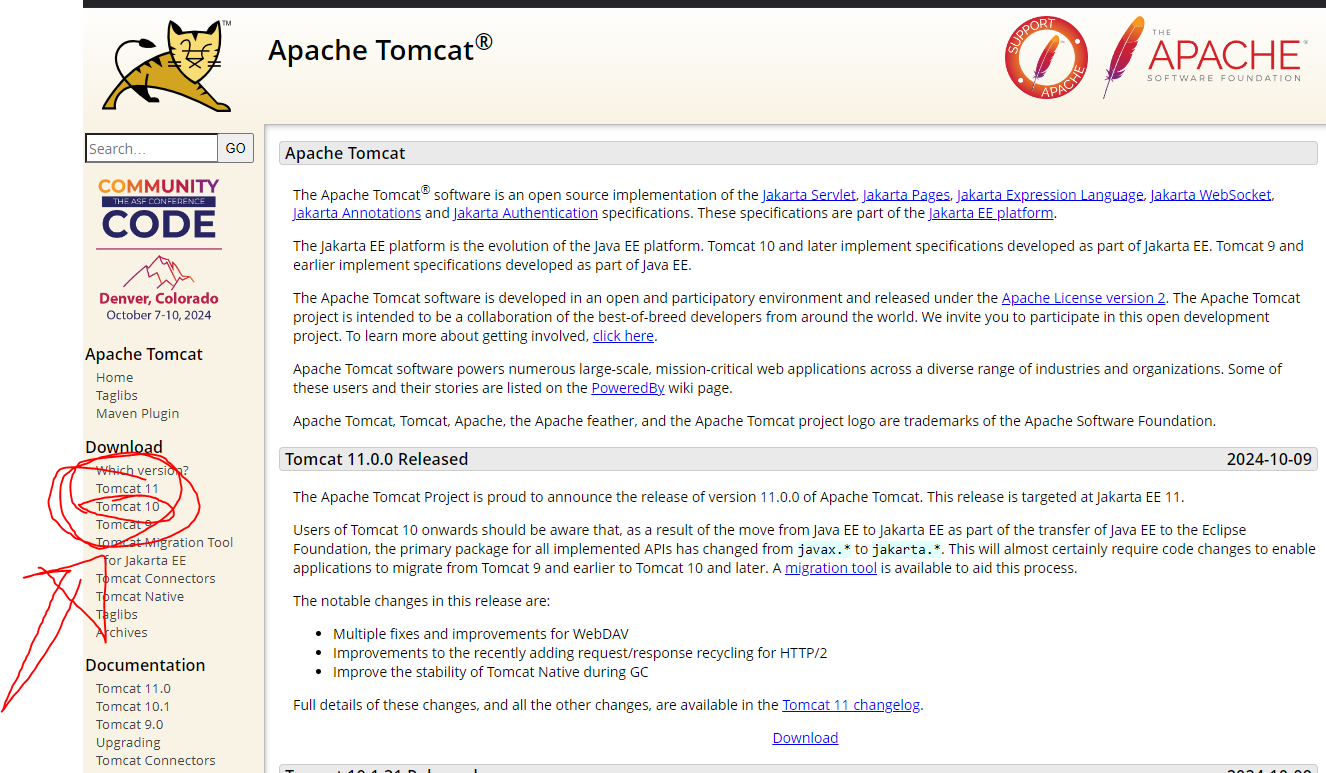
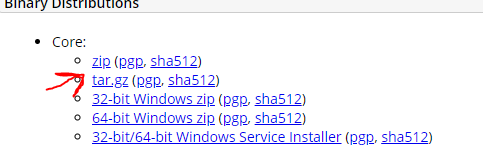
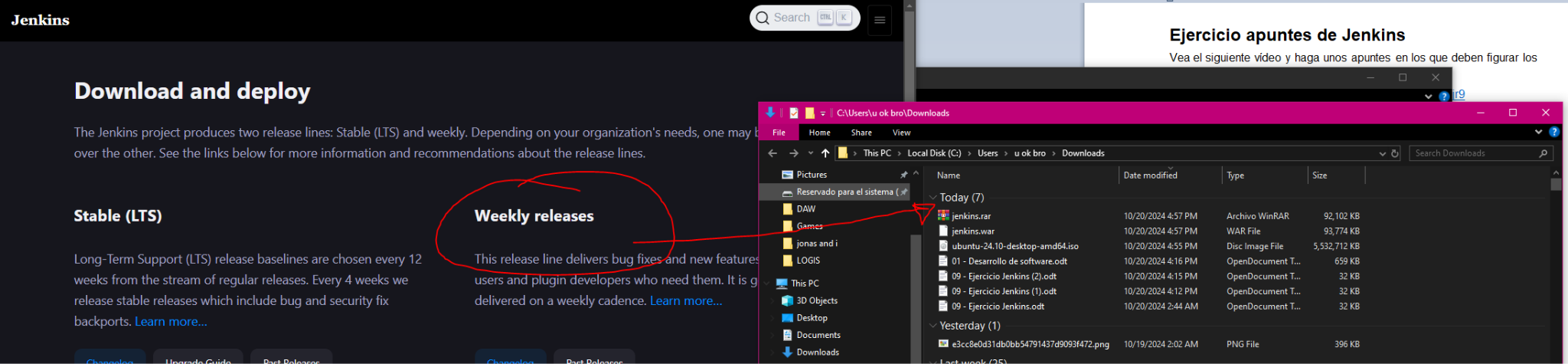
1. Es un programa que automatiza la tarea de construir y testear un proyecto. La integración continua es un proceso de desarrollo de software en el que los desarrolladores integran el nuevo código que han escrito con mayor frecuencia a lo largo del ciclo de desarrollo, añadiéndolo a la base de código al menos una vez al día.
2. Es una aplicación web orientada a LINUX aunque se puede ejecutar en otros programas. Jenkins es una herramienta de código abierto y gratuita que se utiliza para automatizar tareas repetitivas en el desarrollo de software, como la compilación, pruebas y despliegue de proyectos. Es un servidor de automatización extensible que se ejecuta en servlet containers como Apache Tomcat.
3. A través del comando en el terminal sudo apt install openjdk-23 o 22 o 17
4. Primero necesitamos apache tomcat (tomcat) nos vamos a la version 10

Ahí nos descargamos tar.gz

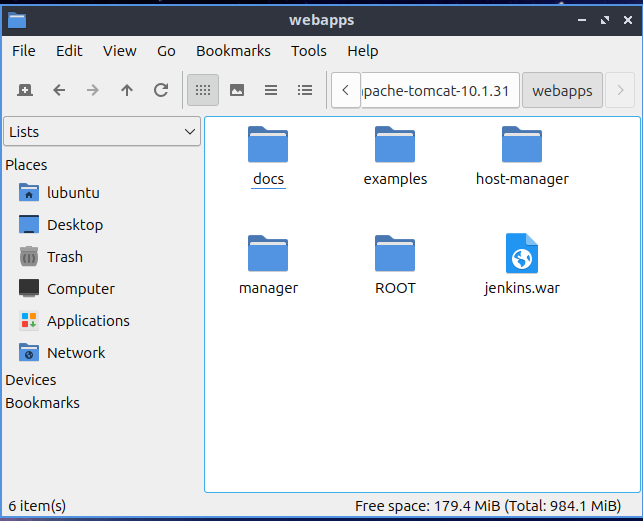
Cuando este descargado extraemos



Ahora nos vamos a descargar Jenkins para eso vamos a su sitio web y clickeamos en descargas semanales, (tenemos que tener descargado java en nuestro equipo sino tecleamos esto en el terminal: sudo apt install openjdk-23)



Cuando jenkins se ha terminado de descargar copiamos y localizamos a tomcat y pegamos en webapps

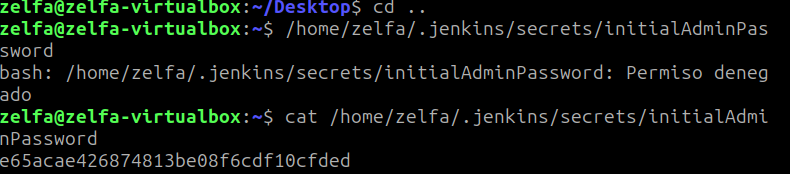


En el terminal vamos a la carpeta de tomcat apache y nos vamos a bin, donde introducimos ./catalina.sh

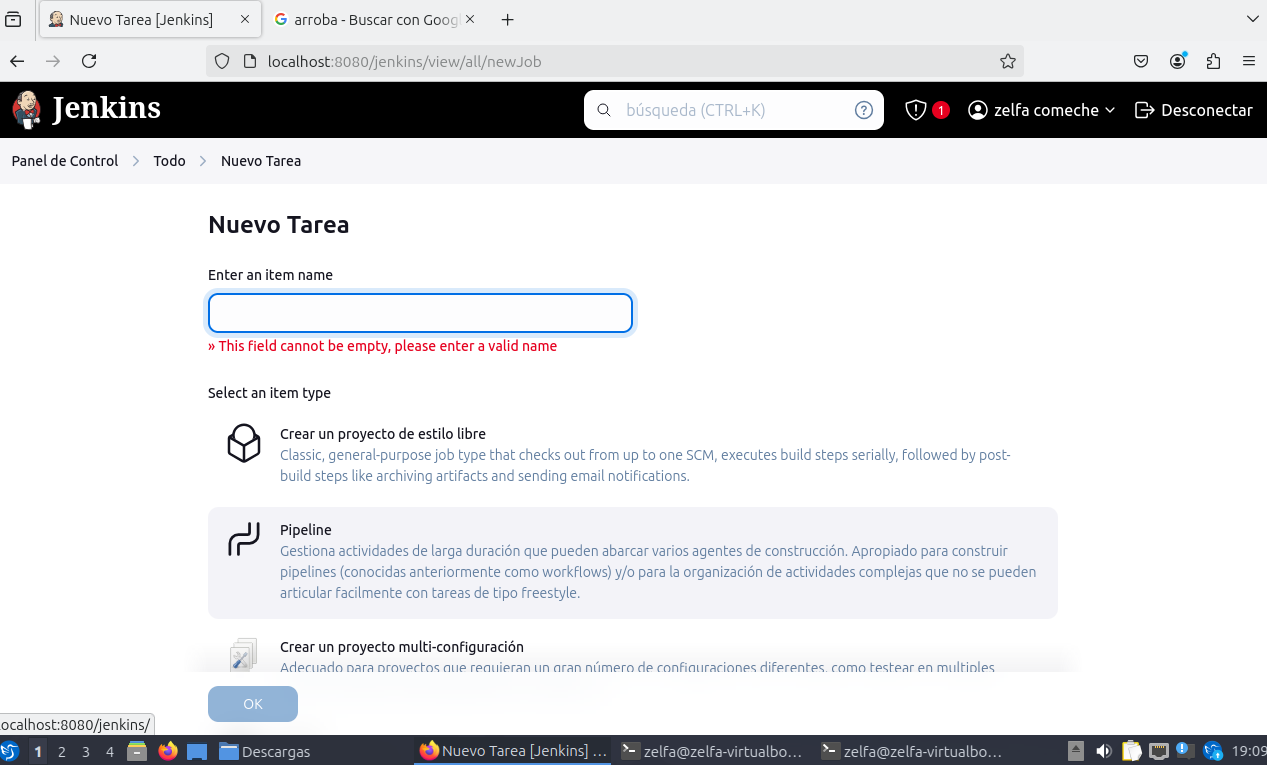


y hacemos catalina.sh run. Para ver si funciona tecleamos esto para ver si funciona localhost:8080/jenkins/

Vamos a la terminal para conseguir la contraseña haciendo un cat e instalamos

Después de esto nos registramos y hacemos una cuenta de jenkins.

¡Y ya estaría!



5. Se desinstala a través de los comandos rm -rf en el terminal

6. Creamos una nueva tarea y le asignamos un nombre, seleccionando crear proyecto de estilo libre. Podemos ponerle una descripción.

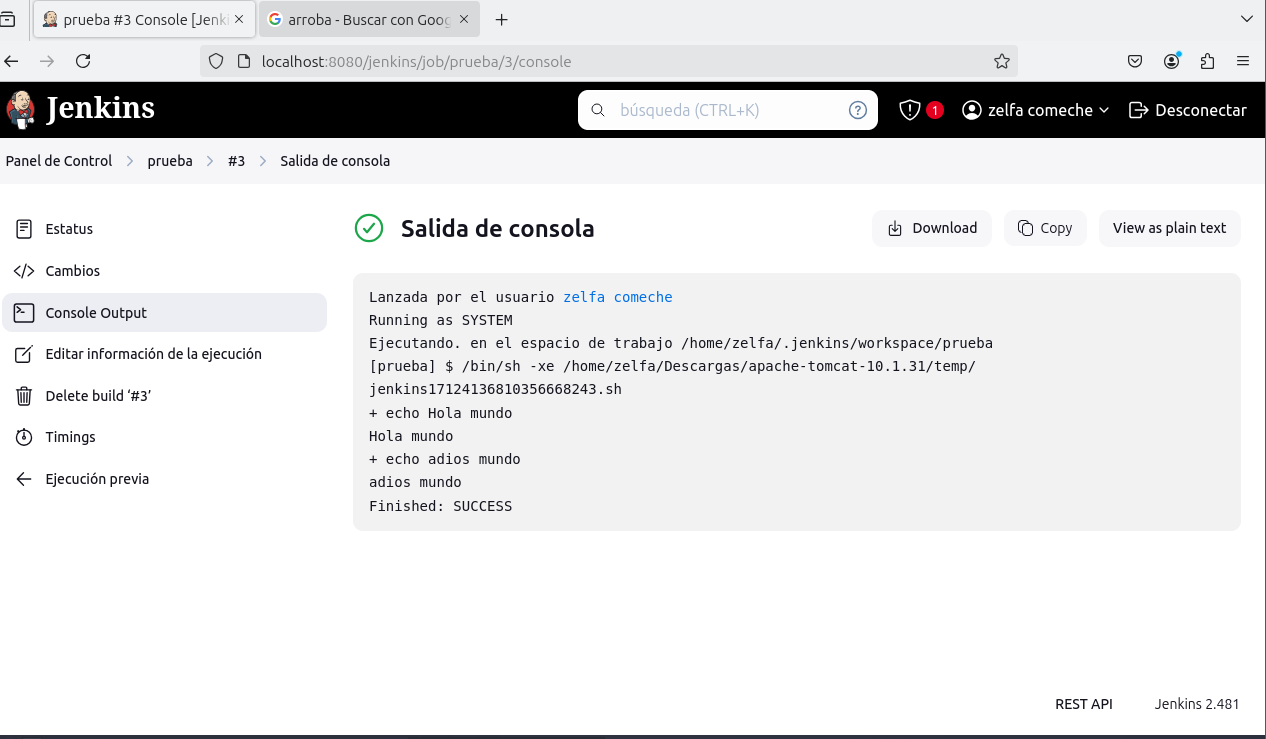
Seleccionamos si quiere asignarle un repositorio git, añadirle credenciales, token, etc.

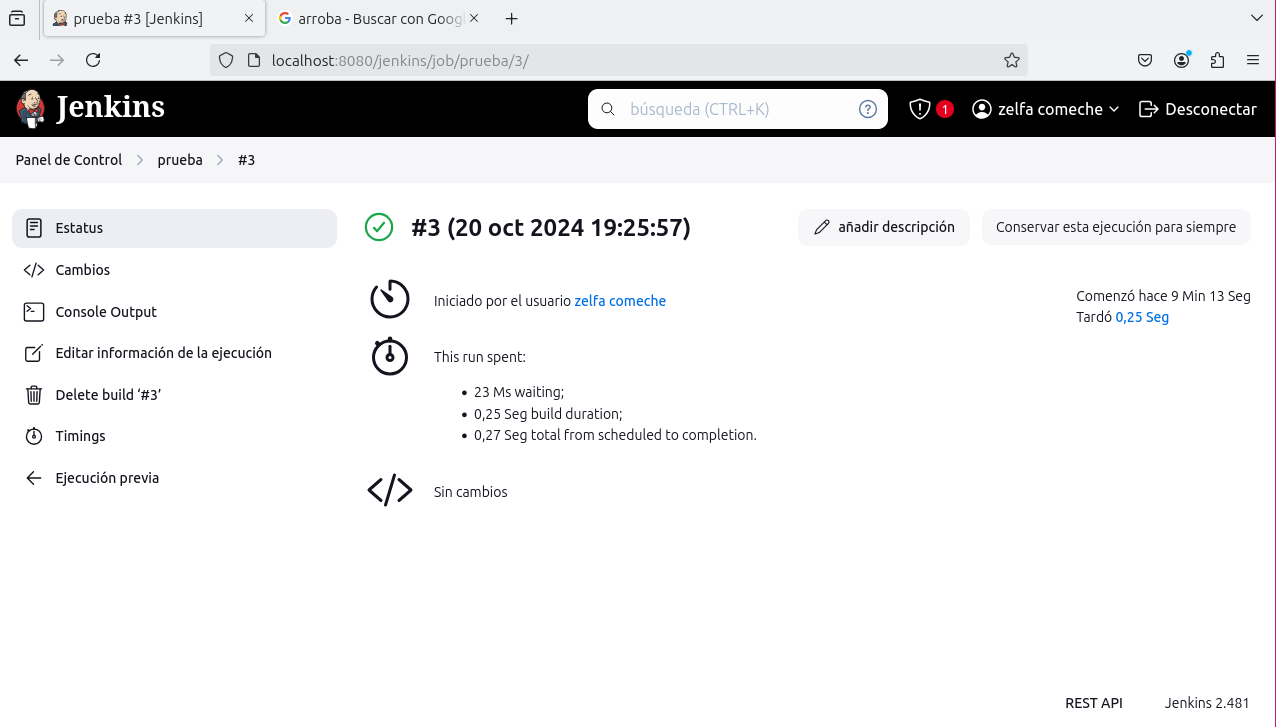
En disparadores de ejecuciones podemos seleccionar donde lo ejecutaremos, suele ser en la nube o esta diseñado para lanzarse a la nube. En la configuración de entornos de ejecución si activamos la opción de “Delete workspace before build starts” se pueden crear archivos durante la ejecución.

7. Los pasos de un pipeline son: descargar el proyecto, construir, tests, y deploy

8. Los build steps puedo añadir mis shell scripts. Podemos por ejemplo hacer un echo “Hola mundo” y podemos decidir que acciones hacer, como mandarme un correo electrónico, subir a git, etc.

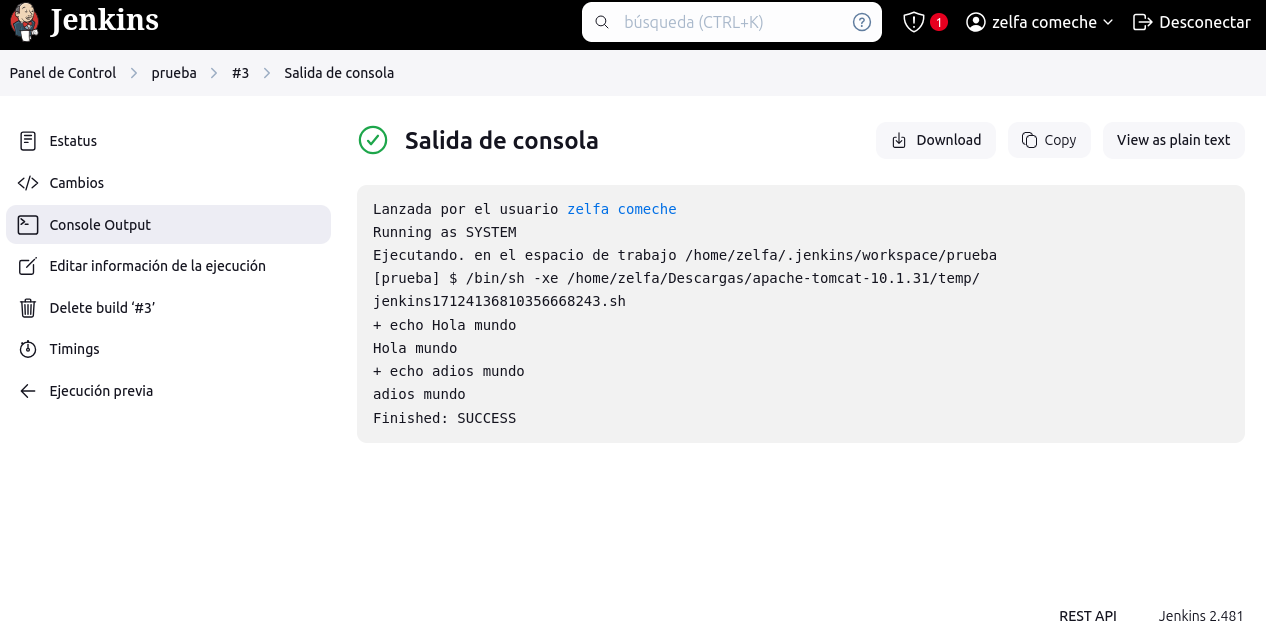
Después lo guardo y lo arranco, ahí puedo ver los errores, cuánto ha tardado. Es diferente clickear en el círculo verde que en la hora ya que el simbolo verde suelta la salida de la consola y la hora muestra cuanto tardo.



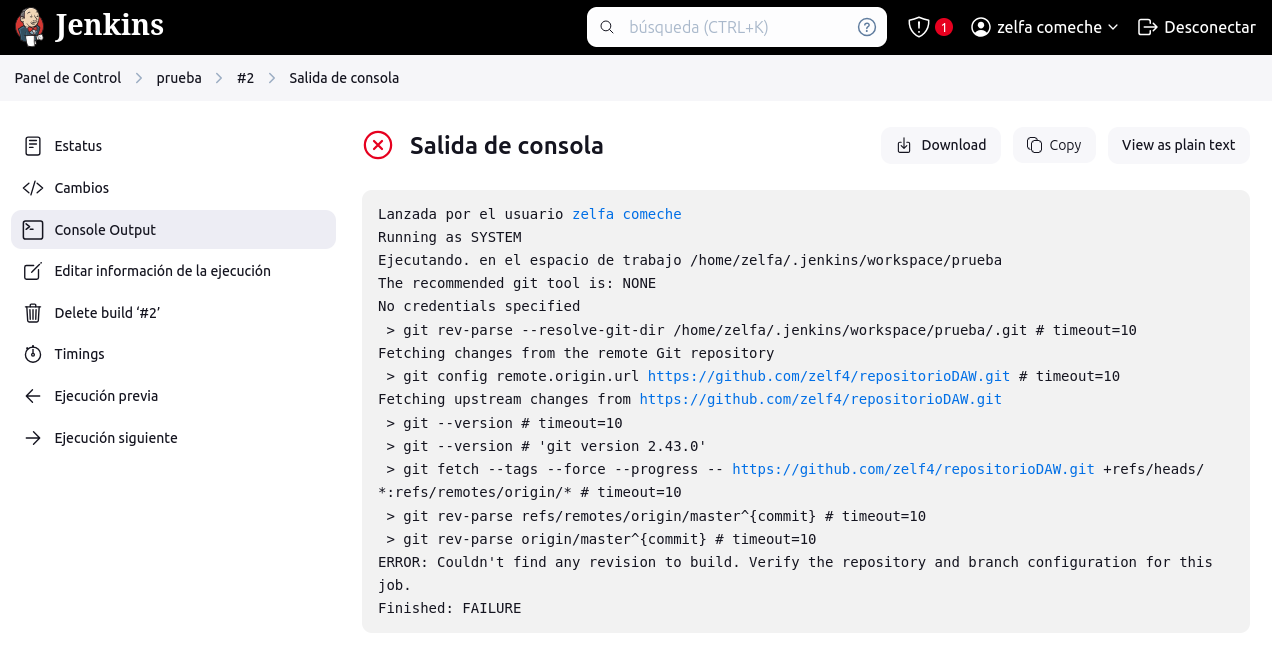


9. Podemos controlar los trabajos. En administra jenkins puego cargarle plugins puedo controlar las nubes, puedo añadirle un nodo (otro equipo en la nube) generar credenciales, nubes, etc

10. Correcta



Errónea



11. pwd para ir home, ls para ver los contenidos, cd proyecto y ./build.sh

12. Puede generar errores si el script tiene exit. Si el exit vale 1 genera un error si es 0 no genera.

13. Para descargar un archivo de internet se usa wget URL en el terminal

14. Con if no es = a 0 then tiene un error. Con un código

15. Si pongo un if (-e archivo) then;

echo El archivo existe

else

echo El archivo no existe