

COMUNICACIONES I – 27139

INTRODUCCIÓN A GITHUB

Procedimiento

GitHub:

1. Si no tiene una cuenta en GitHub, se recomienda crear una. Para esto, abra un navegador web y visite el siguiente vínculo: <https://github.com/>. Allí suscríbase a una cuenta o utilice una de sus cuentas de correo para ingresar (se sugiere utilizar la cuenta de correo institucional).
2. A partir de la cuenta de github, se solicita que se cree un repositorio para el trabajo del semestre como muestra la siguiente guía: <https://docs.github.com/es/get-started/quickstart/create-a-repo>.
Utilice el nombre de la forma **GNURADIO_LABCOMUIS_2025_1_E1A_G2**.
3. Debido a la seguridad que debería aplicar cada grupo, se hace necesario que se genere un token para cada estudiante. Siga las instrucciones de este tutorial (es para el clásico, no para grano fino):
<https://docs.github.com/es/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#creating-a-personal-access-token-classic>
4. En el paso 3 del tutorial (Creating a personal access token (classic)), el ítem 6 debería tener un nombre similar a Comm1 por ejemplo, en el ítem 7 se sugiere dejarlo sin expiración, seleccione el alcance (todo repo):

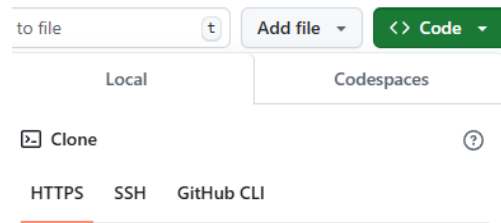
Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

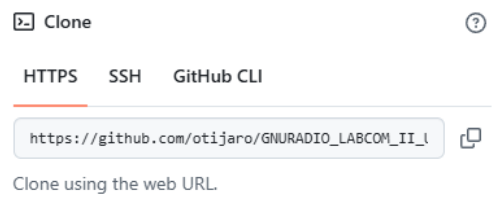
<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events

5. Generar el token y **NO OLVIDE COPIARLO** en un lugar seguro.
6. Después de creado el repositorio se solicita trabajar en la terminal de Linux. Esta aplicación la puede encontrar en el menú o ventana de aplicaciones.
7. Cree a partir del directorio /home/**user** (**user** será el usuario asignado por el profesor) comando mkdir los directorios necesarios para almacenar la información y descargar su repositorio (puede utilizar el comando pwd en la terminal, para estar en el directorio sugerido por ejemplo, /home/**user**/GNURADIO_LABCOMUIS_2025_1_E1A_G2). Verifique que esta ruta se pueda ejecutar en el PC asignado (recuerde que el directorio /home/**user** es el que se le asigna la sesión en la que usted ingresó o el que debería aparecer cuando inicia sesión y abre por primera vez la terminal, el comando pwd puede ser utilizado para verificar).

8. Después de creado el directorio, utilice el comando `cd` para acceder al directorio o a los directorios de la ruta generada en el paso anterior.
9. Ahora, vuelva al navegador a su repositorio de GitHub (el que creó en el punto 2), haga clic en el botón que dice `<> Code`:



10. Y busque que la pestaña `HTTPS` del menú que se despliega quede seleccionada.



11. Copie lo que le indica la página. Y en el directorio creado en el paso 7, ejecute el comando (git ha sido instalado previamente en las estaciones de trabajo):

git clone [git@github.com](https://github.com/otijaro/GNURADIO_LABCOM_II_I).. (las letras en azul las cambia por lo copiado en el paso anterior).

Nota: en las ventanas de terminal puede utilizar la combinación de teclas Shift + Insert para pegar lo copiado en el navegador.

También puede seguir el manual: <https://docs.github.com/es/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#using-a-personal-access-token-on-the-command-line>

Al finalizar este paso, utilice el comando `ls` para verificar que el repositorio haya quedado como un directorio local en su estación de servicio.

12. Ya teniendo el repositorio descargado en local, se puede crear el archivo `Readme.md`, verifique que esta información esté quedando almacenada correctamente en la nube (plataforma GitHub del navegador - <https://docs.github.com/es/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes>).
13. Si encuentra mensajes de error, por favor diríjase al docente o al auxiliar.
14. En la terminal de Linux, por favor ejecute el comando `git pull`.
15. Ahora, utilice una imagen o un mensaje de prueba y guárdelo como archivo en la carpeta de trabajo.
16. Utilice el comando `git status` para verificar que hubo cambios de manera local en su PC.
17. Si en la ventana de comandos se encontraron cambios, se mostrará en color rojo los archivos que cambiaron (se agregaron o quitaron). Si está de acuerdo con esos cambios, utilice el comando `git add .`

18. Si no encuentra errores, ejecute el comando `git commit -m "mensaje"`. El "mensaje" debe ir entre comillas y generalmente es un comentario que hace referencia a los archivos que subió (o eliminó) o por qué hizo los cambios. Si encuentra errores ejecute el paso 13.
19. Ejecute el comando `git push` y verifique que los cambios hayan sido subidos a la nube (plataforma GitHub en el navegador).

Debido a que esta herramienta se configura el primer día de clases prácticas, no es necesario hacer un informe, pero por cada asistencia al laboratorio si es necesario generar una rama a partir de la rama maestra (main o master). Se sugiere adicionalmente que cada práctica esté asociada a un directorio diferente y donde se utilicen los desarrollos de GNU Radio.