

T.U.P

Programación I

Estudiante: Paz Isaías Nicolás

Trabajo Práctico II - Git y GitHub

1

- ¿Qué es GitHub?

Es una plataforma online que funciona como un servicio para los usuarios para guardar, compartir y colaborar en repositorios anexados por Git.

- ¿Cómo crear un repositorio en GitHub?

Crear una cuenta en GitHub e iniciar sesión. Entrar en el perfil en la esquina superior izquierda y tocar "crear nuevo repositorio". Completar los datos "nombre del repositorio" y "Descripción". Se recomienda añadir un archivo README.txt

- ¿Cómo crear una rama en Git?

Con el comando **git branch** podemos ver una lista de ramas y la rama donde estamos posicionados y con el comando **git branch {nombre de la rama}** creamos una nueva rama.

- ¿Cómo cambiar a una rama en Git?

Git checkout {nombre de la rama} es el comando que nos permite cambiar entre ramas.

- ¿Cómo fusionar ramas en Git?

Podemos fusionar ramas ubicándonos en la rama principal y colocar el comando **git merge {nombre de la rama}**.

- ¿Cómo crear un commit en Git?

Con el comando **git commit -m {mensaje}**

- ¿Cómo enviar un commit a GitHub?

Usando el comando **git push -u origin main** en caso de que el commit esté en la rama principal.

- ¿Qué es un repositorio remoto?

Es una carpeta en una plataforma en internet que almacena diferentes archivos relacionados a códigos y herramientas

- ¿Cómo agregar un repositorio remoto a Git?

Con el comando **git remote add origin {enlace del repositorio remoto}**.

- ¿Cómo empujar cambios a un repositorio remoto?

Con el comando **git push origin main**.

- ¿Cómo tirar de cambios de un repositorio remoto?

Con el comando **git pull origin main**.

- ¿Qué es un fork de repositorio?

Un fork es una copia del repositorio de otra persona en tu cuenta de GitHub.

- ¿Cómo crear un fork de un repositorio?

Ingresamos al repositorio en GitHub que queremos copiar. Clic en el botón "Fork" (arriba a la derecha). Elegir la cuenta personal o una organización. GitHub creará una copia del repositorio en tu cuenta.

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

Realizamos el cambio deseado en nuestro fork y después hacemos push al repositorio local.

- ¿Cómo aceptar una solicitud de extracción?

Entrar al repositorio seleccionado y en la pestaña de pull request podemos revisar las diferentes request y analizarlas. Si todo está en orden , se hace un **merge pull request**.

- ¿Qué es una etiqueta en Git?

Es una manera de marcar las diferentes versiones de una rama o un proyecto.

- ¿Cómo crear una etiqueta en Git?

Con el comando **git tag {nombre de la etiqueta}**.

- ¿Cómo enviar una etiqueta a GitHub?

Utilizando el comando **git push origin {nombre del tag}**.

- ¿Qué es un historial de Git?

Un registro de todos los commits que se hicieron en un repositorio.

- ¿Cómo ver el historial de Git?

Con el comando **git log**.

- ¿Cómo buscar en el historial de Git?

Con el comando **git log -grep={palabra o búsqueda}**.

- ¿Cómo borrar el historial de Git?

Comando: **git branch -d {rama}**.

- ¿Qué es un repositorio privado en GitHub?

Un repositorio privado en git es un repositorio cuyo proyecto solo puede ver el dueño del repositorio y todas aquellas personas que el dueño permita.

- ¿Cómo crear un repositorio privado en GitHub?

Cuando se crea un nuevo repositorio y se añade el nombre junto con la descripción del repo, se nos brinda la opción de hacer el repositorio público o privado.

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

Ingresa al repositorio privado y en la pestaña de ajustes, desplegando el menú lateral, accedemos a “**collaborators**”. Hacemos clic en “**add people**” y confirmamos.

- ¿Qué es un repositorio público en GitHub?

Un repositorio público es un proyecto de código abierto a cualquier persona. Todo aquel usuario que desee verlo o clonarlo puede hacerlo. También, el dueño puede aceptar pull request de otros usuarios a fin de modificar el código.

- ¿Cómo crear un repositorio público en GitHub?

Del mismo modo que creando un repositorio privado, al momento de crearlo, un menú desplegable nos permite seleccionar entre privado y público.

- ¿Cómo compartir un repositorio público en GitHub?

Copiando el url del repositorio y pegarlo donde se desee.

2. Crear un nuevo repositorio en gitHub:

<https://github.com/Nicodemus10Mil/Repositoriotest>

```

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git branch
  main
* nuevaRama

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ ls
README.md  UTN-TUPaD-P1/  mi-archivo.txt

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git add "nuevo archivo"
fatal: pathspec 'nuevo archivo' did not match any files

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git add hola.txt
fatal: pathspec 'hola.txt' did not match any files

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git commit -m "cambiohecho"
On branch nuevaRama
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ ^C

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git add README.md

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$ git push origin nuevaRama
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'nuevaRama' on GitHub by visiting:
remote:   https://github.com/Nicodemus10Mil/Repositoriotest/pull/new/nuevaRama
remote:
To https://github.com/Nicodemus10Mil/Repositoriotest.git
 * [new branch]      nuevaRama -> nuevaRama

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Repo de trabajos programación (nuevaRama)
$

```

Ejercicio 3

<https://github.com/Nicodemus10Mil/conflict-exercise>

```

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main|MERGING)
$ git branch
  feature-branch
* main

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main|MERGING)
$ git branch
  feature-branch
* main

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main|MERGING)
$ git add README.md

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main|MERGING)
$ git commit - "Resolved merge conflict"
fatal: cannot do a partial commit during a merge.

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main|MERGING)
$ git commit -m "Resolved merge conflict"
[main c1ec3bf] Resolved merge conflict

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main)
$ git branch
  feature-branch
* main

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main)
$ git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 893 bytes | 893.00 KiB/s, done.
Total 9 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/Nicodemus10Mil/conflict-exercise.git
   ffb3c21..c1ec3bf  main -> main

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main)
$ git push origin feature-branch
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-branch' on GitHub by visiting:
remote:   https://github.com/Nicodemus10Mil/conflict-exercise/pull/new/feature-branch
remote:
To https://github.com/Nicodemus10Mil/conflict-exercise.git
   * [new branch]   feature-branch -> feature-branch

Administrator@DESKTOP-710N71H MINGW64 ~/Desktop/Programas ruby/exercicio 3/conflict-exercise (main)
$ |

```