INSTITUTO POLITÉCNICO NACIONAL
ESCOM

PINEDA GUERRERO CARLOS
DESARROLLO DE SISTEMAS DISTRIBUIDOS
4CV12

ABRAHAM DANIEL TOVAR
2018631321

TAREA 5
**"Implementación de un servicio web
REST para Tomcat"**
25/04/2024

1. Creacion de máquina virtual

- Se crea máquina virtual en +Crear, eliges grupo de recursos
- Ingresar nombre de la máquina virtual T5-2018631321
- Tipo de Seguridad Estandar
- Seleccione Ubuntu Server 20

- Tamaño VM B1s
- Contraseña

- Disco duro HD estandar

- Redes

2. Iniciar sesion a la maquina Virtual

- Se inicia sesion y se upgradea las dependencias
- Tambien instalamos el JDK 8

```
dani@T5-2018631321:~$ sudo apt install openjdk-8-jdk-headless
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
The following additional packages will be installed:
  ca-certificates-java fontconfig-config fonts-dejavu-core java-common libavahi-client3
  libavahi-common-data libavahi-common3 libcups2 libfontconfig1 libjpeg-turbo8 libjpeg8
  liblcms2-2 libpcsclite1 libxi6 libxrender1 libxtst6 openjdk-8-jre-headless x11-common
Suggested packages:
  default-jre cups-common liblcms2-utils pcscd openjdk-8-demo openjdk-8-source libnss-mdns
  fonts-dejavu-extra fonts-nanum fonts-ipafont-gothic fonts-ipafont-mincho
  fonts-wqy-microhei fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  ca-certificates-java fontconfig-config fonts-dejavu-core java-common libavahi-client3
  libavahi-common-data libavahi-common3 libcups2 libfontconfig1 libjpeg-turbo8 libjpeg8
  liblcms2-2 libpcsclite1 libxi6 libxrender1 libxtst6 openjdk-8-jdk-headless
  openjdk-8-jre-headless x11-common
0 upgraded, 19 newly installed, 0 to remove and 28 not upgraded.
Need to get 38.4 MB of archives.
After this operation, 150 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

3. Creacion de archivos

- Instalamos unzip para desempaquetar los archivos que se estaran enviando como en este caso que se envió apache tomcat

```
dani@T5-2018631321:~$ sudo apt install unzip
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 28 not upgraded.
Need to get 169 kB of archives.
After this operation, 593 kB of additional disk space will be used.
Get:1 http://azure.archive.ubuntu.com/ubuntu focal-updates/main amd64 unzip amd64 6.0-25ubun
tu1.2 [169 kB]
Fetched 169 kB in 0s (6850 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 59512 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-25ubuntu1.2_amd64.deb ...
Unpacking unzip (6.0-25ubuntu1.2) ...
Setting up unzip (6.0-25ubuntu1.2) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
dani@T5-2018631321:~$ unzip apache-tomcat-8.5.100.zip
Archive:  apache-tomcat-8.5.100.zip
   creating: apache-tomcat-8.5.100/
   creating: apache-tomcat-8.5.100/bin/
   creating: apache-tomcat-8.5.100/conf/
   creating: apache-tomcat-8.5.100/lib/
   creating: apache-tomcat-8.5.100/logs/
   creating: apache-tomcat-8.5.100/temp/
   creating: apache-tomcat-8.5.100/webapps/
   creating: apache-tomcat-8.5.100/webapps/ROOT/
   creating: apache-tomcat-8.5.100/webapps/ROOT/WEB-INF/
   creating: apache-tomcat-8.5.100/webapps/docs/
   creating: apache-tomcat-8.5.100/webapps/docs/META-INF/
```

al final se mostrar una captura de cómo se enviaron todos los archivos pero aqui
una mini muestra

también pasamos jersey a nuestra máquina virtual y desempaquetamos

vamos a copiar todos los archivos **.jar** en **apache/lib**

descargamos el connector/J 8.3.0 el plataform independent pero el .zip

4. Configuracion Tomcat

- Vamos a definir variables de entorno CATALINA_HOME Y JAVA_HOME
- Probaremos la ejecucion de tomcat sh
  $CATALINA_HOME/bin/catalina.sh start
  $CATALINA_HOME/bin/catalina.sh stop

```
dani@T5-2018631321:~$ export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
dani@T5-2018631321:~$ export CATALINA_HOME=/home/dani/apache-tomcat-8.5.100
dani@T5-2018631321:~$ sh $CATALINA_HOME/bin/catalina.sh start
Using CATALINA_BASE:   /home/dani/apache-tomcat-8.5.100
Using CATALINA_HOME:   /home/dani/apache-tomcat-8.5.100
Using CATALINA_TMPDIR: /home/dani/apache-tomcat-8.5.100/temp
Using JRE_HOME:        /usr/lib/jvm/java-8-openjdk-amd64
Using CLASSPATH:       /home/dani/apache-tomcat-8.5.100/bin/bootstrap.jar:/home/dani/apache-
tomcat-8.5.100/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
dani@T5-2018631321:~$ sh $CATALINA_HOME/bin/catalina.sh stop
Using CATALINA_BASE:   /home/dani/apache-tomcat-8.5.100
Using CATALINA_HOME:   /home/dani/apache-tomcat-8.5.100
Using CATALINA_TMPDIR: /home/dani/apache-tomcat-8.5.100/temp
Using JRE_HOME:        /usr/lib/jvm/java-8-openjdk-amd64
Using CLASSPATH:       /home/dani/apache-tomcat-8.5.100/bin/bootstrap.jar:/home/dani/apache-
tomcat-8.5.100/bin/tomcat-juli.jar
Using CATALINA_OPTS:
dani@T5-2018631321:~$ sudo apt install mysql-server
```

5. Instalacion y configuracion de Mysql

- Se hara la instalacion de MySql con el comando **sudo apt install mysql-server**

- Reiniciamos todas la tablas y usuarios para tener una instalación limpia

```
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.15) ...
dani@T5-2018631321:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: N

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER_USER" command.
See https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management for more information.

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : Y
Success.


Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : Y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.


Remove test database and access to it? (Press y|Y for Yes, any other key for No) : Y
 - Dropping test database...
Success.

 - Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y
Success.

All done!
dani@T5-2018631321:~$
```

Después cambiamos la contraseña por default de root


```
dani@T5-2018631321:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.36-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'dani1234';
Query OK, 0 rows affected (0.08 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)

mysql> quit
Bye
dani@T5-2018631321:~$
```

Se creará un nuevo usuario y le entregamos todo el poder

```
dani@T5-2018631321:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.36-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create user dani@localhost identified by 'dani1234';
Query OK, 0 rows affected (0.05 sec)

mysql> grant all on servicio_web.* to dani@localhost;
Query OK, 0 rows affected (0.03 sec)

mysql> quit
Bye
dani@T5-2018631321:~$
```

Creamos la base de datos y las tablas

```
dani@T5-2018631321:~$ mysql -u dani -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.36-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database servicio_web;
Query OK, 1 row affected (0.04 sec)

mysql> use servicio_web;
Database changed
mysql> create table usuarios
    -> (
    -> id_usuarios integer auto_increment primary key,
    -> email varchar(100) not null,
    -> nombre varchar(100) not null,
    -> apellido_paterno varchar(100) not null,
    -> apellido_materno varchar(100),
    -> fecha_nacimiento datetime not null,
    -> telefono bigint,
    -> genero
Broadcast message from root@T5-2018631321 (Thu 2024-04-25 07:08:07 UTC):

Azure Patch Management initiated a reboot after a patch installation run.
The system is going down for reboot at Thu 2024-04-25 07:18:07 UTC!

 char(1)
    -> );
Query OK, 0 rows affected (0.16 sec)

mysql>
```

no sale seguida la tabla porque se me reinicio la sesion por algo de azure

```
mysql> CREATE TABLE fotos_usuarios (
    →        id_foto INTEGER AUTO_INCREMENT PRIMARY KEY,
    →        foto LONGBLOB,
    →        id_usuario INTEGER NOT NULL
    → );
Query OK, 0 rows affected (0.17 sec)

mysql>
```

```
mysql> alter table fotos_usuarios add foreign key (id_usuario) references usuarios(id_usuari
o);
Query OK, 0 rows affected (0.35 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> create unique index usuarios_1 on usuarios(email);
Query OK, 0 rows affected (0.12 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> quit
Bye
dani@T5-2018631321:~$ |
```

```
mysql> show tables;
+----------------------+
| Tables_in_servicio_web |
+----------------------+
| fotos_usuarios       |
| usuarios             |
+----------------------+
2 rows in set (0.00 sec)

mysql> |
```
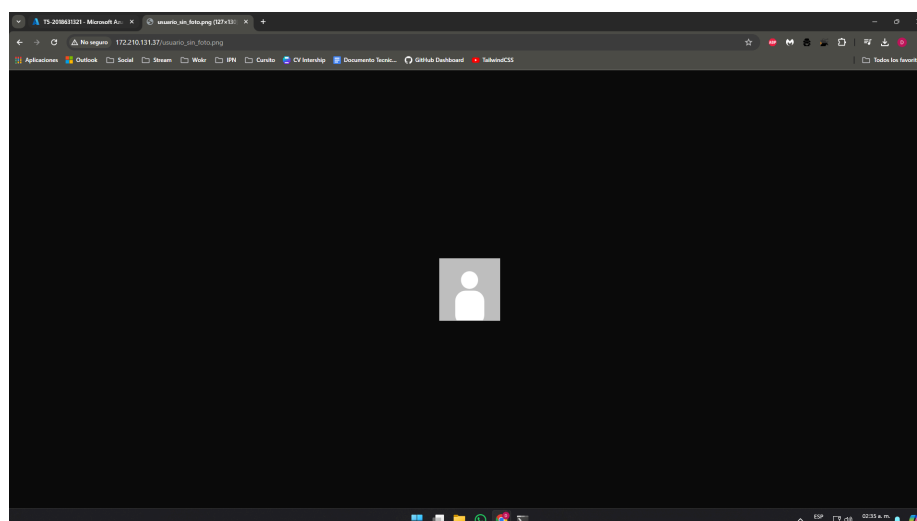
6. Pruebas de aplicaciones

Añadimos los archivos de pruebas Json, URL e imagen a la carpeta
webapps/ROOT, de tomcat
(así fue como se enviaron todos los archivos se me habia olvidado tomar la captura)

- Probamos la configuracion y conexion intentando acceder a la imagen
  usuario_sin_foto.png

- Después descargamos el archivo servicio.zip y lo enviamos a nuestra máquina virtual, extraemos los archivos y nos metemos a la carpeta que se acaba de crear

```
dani@T5-2018631321:~$ unzip Servicio.zip
Archive:  Servicio.zip
  inflating: Servicio/compila_json.sh
  inflating: Servicio/compila_url.sh
   creating: Servicio/META-INF/
  inflating: Servicio/META-INF/context.xml
   creating: Servicio/servicio_json/
  inflating: Servicio/servicio_json/AdaptadorGsonBase64.java
  inflating: Servicio/servicio_json/Error.java
  inflating: Servicio/servicio_json/ParamAltaUsuario.java
  inflating: Servicio/servicio_json/ParamBorraUsuario.java
  inflating: Servicio/servicio_json/ParamConsultaUsuario.java
  inflating: Servicio/servicio_json/ParamModificaUsuario.java
  inflating: Servicio/servicio_json/Servicio.java
  inflating: Servicio/servicio_json/Usuario.java
   creating: Servicio/servicio_url/
  inflating: Servicio/servicio_url/AdaptadorGsonBase64.java
  inflating: Servicio/servicio_url/Error.java
  inflating: Servicio/servicio_url/Servicio.java
  inflating: Servicio/servicio_url/Usuario.java
   creating: Servicio/WEB-INF/
   creating: Servicio/WEB-INF/classes/
   creating: Servicio/WEB-INF/classes/servicio_json/
   creating: Servicio/WEB-INF/classes/servicio_url/
  inflating: Servicio/WEB-INF/web.xml
dani@T5-2018631321:~$ cd Servicio/
dani@T5-2018631321:~/Servicio$
```

- Después accedemos a la carpeta META-INF y modificamos el archivo context.xml añadiendo el usuario y contraseña de nuestra base de datos

```
dani@T5-2018631321:~$ cd Servicio/
dani@T5-2018631321:~/Servicio$ cd META-INF/
dani@T5-2018631321:~/Servicio/META-INF$ nano context.xml
dani@T5-2018631321:~/Servicio/META-INF$ cat context.xml
<Context>
        <Resource name="jdbc/datasource_Servicio" auth="Container" type="javax.sql.DataSourc
e"
                maxActive="100" maxIdle="30" maxWait="10000"
                username="dani" password="dani1234"
                driverClassName="com.mysql.cj.jdbc.Driver"
                url="jdbc:mysql://localhost/servicio_web?serverTimezone=UTC" />
</Context>
```
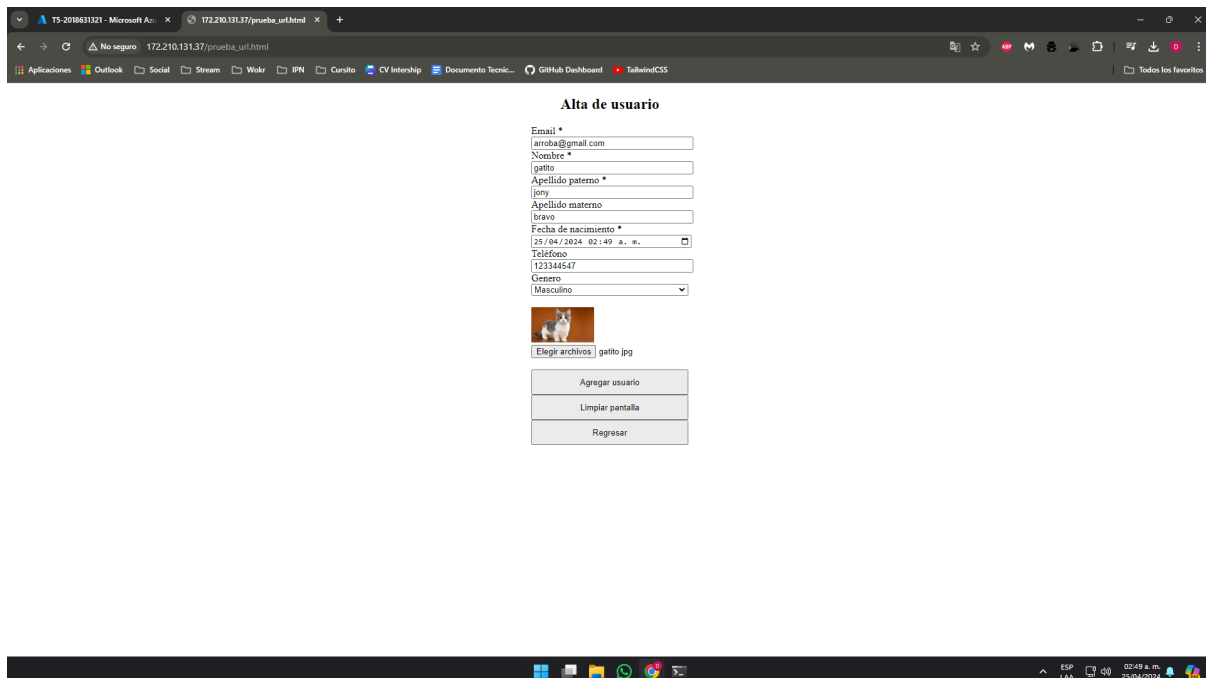
- Compilamos el servicio web con el siguiente comando:

```
dani@T5-2018631321:~/Servicio$ javac -cp $CATALINA_HOME/lib/javax.ws.rs-api-2.0.1.jar:$CATAL
INA_HOME/lib/gson-2.3.1.jar:. servicio_url/Servicio.java
dani@T5-2018631321:~/Servicio$ ls
META-INF  WEB-INF  compila_json.sh  compila_url.sh  servicio_json  servicio_url
```

- Creamos el archivo WAR:

```
dani@T5-2018631321:~/Servicio$ rm WEB-INF/classes/servicio_url/
rm: cannot remove 'WEB-INF/classes/servicio_url/': Is a directory
dani@T5-2018631321:~/Servicio$ rm WEB-INF/classes/servicio_json/
rm: cannot remove 'WEB-INF/classes/servicio_json/': Is a directory
dani@T5-2018631321:~/Servicio$ rm WEB-INF/classes/servicio_url/*
rm: cannot remove 'WEB-INF/classes/servicio_url/*': No such file or directory
dani@T5-2018631321:~/Servicio$ rm WEB-INF/classes/servicio_json/*
rm: cannot remove 'WEB-INF/classes/servicio_json/*': No such file or directory
dani@T5-2018631321:~/Servicio$ cp servicio_url/*.class WEB-INF/classes/servicio_url/.
dani@T5-2018631321:~/Servicio$ jar cvf Servicio.war WEB-INF META-INF
added manifest
adding: WEB-INF/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/web.xml(in = 656) (out= 294)(deflated 55%)
adding: WEB-INF/classes/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/servicio_url/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/servicio_url/Error.class(in = 283) (out= 219)(deflated 22%)
adding: WEB-INF/classes/servicio_url/Servicio.class(in = 8240) (out= 3838)(deflated 53%)
adding: WEB-INF/classes/servicio_url/Usuario.class(in = 1070) (out= 604)(deflated 43%)
adding: WEB-INF/classes/servicio_url/AdaptadorGsonBase64.class(in = 1804) (out= 741)(deflated 58%)
adding: WEB-INF/classes/servicio_json/(in = 0) (out= 0)(stored 0%)
ignoring entry META-INF/
adding: META-INF/context.xml(in = 304) (out= 212)(deflated 30%)
dani@T5-2018631321:~/Servicio$ rm -rf $CATALINA_HOME/webapps/Servicio.war $CATALINA_HOME/webapps/Servicio
dani@T5-2018631321:~/Servicio$ cp Servicio.war $CATALINA_HOME/webapps/.
dani@T5-2018631321:~/Servicio$
```

a. **Aquí probamos desde la pc el archivo prueba_url.html en donde empezamos haciendo el alta de un usuario**

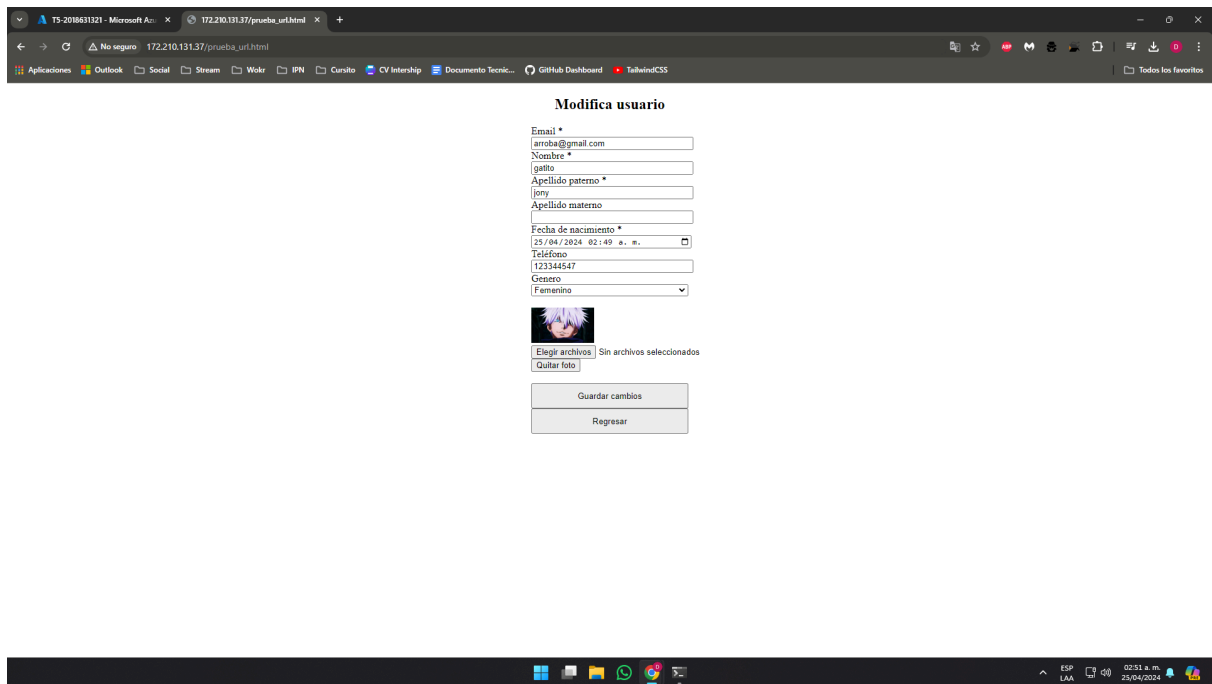**b. Probamos crear con el mismo email y sale que esta duplicado**
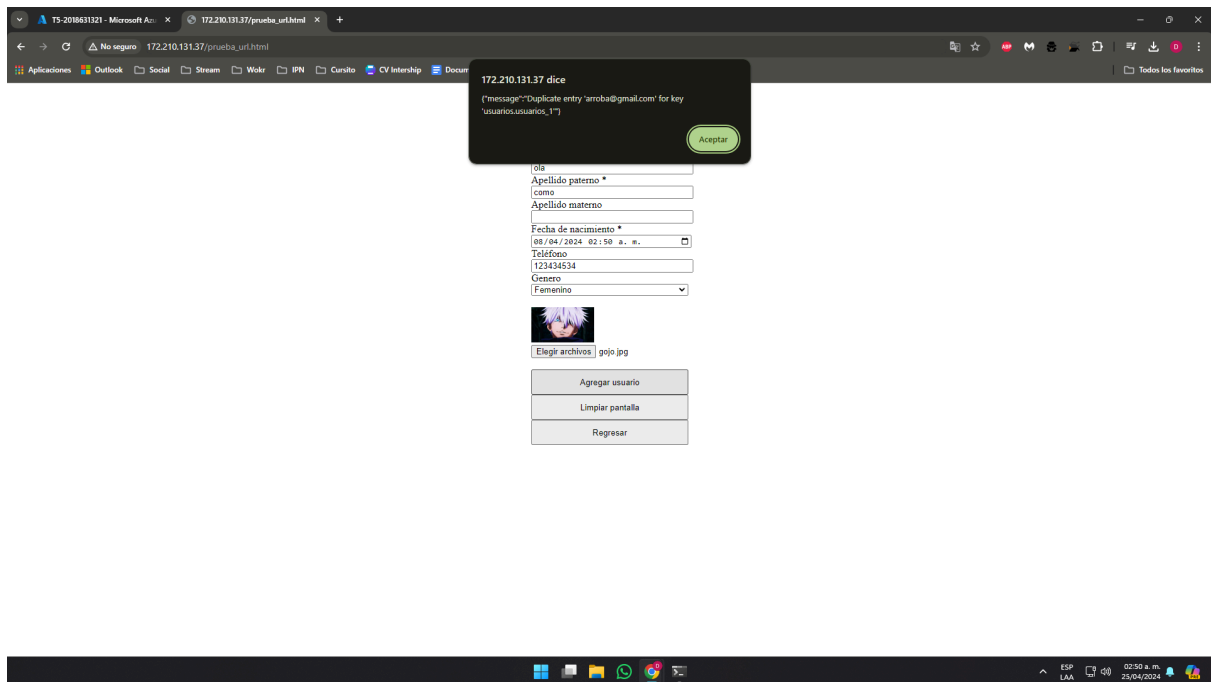
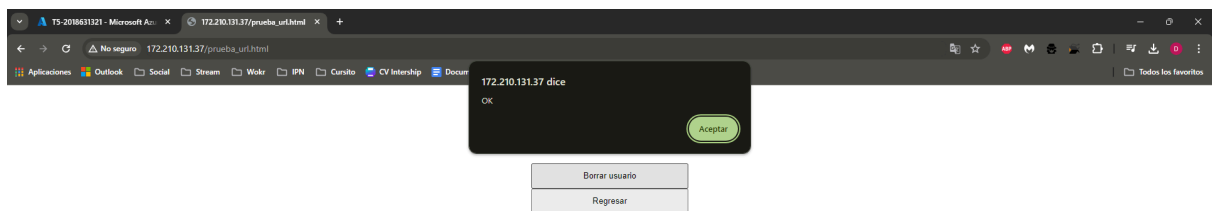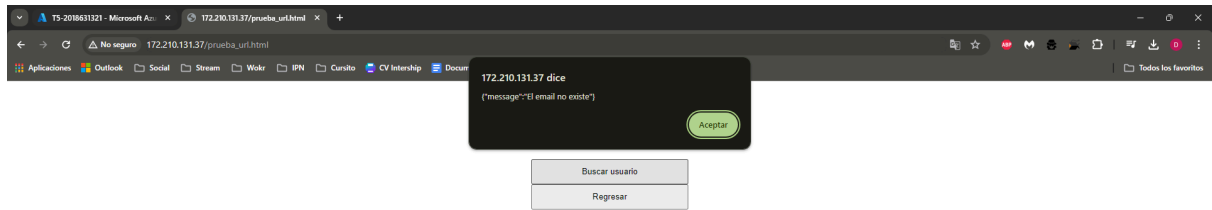**c. aquí consultaremos el usuario crado y se modificara**
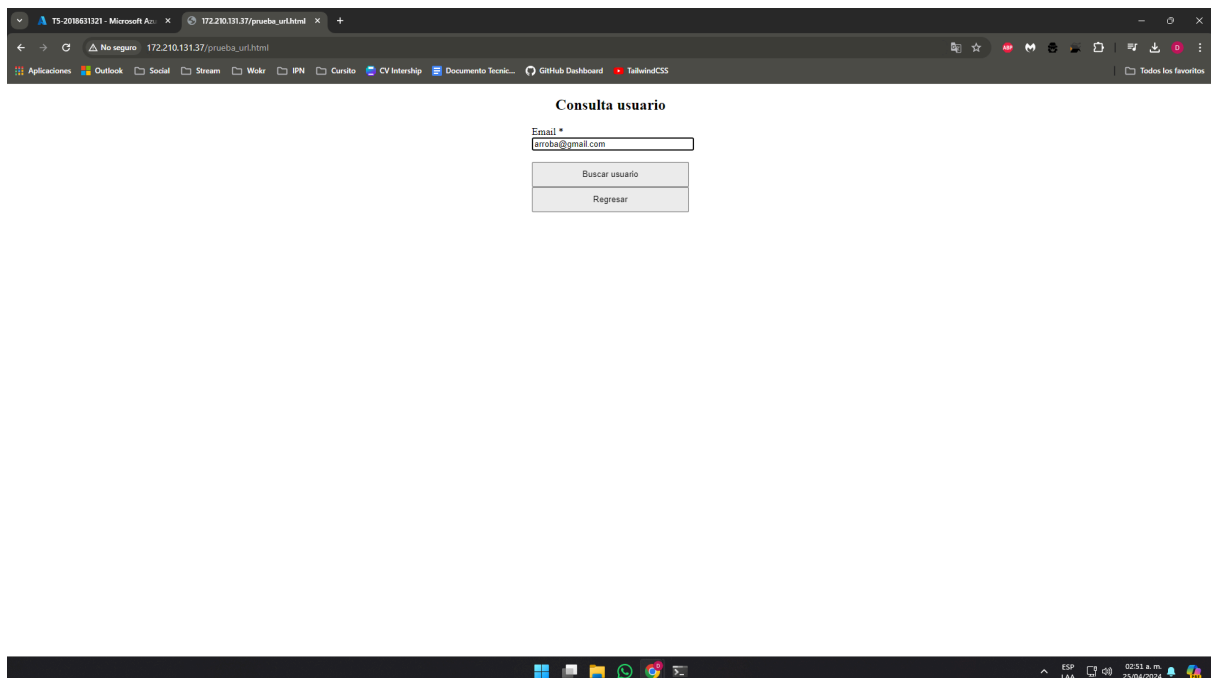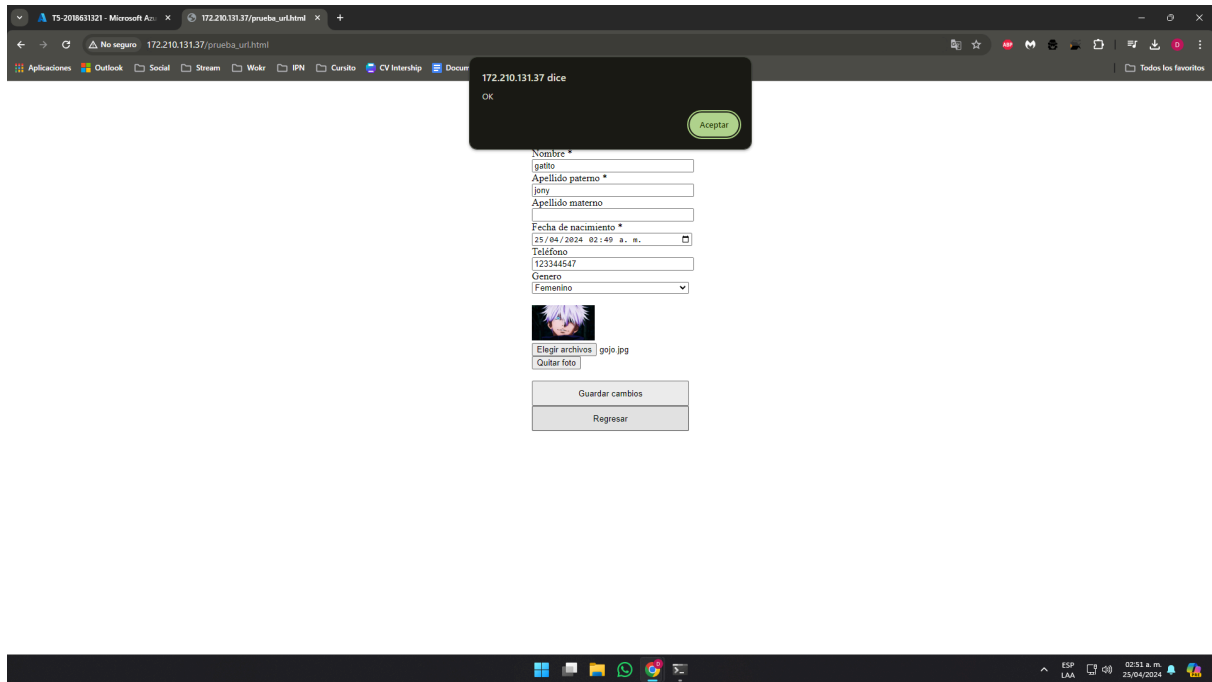
d. muestra de que si se modifico con exito

e. Aqui borramos e intentamos consultar el mismo usuario borrado por lo que sale aviso de que el email ya no existe

7. Repetimos los pasos para prueba_json.html

- Compilamos el servicio web con el siguiente comando
- Creamos el archivo WAR

```
dani@T5-2018631321:~/Servicio$ rm -rf $CATALINA_HOME/webapps/Servicio.war $CATALINA_HOME/web
apps/Servicio
dani@T5-2018631321:~/Servicio$ cp Servicio.war $CATALINA_HOME/webapps/.
dani@T5-2018631321:~/Servicio$ rm -rf $CATALINA_HOME/webapps/Servicio.war $CATALINA_HOME/web
apps/Servicio
dani@T5-2018631321:~/Servicio$ javac -cp $CATALINA_HOME/lib/javax.ws.rs-api-2.0.1.jar:$CATAL
INA_HOME/lib/gson-2.3.1.jar:. servicio_json/Servicio.java
dani@T5-2018631321:~/Servicio$ rm WEB-INF/classes/servicio_url/*
dani@T5-2018631321:~/Servicio$ rm WEB-INF/classes/servicio_json/*
rm: cannot remove 'WEB-INF/classes/servicio_json/*': No such file or directory
dani@T5-2018631321:~/Servicio$ cp servicio_json/*.class WEB_INF/classes/servicio_json/.
cp: target 'WEB_INF/classes/servicio_json/.' is not a directory
dani@T5-2018631321:~/Servicio$ cp servicio_json/*.class WEB-INF/classes/servicio_json/.
dani@T5-2018631321:~/Servicio$ jar cvf Servicio.war WEB-INF META-INF
added manifest
adding: WEB-INF/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/web.xml(in = 656) (out= 294)(deflated 55%)
adding: WEB-INF/classes/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/servicio_url/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/servicio_json/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/servicio_json/Error.class(in = 284) (out= 220)(deflated 22%)
adding: WEB-INF/classes/servicio_json/Servicio.class(in = 8612) (out= 3970)(deflated 53%)
adding: WEB-INF/classes/servicio_json/ParamBorraUsuario.class(in = 259) (out= 206)(deflated
20%)
adding: WEB-INF/classes/servicio_json/ParamModificaUsuario.class(in = 272) (out= 205)(deflat
ed 24%)
adding: WEB-INF/classes/servicio_json/ParamAltaUsuario.class(in = 264) (out= 198)(deflated 2
5%)
adding: WEB-INF/classes/servicio_json/Usuario.class(in = 435) (out= 295)(deflated 32%)
adding: WEB-INF/classes/servicio_json/ParamConsultaUsuario.class(in = 265) (out= 208)(deflat
ed 21%)
adding: WEB-INF/classes/servicio_json/AdaptadorGsonBase64.class(in = 1805) (out= 741)(deflat
ed 58%)
ignoring entry META-INF/
adding: META-INF/context.xml(in = 304) (out= 212)(deflated 30%)
dani@T5-2018631321:~/Servicio$ rm -rf $CATALINA_HOME/webapps/Servicio.war $CATALINA_HOME/web
apps/Servicio
dani@T5-2018631321:~/Servicio$ cp Servicio.war $CATALINA_HOME/webapps/.
dani@T5-2018631321:~/Servicio$
```

a. **En este lo probaremos desde el celular el archivo prueba_json.html en donde empezamos haciendo el alta de un usuario**
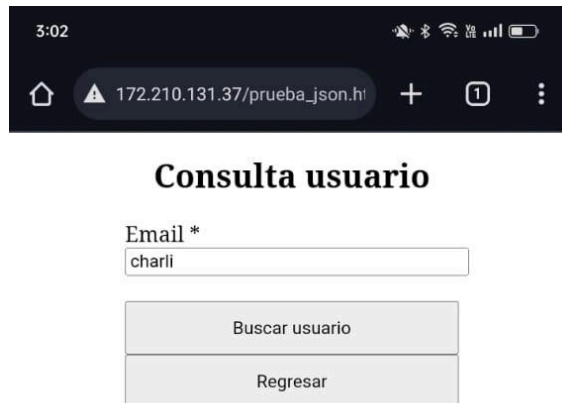

b. **Probamos crear con el mismo email y sale que esta duplicado**

c. **Aquí consultaremos el usuario creado y se modificara**
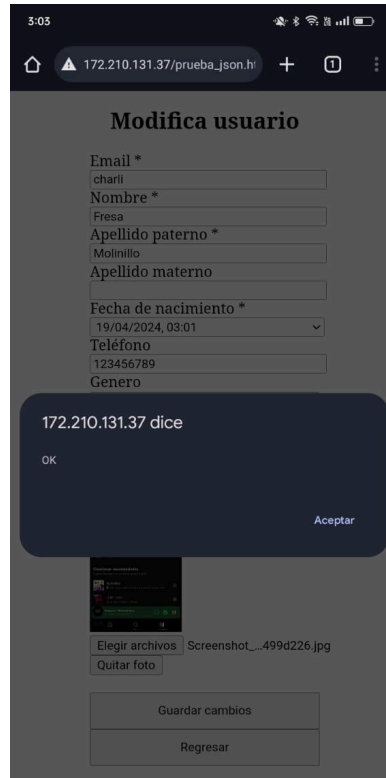
d. **Muestra de que si se modificó con éxito**

e. **Aquí borramos e intentamos consultar el mismo usuario borrado por lo que sale aviso de que el email ya no existe**
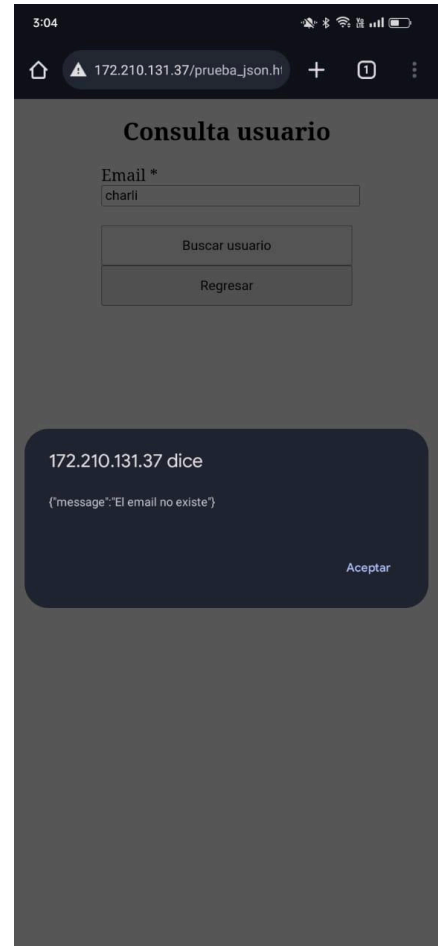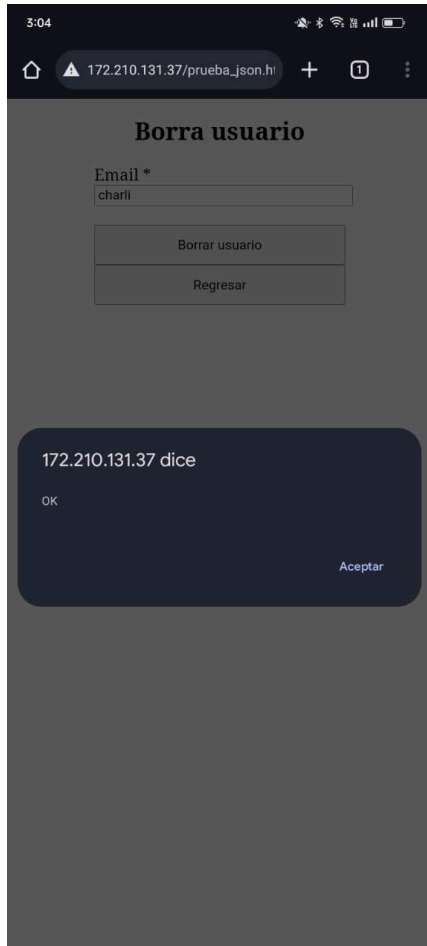
8. Configuramos para que inicie tomcat cuando encienda la maquina virtual

9. Se Creó con éxito la imagen de la máquina virtual

10. Conclusiones

La implementación de un servicio web en la nube con Apache Tomcat involucra una serie de pasos técnicos y de configuración que, cuando se realizan correctamente, permiten desplegar y probar servicios web de manera efectiva y eficiente. Este proceso requiere atención a los detalles y conocimientos sólidos de las tecnologías involucradas, pero una vez completado con éxito, proporciona una plataforma estable y escalable para la entrega de aplicaciones web.