UNIVERSIDAD DEL CENTRO DE LA PROVINCIA DE BUENOS AIRES

FACULTAD DE CS. EXACTAS

TECNICATURA UNIVERSITARIA EN DESARROLLO DE APLICACIONES INFORMÁTICAS

Trabajo práctico especial

[Escriba el subtítulo del documento]

[Seleccione la fecha]

[Escriba el nombre de la compañía]

Integrantes:

Argüello, Nadia

Brasesco Hernández, Fidel

Cappi, Martin

Ehazarreta, Cecilia

Fernández, Florencia

Vaninetti,  Valentín

Montero, Sebastián

Navarro, Fabián

Rubiales, Lucia

Urban, Matías

Materia: Tecnología de la información en las organizaciones

Fecha: 10/11/21

**ÍNDICE**

Introducción…………………………………………………………………………..

Desarrollo……………………………………………………………………………..

Conclusión……………………………………………………………………………..

Anexos………………………………………………………………………………….

Bibliografía……………………………………………………………………………..

**INTRODUCCIÓN**

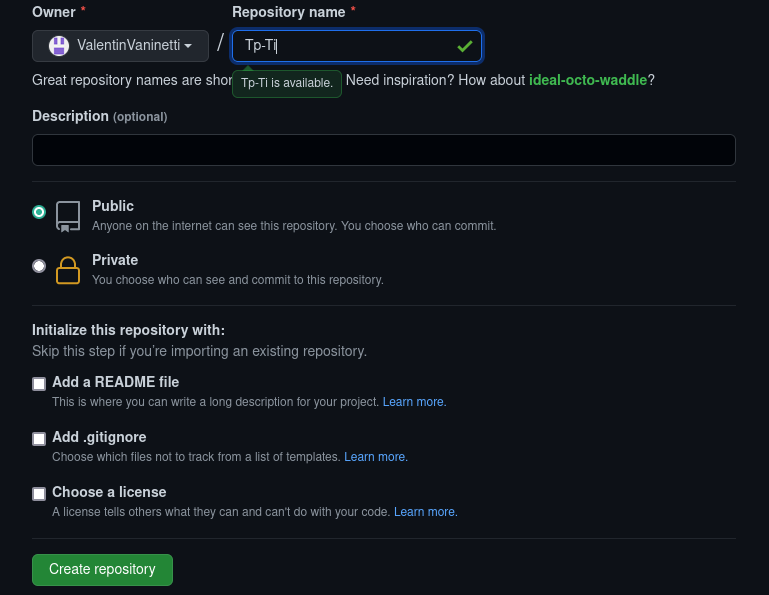
**DESARROLLO**

Se decidió trabajar colaborativamente sobre un proyecto de página web cuya temática es Git, qué es, para qué se utiliza,  comandos útiles y demás información que se consideró de ayuda para los estudiantes y/o personas que buscan aprender un poco más sobre Git y GitHub.

Para organizar y guiar el trabajo en equipo se utilizó la herramienta de gestión Trello, en la cual se creó un tablero y se plasmó con tarjetas cómo se desarrollaría el trabajo.

En primera instancia se definió el proyecto y luego se procedió a dividir las tareas:  uno sería el encargado de redactar el informe, dos de crear el HTML correspondiente a cada página, otros dos del CSS y los restantes dos de agregar funcionalidades con JavaScript.

Uno de los integrantes creó una carpeta llamada tp -tio en el  repositorio local  y lo subió a un repositorio remoto.

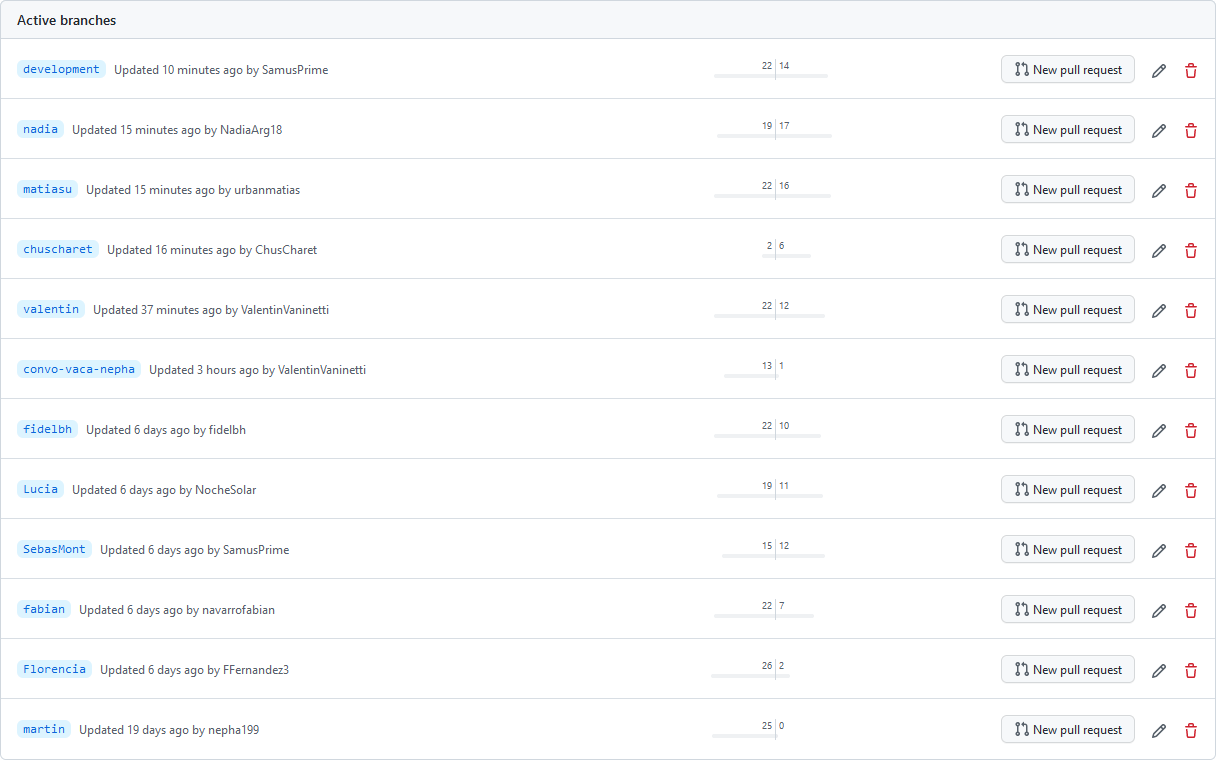


*Creación de un repositorio remoto y ramas remotas*

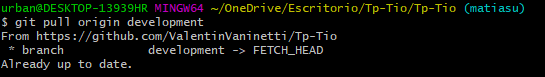
Se creó un repositorio remoto en GitHub y se agregó a todos los integrantes del equipo como colaboradores. Por ende cada uno debió vincular su repositorio local con el remoto mediante el comando *git remote add origin URL\_GitHub.*

Seguidamente se clonó la carpeta que inicialmente había subido el integrante que creó el repositorio remoto.

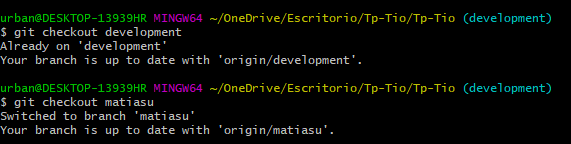
Para trabajar de manera más organizada cada uno creó una rama local con su nombre y luego se subieron al repositorio remoto (*git push -u origin nombre\_rama\_integrante).*

**

En una rama alternativa llamada development se irá subiendo el desarrollo del trabajo, por todos los integrantes. Fue necesario que, luego de que un integrante la creara localmente y la subiera al repositorio remoto, cada uno actualizó su propia rama (con el nombre de cada uno)  con los cambios hechos en la rama de desarrollo. Para eso se utilizó el comando *git pull origin development.*

**

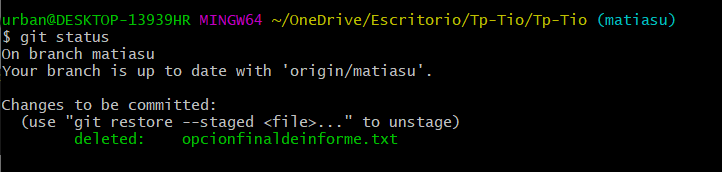
Para que cada integrante pueda trabajar de manera local en su branch se utilizó el comando *git checkout nombre\_de\_la\_rama*

**

A medida que se avanzaba con el trabajo y los integrantes del grupo fueron creando los archivos, se utilizó el commando *git add .* así como *git add nombre\_del\_archivo* para poder agregarlos al repositorio.

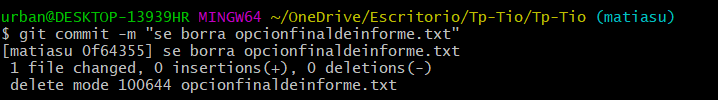
**

*Al utilizar este comando, luego podemos usar “git status” para ver el estado, y los cambios por realizar.*

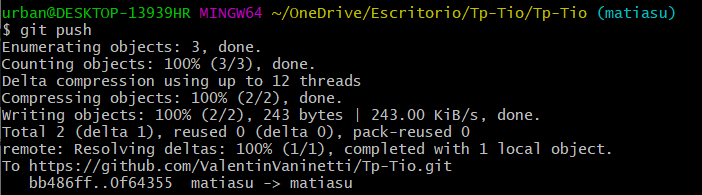
**

En el ejemplo de la foto podemos observar que en color verde se muestran los cambios realizados listos para hacer commit.

Con el comando *git commit –m “mensaje”* confirmamos el cambio, agregando un mensaje para que cada uno de los integrantes vea que fue lo que se agregó, o modificó en los archivos.

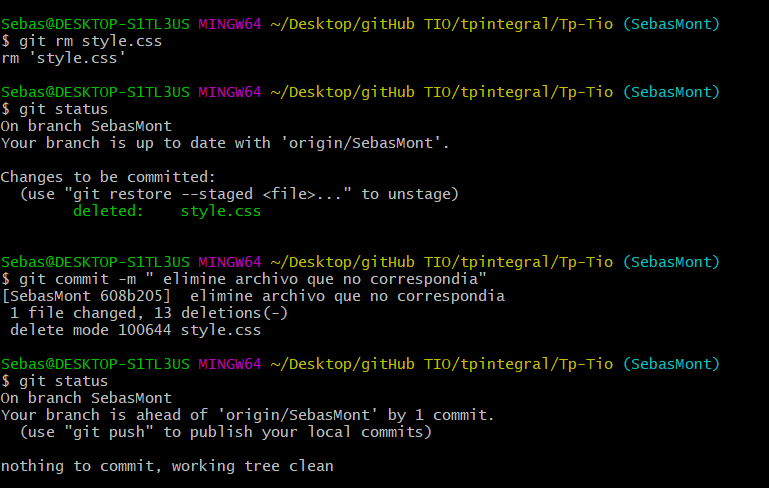
**

Luego de confirmar los cambios, procedemos a subirlos al repositorio remoto utilizando el comando *git push.*

**

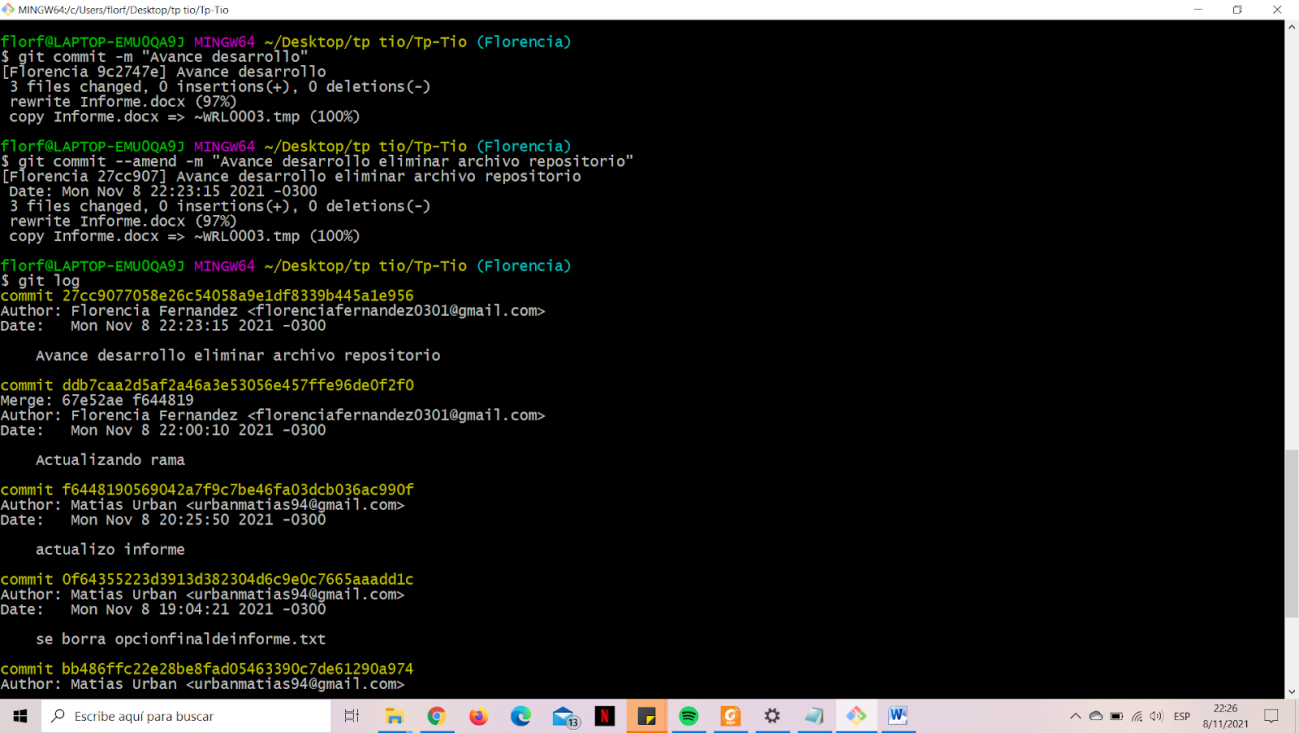
Eliminar archivo del repositorio

Se eliminó un archivo CSS que no correspondía del repositorio con *git rm nombre\_archivo\_a\_eliminar,* seguidamente fue necesario realizar un commit para confirmar la eliminación.

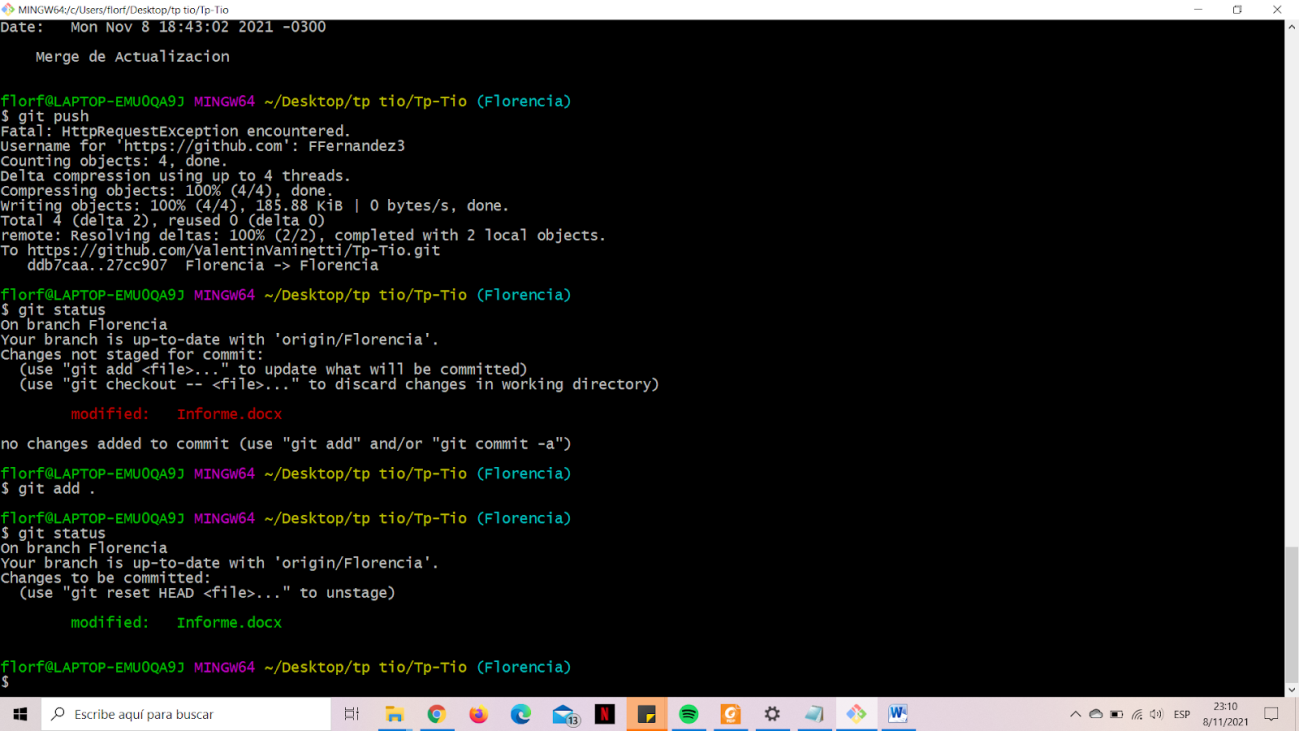


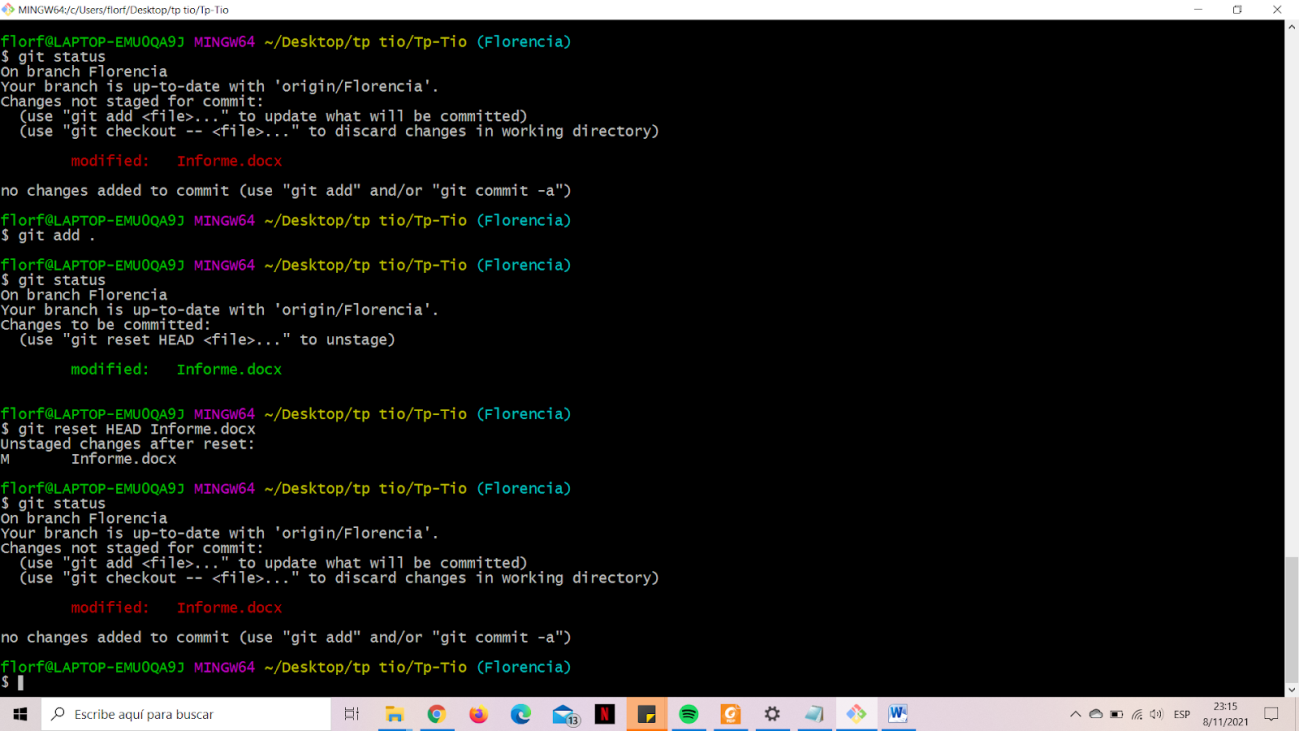
Cambio mensaje de commit

Después de realizar un commit con avances sobre el desarrollo, se evidencia que el mensaje podría ser más conciso. Para enmendar este error se usa *git commit --amend -m “nuevo mensaje”.* Si se lee el historial de commits en este momento con *git log* se observa que el mensaje que se redactó en un primer momento ya no está, sino que fue reemplazado por lo escrito del amend.

  
  
Deshaciendo cambios

Uno de los integrantes modificó el contenido del informe agregando los archivos al staging area con un *git add .* Pero se dió cuenta de un error en la bibliografía y decidió quitar los cambios de la zona de intercambio temporal. Para eso utilizo el comando *git reset HEAD Informe.docx,* y luego se observa que los cambios efectivamente fueron quitados del staging area (vuelven a estar en rojo).

**

**

**CONCLUSIÓN**

**ANEXOS**

Enlace al repositorio remoto: https://github.com/ValentinVaninetti/Tp-Tio

Enlace al tablero de Trello: https://trello.com/b/25r8QIE6/tp-tio

Capturas de pantalla

**BIBLIOGRAFÍA**

Chacon, S. & Straub, B. (2021). *Pro Git: todo lo que necesitas saber acerca de git*. 2a. ed. Apress. 531 pp. https://git-scm.com/book/en/v2

Dudler, Roger (s.f.) *Git la guía sencilla.* http://rogerdudler.github.io/gitguide/index.es.html