

Construcción de software

Trabajando con ramas

Instrucciones: Desarrolle el tutorial consignado en el anexo.

I. Propósito

Desarrolle el tutorial trabajando con ramas sobre control de versiones en Git y GitHub.

ANEXO

TRABAJANDO CON RAMAS SOBRE CONTROL DE VERSIONES

PREPARANDO EL ENTORNO

Hacer fork del repositorio de recetas

1. Inicie sesión utilizando su cuenta GitHub.
2. Usando el navegador web vaya a la página del proyecto base: https://github.com/jdgamarram/recetas_hispanoamericanas.git;
3. Clic en el botón Fork ubicado en la parte superior derecha de la pantalla.
4. Desactive la casilla de verificación "Copy the main branch only"
5. Clic "Create fork"

Note que GitHub ha creado una copia del repositorio base en su dominio y que el nuevo repositorio tiene el nombre de su usuario el nombre del repositorio base.

6. Haga clic en el botón Clone or download para copiar la URL del nuevo repositorio.
7. Clone el nuevo repositorio en su computador.
8. Verifique las rutas del repositorio local.

```
Git branch
```

9. Verifique las rutas del repositorio local y remoto.

```
Git branch -a
```

CREAR RAMA

Antes de empezar tenga en cuenta que el repositorio base tiene configurada una rama por cada país que se incluye en el recetario.

1. Cree una rama llamada Brasil.

```
git branch Brasil
```

2. Consulte las ramas del repositorio para verificar la operación anterior.

```
git branch -a
```

Note que el repositorio remoto contiene 3 ramas diferentes, Colombia, México y main (master). Mientras que el repositorio local sólo tiene main (master) y la rama que se acaba de crear (Brasil).

3. Cambie de rama para agregar una nueva receta.

```
git checkout Brasil
```

AGREGAR RECETA

Para realizar esta parte del tutorial tenga en cuenta los archivos en la carpeta: /files_to_add. Siga los siguientes pasos para agregar una receta de Brasil:

1. Copie y pegue la carpeta /files_to_add/Brasil en el directorio /recetas de su copia de trabajo local.
2. Copie y pegue la imagen /files_to_add/imagenes/arrolladito.jpg en la carpeta /imagenes de su copia de trabajo local.
3. Incluya el nuevo país en el archivo index.html.
4. Verifique el estado del repositorio.

```
git status
```

Evidencie que los nuevos archivos han sido identificados en la copia de trabajo local, pero aún no han sido incluidos en el área de preparación.

5. Agregue los cambios al área de preparación.

```
git add .
```

6. Confirme los cambios en el repositorio local.

```
git commit -m "Se incluye receta de Brasil"
```

7. Sincronice los cambios con el repositorio remoto.

```
git push --set-upstream origin Brasil
```

La opción --set-upstream se encarga de publicar la rama en el repositorio remoto.

8. Consulte nuevamente las ramas del repositorio para verificar la operación anterior.

```
git branch -a
```

CREAR Y EXTRAER ETIQUETAS

Git permite etiquetar puntos específicos del historial como importantes. Esta funcionalidad facilita la búsqueda de los commit marcados a lo largo del tiempo. Existen dos tipos de etiquetas: ligeras y anotadas. Una etiqueta ligera es muy parecida a una rama que no cambia, simplemente es un puntero a un commit específico, mientras que una etiqueta anotada es algo más complejo, pues contiene el nombre del etiquetador, correo electrónico, fecha y un mensaje asociado. En este tutorial trabajará con etiquetas anotadas, las listará, consultará y eliminará.

1. Cree una etiqueta anotada para el último commit.

```
git tag -a v1.0 -m "Receta brasileña: Arrolladitos"
```

2. Liste las etiquetas del repositorio.

```
git tag
```

3. Consulte la etiqueta creada en el primer punto.

```
git show v1.0
```

Este comando muestra toda la información relacionada con la etiqueta: su autor, la fecha, el commit correspondiente y una previsualización de los archivos involucrados.

4. Elimine la etiqueta v1.0.

```
git tag -d v1.0
```

La opción -d indica que se hará un delete de la etiqueta

COMPARAR RAMAS

Ejecute el siguiente comando para comparar dos ramas:

```
git diff --name-status main..Brasil
```

La opción `--name-status` permite consultar el nombre de los archivos y su estado (Delete, Add or Modified).

El comando anterior permite comparar master con respecto a Brasil. Note que los archivos marcados con la letra A corresponden a los agregados en la sección anterior y el archivo marcado con la letra M corresponde al modificado. Este comando tiene muchas variaciones dependiendo de lo requerido.

En el caso de querer consultar las diferencias incluyendo el contenido de los archivos implicados, utilice el comando:

```
git diff main..Brasil
```

Si desea consultar el nombre y cantidad de archivos modificados, así como la cantidad de líneas insertadas o eliminadas, utilice el comando:

```
git diff --stat main..Brasil
```

La opción `--stat` permite consultar la cantidad de archivos modificados y la cantidad de cambios.

MEZCLAR RAMAS

En esta sección del tutorial fusionará los cambios realizados en la rama Brasil con la rama master. Siga los siguientes pasos:

1. Ubíquese en la rama master.

```
git checkout main
```

2. Fusione la rama Brasil en master.

```
git merge Brasil
```

3. Sincronizar el repositorio remoto

```
git push
```

De esta forma habrá realizado una fusión entre ramas sin conflicto. En la siguiente sección, realizará un ejemplo de fusión con conflictos.

CONFLICTOS DE FUSIÓN

Los conflictos de fusión ocurren al tratar de integrar dos ramas con cambios en el mismo archivo o en la misma parte del archivo.

A continuación, simulará el trabajo de 2 desarrolladores, uno desde la rama main y otro desde la rama Colombia. Ambos provocarán un conflicto al agregar una receta de este país y modificar el index simultáneamente.

Antes de empezar, abra dos terminales o consolas de Git Bash, asegúrese que cada una esté en una ubicación diferente y que tenga el repositorio del proyecto clonado.

Personalice su identidad en cada una de las terminales:

Para el desarrollador 1:

```
git config user.name "desarrollador_1"  
git config user.email desarrollador_1@continental.edu.pe
```

Para el desarrollador 2:

```
git config user.name "desarrollador_2"  
git config user.email desarrollador_2@continental.edu.pe
```

Para el desarrollador 1:

Ahora, siga los siguientes pasos para el desarrollador 1:

1. Ubíquese en la rama Colombia.

```
git checkout Colombia
```

Si aparece un error. Esto porque no se ha creado localmente una referencia a esa rama remota. Lo puede verificar consultando las ramas del repositorio.

```
git branch -a
```

Para hacer seguimiento de forma local a la rama Colombia ejecute el comando:

```
git checkout --track origin/Colombia
```

Consulte nuevamente el listado de ramas para evidenciarlo.

```
git branch -a
```

2. Copie y pegue el archivo /files_to_add/Colombia/bandeja_paisa.html en el directorio /recetas/Colombia de su copia de trabajo local.
3. Copie y pegue la imagen /files_to_add/imagenes/bandeja_paisa.jpg en la carpeta /imagenes de su copia de trabajo local.
4. Incluya la nueva receta en el archivo index.html.
5. Confirme los cambios en el repositorio local.

```
git status
git add .
git status
git commit -m "Se incluye receta colombiana: Bandeja paisa"
```

6. Sincronice los cambios con el repositorio remoto.

```
git push
```

Para el desarrollador 2:

Ahora, siga los siguientes pasos para el desarrollador 2:

1. Ubíquese en la rama master.

```
git checkout main
```

2. Copie y pegue el archivo /files_to_add/Colombia/arroz_coco.html en el directorio /recetas/Colombia de su copia de trabajo local.
3. Copie y pegue la imagen /files_to_add/imagenes/arroz_coco.jpg en la carpeta /imagenes de su copia de trabajo local.
4. Incluya la nueva receta en el archivo index.html.
5. Confirme los cambios en el repositorio local.

```
git status
git add .
git status
git commit -m "Se incluye receta colombiana: Arroz con coco"
```

6. Sincronice los cambios con el repositorio remoto.

```
git push
```

7. Integre los cambios de la rama Colombia con la rama main.

```
git merge Colombia
```

Si Git no reconoce Colombia en su espacio local, mezcle con la rama remota de Colombia.

```
git merge origin/Colombia
```

Note que Git ha detectado un conflicto pues el archivo index.html tiene cambios diferentes en cada una de las ramas.

Para solucionarlo abra el archivo en su IDE preferido. Resaltado en verde aparecen los cambios presentes en la rama main y resaltado en azul los cambios presentes en la rama origin/Colombia, acepte los 2 cambios y haga commit en su repositorio local.

Ahora guarde los cambios.

```
git status
git add .
git status
git commit -m "Se soluciona conflicto"
```

8. Sincronice los cambios con el repositorio remoto.

```
git push
```

Con este último paso, usted ha solucionado el conflicto generado por modificar la misma parte del mismo archivo en diferentes ramas.