

- CRISTHOFER PATZÁN 19218

Laboratorio GIT

Parte 0:

Check point

pat19218@uvg.edu.gt

git clone <https://CrisPat@bitbucket.org/parte1/uvgie303020221.git>

Parte 1:

- Se clonara el repositorio llamado “UVGIE303020221” y se almacenará en la carpeta en la que se ubique el CMD en ese momento.
- En mi caso lo voy a almacenar en “User”
- Al loguearse a BitBucked no se encuentran cambios en el repositorio.

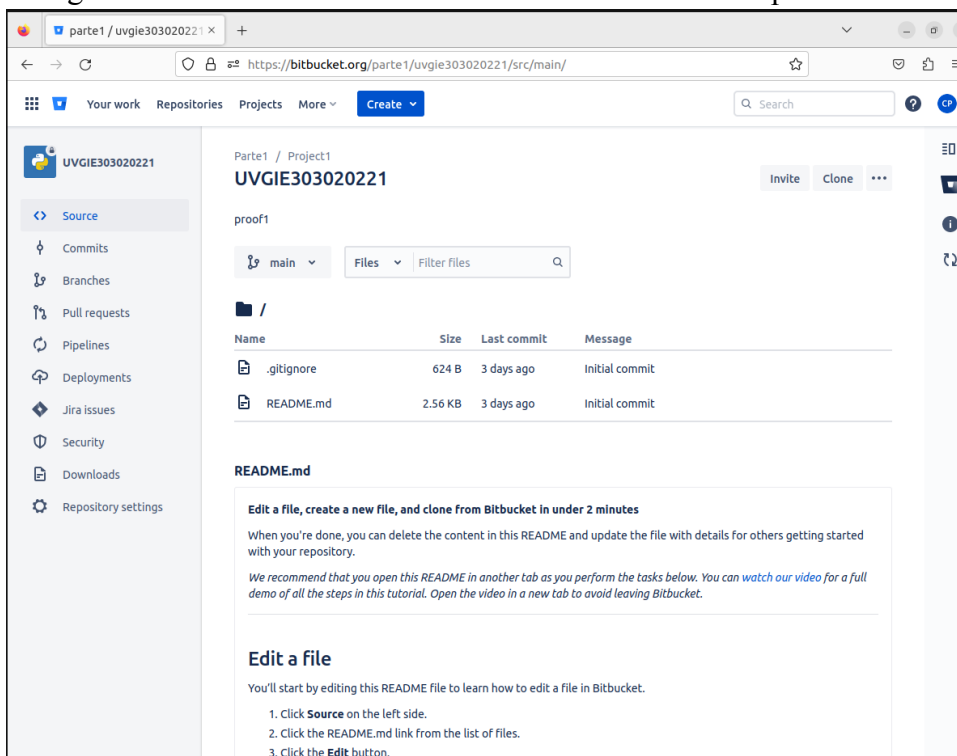


Imagen 1. Creación del repositorio

- Sigue igual.

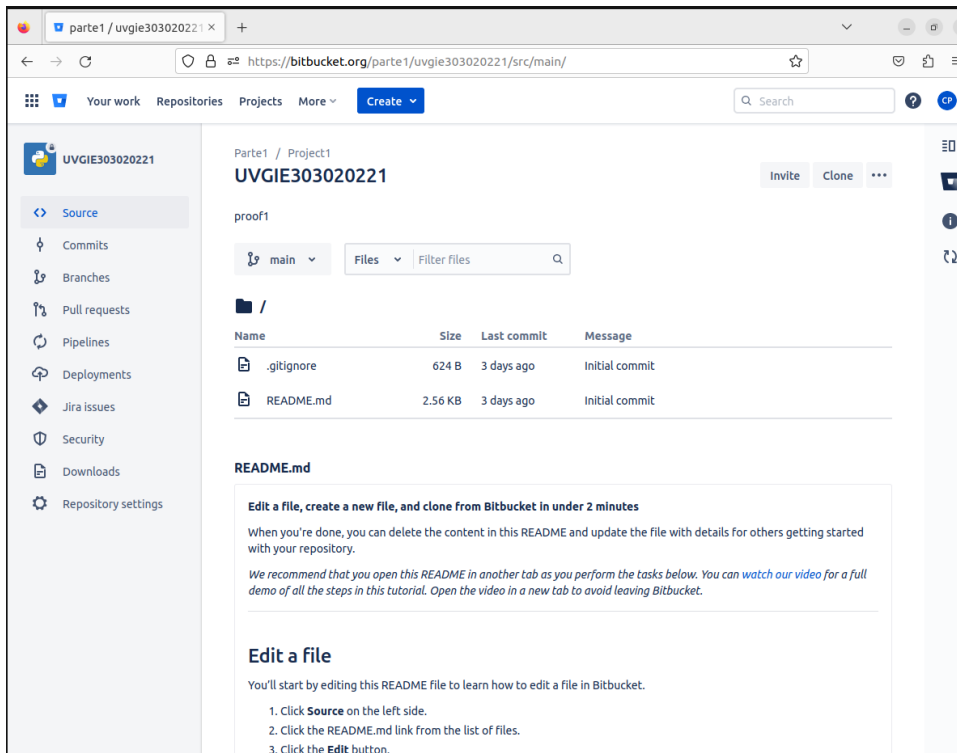


Imagen 2. Modificación del branch en Linux

- Al realizar el push se han cargado todos los documentos realizados.

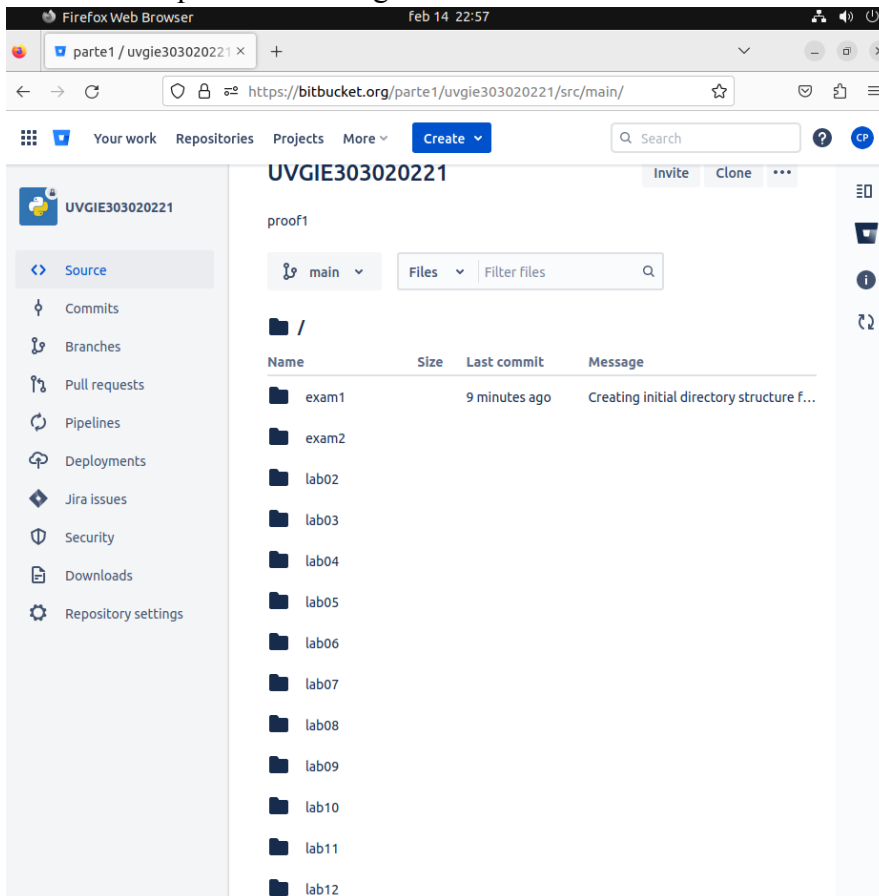


Imagen 3. Repositorio tras realizar el pull

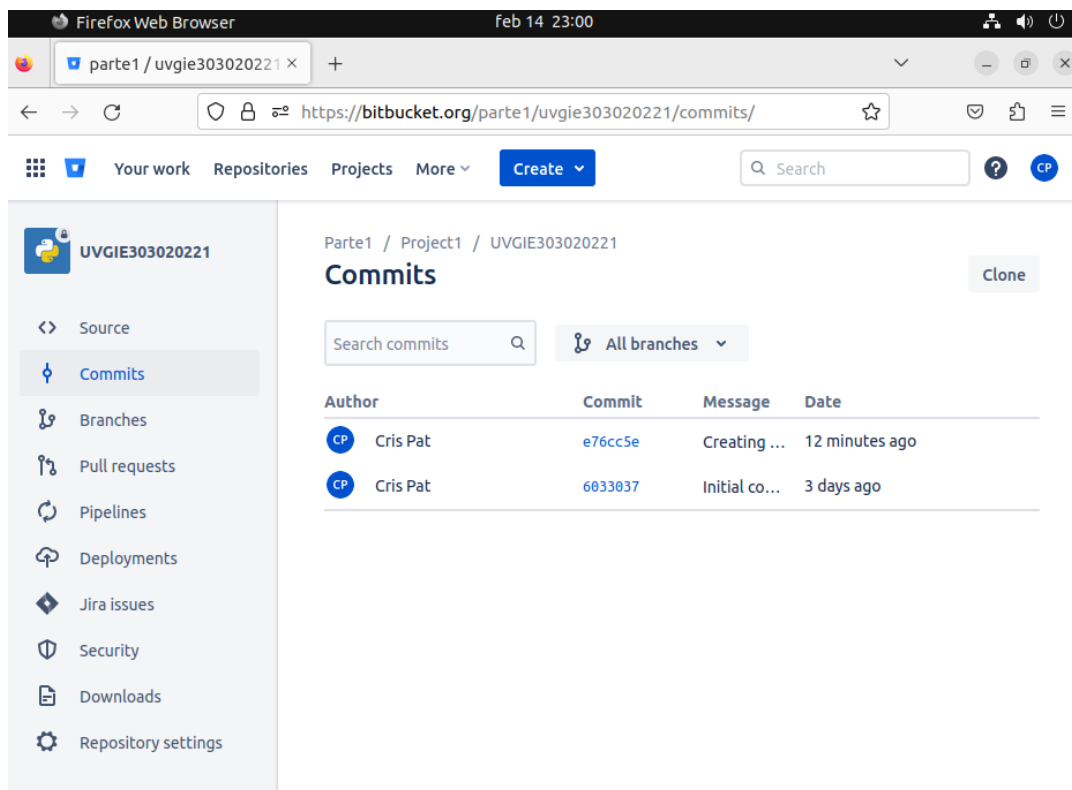


Imagen 4. Historial del repositorio tras el primer pull

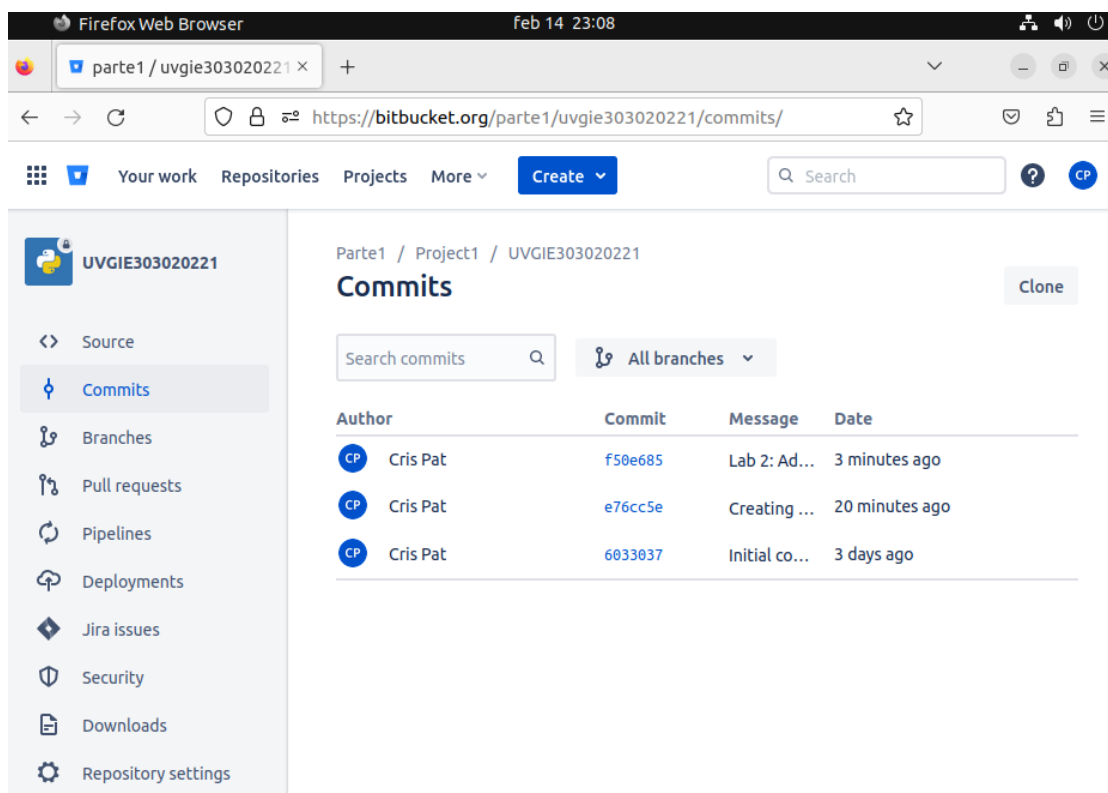


Imagen 5. Historial del repositorio tras el segundo pull

- **What does the line starting in --- represent?**
Normalmente representa una línea "antes" en una salida diff.
- **What does the line starting in +++ represent?**
Normalmente representa una línea "después" en una salida diff

- **What do the lines starting in + represent?**

Típicamente representa una adición en una salida diff

- **Can you guess what a line starting in - would represent?**

Eliminar una salida diff

- **What benefits would a large team of developers get from version control? Identify at least two.**

Colaboración:

Varios desarrolladores pueden trabajar en el mismo código base simultáneamente sin preocuparse por los conflictos de versión. Los cambios se pueden rastrear y fusionar fácilmente utilizando las herramientas proporcionadas por el sistema de control de versiones.

Responsabilidad:

El control de versiones puede rastrear quién realiza qué cambios en el código base y cuándo, lo que facilita la asignación de responsabilidades y la resolución de problemas.

- **What benefits would a single developer (working alone) get from version control? Identify at least two.**

Seguimiento del historial:

Los sistemas de control de versiones proporcionan un historial detallado de los cambios realizados en el código base a lo largo del tiempo. Esto permite a un desarrollador ver cómo ha evolucionado el código, volver a versiones anteriores si es necesario y señalar cuándo se introdujeron errores o fallas.

Copia de seguridad y recuperación:

Al mantener el código base en un sistema de control de versiones, un desarrollador tiene una copia de seguridad confiable de su trabajo que se puede recuperar fácilmente en caso de falla del hardware, eliminación accidental u otros problemas. Esto puede ser particularmente importante para proyectos a largo plazo o cuando se trabaja en código que es fundamental para el trabajo o el negocio del desarrollador.

- **What kind of files should you put in version control?**

En general, cualquier tipo de archivo que sea importante para el proceso de desarrollo se debe poner en control de versiones. Esto incluye archivos de código fuente, documentación, archivos de configuración y más. Los sistemas de control de versiones no se limitan a ningún tipo específico de archivo, por lo que cualquier archivo que forme parte del proceso de desarrollo se puede almacenar y rastrear mediante el control de versiones.

- **What kind of files should you not put in version control?**

Los archivos binarios que son grandes y/o cambian con frecuencia, como los archivos multimedia (imágenes, videos, audio), generalmente no se deben poner en el control de versiones. Esto se debe a que pueden aumentar rápidamente el tamaño del repositorio, lo que dificulta el trabajo y ralentiza el sistema de control de versiones. Además, los archivos binarios no se fusionan fácilmente, por lo que pueden surgir conflictos cuando varios desarrolladores trabajan en el mismo archivo.

- **What is the difference between an add, commit and a push?**

En Git, add agrega cambios al área de preparación, commit crea una nueva versión del repositorio con los cambios en el área de preparación y push envía los cambios a un repositorio remoto. Específicamente, add prepara los cambios para la confirmación, la confirmación crea una instantánea de los cambios y envía los cambios a un servidor remoto

- **Why is it better to use git mv instead of plain-old regular mv to copy a file within the repository?**

Es mejor usar git mv en lugar del comando mv regular para copiar un archivo dentro del repositorio porque git mv le dice a Git que organice el movimiento como una sola operación, en lugar de eliminar el archivo original y agregar un nuevo archivo. . Esto ayuda a mantener el historial de archivos y evitar cualquier confusión o conflicto que pueda surgir de un comando mv normal.

- **How would you suggest improving this lab in future semesters?**

Las sugerencias para mejorar este laboratorio en futuros semestres podrían incluir agregar más ejercicios prácticos y ejemplos para reforzar los conceptos cubiertos, brindando más oportunidades para que los estudiantes trabajen en colaboración usando el control de versiones e introduciendo temas más avanzados, como bifurcación y fusión. Además, proporcionar acceso a un repositorio Git remoto para que los estudiantes practiquen ayudaría a prepararlos para los flujos de trabajo de desarrollo del mundo real