

GitHub: Introducción

GitHub es un sistema de control de versiones que se utiliza como ayuda a la hora de desarrollar software. Lo primero es crearse una cuenta en la página web de [GitHub](#).

Paso 1: Crear una cuenta y un repositorio en GitHub

1. **Crear una cuenta en GitHub:** Si aún no tienes una cuenta, ve a [GitHub](#) y regístrate.
2. **Crear un nuevo repositorio:**
 - Después de iniciar sesión, en la página principal de GitHub:
 - Si acabas de crearte la cuenta, haz clic en **"Create repository"**.
 - Si no es así, haz clic en el botón **"New"** (Nuevo) que aparece en la esquina superior derecha de tu pantalla, o en tu perfil, busca la opción de **"Repositories"** y selecciona **"New"**.
 - En la página que se abre, rellena los campos:
 - **Repository name:** Pon un nombre para tu repositorio (ej. `pruebaGitHub`).
 - **Description:** Puedes poner una descripción breve (opcional).
 - Elige si quieres que el repositorio sea **público** o **privado**. En nuestro caso, **público**.
 - Marca la opción **"Initialize this repository with a README"** para crear un archivo README inicial (esto es opcional, pero en nuestro caso lo marcaremos).
 - Haz clic en el botón **"Create repository"**.

Paso 2: Subir archivos a tu repositorio

Una vez creado el repositorio, puedes comenzar a subir archivos de manera fácil desde la página web. Aquí te muestro cómo hacerlo:

1. **Ir al repositorio:**
 - Después de crear el repositorio, serás redirigido a la página principal de ese repositorio.

2. Subir archivos:

- En la página del repositorio, haz clic en el botón **"Add file"** que se encuentra justo encima de la lista de archivos, y selecciona **"Upload files"**.

3. Subir el archivo de tu computadora:

- Arrastra y suelta el archivo o haz clic en el área de arrastre para seleccionar los archivos desde tu computadora.
- Si ya tienes un archivo en tu computadora, por ejemplo `prueba.c`, selecciona el archivo y súbelo.

4. Confirmar los cambios:

- Una vez subido el archivo, verás una pantalla donde puedes añadir un mensaje de confirmación (similar al `git commit`).
- En el campo **"Commit changes"**, escribe un mensaje que describa los cambios que estás haciendo, por ejemplo, **"Subiendo archivo de prueba"**.
- Asegúrate de dejar seleccionada la opción **"Commit directly to the main branch"** si quieres que los cambios se realicen directamente en la rama principal (`main`).
- Haz clic en **"Commit changes"** para completar el proceso.

Paso 3: Ver tus archivos y commits

1. Ver archivos:

- Una vez hayas subido tus archivos, aparecerán en el repositorio. Puedes ver el archivo `prueba.c` y hacer clic sobre él para ver su contenido.

2. Ver historial de commits:

- Para ver el historial de cambios (como los commits), haz clic en la pestaña **"Commits"** que está justo encima de la lista de archivos.
- Ahí verás una lista de todos los commits, junto con los mensajes de commit y quién los realizó.

Paso 4: Crear y administrar ramas (Branches)

Si deseas trabajar en diferentes versiones de tu proyecto sin afectar la rama principal (**main**), puedes crear nuevas ramas (branches):

1. Crear una nueva rama:

- En la página principal de tu repositorio, haz clic en el menú desplegable **"main"** (si es que estás en la rama principal).
- Luego, escribe el nombre de la nueva rama y haz clic en **"Create branch"**.
- Esto te llevará automáticamente a esa nueva rama.

2. Hacer cambios en una rama:

- Desde esa rama, puedes seguir agregando o modificando archivos de la misma manera que lo hicimos en el paso anterior. Cualquier cambio hecho en esta rama no afectará a la rama **main** hasta que decidas fusionarlas.

3. Fusionar ramas (Merge):

- Si deseas fusionar los cambios de una rama con la rama principal (**main**), debes hacer un **pull request**:
 - Ve a la pestaña **"Pull requests"** en tu repositorio.
 - Haz clic en **"New pull request"**.
 - Selecciona las ramas que deseas fusionar y haz clic en **"Create pull request"**.
 - Después de revisar, puedes hacer clic en **"Merge pull request"** para combinar los cambios de la rama en la rama principal.

Paso 5: Gestionar configuración y permisos

GitHub te permite configurar diferentes aspectos del repositorio, como los colaboradores y la visibilidad.

1. Configuración del repositorio:

- Haz clic en el botón **"Settings"** en la parte superior del repositorio para cambiar la configuración (como nombre, descripción, visibilidad, etc.).

- También puedes añadir colaboradores para que otras personas puedan trabajar en el repositorio contigo.

2. Permisos y colaboraciones:

- En la sección **"Collaborators"**, puedes agregar personas para que trabajen en tu repositorio. Ellos podrán tener acceso para hacer cambios, dependiendo de los permisos que les des.

Resumen visual de las acciones en GitHub

| Acción | Descripción |
|----------------------------------|--|
| Crear un repositorio | Crea un nuevo repositorio desde la página de GitHub. |
| Subir archivos | Usa el botón "Add file" para cargar archivos desde tu computadora. |
| Hacer commits | Confirma los cambios escribiendo un mensaje de commit. |
| Crear ramas (branches) | Crea nuevas ramas desde la interfaz web de GitHub. |
| Fusión de ramas (Merge) | Realiza un pull request para fusionar ramas. |
| Administrar colaboradores | Añade colaboradores desde la pestaña de "Settings". |

Conclusión:

GitHub te permite realizar todas estas operaciones de Git de manera gráfica, sin necesidad de usar la terminal. Puedes subir archivos, hacer commits, gestionar ramas, fusionarlas, y colaborar con otros directamente desde la interfaz web. Todo esto lo hace accesible incluso para aquellos que no están familiarizados con la línea de comandos.

Si te gustaría ver cómo hacer algo más en GitHub o necesitas más detalles, no dudes en preguntar.

Criterios de evaluación

- e) Se ha instalado, configurado y utilizado un sistema de control de versiones.
- f) Se ha garantizado la accesibilidad y seguridad de la documentación almacenada por el sistema de control de versiones.
- g) Se ha documentado la instalación, configuración y uso del sistema de control de versiones utilizado.