



ÍNDICE

BLOQUE 1: Preparando el entorno.....	1
1. General	1
2. Base de Datos	3
1. Conector	3
2. Creación Usuario y Base de Datos en MariaDB.....	5
3. Conexión en el context.xml	7
4. Comprobación funcionamiento	8

BLOQUE 1: Preparando el entorno

1. General

Partiremos de la tarea 2 de Miguel Ángel, lo que implica que tendremos Apache (apache2) corriendo por el puerto 5000, Tomcat (tomcat9) por el puerto 7000, OpenSSH (sshd) por el puerto 7777 y MariaDB (mysqld) por el puerto 8000:

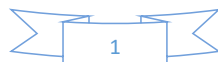
Arrancamos los respectivos servicios, y comprobamos que efectivamente están funcionando por esos puertos:

```
miguelang611@miguelang611-VMWare:~$ sudo systemctl start apache2
miguelang611@miguelang611-VMWare:~$ sudo systemctl start tomcat9
miguelang611@miguelang611-VMWare:~$ sudo systemctl start sshd
miguelang611@miguelang611-VMWare:~$ sudo systemctl start mariadb
miguelang611@miguelang611-VMWare:~$ sudo netstat -tulnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:53:53        0.0.0.0:*               LISTEN      714/systemd-resolve
tcp        0      0 127.0.0.1:631         0.0.0.0:*               LISTEN      26922/cupsd
tcp        0      0 127.0.0.1:8000        0.0.0.0:*               LISTEN      30961/mysqld
tcp        0      0 0.0.0.0:7777          0.0.0.0:*               LISTEN      30894/sshd: /usr/sb
tcp        0      0 :::5000                :::*                    LISTEN      30754/apache2
tcp        0      0 :::1:631               :::*                    LISTEN      26922/cupsd
tcp        0      0 :::7000                :::*                    LISTEN      30834/java
tcp        0      0 :::7777                :::*                    LISTEN      30894/sshd: /usr/sb
udp        0      0 127.0.0.1:53:53        0.0.0.0:*               714/systemd-resolve
udp        0      0 0.0.0.0:631           0.0.0.0:*               26923/cups-browsed
udp        0      0 0.0.0.0:5353          0.0.0.0:*               743/avahi-daemon: r
udp        0      0 0.0.0.0:38279         0.0.0.0:*               743/avahi-daemon: r
udp6       0      0 :::36476               :::*                    743/avahi-daemon: r
udp6       0      0 :::5353                :::*                    743/avahi-daemon: r
```

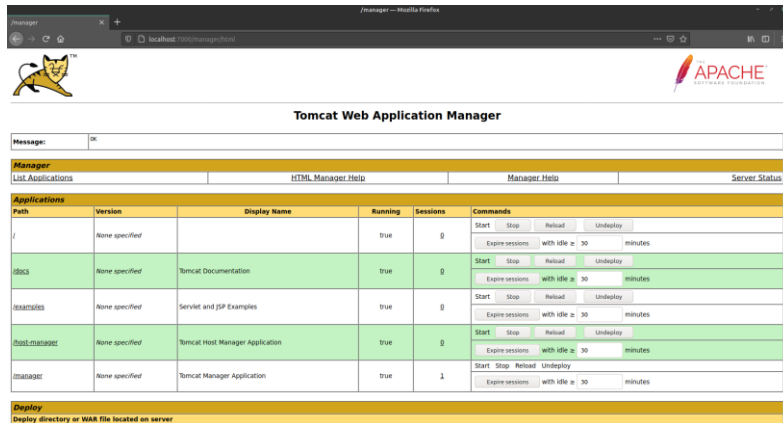
Para comenzar, tenemos que desplegar el archivo .war proporcionado. Para ello nos dirigimos a <http://localhost:7000>, es decir, a la web de Tomcat:

Accedemos al menú de administración web: <http://localhost:7000/manager/html>

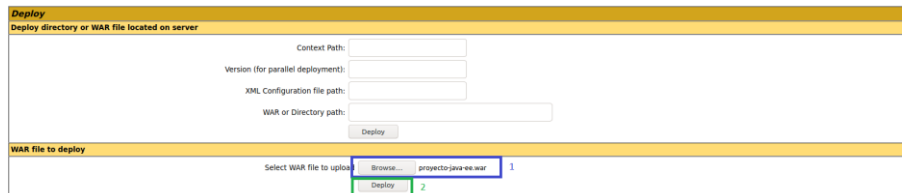
Nos identificamos con las credenciales correspondientes: admin/test:



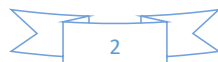
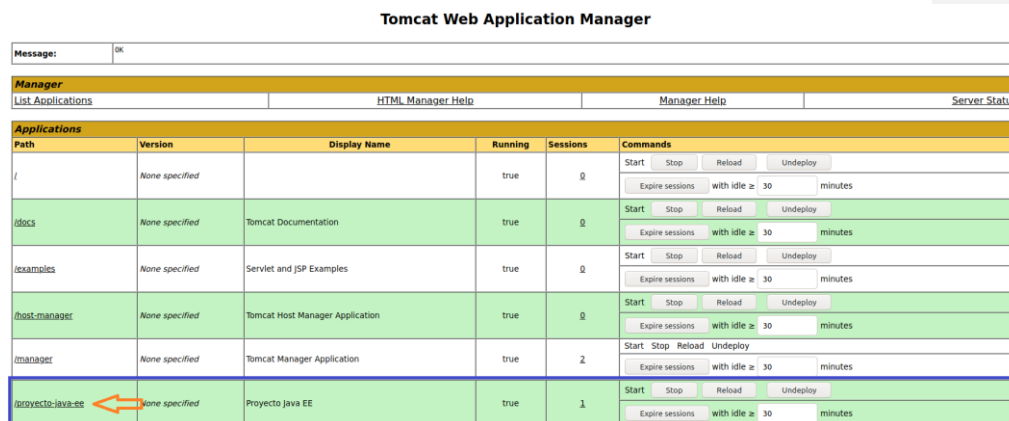
Se nos da la bienvenida al gestor de aplicaciones de Tomcat:



Nos dirigimos al menú de deploy, y dentro de la sección WAR, escogemos el fichero del proyecto, y hacemos click en deploy.



Hecho esto, nos dirigimos de nuevo a la interfaz del manager, observando que el proyecto ha sido cargado. Haremos click sobre /projecto-java-ee, en la columna path, para dirigirnos a la página principal del proyecto:



Se nos da la bienvenida:



2. Base de Datos

En este momento, vamos a comenzar con la Base de Datos. Si hacemos click en probar, como cabría esperar, no hace nada, ya que no hemos configurado el recurso y conecto JDBC.



Vamos a tratar de poner remedio a esto...

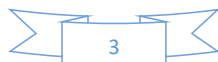
1. Conector

Para ello, vamos a descargar inicialmente el conector JDBC de MariaDB:



Escogeremos la versión estable 2.7.2, en su versión .jar

La guardaremos provisionalmente en el Escritorio.



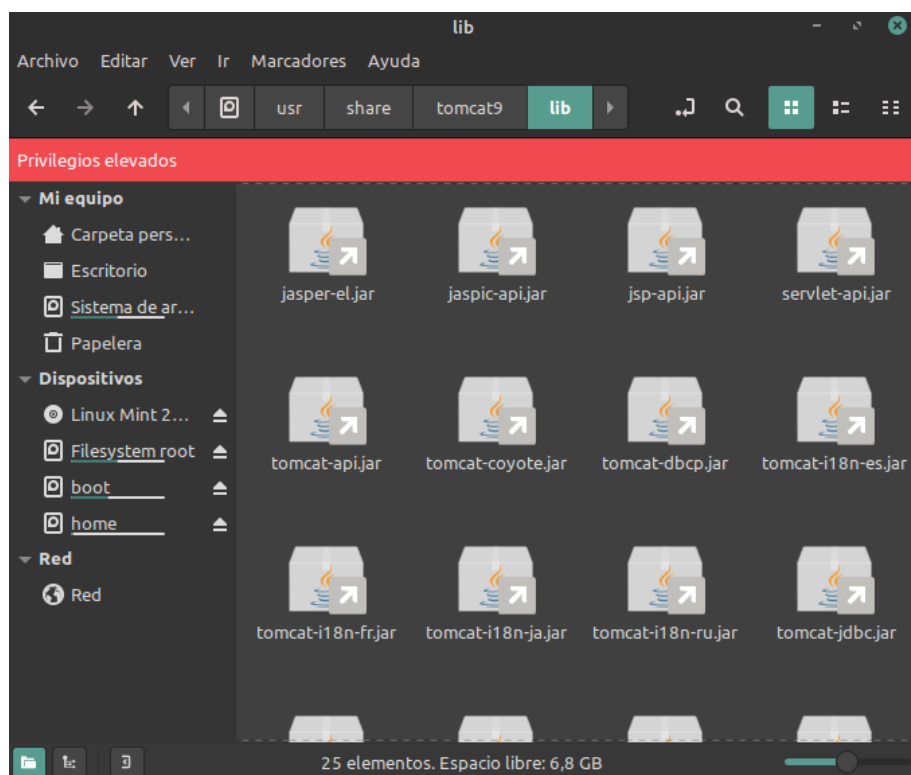
A continuación, vamos a colocar dicha librería en el directorio correspondiente de Tomcat. Puesto que la idea es mantener la carpeta del proyecto intacta, vamos a colocar la librería en el lugar “principal” de las librerías de Tomcat:

```
/usr/share/tomcat9/lib
```

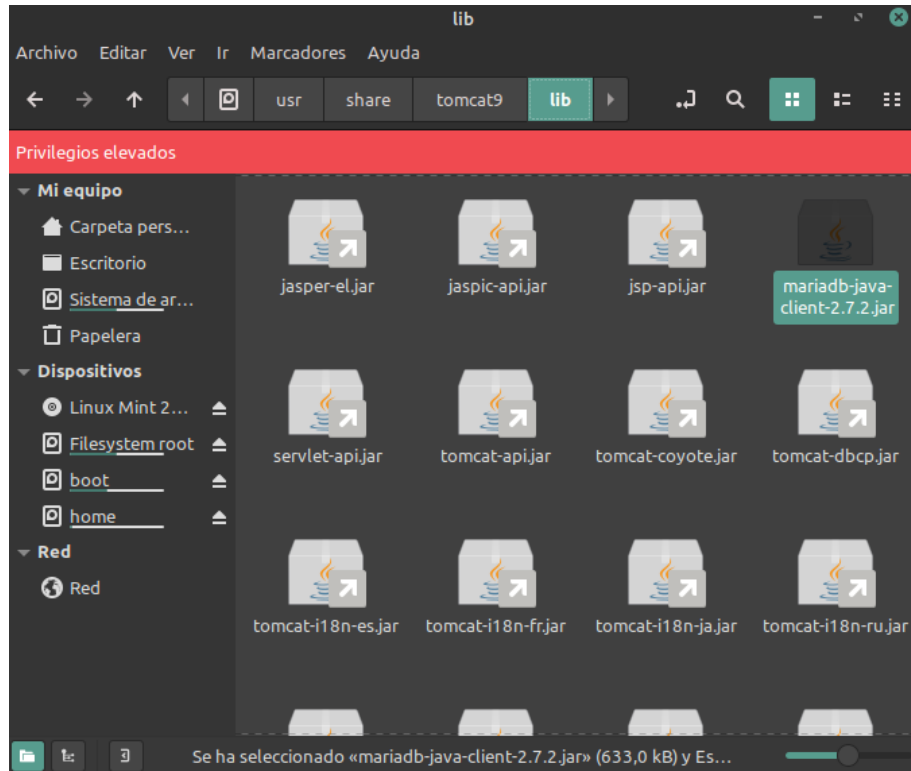
Para ello, ejecutamos desde la Terminal el comando

```
sudo nemo /usr/share/tomcat9/lib
```

Lo cual nos dirigirá al explorador de archivos y al directorio correspondiente, con permisos de superusuario:



Copiamos desde el escritorio el conector:



2. Creación Usuario y Base de Datos en MariaDB

Hecho esto, procederemos a crear el usuario, contraseña, base de datos y tabla desde MariaDB:

Crear BBDD → `CREATE DATABASE MiBaseDatos;`

Crear usuario y darle una contraseña → `CREATE USER 'actividad3'@'localhost' IDENTIFIED BY 'admin';`

Otorgarle privilegios → `grant all privileges on MiBaseDatos.* to 'actividad3'@'localhost';`

Limpiar caché → `flush privileges;`

Seleccionar BBDD → `use MiBaseDatos`

Crear tabla →

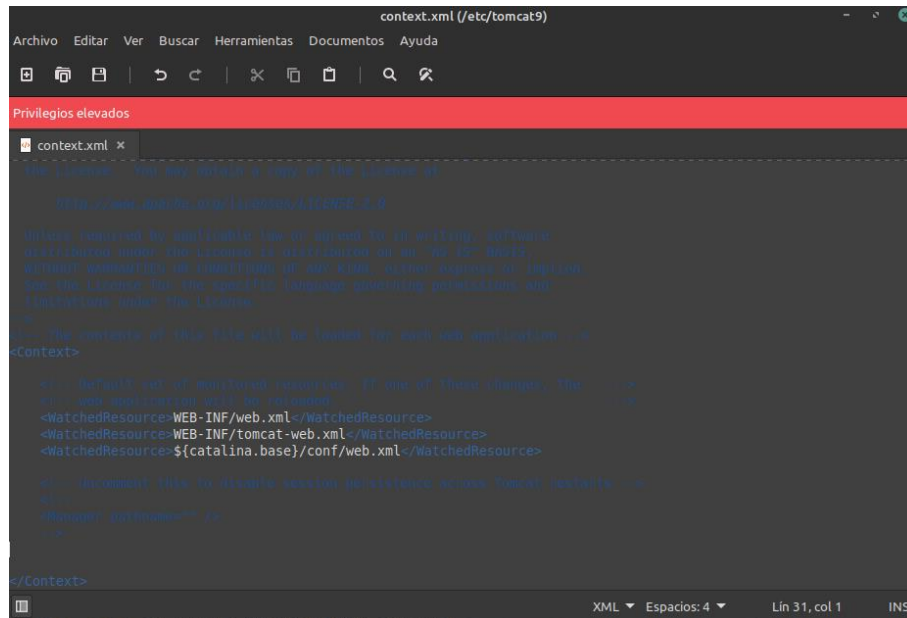
```
CREATE TABLE PERSONAS (  
  
    DNI VARCHAR(8) DEFAULT '00000000' NOT NULL,  
  
    NOMBRE VARCHAR(20),  
  
    APELLIDOS VARCHAR(20),  
  
    PRIMARY KEY (dni)  
  
);
```

```
miguelang611@miguelang611-VMWare:~$ sudo mariadb  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 50  
Server version: 10.3.25-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
MariaDB [(none)]> CREATE DATABASE MiBaseDatos;  
Query OK, 1 row affected (0.001 sec)  
  
MariaDB [(none)]> CREATE USER 'actividad3'@'localhost' IDENTIFIED BY 'admin';  
Query OK, 0 rows affected (0.003 sec)  
  
MariaDB [(none)]> grant all privileges on MiBaseDatos.* to 'actividad3'@'localhost';  
Query OK, 0 rows affected (0.001 sec)  
  
MariaDB [(none)]> flush privileges;  
Query OK, 0 rows affected (0.001 sec)  
  
MariaDB [(none)]> use MiBaseDatos  
Database changed  
MariaDB [MiBaseDatos]> CREATE TABLE PERSONAS (  
    -> DNI VARCHAR(8) DEFAULT '00000000' NOT NULL,  
    -> NOMBRE VARCHAR(20),  
    -> APELLIDOS VARCHAR(20),  
    -> PRIMARY KEY (dni)  
    -> );  
Query OK, 0 rows affected (0.007 sec)  
  
MariaDB [MiBaseDatos]> exit  
Bye
```

3. Conexión en el context.xml

Ahora nos toca incluir el recurso correspondiente a la conexión a la base de datos en el context.xml de Tomcat, que se encuentra localizado en /etc/tomcat9/context.xml :

```
sudo xed /etc/tomcat9/context.xml
```



```
context.xml (/etc/tomcat9)
Archivo Editar Ver Buscar Herramientas Documentos Ayuda
Privilegios elevados
context.xml x
The contents of this file will be loaded for each web application -->
<Context>
  <!-- Default set of watched resources. If one of these changes, the
  <!-- web application will be reloaded. -->
  <WatchedResource>WEB-INF/web.xml</WatchedResource>
  <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
  <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>

  <!-- document this to simplify version persistence across Tomcat installs -->
  <!--
  <Resource name="" type=""
  -->
</Context>
XML Espacios: 4 Lín 31, col 1 INS
```

Deberemos añadir las siguientes líneas:

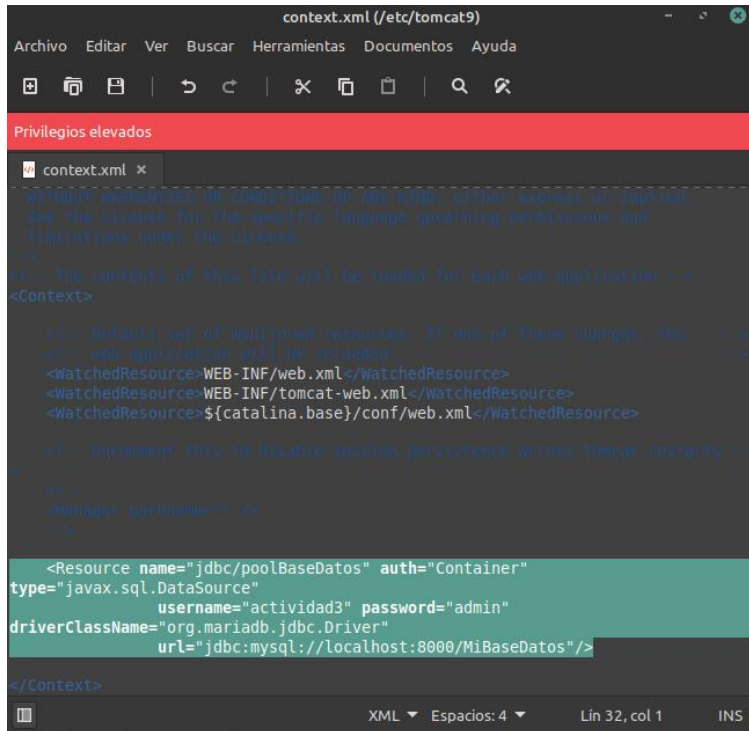
```
<Resource          name="jdbc/poolBaseDatos"          auth="Container"
type="javax.sql.DataSource"

                username="actividad3"                password="admin"
driverClassName="org.mariadb.jdbc.Driver"

                url="jdbc:mysql://localhost:8000/MiBaseDatos"/>
```

Indicamos pues el nombre de usuario y contraseña que creamos antes, indicamos el nombre de la clase del driver: org.mariadb.jdbc.Driver ; y en la url indicaremos jdbc:mysql://localhost:8000/MiBaseDatos en referencia a que está en local, en el puerto 8000, y que el nombre de la misma es MiBaseDatos.

La tabla se leerá desde la aplicación Java, y esa es la que se denomina Personas.



```
context.xml (/etc/tomcat9)
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda

Privilegios elevados

context.xml x
<!-- WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either expressed or implied.
See the license for the specific language governing permissions and
limitations under the license.
-->
<!-- The contents of this file will be loaded for each web application -->
<Context>

    <!-- Default set of monitored resources. If one of these changes, the
    web application will be restarted. -->
    <WatchedResource>WEB-INF/web.xml</WatchedResource>
    <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
    <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>

    <!-- Uncomment this to disable session persistence across Tomcat restarts -->
    <!--
    <Manager pathname="" />
    -->

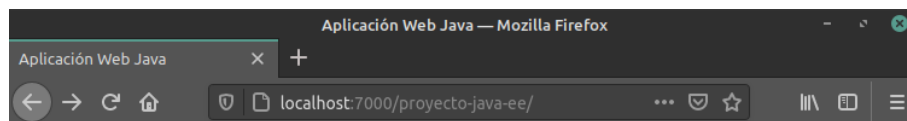
    <Resource name="jdbc/poolBaseDatos" auth="Container"
type="javax.sql.DataSource"
        username="actividad3" password="admin"
driverClassName="org.mariadb.jdbc.Driver"
        url="jdbc:mysql://localhost:8000/MiBaseDatos"/>
</Context>

XML  Espacios: 4  Lín 32, col 1  INS
```

4. Comprobación funcionamiento

Procedemos a parar e iniciar el servicio de MariaDB y de Tomcat.

Nos dirigimos a la página principal del proyecto:



Bienvenido a la aplicación Web Java de ejemplo para JNDI, Apache y SQL

Probar BD

Probar CDN

Probar FTP

1. Prueba de BBDD

Hacemos click en probar BD:

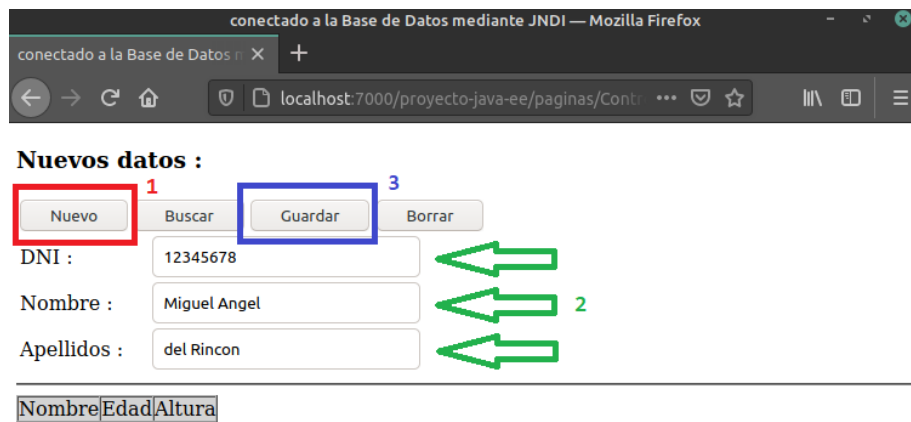


The screenshot shows a Mozilla Firefox browser window with the title "conectado a la Base de Datos mediante JNDI". The address bar shows "localhost:7000/proyecto-java-ee/paginas/vista". The page contains a form with four buttons: "Nuevo", "Buscar", "Guardar", and "Borrar". Below the buttons are three input fields labeled "DNI :", "Nombre :", and "Apellidos :". At the bottom of the form, there is a table header with three columns: "Nombre", "Edad", and "Altura".

Comprobamos que la conexión es exitosa.

2. Inserción y guardado

Vamos a probar a insertar datos, haciendo click en Nuevo, introduciendo los datos, y después haciendo click en guardar:



The screenshot shows the same web browser window as before, but now the form is filled with data. The "Nuevo" button is highlighted with a red box and a red number "1". The "Guardar" button is highlighted with a blue box and a blue number "3". The input fields contain "DNI : 12345678", "Nombre : Miguel Angel", and "Apellidos : del Rincon". Three green arrows, labeled with a green number "2", point to the input fields. At the bottom of the form, the table header "NombreEdadAltura" is visible.

El resultado es el siguiente:

conectado a la Base de Datos mediante JNDI — Mozilla Firefox

conectado a la Base de Datos

localhost:7000/proyecto-java-ee/paginas/Contr

persona guardada!!

Nuevo Buscar Guardar Borrar

DNI : 12345678

Nombre : Miguel Angel

Apellidos : del Rincon

Nombre	Edad	Altura
12345678	Miguel Angel	del Rincon

Procedemos análogamente con el nombre de todos los integrantes, siendo este el resultado:

conectado a la Base de Datos mediante JNDI — Mozilla Firefox

conectado a la Base de Datos

localhost:7000/proyecto-java-ee/paginas/Contr

persona guardada!!

Nuevo Buscar Guardar Borrar

DNI : 87654321

Nombre : Sergio

Apellidos : Gutierrez

Nombre	Edad	Altura
12345678	Miguel Angel	del Rincon
22333444	Pamela	Carrasco
87654321	Sergio	Gutierrez

3. Búsqueda

Ahora, podemos introducir un DNI (clave primaria), y hacemos click en buscar:

Nuevo **Buscar 2** Guardar Borrar

DNI : 22333444 **1**

Nombre :

Apellidos :

Nombre	Edad	Altura
12345678	Miguel Ángel	del Rincon
22333444	Pamela	Carrasco
87654321	Sergio	Gutierrez

Nos devuelve un aviso de persona encontrada, y la ficha de la persona:

persona encontrada!!



Nuevo Buscar Guardar Borrar

DNI : 22333444

Nombre : Pamela

Apellidos : Carrasco

Nombre	Edad	Altura
12345678	Miguel Ángel	del Rincon
22333444	Pamela	Carrasco
87654321	Sergio	Gutierrez

4. Borrado

Si hacemos click sobre borrar, se nos eliminará este registro de la BBDD, de lo que se nos avisa, y además se refresca la lista de datos, desapareciendo Pamela:

persona borrada!!

Nuevo	Buscar	Guardar	Borrar
DNI :	<input type="text"/>		
Nombre :	<input type="text"/>		
Apellidos :	<input type="text"/>		

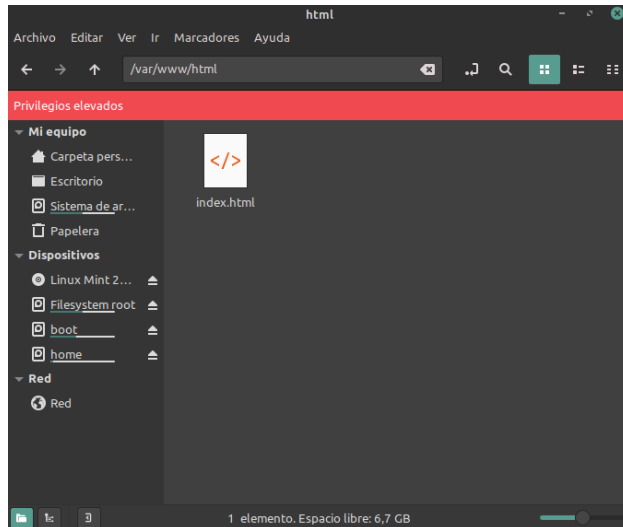
Nombre	Edad	Altura
12345678	Miguel Angel	del Rincon
87654321	Sergio	Gutierrez

Damos así pues por finalizadas las pruebas de la Base de Datos, siendo satisfactorio su funcionamiento.

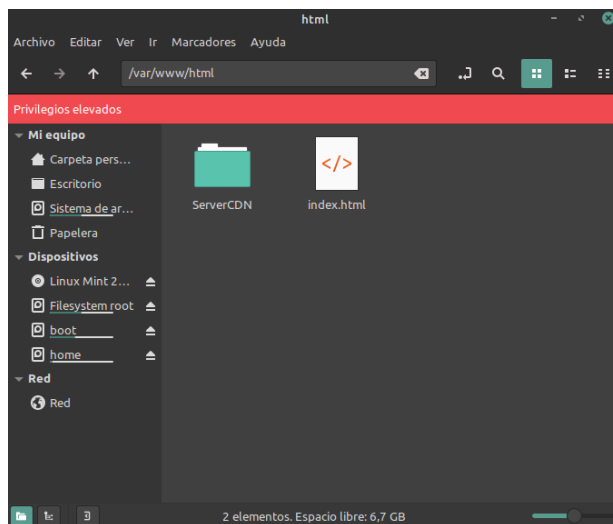
3. CDN

Para hacer funcionar el servicio CDN lo que tenemos que hacer es crear una carpeta dentro del servidor Apache donde localizaremos los archivos:

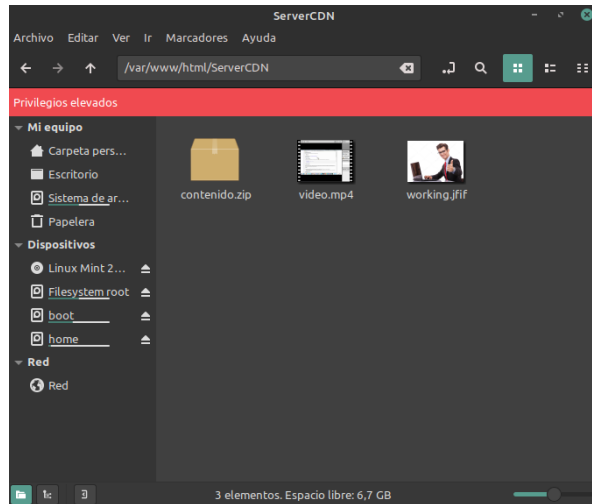
```
sudo nemo /var/www/html
```



Crearemos una carpeta llamada ServerCDN, donde alojaremos los 3 archivos, de modo que el directorio /var/www/html quedará así:



Y dentro de dicho directorio tendremos los 3 ficheros:



Reiniciaremos el servicio de Apache:

```
sudo systemctl stop apache2
```

```
sudo systemctl start apache2
```

Ahora, probaremos que podemos acceder desde el navegador web a estos ficheros por medio del servidor Apache, por ejemplo: <http://localhost:5000/ServerCDN/working.jfif>



Una vez comprobado esto, procederemos a realizar los cambios necesarios en Tomcat para que el programa pueda acceder a los ficheros.

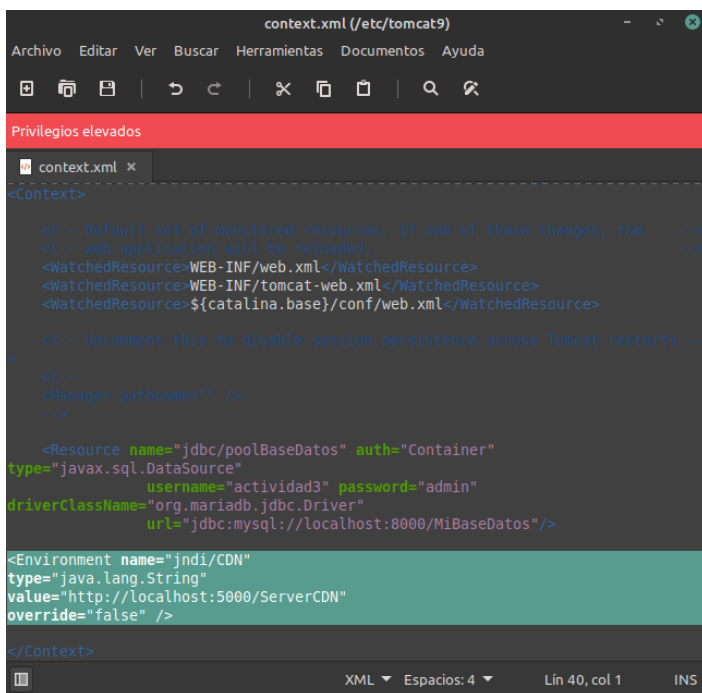
Deberemos crear una variable de entorno, que localizaremos dentro del context.xml, para ello ejecutaremos el comando desde la consola para abrir con permisos elevados el editor de texto:

```
sudo xed /etc/tomcat9/context.xml
```

Deberemos indicar el nombre de acuerdo a lo requerido por el programa, indicaremos que es una variable de entorno de tipo String; en el value colocaremos la dirección del servidor Apache con el puerto y con la carpeta dentro del Apache en la que está:

```
<Environment name="jndi/CDN"
type="java.lang.String"
override="false"
value="http://localhost:5000/ServerCDN"
/>
```

De esta manera, nos quedaría así el context.xml:



```
context.xml (/etc/tomcat9)
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Privilegios elevados
context.xml x
<Context>
  <!-- Default set of monitored resources. If one of these changes, the
  <!-- web application will be reloaded. -->
  <WatchedResource>WEB-INF/web.xml</WatchedResource>
  <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
  <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>

  <!-- Uncomment this to disable session persistence across Tomcat restarts -->
  <!--
  <Manager pathname="" />
  -->

  <Resource name="jdbc/poolBaseDatos" auth="Container"
type="javax.sql.DataSource"
username="actividad3" password="admin"
driverClassName="org.mariadb.jdbc.Driver"
url="jdbc:mysql://localhost:8000/MiBaseDatos"/>

<Environment name="jndi/CDN"
type="java.lang.String"
value="http://localhost:5000/ServerCDN"
override="false" />
</Context>
XML  Espacios: 4  Lín 40, col 1  INS
```


Reiniciaremos el servicio de Tomcat con:

```
sudo systemctl stop tomcat9
```

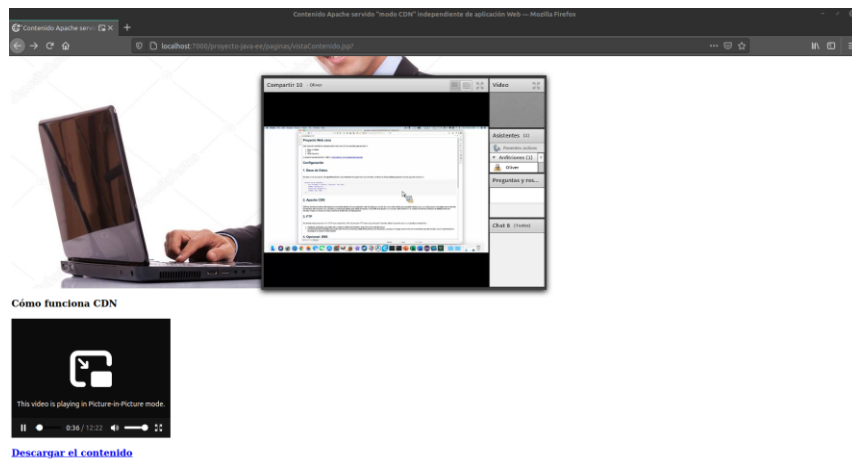
```
sudo systemctl start tomcat9
```

Comprobamos que el CDN ha quedado funcionando de manera correcta:

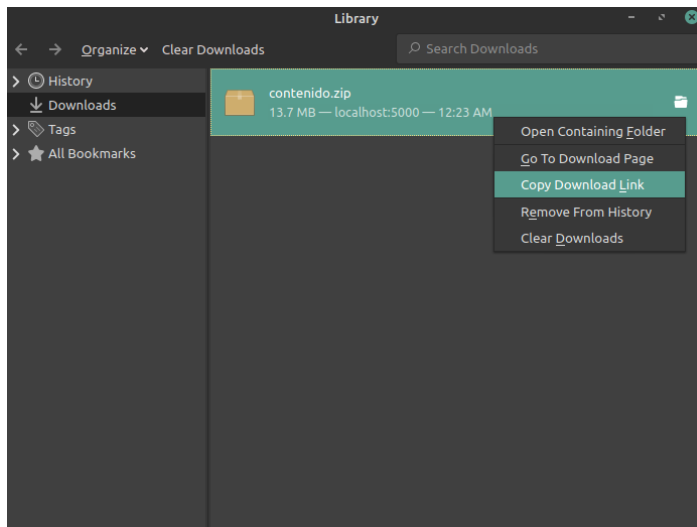
Vemos la imagen:



Vemos el vídeo:

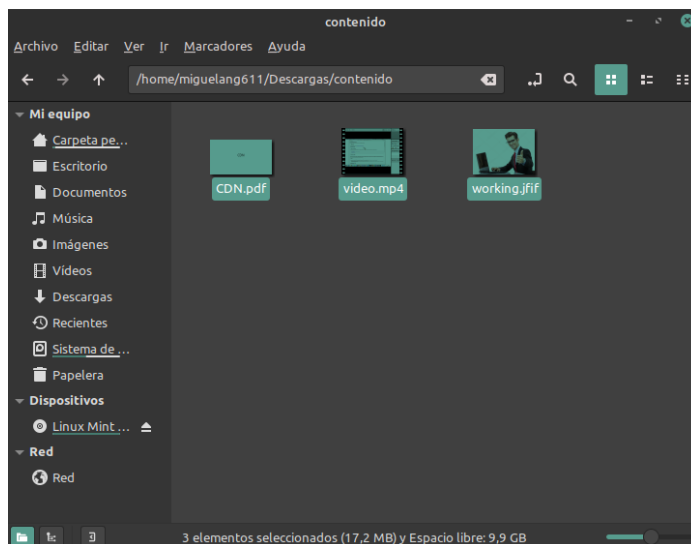


Si hacemos click en descargar, vemos que funciona correctamente, y además, observamos que la dirección de enlace es la del CDN Apache:



<http://localhost:5000/ServerCDN/contenido.zip>

Dentro del zip descargado encontramos lo esperado: el PDF de CDN, el vídeo y la imagen:



4. FTP

Instalamos **FTP Server** con el comando :

```
sudo apt install vsftpd
```

```
sergio@sergio-Lm:/$ sudo apt install vsftpd
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  vsftpd
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 160 no actualizados.
Se necesita descargar 115 kB de archivos.
Se utilizarán 338 kB de espacio de disco adicional después de esta operación.
Des:1 http://archive.ubuntu.com/ubuntu focal/main amd64 vsftpd amd64 3.0.3-12 [115 kB]
Descargados 115 kB en 1s (189 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete vsftpd previamente no seleccionado.
(Leyendo la base de datos ... 286180 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../vsftpd_3.0.3-12_amd64.deb ...
Desempaquetando vsftpd (3.0.3-12) ...
Configurando vsftpd (3.0.3-12) ...
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service → /lib/systemd/system/vsftpd.service.
vsftpd.conf:1: Line references path below legacy directory /var/run/, updating /var/run/vsftpd/empty → /run/vsftpd/empty; please update the tmpfiles.d/ drop-in file accordingly.
Procesando disparadores para man-db (2.9.1-1) ...
Procesando disparadores para systemd (245.4-4ubuntu3.3) ...
sergio@sergio-Lm:/$
```

Creamos una copia de seguridad antes de continuar ya que vamos a hacer cambios en el archivo vsftpd.conf

```
sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.confinicial
```

```
sergio@sergio-Lm:/$ sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.confinicial
```

Ahora pasamos a habilitar y configurar el firewall:

```
sudo ufw status
```

```
sudo ufw allow 20/tcp
```

```
sudo ufw allow 21/tcp
```

```
sergio@sergio-Lm:/$ sudo ufw status
Estado: inactivo
sergio@sergio-Lm:/$ sudo ufw allow 20/tcp
Omitiendo adición de regla ya existente
Omitiendo adición de regla ya existente (v6)
sergio@sergio-Lm:/$ sudo ufw allow 21/tcp
Omitiendo adición de regla ya existente
Omitiendo adición de regla ya existente (v6)
```

20 y 21 puertos de ftp.

Después vamos a crear los usuarios registrado y administrador y vamos a darles una password.

Comentado [SGP1]:

Primero creamos un directorio donde vamos a guardar una lista con los usuarios:

```
sudo mkdir /etc/vsftpd/user_list
```

```
sergio@sergio-Lm:/$ sudo useradd registrado
sergio@sergio-Lm:/$ sudo passwd registrado
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
sergio@sergio-Lm:/$ sudo useradd administrador
sergio@sergio-Lm:/$ sudo passwd administrador
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
sergio@sergio-Lm:/$
```

```
sudo nano /etc/vsftpd/user_list/user.list
```

Añadimos restricciones a cada usuario:

```
sudo echo "registrado" | tee -a /etc/vsftpd/user_list
```

```
sudo echo "administrador" | tee -a /etc/vsftpd/user_list
```

```
sergio@sergio-Lm:/$ sudo echo "registrado" | tee -a /etc/vsftpd/user_list
tee: /etc/vsftpd/user_list: Es un directorio
registrado
sergio@sergio-Lm:/$ sudo echo "administrador" | tee -a /etc/vsftpd/user_list
tee: /etc/vsftpd/user_list: Es un directorio
administrador
sergio@sergio-Lm:/$
```

Creamos las carpetas para los usuarios y le asignamos los permisos correspondientes: (leer y escribir)

```
sergio@sergio-Lm:/$ sudo mkdir -p /home/registrado/ftp/upload
sergio@sergio-Lm:/$ sudo mkdir -p /home/administrador/ftp/upload
sergio@sergio-Lm:/$ sudo chmod 550 /home/registrado/ftp
sergio@sergio-Lm:/$ sudo chmod 750 /home/registrado/ftp/upload
sergio@sergio-Lm:/$ sudo chmod 550 /home/administrador/ftp
sergio@sergio-Lm:/$ sudo chmod 750 /home/administrador/ftp/upload
sergio@sergio-Lm:/$ sudo chown -R registrado: /home/registrado/ftp
sergio@sergio-Lm:/$ sudo chown registrado:registrado /etc/vsftpd/user_list
sergio@sergio-Lm:/$ sudo chown -R administrador: /home/administrador/ftp
sergio@sergio-Lm:/$ sudo chown administrador:administrador /etc/vsftpd/user_list
```

A través del commando `sudo nano /etc/sudoers` agregamos los permisos:

```
sergio@sergio-Lm: /
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
GNU nano 4.8 /etc/sudoers Modificado
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults      env_reset
Defaults      mail_badpass
Defaults      secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
sergio  ALL=(ALL:ALL) ALL
administrador  ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

^G Ver ayuda  ^O Guardar  ^W Buscar  ^K Cortar Texto  ^J Justificar  ^C Posición
^X Salir      ^R Leer fich. ^_ Reemplazar  ^U Pegar      ^T Ortografía  ^_ Ir a línea
```

Vamos a crear una carpeta en `/home` que se llama `ftp`:

```
sudo mkdir /home/ftp
```

Y le agregamos los permisos de lectura y escritura al usuario y al grupo de propietario con el 6, y el 4 al resto, que es de solo lectura.

```
sergio@sergio-Lm:/$ sudo mkdir /home/ftp
sergio@sergio-Lm:/$ sudo chmod 664 /home/ftp
sergio@sergio-Lm:/$
```

Creamos un archivo `vsftpd.chroot_list` y añadimos los dos usuarios en el mismo:

```
sudo nano /etc/vsftpd/vsftpd.chroot_list
```

```
sergio@sergio-Lm: /
GNU nano 4.8 /etc/vsftpd/vsftpd.chroot_list Modificado
registrado
administrador

[ Archivo nuevo ]
^G Ver ayuda  ^O Guardar  ^W Buscar  ^K Cortar Texto  ^J Justificar  ^C Posición
^X Salir      ^R Leer fich.  ^N Reemplazar  ^U Pegar      ^T Ortografía  ^_ Ir a línea
```

Ahora vamos a configurar los permisos de los usuarios en el archivo vsftpd.conf:

```
sudo nano /etc/vsftpd.conf
```

Dentro de este archivo vamos a añadir y a modificar los siguientes parametros:

```
sergio@sergio-Lm: /
GNU nano 4.8 /etc/vsftpd.conf
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (:::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
write_enable=YES
chroot_local_user=YES
allow_writeable_chroot=YES
userlist_enable=YES
userlist_file=/etc/vsftpd.userlist
userlist_deny=NO
local_root=/home/ftp
chroot_local_user=YES
chroot_list_enable=NO
chroot_list_file=/etc/vsftpd/vsftpd.chroot_list
#
# Uncomment this to enable any form of FTP write command.
```

Despues de todos los cambios realizados vamos a iniciar el servidor vsftpd con :

```
sudo systemctl start vsftpd  
sudo systemctl enable vsftpd
```

Finalmente utilizamos el commando **sudo ftp localhost**, en el cual nos pedira usuario y password, e iniciaremos conexión.

```
sergio@sergio-Lm:/$ sudo ftp localhost  
Connected to localhost.  
220 (vsFTPd 3.0.3)  
Name (localhost:sergio): registrado  
331 Please specify the password.  
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> █
```

Como podemos observar lista los archivos y directorios que hay en la carpeta y se logea correctamente, pero por algún motivo que no hemos conseguido resolver de los permisos, no es capaz de subir los archivos.

El servidor FTP no está configurado correctamente

Salida FTP

220 (vsFTPd 3.0.3)
230 Login successful.

Indica el parámetro para el comando que quieres ejecutar :

CD

Get

Selecciona un fichero para subir y el nombre que quieres darle :

Put

Browse...

No file selected.

Contenido del servidor FTP :

Tipo	Nombre	Tamaño	Fecha de creación
f	bin	7 bytes	Tue Feb 23 19:27:00 CET 2021
d	boot	4096 bytes	Tue Feb 23 19:36:00 CET 2021
d	cdrom	4096 bytes	Tue Feb 23 19:31:00 CET 2021
d	dev	4100 bytes	Mon Mar 01 10:08:00 CET 2021
d	etc	12288 bytes	Mon Mar 01 12:33:00 CET 2021
d	home	4096 bytes	Mon Mar 01 12:23:00 CET 2021
f	lib	7 bytes	Tue Feb 23 19:27:00 CET 2021