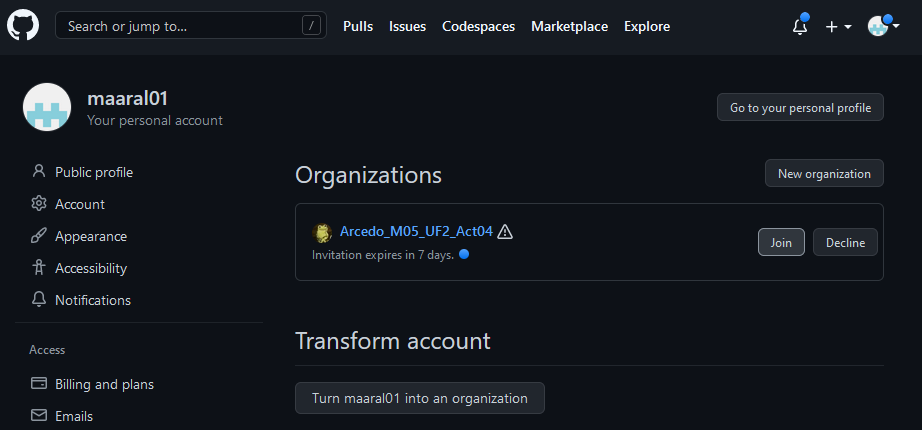


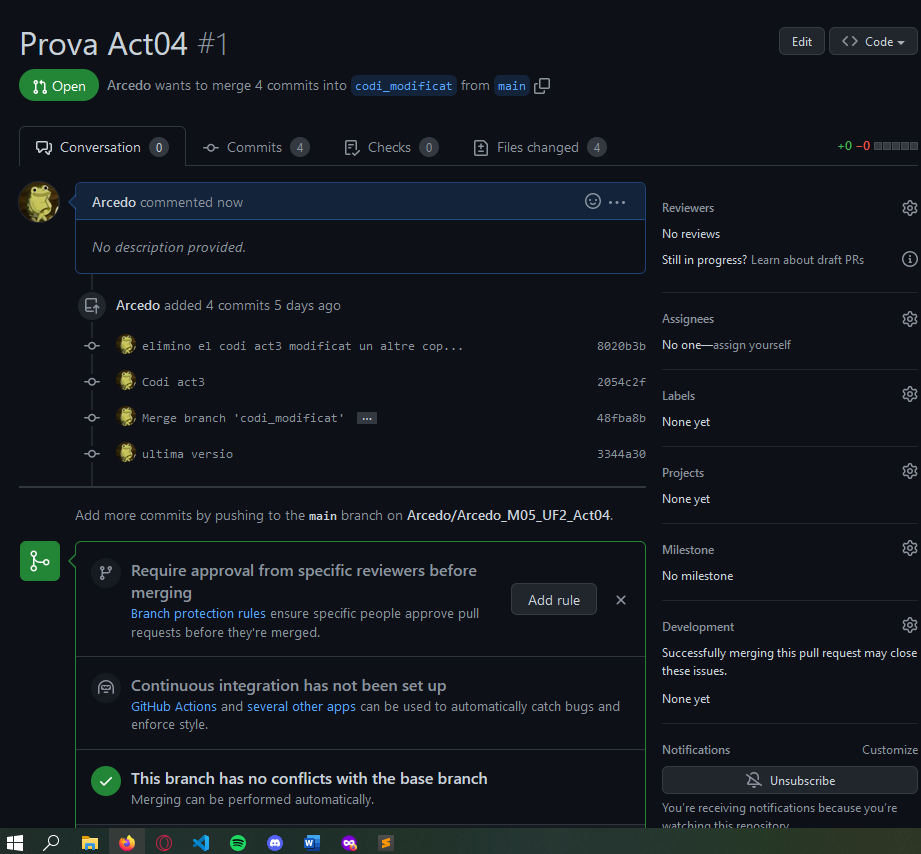


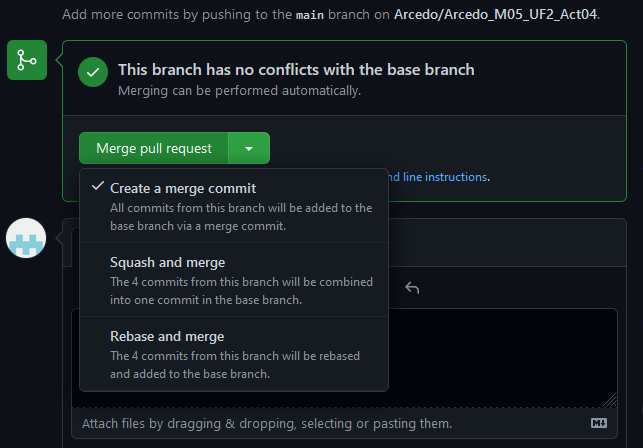












Explicar esto

Git init (inicia repositorio)

Git add . (subes todo lo modificado)

Git add arxiu.txt (solo sube el fitchero)

Git push (guardar global)

Git status (comprivar l’estat dels arxius escrits)

Git commit -m ‘mensaje del commit’ (para añadir los cambios que estan en espera al repositorio)

.gitignore (archivos que no quieras que se suban)

Git checkout arxiu.txt (para volver una verison atras/ borrar)

Git rm arxiu.txt (borrar archivo)

Git log (historial de commit)

Master /Main (rama principal)

Git checkout nombreramba (para cambiar de rama)

Git branch -d nombrerama (esborrar una rama)

Git merge nombrerama (fusionar dos ramas)

Pull (coje la ultima version en el repositorio)

Push(Subir el archivo editado)