**AE-2. Control de versiones con GIT y GitHub**

**Grupo 10:**

Borja Ernesto Del Valle García

Lidia Díaz Mendoza

Sergio Martínez Rivera

Carlos Rábago Torcates

**URL del repositorio :** <https://github.com/Cambixx/WS_Actividad_ED.git>

**ÍNDICE**

1. Resta (realizado por Borja Ernesto Del Valle García)

2. Suma (realizado por Lidia Díaz Mendoza)

3. Producto (realizado por Sergio Martínez Rivera)

4. Cociente (realizado por Carlos Rábago Torcates)

**1. RESTA**

**2. SUMA**

1. Lo primero que he hecho ha sido crearme una cuenta de GitHub para poder clonar el repositorio remoto que ha creado Carlos para el proyecto. Y luego clonarlo con **git clone DIRECCION\_REPOSITORIO\_GITHUB**
2. Después, he creado una rama llamada Suma con el comando **git checkout -b Suma** para poder trabajar sobre mi parte del trabajo.
3. He cambiado los nombres de los métodos para que sean más representativos y he implementado el código en los mismos.
4. También he cambiado un poco la redacción de la documentación.
5. A continuación, he ido haciendo pruebas en el **main Calculadora()** para ver que los métodos funcionaban.
6. Cuando he terminado con las pruebas, he usado **git add .** para pasar los cambios al stage y después hacer un commit con **git commit -m "comentario".**
7. Después he descomentado el atributo acumulado (lo habíamos comentado porque daba un error al no tener el código implementado) y he implementado el código del método. Para comprobar que funcionaba, he creado el método **getValorAcumulado** y así mostrar la suma acumulada.
8. Después de probar que funcionaba he vuelto a hacer stage y luego commit con los cambios.
9. Cuando ya lo tenía todo, he cambiado a master con **git checkout master** y he hecho **git pull** por si mis compañeros habían hecho cambios. A continuación, he fusionado mi rama con la de master con **git merge Suma**.
10. Por último, he subido mis cambios al repositorio remoto con **git push origin master**.
11. Como era necesario un documento sobre la actividad, Carlos ha creado el documento, para que cada uno de nosotros rellene su parte. He hecho un **git pull** para sincronizarlo, lo he rellenado y lo he vuelto a subir con un **git push origin master**.

**3. PRODUCTO**

1. Primero, he clonado el repositorio remoto creado por Carlos.
2. Una vez clonado, he creado una rama para trabajar sobre ella, donde he realizado la corrección de errores y las mejoras oportunas sobre mi clase producto de la actividad anterior. He pasado la documentación de mayúsculas a minúsculas y he cambiado la clase para que devuelva tan solo el resultado, en lugar de Strings.
3. He creado una clase Main llamada Calculadora() sobre la que hemos ido trabajando e implementando el código todos los integrantes del grupo.
4. He implementado el código de la clase producto.
5. Una vez realizados estos cambios, he realizado el git add .
6. Luego, un git commit - m "comentario ".
7. Me he movido a la rama master y la he fusionado con la rama con la que he estado trabajando con git merge NOMBRE\_RAMA.
8. Por último, he realizado el git push origin master para subir los cambios al repositorio remoto de GitHub.

\***No** he realizado todo lo anteriormente descrito directamente en una rama. Una vez clonado he creado la rama donde he implementado el código de la clase producto y he solventado los errores descritos de la actividad anterior. Una vez fusionada esa rama y subidos los cambios al GitHub he mirado que se hayan realizado correctamente y lo he informado al grupo para que cada uno pudiéramos ir mirando si se generaban correctamente los cambios.

La clase Calculadora() la he creado tras hacer un git pull para actualizar mi repositorio remoto con los cambios realizados por los compañeros, y crearme otra rama, sobre la que he trabajado. De nuevo hice commit y la fusioné con la master para después hacer el git push.

Hice lo mismo para escribir esta parte del documento, así como la portada e índice.

1. **COCIENTE**
2. Lo primero ha sido crear un repositorio en Github y enlazarlo con mi repositorio local (git remote add origin Url\_repositorio) para que todo el equipo tenga acceso al mismo y puedan hacer los cambios necesarios en el proyecto
3. Lo siguiente ha sido hacer un git pull para actualizar el repositorio local con los cambios que pudiese tener el repositorio remoto.
4. He creado una nueva rama (git checkout -b desarrollo\_cociente) para desarrollar el código correspondiente a los métodos de la clase cociente.
5. He incluido el código para el método divisionReales, seguidamente git add . para agregar los cambios al stage y he hecho un primer commit (git commit -m “mensaje”). Luego he fusionado la rama al master para comprobar que todo estaba funcionando, para ello me he posicionado en la rama master (git checkout master) y desde ahí he hecho la fusión de la rama desarrollo\_cociente (git merge desarrollo\_cociente).
6. Seguidamente he vuelto a la rama desarrollo\_cociente y he ido agregando el código para cada método y haciendo un commit por cada uno, luego he vuelto a fusionar la rama a el master igual que en el paso anterior.
7. Luego he creado una nueva rama llamada desarrollo\_main en la cual he agregado las pruebas en el main para determinar que los métodos de la clase funcionan correctamente, de igual manera la he fusionado posteriormente a la rama master.
8. También he hecho correcciones de algunos detalles directamente en el master.
9. Añadí un Documento de word donde todos pudiésemos añadir los pasos que hemos seguido para desarrollar nuestra parte de la actividad.

Hemos decidido desarrollar el main cada uno sin borrar el código ni sobrescribiendo el de otro compañero para evitar problemas de sincronización.