



Instituto Politécnico Nacional Escuela Superior de Cómputo

Tarea 7. Implementación de un servicio web estilo REST

PRESENTA

Vladimir Azpeitia Hernández

PROFESOR

Carlos Pineda Guerrero

ASGINATURA

Desarrollo de sistemas distribuidos

30 de abril de 2021

1.1. Planteamiento del problema

Cada alumno ejecutará el procedimiento que vimos en clase, dónde instalamos Tomcat, instalamos MySQL, y creamos un servicio web estilo REST. Se deberá probar el servicio web utilizando la aplicación web prueba.html tal como se explicó en clase.

1.2. Formato de entrega

- Se deberá subir a la plataforma el código fuente de los programas y un reporte de la tarea en formato PDF con portada, desarrollo y conclusiones como mínimo.
- El reporte PDF deberá incluir las capturas de pantalla de la compilación y ejecución del programa, se deberá incluir la captura de pantalla correspondiente a cada paso de la creación de la máquina virtual.
- No se admitirá la tarea si no incluye las pantallas correspondientes a cada paso del procedimiento de creación de la máquina virtual.
- El nombre de la máquina virtual deberá ser el número de boleta del alumno, si el número de boleta del alumno es 12345678, entonces la máquina virtual deberá llamarse: A12345678. No se admitirá la tarea si la máquina virtual no se nombra como se indicó anteriormente.
- La captura de pantallas deberá estar completa, no se admitirá la tarea si incluye imágenes que sean cortes de las capturas de pantalla.
- Recuerden que deben eliminar la máquina virtual cuando no la usen, con la finalidad de ahorrar el saldo de sus cuentas de Azure.
- No se admitirá la tarea si se envía en formato RAR o en formato WORD.

1.3. Puntaje de la tarea

Valor de la tarea: 20 % (1.2 punto de la segunda evaluación parcial)

2.1. Creación y configuración de las máquina virtual

A continuación se muestran los pasos para configurar la máquina virtual de Ubuntu Server 18.04, la cual se utilizara para el desarrollo de esta práctica

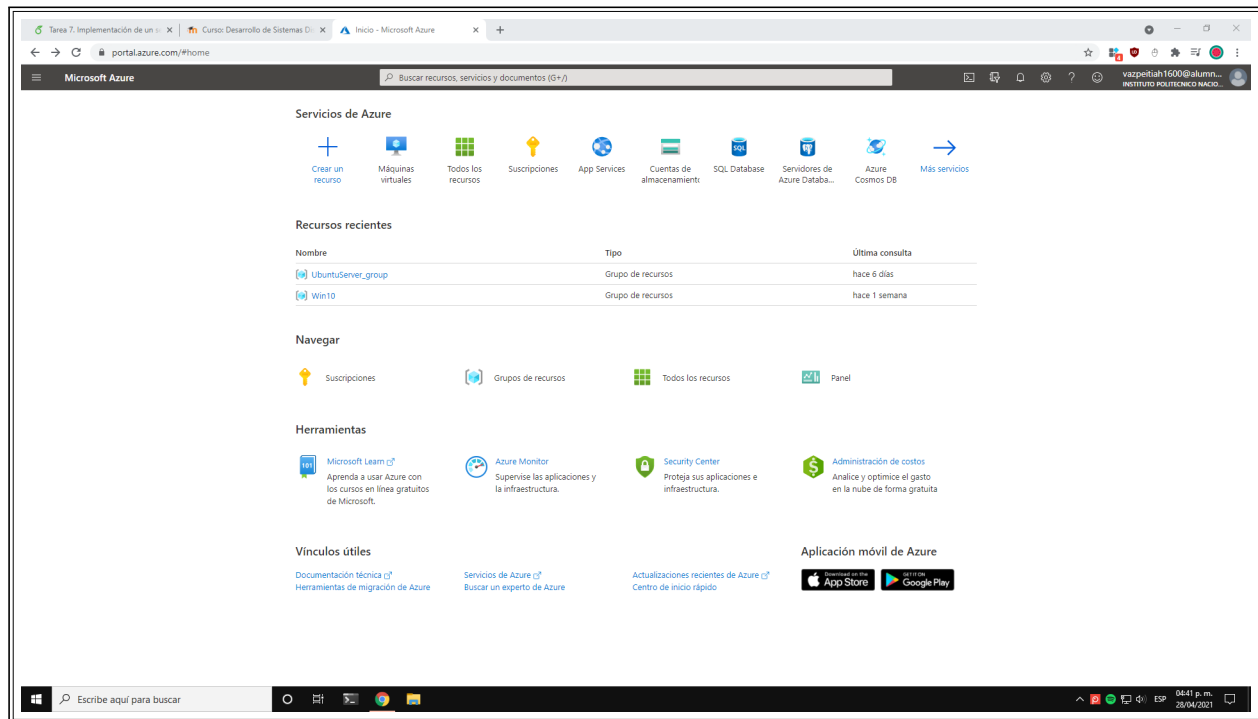


Figura 2.1: Creación de la máquina virtual: Portal de azure

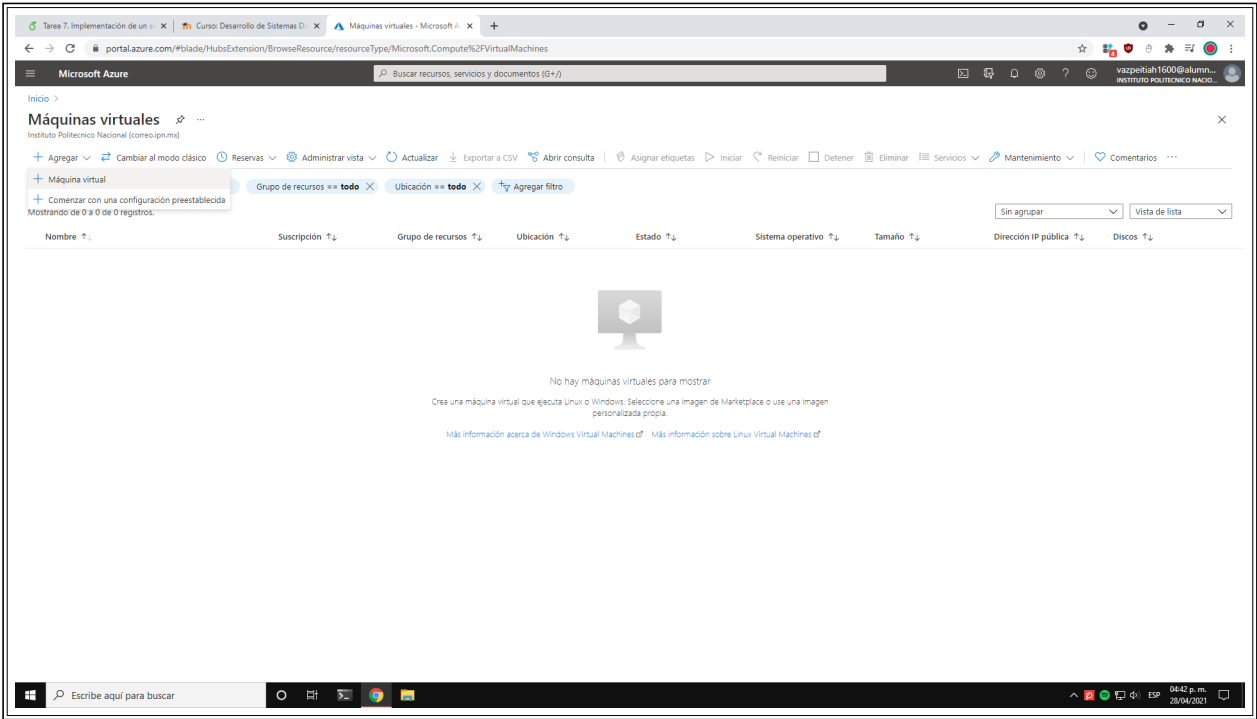


Figura 2.2: Creación de la máquina virtual: Máquinas virtuales

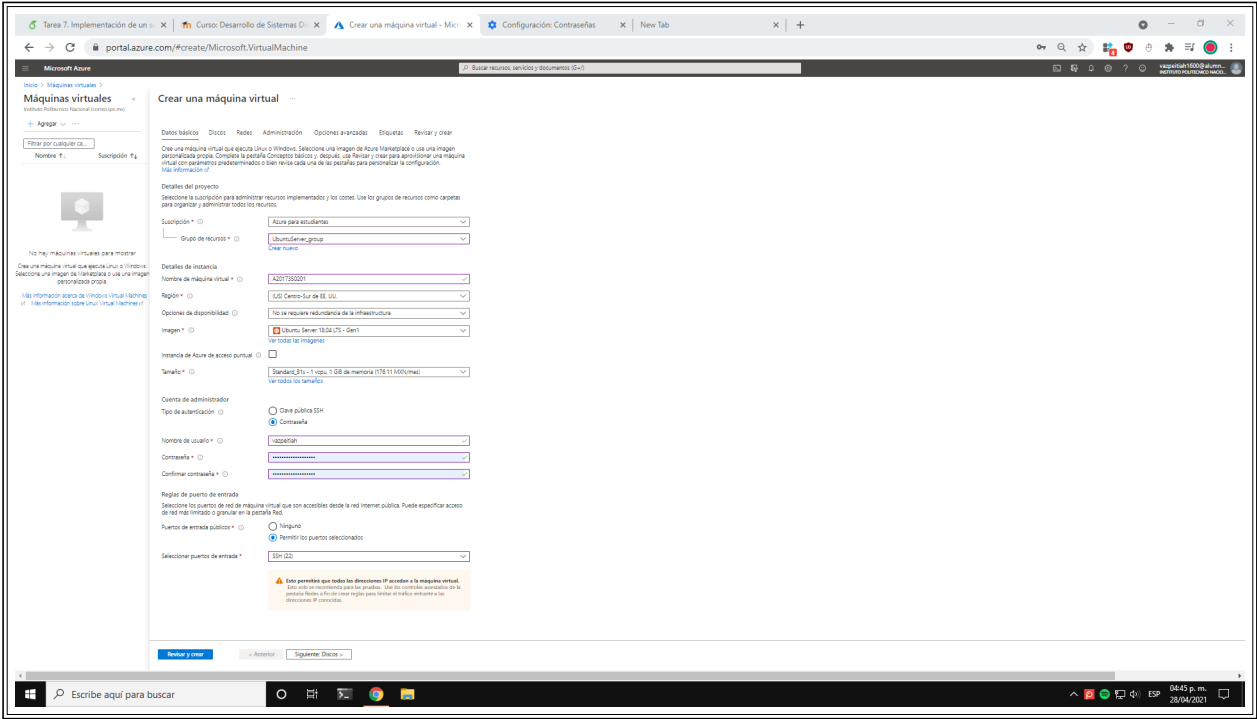


Figura 2.3: Creación de la máquina virtual: Datos básicos

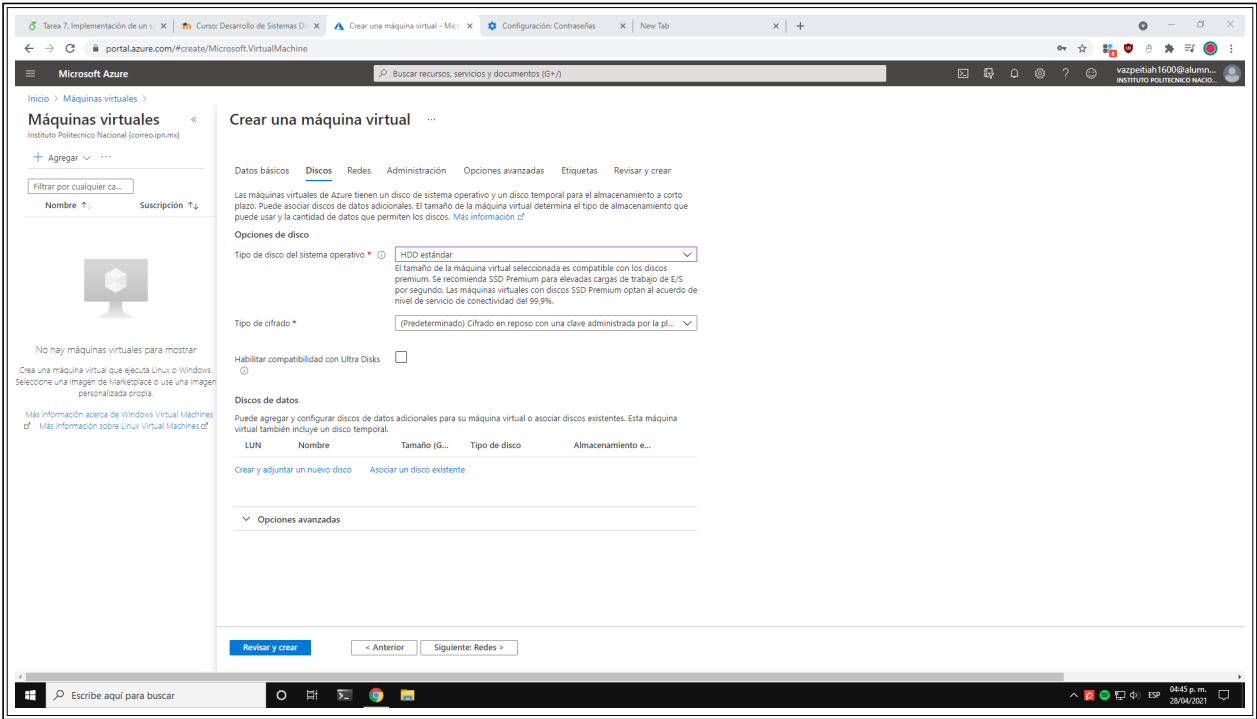


Figura 2.4: Creación de la máquina virtual: Discos

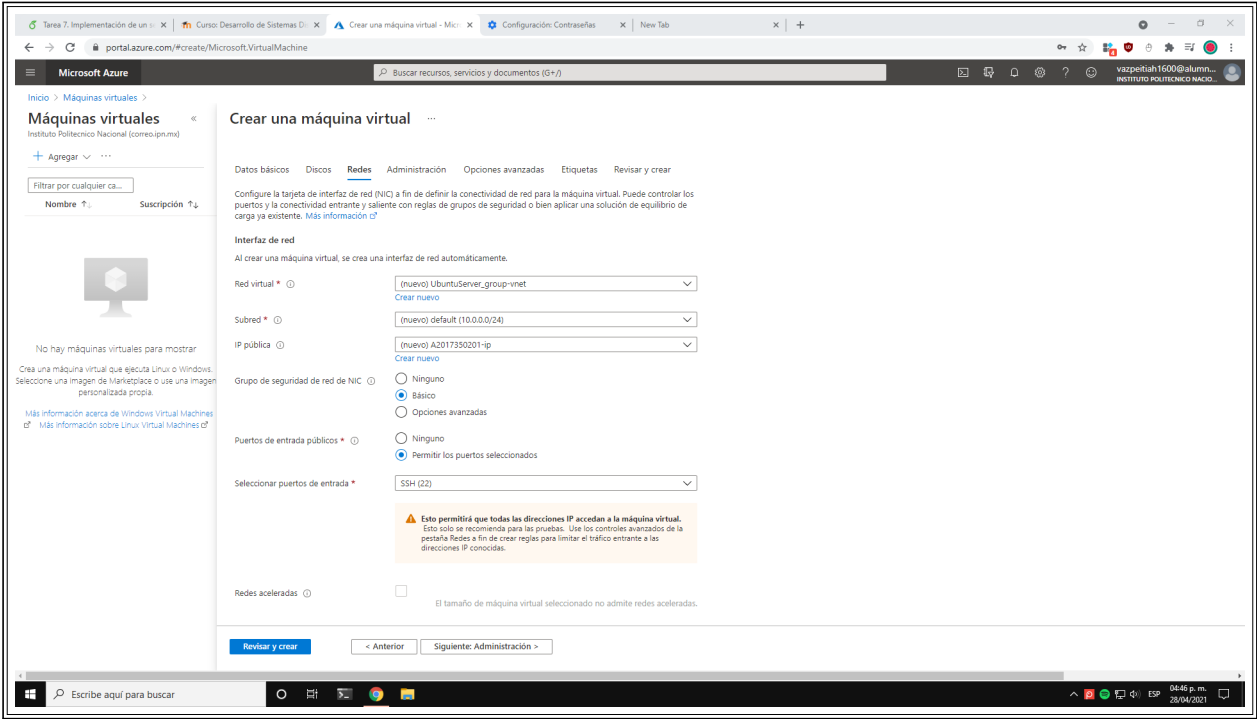


Figura 2.5: Creación de la máquina virtual: Redes

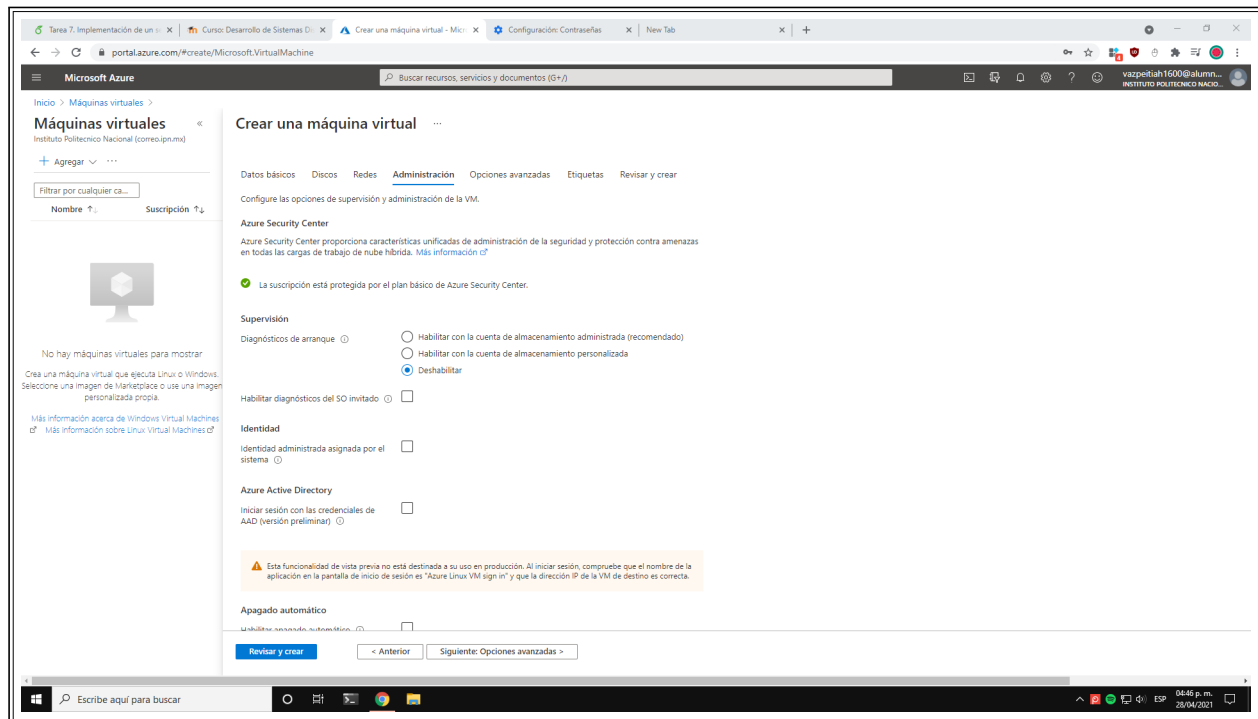


Figura 2.6: Creación de la máquina virtual: Administración

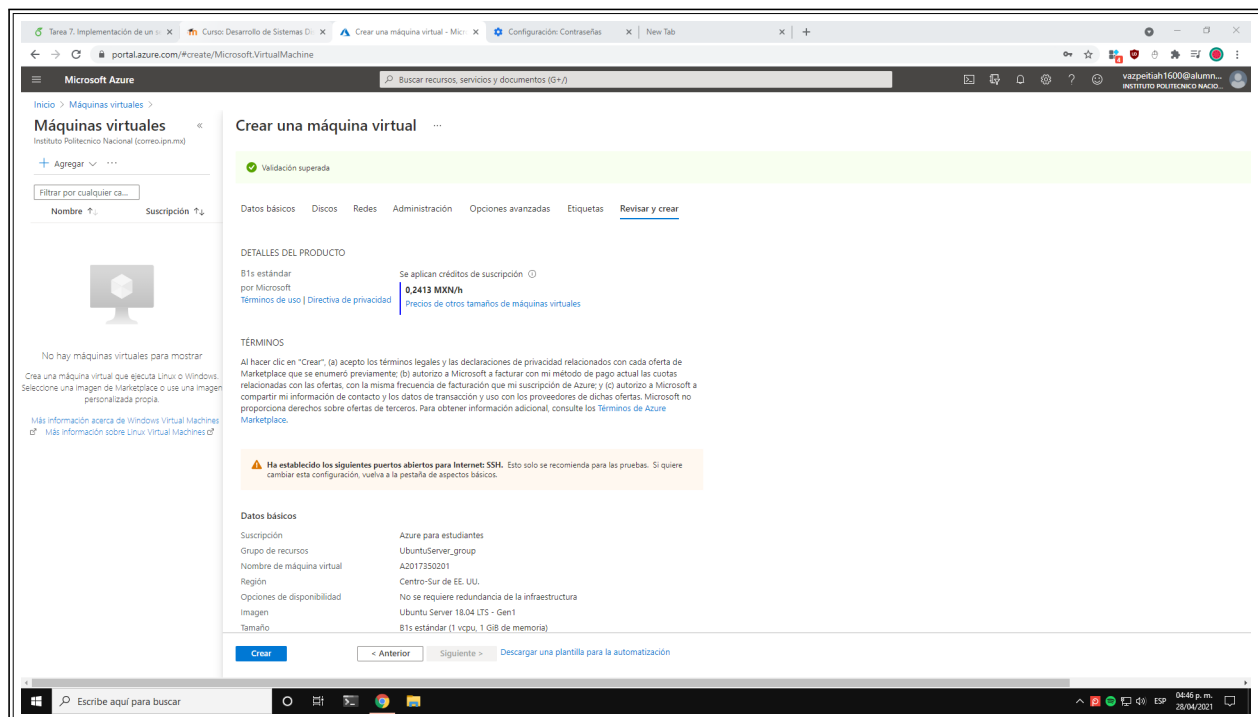


Figura 2.7: Creación de la máquina virtual: Validación superada

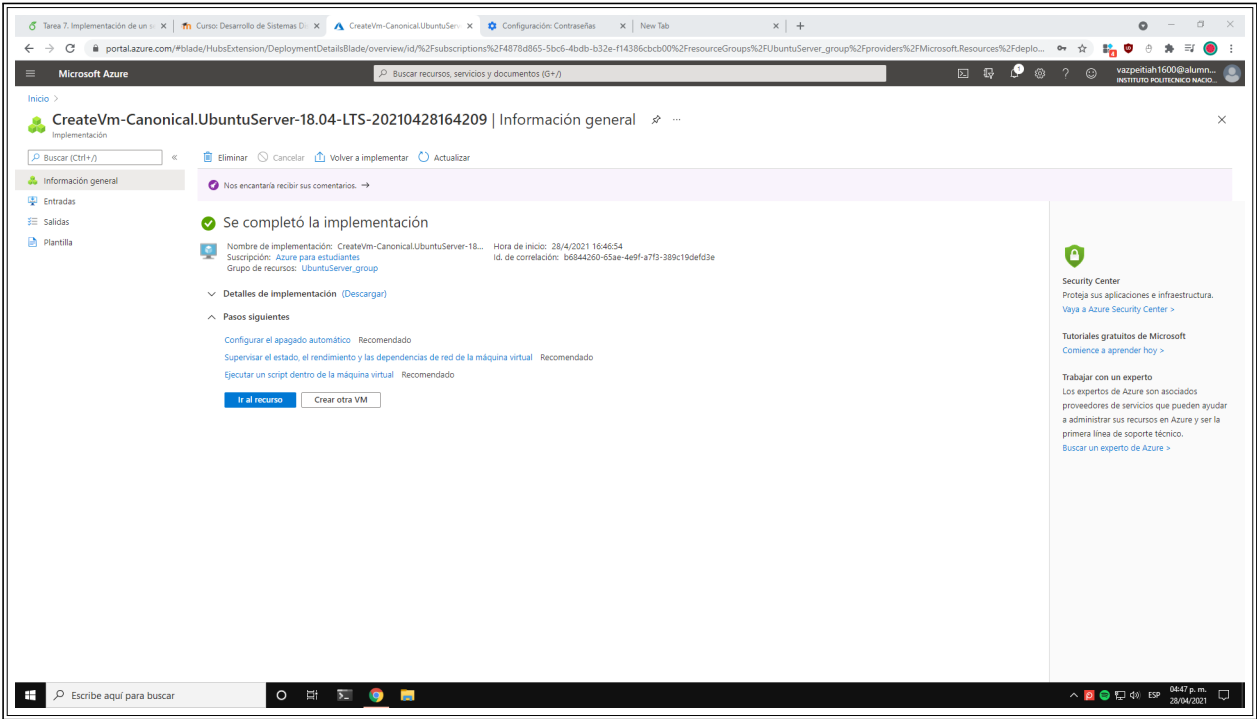


Figura 2.8: Creación de la máquina virtual: Implementación completada

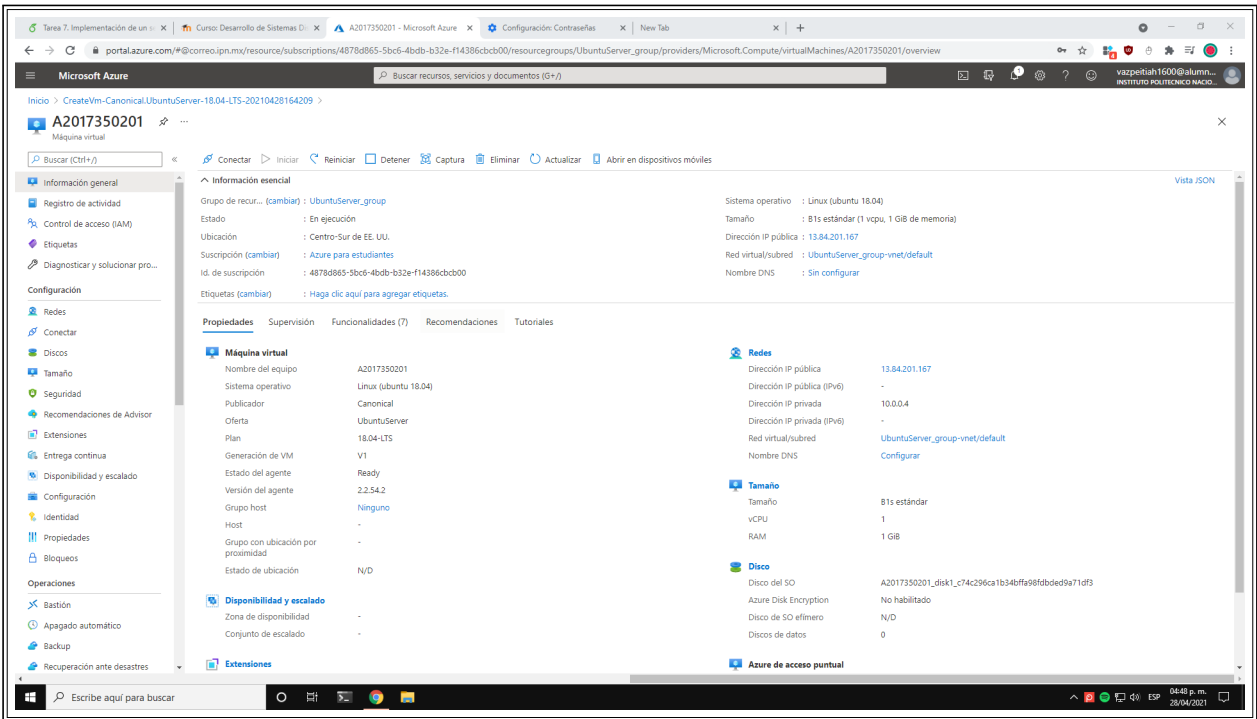


Figura 2.9: Creación de la máquina virtual: Panel de control de la máquina virtual

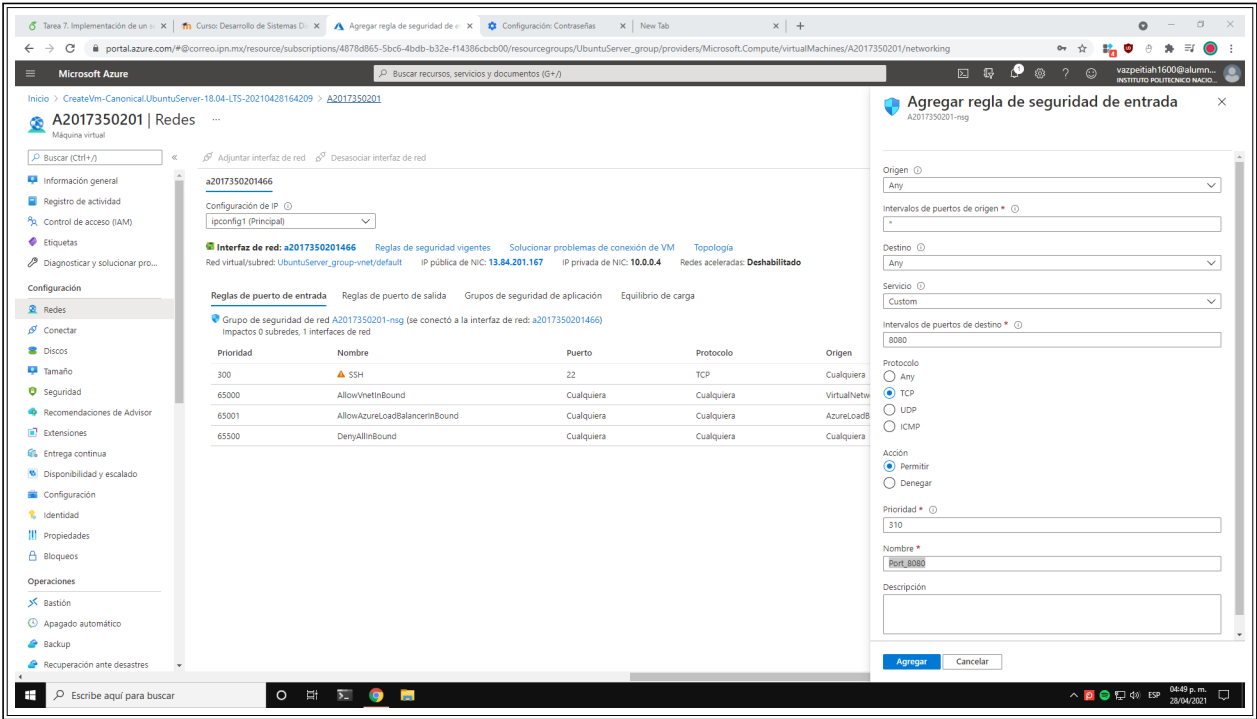


Figura 2.10: Creación de la máquina virtual: Configurando puerto 8080

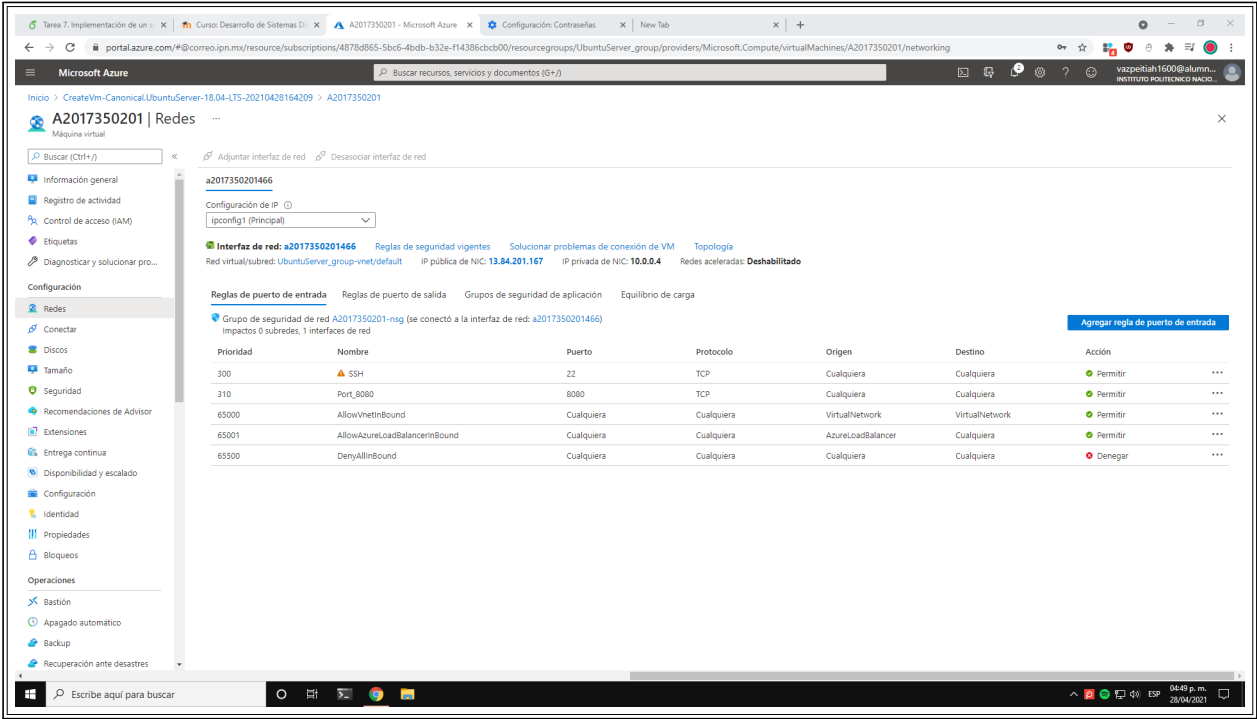


Figura 2.11: Creación de la máquina virtual: Puerto 8080 abierto para TCP

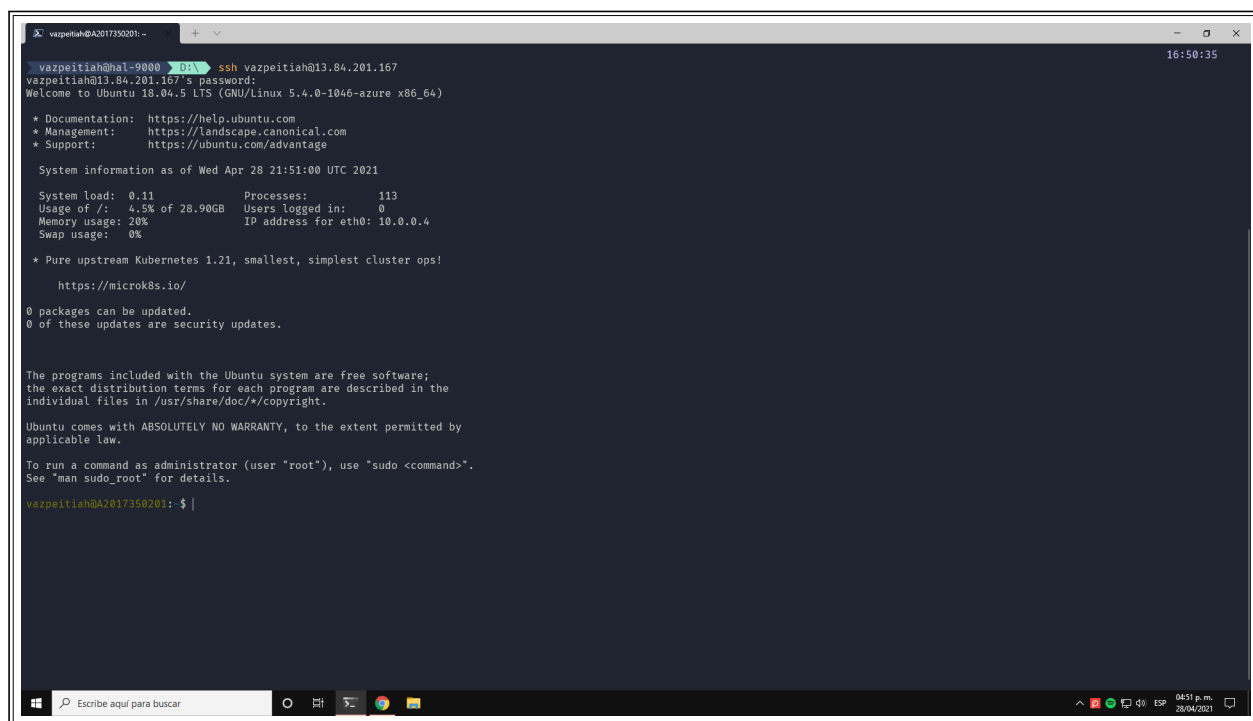


Figura 2.12: Creación de la máquina virtual: Conexión a través de ssh a la máquina virtual

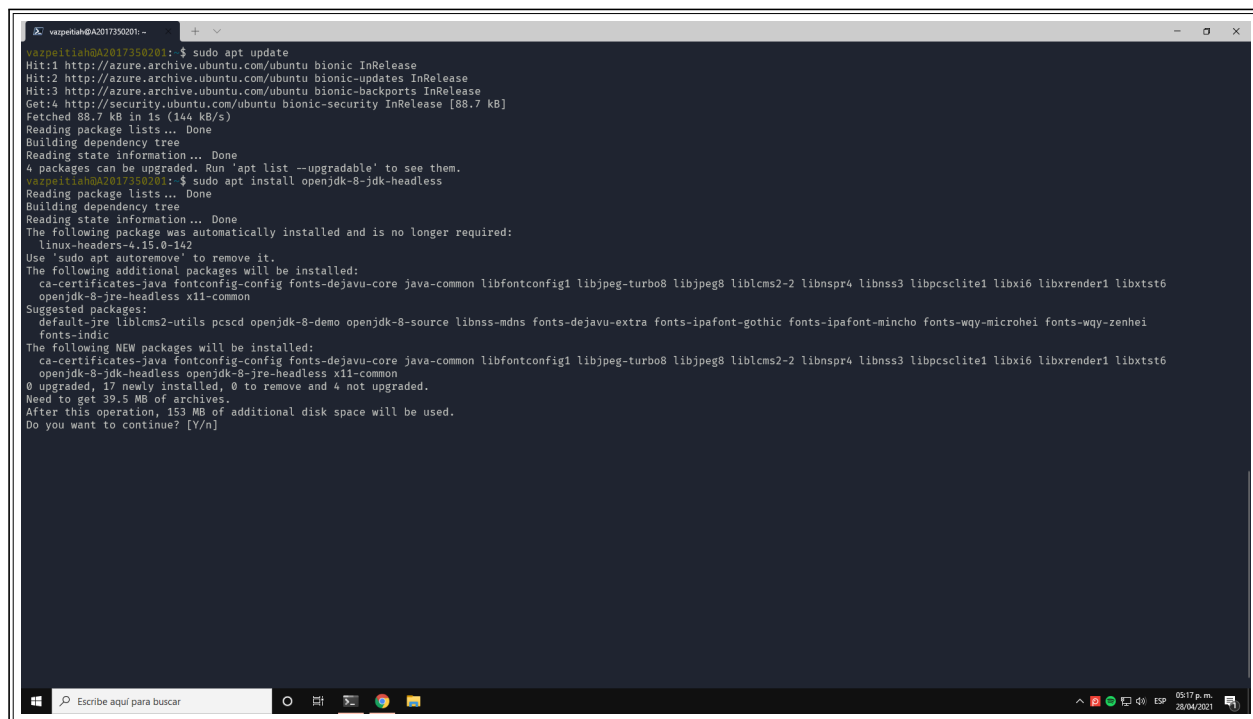


Figura 2.13: Creación de la máquina virtual: Actualizando repositorios y descargando Java Open JDK 8

2.2. Instalación de Tomcat con soporte REST

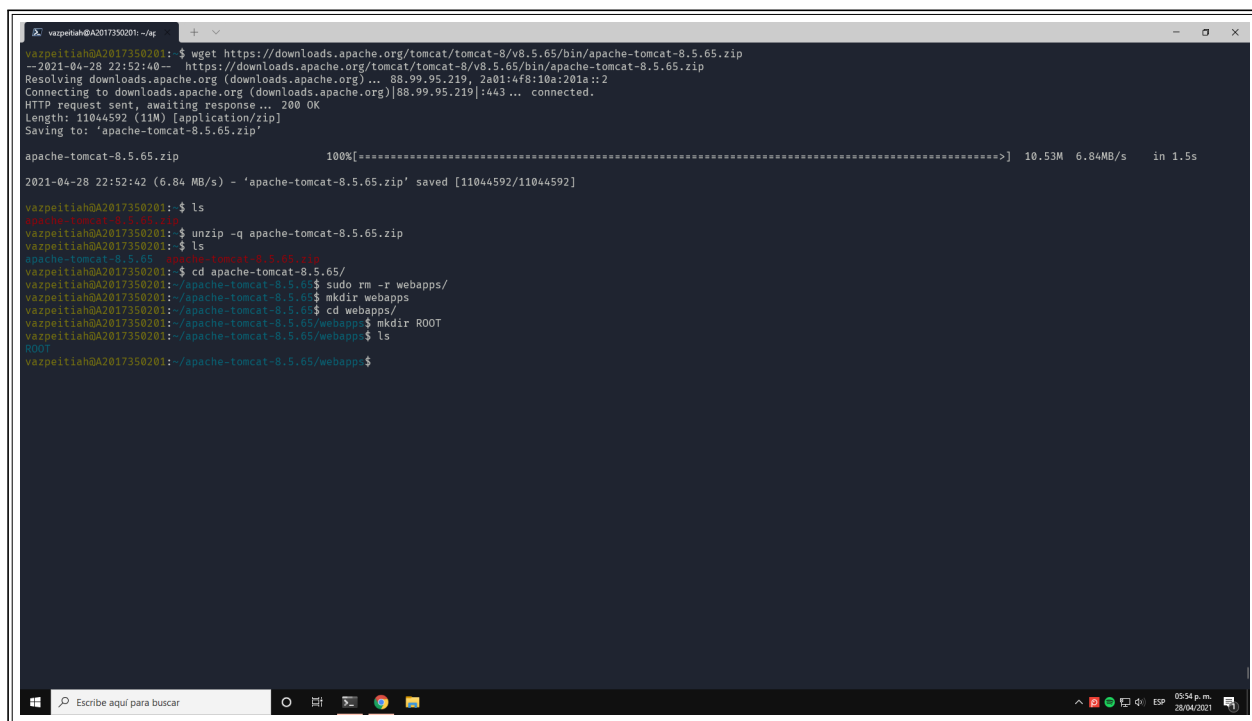


Figura 2.14: Instalación de Tomcat con soporte REST: Descargando Tomcat 8

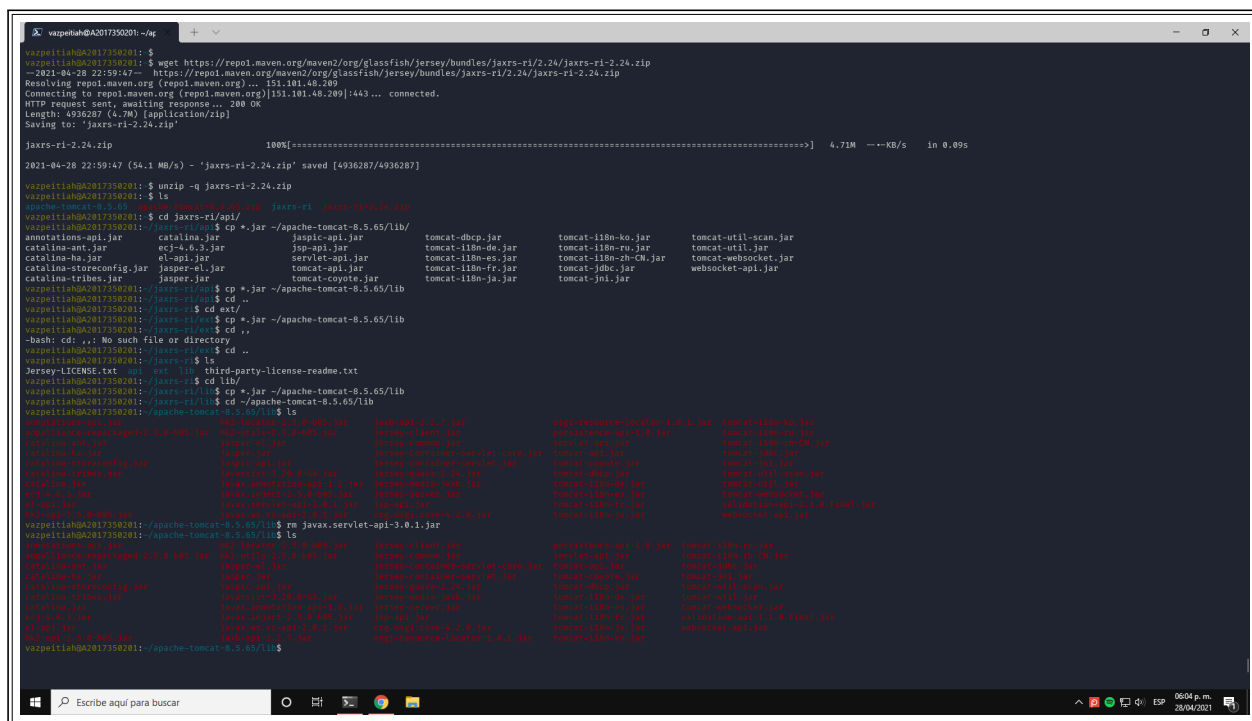
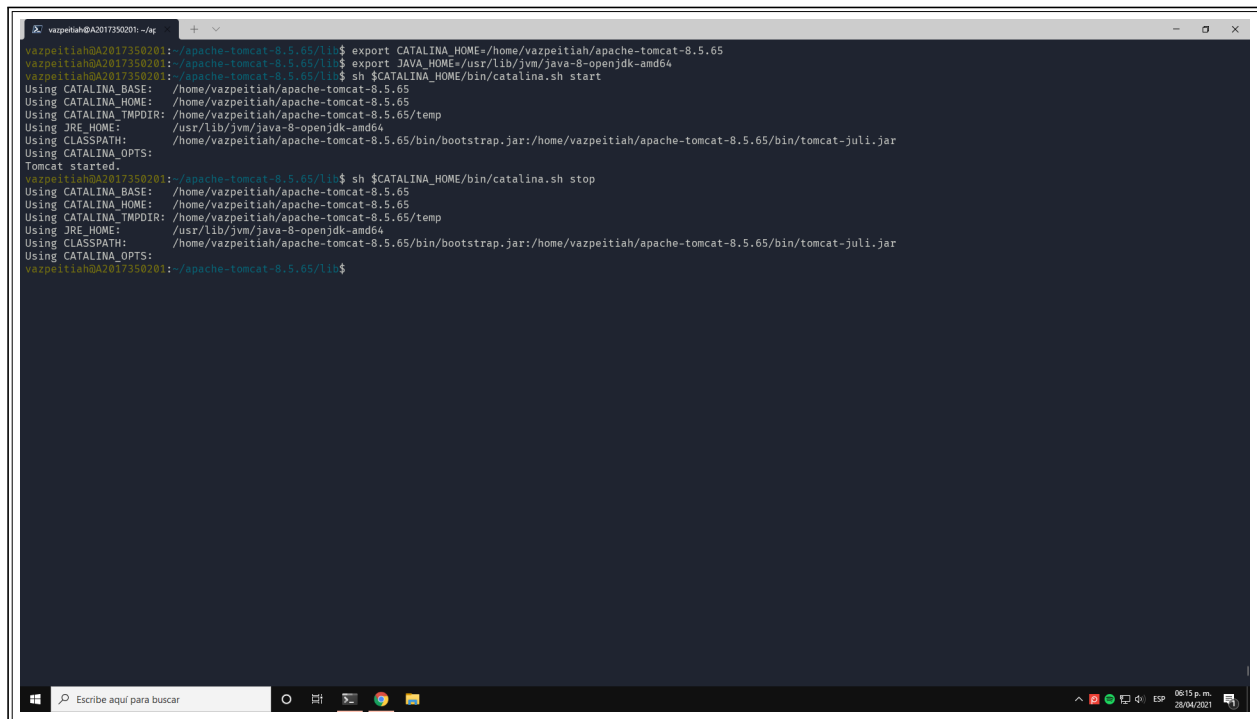


Figura 2.15: Instalación de Tomcat con soporte REST: Descargando la biblioteca "Jersey" para Tomcat





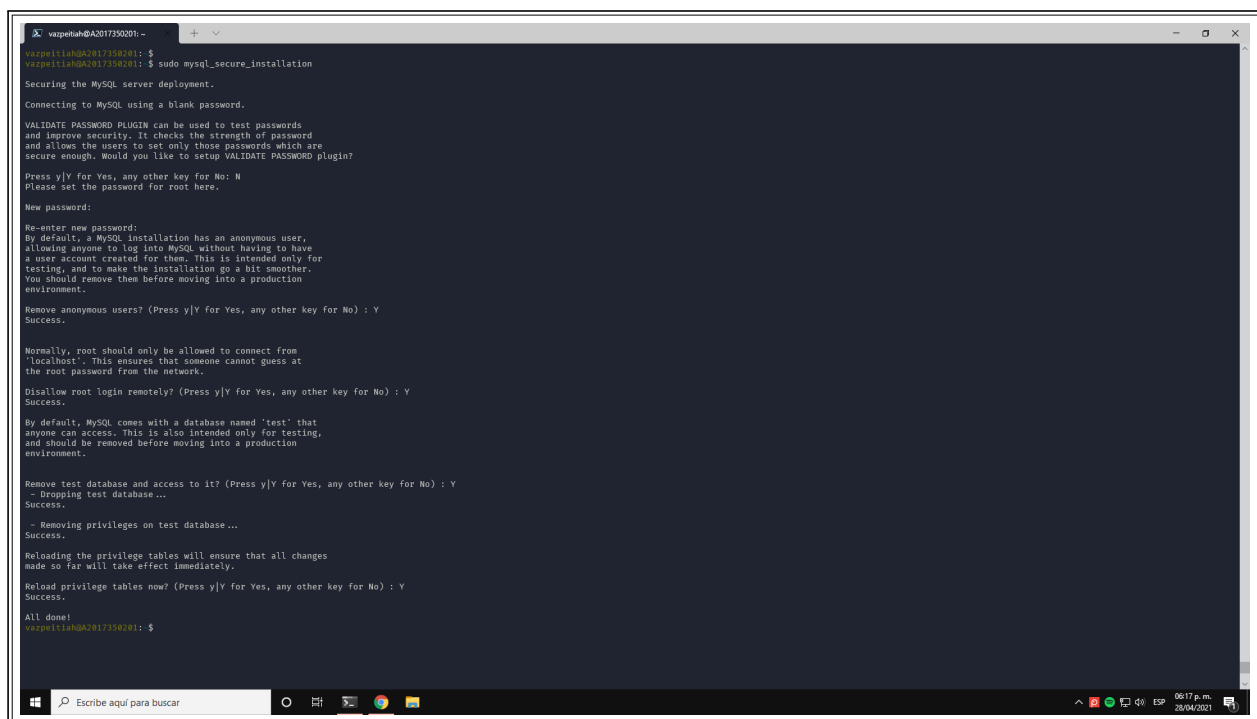
```

vazpeitiah@A2017350201: ~$ export CATALINA_HOME=/home/vazpeitiah/apache-tomcat-8.5.65
vazpeitiah@A2017350201: ~$ export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
vazpeitiah@A2017350201: ~$ sh $CATALINA_HOME/bin/catalina.sh start
Using CATALINA_BASE:   /home/vazpeitiah/apache-tomcat-8.5.65
Using CATALINA_HOME:   /home/vazpeitiah/apache-tomcat-8.5.65
Using CATALINA_TMPDIR: /home/vazpeitiah/apache-tomcat-8.5.65/temp
Using JRE_HOME:        /usr/lib/jvm/java-8-openjdk-amd64
Using CLASSPATH:        /home/vazpeitiah/apache-tomcat-8.5.65/bin/bootstrap.jar:/home/vazpeitiah/apache-tomcat-8.5.65/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
vazpeitiah@A2017350201: ~$ sh $CATALINA_HOME/bin/catalina.sh stop
Using CATALINA_BASE:   /home/vazpeitiah/apache-tomcat-8.5.65
Using CATALINA_HOME:   /home/vazpeitiah/apache-tomcat-8.5.65
Using CATALINA_TMPDIR: /home/vazpeitiah/apache-tomcat-8.5.65/temp
Using JRE_HOME:        /usr/lib/jvm/java-8-openjdk-amd64
Using CLASSPATH:        /home/vazpeitiah/apache-tomcat-8.5.65/bin/bootstrap.jar:/home/vazpeitiah/apache-tomcat-8.5.65/bin/tomcat-juli.jar
Using CATALINA_OPTS:
vazpeitiah@A2017350201: ~$

```

Figura 2.18: Instalación de Tomcat con soporte REST: Iniciar/detener el servidor Tomcat

2.3. Instalación y configuración de MySQL



```

vazpeitiah@A2017350201: ~$ mysql_secure_installation
Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD PLUGIN 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 plugin?

Press y|Y for Yes, any other key for No: N
Please set the password for root here.

New password:

Re-enter new password:
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.

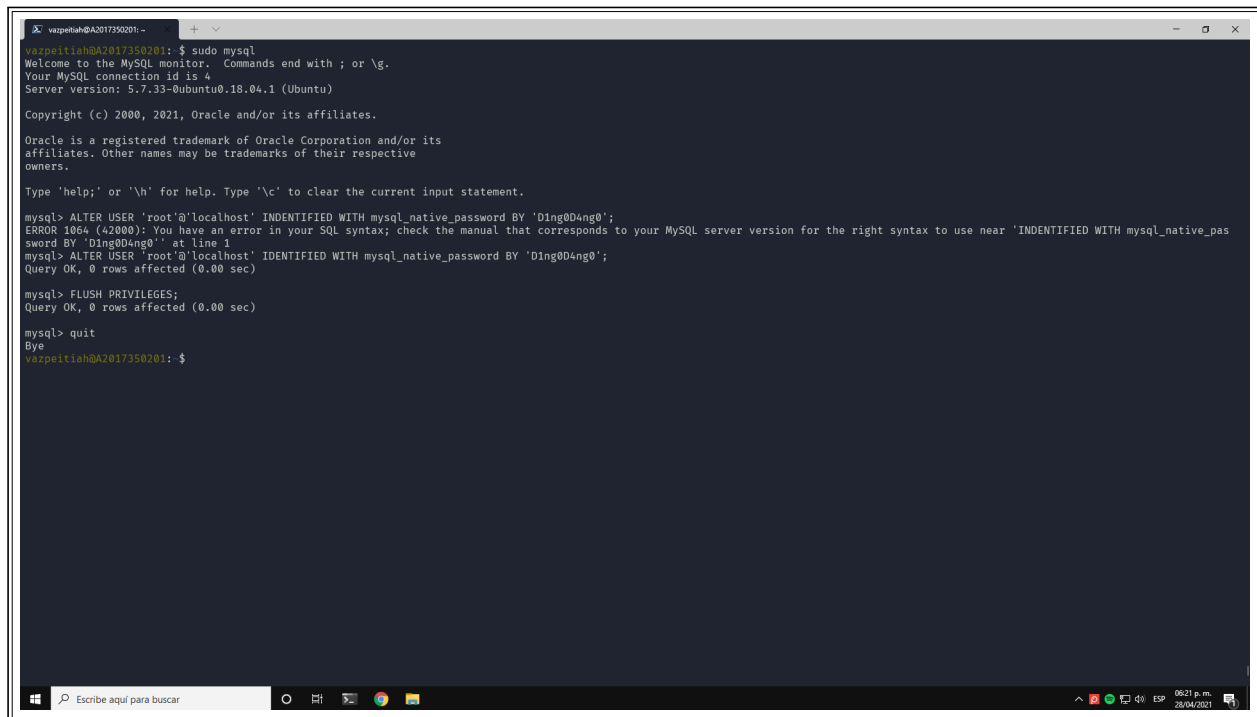
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

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

All done!
vazpeitiah@A2017350201: ~$

```

Figura 2.19: Instalación de MySQL: Instalación segura de MySQL

A terminal window titled 'vazpeitiah@A2017350201: ~' showing the execution of 'sudo mysql'. The MySQL monitor displays the root user's password being updated from 'Ding0D4ng0' to 'Ding0D4ng0'. The user then flushes privileges and quits the monitor.

```
vazpeitiah@A2017350201: ~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.33-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2021, 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.

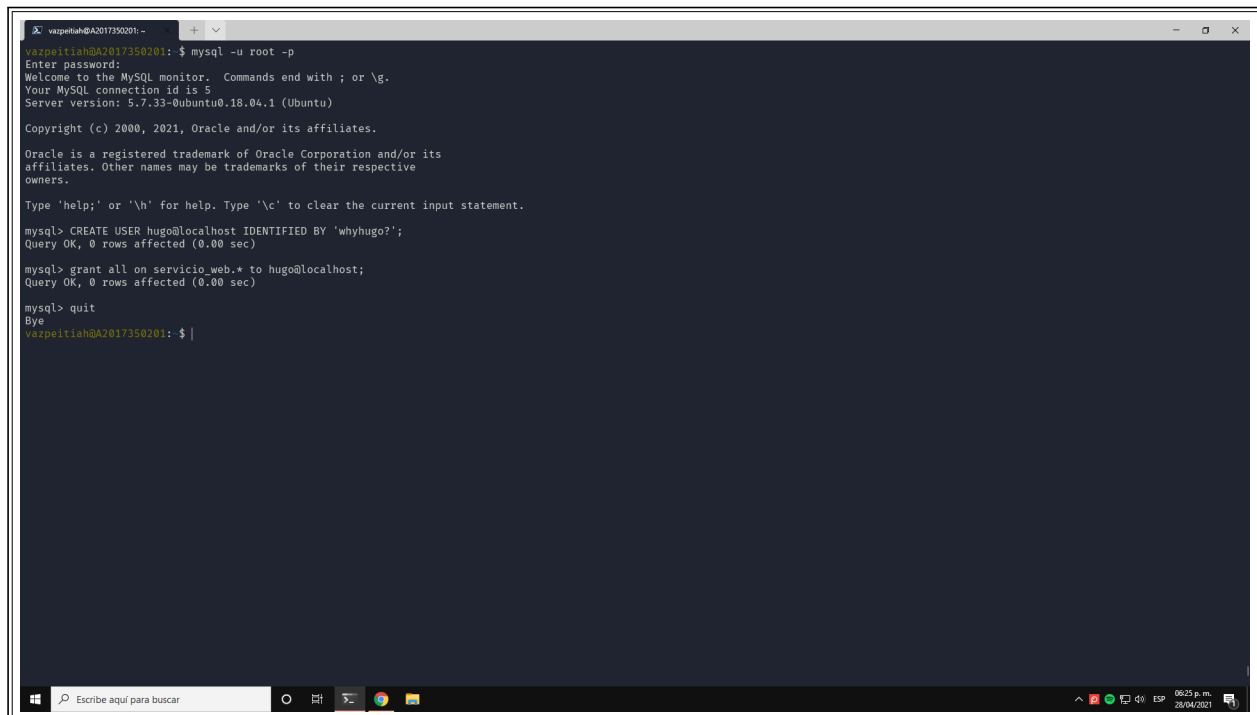
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'Ding0D4ng0';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'IDENTIFIED WITH mysql_native_password BY 'Ding0D4ng0'' at line 1
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'Ding0D4ng0';
Query OK, 0 rows affected (0.00 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
vazpeitiah@A2017350201: ~$
```

Figura 2.20: Instalación de MySQL: Actulizando la contraseña del usuario root

A terminal window titled 'vazpeitiah@A2017350201: ~' showing the execution of 'mysql -u root -p'. The MySQL monitor displays the creation of a new user 'hugo@localhost' with password 'whyhugo?'. The user then grants all privileges on the 'servicio_web.*' database to 'hugo@localhost' and quits the monitor.

```
vazpeitiah@A2017350201: ~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.33-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2021, 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.

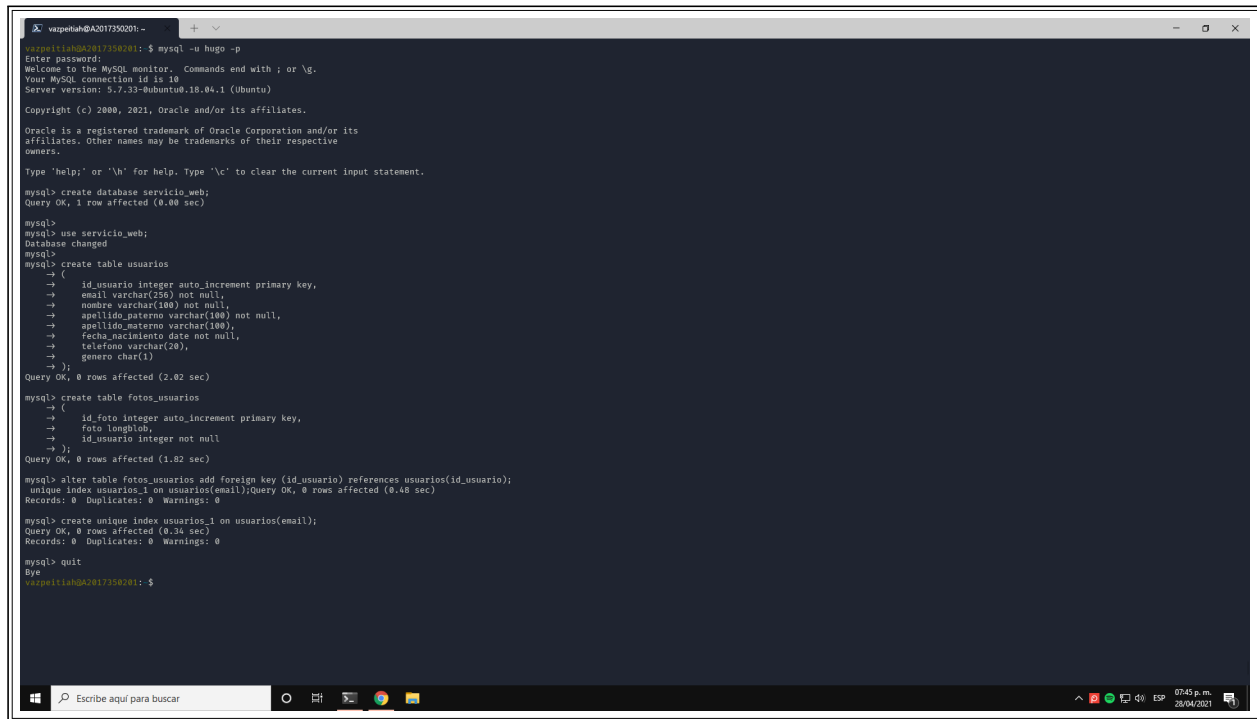
mysql> CREATE USER hugo@localhost IDENTIFIED BY 'whyhugo?';
Query OK, 0 rows affected (0.00 sec)

mysql> grant all on servicio_web.* to hugo@localhost;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
vazpeitiah@A2017350201: ~$
```

Figura 2.21: Instalación de MySQL: Creando un nuevo usuario para nuestra base de datos

2.4. Creación de la base de datos en MySQL



```
vazpeitah@A2017350201:~$ mysql -u hugo -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.7.33-Debian0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2021, 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 (0.00 sec)

mysql> use servicio_web;
Database changed
mysql> create table usuarios
  (
    → id_usuario integer auto_increment primary key,
    → email varchar(256) not null,
    → nombre varchar(100) not null,
    → apellido_paterno varchar(100) not null,
    → apellido_materno varchar(100),
    → fecha_nacimiento date not null,
    → telefono varchar(20),
    → genero char(1)
  );
Query OK, 0 rows affected (2.02 sec)

mysql> create table fotos_usuarios
  (
    → id_foto integer auto_increment primary key,
    → foto longblob,
    → id_usuario integer not null
  );
Query OK, 0 rows affected (1.82 sec)

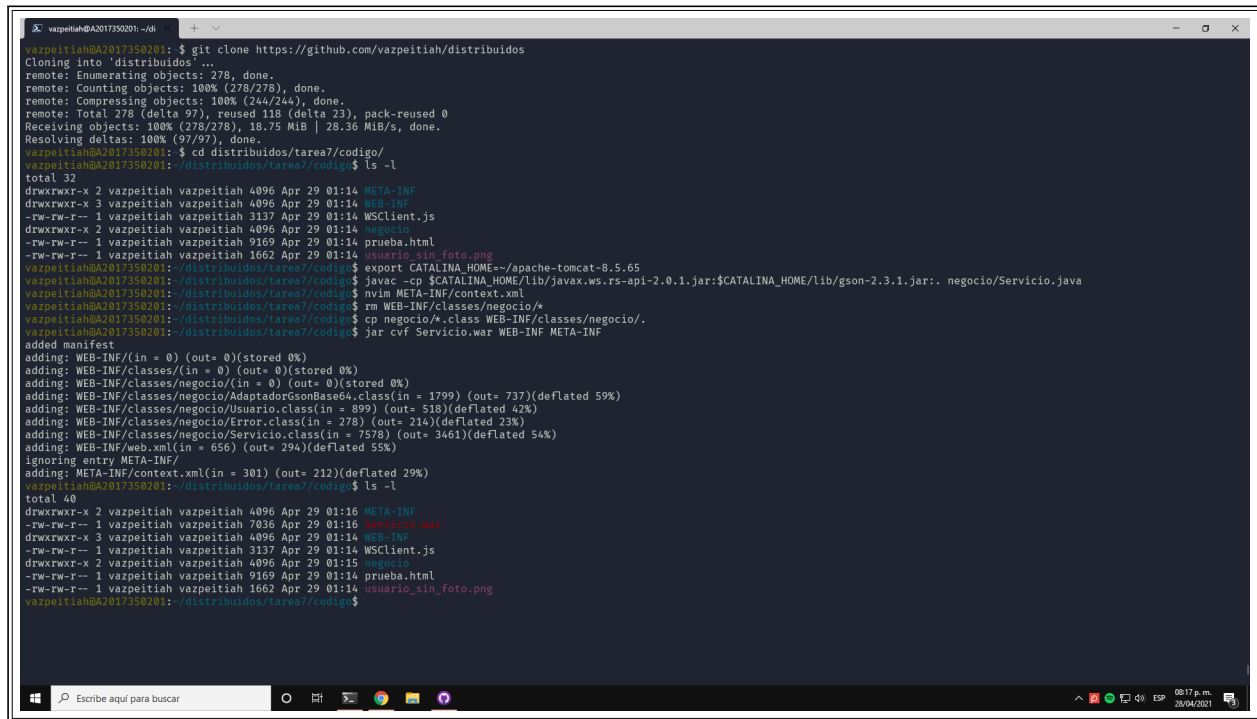
mysql> alter table fotos_usuarios add foreign key (id_usuario) references usuarios(id_usuario);
unique index usuarios_1 on usuarios(email);Query OK, 0 rows affected (0.48 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> create unique index usuarios_1 on usuarios(email);
Query OK, 0 rows affected (0.34 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> quit
Bye
vazpeitah@A2017350201:~$
```

Figura 2.22: Creación de la base de datos en MySQL: Creando la base de datos de la aplicación

2.5. Compilar, empacar y desplegar el servicio web

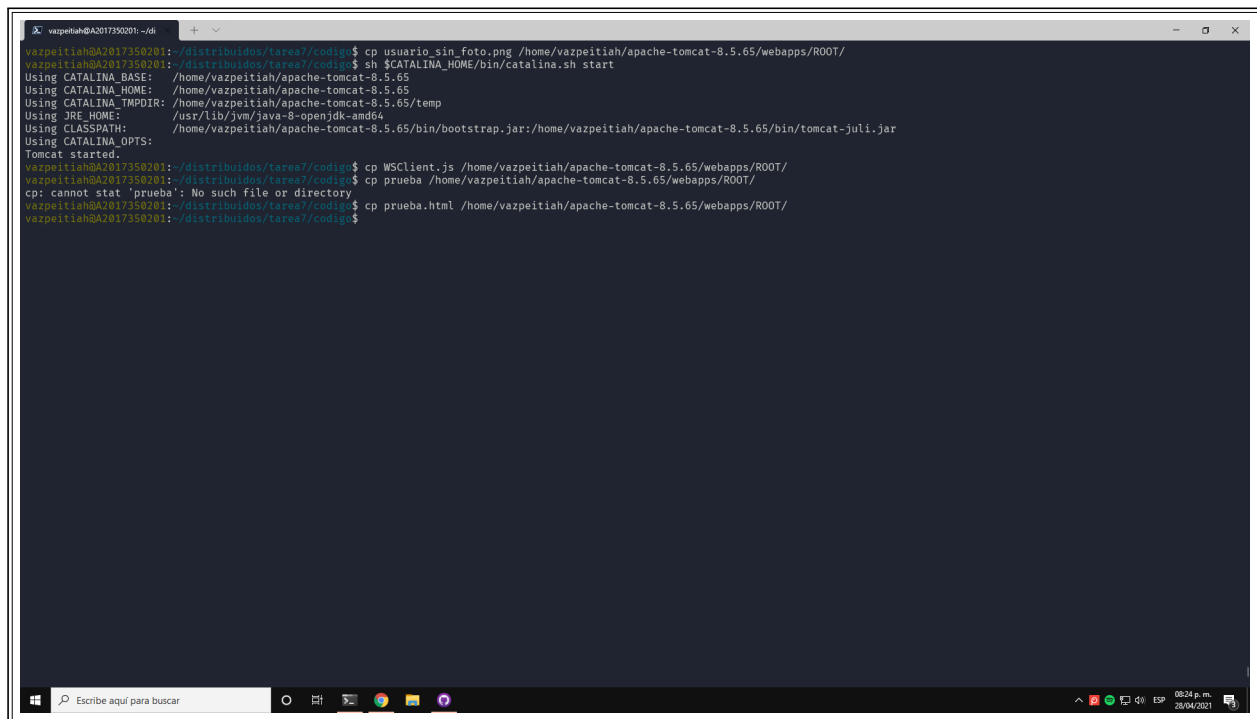


```

vazpeitiah@A2017350201: ~$ git clone https://github.com/vazpeitiah/distribuidos
Cloning into 'distribuidos' ...
remote: Enumerating objects: 278, done.
remote: Counting objects: 100% (278/278), done.
remote: Compressing objects: 100% (244/244), done.
remote: Total 278 (delta 97), reused 118 (delta 23), pack-reused 0
Receiving objects: 100% (278/278), 18.75 MiB | 28.36 MiB/s, done.
Resolving deltas: 100% (97/97), done.
vazpeitiah@A2017350201: ~$ cd distribuidos/tarea7/codigo/
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ ls -l
total 32
drwxrwxr-x 2 vazpeitiah vazpeitiah 4096 Apr 29 01:14 META-INF
drwxrwxr-x 3 vazpeitiah vazpeitiah 4096 Apr 29 01:14 WEB-INF
-rw-rw-r-- 1 vazpeitiah vazpeitiah 3137 Apr 29 01:14 WSClient.js
drwxrwxr-x 2 vazpeitiah vazpeitiah 4096 Apr 29 01:14 usuario
-rw-rw-r-- 1 vazpeitiah vazpeitiah 9169 Apr 29 01:14 prueba.html
-rw-rw-r-- 1 vazpeitiah vazpeitiah 1662 Apr 29 01:14 usuario_sin_foto.png
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ export CATALINA_HOME=/apache-tomcat-8.5.65
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ javac -cp $CATALINA_HOME/lib/javax.ws.rs-api-2.0.1.jar:$CATALINA_HOME/lib/gson-2.3.1.jar:. negocio/Servicio.java
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ mvn META-INF/context.xml
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ rm WEB-INF/classes/negocio/*
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ cp negocio/*.class WEB-INF/classes/negocio/
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ jar cvf Servicio.war WEB-INF META-INF
added manifest
adding: WEB-INF/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/negocio/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/negocio/AdaptadorGsonBase64.class(in = 1799) (out= 737)(deflated 59%)
adding: WEB-INF/classes/negocio/Usuario.class(in = 899) (out= 518)(deflated 42%)
adding: WEB-INF/classes/negocio/Error.class(in = 278) (out= 214)(deflated 23%)
adding: WEB-INF/classes/negocio/Servicio.class(in = 7576) (out= 3461)(deflated 54%)
adding: WEB-INF/web.xml(in = 656) (out= 294)(deflated 55%)
ignoring entry META-INF/
adding: META-INF/context.xml(in = 301) (out= 212)(deflated 29%)
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ ls -l
total 40
drwxrwxr-x 2 vazpeitiah vazpeitiah 4096 Apr 29 01:16 META-INF
-rw-rw-r-- 1 vazpeitiah vazpeitiah 7036 Apr 29 01:16 Servicio.war
drwxrwxr-x 3 vazpeitiah vazpeitiah 4096 Apr 29 01:14 WEB-INF
-rw-rw-r-- 1 vazpeitiah vazpeitiah 3137 Apr 29 01:14 WSClient.js
drwxrwxr-x 2 vazpeitiah vazpeitiah 4096 Apr 29 01:15 usuario
-rw-rw-r-- 1 vazpeitiah vazpeitiah 9169 Apr 29 01:14 prueba.html
-rw-rw-r-- 1 vazpeitiah vazpeitiah 1662 Apr 29 01:14 usuario_sin_foto.png
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$

```

Figura 2.23: Compilar, empacar y desplegar el servicio web: Captura 01



```

vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ cp usuario_sin_foto.png /home/vazpeitiah/apache-tomcat-8.5.65/webapps/ROOT/
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ sh $CATALINA_HOME/bin/catalina.sh start
Using CATALINA_BASE:   /home/vazpeitiah/apache-tomcat-8.5.65
Using CATALINA_HOME:   /home/vazpeitiah/apache-tomcat-8.5.65
Using CATALINA_TMPDIR: /home/vazpeitiah/apache-tomcat-8.5.65/temp
Using JRE_HOME:         /usr/lib/jvm/java-8-openjdk-amd64
Using CLASSPATH:        /home/vazpeitiah/apache-tomcat-8.5.65/bin/bootstrap.jar:/home/vazpeitiah/apache-tomcat-8.5.65/bin/tomcat-juli.jar
Tomcat started.
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ cp WSClient.js /home/vazpeitiah/apache-tomcat-8.5.65/webapps/ROOT/
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ cp prueba /home/vazpeitiah/apache-tomcat-8.5.65/webapps/ROOT/
cp: cannot stat 'prueba': No such file or directory
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$ cp prueba.html /home/vazpeitiah/apache-tomcat-8.5.65/webapps/ROOT/
vazpeitiah@A2017350201: ~/distribuidos/tarea7/codigo$

```

Figura 2.24: Compilar, empacar y desplegar el servicio web: Captura 02

2.6. Probar el servicio web utilizando HTML-Javascript

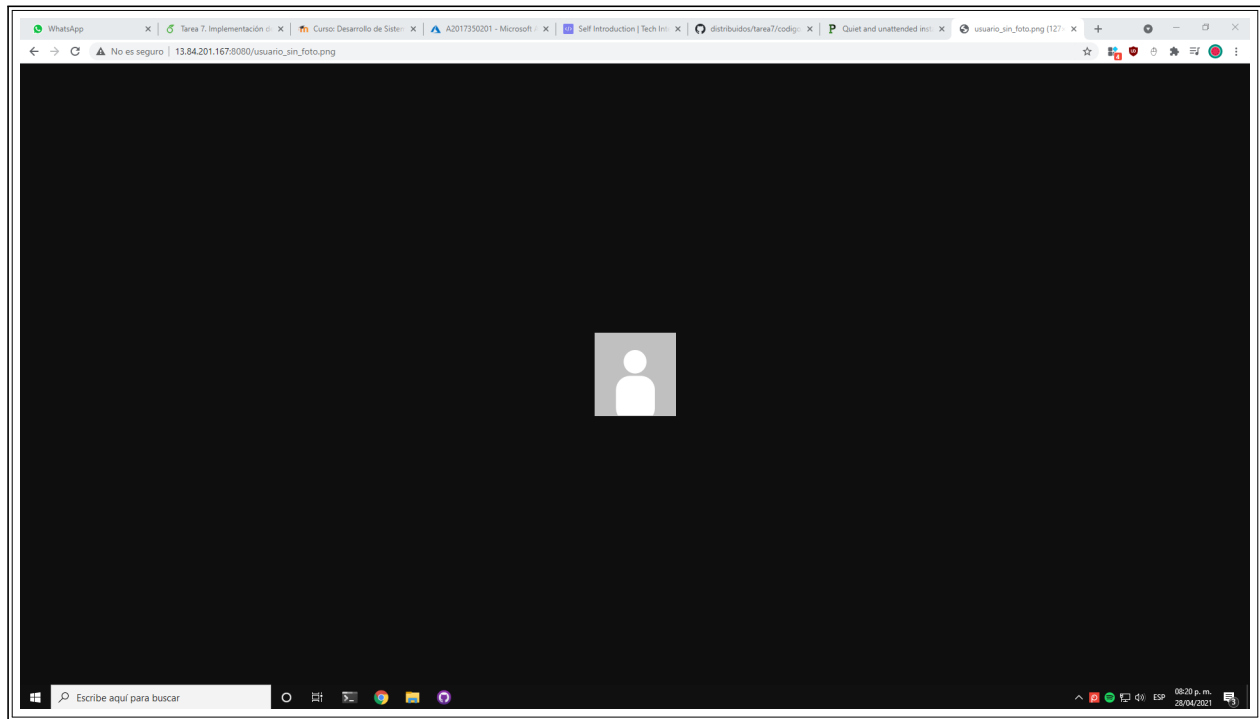


Figura 2.25: Probar el servicio web utilizando HTML-Javascript: Probando imagen almacenada en ROOT

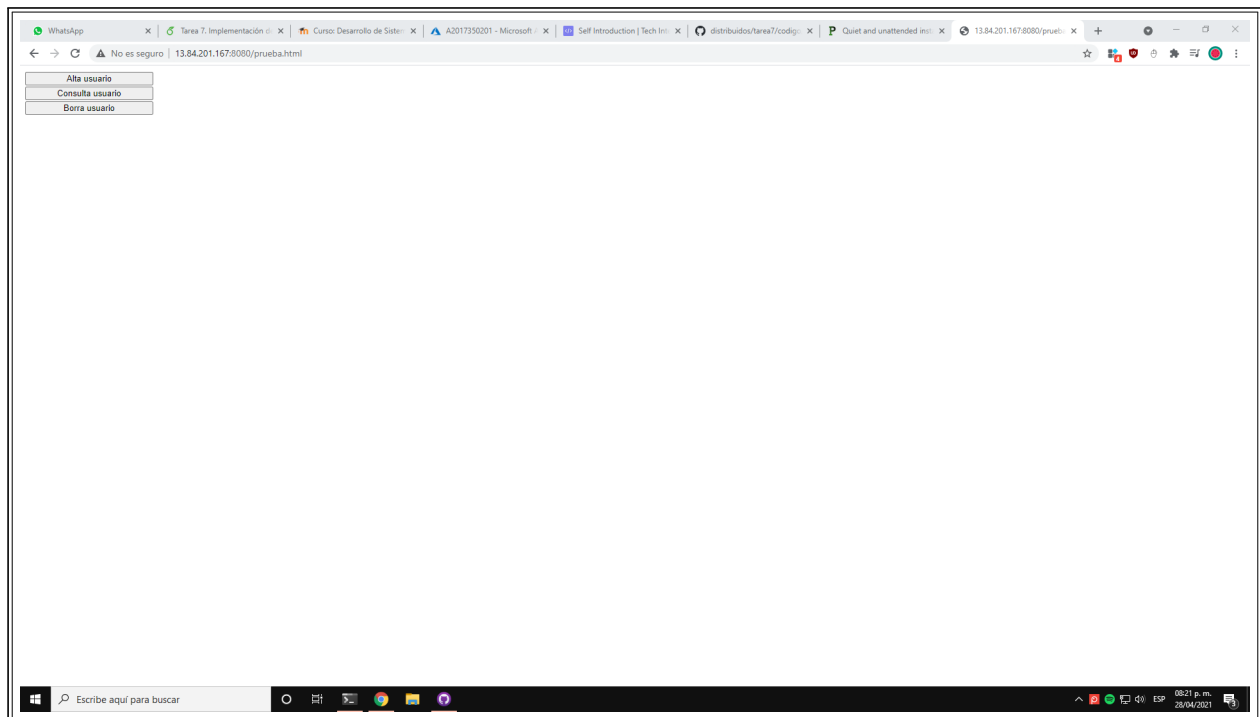


Figura 2.26: Probar el servicio web utilizando HTML-Javascript: Menú de la aplicación

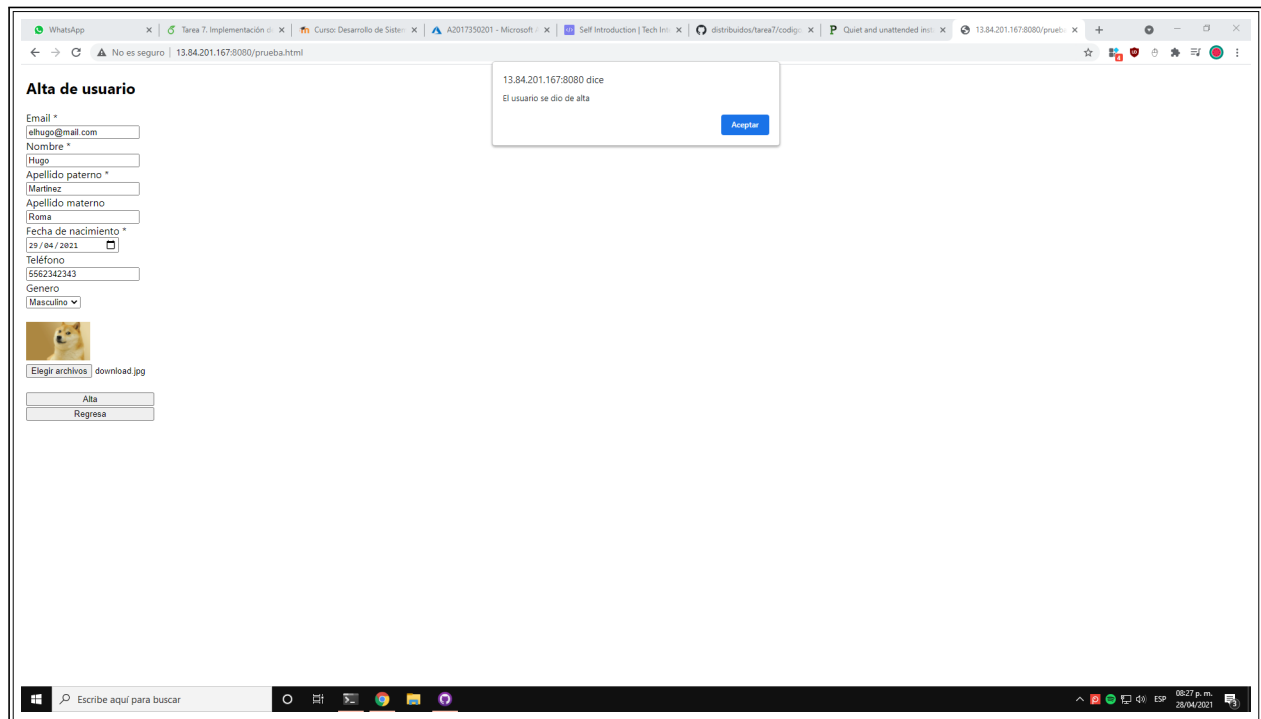


Figura 2.27: Probar el servicio web utilizando HTML-Javascript: Dando de alta a un nuevo usuario

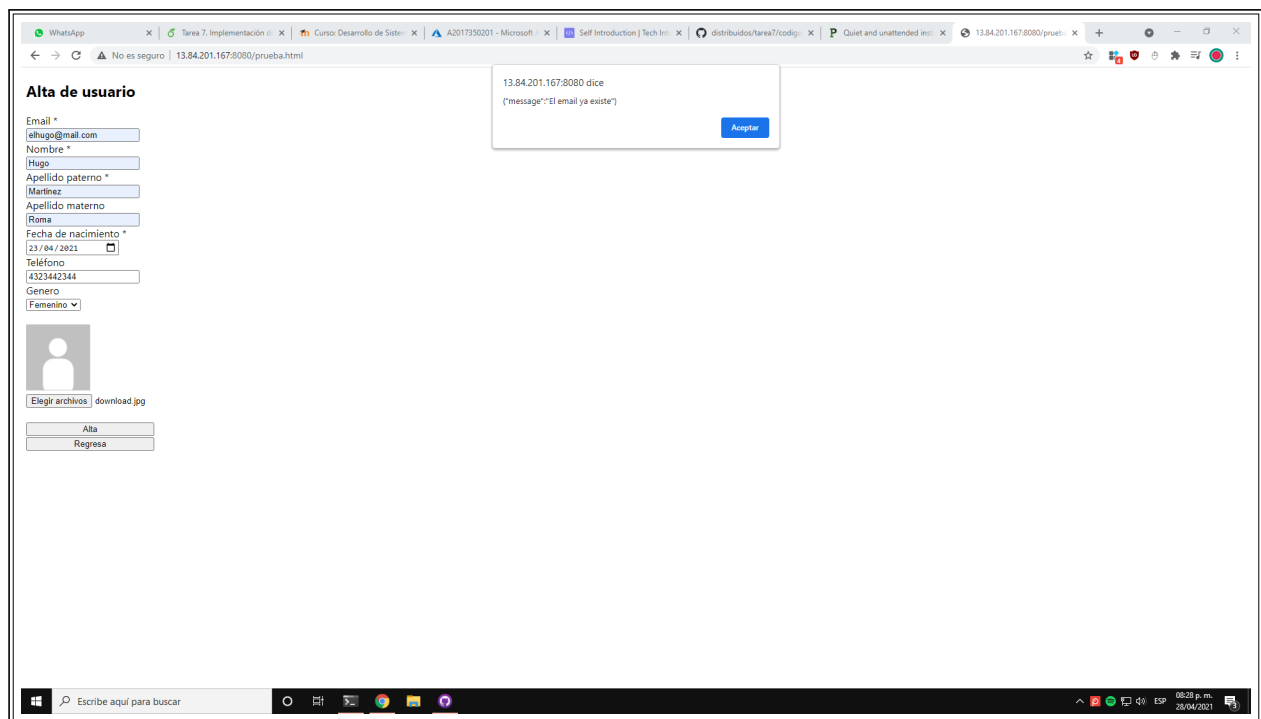


Figura 2.28: Probar el servicio web utilizando HTML-Javascript: Intentando ingresar un usuario con email duplicado

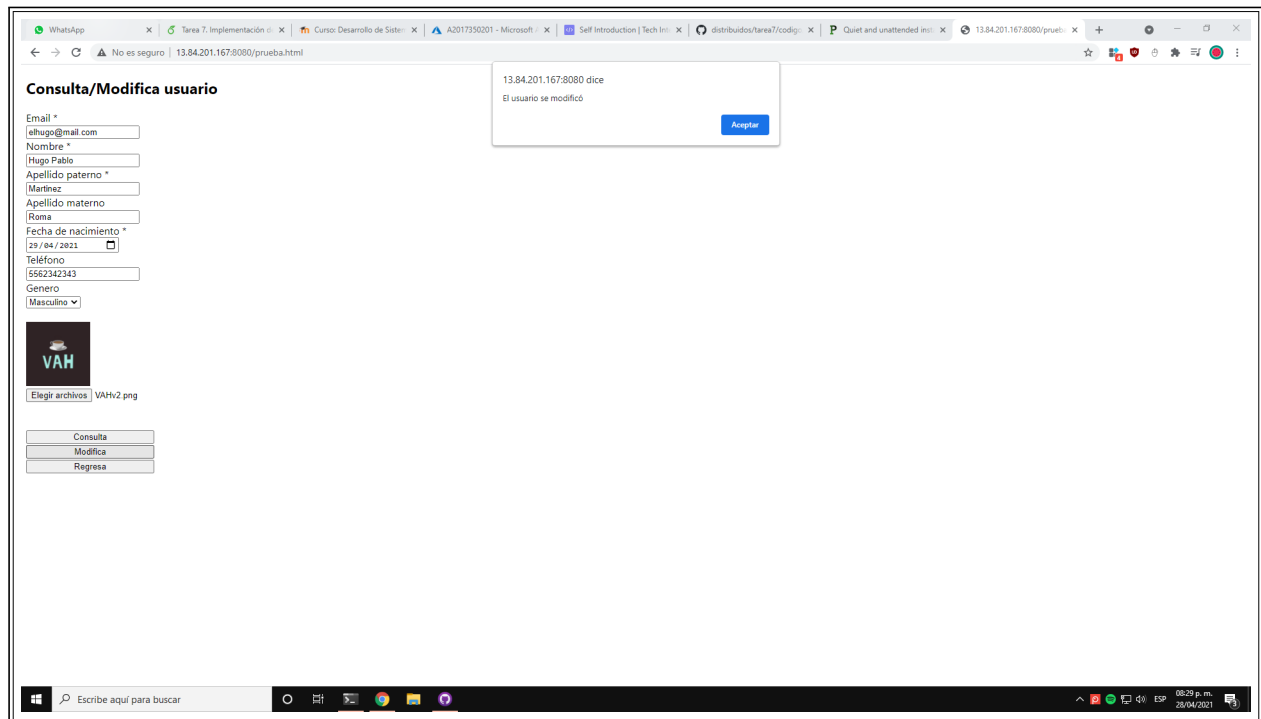


Figura 2.29: Probar el servicio web utilizando HTML-Javascript: Actualizando información del usuario

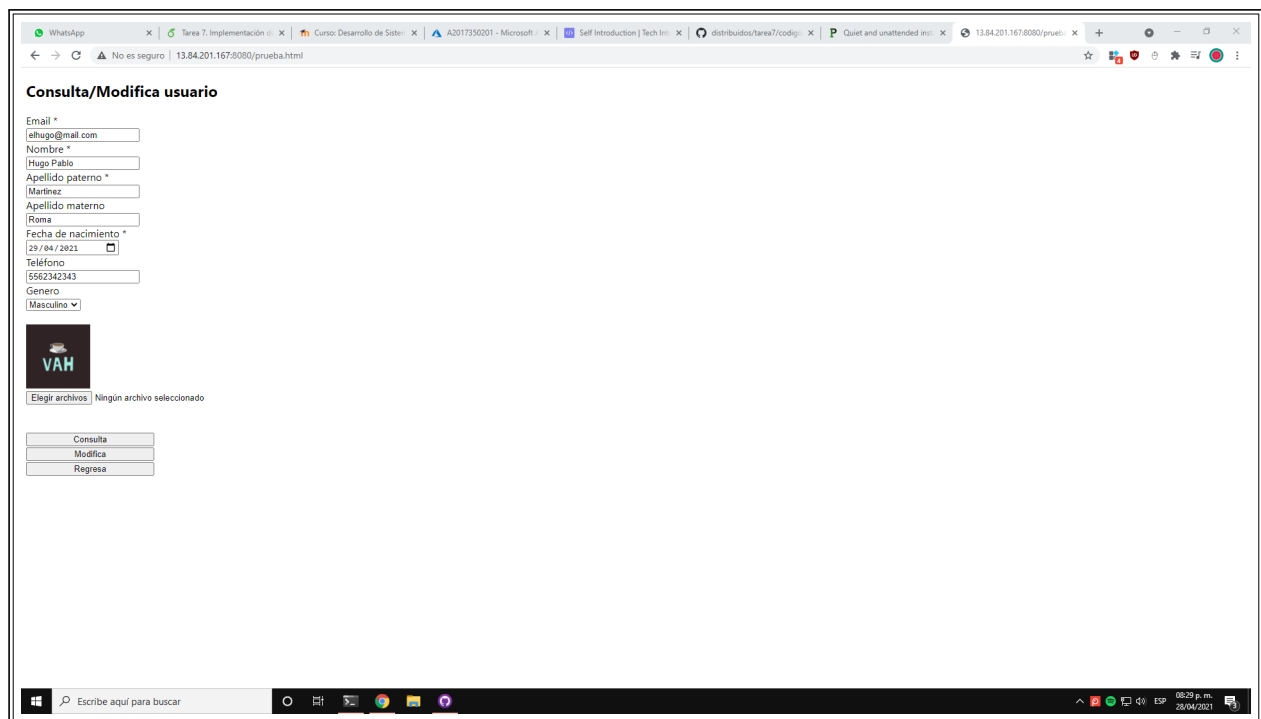


Figura 2.30: Probar el servicio web utilizando HTML-Javascript: Consultado información de usuario

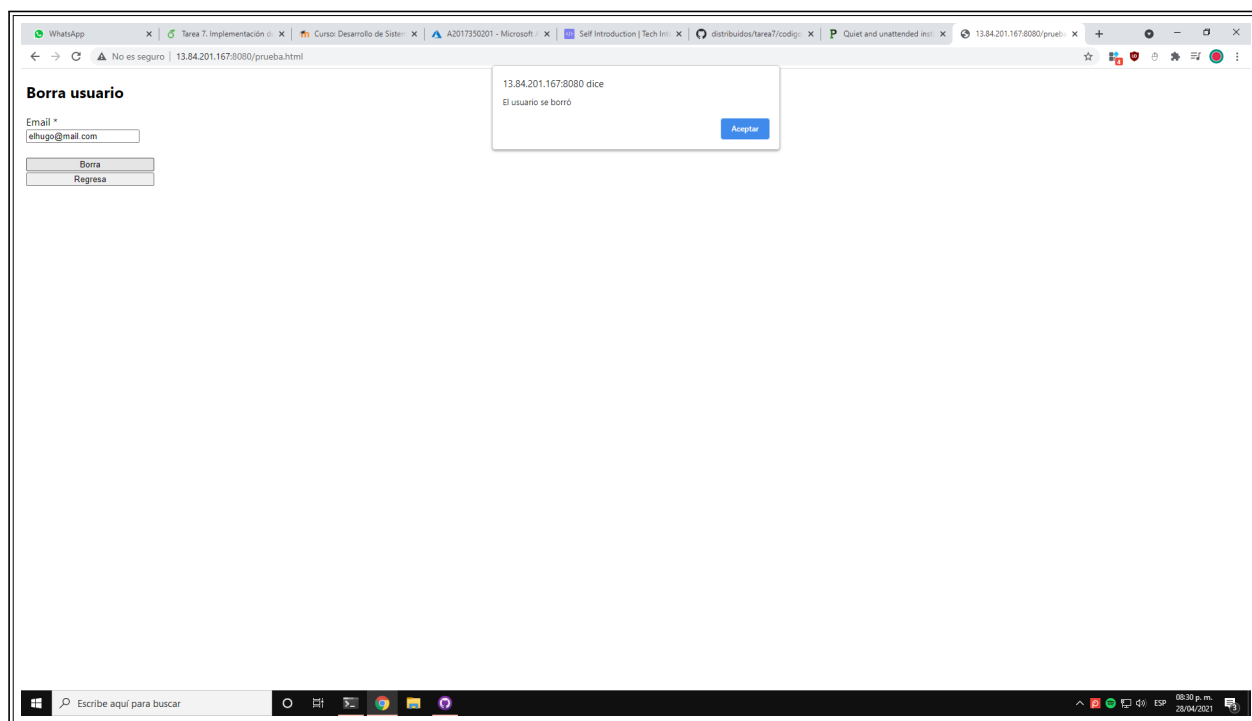


Figura 2.31: Probar el servicio web utilizando HTML-Javascript: Eliminando Usuario

2.7. Probar el servicio web desde un dispositivo móvil

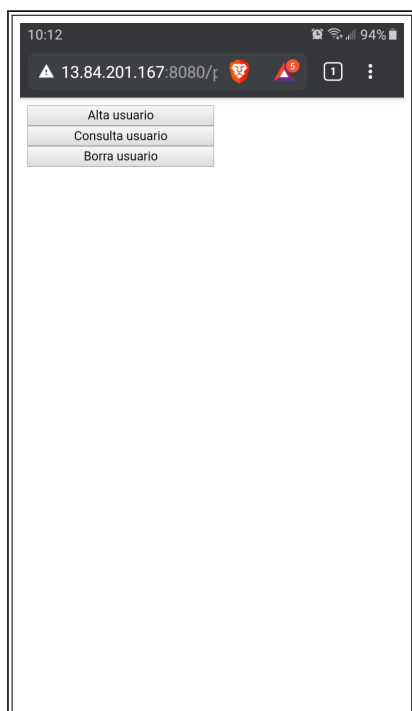


Figura 2.32: Dispositivo móvil: Menú principal de la aplicación

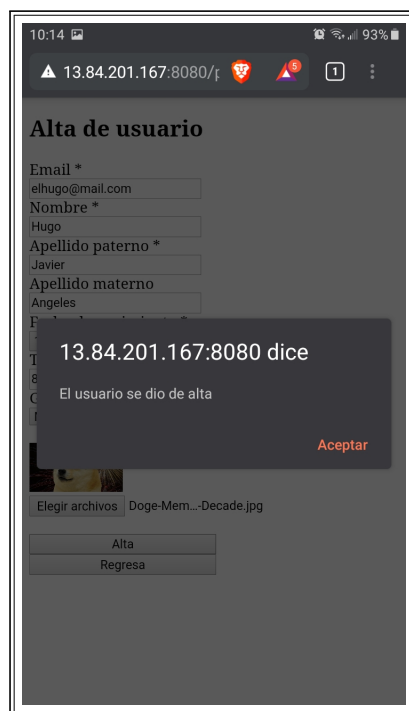


Figura 2.33: Dispositivo móvil: Creando un nuevo usuario

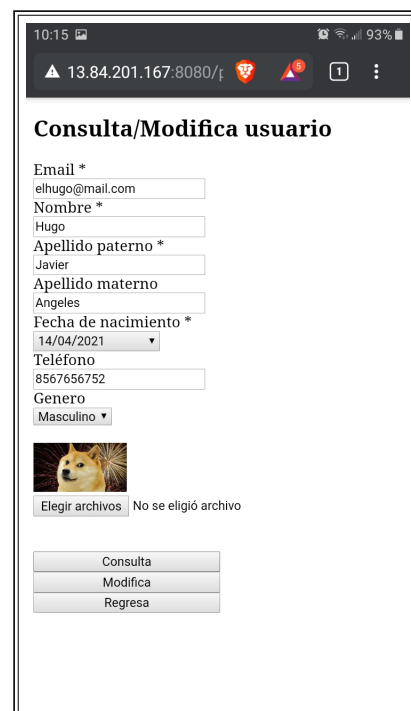


Figura 2.34: Dispositivo móvil: Consultando información de usuario



Figura 2.35: Dispositivo móvil: Actualizar información del usuario



Figura 2.36: Dispositivo móvil: Eliminar usuario

3.1. Conclusiones finales

En esta práctica observamos el funcionamiento de una aplicación REST, y su comportamiento básico. El desplegar la aplicación en una máquina virtual y a su vez montarlo en un servidor web Tomcat, nos hace apreciar el sistema distribuido como si fuese una sola aplicación corriendo en una consola de Linux. Se aprecia correctamente el uso de un back-end por parte de la aplicación, haciendo uso de MySQL como manejador de base de datos y compilando el programa con JDBC para que pudiera tener la relación que esperábamos al conectar la base de datos. Y la interacción que tiene con el front-end por parte de lo que muestra la aplicación al usuario final y su interacción a su vez con este.