

# Documentación y Control de Versiones

**Unidad: DAW05**

Ciclo formativo y módulo: Desarrollo de aplicaciones Web. Despliegue de  
Aplicaciones Web  
Curso académico: 2021/2022

Dámaris Angulo Navarro

# Indice

Ejercicio 1.- Documentación de Aplicaciones.....	3
Actividad 1.1.- Instalación de PHPDocumentor y Etiquetas más importantes.....	3
Actividad 1.2.- Documentación de un Script PHP.....	7
Actividad 1.3.- Generación de documentación con PHPDocumentor.....	8
Ejercicio 2.- Control de versiones con GIT/GitHUB.....	10
Actividad 2.1.- Tutorial de Instalación de GIT.....	10
Actividad 2.2.- Configuración inicial de GIT.....	13
Actividad 2.2.- Clonando un repositorio de GitHub.....	15

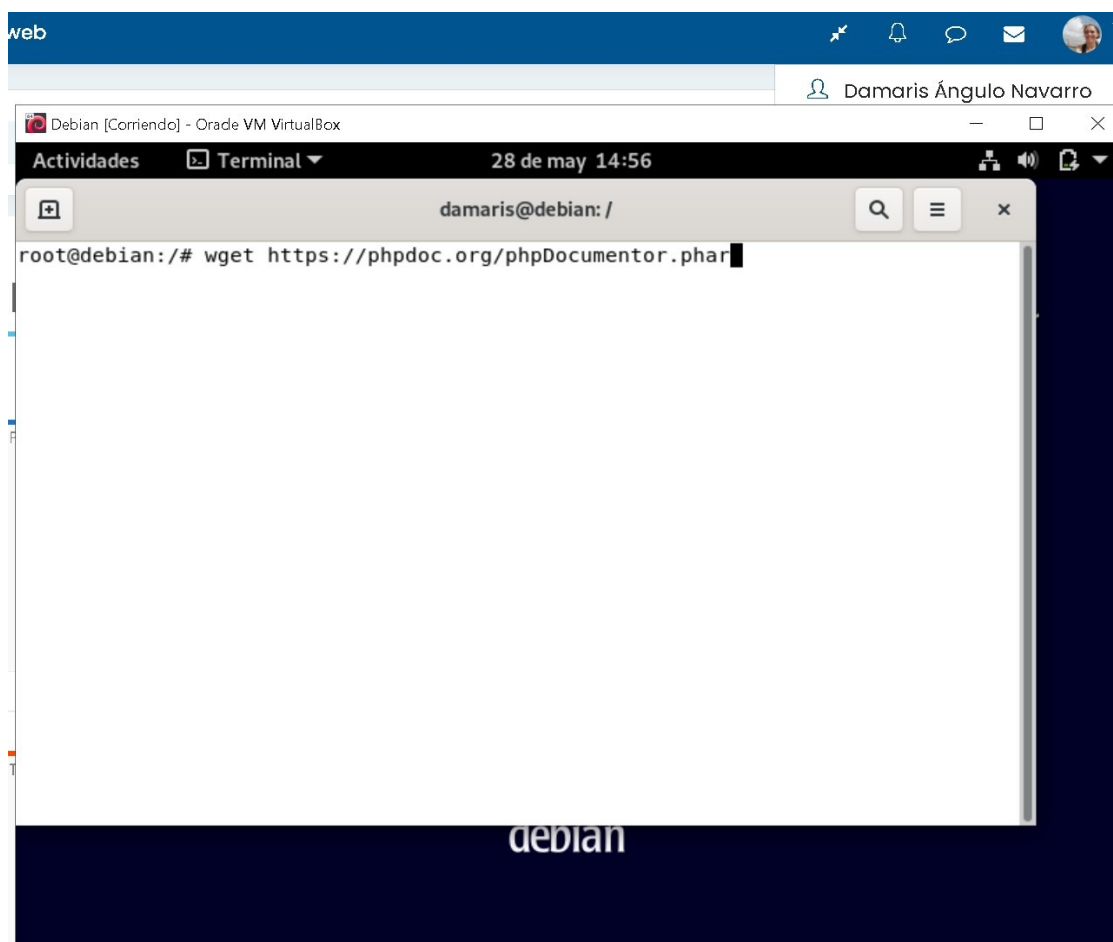
## Ejercicio 1.- Documentación de Aplicaciones.

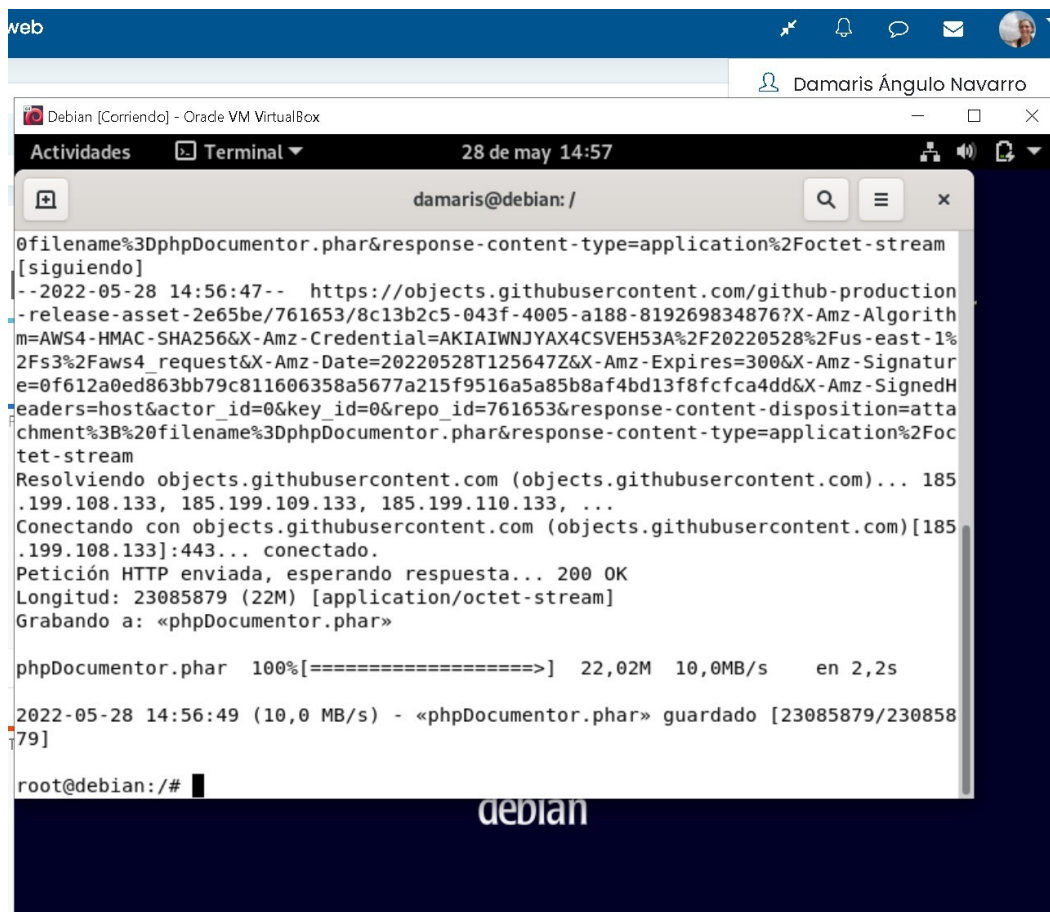
En este primer ejercicio realizaremos la instalación de PhpDocumentor y generaremos documentación a partir de un bloque de código PHP. Previamente analizaremos la utilidad de esta herramienta.

### Actividad 1.1.- Instalación de PHPDocumentor y Etiquetas más importantes.

Instala la herramienta PhpDocumentor en tu servidor Linux y comenta los aspectos más importantes de tal herramienta, así como las etiquetas principales que se usan. (Se recomienda hacer la instalación manual. En el foro de la actividad se darán indicaciones para poder realizar la instalación de la última versión.)

*Descargamos el paquete .phar de la página de PhpDocumentor*





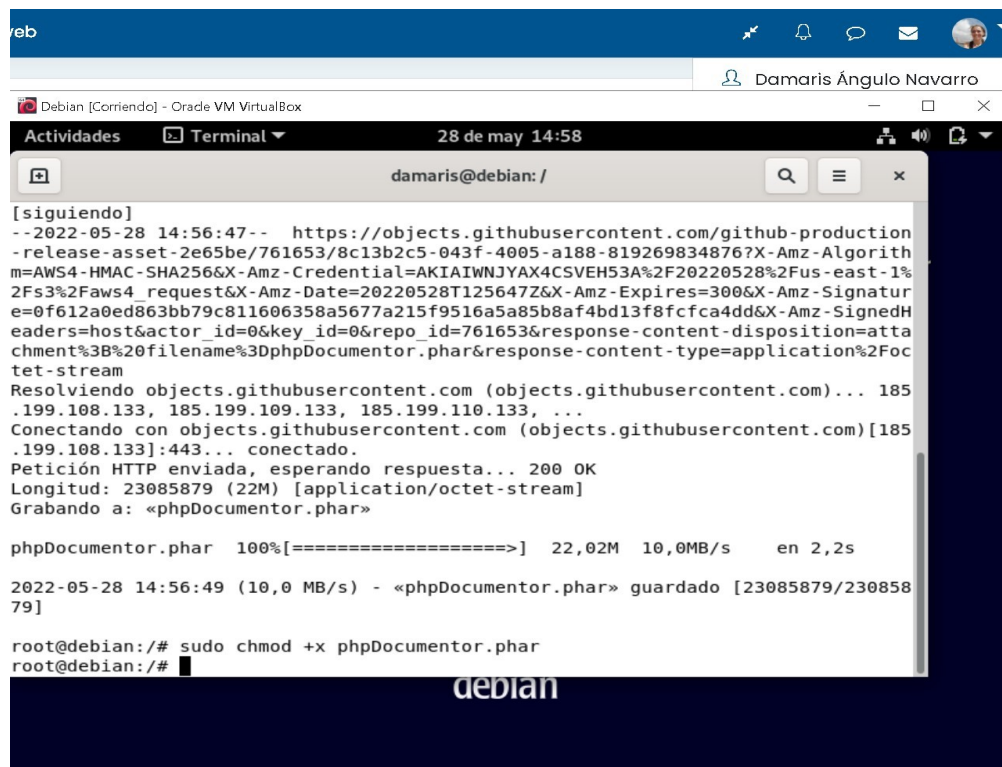
```
web
Damaris Ángulo Navarro
Debian [Corriendo] - Oracle VM VirtualBox
28 de may 14:57
damaris@debian: /
0filename%3DphpDocumentor.phar&response-content-type=application%2Foctet-stream
[siguiendo]
--2022-05-28 14:56:47-- https://objects.githubusercontent.com/github-production
-release-asset-2e65be/761653/8c13b2c5-043f-4005-a188-819269834876?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220528%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20220528T125647Z&X-Amz-Expires=300&X-Amz-Signature=0f612a0ed863bb79c811606358a5677a215f9516a5a85b8af4bd13f8fcfa4dd&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=761653&response-content-disposition=attachment%3B%20filename%3DphpDocumentor.phar&response-content-type=application%2Foctet-stream
Resolviendo objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Conectando con objects.githubusercontent.com (objects.githubusercontent.com)[185.199.108.133]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 23085879 (22M) [application/octet-stream]
Grabando a: «phpDocumentor.phar»

phpDocumentor.phar 100%[=====] 22,02M 10,0MB/s en 2,2s

2022-05-28 14:56:49 (10,0 MB/s) - «phpDocumentor.phar» guardado [23085879/23085879]

root@debian:/#
```

*Cambiamos los permisos para que se pueda ejecutar*



```
web
Damaris Ángulo Navarro
Debian [Corriendo] - Oracle VM VirtualBox
28 de may 14:58
damaris@debian: /
[siguiendo]
--2022-05-28 14:56:47-- https://objects.githubusercontent.com/github-production
-release-asset-2e65be/761653/8c13b2c5-043f-4005-a188-819269834876?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220528%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20220528T125647Z&X-Amz-Expires=300&X-Amz-Signature=0f612a0ed863bb79c811606358a5677a215f9516a5a85b8af4bd13f8fcfa4dd&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=761653&response-content-disposition=attachment%3B%20filename%3DphpDocumentor.phar&response-content-type=application%2Foctet-stream
Resolviendo objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Conectando con objects.githubusercontent.com (objects.githubusercontent.com)[185.199.108.133]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 23085879 (22M) [application/octet-stream]
Grabando a: «phpDocumentor.phar»

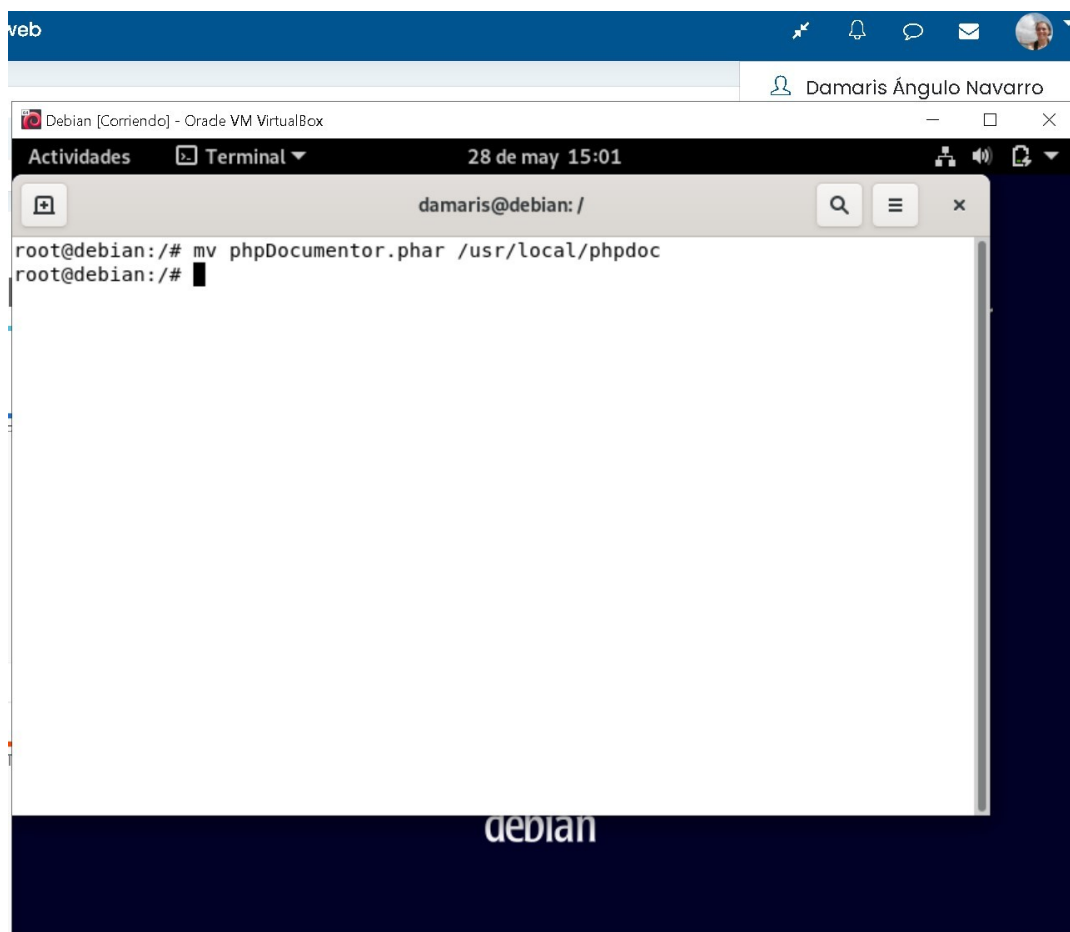
phpDocumentor.phar 100%[=====] 22,02M 10,0MB/s en 2,2s

2022-05-28 14:56:49 (10,0 MB/s) - «phpDocumentor.phar» guardado [23085879/23085879]

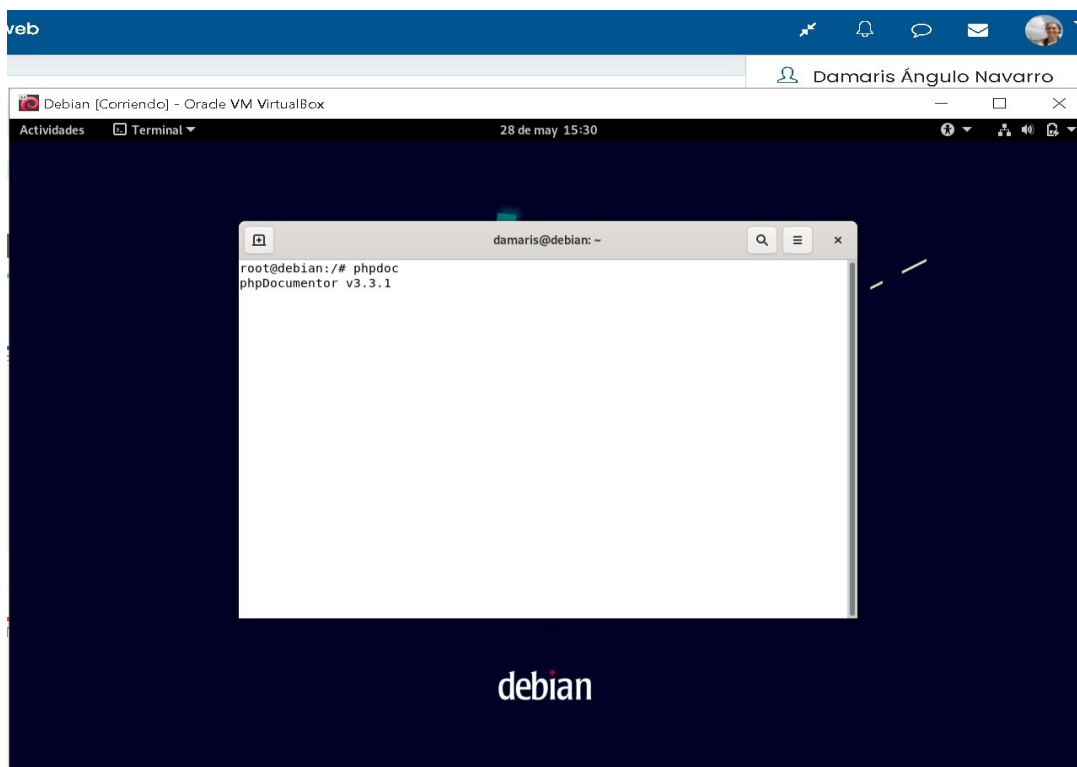
root@debian:/# sudo chmod +x phpDocumentor.phar
root@debian:/#
```



*Lo movemos a la carpeta /usr/local/bin cambiando además su nombre por uno más corto y fácil de recordar (phpdoc)*



*Lo ejecutamos con phpdoc*



Los aspectos más importantes de phpDocumentor son:

- Hace comentarios que pueda leerse en un método estándar para animar a los programadores a definir y comentar los aspectos del código que normalmente se ignoran.
- Permite que los generadores de documentos externos como phpDocumentor puedan crear la documentación API en buen formato y fácil de entender.
- Permite que algunos IDEs como Zend Studio, NetBeans, Aptana Studio y PhpStorm interpreten los tipos de variables y otras ambigüedades en el lenguaje de programación.

Las etiquetas utilizadas en phpDocumentor son:

- **@access** Si @access es 'private' no se genera documentación para el elemento (a menos que se indique explícitamente). Muy interesante si sólo se desea generar documentación sobre la interfaz (métodos públicos) pero no sobre la implementación (métodos privados).
- **@author** Autor del código
- **@copyright** Información sobre derechos
- **@deprecated** Para indicar que el elemento no debería utilizarse, ya que en futuras versiones podría no estar disponible.
- **@example** Permite especificar la ruta hasta un fichero con código PHP. phpDocumentor se encarga de mostrar el código resaltado (syntax-highlighted).
- **@ignore** Evita que phpDocumentor documente un determinado elemento.
- **@internal** inline {@internal} Para incluir información que no debería aparecer en la documentación pública, pero sí puede estar disponible como documentación interna para desarrolladores.
- **@link** inline {@link} Para incluir un enlace (http://...) a un determinado recurso.
- **@see** Se utiliza para crear enlaces internos (enlaces a la documentación de un elemento).
- **@since** Permite indicar que el elemento está disponible desde una determinada versión del paquete o distribución.
- **@version** Versión actual del elemento

Para declaración de funciones (elemento function):

- **@global** Permite especificar el uso de variables globales dentro de la función.
- **@param** Parámetros que recibe la función. Formato: \* @param tipo \$nombre\_var comentario
- **@return** Valor devuelto por la función. Formato: \* @return tipo comentario

## Actividad 1.2.- Documentación de un Script PHP.

En esta actividad debes crear en tu servidor un script PHP con el nombre practica-XXXXXX.php, donde XXXXXX será tu primer apellido. A continuación escribe dentro de este script bloques de código PHP y DocBlocks para que luego se pueda generar la documentación correspondiente. El script debe contener al menos dos funciones documentadas, indicando mediante las etiquetas vistas en la unidad los siguientes elementos:

Parámetros de entrada de la función.

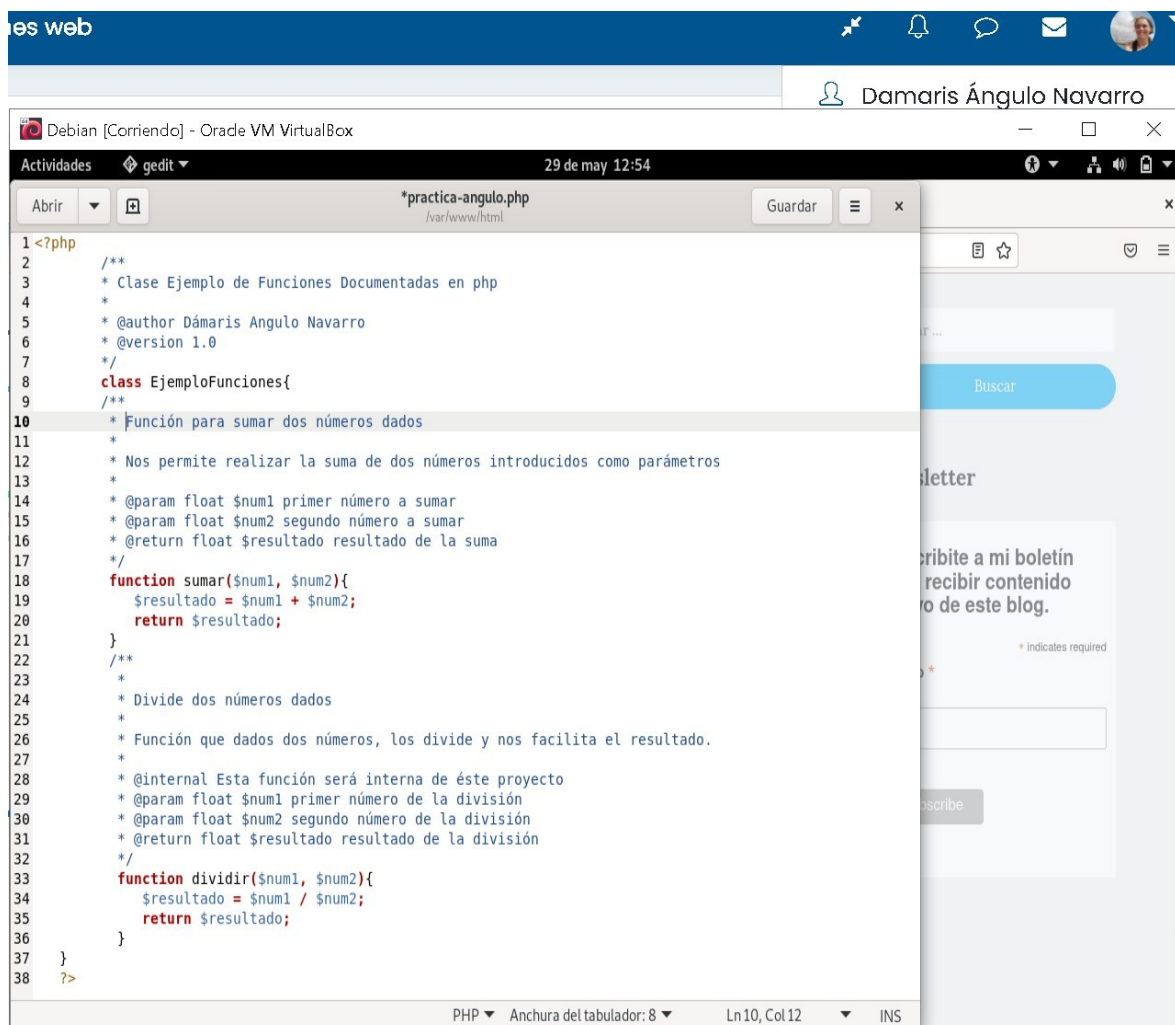
Parámetros de devuelve la función.

Autor y versión del script.

Una anotación que solo sea visible en la documentación para desarrolladores.

(Para quien no conozca PHP, se facilitará en el foro de la tarea ejemplos de código que podrá reutilizar).

*Generamos el archivo practica-angulo.php*



The screenshot shows a web browser interface. The top bar is blue with the text "es web" and several icons. Below it, a user profile "Damaris Ángulo Navarro" is visible. The main content area displays a code editor for a file named "practica-angulo.php". The code is written in PHP and includes DocBlock comments for two functions: "sumar" and "dividir". The "sumar" function is documented with parameters \$num1 and \$num2, and a return value \$resultado. The "dividir" function is documented with parameters \$num1 and \$num2, and a return value \$resultado. The code is as follows:

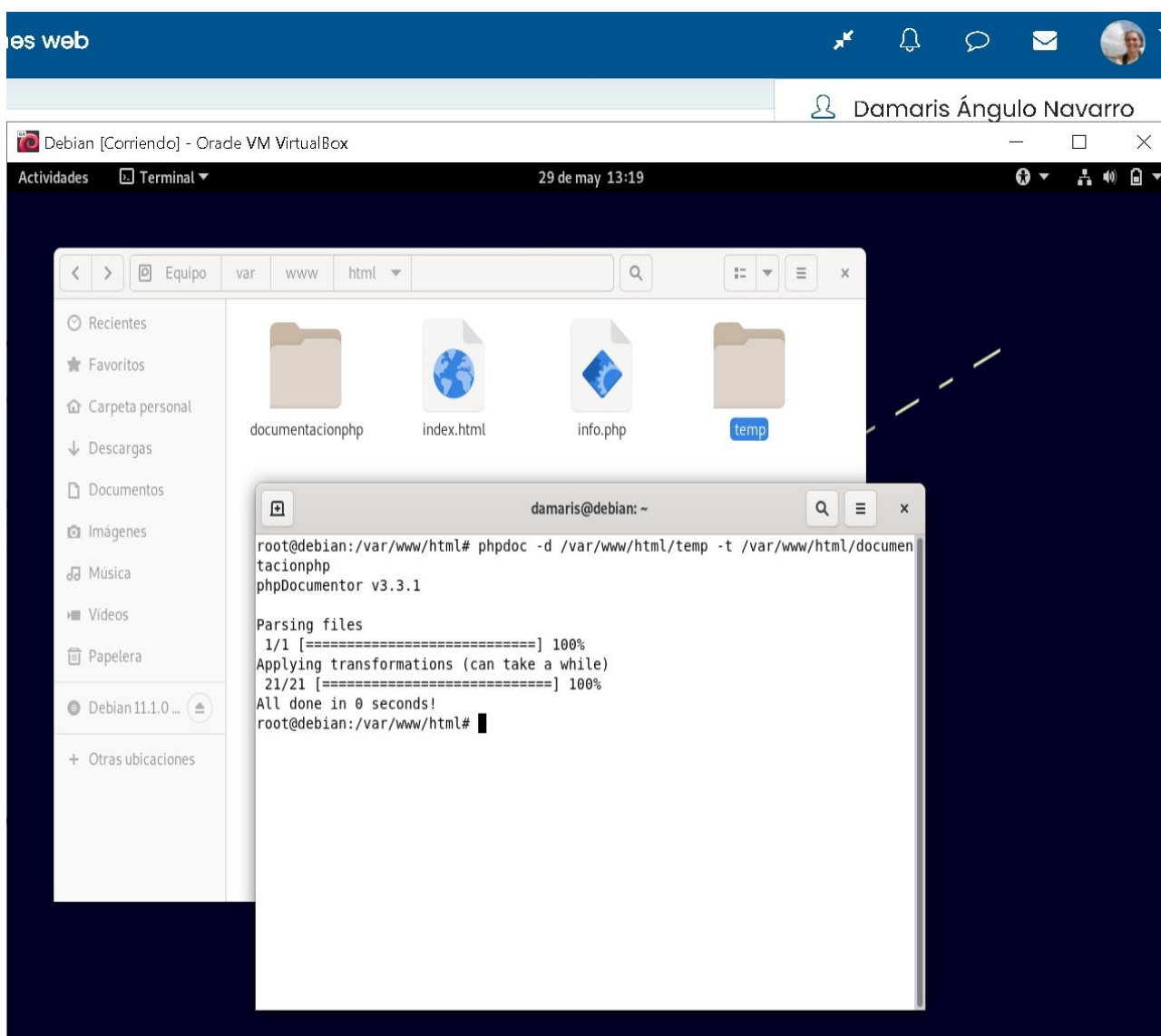
```
1 <?php
2 /**
3  * Clase Ejemplo de Funciones Documentadas en php
4  *
5  * @author Damaris Angulo Navarro
6  * @version 1.0
7  */
8 class EjemploFunciones{
9     /**
10     * Función para sumar dos números dados
11     *
12     * Nos permite realizar la suma de dos números introducidos como parámetros
13     *
14     * @param float $num1 primer número a sumar
15     * @param float $num2 segundo número a sumar
16     * @return float $resultado resultado de la suma
17     */
18     function sumar($num1, $num2){
19         $resultado = $num1 + $num2;
20         return $resultado;
21     }
22     /**
23     *
24     * Divide dos números dados
25     *
26     * Función que dados dos números, los divide y nos facilita el resultado.
27     *
28     * @internal Esta función será interna de éste proyecto
29     * @param float $num1 primer número de la división
30     * @param float $num2 segundo número de la división
31     * @return float $resultado resultado de la división
32     */
33     function dividir($num1, $num2){
34         $resultado = $num1 / $num2;
35         return $resultado;
36     }
37 }
38 ?>
```

The sidebar on the right contains a search bar with the text "Buscar" and a newsletter sign-up form with the text "Escribite a mi boletín recibir contenido de este blog." and a "subscribe" button.

## Actividad 1.3.- Generación de documentación con PHPDocumentor.

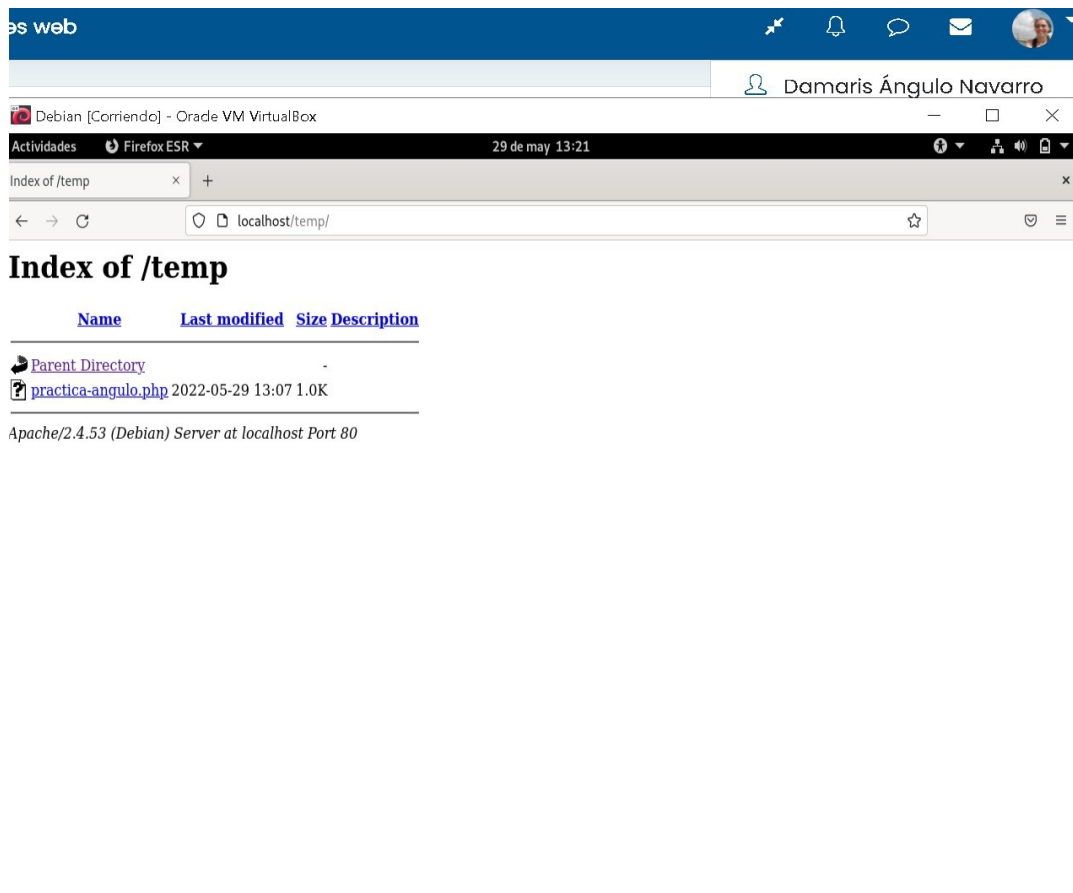
Después de revisar la documentación, especialmente el apartado 2.2, instala la aplicación PHPDocumentor en tu máquina Linux y genera la documentación del script PHP que hemos creado en la actividad 1.2. Ten en cuenta que el directorio de salida debe ser una carpeta de tu servidor web para que podamos ver el contenido desde el navegador. (por ejemplo si uso como carpeta de salida la ruta /var/www/html/doc-profesor/ luego podré ver la documentación en <http://localhost/doc-profesor> el árbol de carpetas y archivos que genera como salida (comando tree de Linux) y un par de capturas del navegador web mostrando la documentación generada.

*Guardamos nuestro script en la carpeta temp y creamos la carpeta donde se guardará la documentación: documentacionphp y lanzamos el siguiente comando que generará la documentación:*

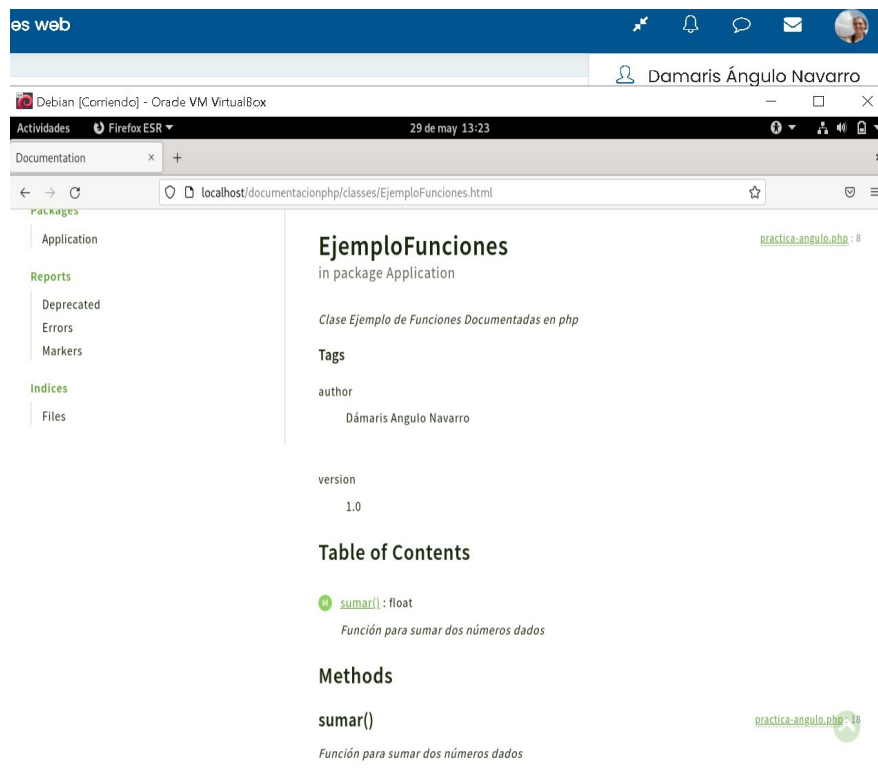


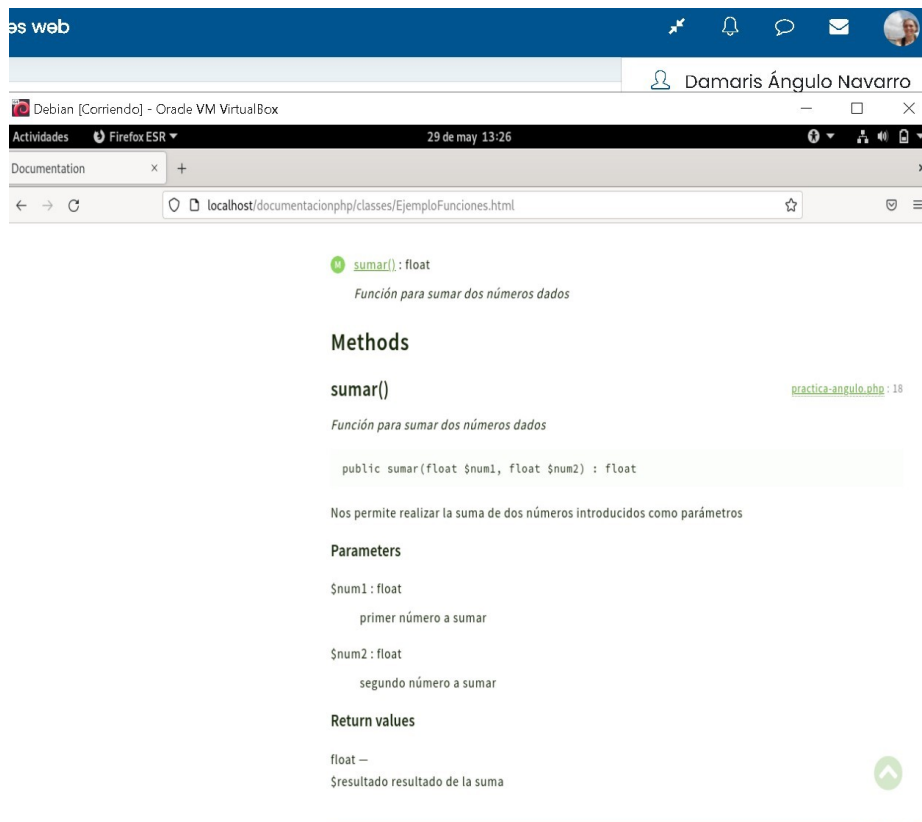


Vemos la carpeta temp con el script:



Vemos la documentación generada al entrar en `localhost/documentacionphp/EjemploFunciones.html`:





## Ejercicio 2.- Control de versiones con GIT/GitHUB.

En este ejercicio realizaremos la instalación y configuración de la aplicación GIT. Para ello realizaremos las siguientes actividades:

### Actividad 2.1.- Tutorial de Instalación de GIT.

Realiza la instalación de GIT en tu servidor Linux y realiza un breve tutorial donde se explique la instalación y configuración de esta herramienta en tu computadora, respondiendo a cuestiones teóricas como ¿Qué es Git? ¿Para que sirve? ¿Cuál es la última versión? etc. Documenta todo el proceso e incluye las capturas de los procesos importantes.

## ¿Qué es Git?

*Es el sistema de control de versiones moderno más utilizado del mundo. Git es un proyecto de código abierto maduro y con un mantenimiento activo que desarrolló originalmente Linus Torvalds, el famoso creador del kernel del sistema operativo Linux, en 2005.*

*En Git, la copia de trabajo del código de cada desarrollador es también un repositorio que puede albergar el historial completo de todos los cambios.*

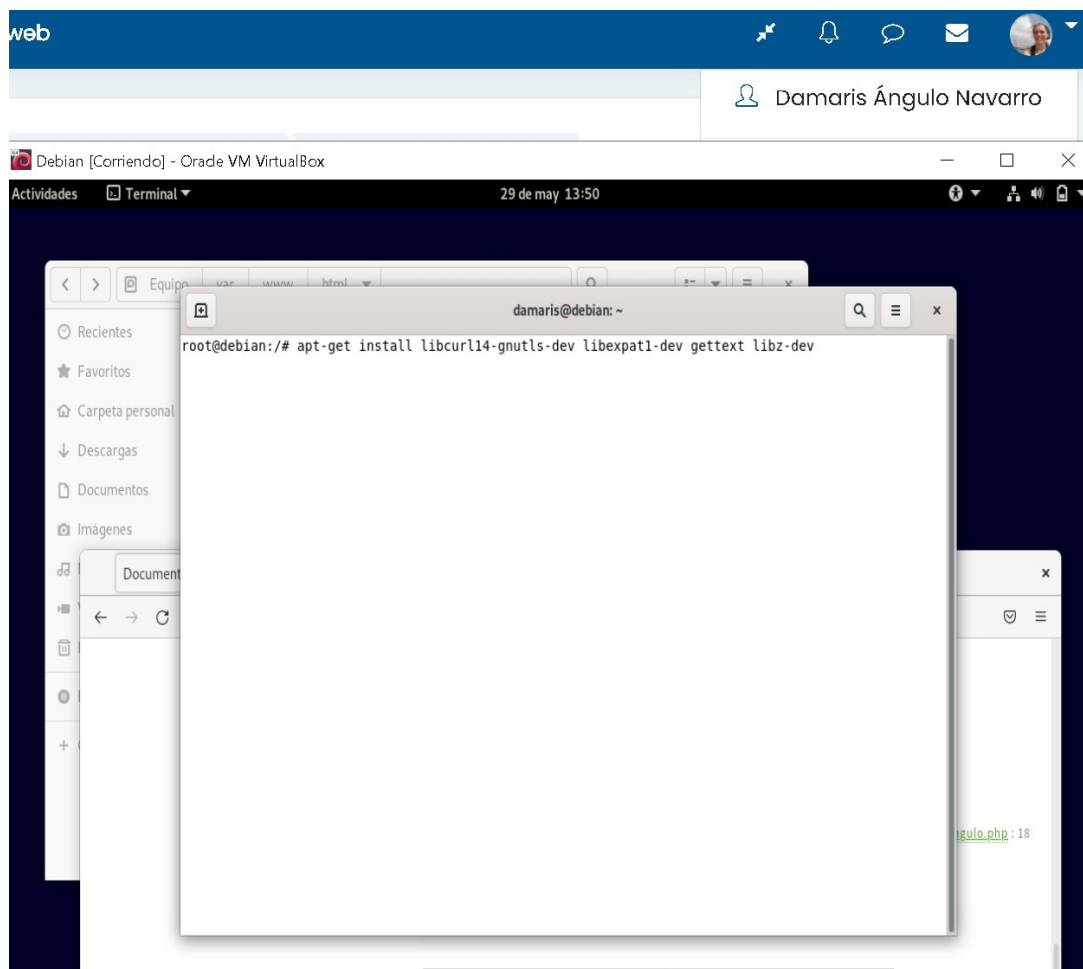
## ¿Para que sirve?

*El control de versiones distribuido permite a los desarrolladores descargar un software, realizar cambios y subir la versión que han modificado. Todas las modificaciones subidas se guardan en versiones independientes, no sobrescribiendo en el archivo original.*

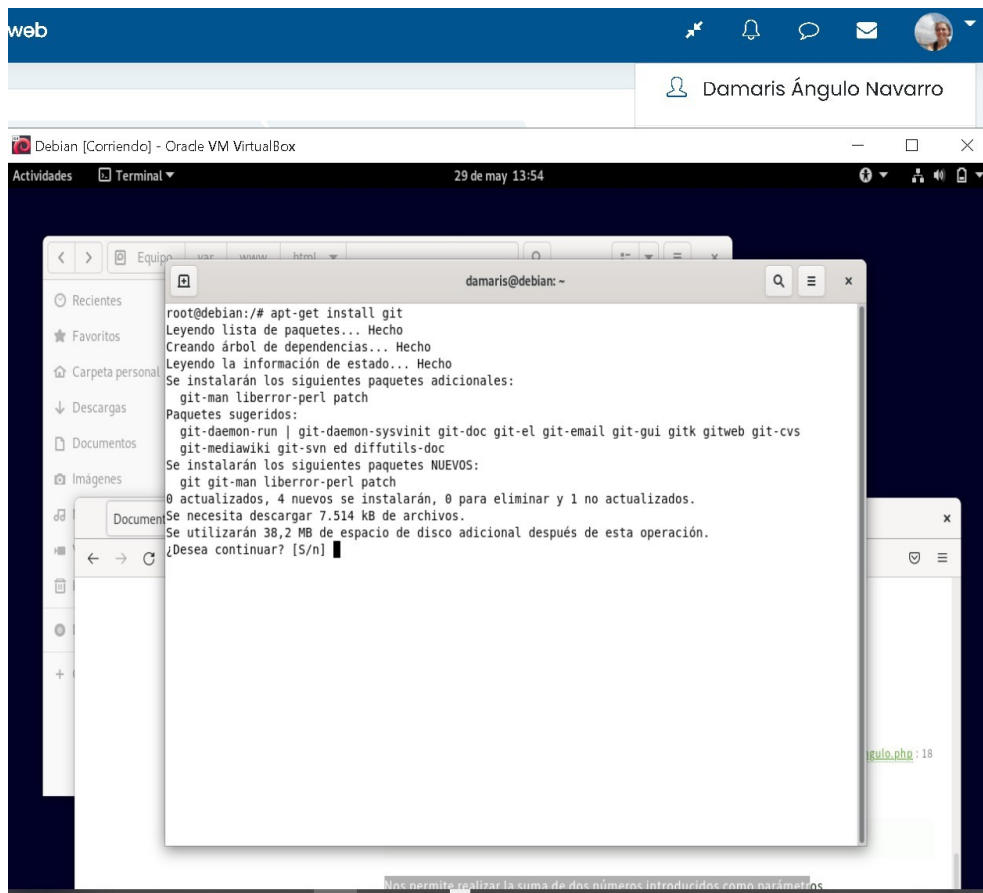
## ¿Cuál es la última versión?

*Última versión estable: 2.35.1 (info) ( 19 de enero de 2022 ).*

*Para instalar git primeramente instalaremos las librerías necesarias:*



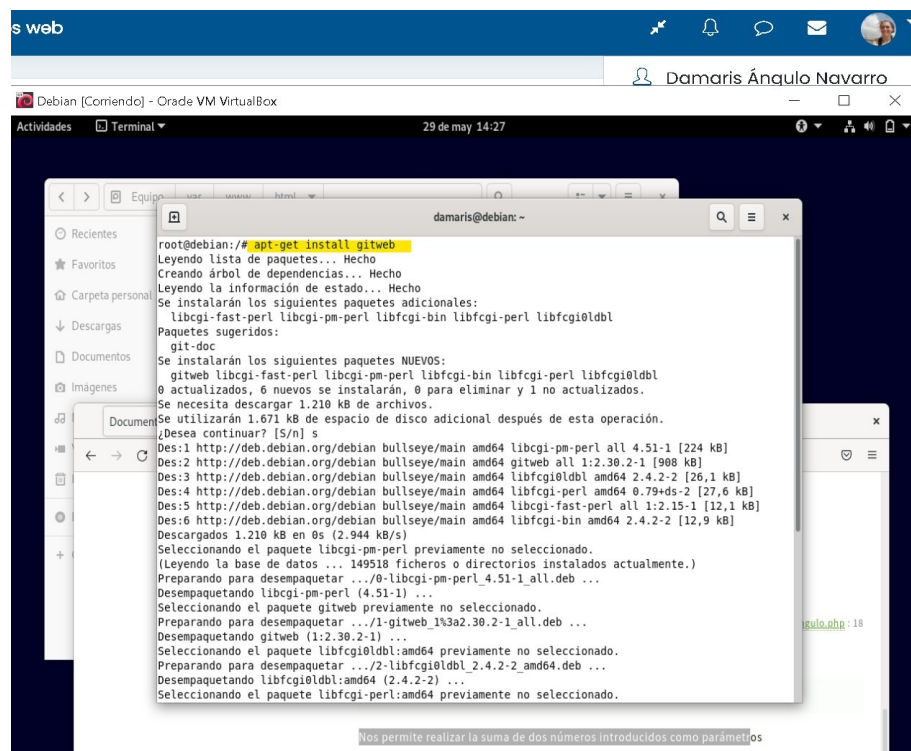
*A continuación realizamos la instalación de Git con el siguiente comando:*



The screenshot shows a terminal window titled 'damaris@debian: ~' with the following output:

```
root@debian:~# apt-get install git
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
git-man liberror-perl patch
Paquetes sugeridos:
git-daemon-run | git-daemon-sysvinit git-doc git-el git-email git-gui gitk gitweb git-cvs
git-mediawiki git-svn ed diffutils-doc
Se instalarán los siguientes paquetes NUEVOS:
git git-man liberror-perl patch
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 1 no actualizados.
Se necesita descargar 7.514 kB de archivos.
Se utilizarán 38,2 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

*E instalamos también gitweb*



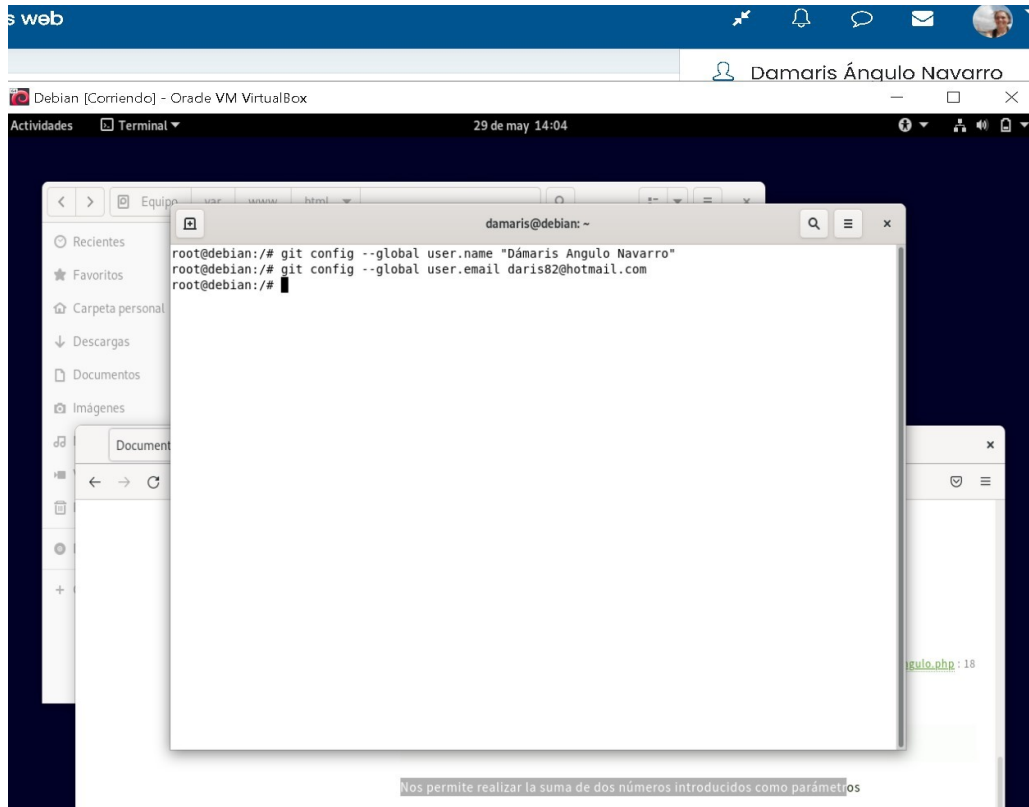
The screenshot shows a terminal window titled 'damaris@debian: ~' with the following output:

```
root@debian:~# apt-get install gitweb
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
libcgi-fast-perl libcgi-pm-perl libfcgi-bin libfcgi-perl libfcgi0ldb
Paquetes sugeridos:
git-doc
Se instalarán los siguientes paquetes NUEVOS:
gitweb libcgi-fast-perl libcgi-pm-perl libfcgi-bin libfcgi-perl libfcgi0ldb
0 actualizados, 6 nuevos se instalarán, 0 para eliminar y 1 no actualizados.
Se necesita descargar 1.210 kB de archivos.
Se utilizarán 1.671 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://deb.debian.org/debian bullseye/main amd64 libcgi-pm-perl all 4.51-1 [224 kB]
Des:2 http://deb.debian.org/debian bullseye/main amd64 gitweb all 1:2.30.2-1 [908 kB]
Des:3 http://deb.debian.org/debian bullseye/main amd64 libfcgi0ldb amd64 2.4.2-2 [26,1 kB]
Des:4 http://deb.debian.org/debian bullseye/main amd64 libfcgi-perl amd64 0.79+ds-2 [27,6 kB]
Des:5 http://deb.debian.org/debian bullseye/main amd64 libcgi-fast-perl all 1:2.15-1 [12,1 kB]
Des:6 http://deb.debian.org/debian bullseye/main amd64 libfcgi-bin amd64 2.4.2-2 [12,9 kB]
Descargados 1.210 kB en 0s (2.944 kB/s)
Seleccionando el paquete libcgi-pm-perl previamente no seleccionado.
(Leyendo la base de datos ... 149518 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../0:libcgi-pm-perl_4.51-1_all.deb ...
Desempaquetando libcgi-pm-perl (4.51-1) ...
Seleccionando el paquete gitweb previamente no seleccionado.
Preparando para desempaquetar .../1:gitweb_1:2.30.2-1_all.deb ...
Desempaquetando gitweb (1:2.30.2-1) ...
Seleccionando el paquete libfcgi0ldb:amd64 previamente no seleccionado.
Preparando para desempaquetar .../2:libfcgi0ldb_2.4.2-2_amd64.deb ...
Desempaquetando libfcgi0ldb:amd64 (2.4.2-2) ...
Seleccionando el paquete libfcgi-perl:amd64 previamente no seleccionado.
```

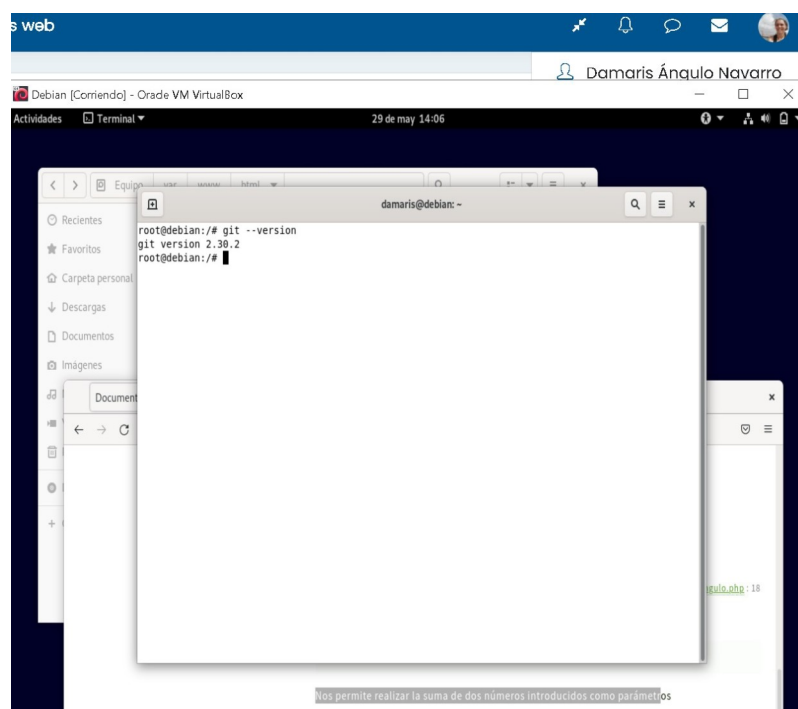
## Actividad 2.2.- Configuración inicial de GIT.

Una vez instalada la herramienta GIT deberás realizar la configuración inicial:

Configura tu nombre, apellidos y cuenta de correo



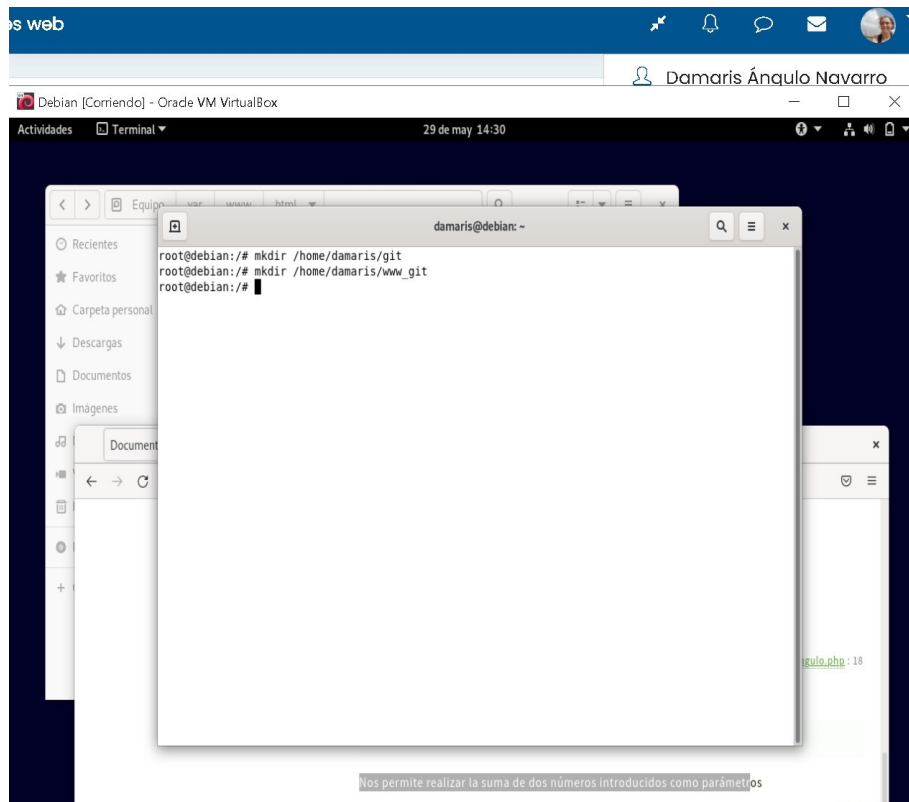
Muestra la versión instalada.



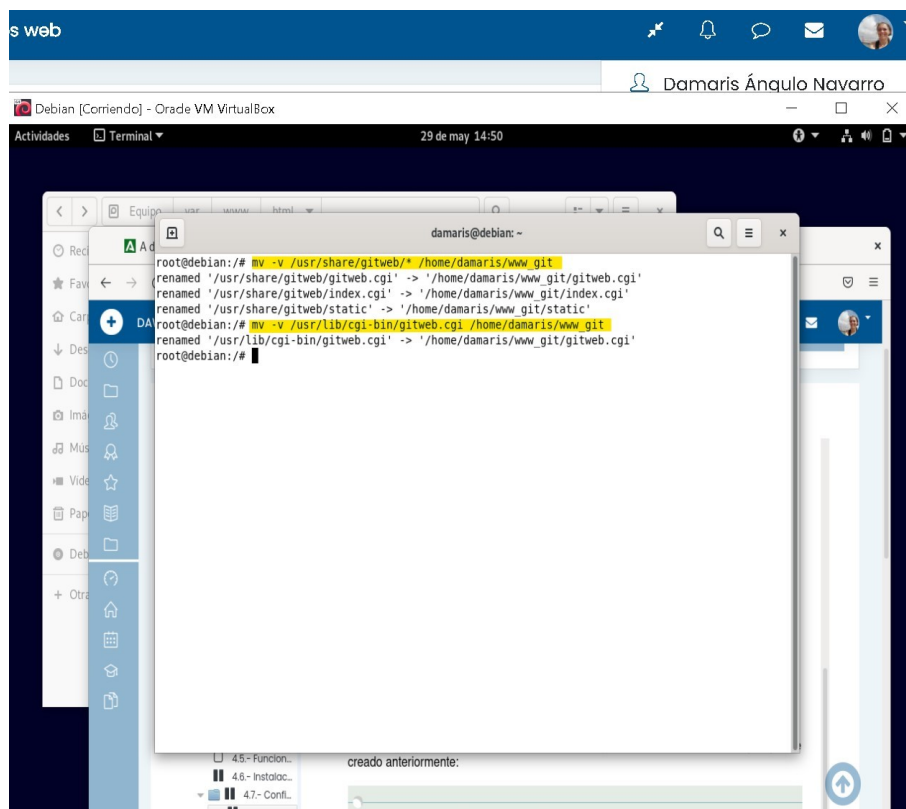


Indica cuál es tu directorio de trabajo.

*Crearemos los directorios de trabajo como se ve en la captura:*



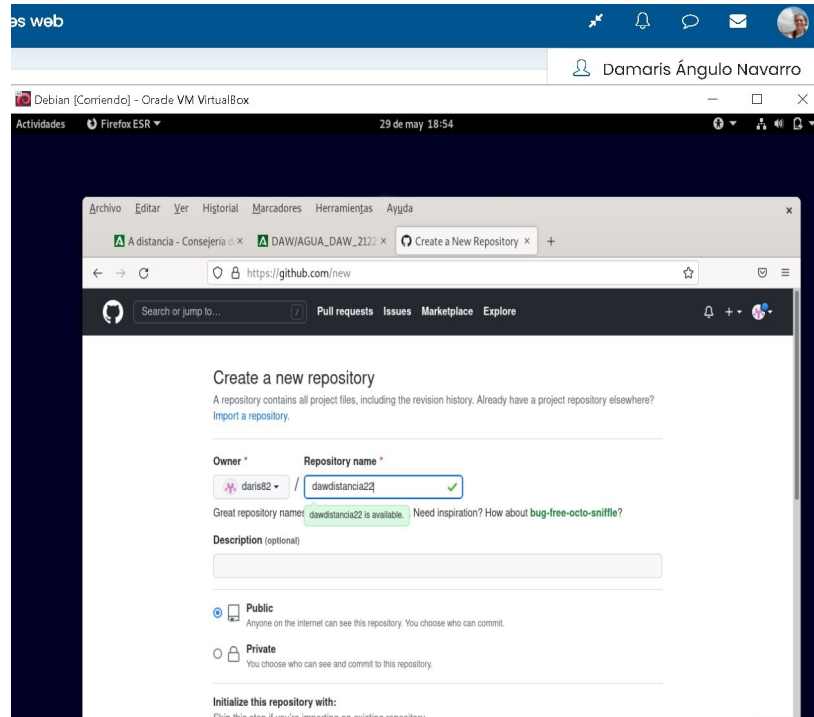
*Movemos los archivos básicos del gitweb al directorio de trabajo Apache y reiniciamos Apache*



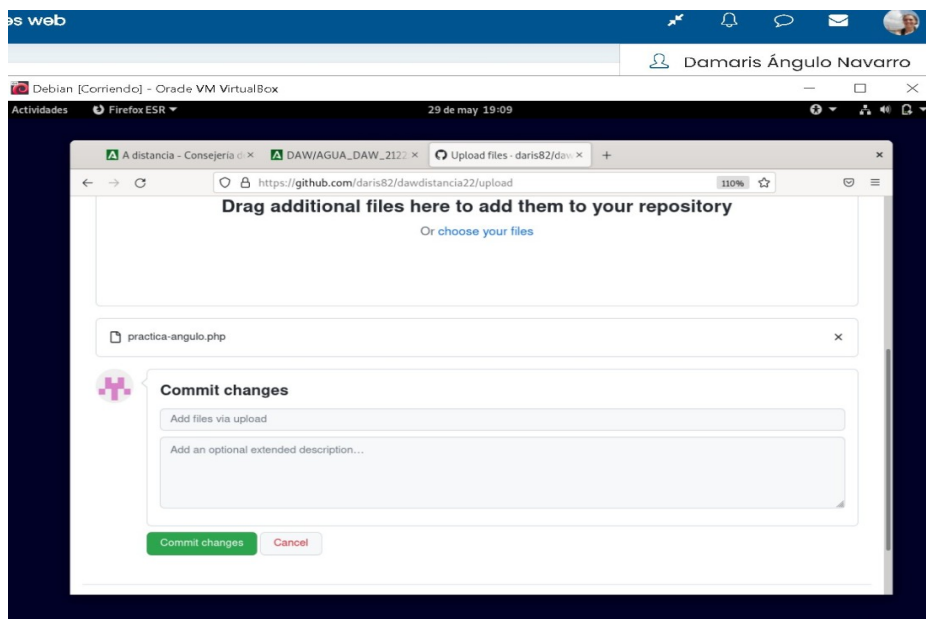
## Actividad 2.2.- Clonando un repositorio de GitHub

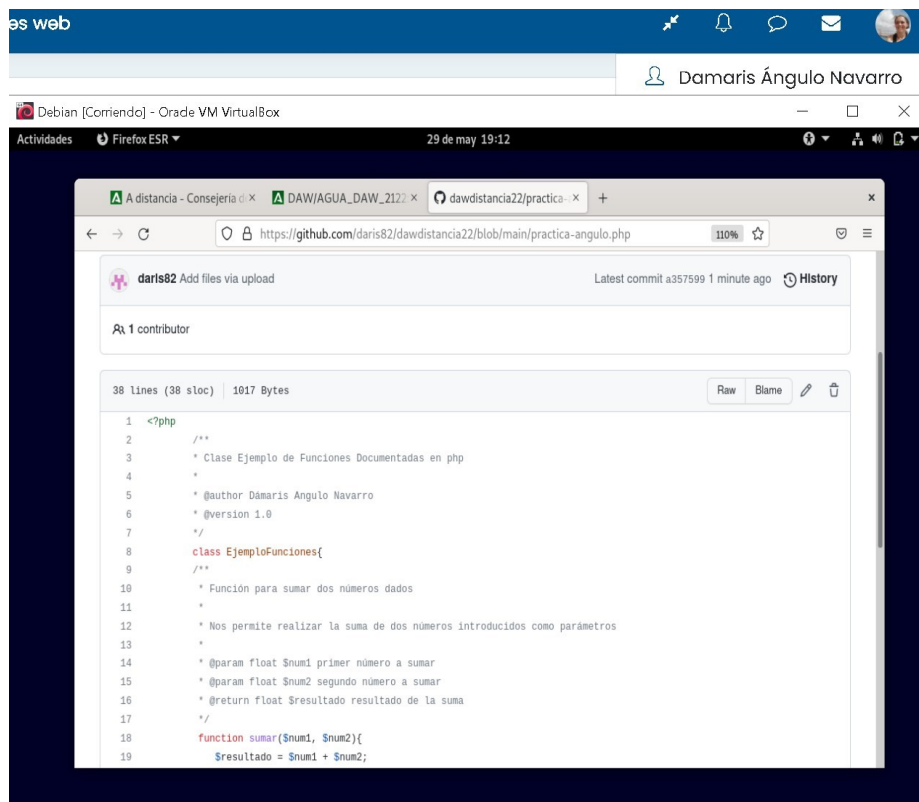
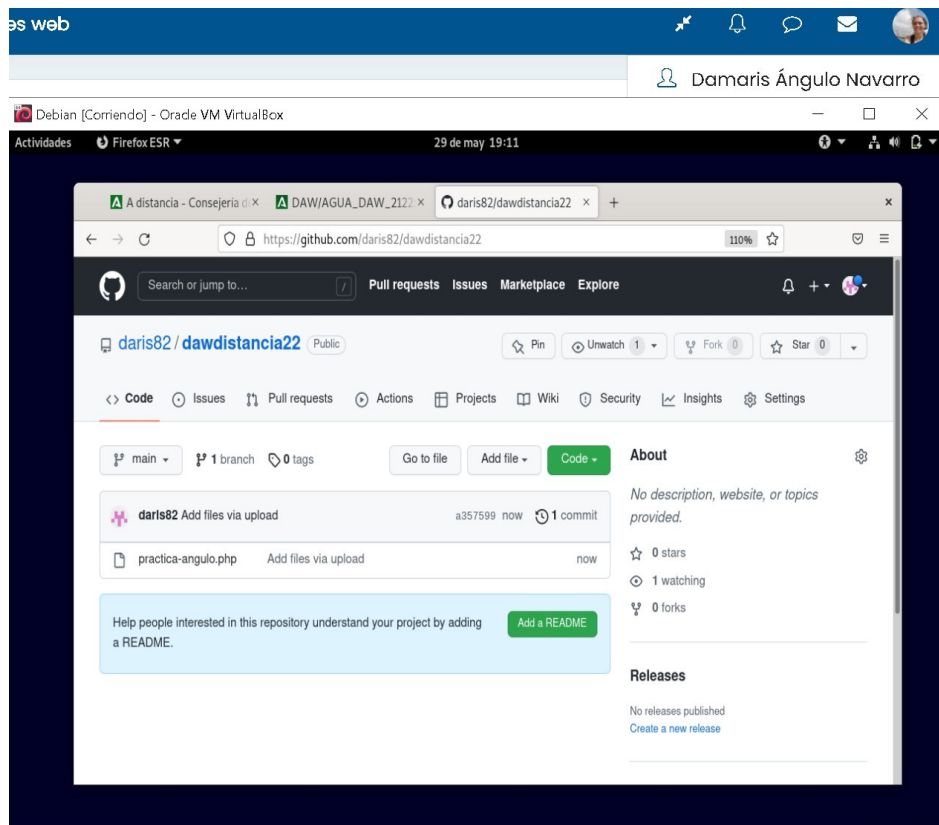
En esta actividad sincronizaremos nuestro repositorio local de Git con un repositorio remoto en Github.

Para ello crearemos un repositorio público en GITHUB, con el nombre 'dawdistancia22'. (Si no tienes cuenta en GitHub, deberás crear una).



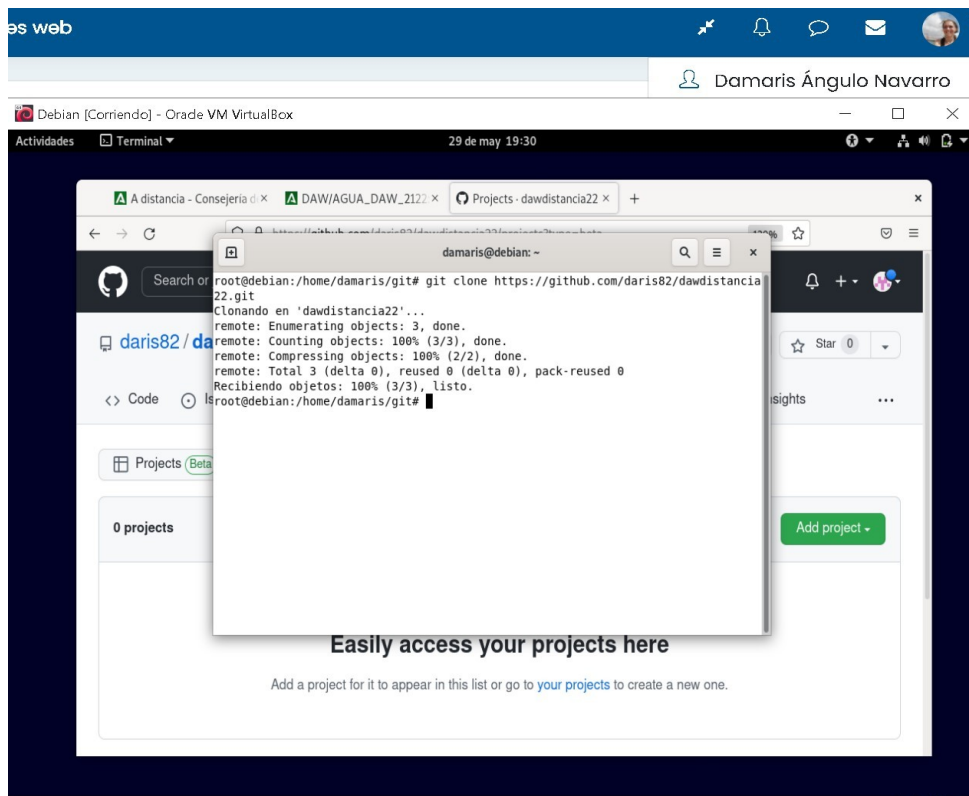
Carga desde la interfaz web de Github a tu repositorio el script PHP. Haz una captura del repositorio con el fichero subido.





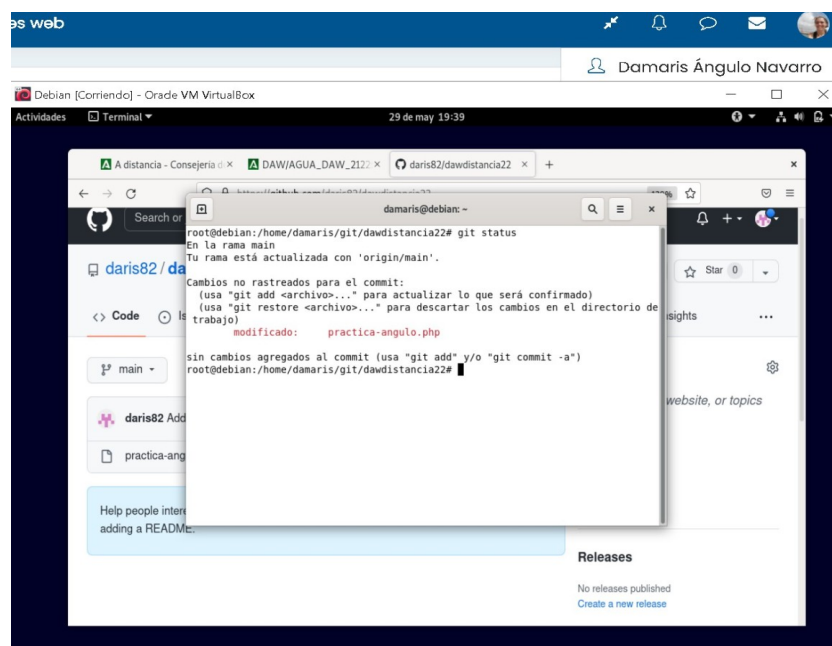
Ahora, desde tu máquina Linux, desde la terminal de comandos, clona el repositorio creado en tu repositorio local de Git y comprueba que está disponible en tu directorio de trabajo.

### *Clonamos el repositorio con git clone*

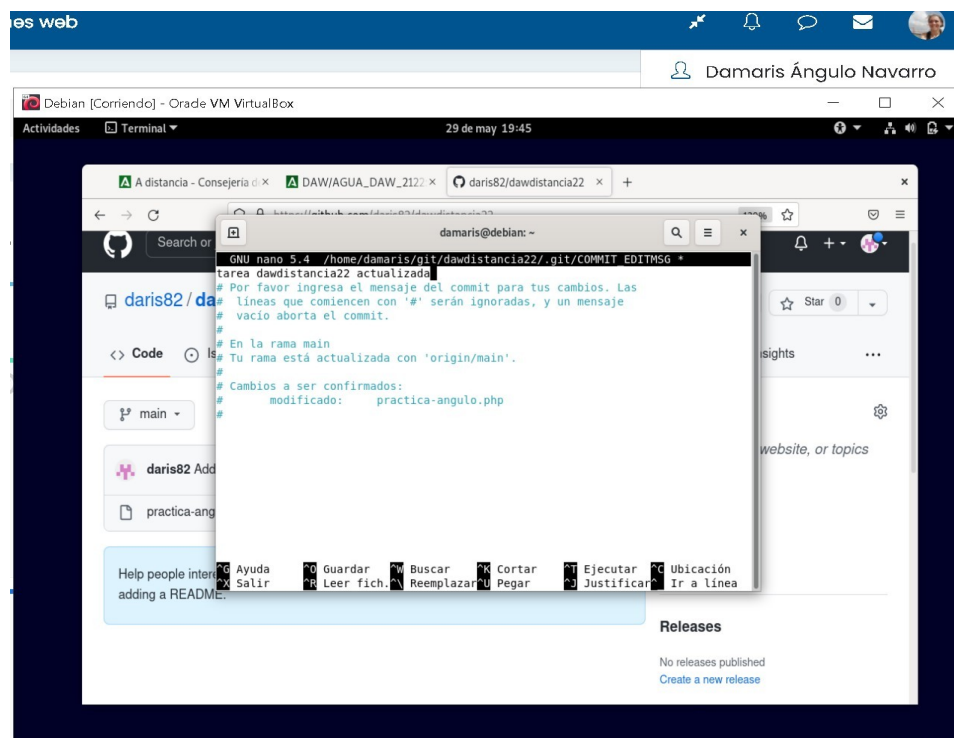


Ahora modifica el script y utiliza las herramientas disponibles en Git para detectar que se ha modificado y para actualizar el repositorio remoto en Github, documentando el commit con el mensaje "tarea dawdistancia22 actualizada".

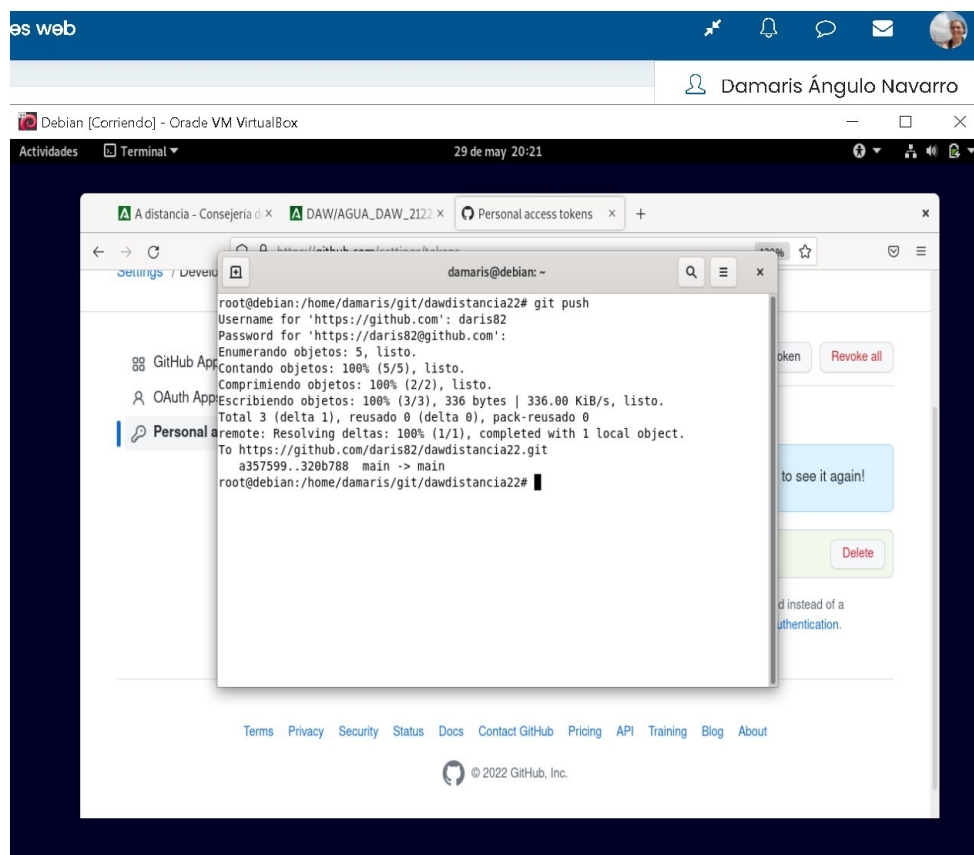
### *Modificamos el script en local y comprobamos con git status que tenemos una actualización:*



*Añadimos el archivo al repositorio con git add y git commit, con el nombre “tarea dawdistancia22 actualizada”*

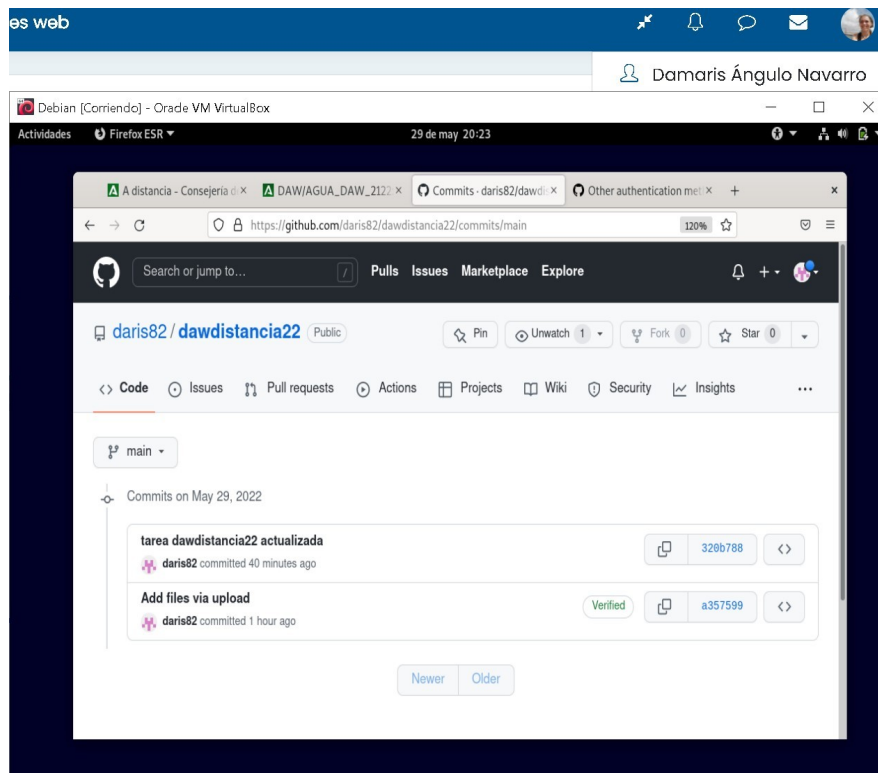


*Confirmamos con git push la actualización en el servidor*





*Desde github vemos la actualización que hemos realizado*



*Podemos ver los cambios realizados en el archivo:*

