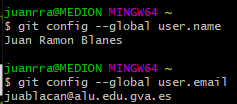
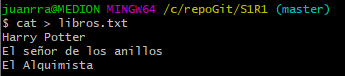
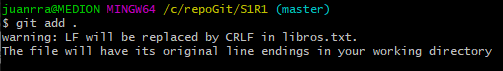
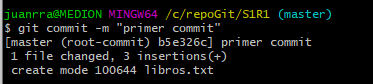
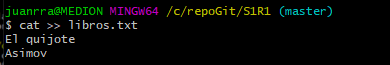
1.Instala Git

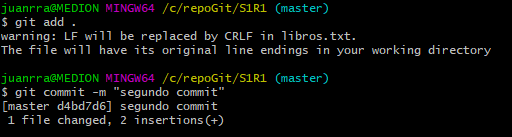


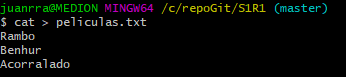
2. Configuración de git  


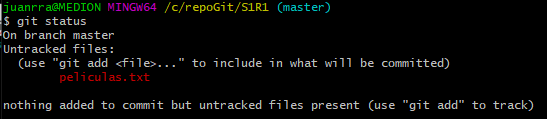
3.Crea una carpeta denominada S1R1  
  
3a) Crea un repositorio Git  
  
3b) crea un fichero libros.txt. Añade tres títulos de libros cada uno en una línea distinta.  


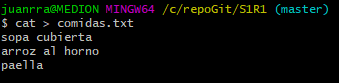
3c) Haz un primer commit.  
  


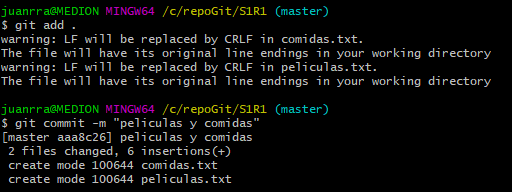
3d) Añade dos libros al archivo libros.txt  


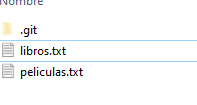
3e) Haz un segundo commit.  


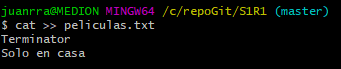
3f) Crea un fichero denominado peliculas.txt. Añade tres títulos de películas a dicho archivo.  


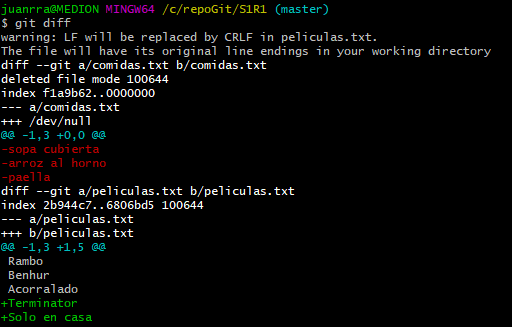
3g) captura de git status  


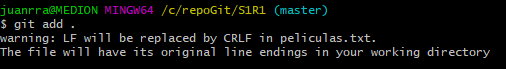
3h) Crea un fichero denominado comidas.txt. Añade tres nombres de comidas a dicho archivo.  


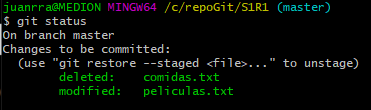
3i) Haz un tercer commit que incluya peliculas.txt y comidas.txt  


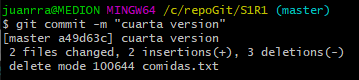
3j) Elimina el archivo comidas.txt desde el navegador de archivos  


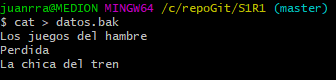
3k) Añade dos películas más al archivo peliculas.txt  


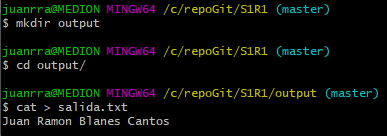
3l) Captura con los cambios en el directorio de trabajo  


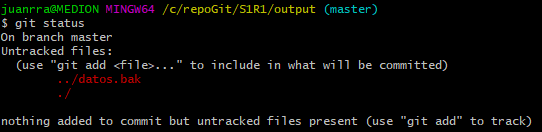
3m) Añade los cambios al área de preparación  


3n) Haz una captura de pantalla del comando git status. Debe indicar que se ha borrado el archivo comidas.txt y que se ha modificado el archivo peliculas.txt.  


3ñ) Haz un cuarto commit  


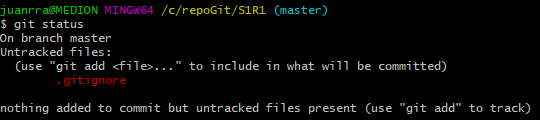
3o) Crea un archivo denominado datos.bak. Añade tres títulos de libros a dicho archivo. ¡IMPORTANTE! No añadas el archivo al área de preparación ni hagas ningún commit.  


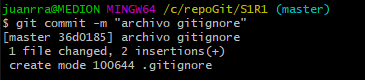
3p) Crea una subcarpeta denominada output. Crea un archivo denominado salida.txt en su interior. Escribe tu nombre y apellidos en dicho archivo. ¡IMPORTANTE! No añadas los archivos al área de preparación ni hagas ningún commit.  


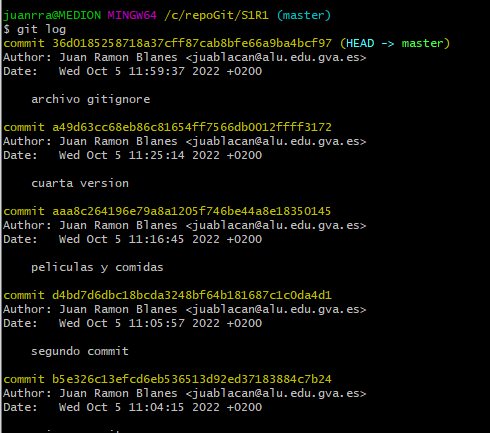
3q) Haz una captura de pantalla del comando git status.  


3r) Crea un archivo .gitignore para que los ficheros con extensión .bak y el contenido de la carpeta output/ no se incluyan en el repositorio.  


mv .gitignore ../

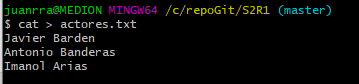
3s) Haz una nueva captura de pantalla del comando git status. Ahora no deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos, sino que en su lugar debe aparecer únicamente el archivo .gitignore.  


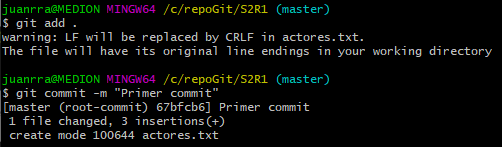
3t)Haz un último commit para incluir el archivo .gitignore en el repositorio  


3u) Haz una captura de pantalla que muestre el histórico de cambios del repositorio.  


3. Fundamentos de GIT II

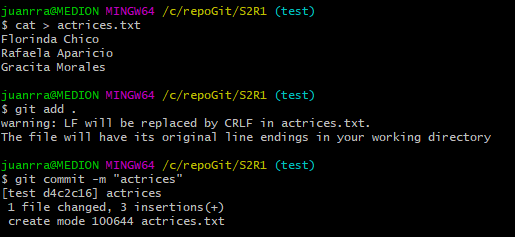
1. Crea una carpeta denominada S2R1. Realiza las siguientes acciones en ella:  
  
a) Crea un repositorio Git  

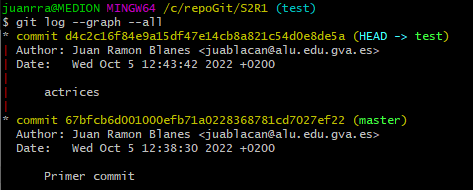

b) Crea un fichero denominado actores.txt. Añade tres nombres de actores cada uno en una línea distinta.  


c) Haz un primer commit.  


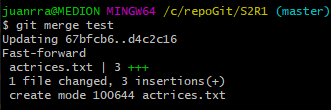
d) Crea una rama denominada test.  


e) Cambia a la rama test.  

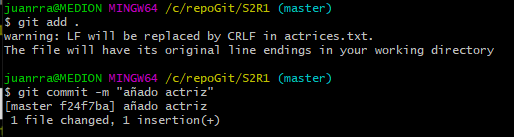

f) En la rama test crea un fichero denominado actrices.txt. Añade tres nombres de actrices y realiza un commit en dicha rama.  


g) Haz una captura de pantalla del resultado del comando git log --graph --all.  


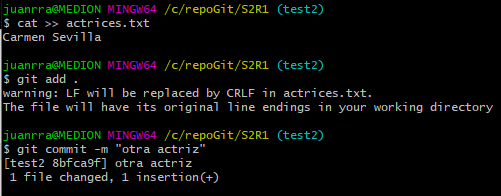
h) Cambia a la rama master.  

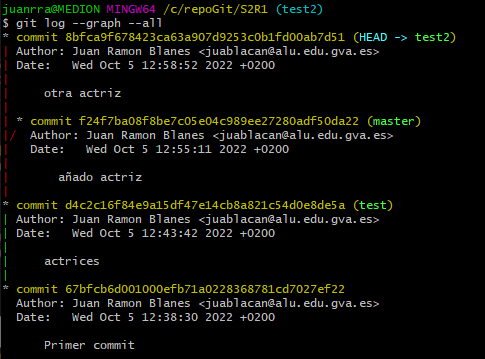

i) Incorpora los cambios de la rama test a la rama master. Haz una captura de pantalla de los comandos que has utilizado y de su resultado.  


j) Crea una segunda rama denominada test2. La rama test2 apunta al mismo commit que la rama master en éste momento.  

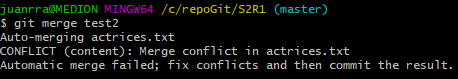
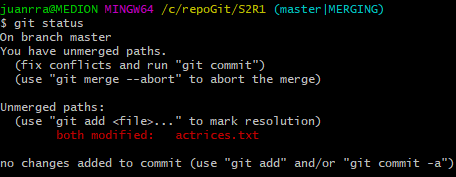

k) En la rama master, añade una actriz al fichero actrices.txt y haz un commit.  
  


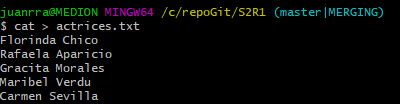
l) Cambia a la rama test2  

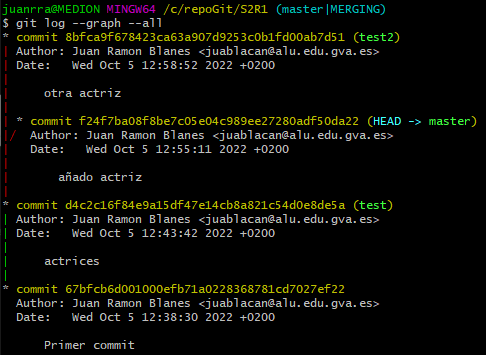

m) En la rama test2, añade una actriz al fichero actrices.txt y haz otro commit.  


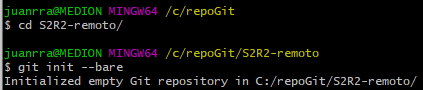
n) Haz una captura de pantalla del resultado del comando git log --graph --all. Debe haber dos caminos distintos: uno para la rama master y otro para la rama test2.  


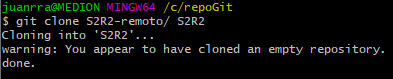
ñ) Cambiar a la rama master.  

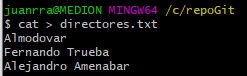

o) Incorpora los cambios de la rama test2 a la rama master. ¿Se produce un conflicto? De ser así realiza una captura del comando git status.  
  


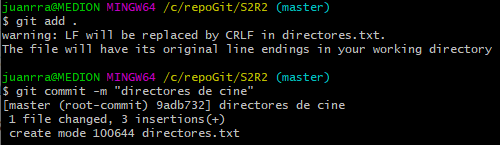
p) Resuelve el conflicto incorporando los 2 nombres de actrices.  


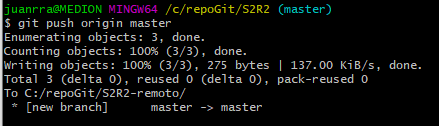
q) Haz una captura de pantalla del resultado del comando git log --graph --all. Observa que se ha creado un nuevo commit que integra los dos caminos anteriores.  


3.2 Crea una carpeta denominada S2R2-remoto. Inicializa un repositorio Git en su interior mediante el comando git init --bare. Esta carpeta se utilizará como repositorio remoto.  


3.3 Clona el repositorio S2R2-remoto en una carpeta denominada S2R2. Adjunta captura de pantalla del resultado del comando de clonado.  


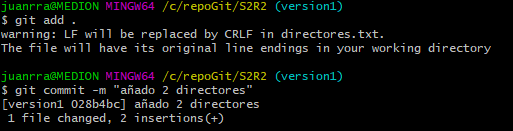
Sobre el directorio S2R2, haz lo siguiente:  
3.3a) Crea un archivo denominado directores.txt. Añade el nombre de tres directores de cine.  


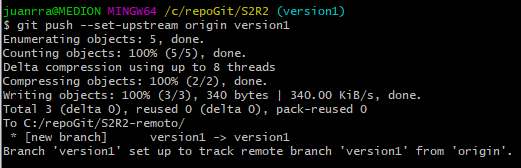
3.3b) Haz un commit  


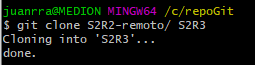
3.3c) Realiza un push al repositorio remoto. Adjunta captura de pantalla del resultado.  


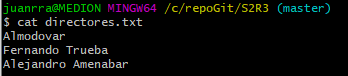
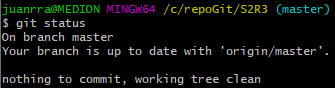
3.3d) Crea una rama denominada version1.  

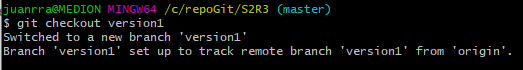

3.3e) Cambia a la rama version1.  

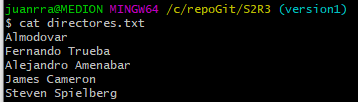

3.3f) En la rama version1 añade el nombre de dos directores de cine más al archivo directores.txt y haz un commit de los cambios.  


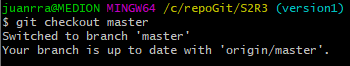
3.3g) Realiza un push de la rama al repositorio remoto de manera que quede asociada a la rama remota del mismo nombre. Adjunta captura de pantalla del resultado.  


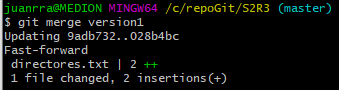
3.4. Clona el repositorio S2R2-remoto en una segunda carpeta denominada S2R3.   


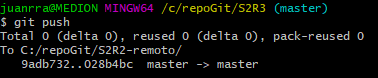
Realiza las siguientes acciones sobre ella:  
3.4a) Muestra en la consola el contenido del fichero directores.txt y el resultado del comando git status. Debe mostrar tres directores.  
  


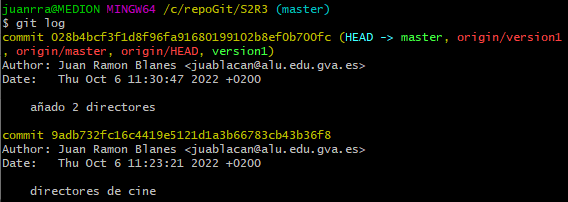
3.4b) Cambia a la rama version1. Muestra el resultado del comando. Comprueba que se crea una rama local version1 con el contenido de la rama remota origin/version1 y enlazada con ella. Al clonar el repositorio la rama no existía (solo se clona la rama principal, master), pero al cambiar a una rama que existe en el remoto se produce su creación local y enlazado con su correspondiente remota.  


3.4c) Muestra el contenido del fichero directores.txt por la pantalla. Comprueba que se muestran los 5 nombres de directores esperados. Adjunta captura de pantalla.  


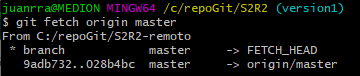
3.4d) Cambia a la rama master.  


3.4e) Incorpora los cambios de la rama version1 a la rama master.  


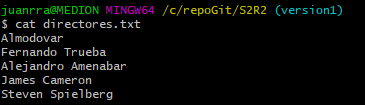
3.4f) Sube la rama master actualizada al servidor. Adjunta captura de pantalla del resultado del comando.  


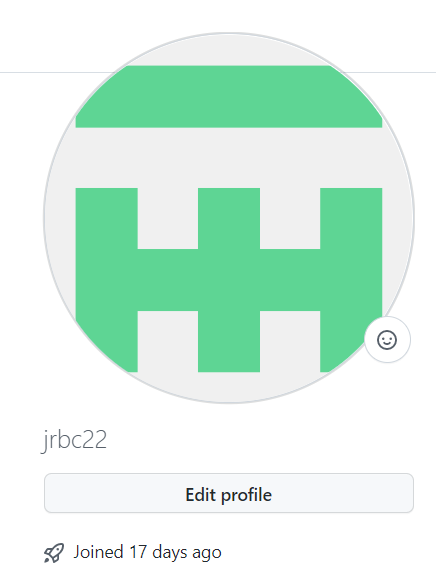


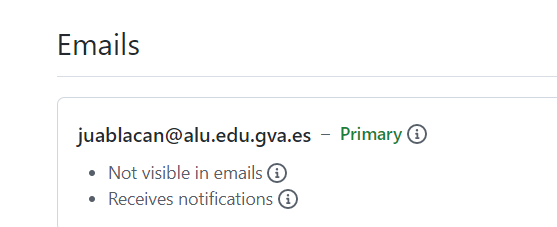
3.5. Vuelve de nuevo a la carpeta S2R2 y realiza las siguientes acciones:  

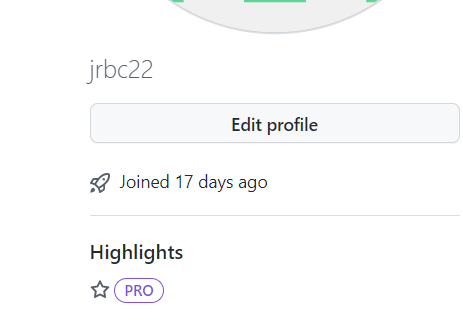

3.5a) Obtén los cambios que hay en el repositorio remoto sin fusionarlos en la rama local. Adjunta captura de pantalla del resultado del comando utilizado.  


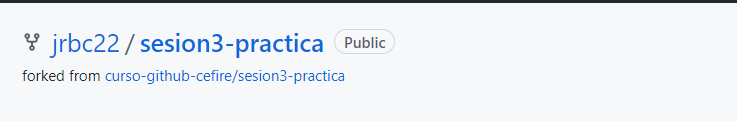
3.5b) Actualiza la rama master local con el contenido de la rama master del repositorio remoto. Adjunta captura de pantalla del resultado del comando utilizado.  

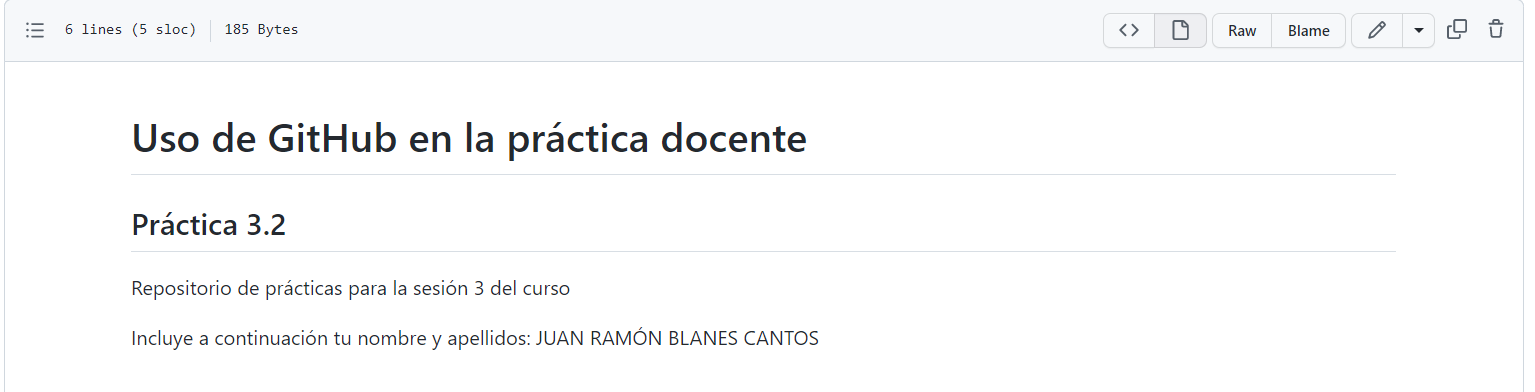

3.5c) Comprueba que aparecen los 5 nombres de directores esperados.  


4.1 Crea una cuenta en Github  


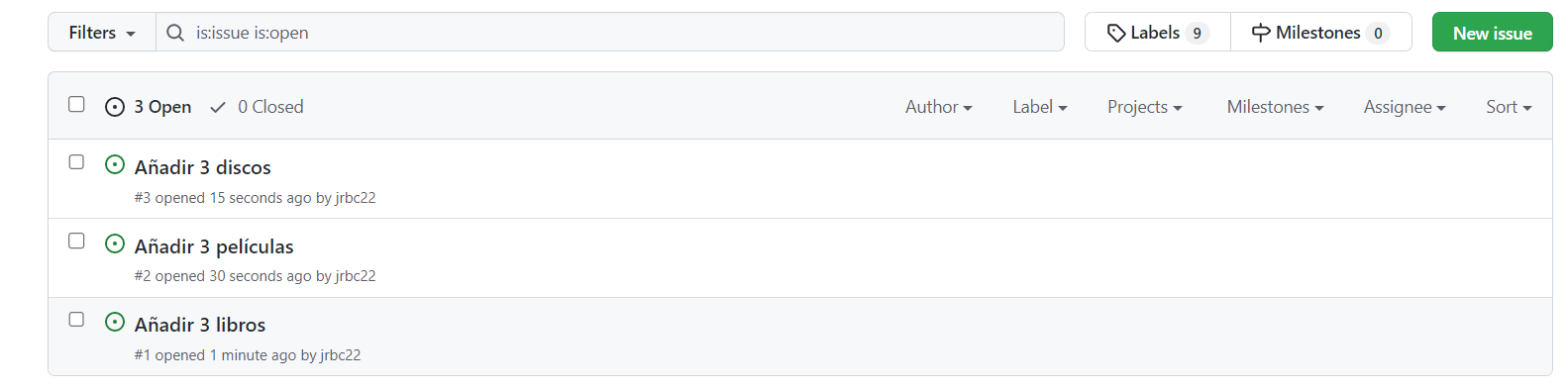
4.2 Añade tu dirección de correo de educación  


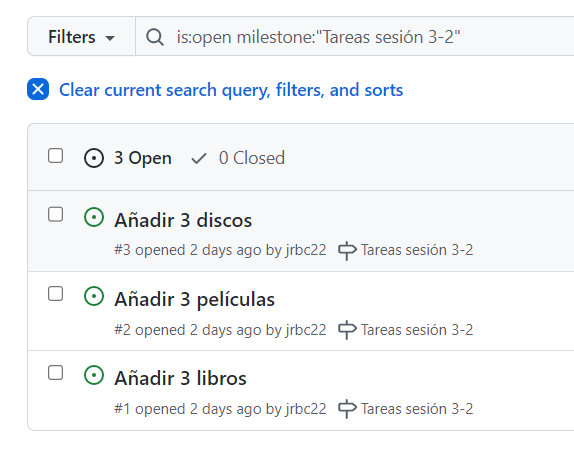
4.3 Solicita un descuento para uso educativo: <https://education.github.com/pack>  


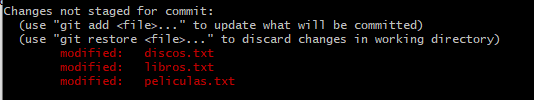
4.4 Haz un fork del repositorio localizado en la siguiente url: https://github.com/curso-github-cefire/sesion3-practica  


4.4a) Realiza un primer commit para poner tu nombre y apellidos en el fichero README.md  


4.4b) Crea 3 issues con los siguientes títulos: Añadir 3 libros, Añadir 3 películas, Añadir 3 discos.



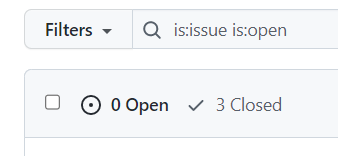
4.4c) Crea una milestone denominada Tareas sesión 3-2 que contenga los 3 issues creados.  


4.4d) Modifica los ficheros correspondientes y realiza 3 commits para realizar cada una de las tareas que se indican en los issues. El mensaje del commit debe hacer que se cierren los issues correspondientes de manera automática.  


Hay que hacer la modificación de cada archivo y después realizar:  
 + 

 + 

 + 

Una vez se han realizado esas operaciones ya vemos los issues cerrados.  
  


4.4e) Haz una captura de pantalla de los comandos