

Title: Uso de Git

Keyword

Topic:

¿Qué es Git

- Proyecto
- desarrollo
- guardar
- repositorio
- grupos
- Versiones
- Control.

Git es una herramienta que usan los programadores o desarrolladores como un sistema de control de versiones (VCS), esto es, para guardar versiones del proyecto actual a medida que el equipo lo va desarrollando.

Questions

La utilidad del programa destaca en facilitar el desarrollo de un programa complejo mediante respaldos remotos a las que varios desarrolladores pueden acceder desde donde quieran y de forma segura, de modo que si surge una pérdida de datos o un error sin solución los desarrolladores tengan un nuevo punto de partida seguro.

Summary:

Git es un programa que permite guardar programas en desarrollo en espacios llamados repositorios, para que los programadores puedan guardar y compartir sus avances entre sí.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Seandro D. Arias	2/5	Carlos Pichardo	9/13/2023

Title: Uso de Git

Keyword

- Cambios
- Comando
- Git
- Commit
- add
- Repositorio
- Status
- Comentar
- Guardar
- Version

Questions

Topic: Comando "Commit"

El comando "Commit" nos permite registrar los cambios subidos en nuestra carpeta y comentarlos, dejándolos listos de ese modo para enviarlos a nuestro repositorio online.

Si no tenemos una carpeta seleccionada para el repositorio, basta con entrar a la deseada, y con el programa Git instalado damos click derecho y clickamos "Git bash".

Si hemos realizado cambios en la carpeta, podremos verlos escribiendo "Git status". Para guardar esos cambios, escribimos "Git add" para agregar y "Git commit" para comentar acerca de esos cambios. Guardamos y ya están listos para pushearse al repositorio.

Summary: Para recordar o aclarar los cambios realizados en una versión del proyecto es necesario comentar acerca de ellos. Por eso usamos "Commit" que, además de permitirnos comentar, hace posible la subida de los cambios al repositorio.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Leandro D. Arias	3/5	Carlos Pichardo	9/13/2023

Title: *Uso de Git*

Keyword	Topic: <i>Comando "Push"</i>
<ul style="list-style-type: none"> - Push - Subir - GitHub - Remote - online - sitio web 	<p>Después de tener nuestro cambio comentado y guardado, debemos subir nuestro desarrollo a un servidor remoto. El que usaremos será GitHub, un sitio web que nos permite crear y compartir un repositorio de forma remota. El cliente de escritorio de GitHub nos facilita realizar el commit y el push con solo unos clicks, luego de registrar la carpeta una vez, pero en la consola de Git se hace de otra manera también sencilla.</p> <p>Usando nuestra cuenta de GitHub creamos el repositorio del cual obtendremos un enlace. En la consola escribimos "git remote add origen (enlace anterior)" y ya quedamos registrados. Luego solo escribimos "git push (origen) (rama)" y el commit anterior se subirá al repositorio.</p>

Summary: El comando "Push" es el comando que sube el "commit" último al repositorio online creado en GitHub. Luego de agregar nuestra carpeta solo usamos "git push" para subir los cambios.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Leandro D. Ariza	4/5	Carlos Pichardo	9/13/2023

Title: Usode Git

Keyword	Topic: Clonación y Pull
<ul style="list-style-type: none"> - clone - pull - copia - actualización - disco - ubicación - cambios. 	<p>Naturalmente, guardamos el proyecto para usarlo después. Recordemos que lo que se guarda en el repositorio es una copia de nuestro proyecto. De ese modo, hay cosas importantes que podemos hacer con ellas: clonar el proyecto o actualizarlo, y para eso usamos "clone" y "pull", respectivamente.</p> <p>"Git clone" nos crea una copia de lo que tenemos en el repositorio en una ubicación específica de la página web o en nuestro disco, mientras que "git pull" actualiza el contenido en caso de que otros hayan realizado cambios.</p> <p>Lo he de destacar que ambos comandos convergen todas las commit realizadas en cada pull, por lo que se veen todos al dar un o fullear.</p>
Questions	

Summary: "Git clone" es un comando que crea una copia de lo que existe en el repositorio con todas sus versiones anteriores, en el almacén físico, y "git pull" actualiza el proyecto actual si se realizó cambios en el repositorio y no en el ordenador.

Title: Uso de Git

Keyword	Topic: <u>Ramas / Branches</u>
<ul style="list-style-type: none"> - Rama - Branche - Checkout - Paralelo - Comparar - Crear - Cambiar 	<p>Las ramas o "branches", por decirlo así, son versiones paralelas del proyecto que se está desarrollando actualmente. Esto es útil porque permite a los desarrolladores crear diferentes versiones con características diferentes a la original o versiones de prueba, sin que el foco original se pierda. Además, permite comparar dos versiones del programa con distinto desarrollo pero un mismo propósito.</p> <p>Para crear otra rama en la consola de git se escribe "git branch (nombre)", y para utilizarla se escribe "git checkout (nom)". De este modo, todo lo que se hace ahora será en la rama actual.</p>

Summary: Las branches nos permiten hacer pruebas y comparaciones sin perder el proyecto original, guardando las versiones alternativas a parte.