

UNIVERSIDAD DE LAS FUERZAS ARMADAS (ESPE)

SEGUNDO SEMESTRE

CARRERAS TECNICAS

PRIMER PARCIAL

POO: INFORME DE LA EVIDENCIA

AUTOR:

Murminacho Cabascango Alexis Sebastián

PARALELO:

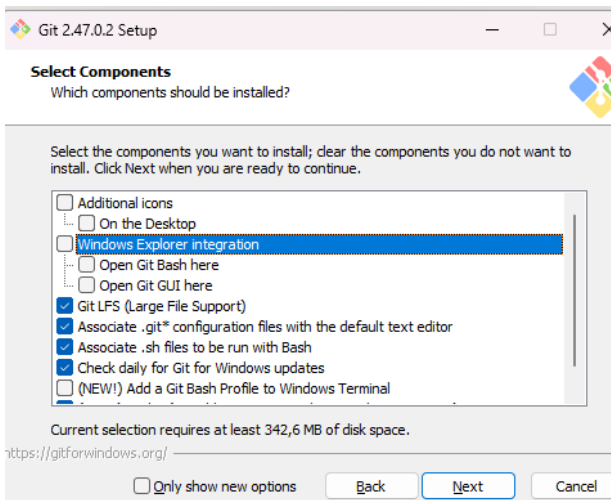
NRC-1323

INTRODUCCIÓN

El objetivo de este informe es documentar el proceso de creación y configuración de un repositorio en GitHub, siguiendo las instrucciones dadas en clase. Este repositorio fue creado para aprender las bases del control de versiones y el trabajo colaborativo con GitHub.

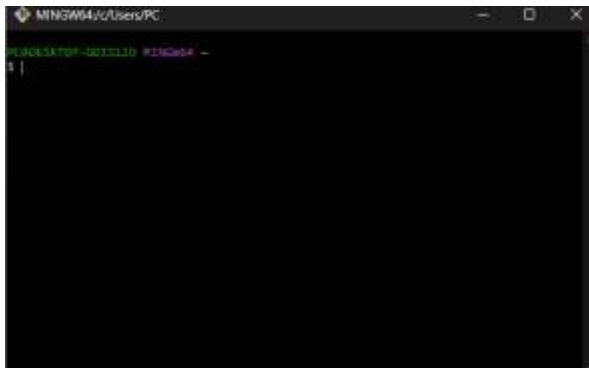
1. PROCESO REALIZADO

-Como primer paso fue instalar git para Windows de 64-bit, después que ya lo descargamos, seguimos los pasos para instalar git con los pasos a seguir en el material adicional del Moodle



- Selecciona "Use Git from Git Bash only" para utilizar Git desde la línea de comandos de Git Bash o "Use Git from the Windows Command Prompt" si deseas utilizar Git desde la línea de comandos de Windows. Luego, haz clic en "Next" (Siguiente).
- Selecciona "Use the OpenSSL library" y deja las demás opciones predeterminadas. Haz clic en "Next" (Siguiente).
- Selecciona "Checkout: Windows-style, commit Unix-style line endings" y haz clic en "Next" (Siguiente).
- Elige "Use Windows' default console window" y haz clic en "Next" (Siguiente).
- En la pantalla "Select the default behavior of 'git pull'", elige "Rebase" y haz clic en "Next" (Siguiente).
- Selecciona "Enable file system caching" y haz clic en "Next" (Siguiente).
- Elige "Enable Git Credential Manager" y haz clic en "Next" (Siguiente).
- En la pantalla "Configuring the line ending conversions", selecciona "Checkout as-is, commit Unix-style line endings" y haz clic en "Next" (Siguiente).
- Haz clic en "Install" (Instalar) para comenzar la instalación de Git.
- Espera a que se complete la instalación y luego haz clic en "Finish" (Finalizar).

-Una vez ya configurado eh instalado, obtenemos el Git Bash Here lo acual abre una terminar con la dirección oh ruta del directorio



- Nos registramos en Git-Hub, es necesario hacer este paso ya que el correo que tenemos registrado nos servirá para vincularlo con Git-Bash



- Como ya tenemos git, procedemos a configurar el nombre de usuario

- Abrimos Git Bash
- Establecemos un nombre de usuario en Git con el siguiente código:
`git config --global user.name "Sebastian"`
- Para confirmar que se ha establecido correctamente el Usuario escribimos el siguiente código:
`git config --global user.name`

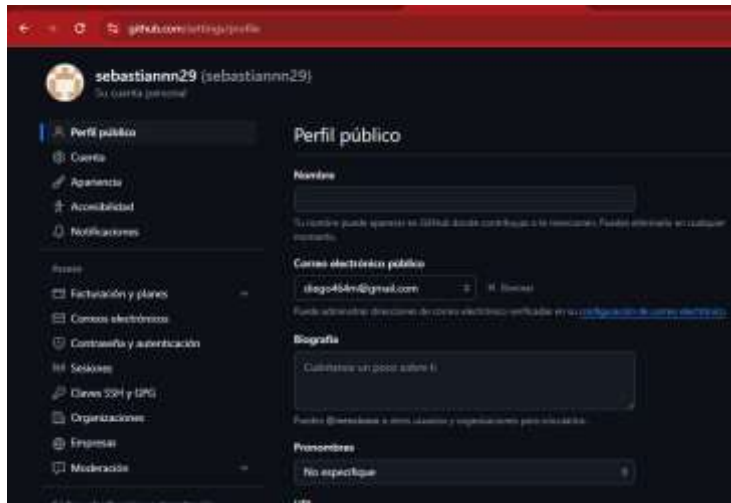
```
MINGW64/c/Users/PC
PCBUESATOP-G015110 MINGW64 -
$ git config --global user.name "Sebastian"

PCBUESATOP-G015110 MINGW64 -
$ git config --global user.name
Sebastian

PCBUESATOP-G015110 MINGW64 -
$ |
```

-Procedemos a configurar el correo electrónico

- Para este paso, es necesario ya tener el registro de Git-hub
- confirmamos el correo en el git hub, abriendo en ajustes, y verificamos si es el correo con el que se registraron



- **Ahora confirmamos el correo en Git Bash**, abriendo el terminal y ponemos el siguiente código:

```
git config --global user.email "YOUR_EMAIL"
```

-Comprobar las SSH

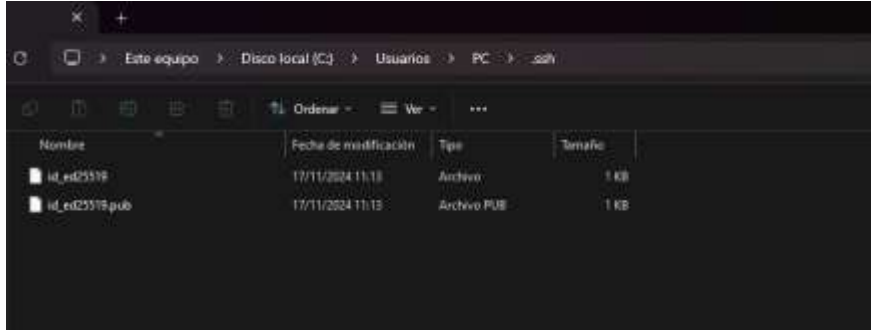
- Abrimos el Git Bash
- Ponemos el siguiente código:

`ls -al ~/.ssh`
- vemos las llaves que tenemos, las que terminen en “.pub” son las que necesitaremos

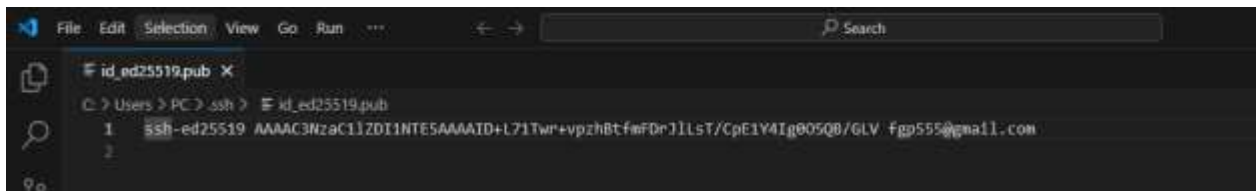
```
MINGW64:/c/Users/PC
PC@DESKTOP-GOIS110 MINGW64 ~
$ ls -al ~/.ssh
total 18
drwxr-xr-x 1 PC 197121  0 Nov 17 11:13 ./
drwxr-xr-x 1 PC 197121  0 Nov 22 19:24 ../
-rw-r--r-- 1 PC 197121 411 Nov 17 11:13 id_ed25519
-rw-r--r-- 1 PC 197121  98 Nov 17 11:13 id_ed25519.pub
PC@DESKTOP-GOIS110 MINGW64 ~
$
```

-Procedemos a buscar donde se guardan las SSH

- Vamos al disco local C, dentro de la carpeta usuarios y abrimos en la carpeta “.ssh”




- Hacemos clic derecho en la clave publica y lo abrimos en visual studio code o en el block de notas



-Agregamos la clave en Git-hub

- Copiamos la clave publica
- Abrimos Git hub en el navegador
- Vamos a configuración de perfil
- Nos dirigimos a claves SSH

 Claves SSH y GPG


- agregamos la clave y ya tendremos vinculado git con git-hub

Claves SSH

[Nueva clave SSH](#)

Esta es una lista de claves SSH asociadas a su cuenta. Elimine las claves que no reconozca

Claves de autenticación


SSH

Caca

SHA256:1cGY4/5pu14/NQDwz0sXAu+sdgCnQL7uqSvx5szMyhM

Agregado el 17 de noviembre de 2024

Nunca usado — Lectura/escritura

[Borrar](#)

Consulta nuestra guía para [conectarte a GitHub usando claves SSH](#) o solucionar [problemas comunes de SSH](#).

-CREAR UN REPOSITORIO

- Abrimos la carpeta donde del proyecto
- damos clic derecho en la carpeta
- abrimos con git bash
- ponemos el código git init
- ponemos el código git status

```
CHENAO@LAPTOP-4NUN6GBP MINGW64 /d/SENA/Eclipse2021/Worckspace914/ClinicaVeterinari
UI (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .classpath
    .project
    .settings/
    bin/
    src/
```

- ponemos el código git add . para agregar los doc al repositorio
- después ponemos el código: git Branch -M main
- y por ultimo copiamos el código remoto que nos da al momento de crear un repositorio en git hub

...or create a new repository on the command line

```
echo "# ClinicaVeterinariaGUI" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/chenaoh/ClinicaVeterinariaGUI.git
git push -u origin main
```