

Universidad de Nariño.

Ingeniería de Sistemas.

**Diplomado de actualización en nuevas tecnologías para el
desarrollo de Software.**

Taller Unidad 1 GIT.

Presentado: Fredilton Daniel Getial Torres

219036091

3187863224

El presente taller debe desarrollarse de manera individual.

1. Crear un repositorio local en su sistema de archivos con el proyecto de ejemplo Taller0003Git proporcionado en clase, el cual contiene el código de una "Calculadora" desarrollada en JavaScript, una vez creado el repositorio crear un Commit inicial.

Creación del repositorio



Contenido repositorio

Nombre	Fecha de modificación	Tipo	Tamaño
.git	30/11/2023 12:20	Carpeta de archivos	
css	30/11/2023 12:20	Carpeta de archivos	
js	30/11/2023 12:20	Carpeta de archivos	
index	30/11/2023 12:20	Chrome HTML Do...	5 KB
LICENSE	30/05/2019 0:27	Archivo	2 KB
README	30/05/2019 0:27	Archivo de origen ...	1 KB

Commit inicial

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Tall
er0003Git (main)
$ git commit -m "Primer commit subir repositorio"
```

Visualizar repositorio

```
commit 96206a1efedd77e696a85900076e592d7e80cb3c
Author: danielge149 <danielgetial149@gmail.com>
Date: Sat Nov 25 10:08:48 2023 -0500

    Primer commit subir repositorio
```


2. Crear un repositorio remoto en GitHub con el mismo nombre del repositorio local es decir "Taller0003Git", enviar el avance del repositorio local al remoto.

Creación repositorio remoto

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)



Required fields are marked with an asterisk (*).

Owner *  danielge149 / Repository name *

✓ Taller0003Git. is available.

Great repository names are short and memorable. Need inspiration? How about [cautious-octo-funicular](#) ?

Description (optional)

- ☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

- ☐ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.

Create repository

Subir repositorio local a remoto

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git remote add origin https://github.com/danielge149/Taller0003Git.git
```

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git push -u origin main
```

Visualizar el repositorio local en el repositorio remoto

The screenshot shows the GitHub interface for a repository named 'Taller0003Git', which is public. At the top, there are buttons for 'Pin' and 'Unwatch'. Below this, a navigation bar shows the 'main' branch selected, with 3 branches and 0 tags. There are buttons for 'Go to file', 'Add file', and 'Code'. A message states 'Your main branch isn't protected' with a 'Protect this branch' button. The commit history shows a commit by 'danielge149' titled 'commit nuevas funciones' from last week, with 4 commits in total. Below the commit history, a table lists files: 'css' and 'js' (both committed last week), and 'LICENSE', 'README.md', and 'index.html' (all committed 2 weeks ago). At the bottom, there is a section for 'README.md' with an edit icon.

3. En el repositorio local crear una rama llamada "diseno", en ella actualizar las referencia a Bootstrap en su versión mas actual y modificar el diseño visual del proyecto de tal forma que tenga una interfaz gráfica más agradable, realizar los commit que sean necesarios.

Creación de rama

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git branch diseno
```

Visualización de rama

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git branch
diseno
```

Cambio de rama

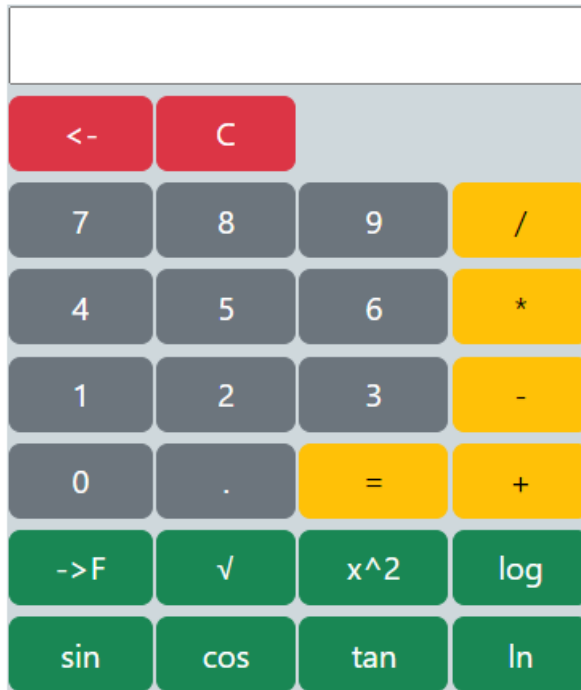
```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git checkout diseno
Already on 'diseno'
Your branch is up to date with 'origin/diseno'.

danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (diseno)
$
```

Actualización de Bootstrap

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet"
  integrity="sha384-T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN" crossorigin="anonymous">
```

Nuevo diseño (nuevas funciones para practicar)



Commit diseño

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (diseno)
$ git log
commit 89cdee7a6786fff8039dfbd0e52da24f57d501ee (HEAD -> diseno, origin/diseno)
Author: danielge149 <danielgetial149@gmail.com>
Date: Sat Nov 25 11:04:11 2023 -0500

    commit diseño listo
```

4. Una vez terminado el diseño realizar un merge (fusión) de los cambios realizados en la rama "diseno" a la rama "main".

Fusión código principal

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git merge diseno
Auto-merging index.html
Merge made by the 'ort' strategy.
 index.html | 6 ++----
 1 file changed, 2 insertions(+), 4 deletions(-)
```

5. Crear una rama llamada "funcionalidad" en la cual agregaremos un botón al final de la GUI del proyecto que me permita convertir grados Centígrados a Fahrenheit, realizar en esta rama los commit que sean necesarios.

Creación rama

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (main)
$ git branch funcionalidad
```

Visualización de rama

```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (funcionalidad)
$ git branch
  disenio
  funcionalidad
* main
```

Nuevas funciones (incremente otras funciones por practicar JS)

```
<tr>
  <td ><input type="button" id="btnConvertir" class="btn btn-success" value="->F" /></td>
  <td ><input type="button" value="√" id="btnRaiz" class="btn btn-success"/></td>
  <td ><input type="button" value="x^2" id="btnCuadrado" class="btn btn-success"/></td>
  <td ><input type="button" value="log" id="btnLog" class="btn btn-success"/></td>
</tr>
<tr>
  <td ><input type="button" value="sin" id="btnSeno" class="btn btn-success"/></td>
  <td ><input type="button" value="cos" id="btnCoseno" class="btn btn-success"/></td>
  <td ><input type="button" value="tan" id="btnTangente" class="btn btn-success"/></td>
  <td ><input type="button" value="ln" id="btnLn" class="btn btn-success"/></td>
</tr>
```

```
function convertirTemperatura(){
  // Obtener el valor en grados Celsius
  var grados= parseFloat(document.getElementById("tela").value);

  // Convertir a Fahrenheit
  var fahrenheit = (grados * 9/5) + 32;

  // Mostrar el resultado en el campo de texto
  document.getElementById("tela").value = fahrenheit.toFixed(2);
}
```

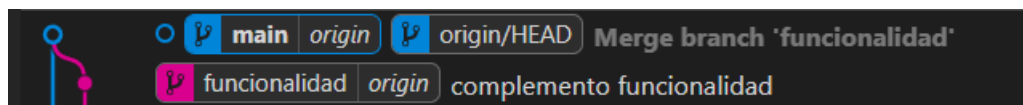
NOTA: las demás visualizar en código

Commit funcionalidad

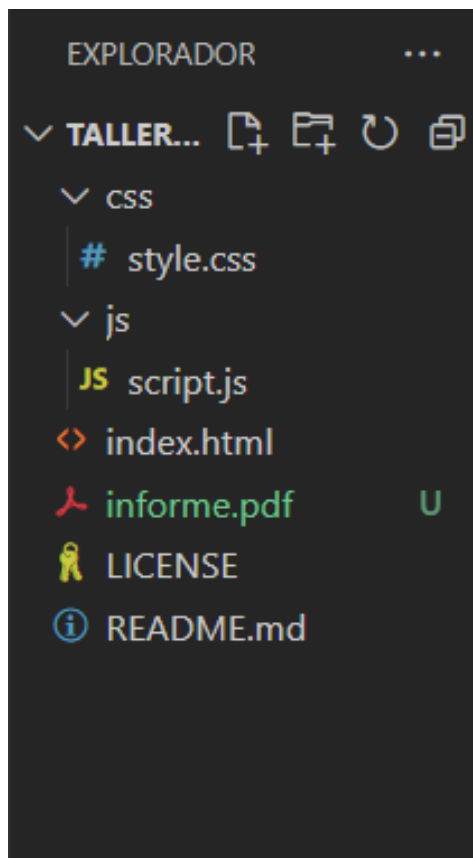
```
danie@DanielGetial MINGW64 ~/Desktop/diplomado archivos/Unidad 1/Taller GIT/Taller0003Git (funcionalidad)
$ git commit -m "complemento funcionalidad"
[funcionalidad b0e82e2] complemento funcionalidad
1 file changed, 2 insertions(+)
```

6. Una vez terminada la funcionalidad realizar un merge (fusión) de los cambios realizados en la rama "funcionalidad" a la rama "main".

Fusión rama funcionalidad (se utilizó visual ya que en la anterior fue por consola)



7. Actualizar el repositorio remoto y realizar un pequeño informe presentado de manera individual en el cual se evidencie por medio de capturas de pantalla el trabajo realizado, el informe debe ser almacenado como un archivo PDF dentro de la carpeta principal del proyecto con el nombre "informe.pdf"



CONCLUSIÓN

En este taller de Git, he adquirido habilidades esenciales para la gestión eficiente de proyectos de desarrollo de software. Desde la creación de repositorios locales y remotos hasta la implementación de ramas para trabajar en nuevas funcionalidades y mejoras de diseño, he comprendido la importancia de un control de versiones estructurado. La capacidad de fusionar ramas y resolver conflictos me ha proporcionado una comprensión más profunda de la colaboración en proyectos. La documentación mediante un informe en PDF destaca la necesidad de registros claros. En conjunto, este taller ha fortalecido mi confianza para enfrentar proyectos más complejos y colaborar efectivamente en equipos de desarrollo.