**ACTIVIDAD ACADÉMICA N°3**

**Apellido, Nombre**: Segovia, Mauricio

**Asignatura:** Diseño y Programación Web I

**Curso:** 1º Nivelatorio

**Carrera**: Lic. en Análisis de Sistemas Informáticos

**Fecha límite de entrega**: 26/03/2021 **Total de puntos:** 5

**Criterios a evaluar**

* Aplica reglas ortográficas (1p)
* Coherencia en las ideas (1p) (Caso contrario se anula todo el trabajo)
* Emite juicio crítico (1p)
* Realiza el trabajo en el formato establecido: repositorio\_nombreApellido.docx (1p)
* Realiza un commit del trabajo en el repositorio correspondiente (1p)

*OBS: una vez que se tenga toda la investigación en un documento word, se deberá subir al repositorio creado desde un commit. En la plataforma moodle se deberá entregar el enlace del repositorio donde se encuentra el trabajo.*

**Investigar sobre acciones que se realiza con un versionador de código. Explica: ¿En qué consiste un...?**

* **Commit**

Un Commit es, en simples palabras, un save file de la carpeta de nuestro proyecto en un momento determinado.

Cuando se hace un commit, se congela el estado de todos los ficheros y subdirectorios, y se guarda el archivo en un repositorio remoto (una actualización de los cambios realizados al trabajo). También puede hacerse al repositorio local.

* **Push**

Es un comando que sube los cambios hechos en el ambiente de trabajo a una rama de trabajo propia o remota. A nivel de trabajo “git *push*” trabaja a nivel de repositorio, es decir con el repositorio remoto.

Permite subir los commits desde la rama (*branch*) local en el repositorio git local al repositorio remoto.

Antes de poder subir al repositorio remoto, hay que asegurar de hacer *commit* a todos los cambios al repositorio local.

* **Branch (Ramas)**

Cuando se trabaja con repositorios remotos, Git usa ramas remotas para mantener copias de los datos en el repositorio remoto.

Todos los commits se aplican sobre una rama. Por defecto, siempre se empieza en una rama master y a partir de ella se crean nuevas ramas para crear flujos de trabajo independientes.

Crear una nueva rama se trata de copiar un commit de cualquier rama, pasarlo a otra rama y continuar el trabajo de una parte específica del proyecto sin afectar el flujo de trabajo principal, el cual continúa en la rama master.

Los equipos de desarrollo tienen un estándar: Todo lo que esté en la rama master va a producción, las nuevas features, características y experimentos van en una rama “development” (para unirse a master cuando estén definitivamente listas) y los issues o errores se solucionan en una rama “hotfix” para unirse a master tan pronto como sea posible.

* **Fetch**

*git fetch* es el comando que le dice al git local que recupere la última información de los metadatos del original.

Se utiliza este comando cuando el usuario necesita revisar qué cambios se han producido en un repositorio remoto antes de realizar una fusión o **merge. Todo esto sin realizar ninguna transferencia de archivos.**

Se puede usar *git fetch* para saber los cambios hechos en el repo/rama remota desde el último pull. Esto es útil para permitir la comprobación antes de hacer un pull real, lo que podría cambiar los archivos en tu rama actual y en la copia de trabajo, potencialmente perdiendo los cambios realizados.

* **Merge**

En git existen dos formas que permiten unir ramas, una de ellas es *git merge*. Esta realiza una fusión de las dos últimas instantáneas de cada rama a la rama original, creando un nuevo commit con los cambios mezclados.

Git merge combinará múltiples secuencias de commits en una simple unificada historia. En los casos más comunes de uso, git merge es usado para combinar 2 ramas.

* **Pull**

*“git pull”* se emplea para extraer y descargar contenido desde un repositorio remoto y actualizar al instante el repositorio local para reflejar ese contenido

Este comando es una combinación de dos comandos: *“git fetch”* y *“git Merge*

En la primera etapa de la operación, *git pull* ejecutará un *git fetch* en la *branch* local a la que apunta HEAD, esta descarga el contenido del repositorio remoto especificado.

Una vez descargado el contenido, *git pull* entrará en un flujo de trabajo de fusión. Se creará una nueva confirmación de fusión y se actualizará HEAD para que apunte a la nueva confirmación.

Después, se ejecuta git merge para fusionar las referencias y los encabezados del contenido remoto en una nueva confirmación de fusión local.