



Universidad Nacional Autónoma de México
Facultad de Contaduría y Administración
Sistema Universidad Abierta y Educación a Distancia

Alumno: López Acevedo Víctor Rafael

Grupo: 9696

Materia: INFORMATICA VI

Unidad: 2

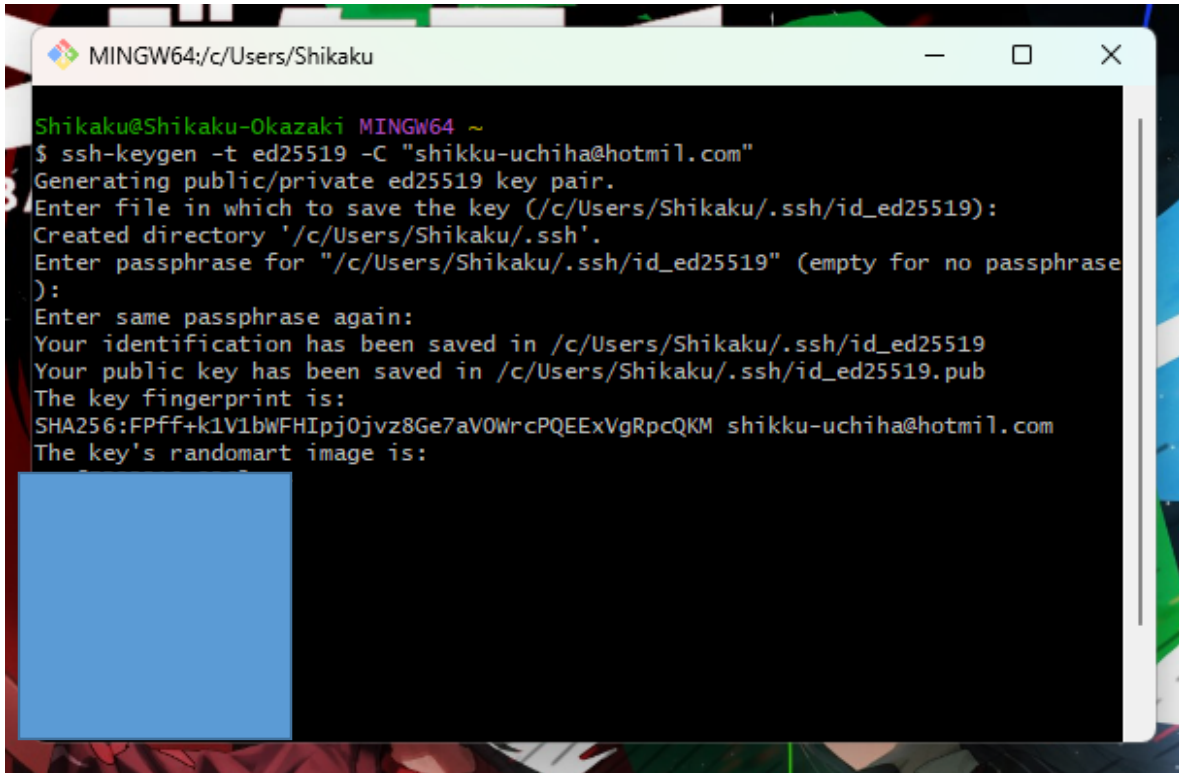
Actividad: M2-01

Fecha: 1 de abril 2025

Actividad: Mi segundo repositorio en Git/Github

Objetivo del aprendizaje: Conocer las ventajas del sistema de control de versiones con protocolo de seguridad Secure Shell Protocol (SSH)

1. Generar un par de llaves SSH



```
MINGW64:/c/Users/Shikaku
Shikaku@Shikaku-Okazaki MINGW64 ~
$ ssh-keygen -t ed25519 -C "shikku-uchiha@hotmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/Shikaku/.ssh/id_ed25519):
Created directory '/c/Users/Shikaku/.ssh'.
Enter passphrase for "/c/Users/Shikaku/.ssh/id_ed25519" (empty for no passphrase
):
Enter same passphrase again:
Your identification has been saved in /c/Users/Shikaku/.ssh/id_ed25519
Your public key has been saved in /c/Users/Shikaku/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:FPff+k1V1bWFHIpj0jvz8Ge7aV0WrcPQEEVgRpcQKM shikku-uchiha@hotmail.com
The key's randomart image is:
```

Para esto usamos el programa de git bash, ingresamos la siguiente línea de comando: `ssh-keygen -t ed25519 -C your_email@example.com` (el e-mail usamos el correo al cual vamos a vincular la llave)

Después nos dirá en qué dirección del disco se creará la carpeta donde estará nuestra llave y le damos enter

Inmediatamente nos pedirá escribir una contraseña, tienes que tener en cuenta que no se mostrara en pantalla, pero la escribirá y después te pedirá que la vuelvas a escribir para confirmar, por eso es importante que recuerdes como la escribiste

```
MINGW64:/c/Users/Shikaku/.ssh

Shikaku@Shikaku-Okazaki MINGW64 ~
$ cd .ssh

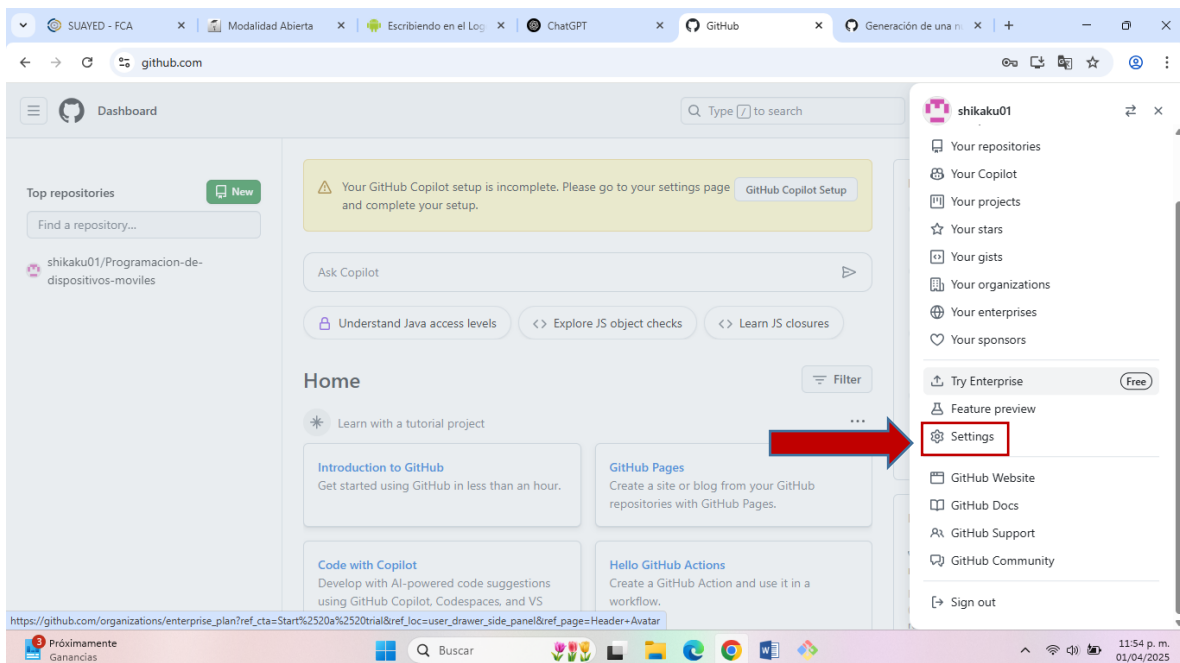
Shikaku@Shikaku-Okazaki MINGW64 ~/.ssh
$ dir
id_ed25519  id_ed25519.pub

Shikaku@Shikaku-Okazaki MINGW64 ~/.ssh
$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAINhaBPwLnJ52YKzyz18nE7pm3mveNVwMs5dD3Z82Er81
shikku-uchiha@hotmail.com

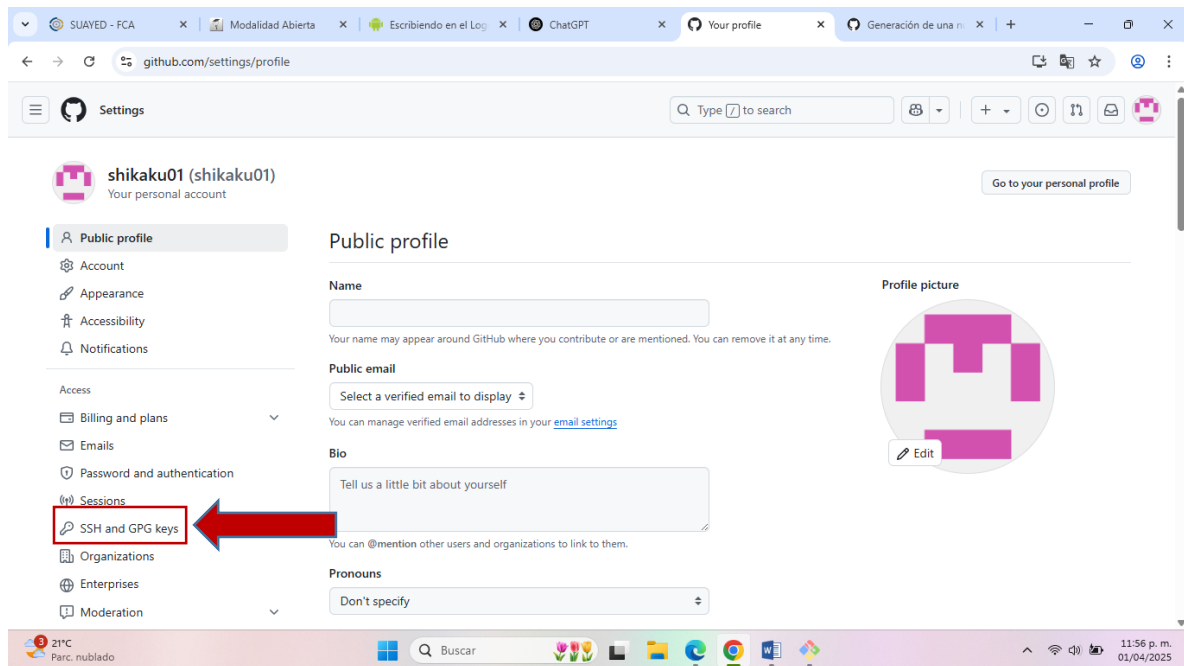
Shikaku@Shikaku-Okazaki MINGW64 ~/.ssh
$
```

Aquí podemos revisar cual es la llave publica que es la que pondremos en el github

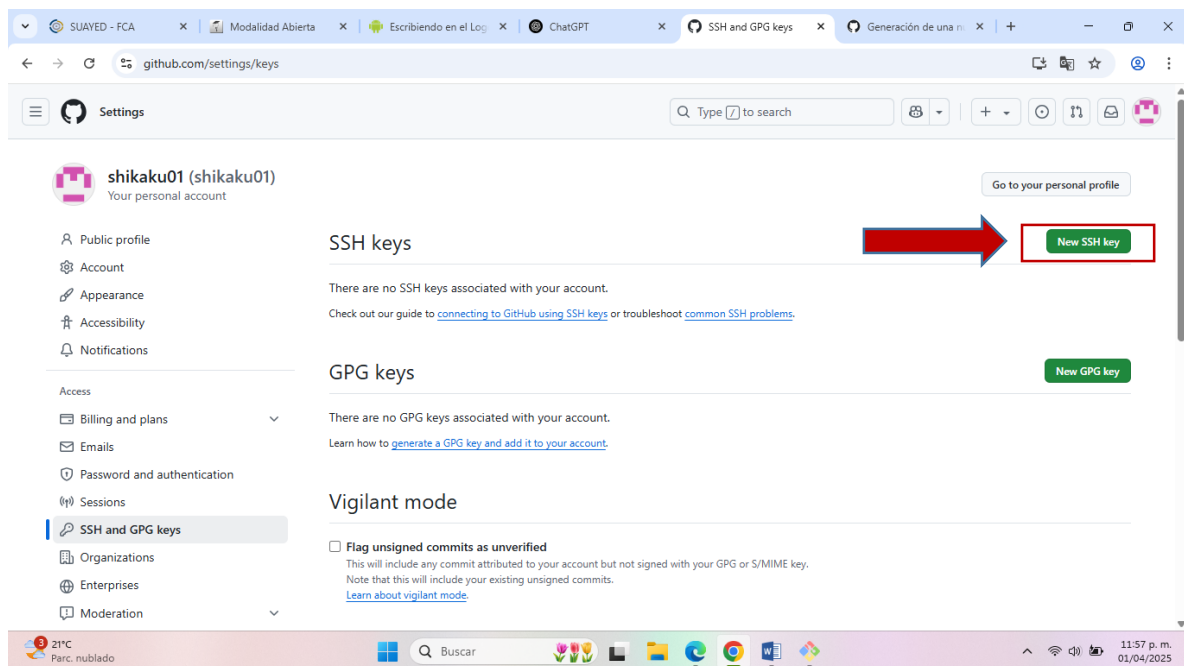
2. Subir llave pública de SSH a su cuenta de Github



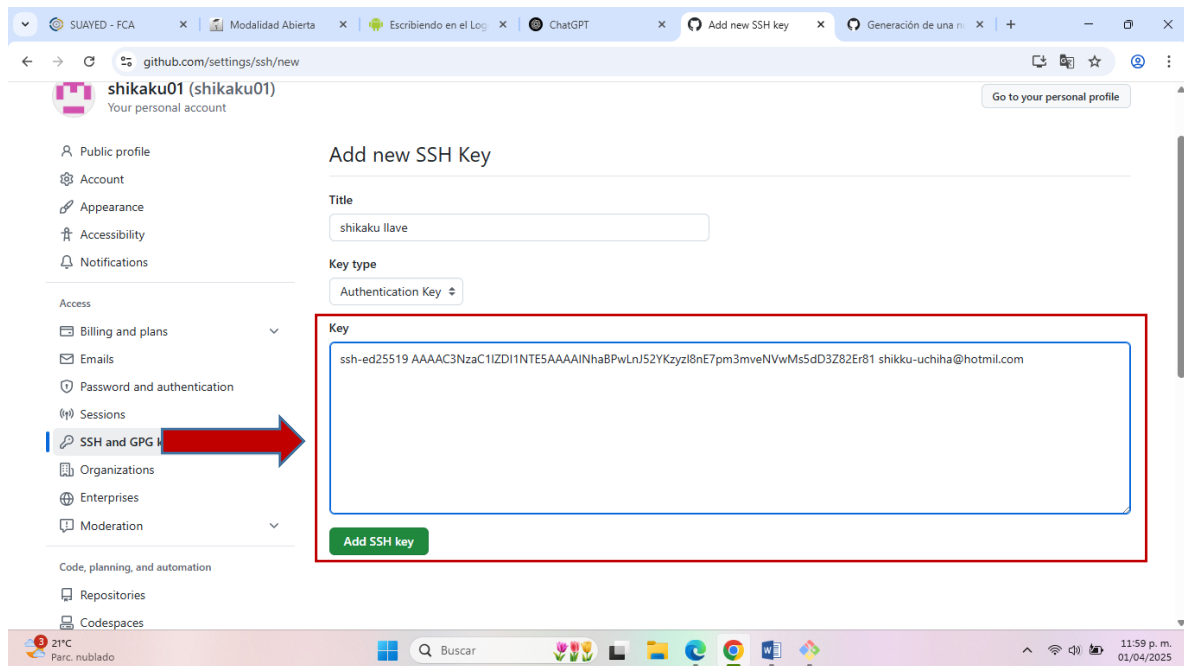
Para subir la llave entramos a nuestra cuenta en github y nos vamos a la opción de settings



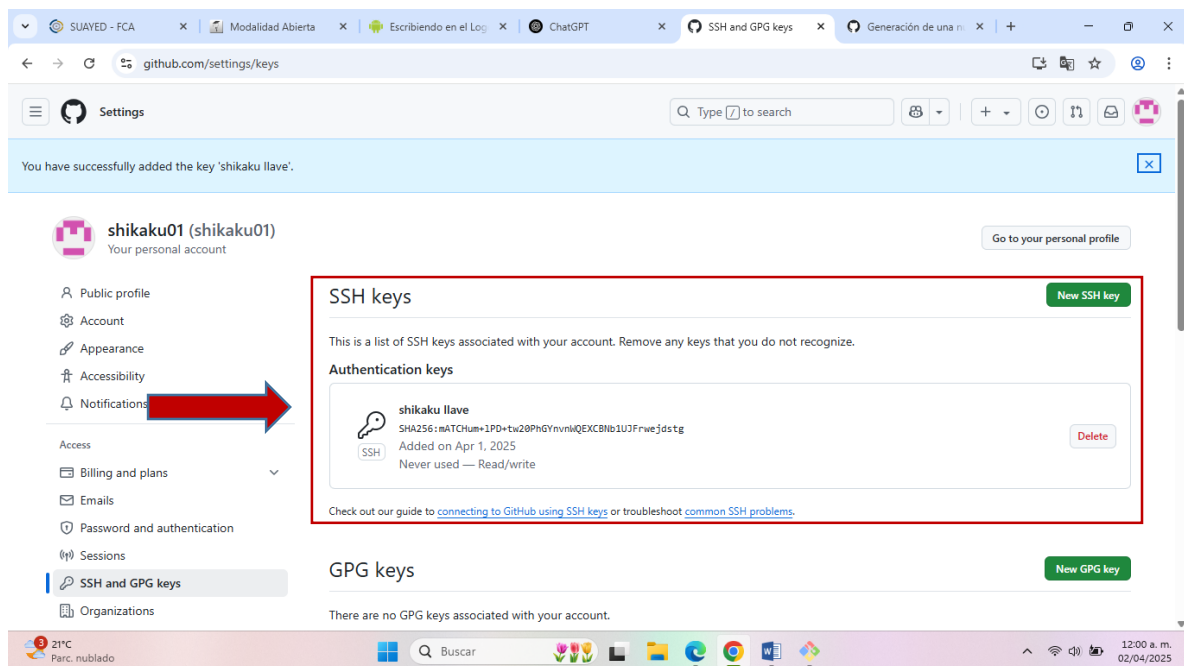
Una vez en settings nos vamos a la sección donde dice SSH and GPG keys



una vez aquí, le damos click a new ssh key

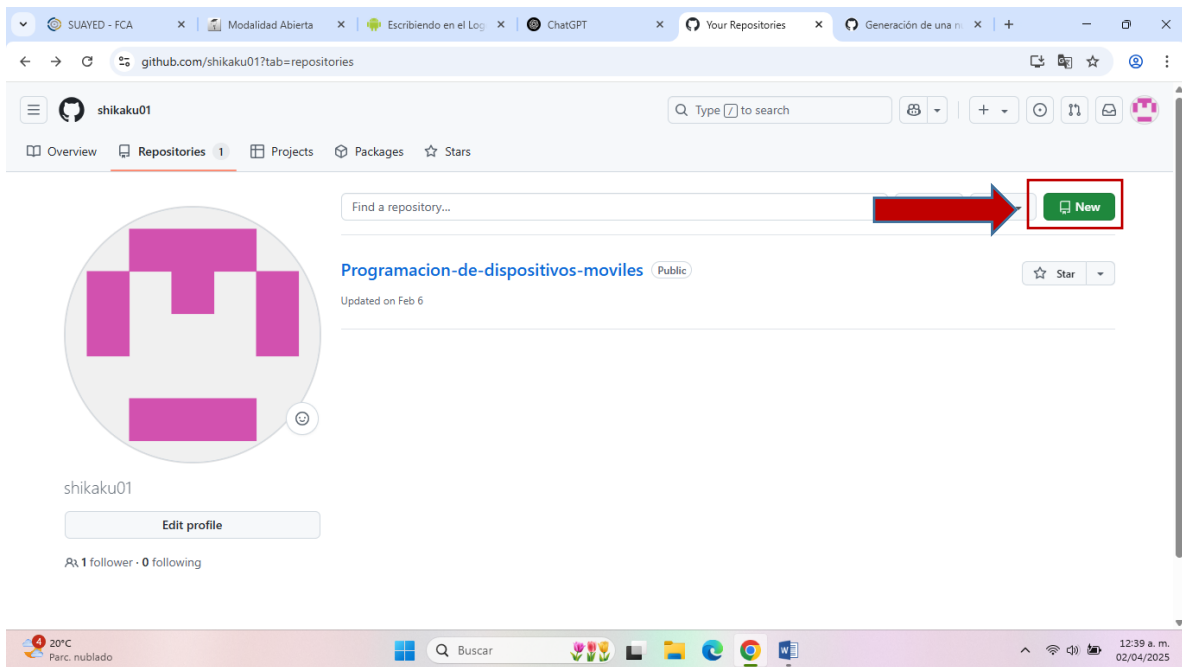


Agregamos el código de la llave publica y la agregamos

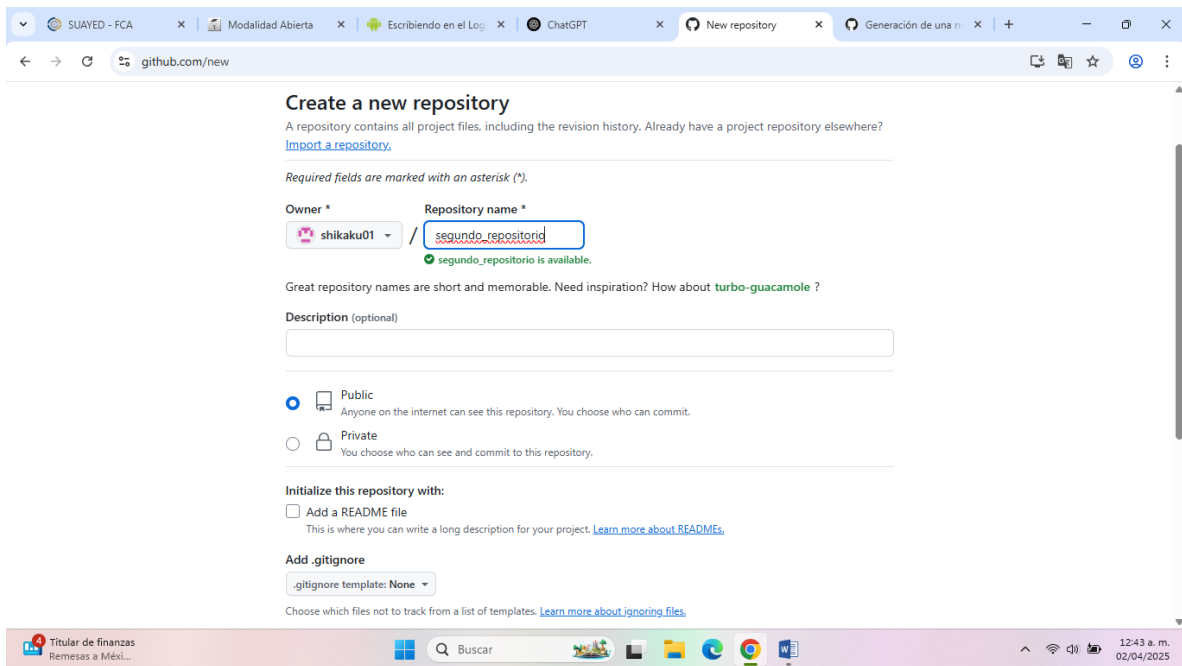


Y listo, ya está nuestra llave puesta en github

3. Crear un segundo repositorio público en github.com



Nos vamos a la sección new para crear el nuevo repositorio



Le asignamos un nombre y lo creamos

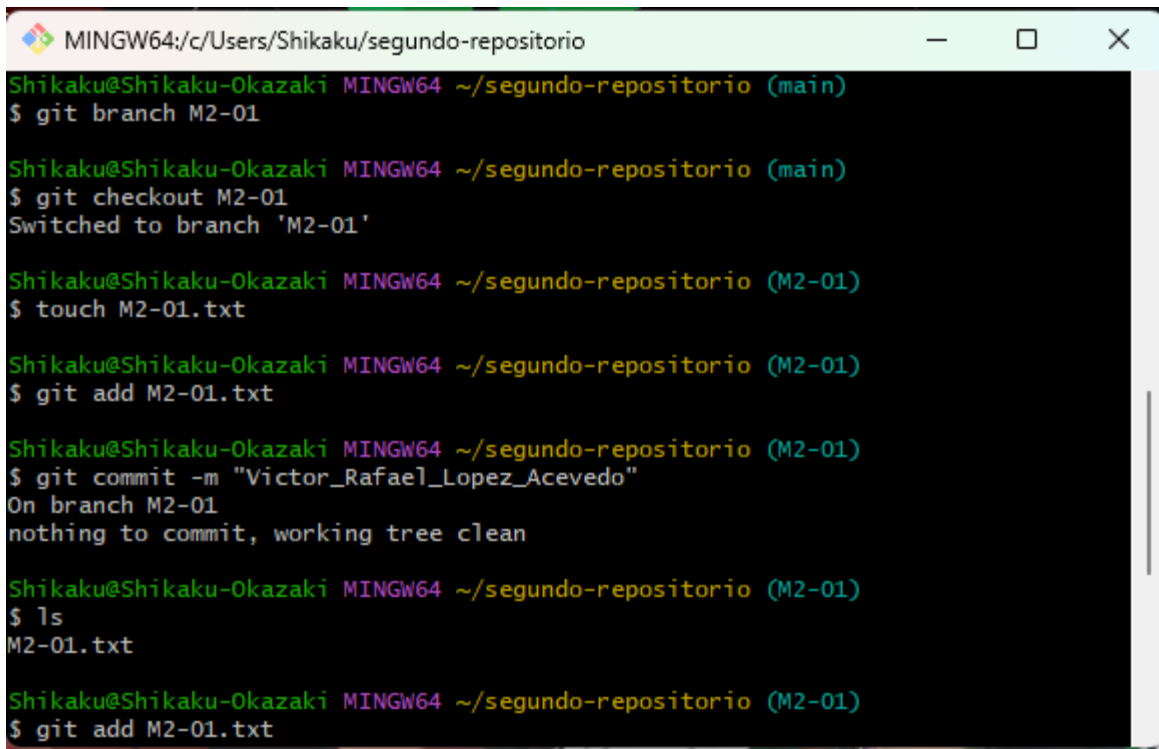
4. Crear una rama bajo el nombre “M2-01”

5. En el directorio raíz subir un archivo en blanco bajo el nombre M2-01.txt

6. Hacer un commit con su nombre completo.

Nombre_apellidoPaterno_apellidoMaterno (ej cristian_cardoso_arellano)

7. Hacer push de los cambios

A screenshot of a terminal window titled 'MINGW64:/c/Users/Shikaku/segundo-repositorio'. The terminal shows the following sequence of commands and output:

```
Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (main)
$ git branch M2-01

Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (main)
$ git checkout M2-01
Switched to branch 'M2-01'

Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ touch M2-01.txt

Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ git add M2-01.txt

Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ git commit -m "Victor_Rafael_Lopez_Acevedo"
On branch M2-01
nothing to commit, working tree clean

Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ ls
M2-01.txt

Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ git add M2-01.txt
```

Una vez creamos el segundo repositorio desde github, nos vamos a git bash y clonamos el repositorio.

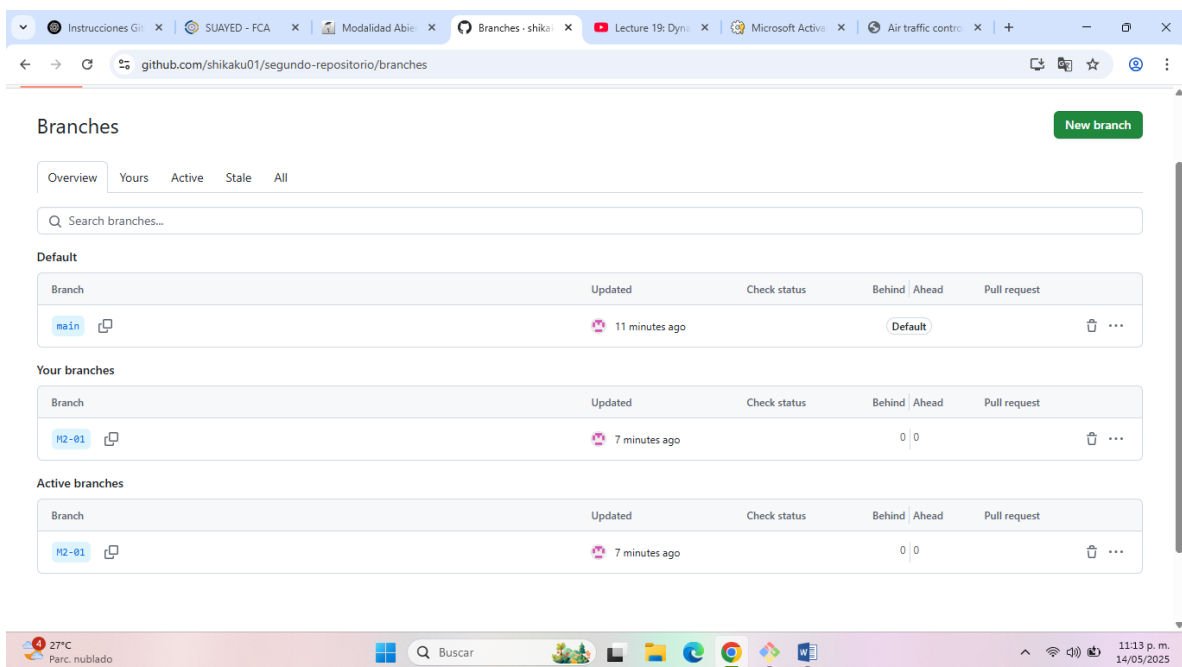
Como podemos ver en la imagen, creamos la rama que se llamara M2-01 y nos posicionamos en ella.

Creamos el archivo en blanco con el nombre “M2-01.txt” y lo agregamos.

Hacemos el commit con nuestro nombre completo y hacemos “push” para guardar los cambios

```
MINGW64:/c/Users/Shikaku/segundo-repositorio
Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ ls
M2-01.txt
Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ git add M2-01.txt
Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ git commit -m "Victor_Rafael_Lopez_Acevedo"
On branch M2-01
nothing to commit, working tree clean
Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$ git push origin M2-01
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'M2-01' on GitHub by visiting:
remote:   https://github.com/shikaku01/segundo-repositorio/pull/new/M2-01
remote:
To github.com:shikaku01/segundo-repositorio.git
 * [new branch]      M2-01 -> M2-01
Shikaku@Shikaku-Okazaki MINGW64 ~/segundo-repositorio (M2-01)
$
```

En esta imagen podemos ver que al hacer el push nos muestra que se guardan los cambios y podemos verificarlo en nuestro perfil de github



Podemos ver que la rama esta creada y si le damos click veremos que se encuentra el archivo en blanco que se creo

The screenshot shows a GitHub repository page for 'segundo-repositorio' by user 'shikaku01'. The repository is public. The main content area shows a file 'M2-01.txt' committed by Victor_Rafael_Lopez_Acevedo 12 minutes ago. The 'About' section is empty. The 'README' section prompts the user to 'Add a README'. The right sidebar shows 'About' (no description), 'Activity' (0 stars, 1 watching, 0 forks), 'Releases' (no releases published), and 'Packages' (no packages published). The browser tabs at the top include 'Instrucciones G...', 'SUAYED - FCA', 'Modalidad Abie...', 'shikaku01/segun...', 'Lecture 19: Dyn...', 'Microsoft Activ...', and 'Air traffic contr...'.