

Práctico 2: Git y GitHub

Fontana, María Cecilia.

Actividades

1) *Contestar las siguientes preguntas utilizando las guías y documentación proporcionada (Desarrollar las respuestas) :*

• ¿Qué es GitHub?

GitHub es una herramienta de control de versiones .Fue creada en 2005 por Linus Torvalds. Funciona a nivel local en una computadora. Este software es muy eficiente para llevar el control del ciclo de vida de todo proyecto y de todos los colaboradores que en él trabajan. Para poder trabajar con otros programadores se deberá compartir nuestro repositorio privado a través de redes privadas o subiéndose a internet en plataformas como por ejemplo GitHub. Además permite dar seguimiento de los cambios, puede verse quienes y cuando hicieron los mismos. Así como cuáles fueron los conflictos que aparecieron en todo su historial.

• ¿Cómo crear un repositorio en GitHub?

Para crear un repositorio en GitHub puedo crearlo directamente en GitHub o subir mi repositorio de Git a GitHub.

En este caso voy a relatar los pasos para subir mi repositorio de Git a GitHub:

- Se inicializa el repositorio en Git con el comando `git init`.
- Luego se agrega el archivo a ese repositorio con `git add`.
- Paso seguido con un `git commit -m "se escribe un comentario que se inicia un repositorio"`.
- Se inicia sesión en GitHub .
- Estando en el perfil hacer click en el signo +, y, al desplegarse el menú, hacer click en "new repository".
- Agregar un nombre en "repository name".
- Elegir la opción para que sea público o privado.
- Hacer click en "create repository". Se ha creado el repositorio en GitHub pero aún está vacío.
- Se selecciona la instrucción del comando `git remote add origin completa` y se copia.
- Se pega esa instrucción en la consola para asociar el repositorio remoto al repositorio local.
- Copiar la instrucción del comando `push` que se indica en Github y lo pegar la misma en la consola llevando así los cambios del repositorio local al repositorio remoto en GitHub.
- Se podrá visualizar los datos agregados al repositorio refrescando la página.

En caso de crearlo desde GitHub se deben realizar los siguiente pasos:

- Se inicia sesión en GitHub .
- Estando en el perfil hacer click en el signo +, y, al desplegarse el menú, hacer click en “new repository”.
- Agregar un nombre en “repository name”.
- Elegir la opción para que sea público o privado.
- Hacer click en “create repository”.

• ¿Cómo crear una rama en Git?

Para crear una nueva rama se utiliza el comando:

- `git branch nombre de la rama.`

• ¿Cómo cambiar a una rama en Git?

Para cambiar a un rama en Git se escribe el comando

- `git checkout nombre de la rama`

• ¿Cómo fusionar ramas en Git?

Para fusionar ramas en GitHub:

- Primero se debe mirar estar en la rama a la cual le quiero fusionar otra rama, con `git branch`. Me va a devolver las ramas existentes y la que tiene el asterisco es la rama en la cual se está. Si es la rama en la cual se quiere integrar la nueva rama entonces se debe hacer:
- `git merge nombre de la rama a integrar.`

• ¿Cómo crear un commit en Git?

Para crear un commit en Git:

- Primero se agrega el archivo modificado con el comando: `git add` .
- Luego se crea el commit con el comando: `git commit m~` “mensaje que describa lo que realizó”

• ¿Cómo enviar un commit a GitHub?

Se puede realizar usando lo siguiente:

- `git push origin nombre de la rama donde se realizó el commit`

• ¿Qué es un repositorio remoto?

Son repositorios que pueden compartirse con otras personas mediante redes privadas como universidades, empresas o también pueden estar en sitios en internet como por ejemplo

GitLab, SourceForge y el más utilizado GitHub. Permite colaborar con proyectos abiertos de manera remota permitiendo trabajar a varios desarrolladores en ese proyecto a la vez.

• ¿Cómo agregar un repositorio remoto a Git?

De la siguiente manera:

- En la página de GitHub hacer click en el botón Code y va a mostrar la URL para copiar.
- En la consola se usa el comando: `git clone direccion URL` copiado en el paso anterior.

• ¿Cómo empujar cambios a un repositorio remoto?

- `git add .`
- `git commit -m "Comentario"`
- `git push origin nombre de la rama.`

• ¿Cómo tirar de cambios de un repositorio remoto?

Puede hacerse utilizando el comando `pull`.

• ¿Qué es un fork de repositorio?

Es copiar un repositorio remoto de otro usuario a tu cuenta de GitHub permitiendo desarrollar cambios sin afectar el original

• ¿Cómo crear un fork de un repositorio?

Siguiendo los siguientes pasos:

- Iniciar sesión con nuestra cuenta en GitHub
- En este caso tenemos un link con el repositorio a copiar. Se hace click en ese link.
- Se hace click en el botón fork.
- Agregar un texto en descripción.
- Hacer click en crear fork.

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

Se deben seguir los siguientes pasos:

- Hacer es un fork en el proyecto que deseamos colaborar.
- Clonar ese repositorio remoto a local para poder trabajar.
- Con Git crear una nueva rama y hacer las modificaciones que se consideren sean útiles para contribuir con el proyecto.
- Pasar esa rama local a remoto con un `git push origin nombre de la nueva rama`.
- En GitHub dentro de la nueva rama hacer click en la pestaña "Contribute" donde se habilita un botón para hacer un pull request.

- Al desplegarse un formulario agregar información sobre los cambios realizados y luego hacer click en el botón “Create pull request” para enviar la contribución al administrador del proyecto original.

• ¿Cómo aceptar una solicitud de extracción?

- En GitHub hacer click en la pestaña “Pull request” donde aparecerá la notificación del número de pull request recibidas pendientes de resolución.
- Al hacer click seleccionar el que se va a evaluar haciendo click sobre su nombre (en la pestaña “filed changes” se verán los cambios aportados).
- Para aceptar esa colaboración hacer click en el botón “Merge pull request” y luego en “Confirm merge” para unir los cambios al proyecto original.

• ¿Qué es una etiqueta en Git?

Las etiquetas o tags son referencias a puntos específicos en la historia de un proyecto y se utilizan para marcar versiones o momentos claves importantes del mismo como por ejemplo cuando se ha completado una funcionalidad, luego de solucionar errores críticos, también en la finalización de etapas del ciclo de vida del proyecto.

Hay 2 tipos de tags:

- Tags ligeras: Son tags simples y con poca información.
- Tags anotadas: Son almacenadas como objetos completos dentro de la base de Git y contienen más información..

También podemos crear una etiqueta con especificación a commit anteriores estando en el último. Utilizando el código SHA podemos especificar a dónde se va a aplicar la etiqueta.

• ¿Cómo crear una etiqueta en Git?

Para crear una etiqueta o tag se utiliza el comando git tag.

Para una etiqueta ligera se utiliza: git tag nombre de la etiqueta.

Para una etiqueta anotada se utiliza: git tag nombre de la etiqueta -m “Mensaje de la etiqueta”.

Para una especificación en etiqueta se utiliza: git tag nombre de la etiqueta -m “Mensaje de la etiqueta” código SHA del commit a donde se va a aplicar etiqueta.

• ¿Cómo enviar una etiqueta a GitHub?

Se envía de la siguiente forma:

- git push origin nombre de la etiqueta

Para enviar varias etiqueta a la vez se utiliza:

- git push origin -tags

• ¿Qué es un historial de Git?

Es donde se pueden ver todos los commit ordenados cronológicamente que se han realizado en un repositorio. El historial es muy útil porque permite ver la evolución del proyecto, así también cuándo y quienes realizaron cambios en el mismo.

• ¿Cómo ver el historial de Git?

Para ver el historial de commits se usa el comando: `git log`

Esta acción va a mostrar la lista de commits ordenados por fecha con sus especificaciones de fecha, mensaje, autor y el sha del mismo.

• ¿Cómo buscar en el historial de Git?

Se puede buscar de distintas maneras. Entre ellas:

- Incluyendo una palabra o frase que incluya el mensaje del commit.
Ej `git log --grep="palabra clave"`.
- Buscando commit que afecte un archivo en particular:
Ej `git log --<archivo>`
- También con: `git log --oneline` (muestra el mensaje y el código ASH en una sola línea)

• ¿Cómo borrar el historial de Git?

- Para borrar el último commit sin perder cambios:
`git reset --soft HEAD~1`
- Para borrar el último y los cambios también:
`git reset --hard HEAD~1`
- Para borrar un commit específico:
`git rebase -i HEAD~n`

Es un proceso delicado si se ha compartido el repositorio con otros colaboradores. Es recomendable coordinar con el resto del equipo para que este proceso no ocasione conflictos en el proyecto.

• ¿Qué es un repositorio privado en GitHub?

Es un repositorio con acceso restringido, sólo se puede acceder a él con permisos o invitaciones por parte del propietario del mismo.

• ¿Cómo crear un repositorio privado en GitHub?

- En el "Home" en GitHub hacer click en el botón "New" y se abre "Create a new repository"

- Agregar un nombre en “repository name”.
- Seleccionar Private.
- Hacer click en “Create new repository”.

• **¿Cómo invitar a alguien a un repositorio privado en GitHub?**

- Seleccionar el repositorio privado desde donde se considera hacer la invitación para añadir colaboradores.
- Hacer click en la pestaña “Settings”.
- En la parte de “Acces” hacer click en “Colaborators”
- Hacer click en el botón “Add people”.
- Se añade el colaborador seleccionandolo.

• **¿Qué es un repositorio público en GitHub?**

Es un repositorio al cual todo el mundo puede acceder sin necesidad de permisos especiales. Con lo cual pueden ver todo lo que hay en el mismo. También se puede clonar y hacer fork en el mismo.

• **¿Cómo crear un repositorio público en GitHub?**

- En el “Home” en GitHub hacer click en el botón “New” y se abre “Create a new repository”
- Agregar un nombre en “repository name”.
- Seleccionar “Public”..
- Hacer click en “Create new repository”.

• **¿Cómo compartir un repositorio público en GitHub?**

Copiar el enlace del repositorio y enviar a la persona con la que se quiere compartir el mismo.