



Ciudad Universitaria
San Lorenzo - Paraguay

UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA

DIRECCIÓN ACADÉMICA
DEPARTAMENTO DE POSTGRADO

Módulo ***Sistema de Información Web***

Reporte técnico **Uso de Git y GitHub**

Profesor: **Julio César Mello Roman**

Alumno: **Guillermo Palacios**

Año 2025

INDICE

Tabla de contenido

INTRODUCCIÓN	3
2.Instalación de Git.....	3
3. Configuración del Usuario	3
4.Creación de una Cuenta en GitHub	4
5.Creación del Repositorio en GitHub	4
7. Creación de una Rama a partir de la Rama Principal	5
8. Creación de Carpetas y Levantamiento de Archivos	5
9. Subir los Cambios a GitHub	6
10. Solicitar un Merge Request	6
11. Mezclar la Rama con la Rama Principal	6
12.CONCLUSIÓN	7
13.REFERENCIAS	7

INTRODUCCIÓN

Este reporte detalla el proceso de instalación y configuración de Git, así como el uso de GitHub para la gestión de versiones en un proyecto denominado *Sistemas de Información Web*. Se describen los pasos desde la instalación del cliente Git hasta la solicitud y fusión de cambios en el repositorio.

2.Instalación de Git

Para utilizar Git, es necesario instalar el cliente en el ordenador. Los pasos son los siguientes:

Descargar el instalador desde la página oficial: <https://git-scm.com/downloads>.

Ejecutar el instalador y seguir las instrucciones predeterminadas.

Verificar la instalación con el siguiente comando en la terminal o símbolo del sistema:

git --version

Si Git está correctamente instalado, se mostrará la versión instalada.

3. Configuración del Usuario

Una vez instalado Git, es necesario configurar la información del usuario en los repositorios locales. Para ello, se utilizan los siguientes comandos:

`git config --global user.name "guillermop"`

`git config --global user.email "gpcios.65@gmail.com"`

Para verificar la configuración:

```
git config --global --list
```

Esto mostrará los datos configurados.

4.Creación de una Cuenta en GitHub

Para gestionar el repositorio en línea, se debe crear una cuenta en GitHub:

Acceder a <https://github.com/>.

Registrarse con un correo electrónico y establecer una contraseña.

Verificar el correo y completar el proceso de configuración.

5.Creación del Repositorio en GitHub

Una vez dentro de GitHub, se crea un repositorio para el proyecto:

Hacer clic en **New repository**.

Asignar el nombre *Sistemas de Información Web*.

Seleccionar la opción *Public* o *Private* según sea necesario.

Inicializar el repositorio con un archivo README.md y hacer clic en **Create repository**.

6.Clonar el Proyecto en la Máquina Local

Para trabajar en el repositorio desde el ordenador, se debe clonar el proyecto:

Copiar la URL del repositorio en GitHub.

Abrir la terminal y ejecutar el siguiente comando:

```
git clone https://github.com/usuario/Sistemas-de-
Informacion-Web.git
```

Acceder a la carpeta clonada:

cd Sistemas-de-Informacion-Web

7. Creación de una Rama a partir de la Rama Principal

Para mantener organizadas las modificaciones, se crea una nueva rama:

git checkout -b feature/diseñoweb

Este comando crea y cambia a la nueva rama.

8. Creación de Carpetas y Levantamiento de Archivos

Dentro del repositorio clonado, se crean las carpetas para organizar los archivos:

- Crear la carpeta **Semana 01**:

mkdir "Semana 01"

- Se copian dentro todos los ejercicios de la **Tarea 1 y Tarea 2**.
- Crear la carpeta **Semana 02**:

mkdir "Semana 02"

- Se agregan los ejercicios del material de JavaScript.
- Añadir los cambios al control de versiones:

git add .

- Confirmar los cambios:

git commit -m "Añadiendo ejercicios de la Semana 01 y Semana 02"

9. Subir los Cambios a GitHub

Para reflejar los cambios en el repositorio remoto, se utiliza:

```
git push origin nueva-rama
```

Esto sube la rama con los archivos nuevos a GitHub.

10. Solicitar un Merge Request

Una vez subida la rama, se realiza una solicitud de fusión (*Merge Request*):

- Ingresar a GitHub y abrir el repositorio.
- Ir a la pestaña **Pull requests** y hacer clic en **New pull request**.
- Seleccionar la rama principal y la nueva rama creada.
- Agregar una descripción y hacer clic en **Create pull request**.

11. Mezclar la Rama con la Rama Principal

Después de la aprobación del *Merge Request*, se fusionan los cambios con la rama principal:

Dentro de GitHub, en el pull request, hacer clic en **Merge pull request**.

Confirmar la fusión.

En la terminal, actualizar el repositorio local con:

```
git checkout main
```

```
git pull origin main
```

12.CONCLUSIÓN

Este reporte garantiza una correcta gestión de versiones en proyectos de desarrollo web, permitiendo organizar y actualizar los archivos de manera eficiente con Git y GitHub.

13.REFERENCIAS

- MDN Web Docs. (2024). *CSS Selectors*.
https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Selectors
- W3Schools. (2024). *CSS Specificity*.
https://www.w3schools.com/css/css_specificity.asp
- Bootstrap. (2024). *Introduction to Bootstrap*.
<https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- Tailwind CSS. (2024). *Documentation*,
<https://tailwindcss.com/docs>