**GIT.**

1. **Un repositorio**es el lugar donde se almacenan los archivos del proyecto junto a todo el historial de cambios que **Git**gestiona.

A efectos prácticos no es más que la carpeta en la que vas a almacenar los archivos que quieres gestionar. Ten en cuenta que esa carpeta puede estar en tu ordenador o en un servidor remoto

Estados de archivos en Git

1. Modificado (modified): indica que has modificado el archivo, pero todavía no lo has confirmado a tu base de datos.
2. Preparado (staged): significa que has marcado un archivo modificado en su versión actual para que vaya en tu próxima confirmación.
3. Confirmado (commited): indica que los datos están almacenados de manera segura en tu base de datos local.

secciones principales:

1. Directorio de trabajo: no es más que una copia de una versión del proyecto. Estos archivos son extraídos de la base de datos del directorio .git y se colocan en la carpeta del proyecto para que puedas trabajar con ellos.
2. Área de preparación (Staging area): en sí es un fichero que indica que los archivos que irán en la siguiente confirmación.
3. Directorio .git: es donde se almacena la base de datos con toda la información que Git necesita para gestionar el proyecto.

flujo de trabajo en Git

1. Modificar una serie de archivos en el directorio de trabajo.
2. Añadir los archivos modificados al área de preparación.
3. Confirmar los cambios. Esto toma los archivos tal y como están en tu área de preparación y los almacena en la base de datos del directorio .git.

Diagrama

Descripción generada automáticamente

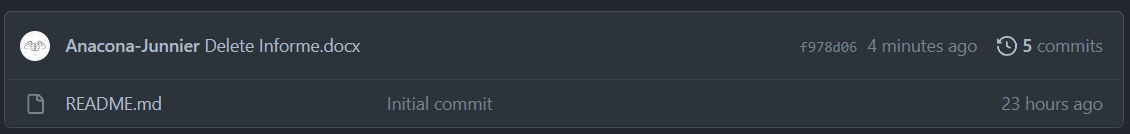
1. Creación del repositorio en GitHub.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente



1. Clonar repositorio.

Se crea una carpeta nueva en Windows 10 con el nombre “Git proyectos” el cual se utilizará para almacenar los datos del repositorio y sus cambios.

Imagen que contiene firmar, calle, señal, palo

Descripción generada automáticamente

Sobre la misma carpeta nueva se selecciona y se presiona clic derecho para visualizar las opciones, seleccionamos Git Bash Here para abrir la interfaz de línea de comandos sobre la carpeta.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se lanza una ventana de comandos donde se podrá ejecutar los comandos de git para clonar, actualizar, conocer el estado de los archivos, etc.



Para clonar el repositorio anteriormente creado en GitHub es necesario obtener la URL del repositorio, por ello, en las opciones suministradas en la plataforma de GitHub hay una opción “Code” que nos dará a conocer la URL del repositorio.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se ejecuta el siguiente comando para clonar el repositorio:

Texto

Descripción generada automáticamente

Se lista el contenido y se observa que ahora hay una nueva carpeta con el mismo nombre del repositorio:



Se accede a la carpeta Git-Sistemas-Telematicos-3 para listar su contenido:



A la hora de listar el contenido se observa un archivo README el cual esta vacío que genera automáticamente GitHub a la hora de crear un repositorio en su plataforma.



Si se accede a esa carpeta mediante la interfaz grafica suministrada por el SO se puede apreciar que la carpeta contiene un directorio .git (donde está la base de datos del repositorio) y el área de trabajo con los archivos de la última versión del proyecto.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Creación de archivos con los nombres general.txt, Jessica.docx, Marlon.docx, Junnier.docx haciendo uso de las herramientas de Windows 10 para crear archivos.

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente



1. Para enviar los cambios (creación de archivos) a la nube se ejecuta en orden los siguientes comandos.
2. Conocer el estado de los archivos mediante el comando *git status.*

Texto

Descripción generada automáticamente

Como puedes ver, el archivo general.txt Jessica.docx Marlon.docx Junnier.docx aparece bajo la cabecera “**Archivos sin seguimiento**” (en inglés Untracked files). Bajo esta cabecera solo aparecen archivos nuevos y Git no los incluirá en el próximo **commit** a no ser que se le indique expresamente.

1. Para que Git inicie un seguimiento de los archivos se utiliza el comando *git add nombre\_archivos*.



Se vuelve a revisar el estado de los archivos. Se observa que ahora los archivos están siendo rastreados y están listos para ser confirmados en el siguiente commit.

Texto

Descripción generada automáticamente

1. Confirmar cambios. Una vez se hayan realizado todos los cambios deseados se confirman con el comando *git commit.*

Este comando ejecutará el editor de texto con una línea en blanco para escribir un mensaje que describa la confirmación que se esta realizando.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Para guardar cambios en el editar de texto ***nano*** se presiona la combinación de teclas Ctrl+X y confirmamos.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Texto, Chat o mensaje de texto

Descripción generada automáticamente

1. Subir los cambios locales a un servidor remoto se utiliza el comando ***git push*** que se estructura de la siguiente manera:

* **git push *nombre\_remoto nombre\_rama***

Donde:

* nombre\_remoto: es el servidor a actualizar, se puede utilizar la **URL** o el termino **origin.**
* nombre\_rama: es la rama a actualizar.



o



**Nota:** Es posible que se les solicite iniciar sesión con su cuenta de GitHub para subir la actualización al repositorio.

Texto

Descripción generada automáticamente

Ahora bien, si se accede al repositorio de GitHub anteriormente creado se podrá visualizar los nuevos archivos subidos en ella.

Captura de pantalla de un celular

Descripción generada automáticamente

Finalmente, se expone todos los pasos anteriormente abarcados en una sola imagen.

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Archivo editado por Junnier en la rama Junnier**