**Guía Detallada para Trabajar con GitHub Desktop**

**Paso 1: Instalación de Git y GitHub Desktop**

**1.1 Instalar Git:**

1. Ve a [Git - Descarga](https://git-scm.com/downloads) y descarga la versión adecuada para tu sistema operativo (Windows, macOS, Linux).
2. Ejecuta el instalador y sigue las instrucciones. Acepta las configuraciones predeterminadas.

**1.2 Instalar GitHub Desktop:**

1. Ve a [GitHub Desktop - Descarga](https://desktop.github.com/) y descarga el instalador.
2. Ejecuta el instalador y sigue las instrucciones.

**Paso 2: Configuración de Git**

**2.1 Configurar tu nombre de usuario y correo electrónico:**

1. Abre Git Bash (instalado con Git) o tu terminal preferida.
2. Configura tu nombre de usuario y correo electrónico (asegúrate de usar el mismo correo que utilizas en GitHub):

bash

Copiar código

git config --global user.name "Tu Nombre"

git config --global user.email "tuemail@example.com"

**Paso 3: Clonar el Repositorio en tu PC**

**3.1 Obtener la URL del Repositorio:**

1. Ve a la página del repositorio en GitHub.
2. Haz clic en el botón verde "Code" y copia la URL del repositorio.

**3.2 Clonar el Repositorio usando GitHub Desktop:**

1. Abre GitHub Desktop.
2. Ve a "File" > "Clone repository".
3. Selecciona la pestaña "URL" y pega la URL del repositorio que copiaste.
4. Selecciona la ubicación donde deseas clonar el repositorio en tu PC y haz clic en "Clone".

**Paso 4: Trabajar con el Repositorio**

**4.1 Añadir Archivos y Realizar Cambios:**

1. Añade tus archivos o realiza cambios en los archivos existentes dentro de la carpeta del repositorio en tu PC.

**4.2 Ver los Cambios en GitHub Desktop:**

1. Abre GitHub Desktop y verás los cambios listados en la pestaña "Changes".

**4.3 Hacer Commit de los Cambios:**

1. Escribe un mensaje descriptivo para tu commit en el campo "Summary".
2. Haz clic en "Commit to main" (o el nombre de la rama que estés usando).

**4.4 Subir Cambios (Push) al Repositorio Remoto:**

1. Después de hacer commit, haz clic en "Push origin" en GitHub Desktop para subir tus cambios al repositorio en GitHub.

**Paso 5: Sincronizar Cambios con el Repositorio Remoto**

**5.1 Obtener los Últimos Cambios del Repositorio (Pull):**

1. Antes de comenzar a trabajar, asegúrate de hacer Fetch origin y luego Pull origin para sincronizar los cambios.

**5.2 Resolver Conflictos:**

1. Si dos miembros del equipo modifican el mismo archivo, puede haber conflictos al hacer Push. GitHub Desktop te notificará y te permitirá resolverlos manualmente.

**Paso 6: Colaboración y Gestión de Acceso**

**6.1 Invitar a Colaboradores:**

1. Ve a la página del repositorio en GitHub.
2. Haz clic en "Settings" > "Manage access".
3. Haz clic en "Invite a collaborator".
4. Ingresa los nombres de usuario de GitHub de tus compañeras y envía las invitaciones.

**6.2 Aceptar Invitaciones:**

1. Tus compañeras recibirán una invitación que deben aceptar para tener acceso al repositorio.

**Flujo de Trabajo Resumido**

1. **Clonar el Repositorio:**
   * Cada miembro del equipo debe clonar el repositorio siguiendo el Paso 3.
2. **Sincronizar Cambios:**
   * Antes de empezar a trabajar, asegúrate de hacer Pull para obtener la última versión del código.
   * Realiza cambios y haz Commit regularmente.
   * Usa Push para subir tus cambios al repositorio remoto.
3. **Revisar y Resolver Conflictos:**
   * Si surgen conflictos, GitHub Desktop te notificará y podrás resolverlos manualmente antes de hacer Push.

**Resumen de Comandos Básicos en la Terminal (Opcional)**

* **Clonar un Repositorio:**

bash

Copiar código

git clone https://github.com/usuario/repositorio.git

* **Verificar el Estado de los Archivos:**

bash

Copiar código

git status

* **Añadir Archivos al Área de Preparación:**

bash

Copiar código

git add .

* **Hacer Commit con un Mensaje:**

bash

Copiar código

git commit -m "Mensaje descriptivo del commit"

* **Subir Cambios al Repositorio Remoto:**

bash

Copiar código

git push origin main

* **Obtener los Últimos Cambios del Repositorio Remoto:**

bash

Copiar código

git pull origin main

## Guía para Trabajar con Ramas en GitHub

### 1. Crear una Rama para Cada Integrante

1. \*\*Abrir GitHub Desktop:\*\*

- Abre GitHub Desktop y asegúrate de que el repositorio correcto esté seleccionado.

2. \*\*Crear una Nueva Rama:\*\*

- En la esquina superior izquierda, haz clic en el menú desplegable que muestra el nombre de la rama actual (por defecto, `main`).

- Haz clic en "New Branch" (Nueva Rama).

- Nombra la nueva rama de acuerdo con el trabajo que realizarás o con tu nombre. Por ejemplo:

- `rama-ana`

- `rama-beatriz`

- `rama-carla`

- `rama-diana`

- Haz clic en "Create Branch" (Crear Rama).

3. \*\*Publicar la Rama:\*\*

- Después de crear la rama, haz clic en "Publish Branch" (Publicar Rama) en GitHub Desktop para subirla al repositorio remoto.

### 2. Trabajar en tu Rama

1. \*\*Cambiar a tu Rama:\*\*

- Asegúrate de estar en tu rama seleccionándola en el menú desplegable de ramas en GitHub Desktop.

2. \*\*Realizar Cambios:\*\*

- Realiza los cambios necesarios en tu proyecto en tu rama local.

- Guarda los cambios en los archivos.

3. \*\*Hacer Commit de los Cambios:\*\*

- En GitHub Desktop, verás los cambios listados en la pestaña "Changes".

- Escribe un mensaje descriptivo para tu commit en el campo "Summary".

- Haz clic en "Commit to [nombre de tu rama]".

4. \*\*Subir Cambios (Push) a la Rama Remota:\*\*

- Después de hacer commit, haz clic en "Push origin" en GitHub Desktop para subir tus cambios a tu rama en GitHub.

### 3. Integrar Cambios al `main`

1. \*\*Crear un Pull Request:\*\*

- Una vez que hayas terminado de trabajar en tu rama y estés lista para integrar los cambios en `main`, ve a la página del repositorio en GitHub.

- Haz clic en el botón "Pull requests".

- Haz clic en "New pull request".

- Selecciona tu rama como la rama de comparación y `main` como la base.

- Revisa los cambios y haz clic en "Create pull request".

2. \*\*Revisión del Pull Request:\*\*

- Las demás integrantes del equipo deben revisar el pull request y aprobar los cambios.

- Una vez aprobado, haz clic en "Merge pull request" para integrar los cambios en `main`.

3. \*\*Eliminar la Rama (Opcional):\*\*

- Después de integrar los cambios, puedes eliminar tu rama si ya no la necesitas. En la página del pull request, haz clic en "Delete branch".

### 4. Qué Debe Ir en el `main`

El `main` debe contener el código estable y completamente funcional. Esto incluye:

1. \*\*Versiones Finalizadas de Funcionalidades:\*\*

- Solo las funcionalidades que han sido completamente desarrolladas, probadas y revisadas deben integrarse en `main`.

2. \*\*Código Estable:\*\*

- Asegúrate de que el código en `main` siempre compile y funcione correctamente sin errores.

3. \*\*Documentación:\*\*

- La documentación importante y actualizada también debe mantenerse en `main`.

4. \*\*Versiones de Lanzamiento:\*\*

- Las versiones que están listas para ser desplegadas o entregadas deben estar en `main`.

### Resumen de Pasos para Crear y Trabajar en una Rama

1. \*\*Crear una Rama:\*\*

- Abre GitHub Desktop > Selecciona el menú de ramas > "New Branch" > Nombra la rama > "Create Branch" > "Publish Branch".

2. \*\*Trabajar en tu Rama:\*\*

- Cambia a tu rama > Realiza cambios > Haz commit de los cambios > Haz push a la rama remota.

3. \*\*Integrar Cambios al `main`:\*\*

- Crear un pull request en GitHub > Revisar y aprobar el pull request > Merge pull request > (Opcional) Eliminar la rama.