



**UNIVERSIDAD DISTRITAL  
FRANCISCO JOSÉ DE CALDAS**

**Facultad de Ingeniería.**

**Asignatura: Programación Avanzada.**

**Docente: Jhon Herrera Cubides.**

**Documento Versiones GIT**

**Estudiante: Quintero Ramírez Ana María**

**Código: 20231020211**

**Estudiante: Barrera Pulido Samuel Andrés**

**Código: 20232020156**

**Estudiante: Avila Alvarez Juan David**

**Código: 20232020154**

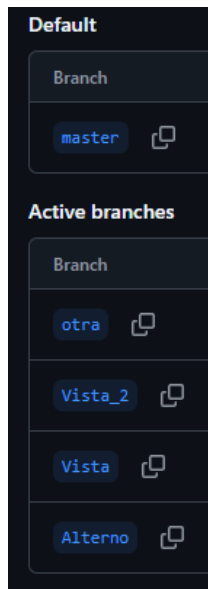
**Bogotá, D.C.**

**22 de Noviembre del 2024.**

## Versiones GIT

Este documento explica cómo se ha gestionado el manejo de versiones y el uso de ramas en nuestro proyecto. Utilizamos Git para asegurar un desarrollo organizado y GitHub para el control de versiones

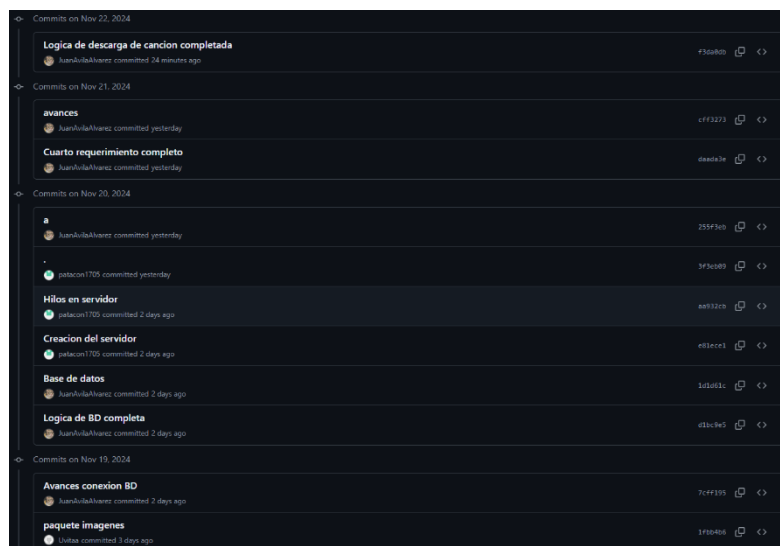
### USO DE RAMAS



Se ha hecho uso de ramas según la necesidad y el avance en el proyecto

Master: donde se encuentra el código principal.  
Vista: donde se realizaron las diferentes pruebas para la interfaz gráfica.  
Vista\_2: Donde se agregaron algunos cambios de vista para no generar conflictos.  
Alterno: Para realizar pequeñas pruebas de versionamiento.  
Otra: Se usó en un punto avanzado del proyecto para hacer ultimas actualizaciones.

### Evidencia Rama Master:



## USO COMANDOS GIT

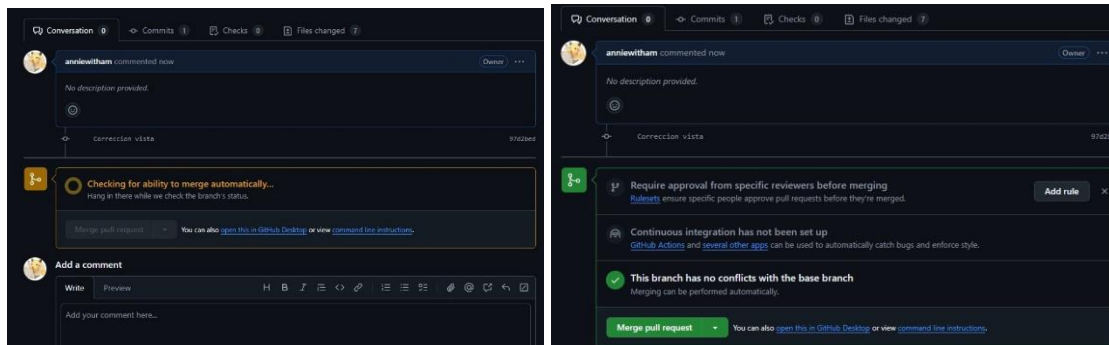
Se realizan cambios en sus repositorios locales y realizan commits con mensajes descriptivos para cada cambio con el fin de comprender qué fue lo que se implementó y/o corrigió en el código. Esto se puede observar en la imagen proporcionada donde se realiza un commit con el mensaje

Se hizo uso de los comandos, para la gestión de versiones y ramas por medio de git (cmd)

- add. : Subir el archivo
- commit: Registra los cambios en el repositorio
- status: Muestra el estado de los archivos en la rama que esté.
- branch: Muestra las ramas del repositorio
- switch: Permite cambiar entre ramas existentes o crear una nueva rama y así poder actualizar los cambios

## Pull Request.

Se usó después de enviar los cambios, se crea un **pull request**. Esto solicita una revisión de código antes de integrar los cambios a la rama `master`



## Historial de commits

Para ver el historial de commits se puede hacer uso del comando `git show-branch` o inclusive con el comando `git log --oneline` para ver un resumen de las ramas y sus respectivos commits.

```
avila@Juanito MINGW64 ~/Documents/Sistemas/Programación Avanzada/Parcial2PA (master)
$ git show-branch
! [Vista] Parte documentacion
* [master] Logica de descarga de cancion completada
--
* [master] Logica de descarga de cancion completada
* [master^] avances
* [master~2] Cuarto requerimiento completo
* [master~3] a
* [master~4] .
* [master~5] Hilos en servidor
* [master~6] Creacion del servidor
+ [Vista] Parte documentacion
+ [Vista^] vista servidor
+ [Vista~2] vista servidor
+* [master~7] Base de datos
```

Y también `git log` para observar el historial completo

```
commit f3da0dbcb944b0372011a7c1506b9d738133fec6 (HEAD -> master, origin/master, origin/HEAD)
Author: JuanAvilaAlvarez <avila.avarez@gmail.com>
Date:   Fri Nov 22 05:46:03 2024 -0500

    Logica de descarga de cancion completada

commit cff327387de5aeaa3feee8e97c952cfd105b5be6
Author: JuanAvilaAlvarez <avila.avarez@gmail.com>
Date:   Thu Nov 21 03:47:16 2024 -0500

    avances

commit daada3ed5823ee9a3a8ff965ef937195fea5b481
Author: JuanAvilaAlvarez <avila.avarez@gmail.com>
Date:   Thu Nov 21 02:06:10 2024 -0500

    Cuarto requerimiento completo

commit 255f3eb5601da5dc798e50787bd3fdcaf42a662c
Author: JuanAvilaAlvarez <avila.avarez@gmail.com>
Date:   Wed Nov 20 23:01:07 2024 -0500

    a

commit 3f3eb09c66871fba57590457f46109be6c6eb509
Author: Patacon <samuelbarrerapulido@gmail.com>
Date:   Wed Nov 20 21:08:38 2024 -0500
```