**Versionamiento**

Nombre: Laura Tatiana Gomez Castillo.

Brayan Gomez Lopez.

Ficha: 2202764

Tecnólogo en Análisis y Desarrollo de Sistemas de Información

Instructora: Tatiana Tovar

Servicio nacional de aprendizaje SENA

Centro de electricidad electrónica y telecomunicaciones

Bogotá D.C

Febrero 12 de 2021

# 

# 

# 

**Tabla de contenido**

[**Actividades de reflexión Inicial**](#_u43kzg5z0das) **3**

[1.1 ¿Qué se puede hacer para deshacer estos cambios sin afectar lo que si funcionaba bien en la página?](#_hribxwinhiud) 3

[**Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje.**](#_v0uu5ix1y7sy) **3**

[2.1 ¿Qué es un repositorio y mencione 2 ejemplos de los mismos (proveedores online)?](#_v2dxjb3oqddu) 3

[2.2 ¿Cuáles son los principales componentes de un versionamiento en la herramienta GIT?](#_z2y56p64qflw) 4

[2.2.1 Recuperar un commit erróneo](#_5mk5qx884sd5) 4

[2.2.2 Registro de referencias (reflog)](#_wa1qey7sl5ez) 5

[**2.2.3 Dejar de lado los cambios con git stash**](#_6u35g86bkjy8) **5**

[2.3 ¿Menciona con sus propias palabras las ventajas que tiene GIT frente a otros proveedores de repositorios?](#_6jd9m4stb9gc) 5

[2.4 Mencione por lo menos 5 ejemplos de los comandos básicos que se usan en GIT.](#_phoutz1md780) 6

[2.5¿Qué son y cuáles son las funciones de los branch?](#_uwvnpcvcif85) 7

[2.6 ¿Cuál es el nombre del branch principal?](#_ypuinog5gfdz) 8

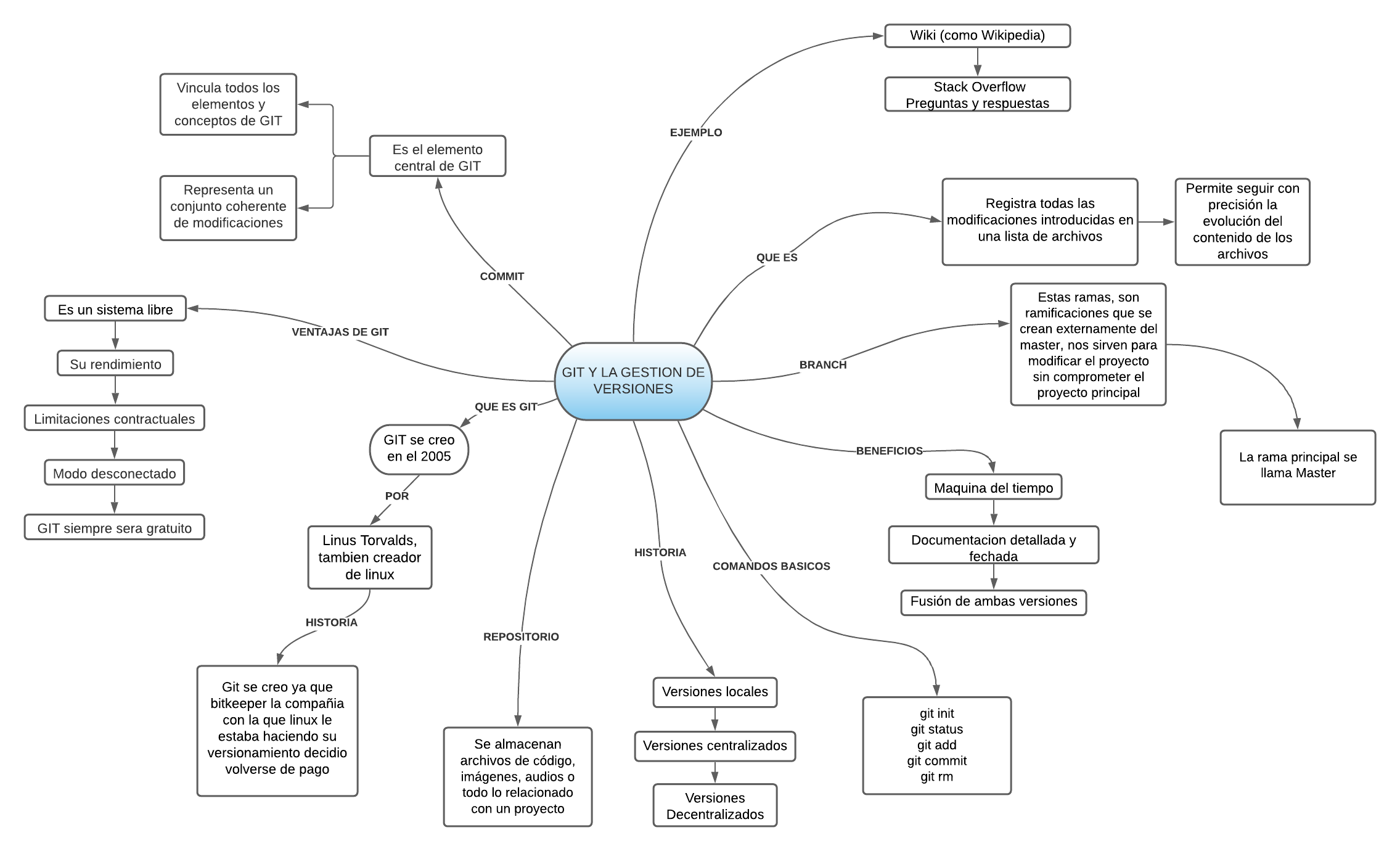
**BIBLIOGRAFÍA**

# Actividades de reflexión Inicial

## 1.1 ¿Qué se puede hacer para deshacer estos cambios sin afectar lo que si funcionaba bien en la página?

Cuando se está creando un software se debe crear una carpeta extra, la llamada “copia de seguridad ” en nuestro desarrollador debía estar trabajando pero en nuestro caso nuestro programador a modificado la versión directa del programa, lo unico que nos queda por hacer en estos casos es volver a una versión anterior donde dichos cambios todavía no se han realizado y de ese momento volver a realizar las correcciones y subir el archivo correcto, aparte tiene que decirle a su jefe dicho error para que las cosas no pasen a mayores. (Fuente propia)

# **Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje.**



<https://lucid.app/lucidchart/8d63a07f-dbe0-4e57-a532-a62e691b28ea/edit?shared=true&page=0_0#?folder_id=home&browser=icon>

## 2.1 ¿Qué es un repositorio y mencione 2 ejemplos de los mismos (proveedores online)?

Un repositorio es un directorio donde se almacenan los archivos de un proyecto. Puede estar ubicado en el almacenamiento de GitHub o en un repositorio local en una computadora. Te permite guardar versiones del código a las que puedes acceder cuando lo necesites. Puede almacenar archivos de código, imágenes, audios o todo lo relacionado con el proyecto en el repositorio. (Gustavo B., 2019)

Ejemplos.

* Youtube : Este producto se va actualizando constantemente teniendo diferentes versiones con correcciones menores y mayores.
* DSpace: Es un software de código abierto que provee herramientas para la administración de colecciones digitales, y comúnmente es usada como solución de repositorio bibliográfico institucional.

## 

## 2.2 ¿Cuáles son los principales componentes de un versionamiento en la herramienta GIT?

### 2.2.1 Recuperar un commit erróneo

En cualquier tipo de desarrollo, a veces ocurre que un desarrollador agrega una regresión en su código. Una regresión significa que, durante el desarrollo de una funcionalidad o en la corrección de un bug, las modificaciones del desarrollador han supuesto hacer que otra parte del software no funcione. Las regresiones se añaden sin que el desarrollador se dé cuenta, y si no se requiere una prueba unitaria o funcional para detectar este tipo de error, entonces ésta sigue vigente hasta que un usuario o un desarrollador aborde el problema.

Git ayuda al desarrollador a encontrar el commit que añadió la regresión mediante una búsqueda dicotómica. En cada etapa de git bisect, Git sustituirá el directorio de trabajo por una versión anterior y habrá que indicar si el error sigue presente. Limita entonces el número de commits que pueden añadir el error y luego realiza una iteración modificando de nuevo el directorio de trabajo. (Dauzon, Mayo 2018, #)

### 2.2.2 Registro de referencias (reflog)

Es un registro que va a almacenar todos los commits por donde pase el puntero. El **reflog**, por ejemplo, será capaz de guardar los cambios de rama o la adición de un commit. (Dauzon, Mayo 2018, #)

# *2.2.3 Dejar de lado los cambios con git stash*

Se trata de un comando que “congela” el estado en el que se encuentra el proyecto en un momento determinado, con todos los cambios que tenemos a "sin comitear", y lo guarda en una pila provisional brindando la posibilidad de poder recuperarlo más adelante. (Dauzon, Mayo 2018, #)

## 2.3 ¿Menciona con sus propias palabras las ventajas que tiene GIT frente a otros proveedores de repositorios?

1. La integridad de la información está asegurada gracias a su modelo de almacenamiento, que permite predecir este tipo de problemas.
2. GIT permite utilizar flujos de trabajo más flexibles que te ayudan a trabajar mejor y con menos estrés en general.
3. Cualquier búsqueda que se ejecute será muchísimo más eficaz.
4. Permite trabajar sin internet.
5. Facilita el trabajo colaborativo ya que distintos programadores pueden estar editando el mismo archivo, o versiones distintas del mismo archivo, y todos los cambios serán reflejados en el documento final.
6. Permite regresar a versiones anteriores de forma sencilla y muy rápida.
7. No puede haber limitaciones contractuales sobre el uso de Git.
8. Es posible trabajar en modo desconectado con su repositorio manteniendo las características avanzadas de colaboración cuando se está conectado.
9. Git incorpora muchas herramientas internas para mejorar su rendimiento. (Fuente propia) (Dauzon, Mayo 2018, #)

## 2.4 Mencione por lo menos 5 ejemplos de los comandos básicos que se usan en GIT.

1. git init. Con este comando de git lo que estamos haciendo es decirle a git que esté pendiente de los cambios que se produzcan en los archivos de ese directorio. Este comando solo lo tenemos que ejecutar una sola vez para cada proyecto que estemos realizando.
2. git status. Este comando git imprimirá los archivos que van a ser guardados. Muestra la lista de los archivos que se han cambiado junto con los archivos que están por ser preparados o confirmados.
3. git add. Este comando agrega todos los archivos que aparecían anteriormente para ser guardados.
4. git rm. Este comando elimina los archivos que acabamos de añadir para ser guardados.
5. git commit -m "Nombre descriptivo del cambio que hemos realizado" un commit es un guardado con mensaje de los cambios que hemos realizado en un momento determinado en nuestro proyecto. (Coding Potions, 2018)

## 2.5 ¿Qué son y cuáles son las funciones de los branch?

Estos branch o en español rama, son ramificaciones que se crean externamente del master (carpeta llamada dev), estas ramas nos sirven para modificar el proyecto sin comprometer el proyecto principal, es donde los desarrolladores pueden crear nuevas funciones en el programa, corregir errores. Según el rol de cada desarrollador. (Dauzon, Mayo 2018, #)

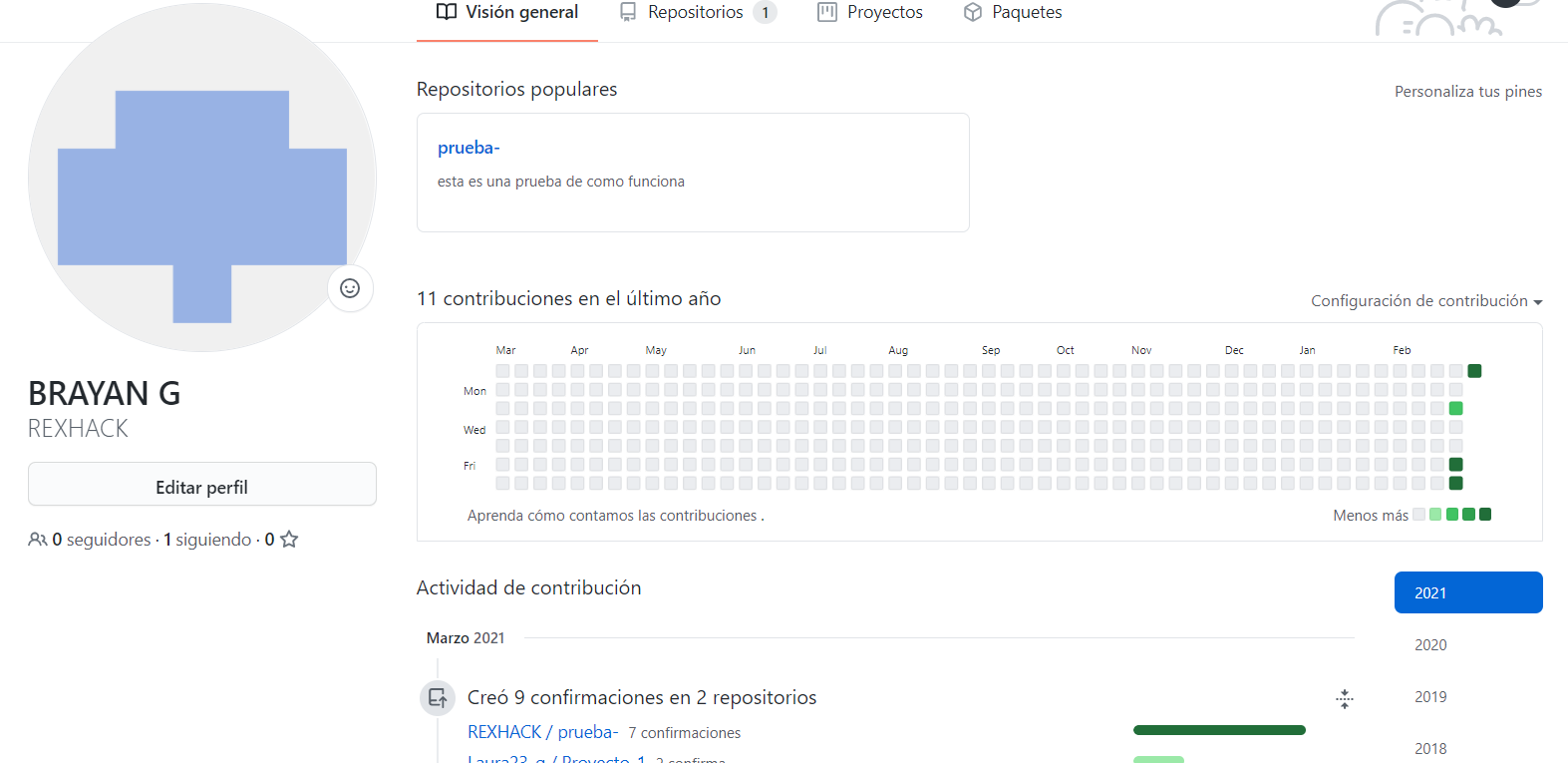
Estas ramas deben volver a la rama dev y probar que no rompa el sistemas o la aplicación, si todo está bien y no daño la carpeta dev se puede unir a la carpeta del proyecto principal (master). (Dauzon, Mayo 2018, #)

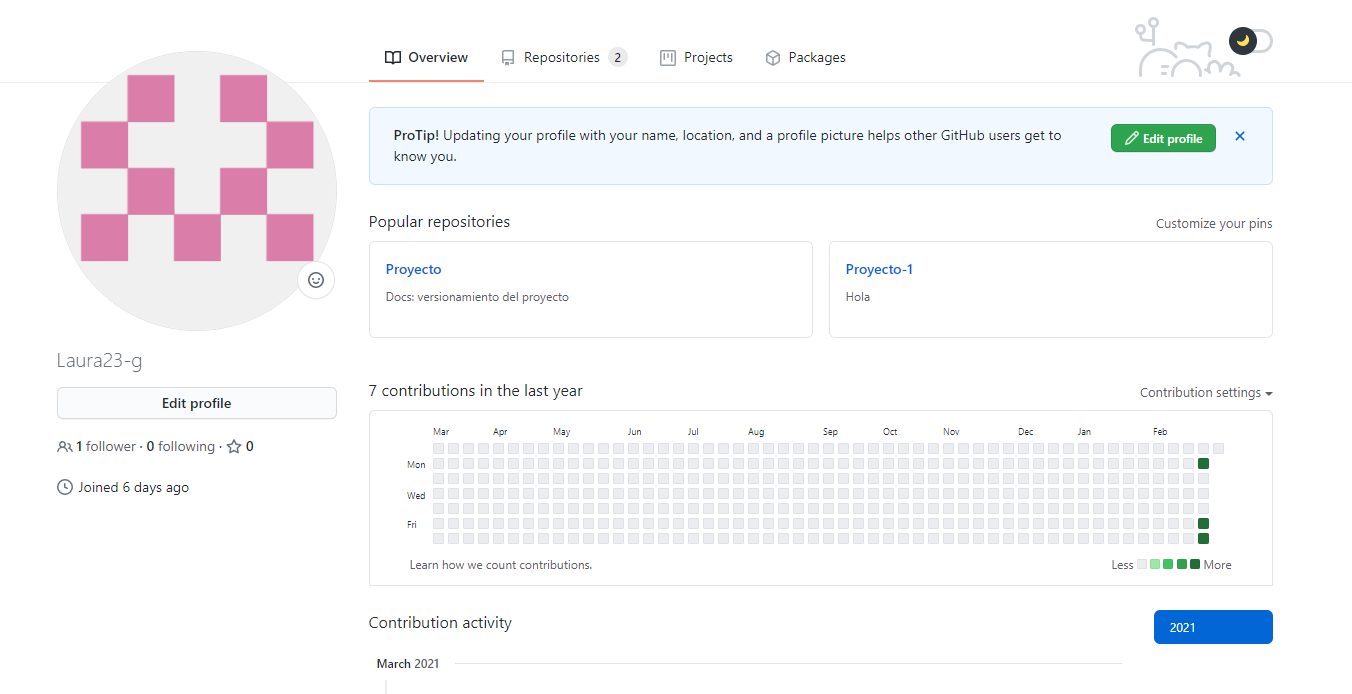
## 2.6 ¿Cuál es el nombre del branch principal?

Cuando comenzamos un proyecto en git lo primero sería crear un proyecto y git le realizará un repositorio y como sabemos de hay se crea la rama principal llamada “master” que es la rama principal que genera.

Esta rama es la que los usuario estarán interactuando, mientras los programadores se les colocara una rama externa llamada dev que es donde se harán las correcciones o la implementación de nuevos funciones que luego se llevarán a “master”. (Dauzon, Mayo 2018, #)

# 3. Actividades de Apropiación del conocimiento

* Crear cuenta en GitHub



* Clonar el repositorio de forma local en el equipo de trabajo
* Realizar el Readme definiendo el componente metodológico de su proyecto Individual.
* Crear un Branch llamado testing
* Crear por lo menos 3 Commits subiendo los trabajos realizados en su proyecto.

**Bibliografía**

Coding Potions. (2018, 11). *Comandos básicos que se usan en GIT*. Coding Potions. Retrieved 02 12, 2021, from https://codingpotions.com/git-comandos-basicos-avanzados

Dauzon, S. (Mayo 2018). *Git - Controle la gestión de sus versiones (conceptos, utilización y casos prácticos)* (Digital ed., Vol. 1). Django Essentials. https://www-eni-training-com.bdigital.sena.edu.co/portal/client/mediabook/home

Gustavo B. (2019, 05 13). *Repositorio*. Hostinger. Retrieved 02 12, 2021, from https://www.hostinger.co/tutoriales/que-es-github