**Control de versiones con Git**

El objetivo de esta tarea será crea un proyecto web utilizando los lenguajes HTML y CSS para lo que se registrará la evolución del mismo con Git.

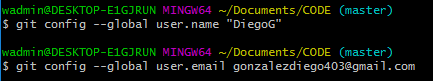
Todo el trabajo se desarrollará desde la línea de comandos, utilizando GitBash para la navegación y vi para la edición de los ficheros.

Completa los siguientes pasos y documenta el proceso con capturas de pantalla:

1. Ubícate en el directorio de trabajo que utilices para la asignatura. 
2. Crea una nueva carpeta con el nombre “bicicletas”.



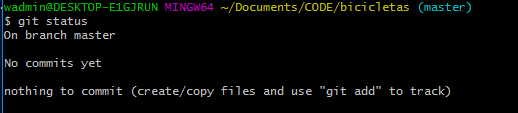
1. Realiza la configuración inicial de Git indicando tu usuario y dirección de correo electrónico.



1. Inicializa el repositorio local en la carpeta bicicletas con git init.



1. Muestra el estado de tu repositorio local. Explica la información presentada.



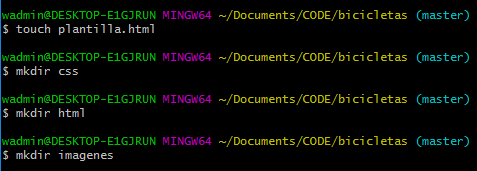
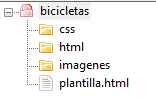
Información presentada:

-Está en la rama principal de archivos

-No hay ningún commit realizado

-No hay nada a lo que realizar un commit

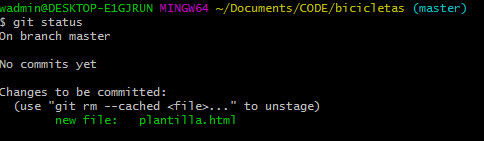
1. Crea un nuevo repositorio en tu cuenta de GitHub con el nombre “bicicletas”. Si no tienes cuenta en GitHub no es necesario que documentes el proceso de de creación de la misma. Solo la creación del repositorio.
2. Regresa a la línea de comandos para crear la siguiente estructura de carpetas. El fichero plantilla.html lo crearás vacío usando el comando *touch*.



1. Agrega los cambios que has realizado en tu directorio de trabajo al área de preparación.



1. Comprueba el estado de tu repositorio local. Explica la información presentada.



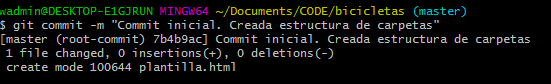
Información:

-Sigue en el repositorio principal

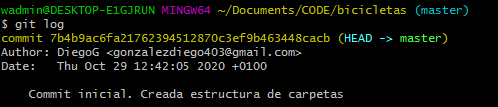
-No hay ningún commit

-Hay cambios en los que hay que realizar el “commit”

1. Confirma los cambios del área de preparación con un primer commit. Indica como mensaje “Commit incial. Creada estructura de carpetas”.



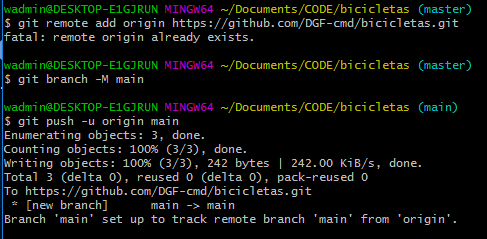
1. Comprueba el histórico de commits.



1. Agrega el repositorio que has creado en el paso 6 como repositorio remoto para tu proyecto local.



1. Sube los cambios de tu repositorio local al repositorio remoto y comprueba en la web de GitHub que figuran los cambios que has realizado en local.



1. Edita con vi el fichero plantilla.html para crear un esqueleto de documento HTML que solo contenga el siguiente código:

<!DOCTYPE html>

<html>

<head>

<title>**Plantilla del proyecto**</title>

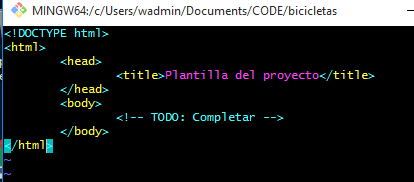
</head>

<body>

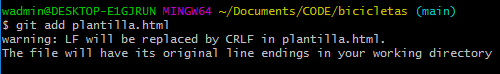
<!-- TODO: Completar -->

</body>

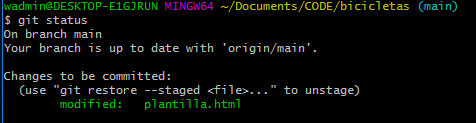
</html>



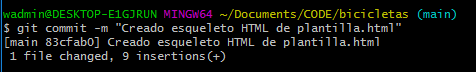
1. Agrega los cambios realizados en plantilla.html al área de preparación.



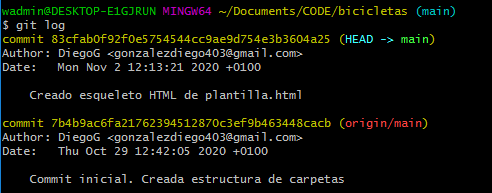
1. Comprueba el estado de tu repositorio local.



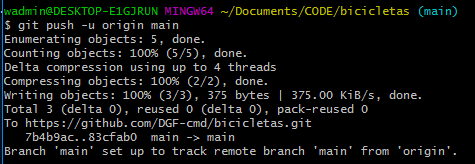
1. Confirma los cambios que se encuentran en el área de preparación con el mensaje “Creado esqueleto HTML de plantilla.html”



1. Comprueba el historial de commits de tu repositorio local.



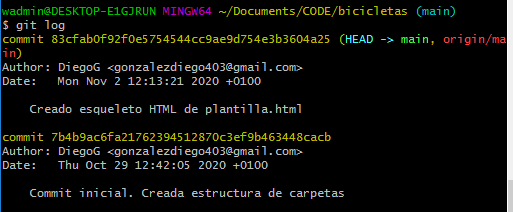
1. Envía los cambios de tu repositorio local al repositorio remoto en GitHub.



1. Comprueba el contenido de plantilla.html en el repositorio remoto de GitHub.

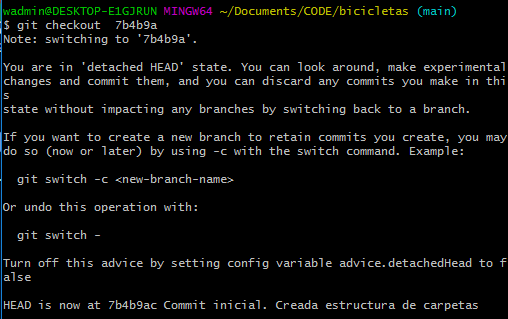


1. Muestra nuevamente el histórico de commits de tu repositorio local y anota los seis primeros caracteres del commit “Commit incial. Creada estructura de carpetas”.



Caracteres:7b4b9a

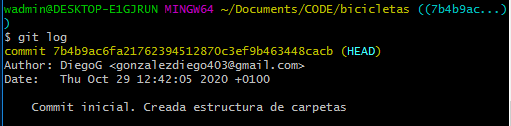
1. Ejecuta el comando:
   1. git checkout identificador\_del\_commit\_anotado



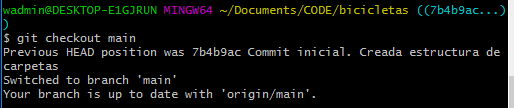
1. Comprueba el contenido del fichero “plantilla.html”. ¿Qué ha ocurrido?

No se muestra ningún contenido por estar en una versión anterior en el historial de commits ()

1. Muestra nuevamente el histórico de commits



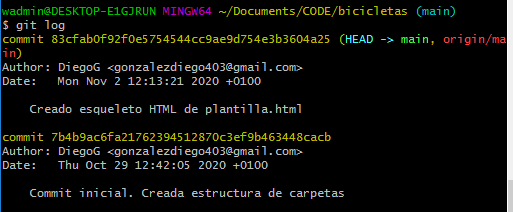
1. Ejecuta el comando:
   1. git checkout main



1. Comprueba el contenido del fichero “plantilla.html”. ¿Qué ha ocurrido?

Ha aparecido de nuevo el contenido porque ha vuelto a la última versión de commit

1. Muestra nuevamente el histórico de commits



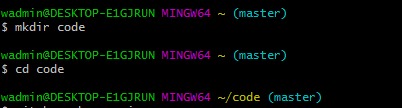
**Parte II**

Todo el trabajo se desarrollará desde la línea de comandos, utilizando GitBash para la navegación y vi para la edición de los ficheros desde la máquina indicada.

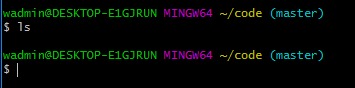
Completa los siguientes pasos y documenta el proceso con capturas de pantalla:

**Máquina Windows 10**

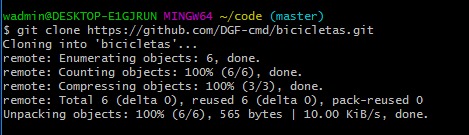
1. Muévete a la carpeta de trabajo CODE que vayas a utilizar en esta máquina virtual.



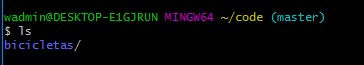
1. Haz un listado de su contenido.



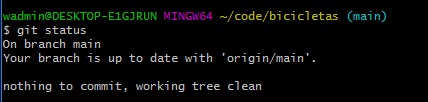
1. Utilizando la máquina virtual Windows 10, clona el proyecto “bicicletas” que has subido a tu cuenta de GitHub.



1. Realiza otro listado del contenido de la carpeta CODE y comprueba que se ha creado la carpeta correspondiente al proyecto “bicicletas”.



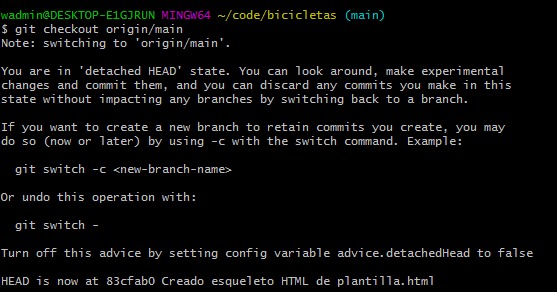
1. Muévete a la carpeta del proyecto y verifica que es un repositorio de Git ¿Cómo lo sabes?



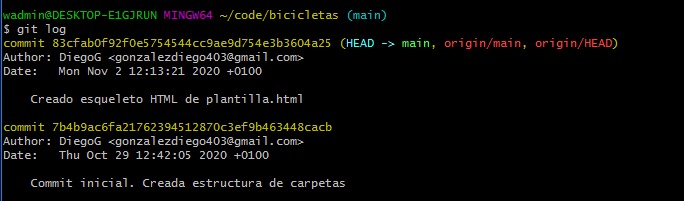
1. Comprueba la información del repositorio remoto “origin” en este proyecto.



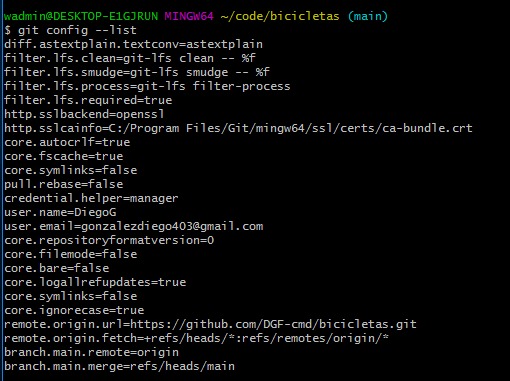
1. Comprueba el estado del repositorio local.



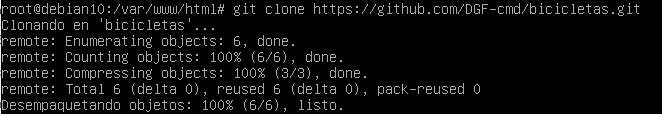
1. Comprueba el histórico de commits.

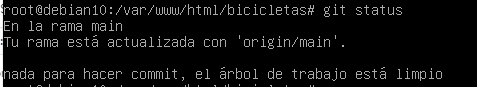


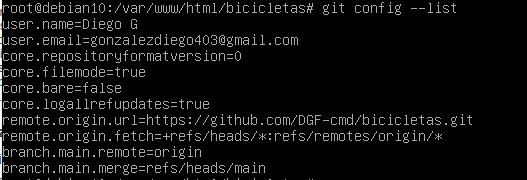
1. Comprueba la configuración de Git en esta máquina, y si no has registrado tu nombre de usuario y correo electrónico agrégalos.



**Máquina Debian**

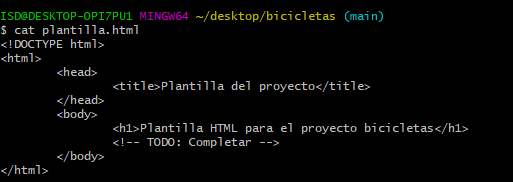
1. Descarga la máquina Debian con Git disponible en el enunciado de la tarea e impórtala en VirtualBox.
2. Haz login en la máquina con el usuario “root” y contraseña “abc123.”
3. Ubícate en el directorio /var/www/html
4. Haz un listado del contenido de este directorio.
5. Clona el proyecto bicicletas que has subido.
6. Lista de nuevo el contenido del directorio /var/www/html y comprueba que se ha creado la carpeta correspondiente al proyecto “bicicletas”.
7. Muévete a la carpeta del proyecto y verifica que es un repositorio de Git ¿Cómo lo sabes?
8. Comprueba la información del repositorio remoto “origin” en este proyecto.
9. Comprueba el estado del repositorio local.



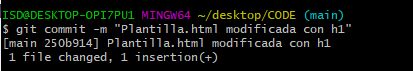
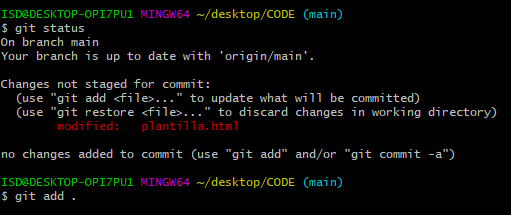
1. Comprueba el histórico de commits.
2. Comprueba la configuración de Git en esta máquina, y si no has registrado tu nombre de usuario y correo electrónico agrégalos.

**Anfitrión**

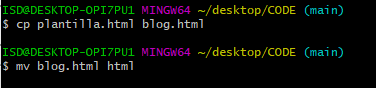
1. Edita el fichero “plantilla.html” y agrega el siguiente contenido dentro de la etiqueta body
   1. <h1>Plantilla HTML para el proyecto bicicletas</h1>



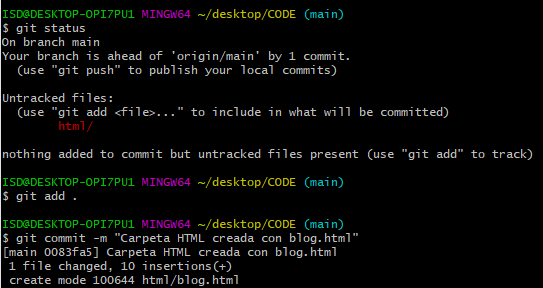
1. Comprueba el estado de tu repositorio local, agrega los cambios al área de preparación y confirma los cambios empleando un mensaje significativo.



1. Usando la línea de comandos copia el fichero “plantilla.html” al interior de la carpeta *html* con el nombre “blog.html”

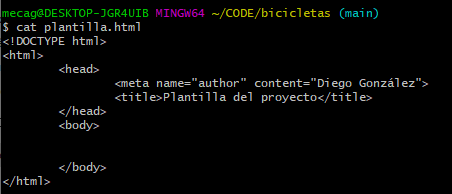


1. Comprueba el estado de tu repositorio local, agrega los cambios al área de preparación y confirma los cambios empleando un mensaje significativo.

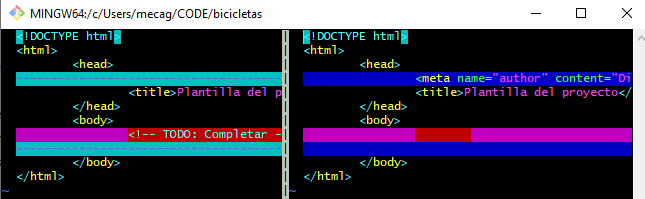


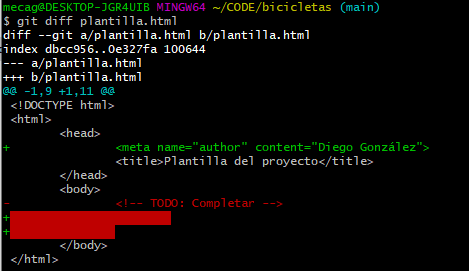
**Máquina Windows 10**

1. Modifica el fichero plantilla.html agregando la meta etiqueta autor con tu nombre.

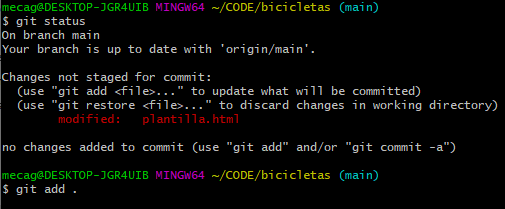


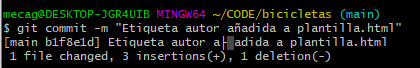
1. Comprueba las diferencias en los cambios que has introducido utilizando git diff y git difftool



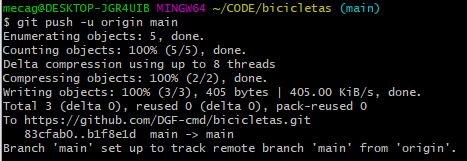


1. Comprueba el estado de tu repositorio local, agrega los cambios al área de preparación y confirma los cambios empleando un mensaje significativo.

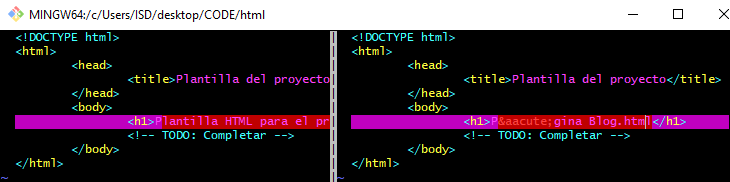




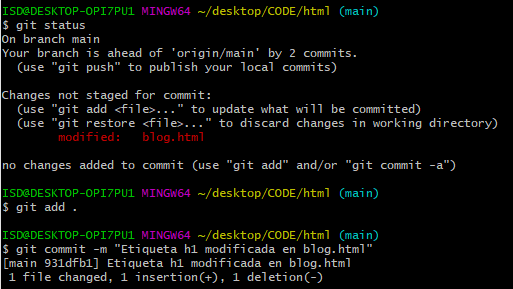
1. Envía los cambios al repositorio remoto

**Anfitrión**

1. Edita el fichero “blog.html” y sustituye el contenido de la etiqueta <h1> por el siguiente texto
   1. <h1>P&aacute;gina Blog.html</h1>
2. Revisa los cambios introducidos con git difftool.



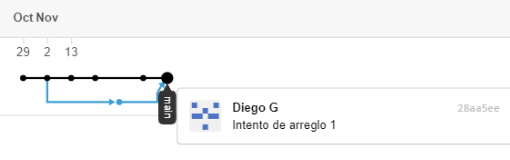
1. Comprueba el estado de tu repositorio local, agrega los cambios al área de preparación y confirma los cambios empleando un mensaje significativo.



1. Envía los cambios al repositorio remoto. Sigue las instrucciones proporcionadas por Git y resuelve el conflicto que se ha generado al intentar subir los cambios al repositorio remoto.
2. Una vez resuelto el conflicto, envía los cambios al repositorio remoto.



1. Desde el navegador revisa el histórico de commits de tu repositorio en GitHub navegando a *insights > network.*



**Máquina Debian**

1. Descarga los cambios del repositorio remoto.



1. Comprueba el histórico de commits.



**Anfitrión**

1. Desde el navegador escribe la dirección y comprueba que puedes ver el contenido de la página.

https://127.0.0.1:8080/biciletas/html/blog.html

