

HOMEWORK 1.

USAC.

Facultad de ingeniería.

Maestría Ingeniería para la industria con especialidad en Ciencias de la Comunicación.

Alumno: German Contreras Jacinto

Carné: 999011685

Glosario:

VSC: VISUAL STUDIO CODE.

Que es GIT: es un software donde se pueden crear proyectos informáticos y se puede hacer seguimiento de las ediciones, versiones, también permite trabajar de forma individual o bien en grupo de programadores para un desarrollo. Esto facilita el manejo de cambios en los programas ya que se puede retroceder o ver en qué punto un programa tiene fallas, errores y tener un registro de todas las ediciones y los usuarios que acceden al programa. Todo este registro con la fecha, persona, versión que se está trabajando.

Control de versiones con GIT: Este software permite tener el registro del control de cambios de versiones de una carpeta que contenga archivos de diferentes tipos. Para ello se utilizan comandos de consola para hacer esta gestión de cambios, se puede utilizar por ejemplo el editor de código VISUAL STUDIO CODE, se pueden utilizar otros. Para ello se abre una carpeta en VSC y de allí se inicia un repositorio con el comando GIT INIT. De allí se agrega un archivo o archivos, o bien una carpeta mediante el comando ADD. Ya que se realizan los cambios se realiza el comando GIT COMMIT con esto se agrega archivos al repositorio.

Estados de un archivo en GIT: Los estados de un archivo se pueden visualizar al ejecutar el comando GIT STATUS -S donde va cambiando de color de acuerdo a los cambios que realizan en el archivo.

Como se configura un repositorio: Para ello se utiliza el comando GIT INIT para iniciar un repositorio, este comando se ejecuta solo una vez por carpeta. Esto se realiza en la computadora local donde se encuentran los archivos.

Para tener un respaldo en la nube, se utiliza GIT-HUB donde se guarda la carpeta y los archivos y se va visualizando los cambios que se van realizando. Para esto hay que crear una cuenta en esta plataforma y allí se crea un repositorio que guardara la carpeta con los archivos que se están gestionando los cambios. Se debe realizar un enlace entre la computadora local y GITHUB mediante comandos en VSC.

Comandos en GIT: Dentro de los comandos básicos que se utilizan podemos mencionar los siguientes:

- GIT VERSION: este comando indica la versión de GIT instalada en la computadora.

- GIT INIT: este comando se ejecuta para iniciar un repositorio, este se ejecuta una vez por carpeta.
- GIT STATUS –S: este comando indica el estado de los archivos de un repositorio. Indica si se han realizado cambios en los archivos de una carpeta.
- GIT ADD: agrega un archivo o grupo de archivos a el área temporal, antes de ser enviados al repositorio.
- GIT COMMIT: envía los archivos al repositorio. Se le puede agregar un texto para ir documentando los cambios realizados en el archivo.
- GIT LOG –ONELINE: este comando nos sirve para ver los COMMIT que se han realizado, aquí aparece el número de identificador de cada COMMIT.
- GIT RESET –HARD: Este comando se utiliza para eliminar o sea volver agregar un cambio que se haya realizado en un archivo.
- GIT REMOTE ADD ORIGIN <https://github.com/gcontrerasx/Homeworks-R.git>: este comando se genera en GITHUB para poder conectarse al repositorio creado en la computadora local y el repositorio: HOMEWORKS-R.git
- GIT BRANCH –M MAIN: Este comando se utiliza para poder crear, rotular y eliminar ramificaciones y con ello también poder modificar el nombre.
- GIT PUSH –U ORIGIN MAIN: Con este comando se utiliza para cargar un repositorio entre la computadora donde se ha creado la carpeta con el repositorio y ser subido a la plataforma de GITHUB en esta hace una verificación del usuario para la conexión.

Cabe mencionar que hay una gran cantidad de comandos que se pueden utilizar, estos son los básicos para el control de cambios en versiones de una carpeta de archivos, y en el que hay solo un programador trabajando.

ANEXOS:

Figura 1. Apertura de carpeta en VSC para iniciar creación de repositorio en GIT.

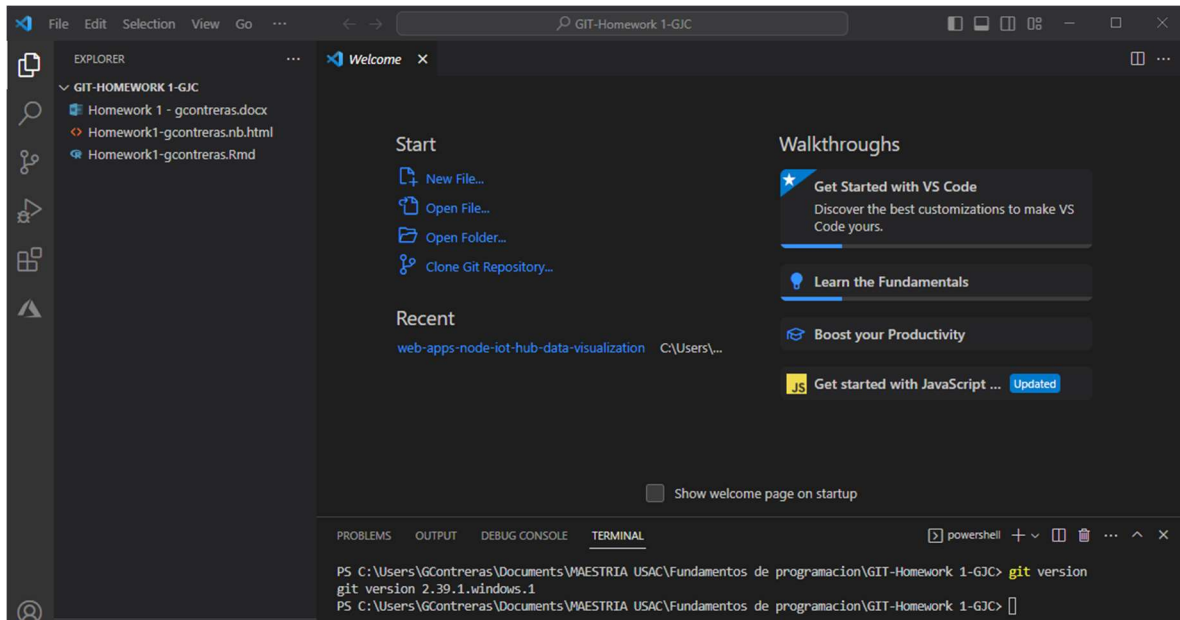


Figura 2. Acceso GIT HUB con cuenta gcontrerasx para creación de repositorio.

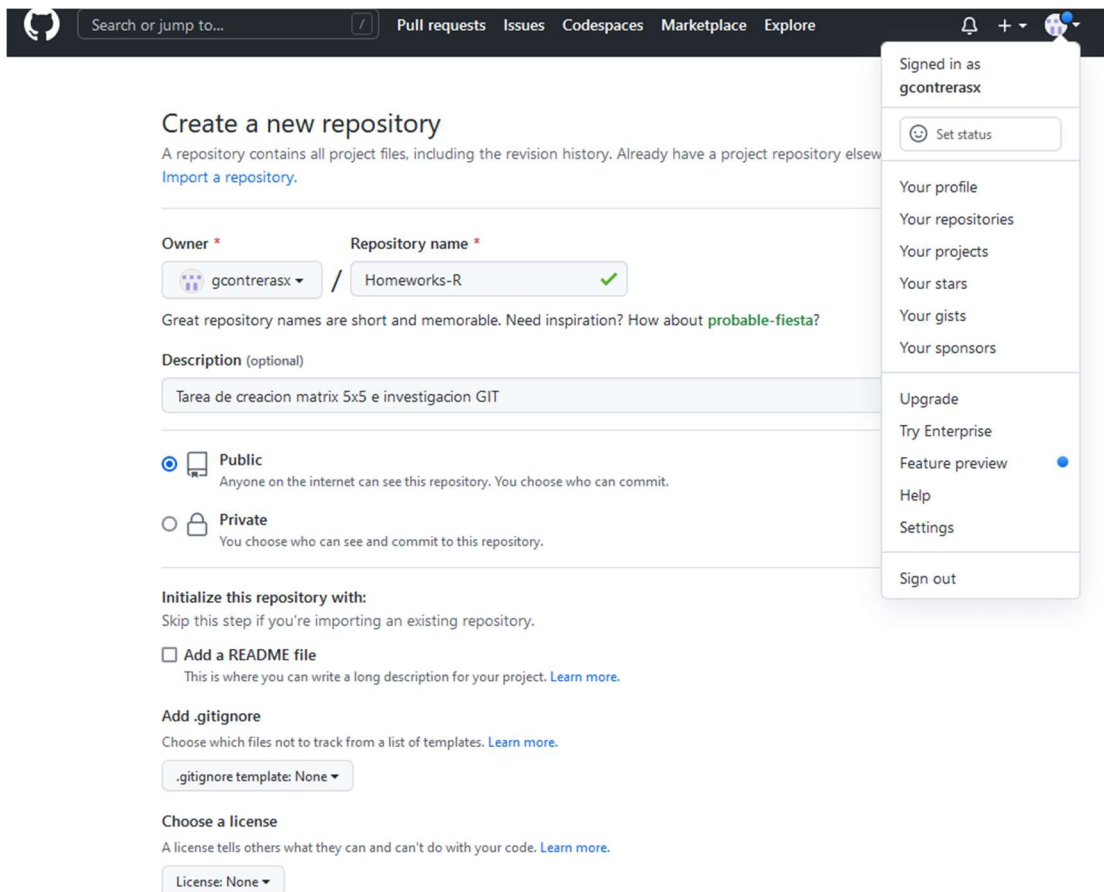
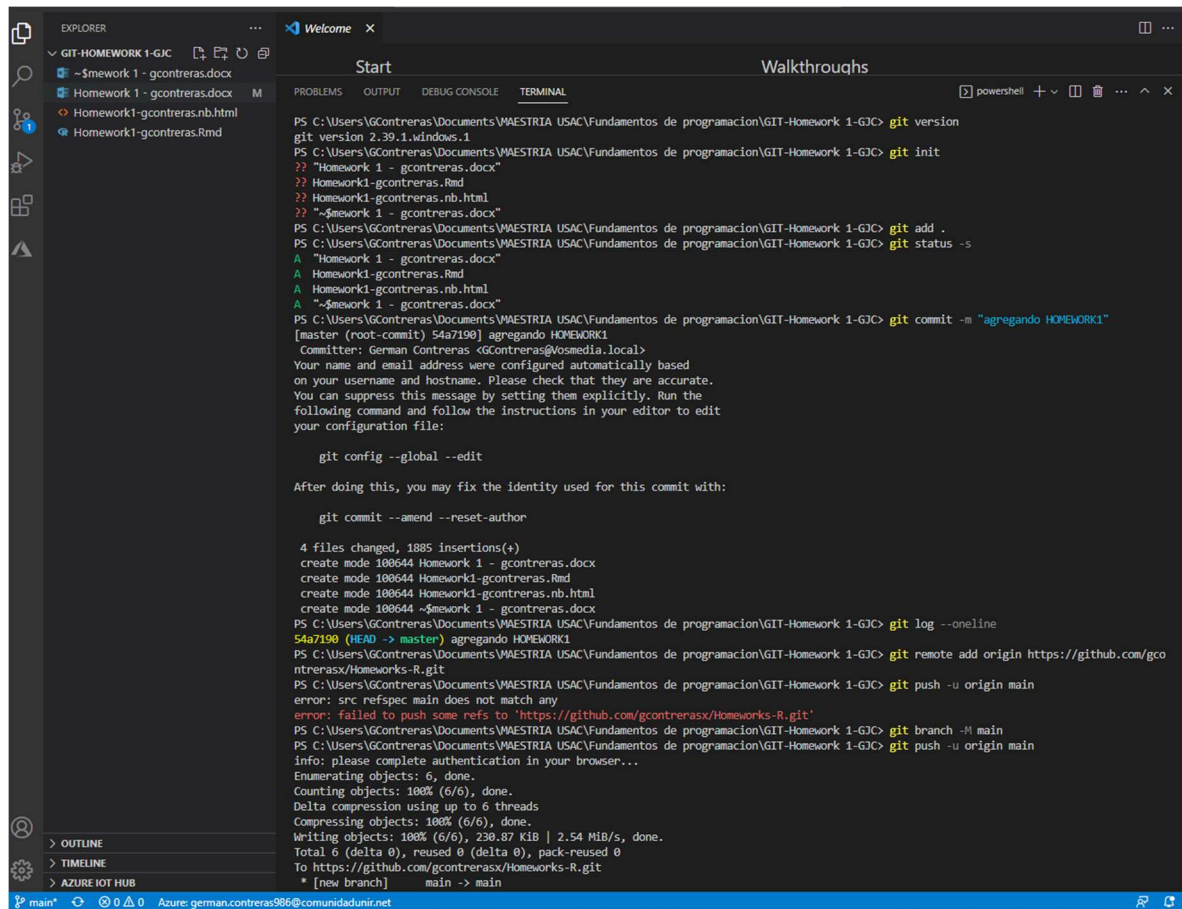


Figura 3. Comandos de consola GIT para iniciar, crear repositorio y acceso a GITHUB.



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal displays the following commands and their output:

```
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git version
git version 2.39.1.windows.1
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git init
?? Homework 1 - gcontreras.docx
?? Homework1-gcontreras.Rmd
?? Homework1-gcontreras.nb.html
?? "~$mework 1 - gcontreras.docx"
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git add .
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git status -s
A Homework 1 - gcontreras.docx
A Homework1-gcontreras.Rmd
A Homework1-gcontreras.nb.html
A "~$mework 1 - gcontreras.docx"
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git commit -m "agregando HOMEWORK1"
[master (root-commit) 54a7190] agregando HOMEWORK1
Committer: German Contreras <GContreras@osmedia.local>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

4 files changed, 1885 insertions(+)
create mode 100644 Homework 1 - gcontreras.docx
create mode 100644 Homework1-gcontreras.Rmd
create mode 100644 Homework1-gcontreras.nb.html
create mode 100644 "~$mework 1 - gcontreras.docx"
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git log --oneline
54a7190 (HEAD -> master) agregando HOMEWORK1
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git remote add origin https://github.com/gco
ntrerasx/Homeworks-R.git
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git push -u origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/gcontrerasx/Homeworks-R.git'
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git branch -M main
PS C:\Users\GContreras\Documents\VAESTRIA USAC\Fundamentos de programacion\GIT-Homework 1-GJC> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 6 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 230.87 KiB | 2.54 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/gcontrerasx/Homeworks-R.git
 * [new branch] main -> main
```

The bottom status bar shows the current branch as 'main' and the user as 'Azure: german.contreras986@comunidadunir.net'.

BIBLIOGRAFIA:

- ¿Cómo usar GitHub? - ¡Todo lo que necesitas saber!.
<https://www.youtube.com/watch?v=44ziZ12rJwU>
- GIT / GITHUB [Tutorial en Español - Parte 1] ♥ Inicio Rápido para Principiantes ♥.
<https://www.youtube.com/watch?v=hWgIK8nWh60>