

Informe trabajo colaborativo Git



16 de octubre de 2021

Universidad: Universidad Nacional Del Centro De La Provincia De Buenos Aires

Autores: - Tomas Mondini

- Joaquín Tobal

- Sebastián Rosende

- Agustín Urrutia

- Patricio De Vicente

Profesor: Florencia Rodríguez

Brian Varona

Materia: Tecnología de la Información en las Organizaciones

Enlace del repositorio: <https://github.com/PDV10/TPE-TIO.git>

Enlace de tablero Trello: <https://trello.com/b/ZPvQnVuN/tpe-tio>

***Índice***

***○*** *Portada.*

***○*** *Pagina 1: índice.*

***○*** *Pagina 2: Introducción.*

***○*** *Pagina 3: Desarrollo y archivos modificados.*

***○*** *Pagina 4: Todolist antes de los cambios.*

***○*** *Pagina 5: Todolist después de los cambios.*

***○*** *Pagina 6 y 7: Errores.*

***○*** *Pagina 8 a 15: Pautas de aprobación.*

**Introducción:**

En el presente trabajo se decidió trabajar en un ejercicio de la materia Web 2 con el objetivo de hacer un refactor en el back-end y front-end de la aplicación. Esta elección se debe a que ya es una base armada donde podremos trabajar y refrescar los conocimientos aprendidos en la mencionada cátedra, buscando solucionar problemas de conexión en la base de datos, ruteo, controladores y vista del mismo.

Para la realización del trabajo se usaron herramientas como Trello, Git, GitHub, Meet, WhatsApp, herramienta de recorte y Word.

**Desarrollo:**

Se inicia el proceso de trabajo con una reunión por Google meet, en esta se realizó la creación de un repositorio Git y un espacio de trabajo colaborativo de Trello, se plantearon y leyeron los puntos del trabajo practico correspondiente y se aclararon dudas en grupo. Mientras que se realizaba la lectura se realizó el tablero de Trello y para dividir el trabajo cada uno tomo una tarea a elección.

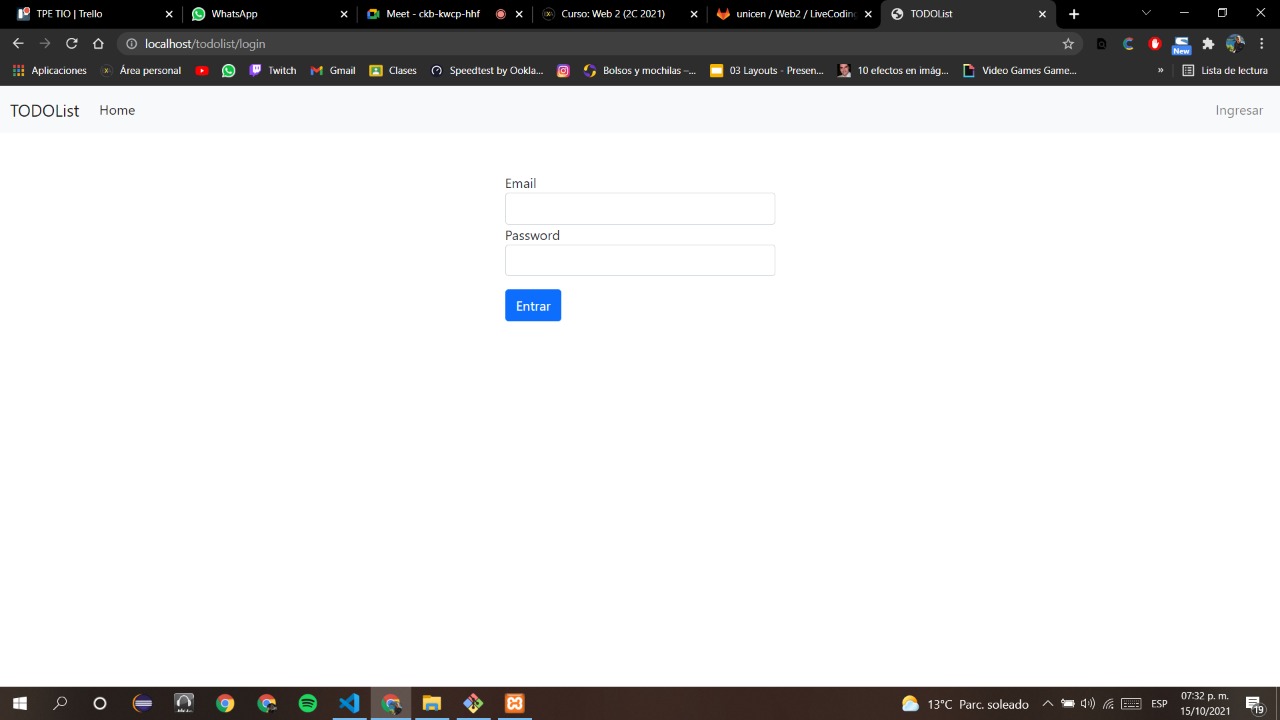
Patricio se encargó de el inicio y creación del repositorio git, un meet para el trabajo que inicio Joaquín, Agustín se encargó de el cambio estético de la página, Tomas fue escribiendo el informe a medida que avanzaba el proyecto y Sebastián se encargó de la pestaña about. Por su parte cada uno inicio su rama branch local, en conjunto se creó la rama remota “desarrollo” con la finalidad de trabajar en ella para luego mergear a rama master por ejemplo el botón modificar. También se hicieron sus respectivos 3 commits, push, pull y la eliminación de su rama local creada anteriormente. A medida que se iban cumpliendo las tareas fuimos tachándolas en la check list llamada “pautas de aprobación” en Trello donde todos fuimos comentando la etiqueta.

Dentro del documento de trabajo se encuentra la carpeta “todolist” que contiene los archivos de la página web a modificar, el archivo .gitignore que debería ignorar git, el documento Word del informe, la carpeta llamada “errores” que solo contiene capturas de pantalla de las situaciones que fueron apareciendo en el transcurso del trabajo. La carpeta “img proceso” contiene imágenes del progreso y comandos usados. Las mismas fueron útiles para incluirlas dentro del informe en el documento Word.

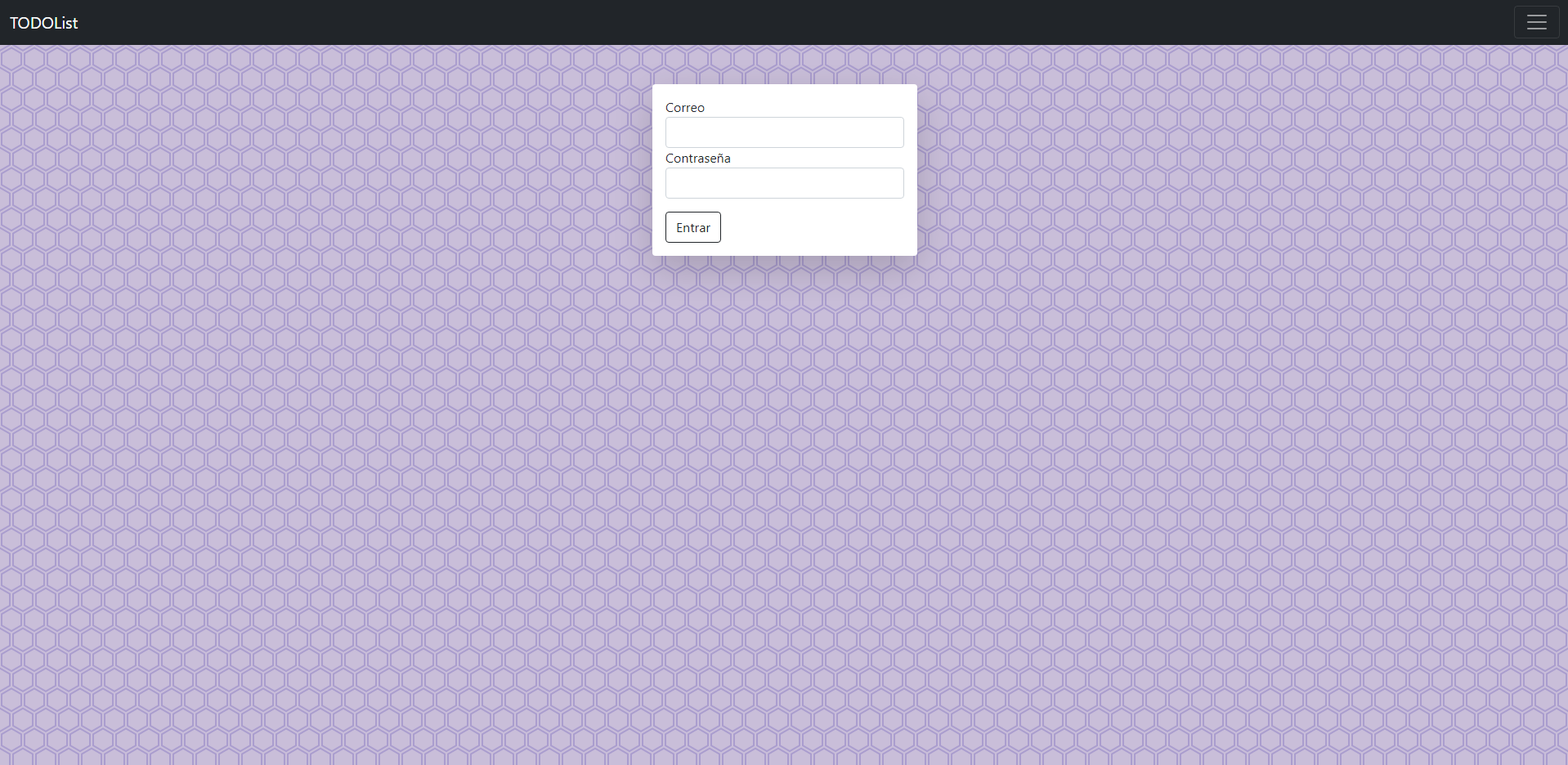
En “todolist” se realizaron modificaciones en archivos como, router.php, style.css, task.controller.php, task.model.php y se generaron 4 archivos tpl (contacto, about, FormModificar y showFormModificar). También se modificó el readme.md y la base de datos db.task.sql .

Lo que es el desarrollo del trabajo como ya hablamos es la modificación de “todolist”, que en un principio se veía así:





Y decidimos hacerles cambios como el estilo del login y menú, como resultado final la página quedo de esta manera:



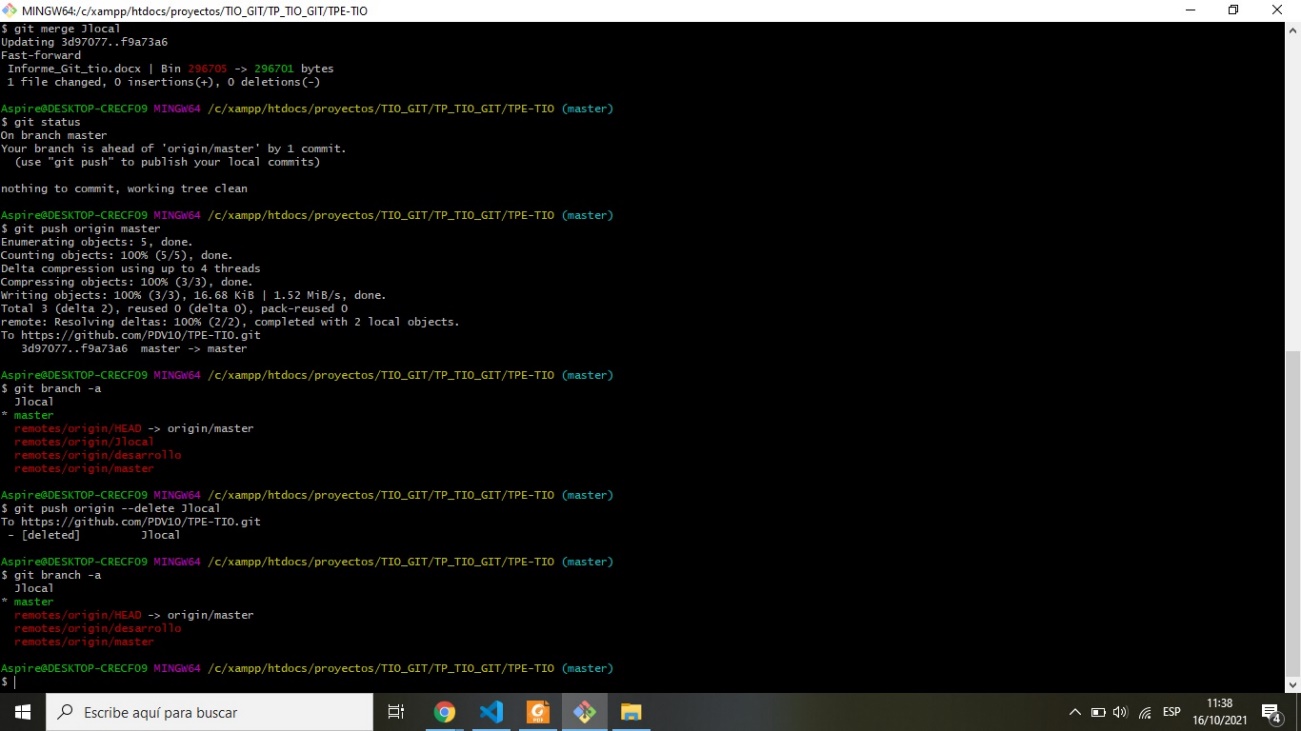


Cambios sencillos como colores, agregar un fondo, cambiar al español las palabras email y password y la implementación de la funcionalidad modificar tarea, pero el enfoque principal estuvo en la organización del trabajo y el uso de git.

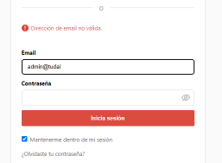
**Errores:**

*En el proceso nos encontramos con tres errores que pudimos capturar.*

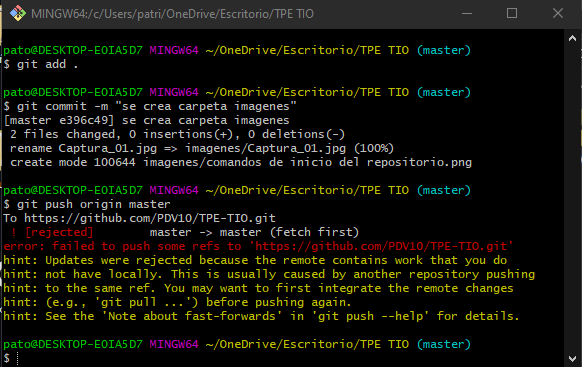
*•* Por error se creó una rama remota. Al intentar hacer un push se pudo ver el error, por lo tanto, fue necesario verificar las ramas remotas y se procedió a eliminar la que correspondía:



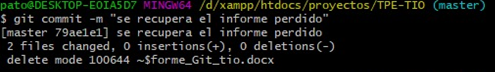
• Surgió un problema al querer ingresar a “todolist” ya que no se había actualizado la base de datos:



• El siguiente error apareció al querer “pushear” cambios, surgió al trabajar 2 personas al mismo tiempo y lo tratamos varias veces:



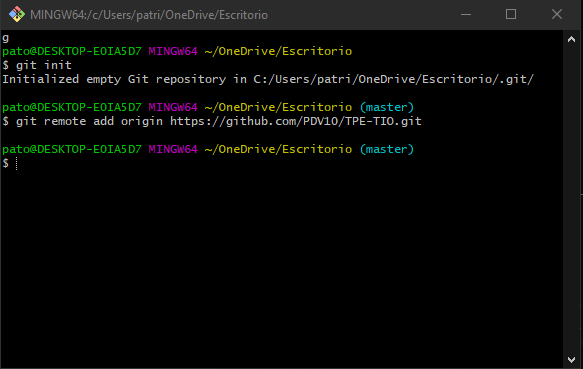
• Tuvimos un inconveniente con el informe realizado, y gracias a los comandos git reset –hard logramos recuperar el archivo perdido.



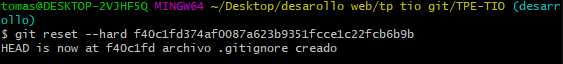
**Pautas de aprobación**

**Recopilación en formato imagen del progreso en la realización del proyecto**:

• En esta imagen se puede ver como iniciamos un repositorio Git con sus respectivos comandos y como se vincula con el repositorio remoto de Git Hub:

:

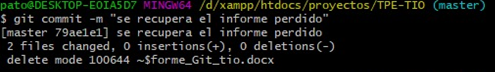
• En esta captura vemos el uso de Git reset para deshacer al menos una confirmación (git reset –hard más el “hash” de dicho commit):



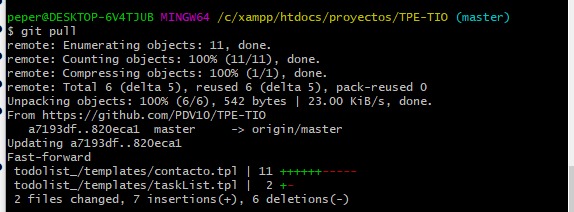
• Git add . para poner los archivos en área de preparación:



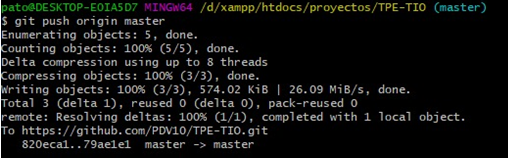
• Git commit –m “comentario” para “commitear” los cambios hechos:



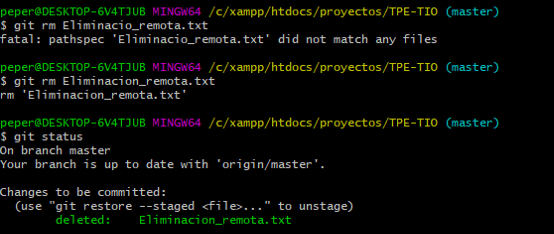
• Comando usado para actualizar repositorio local (git pull):



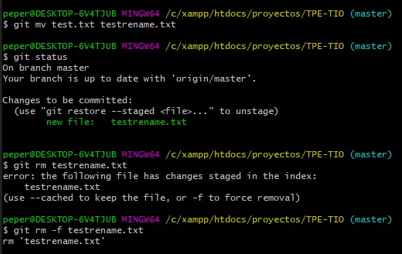
• Comando utilizado para subir cambios commiteados (git push):



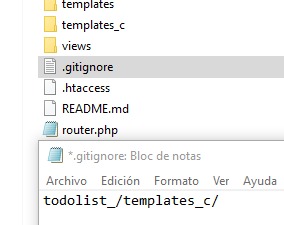
• Eliminación de un archivo remoto a través del comando (git rm):



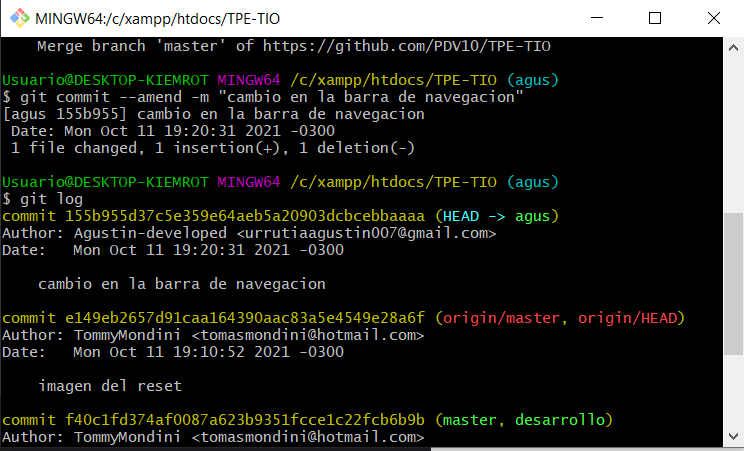
• Renombramos y luego eliminamos un archivo a través de comandos (git mv y git rm):



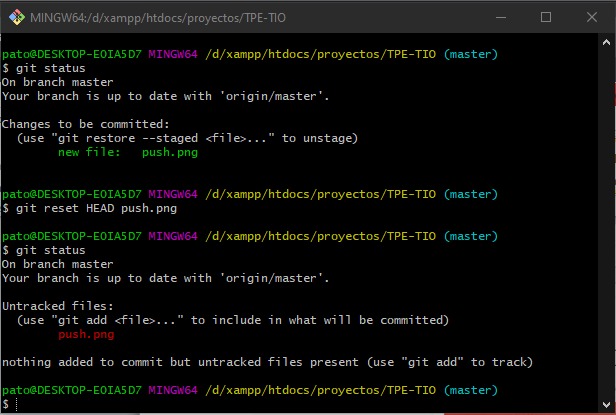
• En el archivo. gitignore se agrega la carpeta templates\_c que se genera automáticamente por Smarty para se ignorada:



• En este recorte se ve como se modifica el comentario de un commit anterior (git commit –ammend):



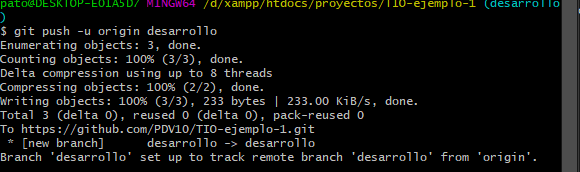
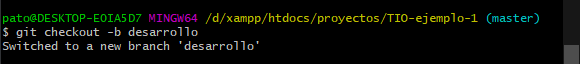
• Comando utilizado para retirar un archivo del área de preparación (git reset HEAD).



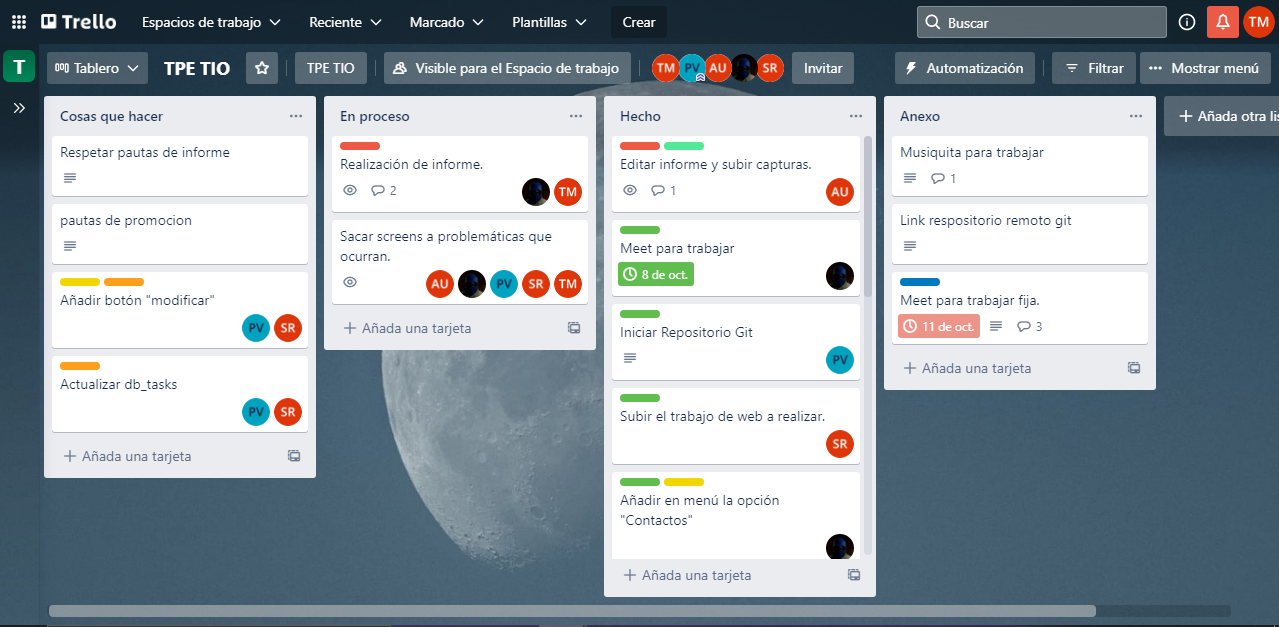
• Se realiza la eliminación de las ramas locales creadas al principio que pertenecían uno a cada participante (git Branch –D nombre de rama):



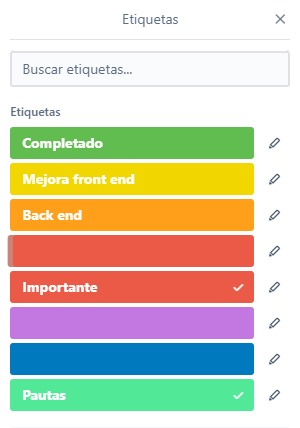
• Por último se crea rama remota desarrollo y se sube al repositorio de GitHub.



• Tablero de Trello compartido donde se fueron asignando y dividiendo las tareas y agregando etiquetas a las mismas con diferentes criterios entre los integrantes del grupo para la realización del proyecto:



• Etiquetas asignadas a las tareas



• Tarea Pautas de aprobación con el uso de la herramienta check list para subdividir tareas.

