

GIT – GITHUB

- 1) Crear una carpeta con los proyectos. Crear en el escritorio una carpeta con nombre ProyectosISPC.
- 2) Dentro de ésta carpeta voy a traer los proyectos.
- 3) Voy a GITHUB a Issue en Project Classics y busco la tarea que tengo asignada.
Creo una de prueba: PRUEBA-101#18 en la parte de Development aparece CREATE BRANCH, o sea sobre la tarea que hay que realizar voy a crear una nueva rama (Branch): Branche.Name: PRUEBA-101.
¡ Importante ! Cambiar rama base desde la que vamos a crear la nueva rama, para esto clic en CHANGE BRANCHE SOURCE en vez de ser MAIN vamos a trabajar sobre DEV.
- 4) Luego CREATE BRANCH.
- 5) Vamos a CODE desplegamos ramas y seleccionamos PRUEBA-101.
- 6) Desde allí vamos a clonar el proyecto en CODE copiar link de CLONE (Copia).
- 7) Vamos a carpeta de ProyectosISPC, botón derecho – aparece Git Gui Here y Git Bash Here, seleccionamos Git Bash, abre ventana de comando GIT y escribo \$ git clone y pegar el link copiado desde GitHub <enter>.
- 8) Comienza a descargar el proyecto.
- 9) Ahora entro al proyecto en carpeta botón derecho – Git Bash, sino otra alternativa desde terminal de Git muestra: Estoy parado en el proyecto /Desktop/ProyectoISPC y escribir \$cd a6g2 <enter> entonces queda en: /Desktop/ProyectoISPC/a6g2 (main)
Y ya se puede comenzar a trabajar.
Recordar que se debe tener seteado la cuenta de usuario de GitHub y email. Se configura así:
\$ git config --local user.name <enter>
Puede ser -- global
Al no aparecer nada escribo \$ git config --local user.name "Sebastian Galvan" <enter>
\$ git config --local user.email sebastiangalvan.ok@gmail.com <enter>
Ésto así queda seteado para hacer los commit.
- 10) Lo que hay que hacer es cambiar de rama : \$ git branch <enter> , como no existe otra rama el resultado es *main, sino aparecerán todas las ramas.
- 11) Para cambiar a las ramas que tengo
\$git checkout prueba-101 <enter>
Muestra Switched to a new branch 'prueba-101'
branch 'prueba-101' setup to track 'origin//prueba-101'
aquí me indica que se cambió de rama: /Desktop/ProyectoISPC/a6g2 (prueba-101)
y ahora si escribo \$ git branch <enter> me responde
main
*prueba-101
Y el asterisco indica la rama en la que estoy.
- 12) Vamos a Visual Studio Code, el proyecto clonado a Visual Studio Code directamente se arrastra del explorador a VSC y lo abre.
- 13) En la ISSUE del proyecto veo la tarea asignada.
- 14) Voy a VSC y genero un nuevo archivo .py
- 15) Tarea: Hacer un programa que devuelva "hello" decirhola.py
- 16) Codificar el programa print("Hello") <ejecutar,probar,etc>
- 17) Una vez que ya probe y verifique tengo que llevar lo trabajado localmente al repositorio de GitHub. El archivo decirhola.py debe aparecer en la rama prueba-101, pero todavía no está en GitBash
- 18) Vamos a terminal de comando, con Ctrl L limpia pantalla y escribo \$ git status para ver el estado y me muestra on Branch prueba-101
your Branch is up to date with 'origin/prueba-101'
untracked files:
(use "git add <file> ..." to include in what will be committed)
decirhola.py
nothing added to commit but untracked files present (use "git add" to track)
Me dice que el archivo no está agregado, ni subido.
- 19) Vamos a hacer

```
git add decirhola.py <enter> y luego
```

```
git commit -n decirhola.py
```

Me responde

```
[prueba-101 127d562] Agregar archivo decirhola
```

```
1 file changed 1 insertion (+)
```

```
create mode 100644 decirhola.py
```

```
ya hicimos commit
```

20) Volvemos a escribir \$ git status y me dice que no hay mas cambios por guardar.

21) Ahora ya se puede subir la rama. "Para subir la rama usar \$ git push

22) Por las dudas antes de esto hay que verificar

```
git remote -v (version) nos tira donde estamos trabajando
```

```
origin https://github.com/ispc-programador2022/a6g2.git (fetch)
```

```
origin https://github.com/ispc-programador2022/a6g2.git (push)
```

23) Ahora si escribo \$ git push <enter>, la primera vez pide usuario y contraseña con formulario de GitHub o navegador y me responde

```
info: please complete authentication in your browser...
```

```
Everything up-to-date
```

```
prueba-101 → prueba-101
```

24) Vamos a GitHub y al recargar la rama prueba-101 ya me aparece el archivo decirhola.py

25) Después de esto hay que hacer un PULL REQUEST, es una solicitud para guardar lo que está en la rama PRUEBA-101 a DEV. Se trabaja siempre sobre DEV no sobre MAIN.

26) Desde rama PRUEBA-101 voy a PULL REQUEST- NEW PULL REQUEST

```
Seleccionar rama base: NO es MAIN debe ser DEV
```

```
Compare ( Seleccionar prueba-101)
```

```
base   compare
```

```
DEV    PRUEBA-101
```

27) CREATE PULL REQUEST, Breve descripción de la rama y luego descripción de lo que hace. Al costado hay confirmaciones del Pull Request. REVIEWERS: Se selecciona a otro colaborador para que vea si lo que hice esta bien, hasta que no da el OK de acuerdo a la configuración del proyecto NO se puede actualizar la principal. Les llega Notificación a los colaboradores para que revisen.

28) Cuando los REVIEWERS apueban me aparece el MERGE PULL REQUEST, se aplica el MERGE y se agrega en DEV, o sea recién va aparecer el archivo decirhola.py en DEV.

29) Por ultimo \$ git pull en MAIN