// INSTALAR Java

```
sudo apt update && sudo apt install -y default-jdk
echo $JAVA_HOME (probar si existe la variable, si no, hacer lo siguiente)
sudo nano /etc/environment
        JAVA_HOME=/usr/lib/jvm/default-java
source /etc/environment
echo $JAVA_HOME
```

// INSTALAR Tomcat10 ip:8080

```
sudo apt update && sudo apt install -y tar wget
wget
https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.33/bin/apache-tomcat-10.1.33.tar.gz  (borra el espacio entre el enlace y wget)
sudo useradd -U -m -d /opt/tomcat -k /dev/null -s /bin/false tomcat
sudo tar xf apache-tomcat-10.1.33.tar.gz -C /opt/tomcat/
sudo chown tomcat: /opt/tomcat/ -R
sudo  ln  -s  /opt/tomcat/apache-tomcat-10.1.33/  /opt/tomcat/apache-tomcat
sudo nano /etc/systemd/system/tomcat10.service
```

```
[Unit]
Description=Tomcat 10.1 servlet container para Debian 12 Bookworm
After=network.target

[Service]
Type=forking
User=tomcat
Group=tomcat
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom"
Environment="CATALINA_BASE=/opt/tomcat/apache-tomcat"
Environment="CATALINA_HOME=/opt/tomcat/apache-tomcat"
Environment="CATALINA_PID=/opt/tomcat/apache-tomcat/temp/tomcat.pid"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server
-XX:+UseParallelGC" (todo seguido no hay salto de linea)
ExecStart=/opt/tomcat/apache-tomcat/bin/startup.sh
ExecStop=/opt/tomcat/apache-tomcat/bin/shutdown.sh
```

```
[Install]
WantedBy=multi-user.target
```

———

```
sudo systemctl start tomcat10 -> status -> enable
```

CONFIGURACIÓN

```
sudo nano /opt/tomcat/apache-tomcat/webapps/docs/META-INF/context.xml
&&     /opt/tomcat/apache-tomcat/webapps/manager/META-INF/context.xml
&&     /opt/tomcat/apache-tomcat/webapps/host-manager/META-INF/context.xml
<!--     <Valve className="org.apache.catalina.valves.RemoteAddrValve"
       allow="127\.\d+\.\d+\.\d+ | : :1 | 0:0:0:0:0:0:0:1" />         -->
```

CREAR USUARIO

```
sudo nano /opt/tomcat/apache-tomcat/conf/tomcat-users.xml
   <user username="usuario" password="12…78" roles="admin-gui,manager-gui"/>
</tomcat-users>
```

Desplegar war desde ip:8080/manager/html

// INSTALAR MAVEN

```
sudo apt update && sudo apt install maven && mvn --v
sudo nano /opt/tomcat/apache-tomcat/conf/tomcat-users.xml
```

Añadir roles: “admin, admin-gui, manager, manager-gui, manager-script”

```
sudo nano /etc/maven/settings.xml
        <server>
                <id>TomcatExamenWSS</id>
                <username>usuario</username>
                <password>12345678</password>
        </server>
mvn archetype:generate -DgroupId=Wss -DartifactId=war-WSS
-DarchetypeArtifactId=maven-archetype-webapp   -DinteractiveMode=false
```

Editar el pom.xml del proyecto

```
<build>
        <finalName>war-WSS</finalName>
        <plugins>
                <plugin>
```

```
            <groupId>org.apache.tomcat.maven</groupId>
            <artifactId>tomcat7-maven-plugin</artifactId>
            <version>2.2</version>
            <configuration>
                <url>http://localhost:8080/manager/text</url>
                <server>TomcatExamenWSS</server>
                <path>/nombreApp</path>
            </configuration>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <configuration>
                <source>1.8</source>
                <target>1.8</target>
            </configuration>
        </plugin>
        <plugin>
            <artifactId>maven-war-plugin</artifactId>
            <version>3.2.2</version>
        </plugin>
    </plugins>
</build>
```

Antes de hacer mvn usa cd para ir al carpeta donde se encuentra

```
mvn tomcat7:deploy || mvn tomcat7:redeploy || mvn tomcat7:undeploy
```

// INSTALAR NODE ip:3000

```
sudo apt update && sudo apt upgrade
```

(no sé si se hace) curl -sL https://deb.nodesource.com/setup_16.x | sudo -E bash -

```
sudo apt install nodejs npm -y && node --version && npm --version
```

SI EL PROYECTO NO EXISTE

```
mkdir App && cd App && npm init && npm install express
```

Sino te funciona despues de escribir npm install , tiene que poner npm audit fix

SI EL PROYECTO EXISTE

git clone link && cd al repositorio

```
npm install && npm install nodemon --save-dev && npm run start
```

SIN CLÚSTER

```
mkdir app && cd app && npm init
sudo nano package.json
sudo npm install express
sudo npm install-g loadtest pm2
sudo nano index.js
        const express = require("express");
        const app = express();
        const port = 3000;
        app.get("/", (req, res) => {res.send("Hello World!"); } );
        app.get("/api/:n", function (req, res) {
                let n = parseInt(req.params.n);
                let count = 0;
                if (n > 5000000000) n = 5000000000;
                for (let i = 0; i <= n; i++) { count += i }
                res.send( `Final count is ${count}` ); } );
        app.listen(port, () => { console.log( `App listening on port ${port}` ); } );
```

```
node index.js && ip:3000/carga
```

Para matar el proceso Ctrl + C

CON CLÚSTER

```
cp -r /app /app2 && sudo nano primary.js
        import cluster from "cluster";
        import os from "os";
        import { dirname } from "path";
        import { fileURLToPath } from "url";
        const__dirname = dirname( fileURLToPath ( import.meta.url ) );
        const cpuCount = os.cpus().length;
        console.log( `El número total de CPUs es ${cpuCount}` );
        console.log( `Primario pid=${process.pid}` );
        cluster.setupPrimary( {
                exec:__dirname + "/index.js",
        } );
        for (let i = 0; i < cpuCount; i++) { cluster.fork(); }
        cluster.on("exit", (worker, code, signal) => {
                console.log(`El worker ${worker.process.pid} ha sido aniquilado`);
                console.log("Iniciando otro worker");
                cluster.fork();
        } );
```

ip:3000/carga

```
cd /app && node index.js
loadtest http://localhost:3000/api/500000 -n 1000 -c 100
```

```
cd /app && node primary.js
loadtest http://localhost:3000/api/500000 -n 1000 -c 100
```

PM2

```
cd /app
pm2 start index.js -i 0 && pm2 stop index.js
```

```
pm2 ecosystem
sudo nano ecosystem.config.js
        module.exports = {
                apps: [
                        {
                        name: "nombre_aplicacion",
                        script: "nombre_aplicacion_sin_cluster.js",
                        instances: 0,
                        exec_mode: "cluster",
                        }
                ]
        };
```

O borrar el archivo anterior y directamente crear uno ecosystem.config.cjs

```
mv  ecosystem.config.js  ecosystem.config.cjs
```

```
pm2 start ecosystem.config.cjs && pm2 stop ecosystem.config.cjs
```