1- ¿Qué es GitHub?

Git Hub es una plataforma en línea para el desarrollo y colaboración de software basada en Git. Permite a los programadores alojar, gestionar y compartir código, facilitando el trabajo en equipo mediante control de versiones, seguimiento de cambios y herramientas de integración continua.

2- ¿Cómo crear un repositorio en GitHub?

Para crear un repositorio hay que iniciar sesión en GitHub, buscar la opción “New repository”, ingresar, y en esa pestaña nos permitirá ponerle un nombre, descripción y visibilidad al repositorio, luego de haber completado los campos, al pie de página, clic a “Create repository”.

3- ¿Cómo crear una rama en Git?

Para crear una rama hay que abrir la terminal o consola en el proyecto, usar el comando:

“ git branch <nombre de la rama> ”

4- ¿Cómo cambiar a una rama en Git?

Para cambiar a una rama de Git : git checkout <nombre de la rama>.

5- ¿Cómo fusionar ramas en Git?

Para fusionar ramas en Git: git merge <nombre de la rama>

6- • ¿Cómo crear un commit en Git?

Para crear un commit, primeto hay que verificar si estamos en el repositorio correcto, luego:

- Git add . = para agregar todos los archivos modificados
- Git add nombre_del_archivo = para agregar un archivo especifico

7- • ¿Cómo enviar un commit a GitHub?

Para el envio de un commit es el siguiente comando : git commit -m <mensaje>

8- • ¿Qué es un repositorio remoto?

Un repositorio remoto es una versión de mi repositorio que esta alojada en un servidor en línea, como GirHub, GitLab , etc.

9- • ¿Cómo agregar un repositorio remoto a Git?

Para agregar un repositorio remoto mediante comando : git init

10- • ¿Cómo empujar cambios a un repositorio remoto?

Para empujar cambios a un repositorio primer hay que ingresar el comado: “git add .” , luego crear un commit con un msj descriptivo : “git commit -m “mensaje” ”, por ultimo para manarlo el Git Hub : “git push -u origin master o main “

11- • ¿Cómo tirar de cambios de un repositorio remoto?

Hay que ejecutar este comando para descargar y fusionar los cambios de la rama remota en la rama local: git pull origin <nombre de la rama>

Si se trabaja en la Master: git pull origin master.

12- • ¿Qué es un fork de repositorio?

Es una copia de un repositorio de GitHub que te permite hacer cambios y experimentar en mi propio espacio sin afectar el repositorio original

13- • ¿Cómo crear un fork de un repositorio?

Para crear un Fork, es ir al repositorio original en GitHub, hacer clic en botón “Fork”, y el repositorio de copiara en mi cuenta de GitHub y ahora puedo trabajar en el repositorio como si fuese mio, ej: El Fork de la universidad

14- • ¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?

Primero hay que ir a la pagina principal del repositorio, seleccionar la rama que contiene los cambios, hacer clic en “Compare & Pull request”, seleccionar la rama base y la rama de comparación, escribir un titulo y descripción. Por ultimo hacer clic en “Crate Pull request”.

15- ¿Cómo aceptar una solicitud de extracción?

Abrir la solicitud de extracción, revisar los cambios propuestos. Hacer clic en “revisar cambios” en la pestaña “Archivos modificados”, “seleccionar aprobar”, por último enviar la revisión.

16 - ¿Qué es una etiqueta en Git?

Una etiqueta es un marcador que se utiliza para señalar puntos específicos en la historia del proyecto. Generalmente, las etiquetas se utilizan para marcas versiones importantes, como un lanzamiento de software, de manera que pueda fácilmente volver a ese punto en el historial del repositorio en el futuro.

17- ¿Cómo crear una etiqueta en Git?

Hay 2 tipos de etiquetas: Etiqueta ligera (lightweight tag) y etiqueta anotada (annotated tag)

Etiqueta Ligera es simplemente un puntero a un commit especifico, es mas un marcador sin información adicional, como autor o el mensaje.

Ej= git tag “nombre_de_etiqueta”

La etiqueta apuntara al commit actual (el ultimo commit en tu rama actual)

Etiqueta anotada: Es más completa, en este incluye detalles como el autor, le fecha y un mensaje. Es la opción más remendada si se quiere almacenar más información sobre la versión.

Ej= git tag -a <nombre de la etiqueta> -m “Mensaje”

18 - ¿Cómo enviar una etiqueta a GitHub?

Después de haber creado las etiquetas, para poder subirlas al repositorio remoto para que otros las vean:

Para subir una etiqueta especifica: git push origin <nombre de la etiqueta>

Para subir todas las etiquetas de una vez: git push origin --tags

19 - ¿Qué es un historial de Git?

EL historial en git hace referencia a las secuencias de commit que han sido realizados en el repositorio a lo largo del tiempo. Cada commit es un conjunto de cambios que se registran de manera permanente en el repositorio, junto con metadatos como el autor , la fecha y un mensaje descriptivo del cambio realizado.

• ¿Cómo ver el historial de Git?

Para ver el historial hay diferentes comandos para los cuales son:

Git log = Este comando te permite ver el historial de commit de tu repositorio, muestra una lista de commits junto con la información como el hash del commit, el autor la fecha y el mensaje.

Git log - -oneline : Para ver el historial de manera más compacta, puede usar esta variante, que mostrara cada commit en una sola línea con el hash abreviado y el mensaje del commit.

Git Log - -grap : Este comando nos muestra una representación gráfica del historial del commit, que es útil para ver como se ramifica y fusión las ramas

Git show <commit-hash> : Si se queire ver mas detalles sobre un commit especifico, como los cambios que introdujo, se puede usar el hash del commit con este comando, git show <commit – hash>

• ¿Cómo buscar en el historial de Git?

Para buscar en el historial de Git hay distintos comandos y pociones para poder filtrar los resultado según que se nesecite:

Busqueda por autor: git log –author = “nombre”

Busqueda por fecha: git log -- sinse=” fecha” y git log -- until=” fecha”.

Busqueda por mensaje de commit: git log --grep="palabra"

• ¿Cómo borrar el historial de Git?

Para borrar el historial y poder comenzar de cero o eliminar ciertos commits, hay varias formas de poder hacerlo, dependiendo de lo que se necesite exactamente:

1- Git reset -- hard: Borra el historial localmente y restablece el repositorio aun commit anterior.
2- Git push – force: Sobrescribe el historial en el repositorio remoto después de modifícalo localmente. (pd: Borre una carpeta sin quererlo en mi repositorio de Git Hub, y puede reestablecerlo desde el repositorio local las mismas carpetas antes de que sea borrado).
3- Rm -rf -.git: Eliminar completamente el historial.
4- Git fliter-branch o git filter-repo: Elimina archivos o cambios específicos del historial

• ¿Qué es un repositorio privado en GitHub?

El repositorio privado es un repositorio de código fuente y archivos que están protegidos por restricciones de acceso. Donde solo las personas que tienen permiso explicito para acceder al repositorio donde pueden verlo, clonar o modificar su contenido. Todo lo contrario al repositorio Publico donde cualquier persona con internet puede acceder.

• ¿Cómo crear un repositorio privado en GitHub?

- Hay que iniciar sesión en la pagina de GitHub, clic en “new” para crear un repositorio.
- Llenas el formulario con nombre, descripción y configuración.
- En la opción “Visibility”, seleccionar “Private” para que el repositorio sea privado, y paso final, hacer clic en “Create repository”

• ¿Cómo invitar a alguien a un repositorio privado en GitHub?

- Invitar a otras personas hay que ir a mi repositorio, clic en “settigs”.
- Menu izquierda, seleccionas “Manage Access”
- Clic en “Invite a collaborator”, escribir el nombre de usuario que quiera invitar.
- Seleccionar el nivel de acceso(Lectura, escritura, administración), y como paso final, clic en “add collaborator”.

• ¿Qué es un repositorio público en GitHub?

Un repositorio público es un espacio virtual donde se almacena código o archivos relacionados con un proyecto, esta disponible para que cualquier persona pueda verlo, descargarlo o copiarlo libremente desde internet

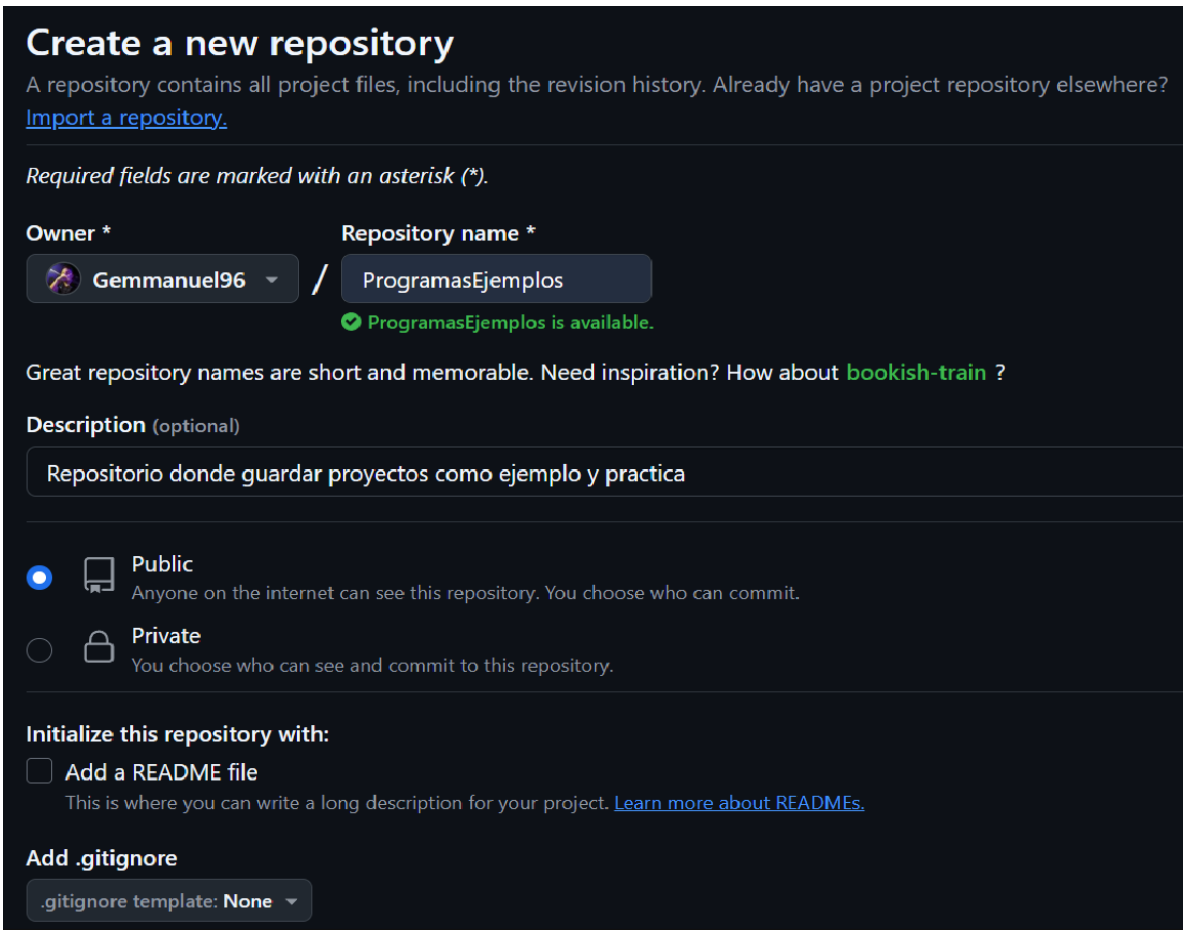
• ¿Cómo crear un repositorio público en GitHub?

- Primero ir a la pagina de GitHub, ir y hacer clic en botón “new” para crear un repositorio
- Rellenar el formulario con nombre, descripción de repositorio
- Seleccionar “Public”, para que cualquiera pueda ver el repositorio
- Como paso final, hacer clic en “Create Repository”

• ¿Cómo compartir un repositorio público en GitHub?

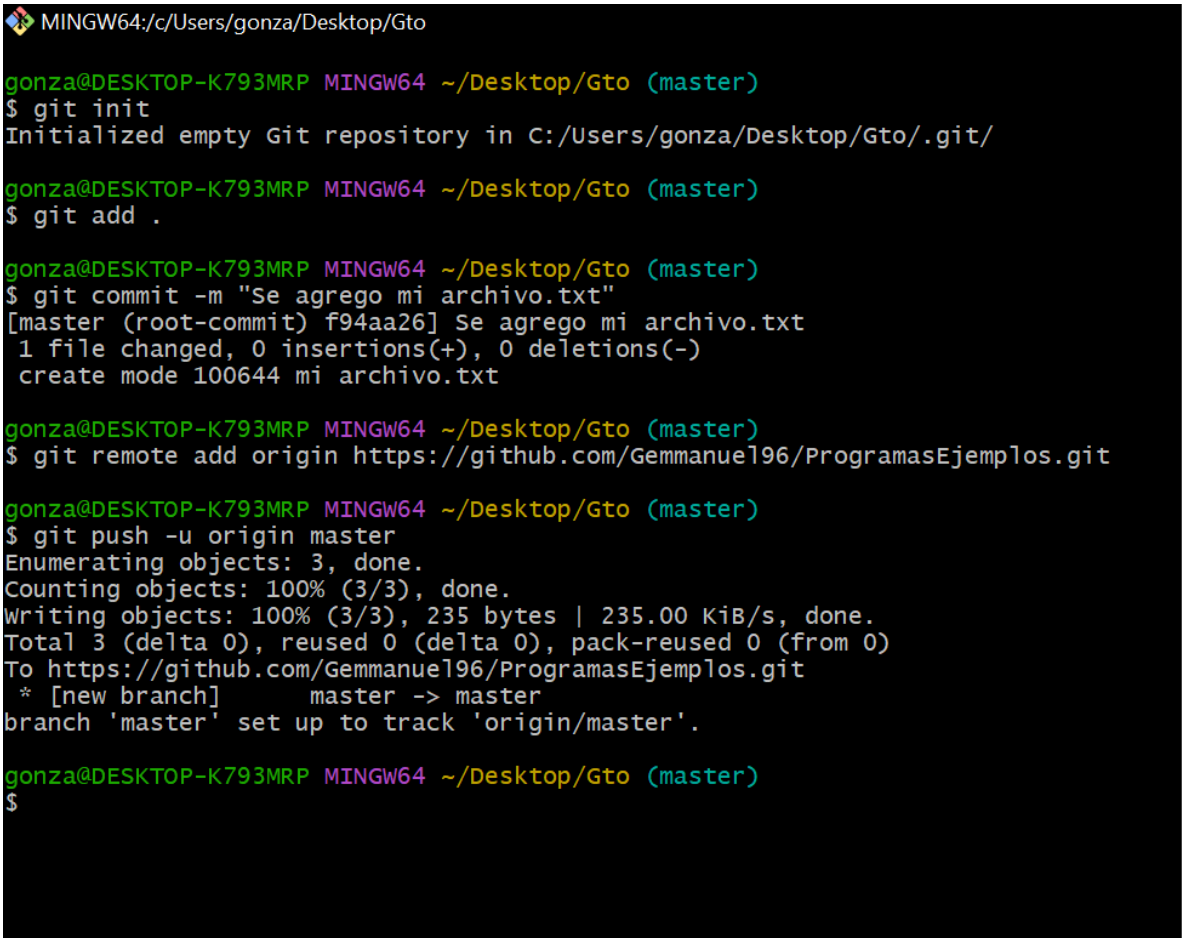
2) Realizar la siguiente actividad:

• Crear un repositorio. o Dale un nombre al repositorio.o Elije el repositorio sea público. o Inicializa el repositorio con un archivo.



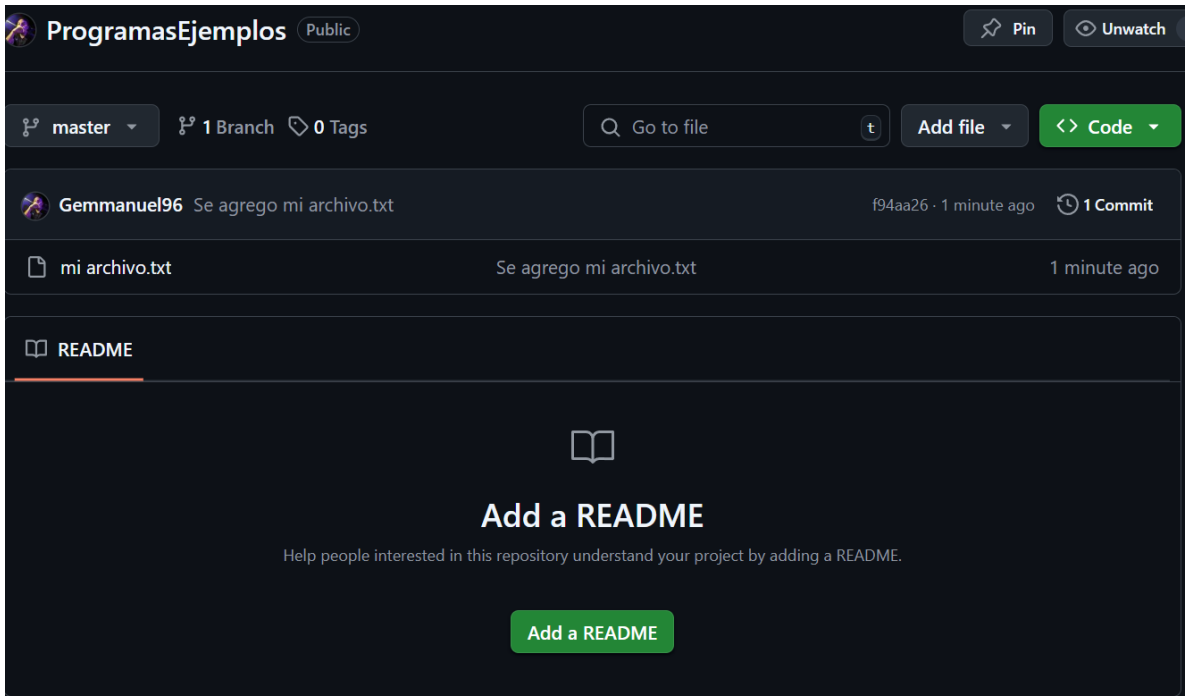
• Agregando un Archivo o Crea un archivo simple, por ejemplo, "mi-archivo.txt".

o Realiza los comandos git add . y git commit -m "Agregando mi-archivo.txt" en la línea de comandos.



o Sube los cambios al repositorio en GitHub con git push origin main (o el nombre de la rama correspondiente).

Como queda en GitHub:


ProgramasEjemplos
Public
Pin
Unwatch
master
1 Branch
0 Tags
Go to file
t
Add file
Code
Gemmanuel96 Se agrego mi archivo.txt
f94aa26 · 1 minute ago
1 Commit
mi archivo.txt
Se agrego mi archivo.txt
1 minute ago
README
Add a README
Help people interested in this repository understand your project by adding a README.
Add a README