



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO

Tarea 12. Balance de carga en la nube

Unidad de aprendizaje: Desarrollo de Sistemas Distribuidos

Grupo: 4CV11

Alumno:
Sanchez Mendez Edmundo Josue

Profesor:
Pineda Guerrero Carlos

Índice

1. Introducción	2
2. Desarrollo	2
2.1. Creación de la maquinas virtuales (nodo 0 y 1)	2
2.2. Creación de la maquina virtual (nodo 3) y creación la base de datos “servicio_web” y el usuario “hugo” en MySQL	9
2.3. Configuración del servicio web en las dos primeras máquinas virtuales para que cada servicio web se conecte a MySQL que ejecuta en la tercera máquina virtual	13
2.4. Utilizar la aplicación web prueba.html para probar que el servicio web en cada máquina virtual tenga acceso a la base de datos en MySQL.	17
2.5. Quitar la IP pública a las máquinas virtuales dónde ejecuta Tomcat	21
2.6. Creación de un balanceador de carga y conexión a las máquinas virtuales creadas en el paso 1	22
2.6.1. Creación de un balanceador de carga en Azure	22
2.6.2. Configuración del balanceador de carga	26
2.6.3. Agregar un sondeo de estado	29
2.6.4. Agregar una regla de equilibrio de carga	31
2.7. Pruebas	33
3. Conclusiones	51
Referencias	52

1. Introducción

En esta tarea cada alumno creará un balanceador de carga en la nube de Azure. El balanceador de carga se conectará a dos máquinas virtuales con Tomcat ejecutando el servicio web que implementamos en la tarea 6.

1. Crear dos máquinas virtuales a partir de la imagen que creamos en la tarea 6. En estas máquinas virtuales no deberá ejecutar MySQL.
2. Crear una máquina virtual con Ubuntu e instalar en esta máquina virtual MySQL. Opcionalmente se podrá utilizar una instancia de MySQL en PaaS en vez de crear una tercera máquina virtual con MySQL.
3. Crear la base de datos “servicio_web” y el usuario “hugo” en MySQL.
4. Configurar el servicio web en las dos primeras máquinas virtuales para que cada servicio web se conecte a MySQL que ejecuta en la tercera máquina virtual o a MySQL en PaaS, según sea el caso.
Para configurar el acceso a MySQL, modificar el atributo “url” en el archivo “context.xml” del servicio web, cambiar localhost por la IP (o dominio) de la máquina virtual dónde ejecuta MySQL.
5. Utilizar la aplicación web prueba.html para probar que el servicio web en cada máquina virtual tenga acceso a la base de datos en MySQL.
6. Quitar la IP pública a las máquinas virtuales dónde ejecuta Tomcat. Para ello seleccionar la máquina virtual, seleccionar la IP pública, seleccionar la opción “Información general”, y seleccionar la opción “Desasociar”.
7. Seguir el procedimiento que vimos en clase para crear un balanceador de carga y conectarlo a las máquinas virtuales creadas en el paso 1.
8. Realizar las pruebas que se hicieron al servicio web en la tarea 6.

2. Desarrollo

2.1. Creación de la maquinas virtuales (nodo 0 y 1)

En esta parte veremos la creación de las maquinas virtuales que se ocuparan para el balanceador de carga, la característica principal de estas maquinas virtuales es que solo tendrán Tomcat corriendo junto con el servicio web. No sin antes ver la creación del conjunto de disponibilidad el cual sera usado para la creación de nuestras maquinas virtuales.

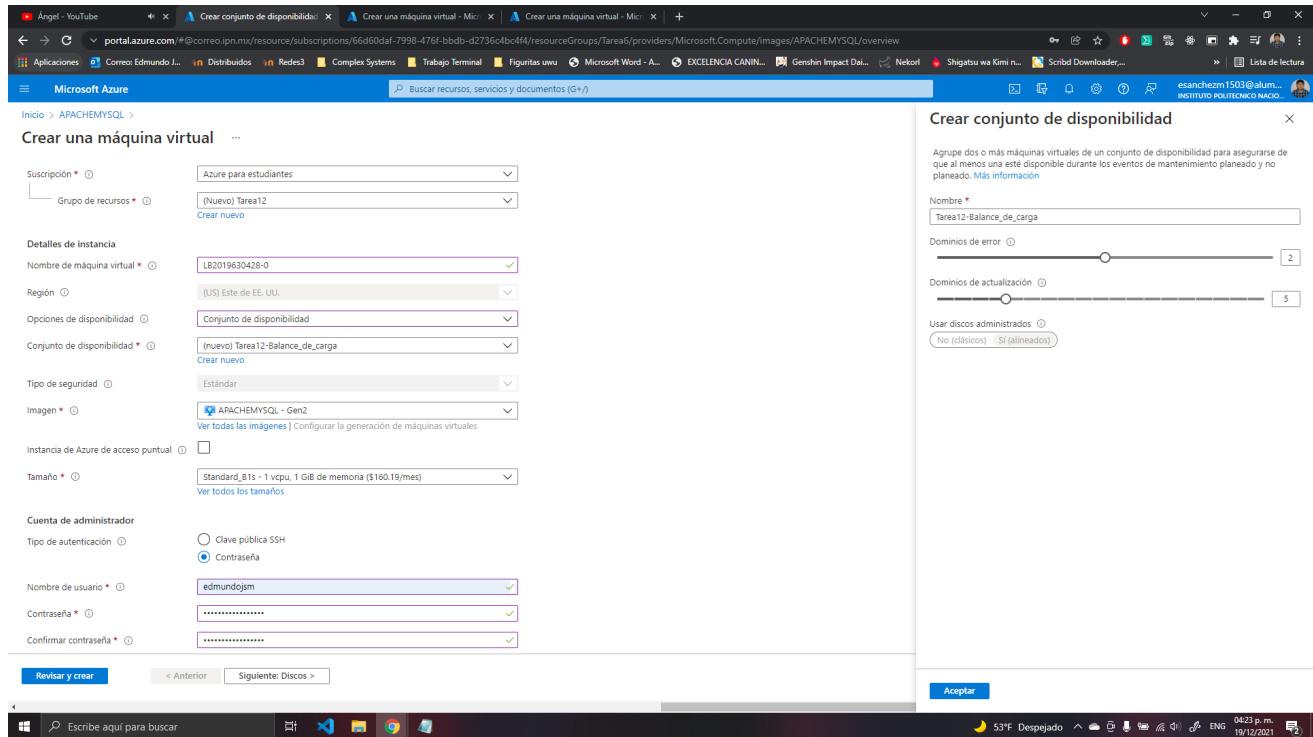


Figura 1: Creación del conjunto de disponibilidad.

Ahora si, procedamos a ver la creación de las maquinas virtuales, empezando con el nodo 0.

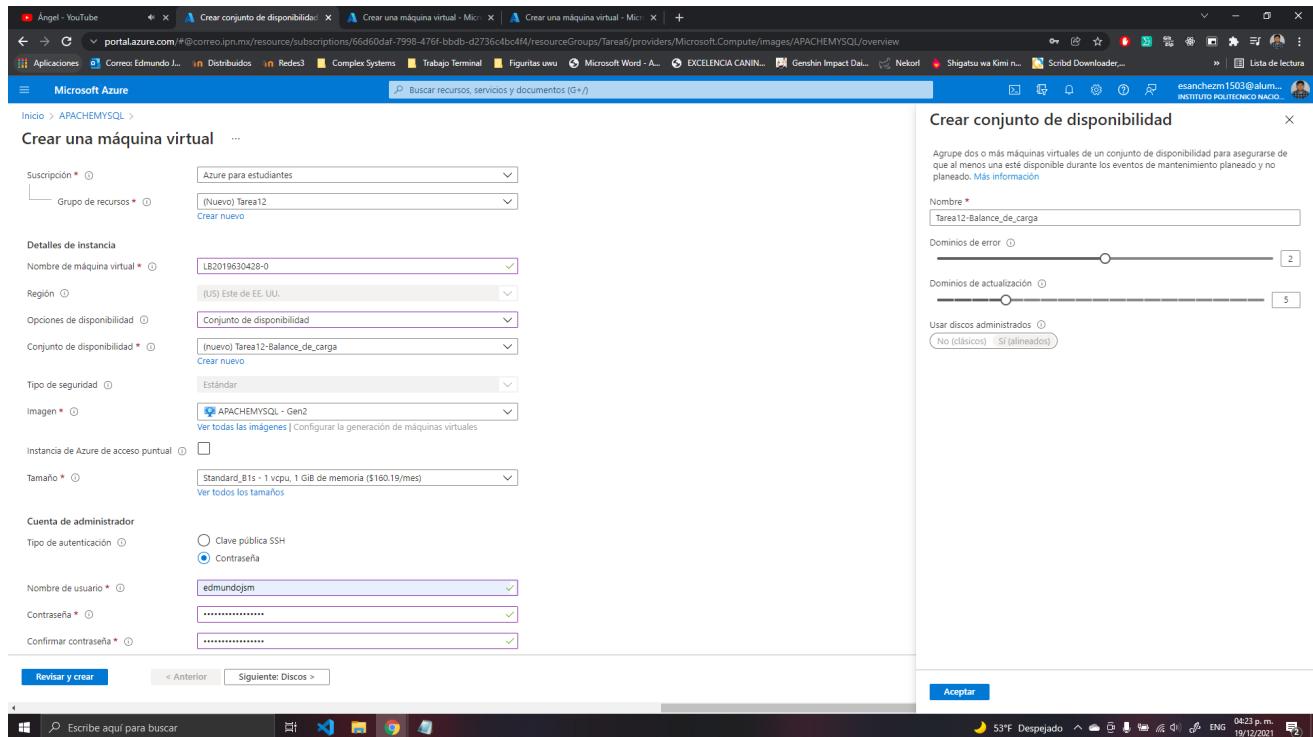


Figura 2: Datos básicos de la maquina virtual. Nodo 0.

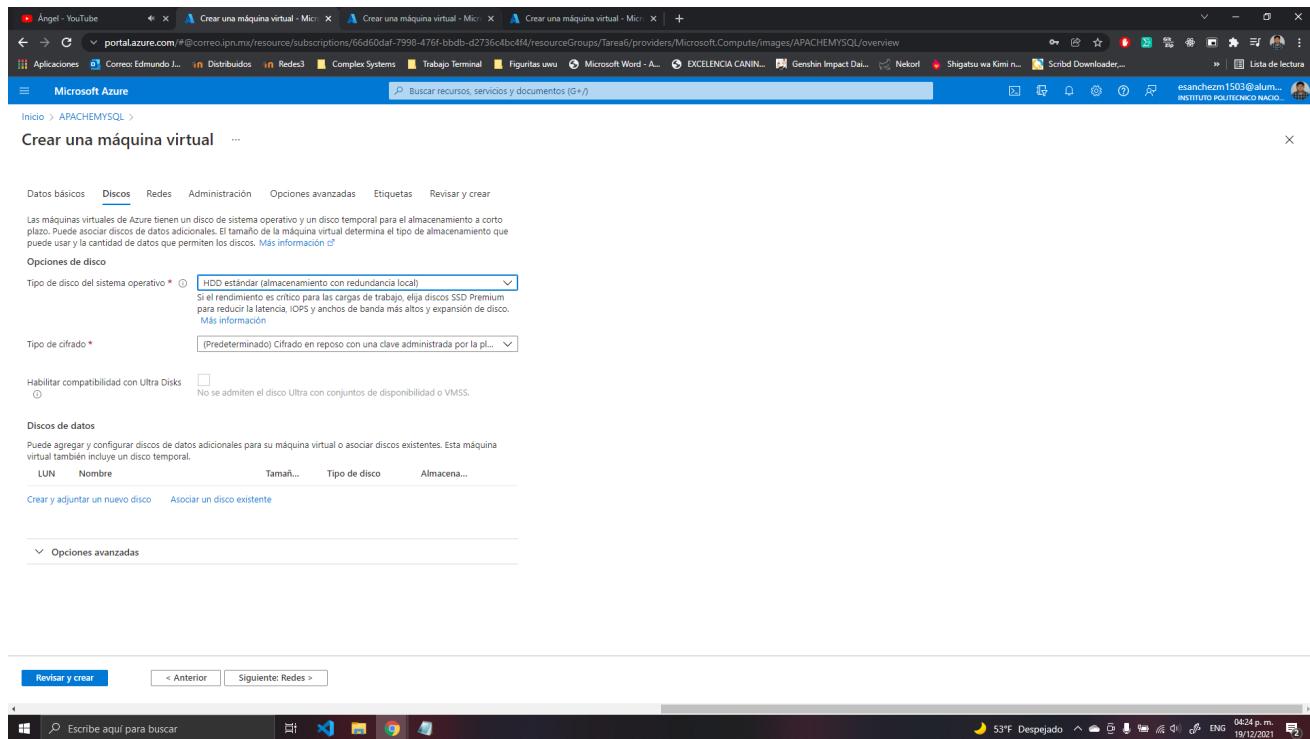


Figura 3: Configuración del tipo de disco de la maquina virtual. Nodo 0.

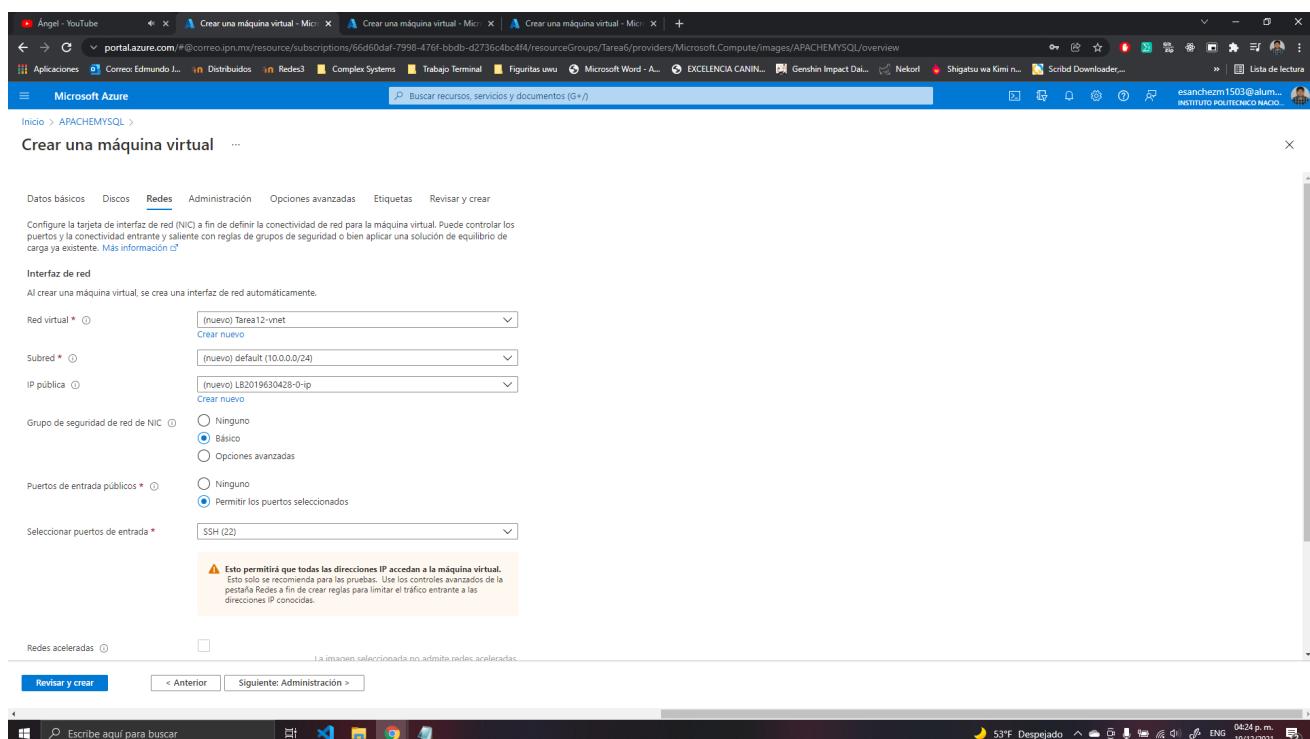


Figura 4: Información sobre las redes de la maquina virtual. Nodo 0.

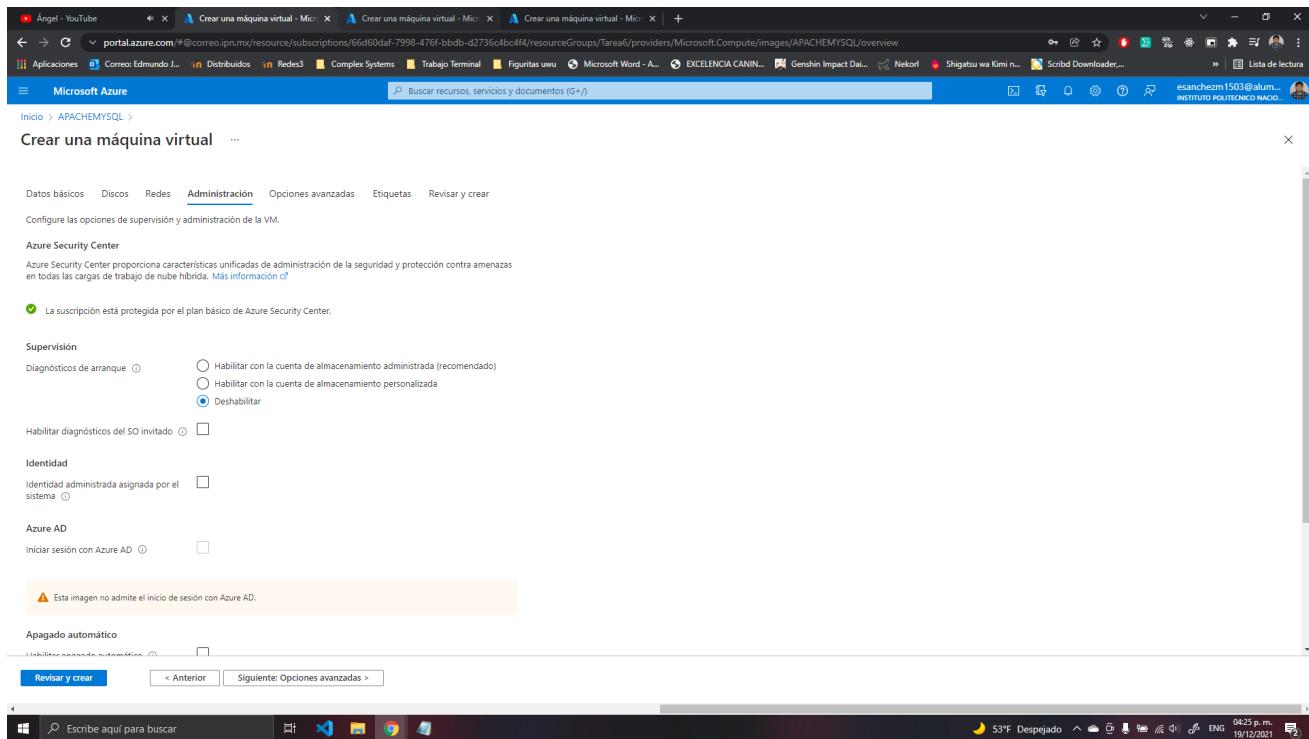


Figura 5: Configuración de la administración de la maquina virtual. Nodo 0.

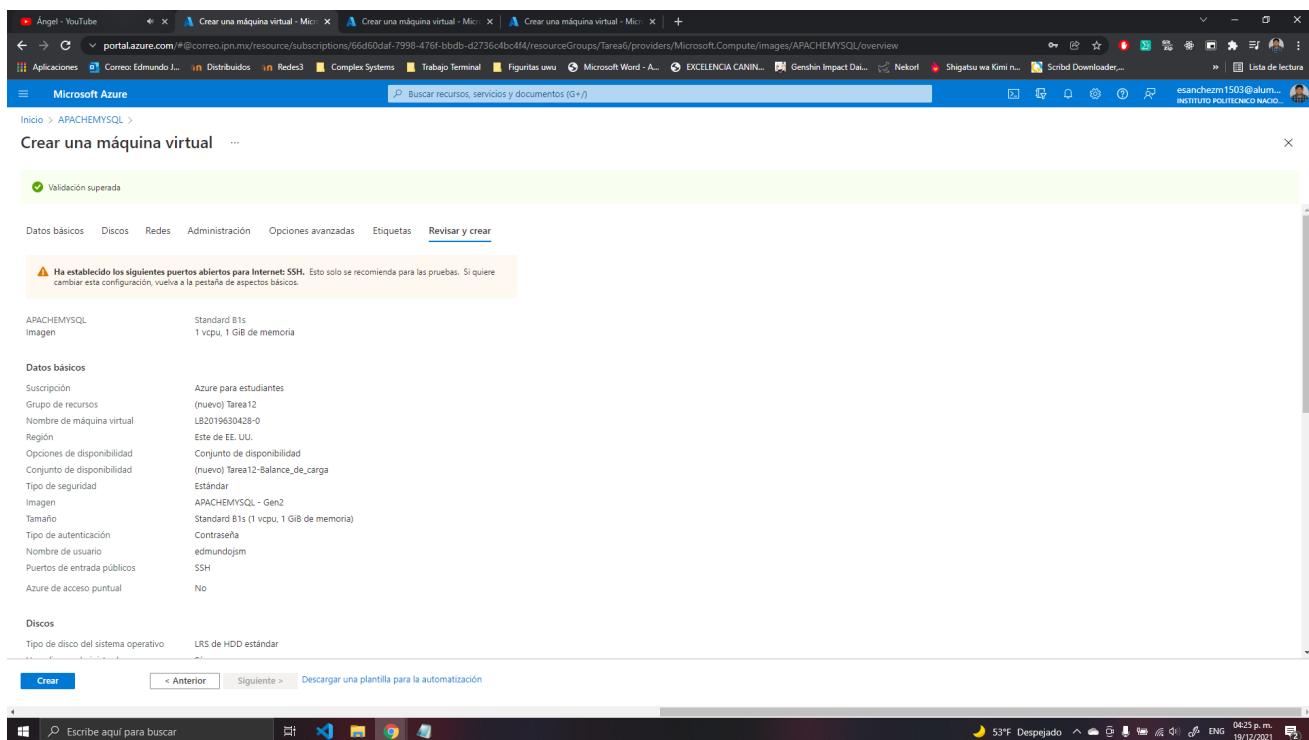


Figura 6: Creación de la maquina virtual. Nodo 0.

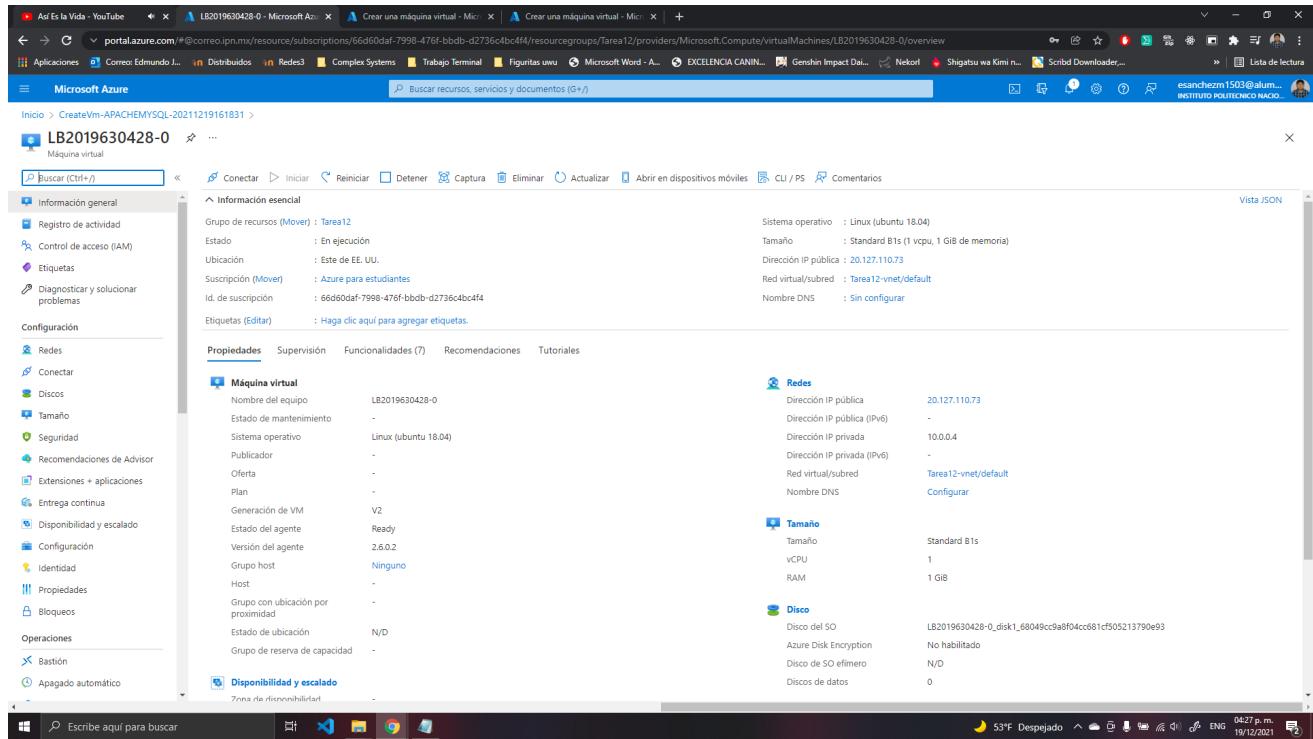


Figura 7: Panel de control de la maquina virtual. Nodo 0.

Ahora veamos la creación de la maquina virtual para el nodo 1

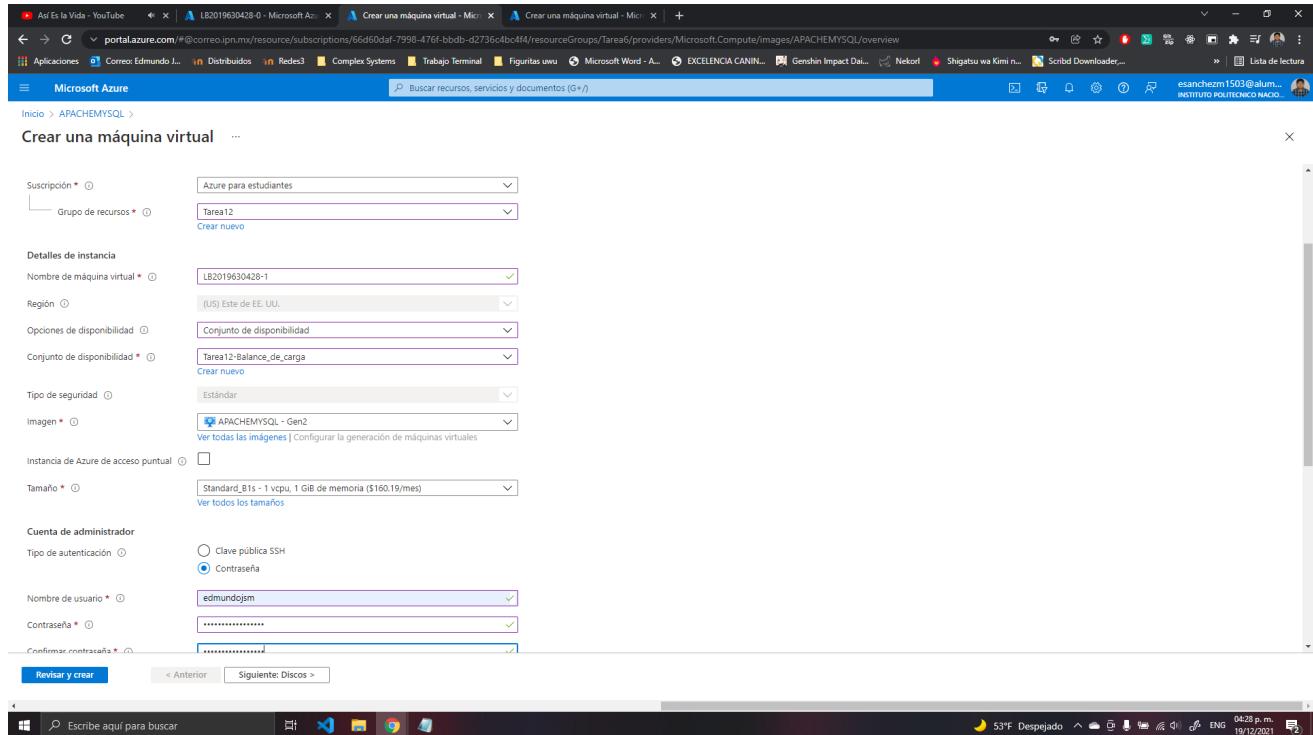


Figura 8: Datos básicos de la maquina virtual. Nodo 1.

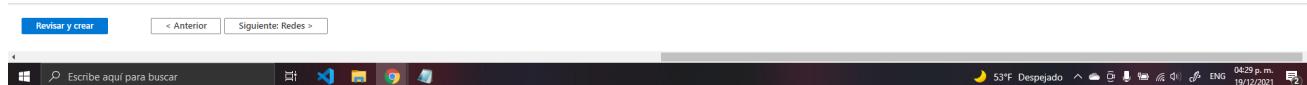
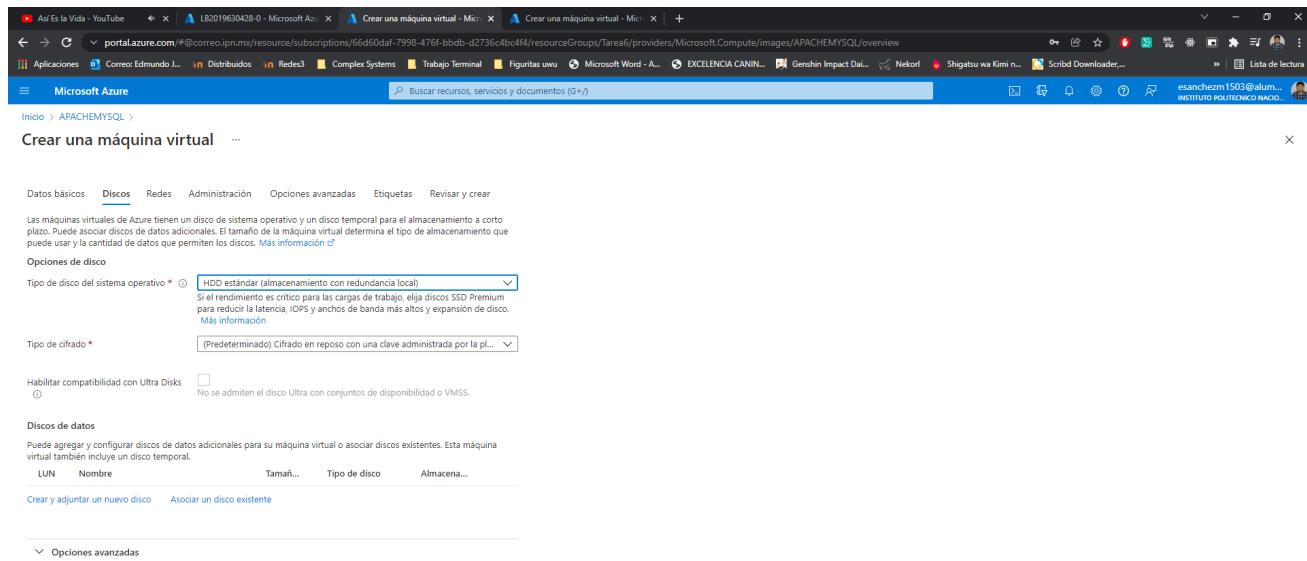


Figura 9: Configuración del tipo de disco de la maquina virtual. Nodo 1.

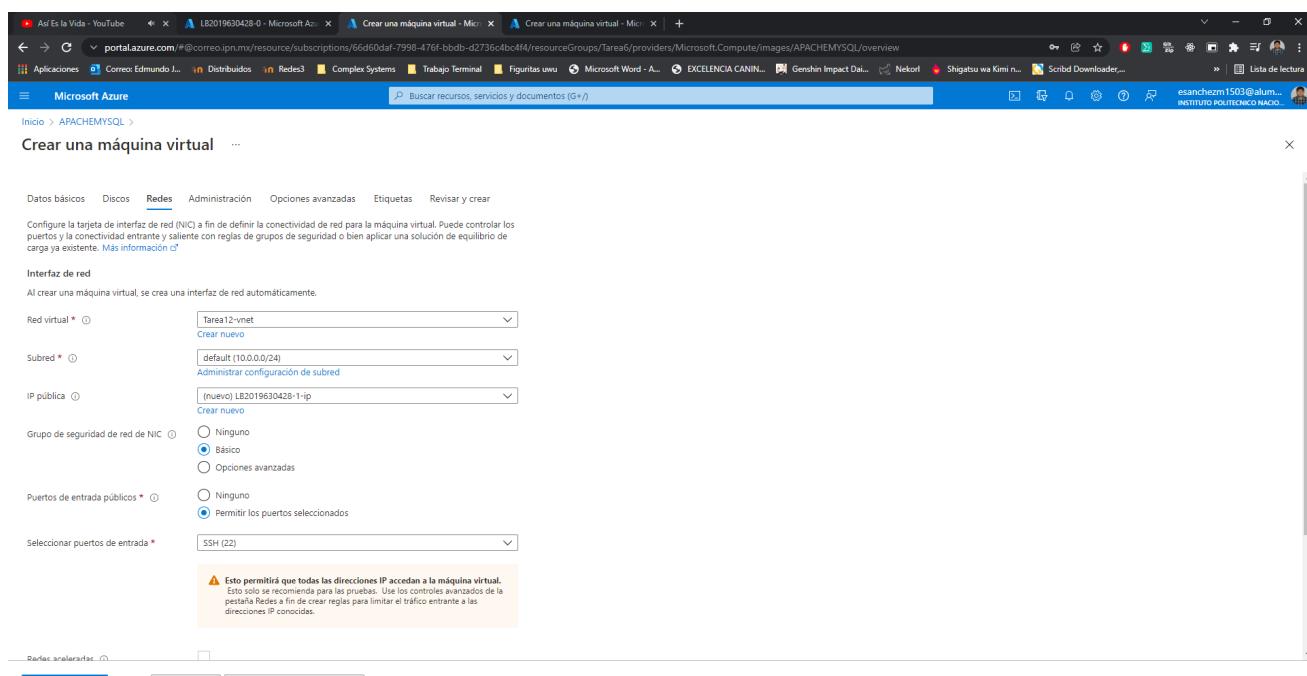


Figura 10: Información sobre la redes de la maquina virtual. Nodo 1.

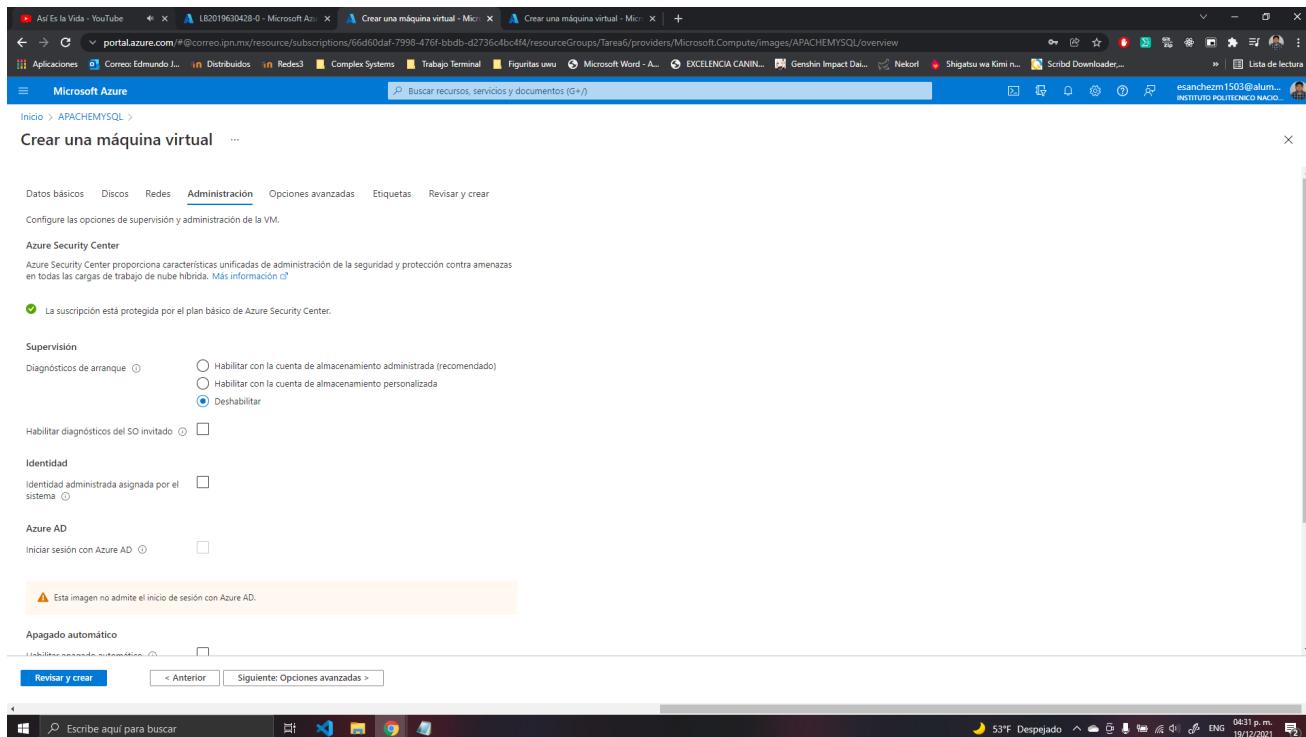


Figura 11: Configuración de la administración de la maquina virtual. Nodo 1.

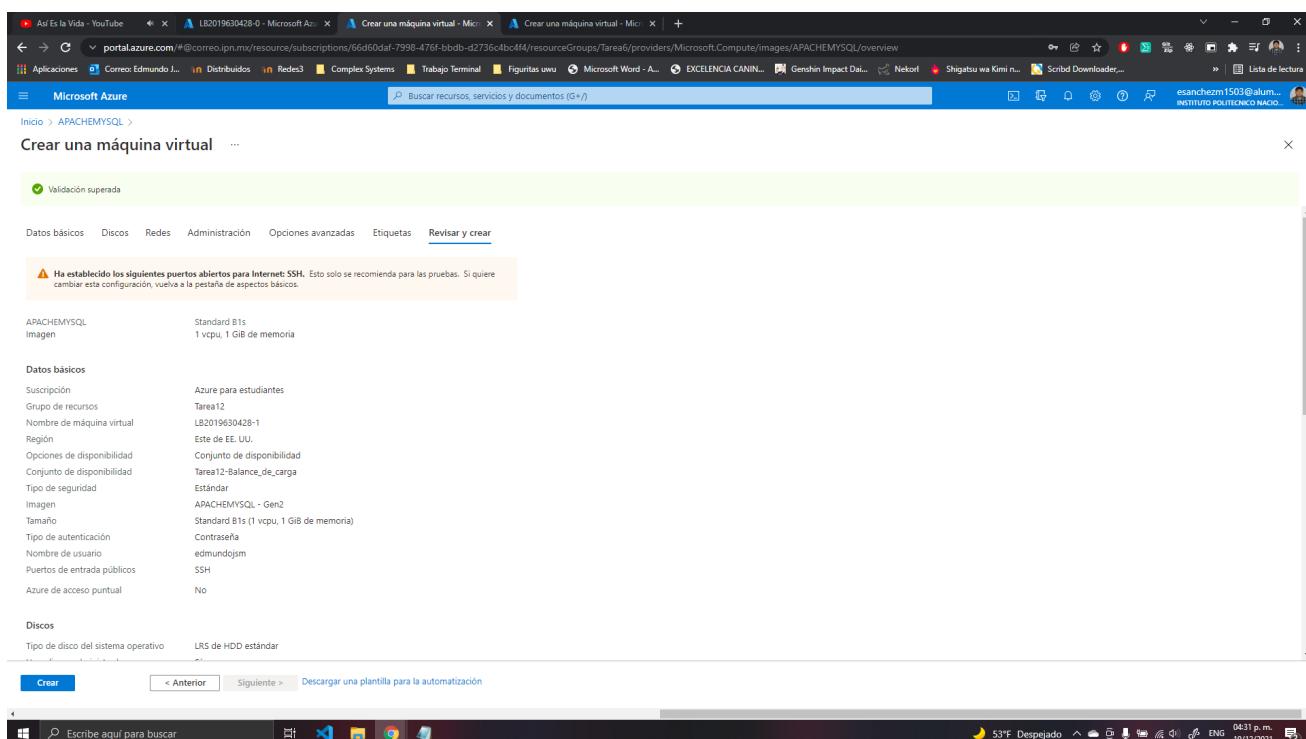


Figura 12: Creación de la maquina virtual. Nodo 1.

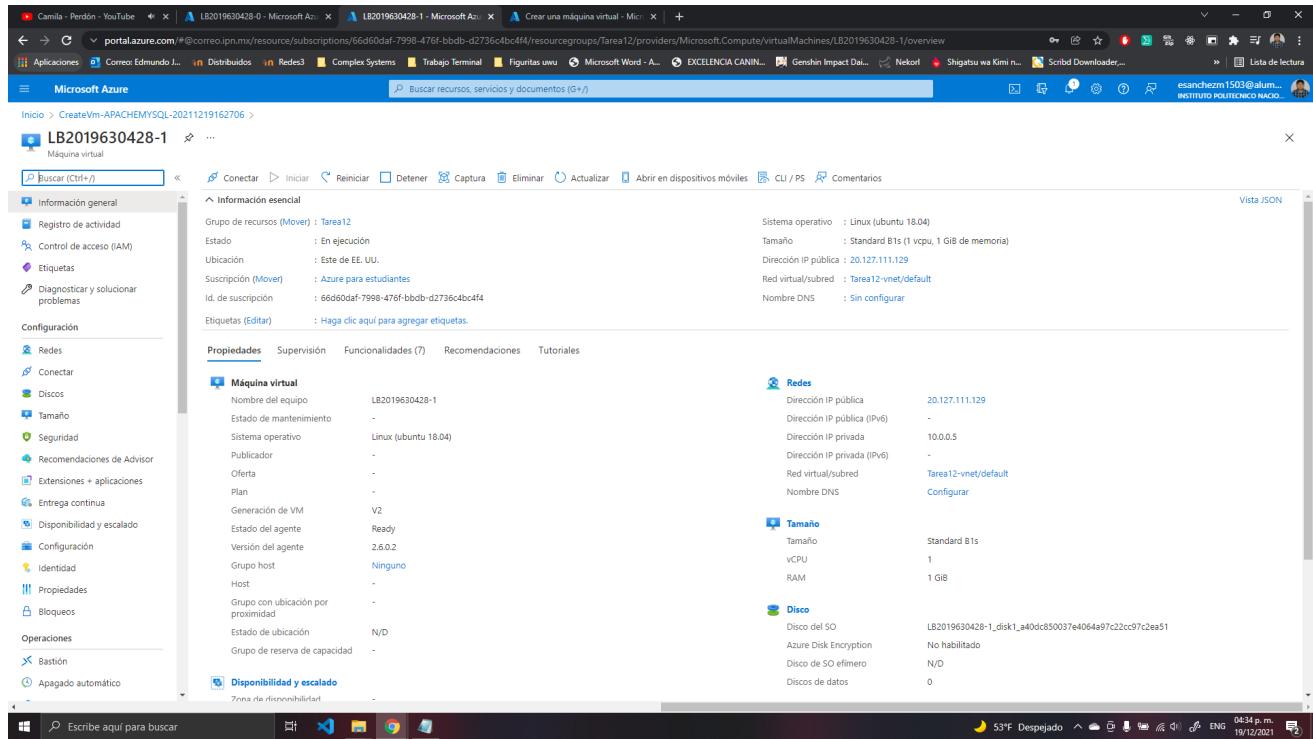


Figura 13: Panel de control de la maquina virtual. Nodo 1.

2.2. Creación de la maquina virtual (nodo 3) y creación la base de datos “servicio_web” y el usuario “hugo” en MySQL

En esta parte veremos la creación de la maquina virtual que se ocupara para tener MYSQL, por lo que tendrá la base de datos “servicio_web” y el usuario “hugo” en MYSQL, mencionar que como estamos tomando como base la imagen creada en la tarea 6 de este curso, ya no es necesario la creación de estos dos, pero si es necesario la actualización del usuario hugo para que sea usable para conexión remota en MYSQL.

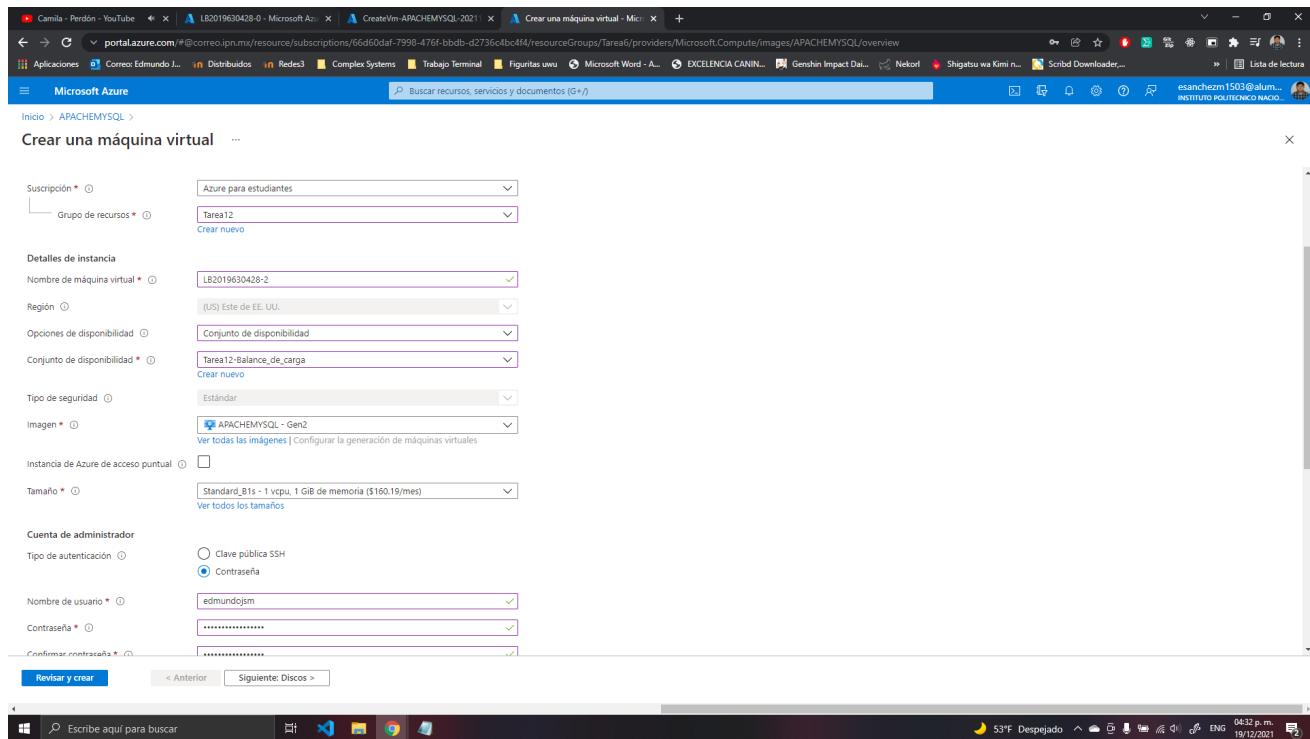


Figura 14: Datos básicos de la maquina virtual. Nodo 2.

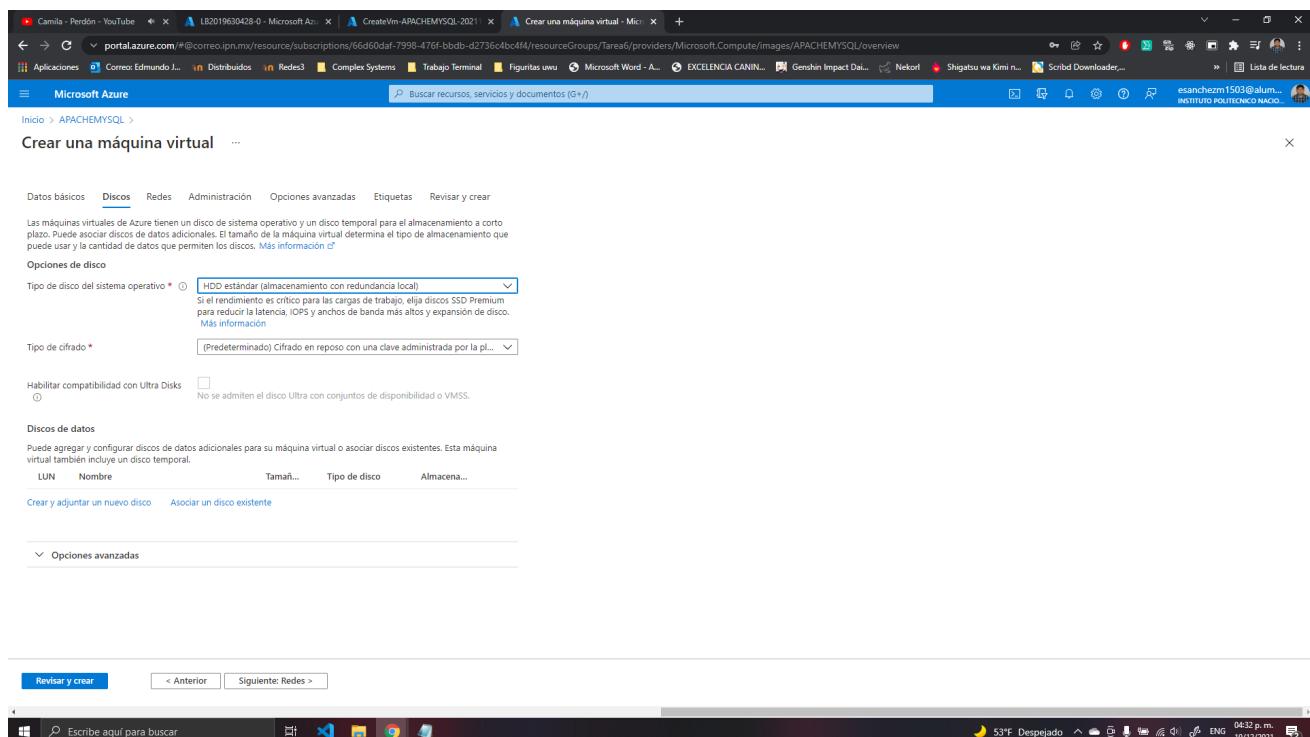


Figura 15: Configuración del tipo de disco de la maquina virtual. Nodo 2.

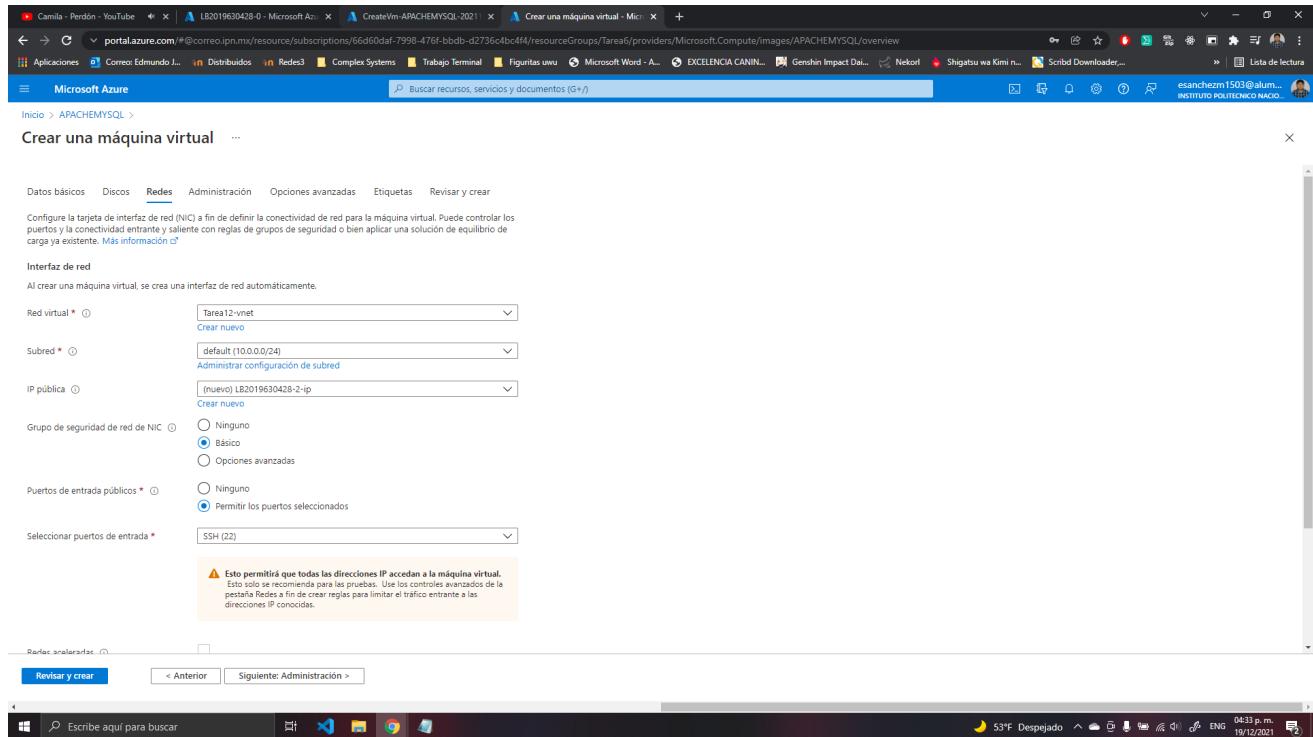


Figura 16: Información sobre la redes de la maquina virtual. Nodo 2.

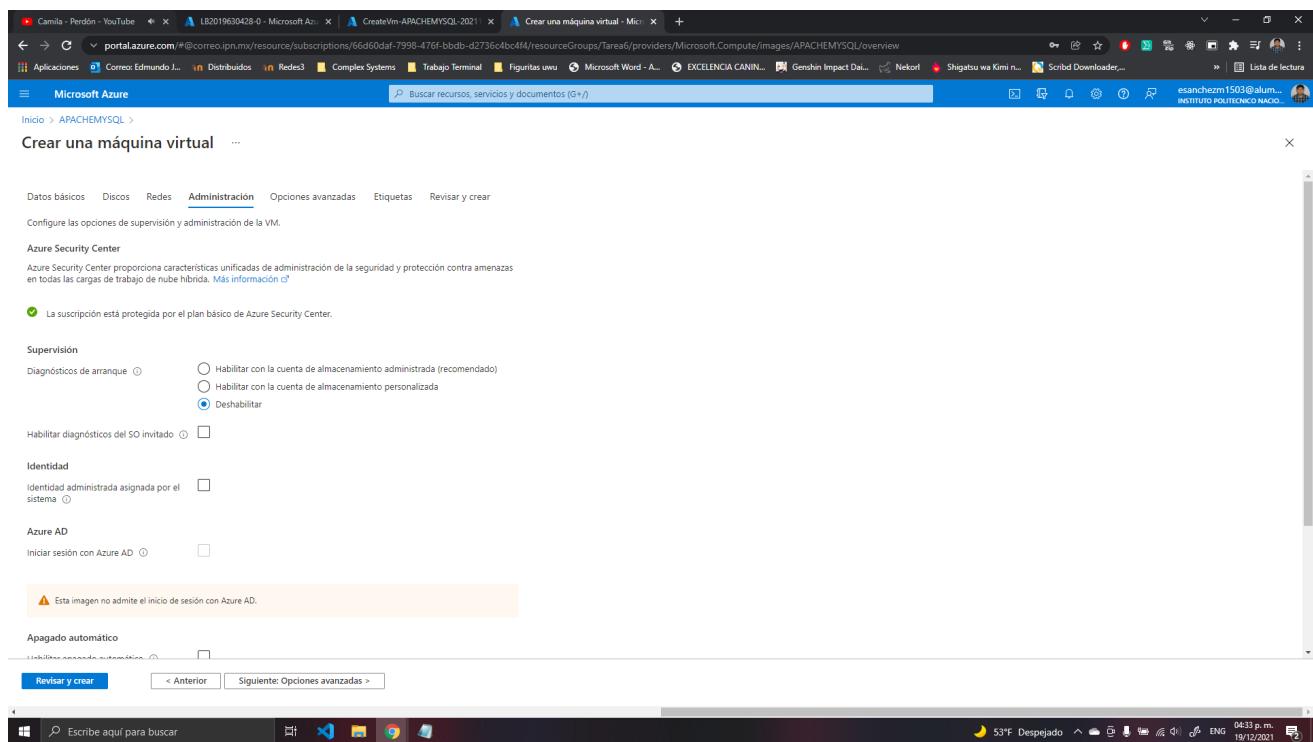


Figura 17: Configuración de la administración de la maquina virtual. Nodo 2.

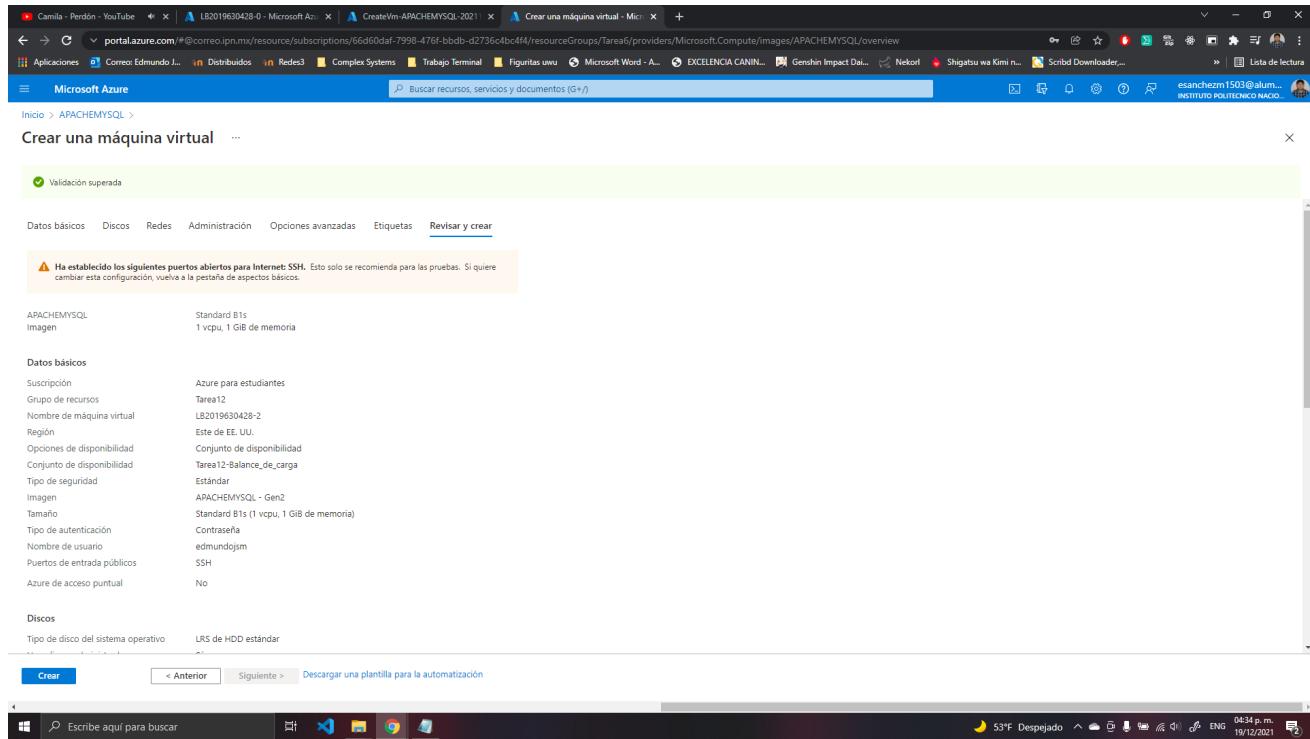


Figura 18: Creación de la maquina virtual. Nodo 2.

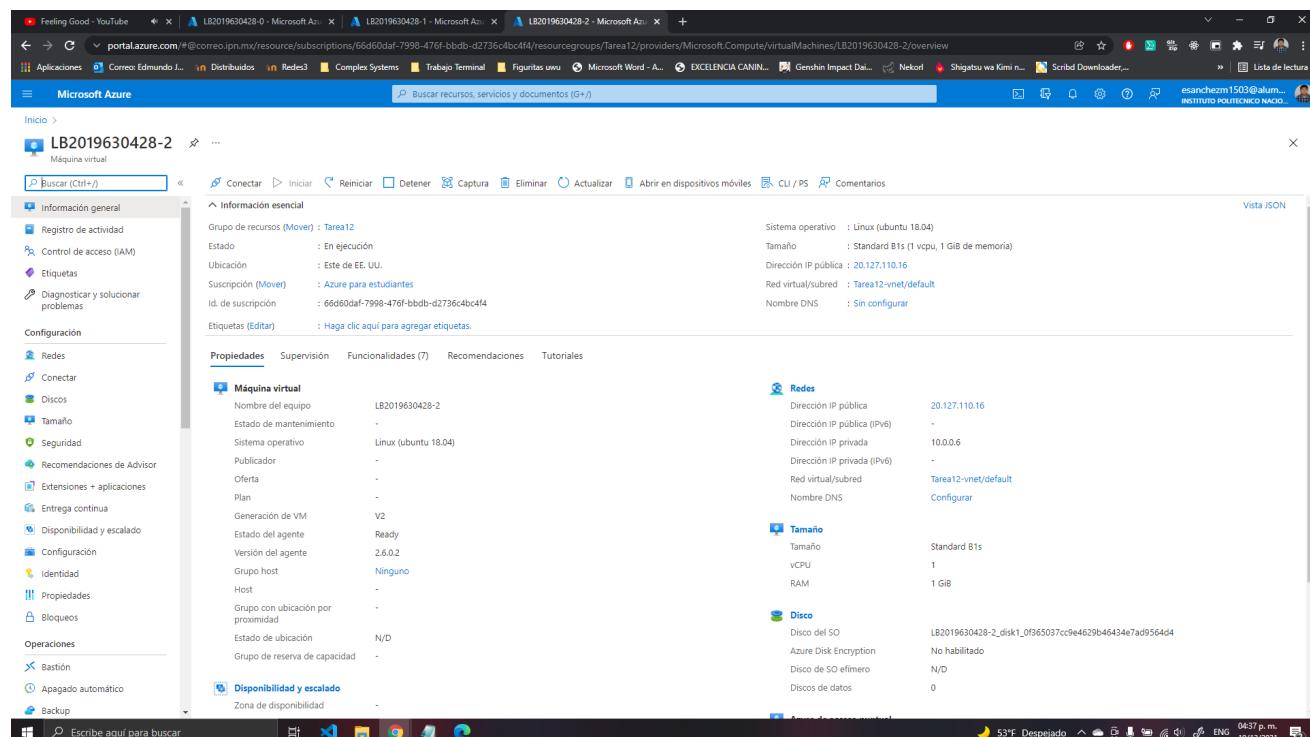


Figura 19: Panel de control de la maquina virtual. Nodo 2.

Finalmente en la figura 20 podemos ver como ya existe el usuario hugo en MYSQL y ademas ya existe la base de datos servicio_web por lo que ya no es necesario crearlas, ademas de que vemos que ya tenemos registros en ella.

```

edmundojm@UB2019630428-2: ~
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.127.110.16' (EDSA) to the list of known hosts.
edmundojm@UB20.127.110.16's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1063-azure x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information disabled due to load higher than 1.0

 * Super-optimized for small spaces - read how we shrink the memory
   footprint of MicroK8s to make it the smallest full K8s around.

https://ubuntu.com/blog/microk8s-memory-optimisation

37 updates can be applied immediately.
26 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Sat Nov 27 07:38:11 2021 from 189.229.14.64
edmundojm@UB2019630428-2: $ mysql -u hugo -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.36-0ubuntu0.18.04.1 (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> use servicio_web;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo | Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>

```

Figura 20: Base de datos y usuarios existentes.

2.3. Configuración del servicio web en las dos primeras máquinas virtuales para que cada servicio web se conecte a MySQL que ejecuta en la tercera máquina virtual

Para configurar el acceso a MySQL, se modifica el atributo `uri` en el archivo `context.xml` del servicio web, en el cual se debe cambiar localhost por la IP de la máquina virtual donde ejecuta MySQL, como vemos en las figuras 21 y 22.

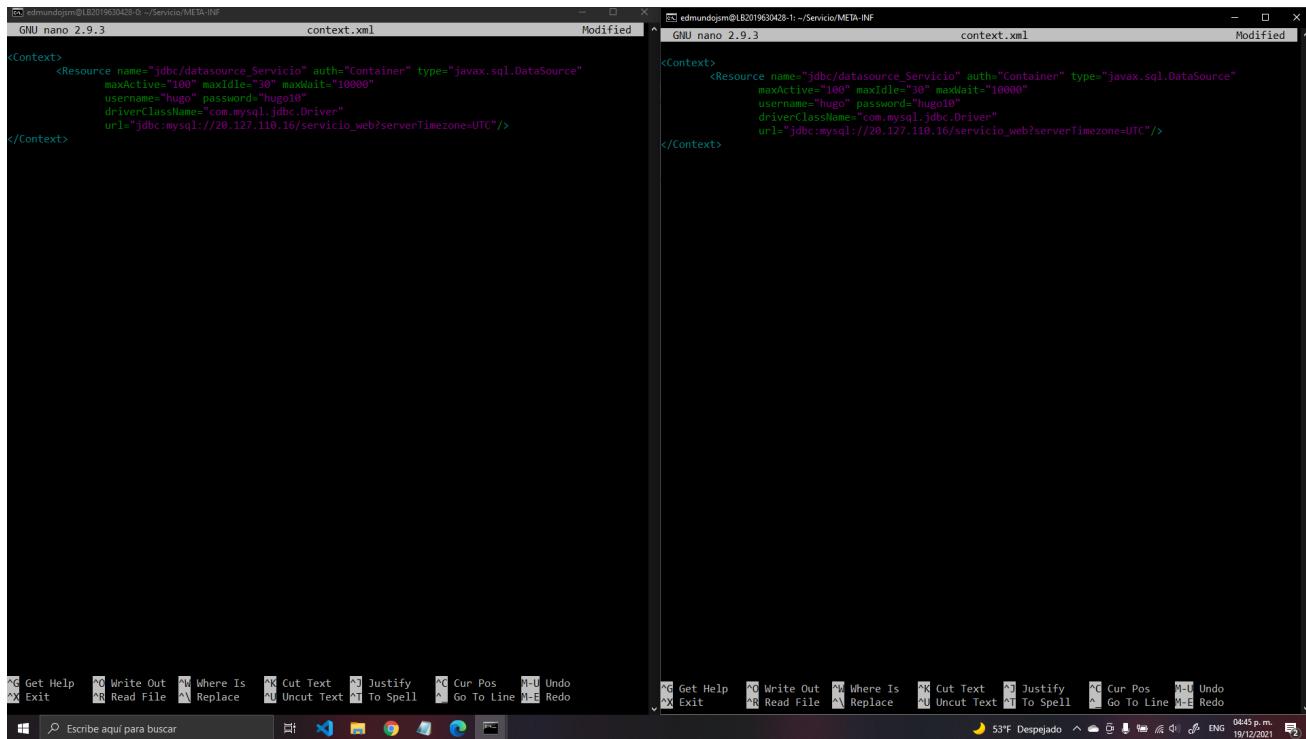


Figura 21: Cambio del archivo context.xml con la IP de la maquina virtual con MYSQL.

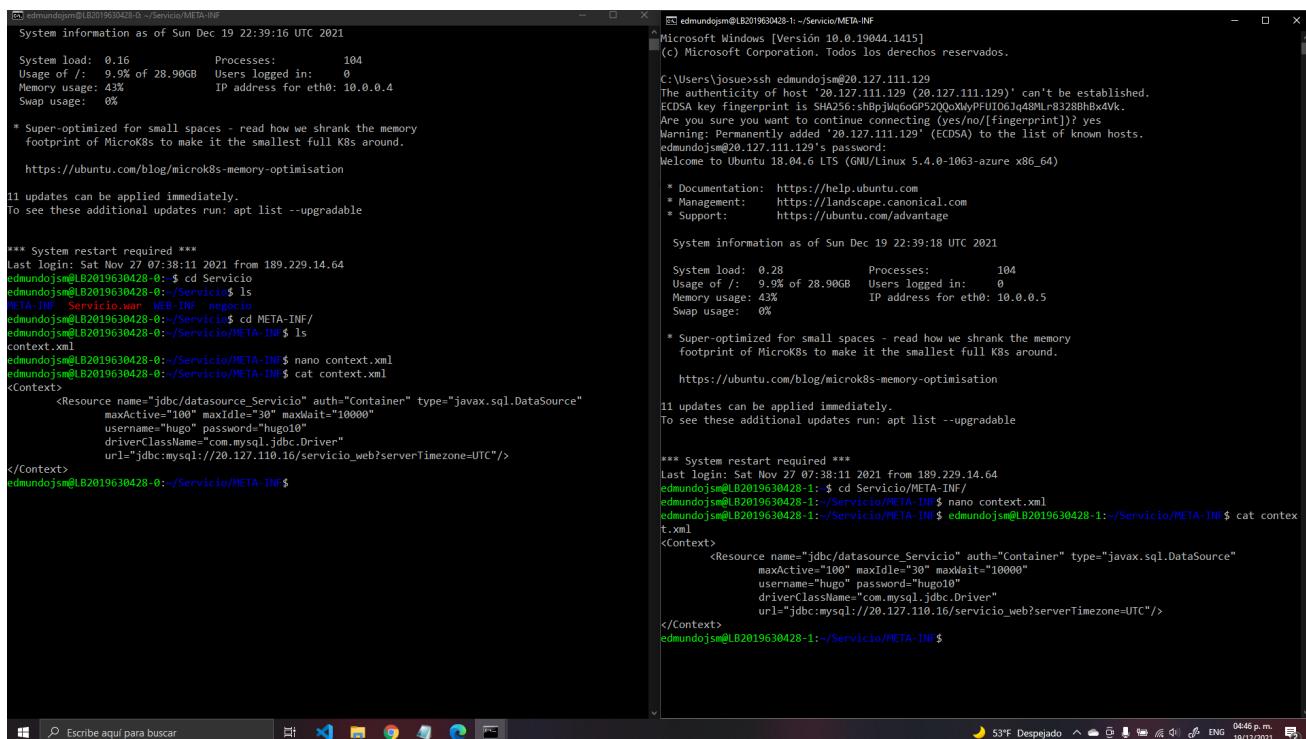
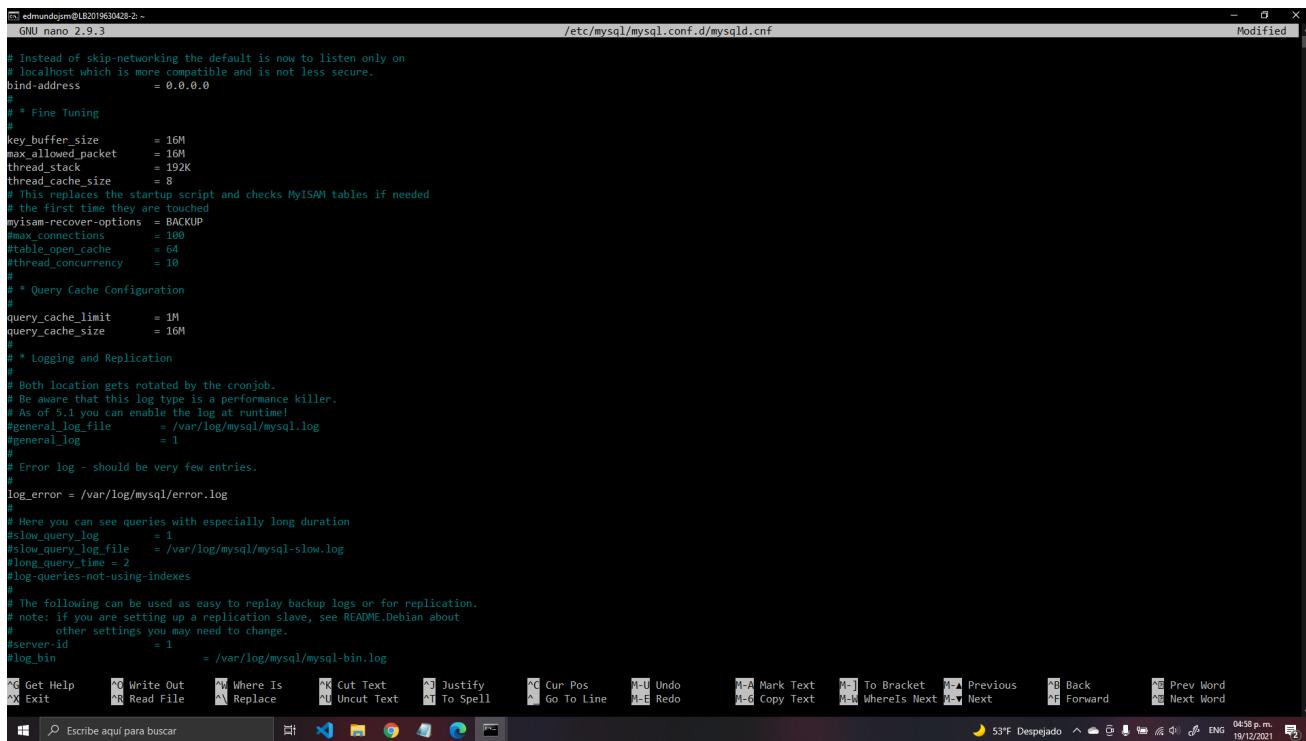


Figura 22: Cambio del archivo context.xml realizado.

Finalmente se creara el servicio web para Tomcat con el cambio realizado en ambas maquinas virtuales, ademas de copiar el .war a webapps de Tomcat, sin embargo, esto no es suficiente ya que necesitamos habilitar conexión remota a MYSQL. Primero tenemos que permitir a MySQL escuchar tráfico externo,

para ello es necesario que se habilite la escucha de direcciones IP externas, para activar esto, se tiene que modificar el archivo mysqld.cnf, ahí se tendrá que modificar una linea que empieza con la directiva bind-address, por defecto el valor asignado es 127.0.0.1 y se tendrá que modificar por una IP externa que se conectara con MYSQL o con comodines para permitir conexiones remotas en general, sin restringir a direcciones IP específicas, para esto asignamos el valor como *, ::, o 0.0.0.0. Se guarda el documento y reiniciamos el demonio de MYSQL para que los cambios realizados tengan efecto, esto lo podemos ver en las figuras 23 y 24.



```

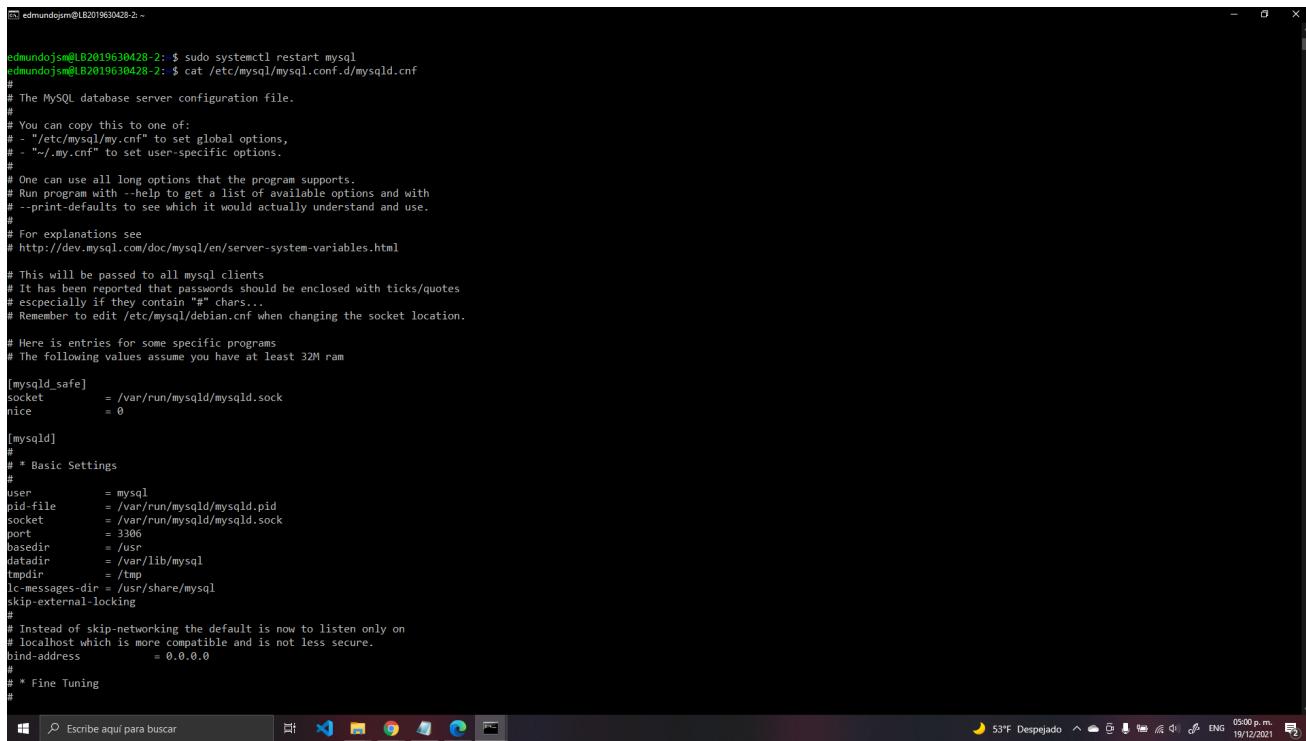
edmundosp@LB019630428-2: ~
GNU nano 2.9.3
/etc/mysql/mysql.conf.d/mysqld.cnf
Modified

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 0.0.0.0
#
# * Fine Tuning
key_buffer_size     = 16M
max_allowed_packet = 16M
thread_stack        = 192K
thread_cache_size   = 0
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched
myisam-recover-options = BACKUP
max_connections      = 100
table_open_cache     = 64
thread_concurrency   = 10
#
# * Query Cache Configuration
#
query_cache_limit    = 1M
query_cache_size     = 16M
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# As of 5.1 you can enable the log at runtime!
general_log_file     = /var/log/mysql/mysql.log
general_log           = 1
#
# Error log - should be very few entries.
#
log_error            = /var/log/mysql/error.log
#
# Here you can see queries with especially long duration
slow_query_log        = 1
slow_query_log_file   = /var/log/mysql/mysql-slow.log
long_query_time       = 2
log-queries-not-using-indexes
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
# other settings you may need to change.
server-id             = 1
log_bin               = /var/log/mysql/mysql-bin.log

Get Help   Write Out   Where Is   Cut Text   Justify   Cur Pos   Undo   Mark Text   To Bracket   Previous   Back   Prev Word
Exit      Read File   Replace   Uncut Text  To Spell   Go To Line  Redo   Copy Text   WhereIs Next  Next   Forward  Next Word
Windows Taskbar: Escribe aquí para buscar
System tray icons: 53°F Despejado, 04:38 p.m., 19/12/2021

```

Figura 23: Permitir a MySQL escuchar tráfico externo.



```

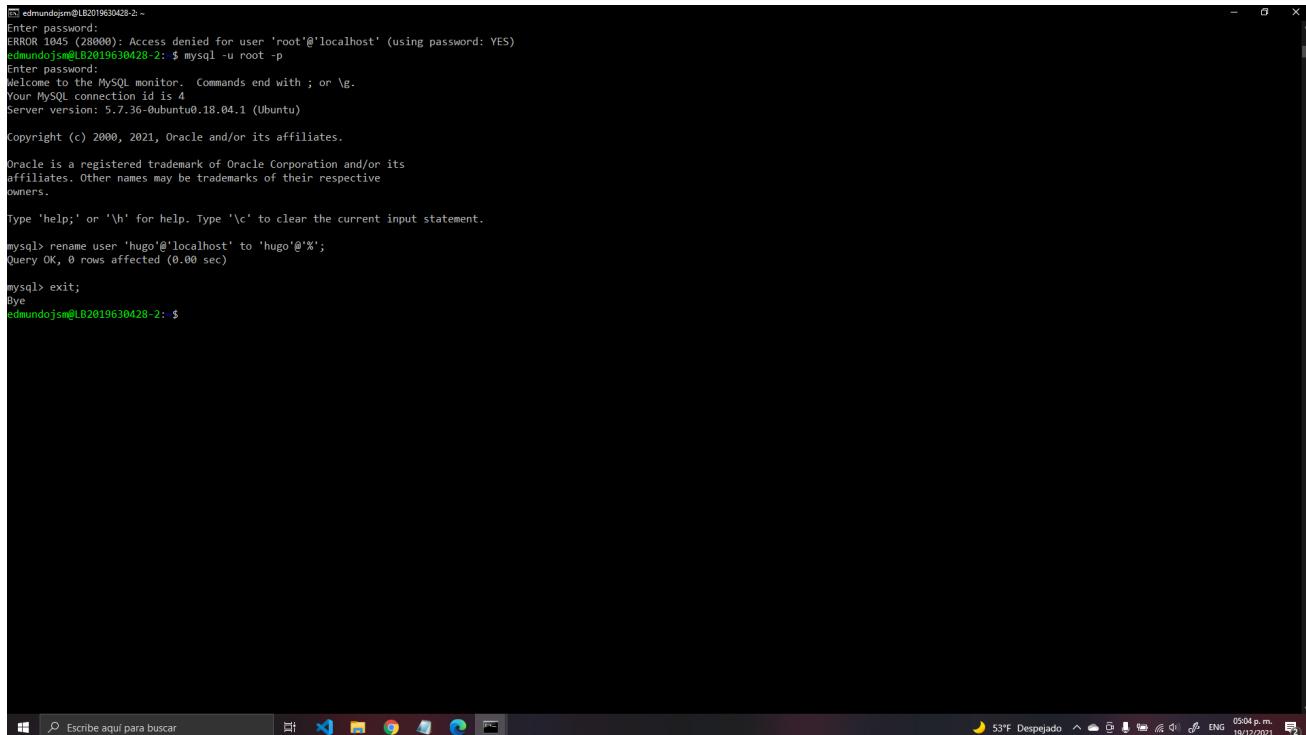
edmundojsm@LB2019630428-2:~$ sudo systemctl restart mysql
edmundojsm@LB2019630428-2:~$ cat /etc/mysql/mysql.conf.d/mysqld.cnf
#
# The MySQL database server configuration file.
#
# You can copy this to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~-./my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html
#
# This will be passed to all mysql clients
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.
#
# Here is entries for some specific programs
# The following values assume you have at least 32M ram
[mysqld_safe]
socket          = /var/run/mysqld/mysqld.sock
nice            = 0

[mysqld]
#
# * Basic Settings
#
user            = mysql
pid-file        = /var/run/mysqld/mysqld.pid
socket          = /var/run/mysqld/mysqld.sock
port            = 3306
basedir         = /usr
datadir         = /var/lib/mysql
tmpdir          = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address    = 0.0.0.0
#
# * Fine Tuning
#

```

Figura 24: Demonio reiniciado con los cambios realizados con éxito.

Ahora actualizamos el usuario hug, para cambiar el host de un usuario, podemos usar el comando RENAME USER de MySQL. Como vemos en la figura 25, mencionar que al final del comando se ingresa el carácter % para permitir que cualquier IP pueda usar el usuario hugo.



```

edmundojsm@LB2019630428-2:~$ Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
edmundojsm@LB2019630428-2:~$ mysql -u root
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.36-0ubuntu0.18.04.1 (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> rename user 'hugo'@'localhost' to 'hugo'@'%';
Query OK, 0 rows affected (0.00 sec)

mysql> exit;
Bye
edmundojsm@LB2019630428-2:~$ 
```

Figura 25: Actualización del usuario hugo.

2.4. Utilizar la aplicación web prueba.html para probar que el servicio web en cada máquina virtual tenga acceso a la base de datos en MySQL.

Antes de probarlo, nos damos cuenta que tenemos que hacer modificaciones a los puertos de las maquinas virtuales ya que tenemos que abrir el puerto 8080 de cada una para tener acceso al servicio web, como vemos en las figuras 26 y 27, estos dos pasos nos servira mas adelante para la implementacion del balance de carga.

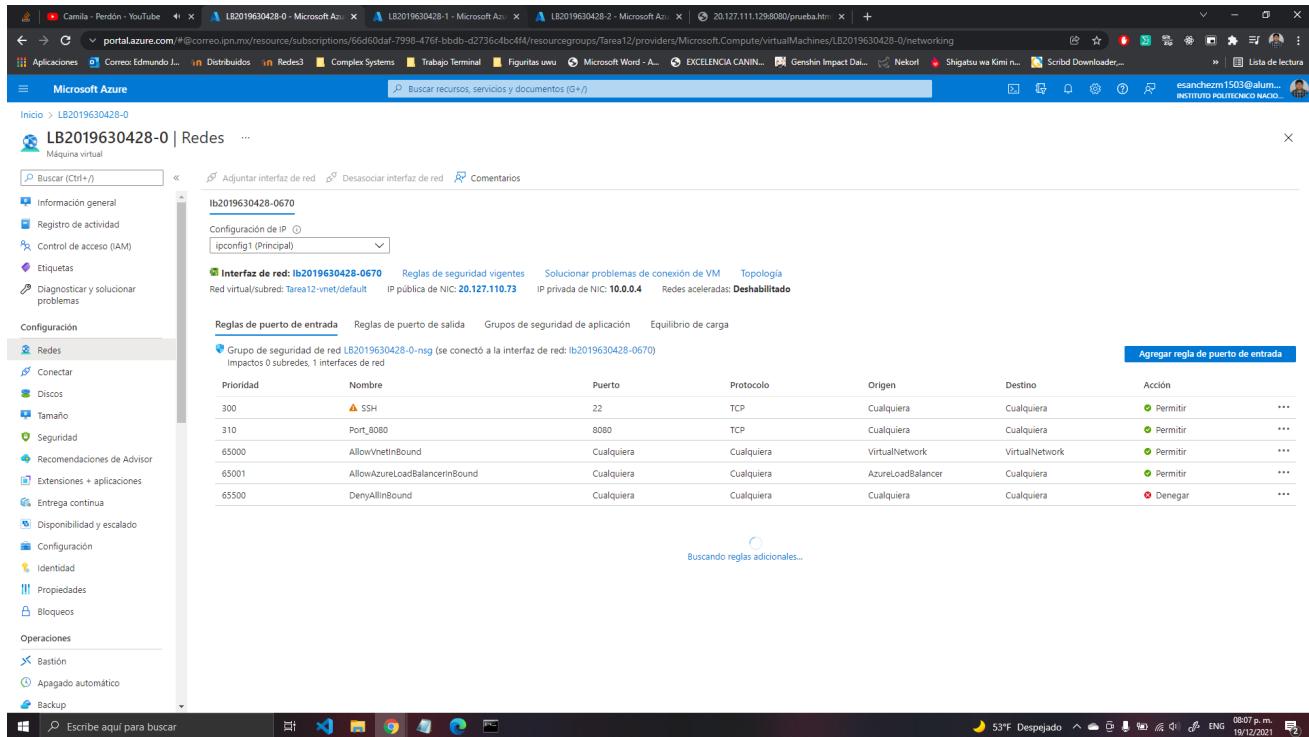


Figura 26: Puerto 8080 abierto en el nodo 0.

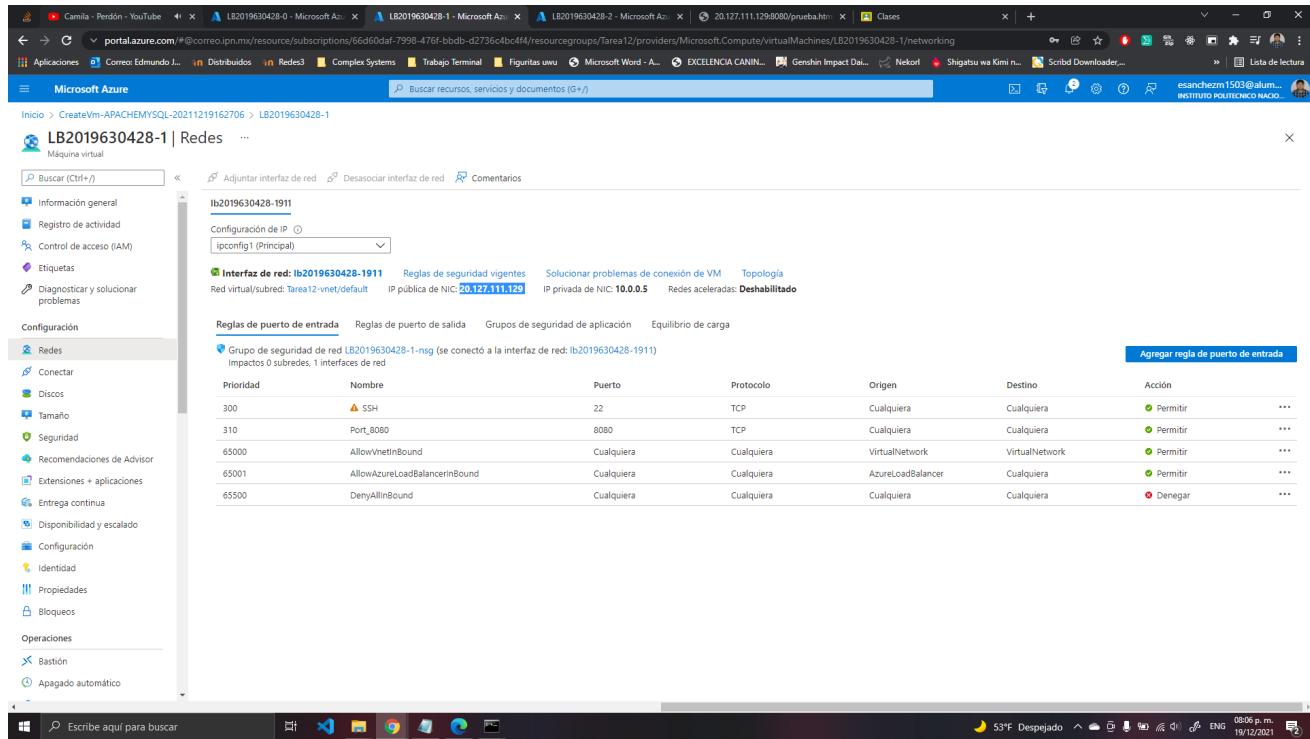


Figura 27: Puerto 8080 abierto en el nodo 1.

Sin embargo, nos encontramos con otro problema como vemos en la figura 28 para ello tenemos que comentar o eliminar la linea `request.setRequestHeader("Content-length", body.length);` como vemos en la figura 29, este problema es provocado ya que XMLHttpRequest no puede configurar estos encabezados, el navegador los configura automáticamente. La razón es que al manipular estos encabezados, es posible que se pueda engañar al servidor para que acepte una segunda solicitud a través de la misma conexión, una que no pasaría por las comprobaciones de seguridad habituales; eso sería una vulnerabilidad de seguridad en el navegador.

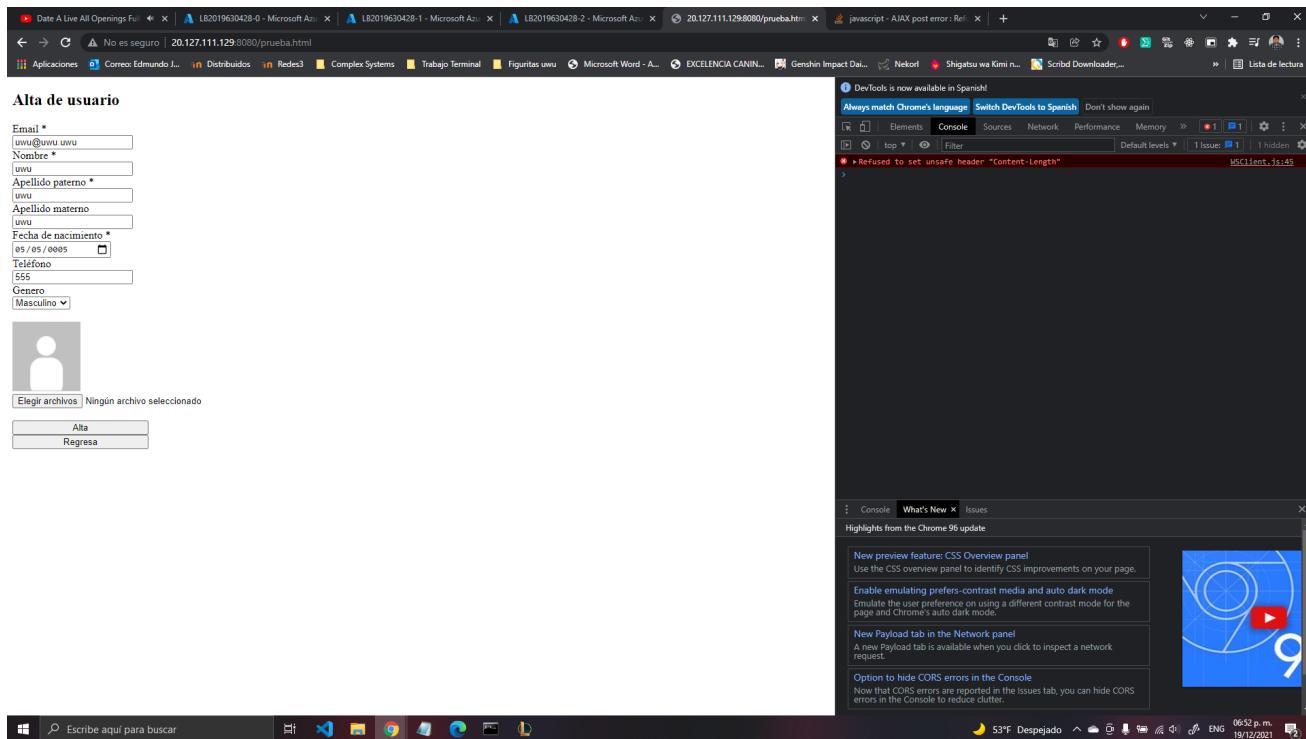


Figura 28: Problema con JavaScript.

```

edmundojm@LB2019630428-1:~/apache-tomcat-5.5.72/webapps/ROOT
GNU nano 2.9.3                               WSClient.js

/*
 * This is a simple XMLHttpRequest-based client for making POST requests.
 */

function WSClient(url)
{
    this.url = url;
    this.post = function(operation,args,callback)
    {
        var request = new XMLHttpRequest();
        var body = "";
        var pairs = [];
        var name;
        try
        {
            for (name in args)
            {
                var value = args[name];
                if (typeof value != "string") value = JSON.stringify(value);
                pairs.push(name + '=' + encodeURIComponent(value).replace(/#/g,"%20").replace(/\&/g,"%26"));
            }
            body = pairs.join("&");
            request.open("POST", url + '/' + operation);
            request.setRequestHeader('Content-type','application/x-www-form-urlencoded');
            //request.setRequestHeader('Content-Length',body.length);
            request.responseType = 'json';
            request.onload = function()
            {
                if (callback != null) callback(request.status,resolveReferences(request.response));
            }
            request.send(body);
        }
        catch (e)
        {
            alert("Error: " + e.message);
        }
    }
}

// http://stackoverflow.com/questions/15312529/resolve-circular-references-from-json-object
function resolveReferences(json)
{
    if (typeof json === 'string') json = JSON.parse(json);
}

```

Figura 29: Solución del problema con JavaScript.

Finalmente veamos las pruebas en la figura 30 en ambas maquinas virtuales, del lado izquierdo tenemos al nodo 0 y del lado derecho tenemos al nodo 1. Podemos ver en la figura 31 como en la base de datos que tenemos en el nodo 2 el registro de usuario fue realizado de manera correcta.

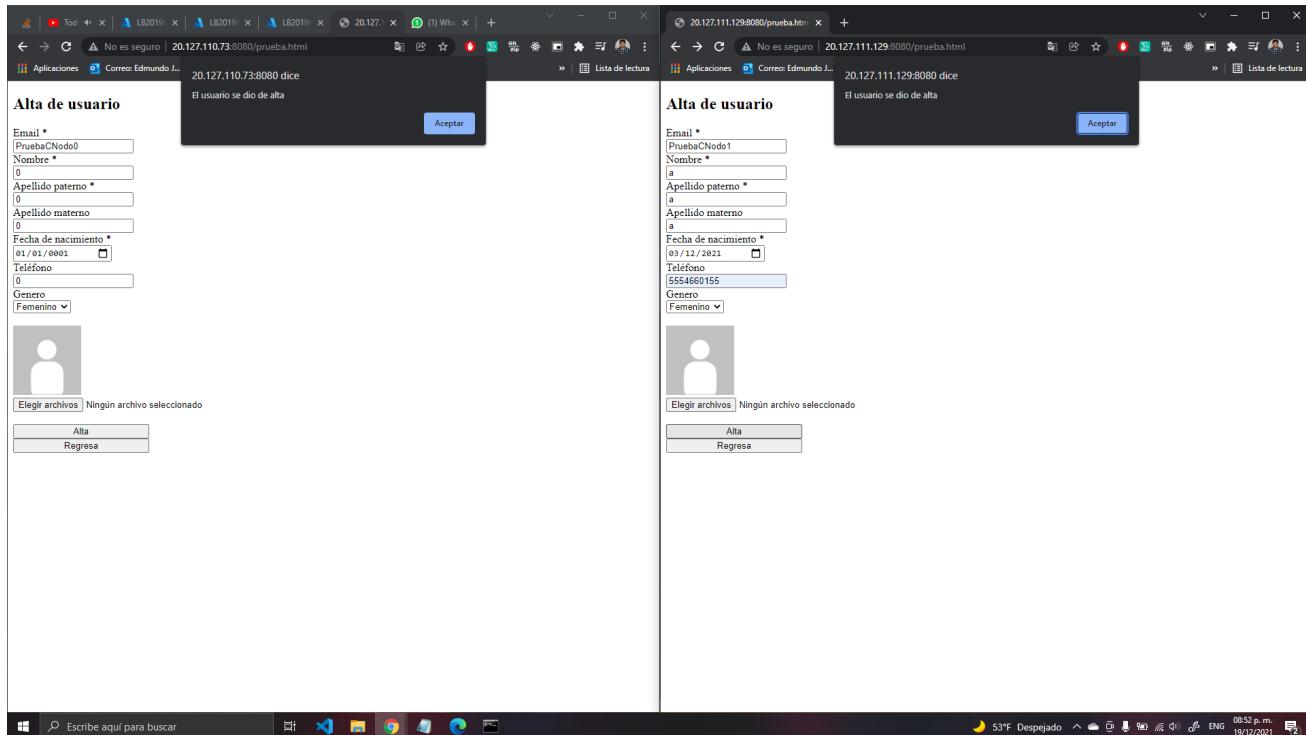


Figura 30: Pruebas del servicio.

```

edmundojm@LB2019630428-2 ~
owners.

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

mysql> use servicio_web;
ERROR 1049 (42000): Unknown database 'servicio_web'.
mysql> use servicio_web;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-03 | 5554660155 | F |
| 8 | PruebaC Nodo0 | 0 | 0 | 0 | 0001-01-01 | 5555 | F |
| 9 | PruebaC Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-03 | 5554660155 | F |
| 8 | PruebaC Nodo0 | 0 | 0 | 0 | 0001-01-01 | 5555 | F |
| 9 | PruebaC Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

Figura 31: Base de datos después de la prueba.

2.5. Quitar la IP pública a las máquinas virtuales dónde ejecuta Tomcat

Una vez que ya verificamos que funciona el servicio debemos de quitar la IP publica de las maquinas virtuales, mencionar que esto se puede hacer desde la creación de la maquina virtual, sin embargo, no podríamos hacer la prueba anterior. Para ello seleccionar la máquina virtual, seleccionar la IP pública, seleccionar la opción “Información general”, y seleccionar la opción “Desasociar”. Esto lo podemos ver en las figuras 32 y 33.

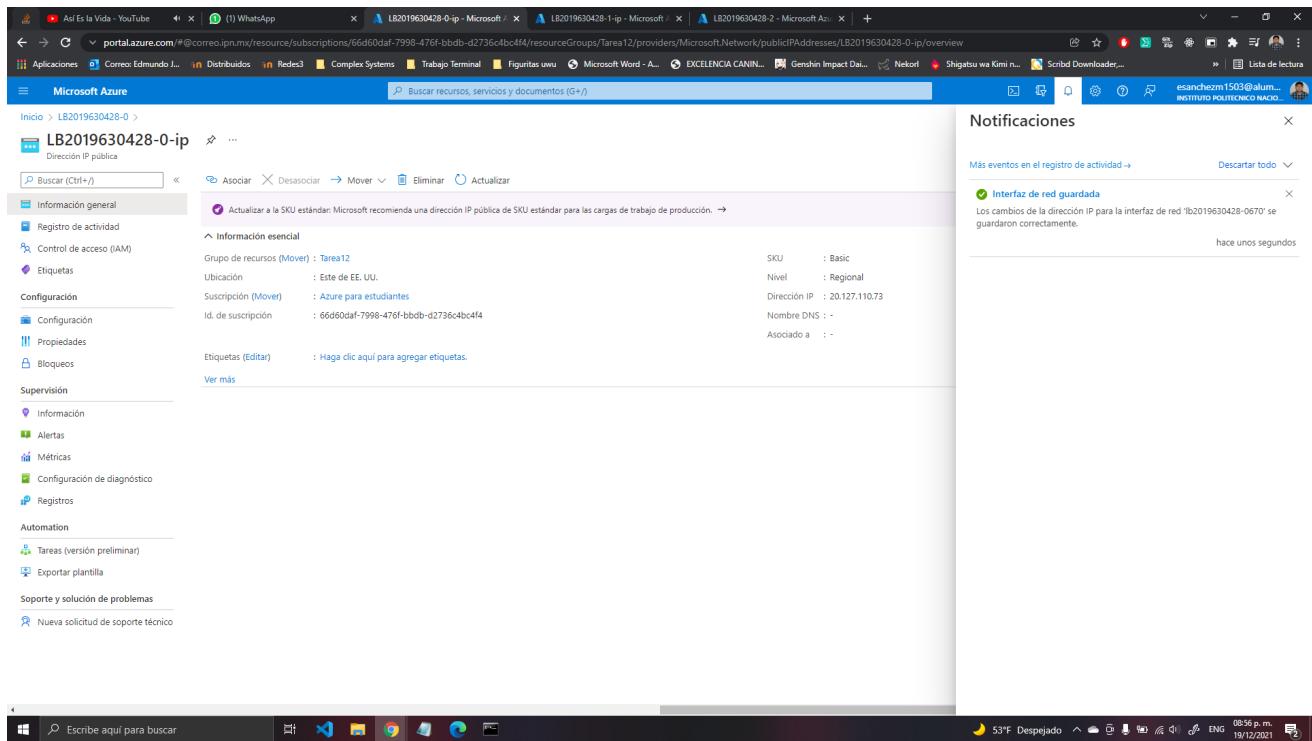


Figura 32: IP publica quitada en el nodo 0.

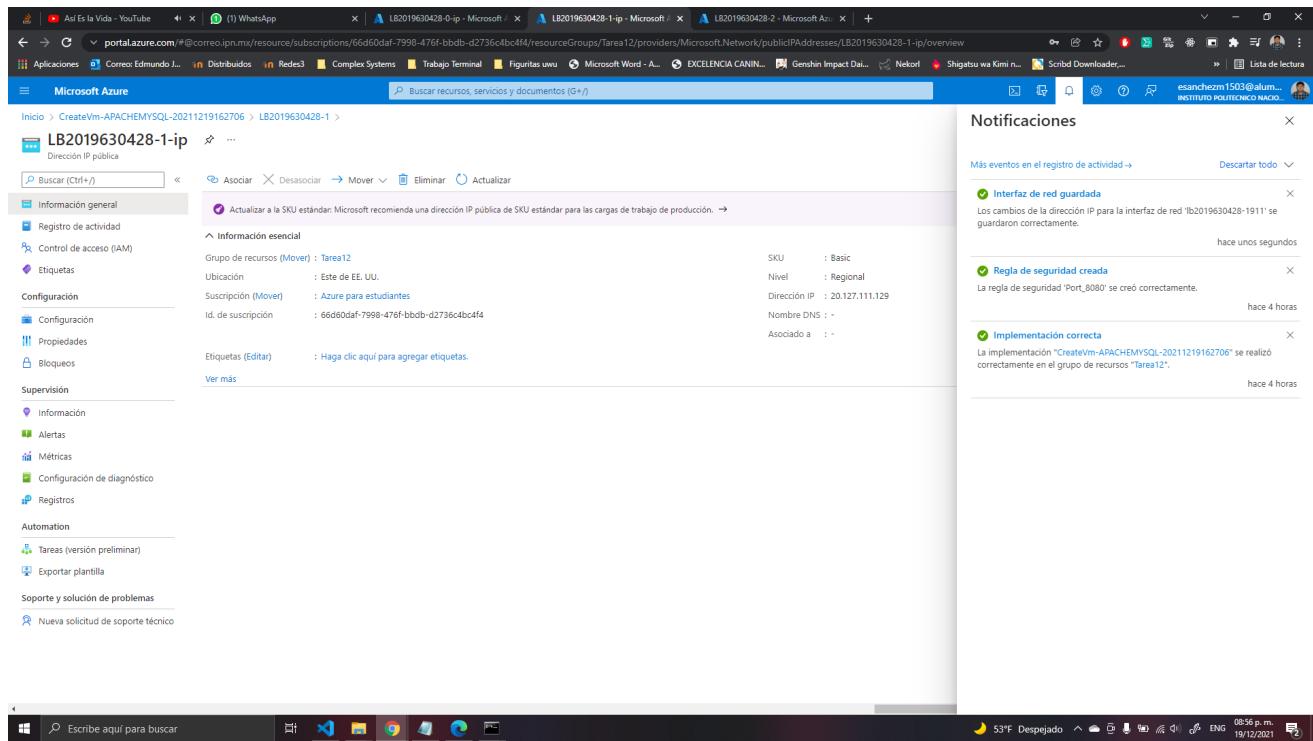


Figura 33: IP publica quitada en el nodo 1.

2.6. Creación de un balanceador de carga y conexión a las máquinas virtuales creadas en el paso 1

Ahora veamos la creación del balanceador de carga, veamos paso a paso.

2.6.1. Creación de un balanceador de carga en Azure

Para esto tenemos que ir Equilibradores de carga como vemos en la figura 34, después nos vamos a la opción de “+Crear” y ahora seleccionamos el grupo de recursos o bien, creamos un nuevo grupo de recursos, sin embargo, como ya tenemos nuestro grupo de recursos ya no es necesario, después ingresamos un nombre para la instancia del balanceador de carga, seleccionamos la región, seleccionamos el tipo de balanceador: Público, seleccionamos el SKU: Básico, seleccionamos el Nivel: Regional como vemos en la figura 35.

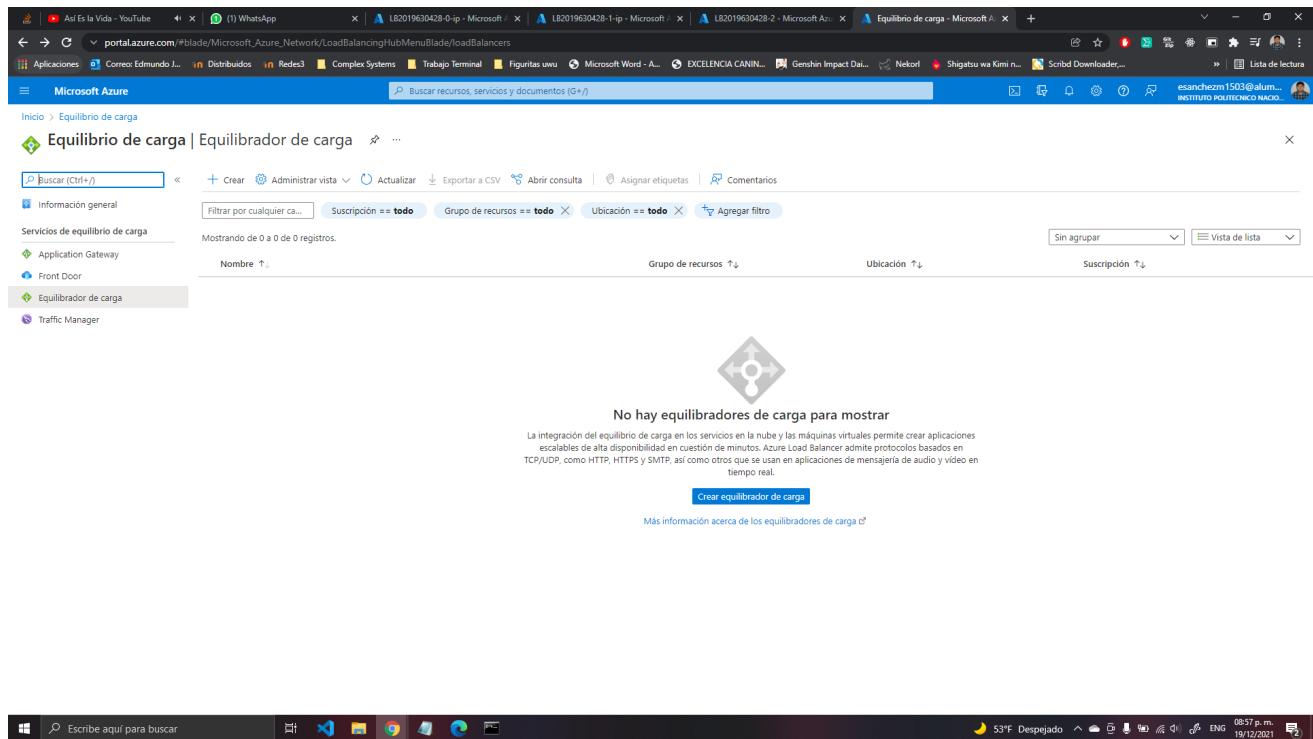


Figura 34: Opción de equilibrio de carga.

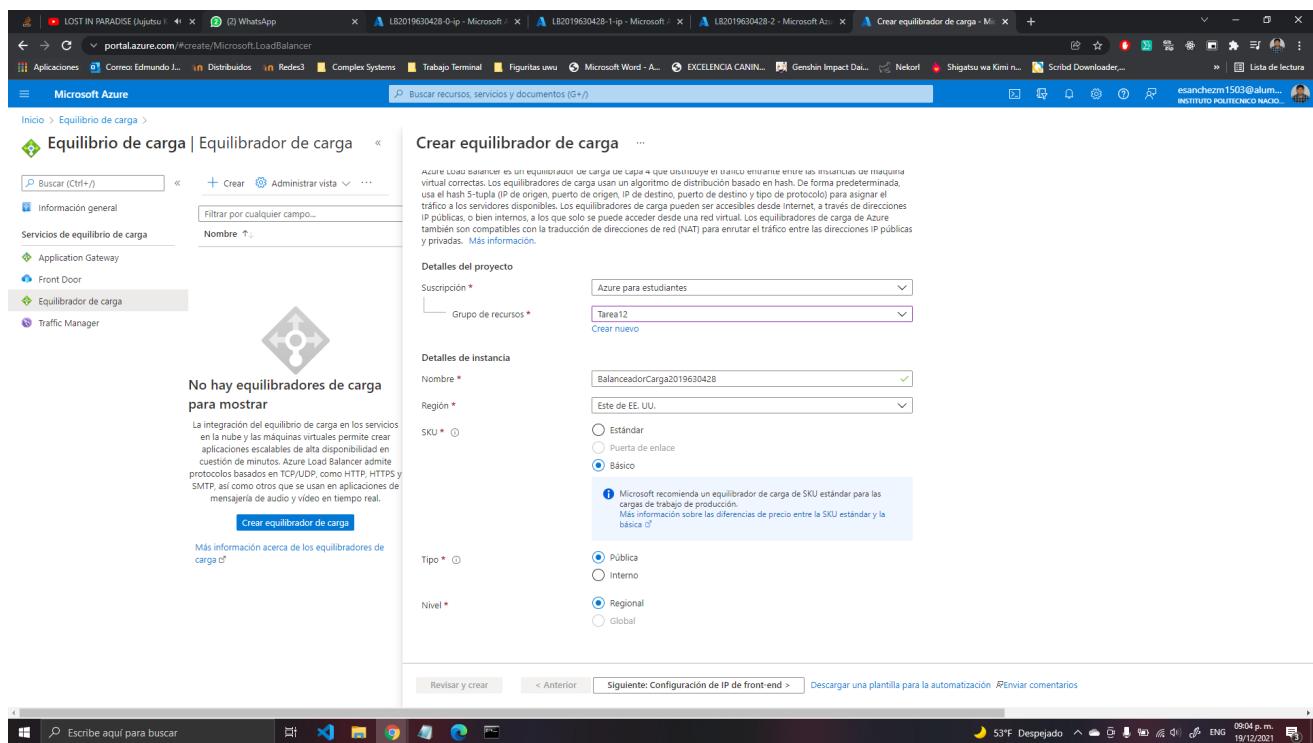


Figura 35: Creación de equilibrador de carga.

Continuando con el proceso, seleccionamos la opción “+ Agregar una configuración de IP de front-end” (Si aparece un error utilizar el navegador Edge.), ingresamos el nombre de la dirección IP pública y en el campo “versión de IP” seleccionamos “IPv4”, en el campo “Dirección de IP pública” seleccionar “Crear”,

ingresamos el nombre de la IP pública, en el campo “Asignación” seleccionar “Dinámica” como vemos en la figura 36.

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with 'Aplicaciones', 'Correos Edmundo J...', 'Distribuidos', 'Redes3', 'Complex Systems', 'Trabajo Terminal', 'Figuras uwu', 'Microsoft Word - A...', 'EXCELENCIA CANIN...', 'Genshin Impact Dau...', 'Nekofl', 'Shigatsu wa Kimi n...', 'Scribd Downloader...', and 'Lista de lectura'. The main navigation bar has tabs for 'Inicio', 'Equilibrio de carga', 'Equilibrador de carga', 'Front Door', 'Traffic Manager', and 'Equilibrador de carga' (selected). The main content area is titled 'Crear equilibrador de carga' and shows a summary of the configuration: 'Nombre': 'IPBalabecedario', 'Versión de IP': 'IPv4', 'Dirección IP pública': 'Elegir dirección IP pública'. A sub-modal window titled 'Agregar dirección IP de front...' is open, showing the 'Agregar una dirección IP pública' section with 'Nombre': 'IPBalabecedario', 'SKU': 'Básico', 'Nivel': 'Regional', and 'Asignación': 'Dinámica'. At the bottom right of the main window, there are 'Aceptar' and 'Cancelar' buttons.

Figura 36: Configuración de IP de front-end.

Finalmente en las figuras 37 y 38 vemos se añade la IP para el front-end y nos dice que esta por crearse y vemos la información general en la figura 38.

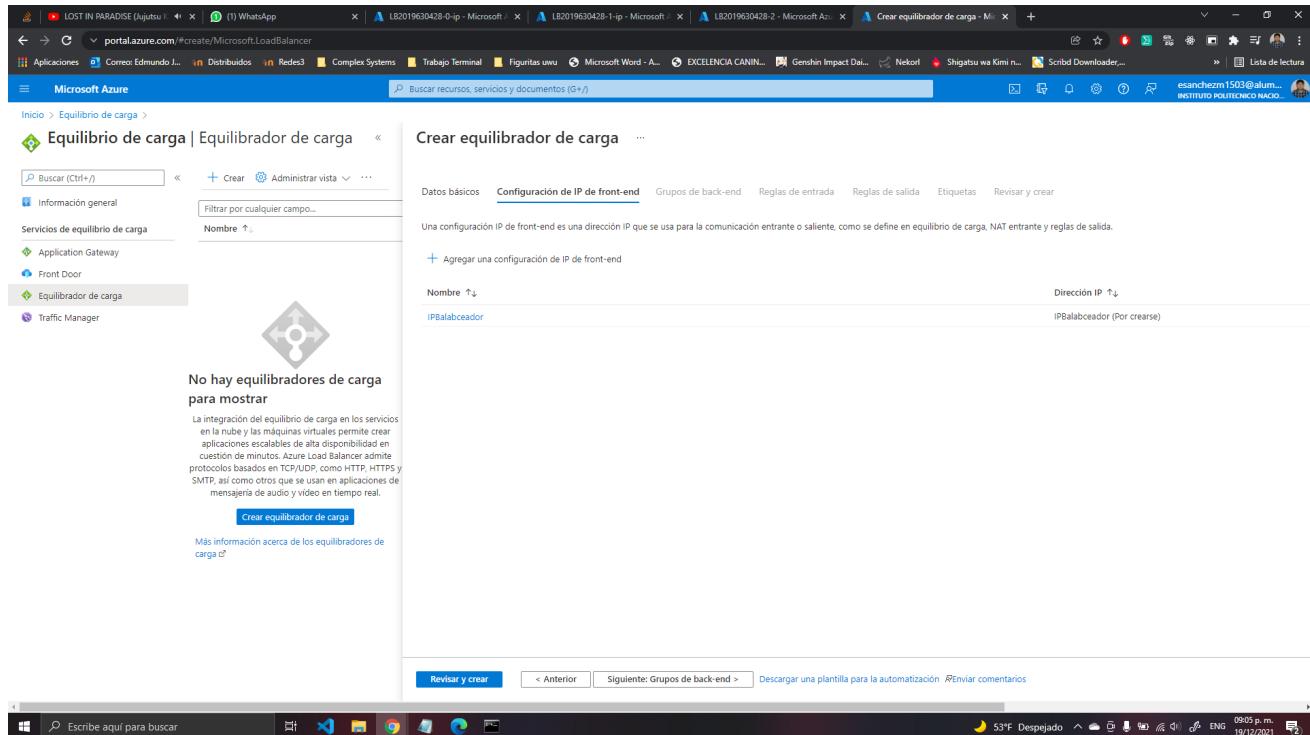


Figura 37: Información de la opción IP de front-end.

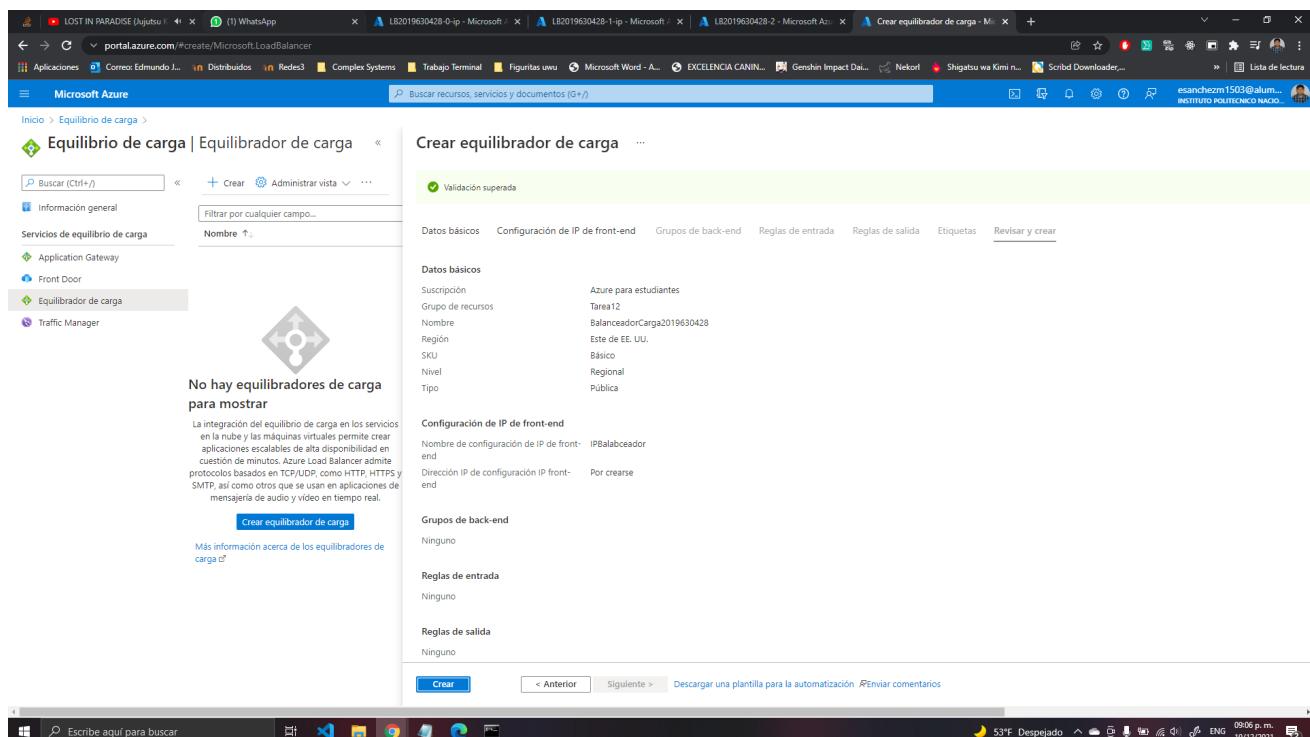


Figura 38: Revisar y crear.

Finalmente vemos como se creo el balanceador de manera correcta en la figura 39.

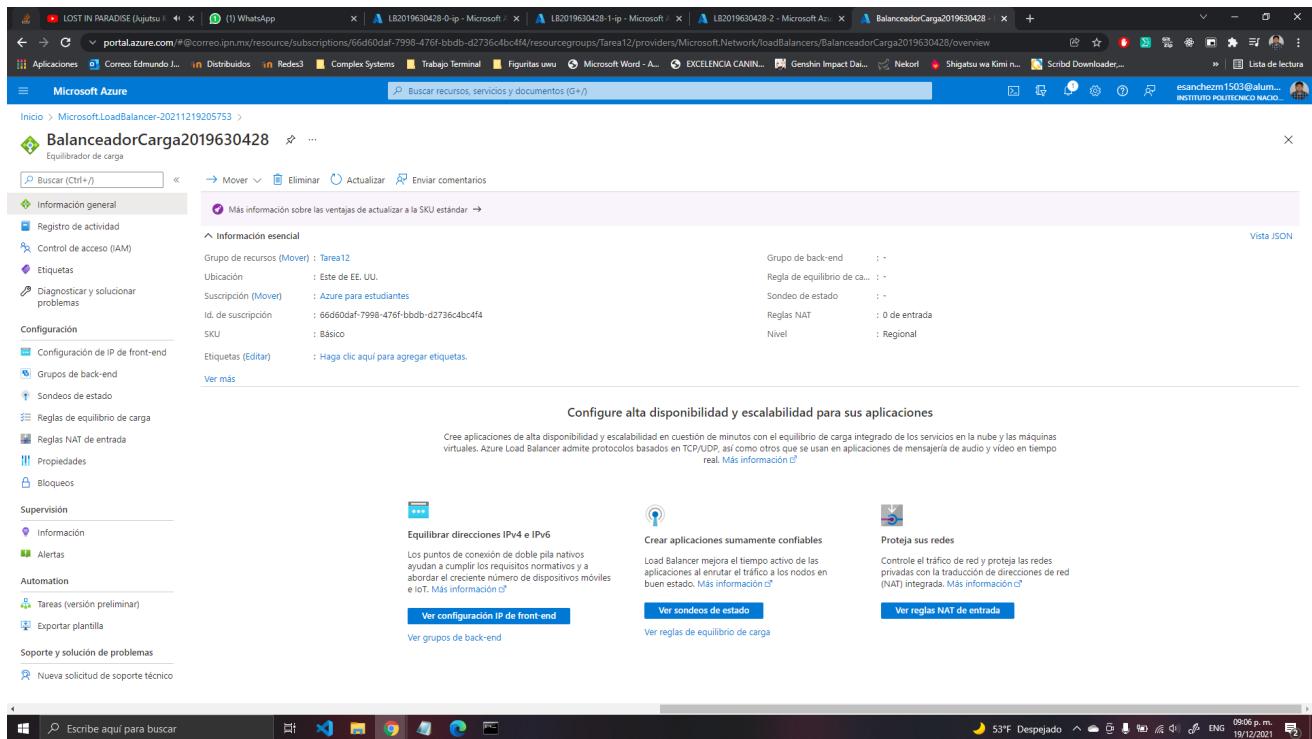


Figura 39: Creación del balanceador exitosa.

2.6.2. Configuración del balanceador de carga

Ahora estando en el balanceador de carga, agregaremos máquinas virtuales al balanceador de carga para esto seleccionamos la opción “Grupos de back-end” en la sección “Configuración” del menú que aparece a la izquierda de la pantalla, como vemos en la figura 40.

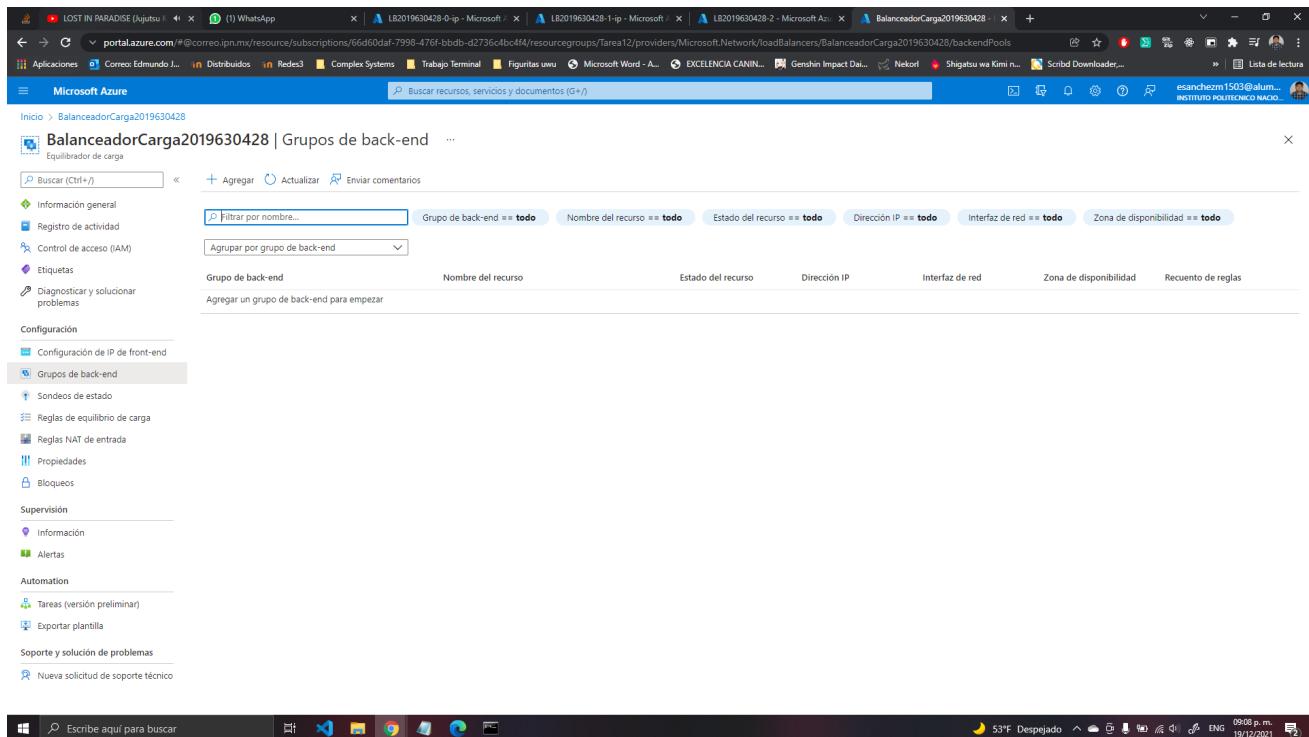


Figura 40: Opción grupos de back-end.

Después seleccionamos la opción “+Agregar”, ingresamos un nombre para el grupo de back-end, seleccionamos la red virtual donde están las máquinas virtuales, en el campo “Asociado a” seleccionar “Maquinas virtuales”, seleccionamos la versión de IP: IPv4, damos clic al botón “+Agregar” para agregar una máquina virtual al grupo back-end. **Nota importante. Las máquinas virtuales no deben tener IP pública y deben estar en la misma ubicación y red virtual que el balanceador de carga. Al crear cada máquina virtual seleccionar “Ninguno” en el campo “IP Pública” en la pestaña “Redes”.**, después marcamos la máquina virtual a agregar las cuales seran los nodos 0 y 1, damos clic al botón “Agregar”, esto lo podemos ver en la figura 41 con los nodos 0 y 1 agregados al balanceador de carga.

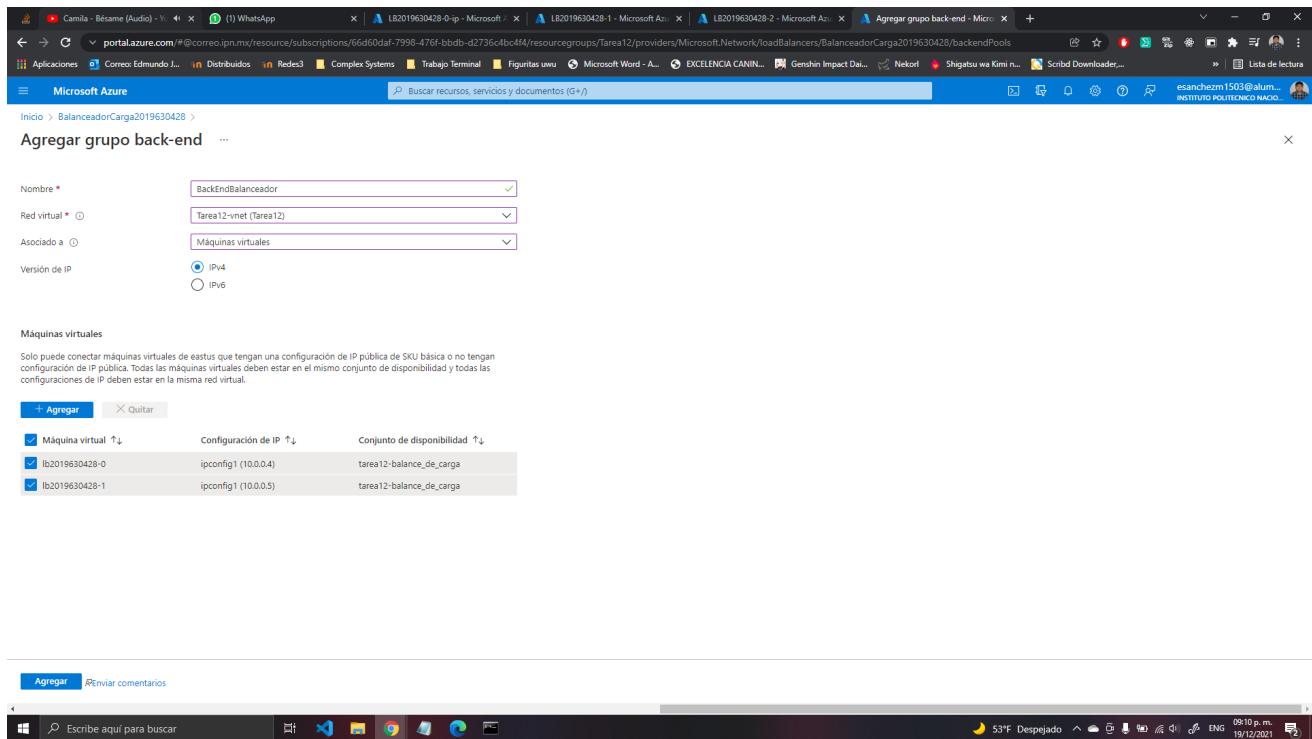


Figura 41: Maquinas agregadas al balanceador de carga.

Finalmente damos clic al botón “Agregar” y vemos como en la figura 42 el procedimiento fue realizado de manera correcta.

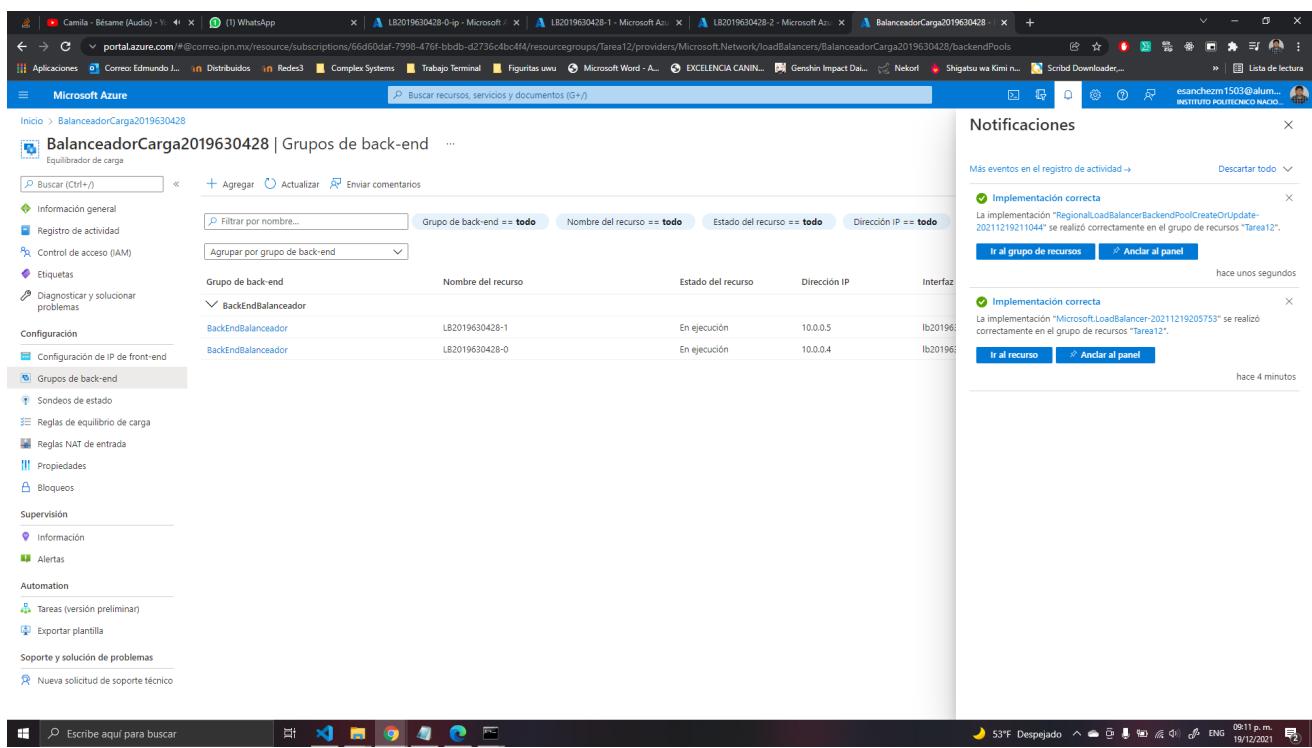


Figura 42: Maquinas agregadas al balanceador de carga de manera correcta.

2.6.3. Agregar un sondeo de estado

Antes de agregar reglas de equilibrio de carga es necesario crear al menos un sondeo de estado, ahora iremos a la opción “Sondeos de estado” en la sección “Configuración” del menú que aparece a la izquierda de la pantalla como vemos en la figura 43.

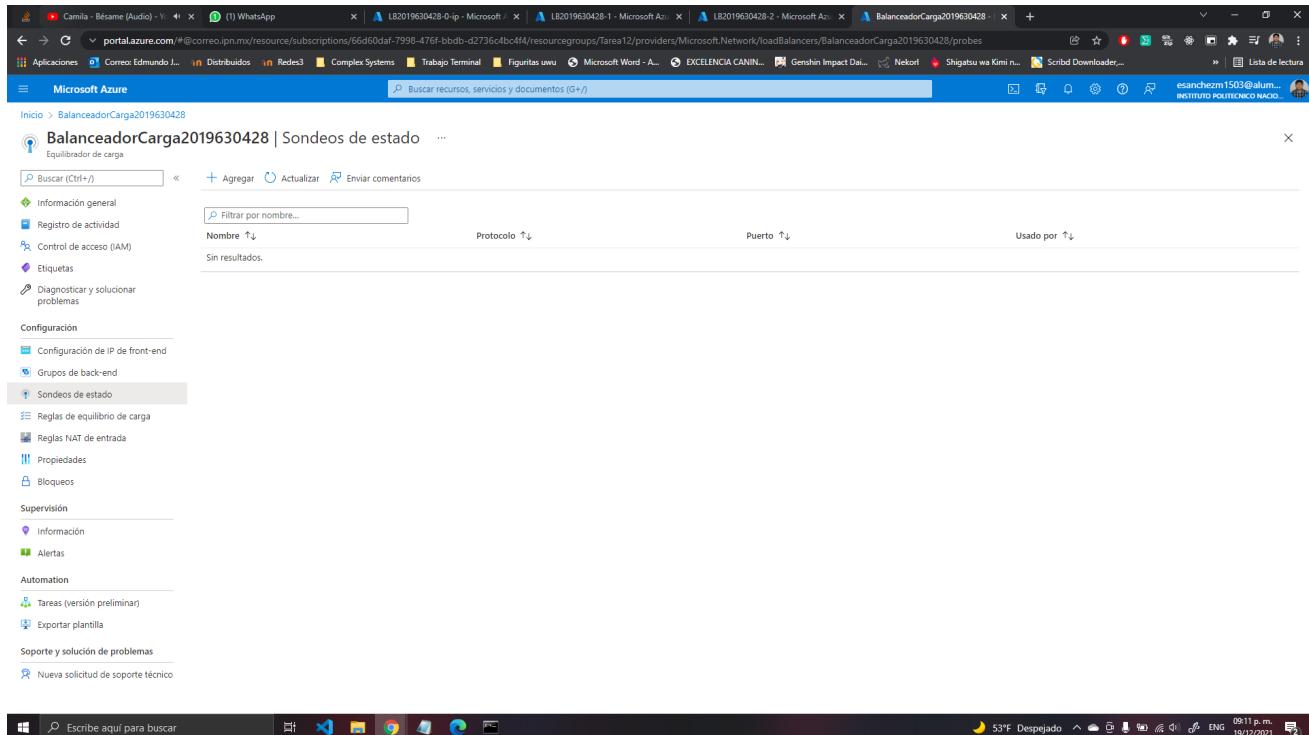


Figura 43: Opción sondeos de estado.

Ahora seleccionamos la opción “+Agregar”, ingresamos el nombre del sondeo, seleccionamos el protocolo: TCP, ingresamos el puerto que se sondeará: 8080, ingresamos el intervalo de sondeo en segundos, ingresar el umbral incorrecto (número de errores de sondeo consecutivos que indican que el estado de la máquina virtual no es correcto) y damos clic en el botón ^{Acepta} como vemos en la figura 44. **Nota. Se deberá abrir el puerto 8080 en cada máquina virtual para que se pueda hacer el sondeo de estado.**

The screenshot shows the 'Agregar sondeo de estado' (Add probe) page in the Microsoft Azure portal. The probe configuration is as follows:

- Nombre:** SondeoBalanceador
- Protocolo:** TCP
- Puerto:** 8080
- Intervalo:** 5 segundos
- Umbral incorrecto:** 2 errores consecutivos
- Usado por:** Sin usar

Figura 44: Creación de sondeo de estado.

Si vemos la figura 45 podemos ver como el sondeo de estado fue creado de manera correcta.

The screenshot shows the 'Sondeos de estado' (Probes) page in the Microsoft Azure portal. A single probe is listed:

Nombre	Protocolo	Puerto	Usado por
SondeoBalanceador	TCP	8080	-

Figura 45: Creación de sondeo de estado correcto.

2.6.4. Agregar una regla de equilibrio de carga

Finalmente iremos a la opción “Reglas de equilibrio de carga” como vemos en la figura 46.

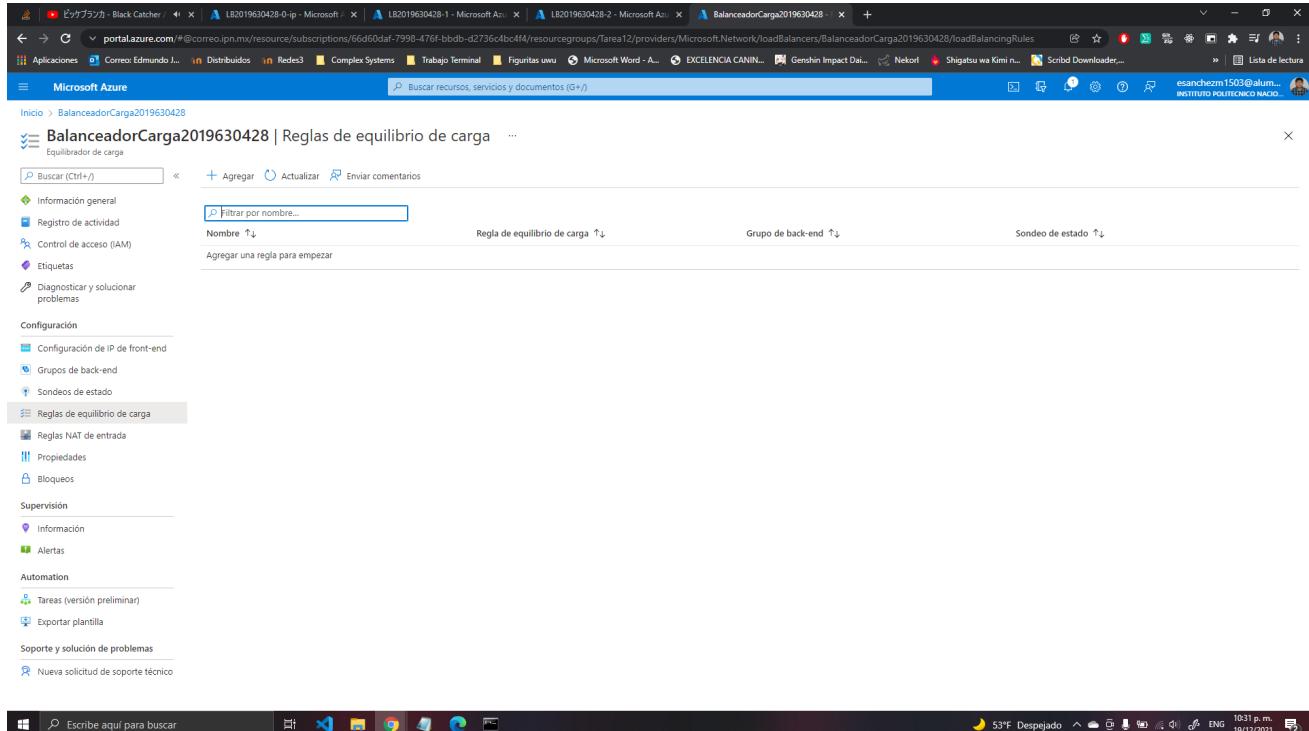


Figura 46: Opción reglas de equilibrio de carga.

Ingresamos el nombre de la regla, seleccionamos la versión de IP: IPv4, seleccionamos la dirección IP de fron-end (la IP del balanceador de carga), seleccionamos el Grupo de back-end (previamente creado), seleccionamos el protocolo: TCP, ingresamos el puerto (del balanceador de carga), por ejemplo: 80, ingresamos el puerto de back-end, por ejemplo: 8080

El puerto en las máquinas virtuales podría ser diferente al puerto que usan los clientes para conectarse con el balanceador de carga. Este puerto deberá estar abierto en todas las máquinas virtuales dentro del grupo back-en. Finalmente seleccionamos el sondeo de estado (previamente creado), todo lo anterior lo podemos ver en al figura 47 y damos clic en el botón “Aceptar” y vemos como en la figura 48 se creo de manera exitosa.

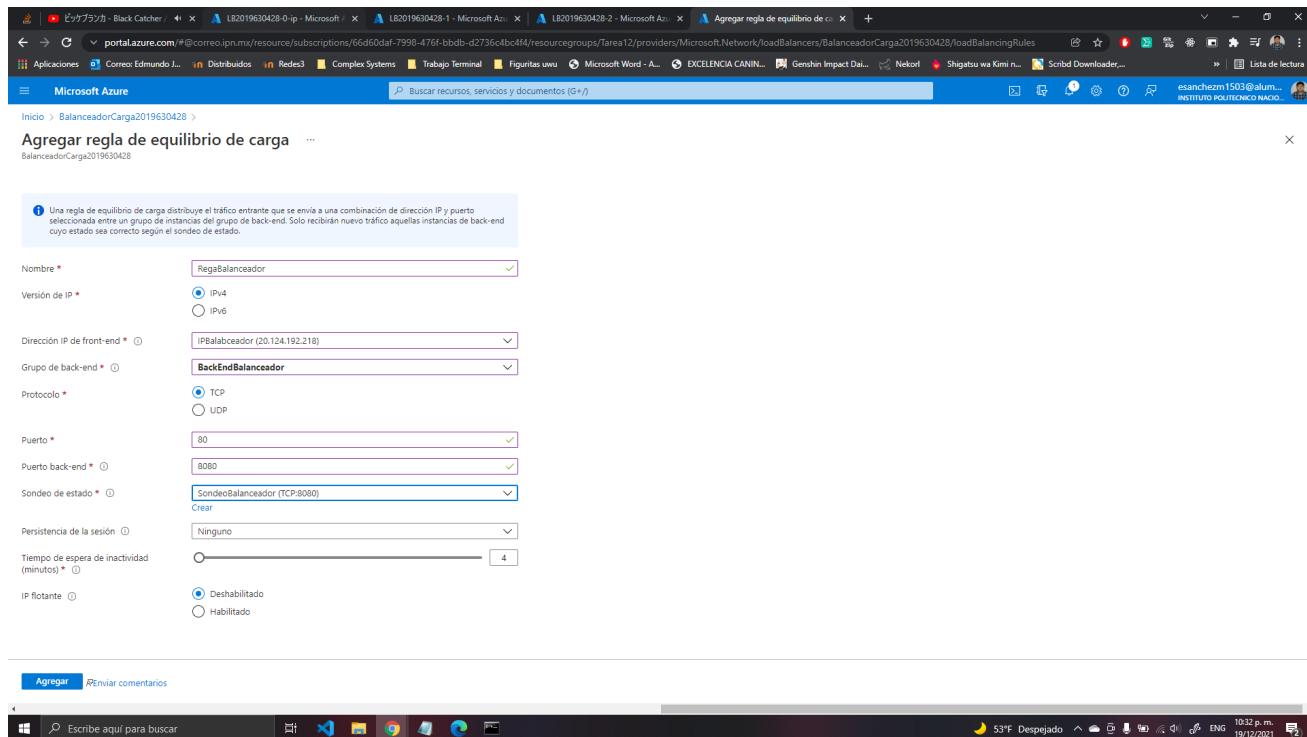


Figura 47: Creación de reglas de equilibrio de carga.

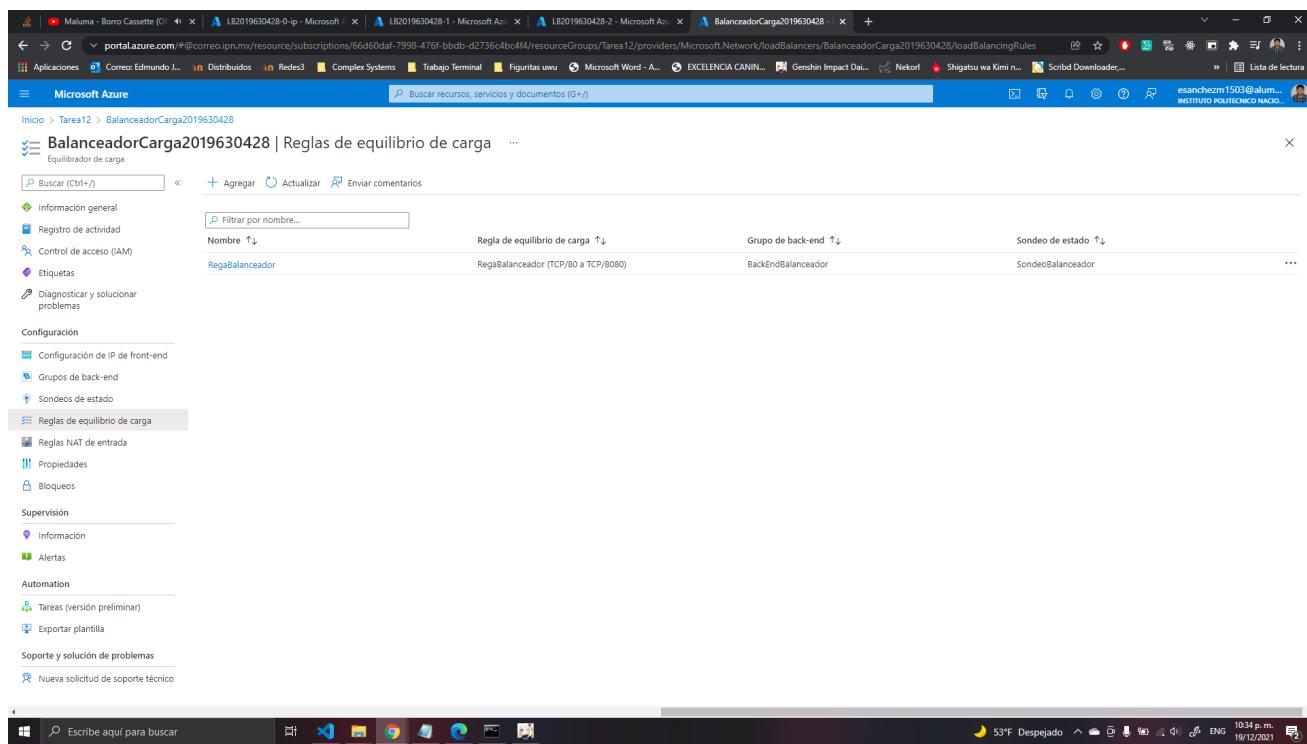


Figura 48: Creación de reglas de equilibrio de carga correcto.

Estos son todos los pasos que se tienen que hacer para la creación de nuestro balanceador de carga, ahora pasemos a las pruebas.

2.7. Pruebas

Antes de todo veamos el estado de base de datos como vemos en la figura 49.

```

edmundojm@B2019630428-2: ~ $ mysql -u hugo -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 24
Server version: 5.7.36-0ubuntu0.18.04.1 (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> use servicio_web
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from usuarios;
+----+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | prueba@nodo.ovo | owo | owo | owo | 0005-09-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | Prueba@Nodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | Prueba@Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+----+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>

```

Figura 49: Estado de la base de datos.

Ahora vayamos netamente a las pruebas. Para empezar nos dirigiremos a la URL <http://20.124.192.218/prueba.html> (ver figura 50) la cual es la IP del balanceador de carga esto para cargar una pagina HTML en la cual podemos hacer el clásico ABCC, es decir altas, bajas, cambios y consultas de una base de datos. Empecemos así con una alta, para esto veamos la figura 51 vemos los datos que serán dados de alta y en la figura 52 vemos el mensaje de que el usuario se dio de alta y vemos como en la figura 53 el alto se realizó de manera correcta.

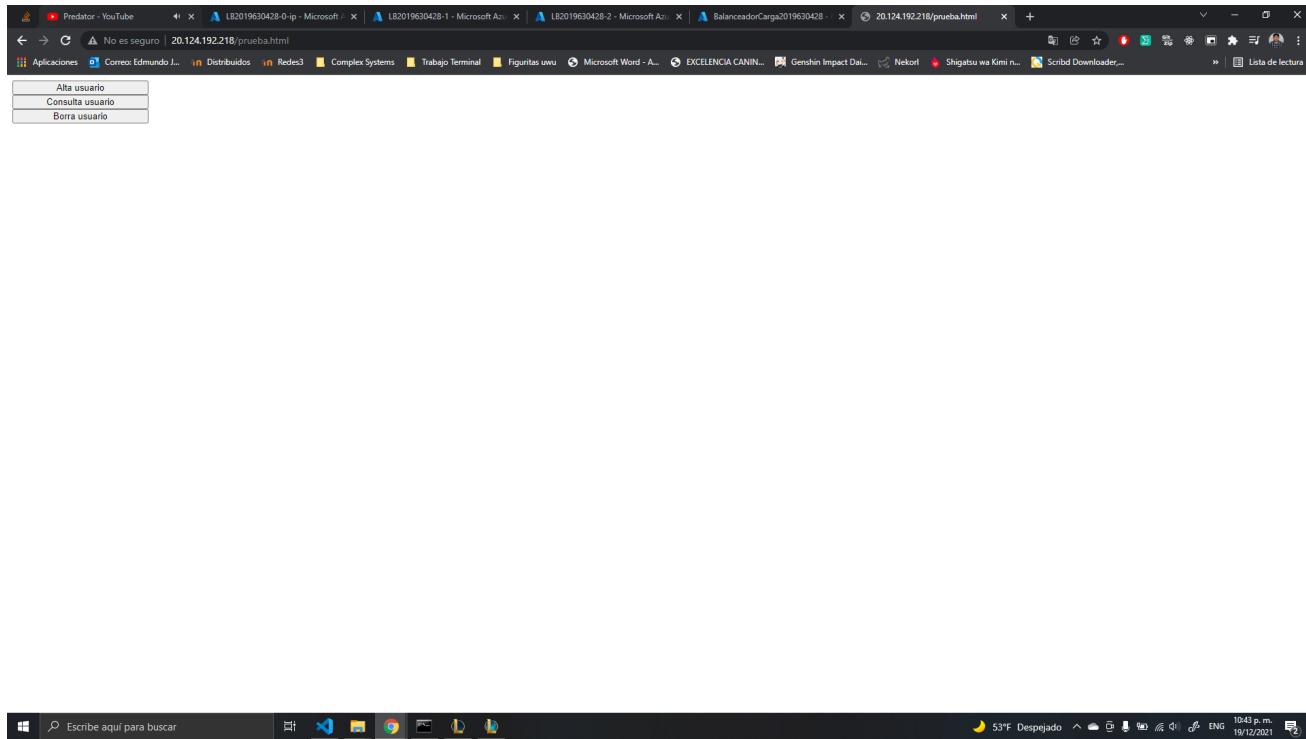


Figura 50: Pagina prueba.html.

A screenshot of a web form titled 'Alta de usuario'. The form fields include: Email * (balanceador@prueba.com), Nombre * (Warwick), Apellido paterno * (Ramirez), Apellido materno (Mendoza), Fecha de nacimiento * (05/08/2010), Telefono (775796125), Genero (Masculino), and a file upload field (Elegir archivos) containing the file '2uyj7hu2d4x31.jpg'. At the bottom are two buttons: 'Alta' and 'Regresa'.

Figura 51: Alta de usuario.

The screenshot shows a user registration form titled "Alta de usuario". The fields filled are: Email (balanceador@prueba.com), Nombre (Warwick), Apellido paterno (Ramirez), Apellido materno (Mendez), Fecha de nacimiento (05/08/2010), Telefono (7757996125), Genero (Masculino). A profile picture is uploaded. A confirmation message "El usuario se dio de alta" is displayed in a modal window. Buttons for "Aceptar" and "Regresa" are visible.

Figura 52: Alta de usuario correcto.

```

edmundojm@LB2019630428-2:~ 
mysql> select * from usuarios;
+----+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaC Nodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaC Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+----+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> select * from usuarios;
+----+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundodelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaC Nodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaC Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 10 | balanceador@prueba.com | Warwick | Ramirez | Mendez | 2010-08-05 | 7757996125 | M |
+----+-----+-----+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql>

```

Figura 53: Cambio en la base de datos después del registro.

La siguiente prueba sera dar de alta un usuario con el mismo correo que se ocupó, como vemos nos regresa un mensaje diciendo que el email ya existe como se ve en la figura 54.

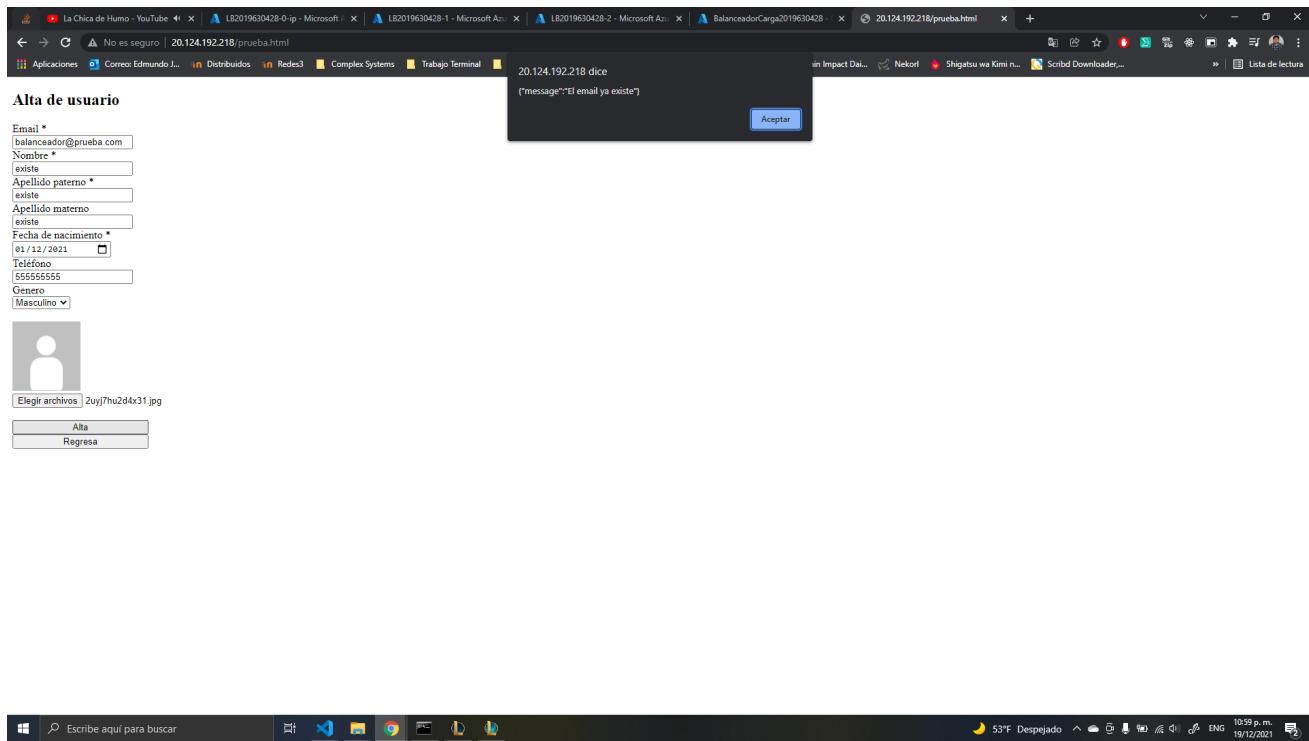


Figura 54: Alta de usuario con el mismo correo.

Una prueba mas sera la de consultar un usuario por medio del correo electrónico como vemos en la figura 55 solo colocamos el correo electrónico que se registro y como vemos en la figura 56 nos muestra la información que registramos en la figura 51.

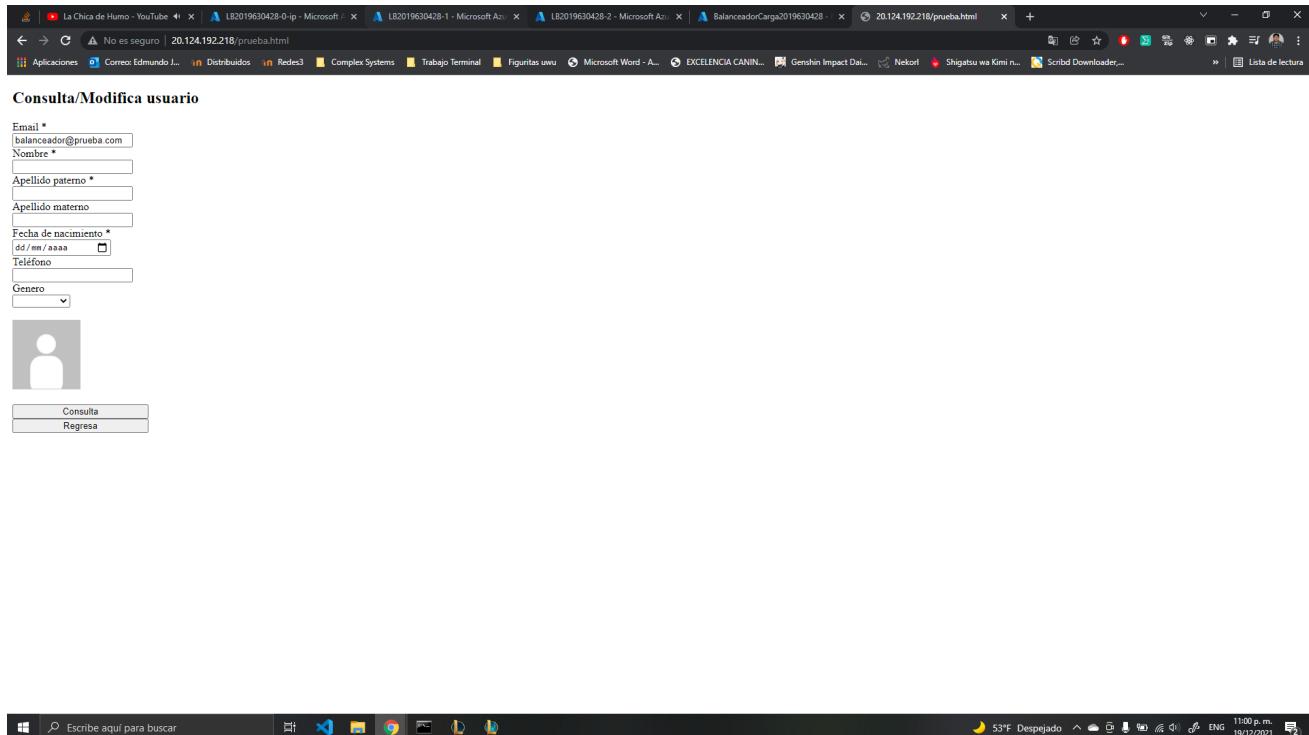


Figura 55: Consulta por medio del correo.

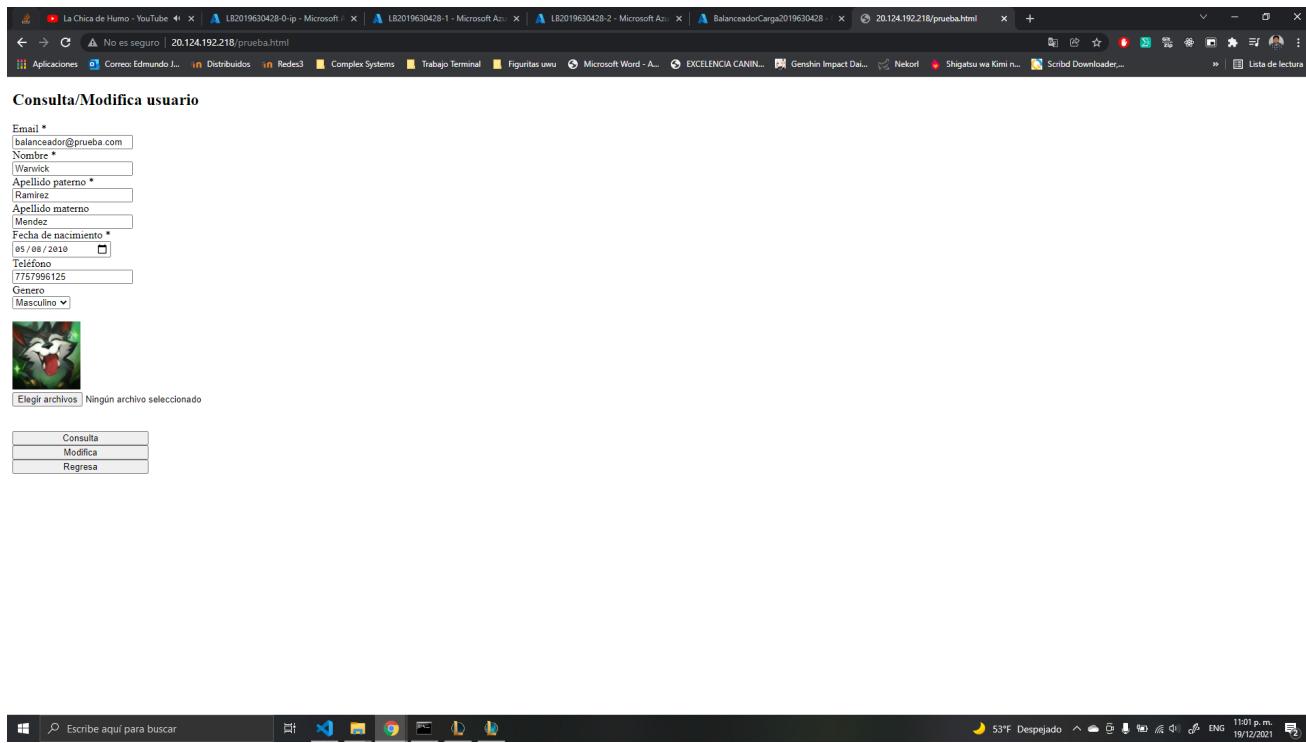


Figura 56: Resultado de la consulta anterior.

Como penúltima prueba sera modificar el usuario anteriormente registrado, en este caso solo modificamos el nombre completo del usuario y la foto para esto podemos ver la figura 57 y que nos devuelve un mensaje con el contenido “El usuario se modificó” y como vemos en la figura 58 al consultar el usuario tiene la información actualizada.

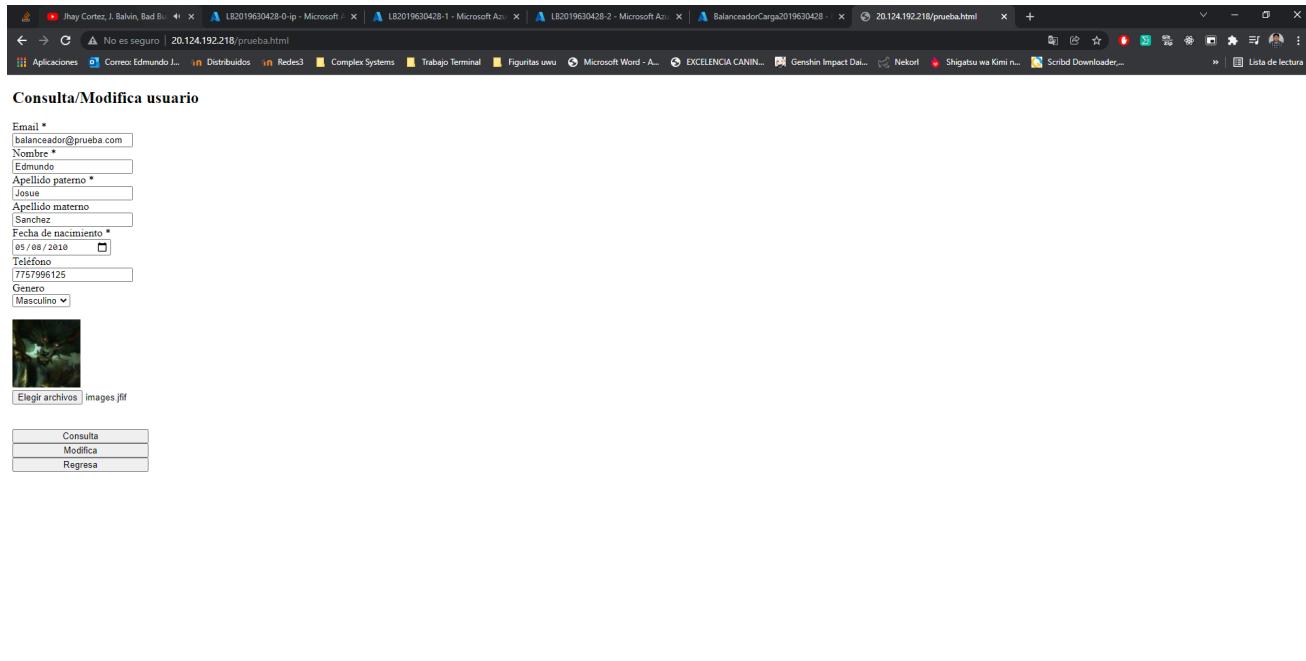


Figura 57: Cambio en la información del usuario.

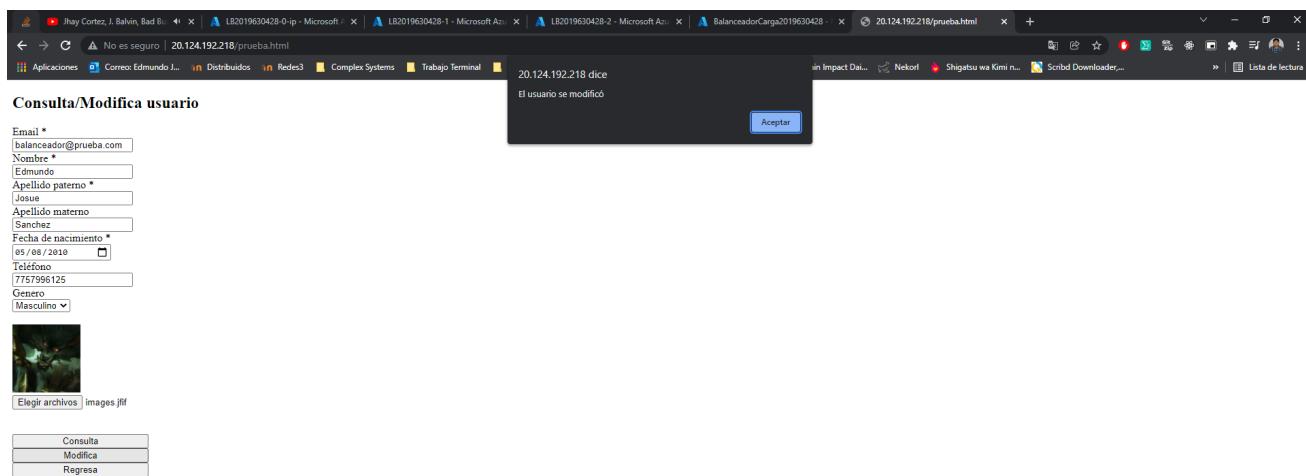


Figura 58: Resultado del cambio realizado.

En la figura 59 podemos ver la base de datos modificada después del cambio realizado en el front end

```
edmundojm@LB019630428-2:~
mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | Prueba@Nodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | Prueba@Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | Prueba@Nodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | Prueba@Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 10 | balanceador@prueba.com | Warwick | Ramirez | Mendez | 2010-08-05 | 7757996125 | M |
+-----+-----+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | Prueba@Nodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | Prueba@Nodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 10 | balanceador@prueba.com | Edmundo | Josue | Sanchez | 2010-08-05 | 7757996125 | M |
+-----+-----+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql>
```

Figura 59: Resultado del cambio realizado en la base de datos.

Como ultima prueba borraremos el usuario registrado por medio del correo que se registro, como vemos en la figura 60 nos arroja un mensaje con el contenido “El usuario se borro” y en la figura 61 vemos como al querer consultar después de que el usuario fue borrado nos arroja un mensaje de error diciendo que el email no existe.

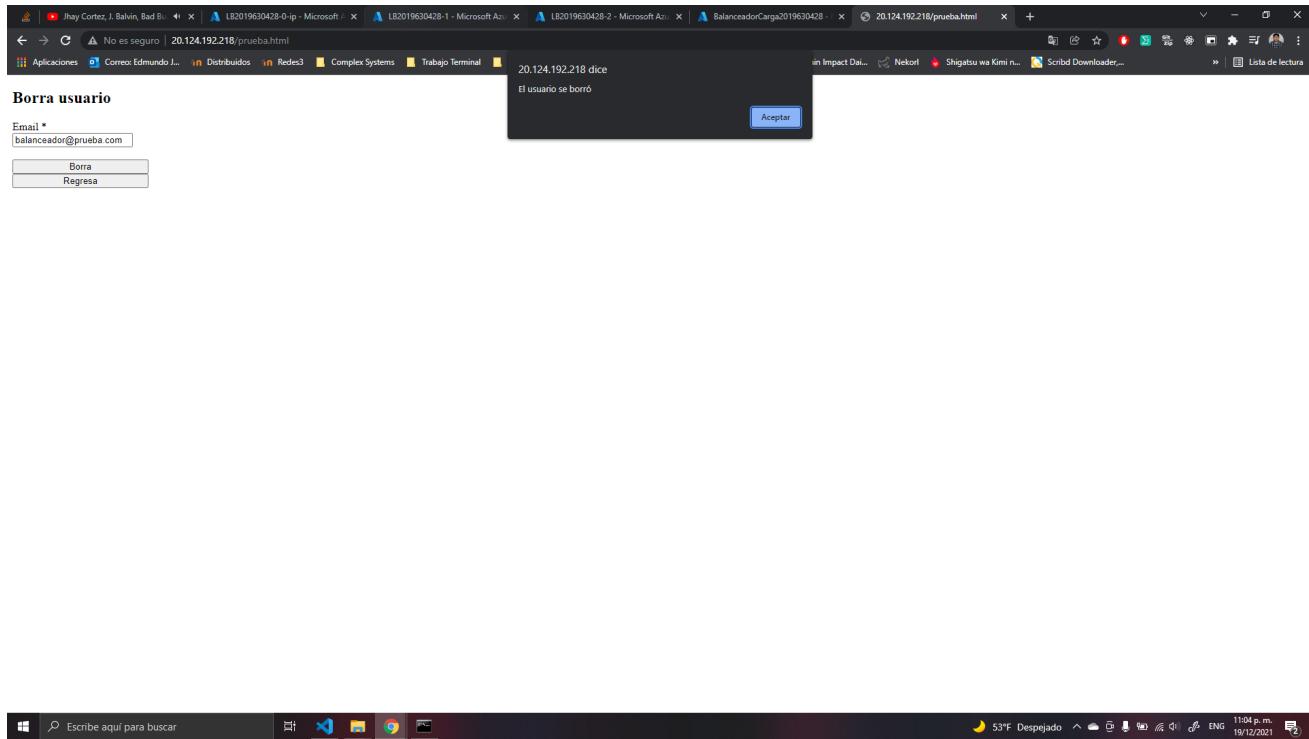


Figura 60: Eliminación del usuario.

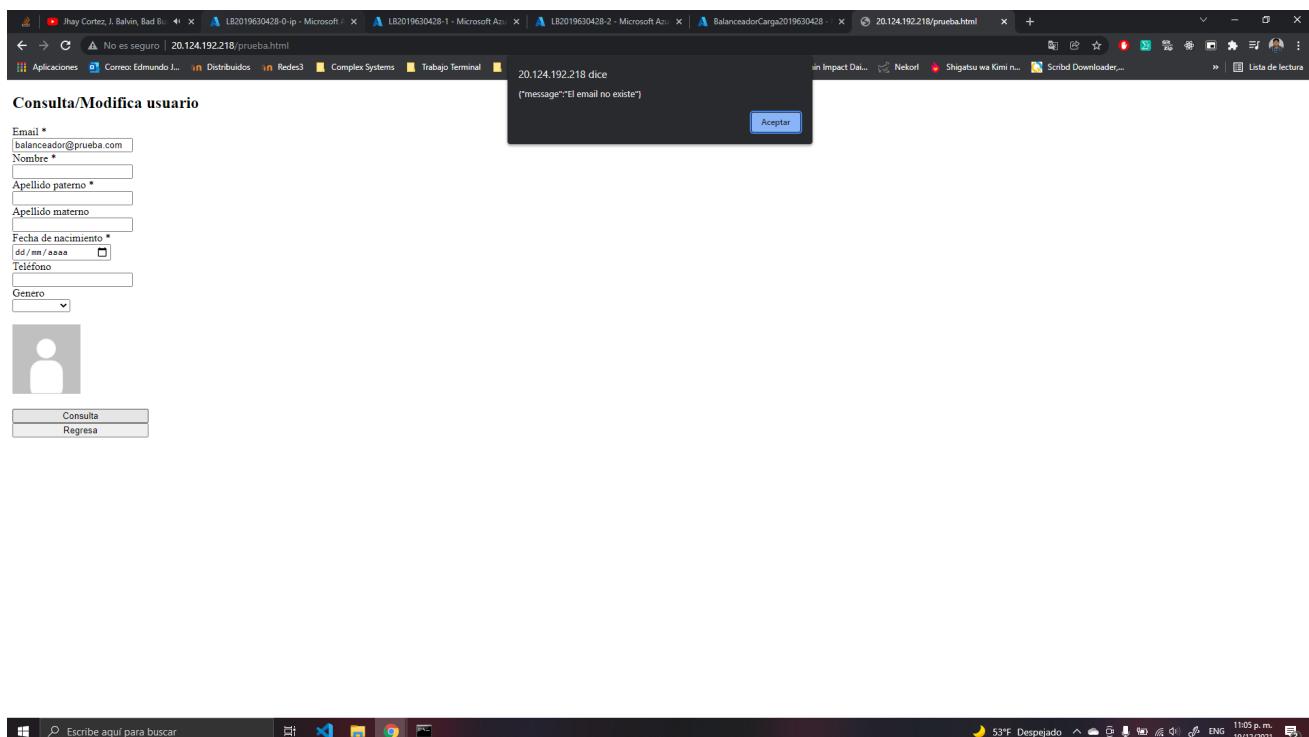


Figura 61: Consulta del usuario borrado en la figura anterior.

Vemos en la figura 62 como en la base de datos se borro de manera correcta

```
edmundojm@LB2019630423-2:~
```

8 rows in set (0.00 sec)

```
mysql> select * from usuarios;
```

id_usuario	email	nombre	apellido_paterno	apellido_materno	fecha_nacimiento	telefono	genero	
1	edmundoelpro1@gmail.com	Edmundo	Josue	Sanchez	Mendez	2020-09-15	5557329497	M
2	prueba	prueba	prueba	prueba	prueba	0001-01-01	1	M
4	prueba2	prueba2	prueba2	prueba2	prueba2	0001-01-01	12	F
5	uwu@uwu.uwu	uwu	uwu	uwu	uwu	0005-05-05	555	M
6	a	a	a	a	a	2021-12-03	5554660155	F
7	cambios	cambios	cambios	cambios	cambios	2021-12-01	5555	F
8	Prueba@Nodo0	0	0	0	0	0001-01-01	0	F
9	Prueba@Nodo1	a	a	a	a	2021-12-03	5554660155	F
10	balanceador@prueba.com	Warwick	Ramirez	Mendez	Mendez	2010-08-05	7757996125	M

9 rows in set (0.00 sec)

```
mysql> select * from usuarios;
```

id_usuario	email	nombre	apellido_paterno	apellido_materno	fecha_nacimiento	telefono	genero	
1	edmundoelpro1@gmail.com	Edmundo	Josue	Sanchez	Mendez	2020-09-15	5557329497	M
2	prueba	prueba	prueba	prueba	prueba	0001-01-01	1	M
4	prueba2	prueba2	prueba2	prueba2	prueba2	0001-01-01	12	F
5	uwu@uwu.uwu	uwu	uwu	uwu	uwu	0005-05-05	555	M
6	a	a	a	a	a	2021-12-03	5554660155	F
7	cambios	cambios	cambios	cambios	cambios	2021-12-01	5555	F
8	Prueba@Nodo0	0	0	0	0	0001-01-01	0	F
9	Prueba@Nodo1	a	a	a	a	2021-12-03	5554660155	F
10	balanceador@prueba.com	Edmundo	Josue	Sanchez	Sanchez	2010-08-05	7757996125	M

9 rows in set (0.00 sec)

```
mysql> select * from usuarios;
```

id_usuario	email	nombre	apellido_paterno	apellido_materno	fecha_nacimiento	telefono	genero	
1	edmundoelpro1@gmail.com	Edmundo	Josue	Sanchez	Mendez	2020-09-15	5557329497	M
2	prueba	prueba	prueba	prueba	prueba	0001-01-01	1	M
4	prueba2	prueba2	prueba2	prueba2	prueba2	0001-01-01	12	F
5	uwu@uwu.uwu	uwu	uwu	uwu	uwu	0005-05-05	555	M
6	a	a	a	a	a	2021-12-03	5554660155	F
7	cambios	cambios	cambios	cambios	cambios	2021-12-01	5555	F
8	Prueba@Nodo0	0	0	0	0	0001-01-01	0	F
9	Prueba@Nodo1	a	a	a	a	2021-12-03	5554660155	F

8 rows in set (0.00 sec)

```
mysql>
```

Figura 62: Base de datos modifica.

Finalmente haremos las mismas pruebas realizadas pero desde un teléfono inteligente, el orden sera el mismo que las pruebas anteriores, solo que esta vez no detallare mucho, sin embargo, cada imagen tendrá lo que ahí se hizo.



Figura 63: Pagina prueba.html.

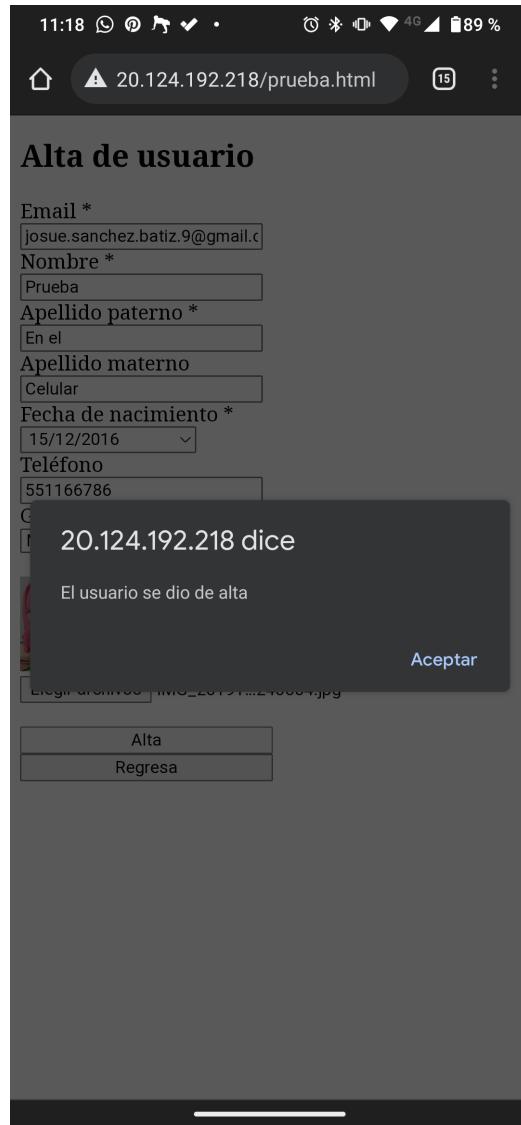


Figura 64: Alta de usuario.

```
edmundojm@LB2019630428-2: ~
mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaNodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaNodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 10 | balanceador@prueba.com | Edmundo | Josue | Sanchez | 2010-08-05 | 7757996125 | M |
+-----+-----+-----+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaNodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaNodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+-----+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaNodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaNodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 10 | balanceador@prueba.com | Warwick | Ramirez | Mendez | 2010-08-05 | 7757996125 | M |
| 12 | josue.sanchez.batiza.9@gmail.com | Prueba | En el | Celular | 2016-12-15 | 551166786 | M |
+-----+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>
```

Figura 65: Cambio en la base de datos después del registro.

Figura 66: Consulta de usuario.

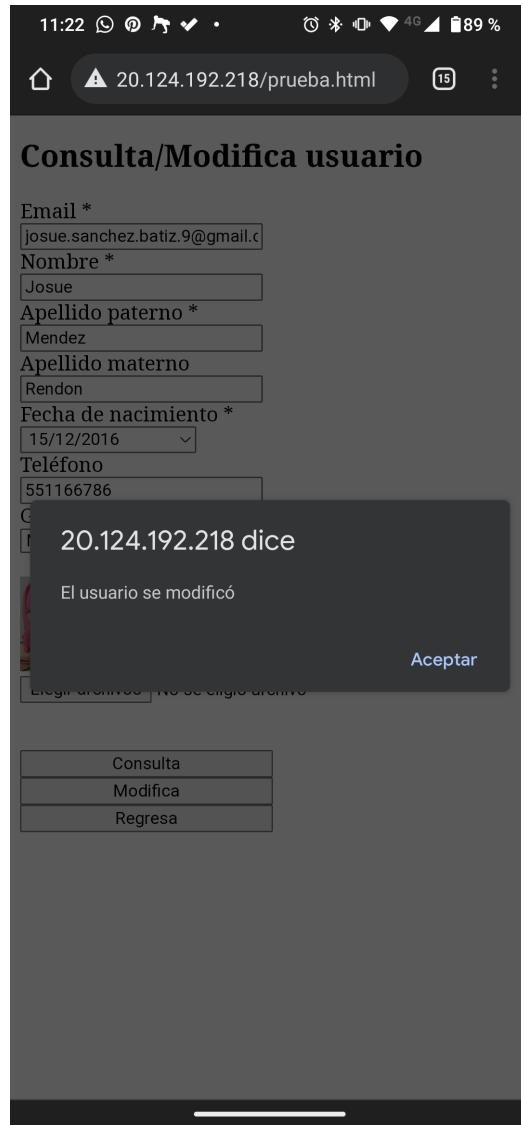


Figura 67: Modificación de usuario.

```
edmundojm@LB019630428-2:~
mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaNodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaNodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaNodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaNodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 11 | balanceador@prueba.com | Warwick | Ramirez | Mendez | 2010-08-05 | 7757996125 | M |
| 12 | josue.sanchez.batiz.9@gmail.com | Prueba | En el | Celular | 2016-12-15 | 551166786 | M |
+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> select * from usuarios;
+-----+-----+-----+-----+-----+-----+-----+
| id_usuario | email | nombre | apellido_paterno | apellido_materno | fecha_nacimiento | telefono | genero |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | edmundoelpro1@gmail.com | Edmundo Josue | Sanchez | Mendez | 2020-09-15 | 5557329497 | M |
| 2 | prueba | prueba | prueba | prueba | 0001-01-01 | 1 | M |
| 4 | prueba2 | prueba2 | prueba2 | prueba2 | 0001-01-01 | 12 | F |
| 5 | uwu@uwu.uwu | uwu | uwu | uwu | 0005-05-05 | 555 | M |
| 6 | a | a | a | a | 2021-12-03 | 5554660155 | F |
| 7 | cambios | cambios | cambios | cambios | 2021-12-01 | 5555 | F |
| 8 | PruebaNodo0 | 0 | 0 | 0 | 0001-01-01 | 0 | F |
| 9 | PruebaNodo1 | a | a | a | 2021-12-03 | 5554660155 | F |
| 11 | balanceador@prueba.com | Warwick | Ramirez | Mendez | 2010-08-05 | 7757996125 | M |
| 12 | josue.sanchez.batiz.9@gmail.com | Josue | Mendez | Rendon | 2016-12-15 | 551166786 | M |
+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>
```

Figura 68: Cambio en la base de datos después del cambio.

11:22 ☺ ⚡ 🔍 ✓ ⚡ ⚡ * 4G 88 %

20.124.192.218/prueba.html [15] ⋮

Consulta/Modifica usuario

Email *

Nombre *

Apellido paterno *

Apellido materno

Fecha de nacimiento *

Teléfono

Género



Elegir archivos No se eligió archivo

Consulta
Modifica
Regresa

Figura 69: Nueva información de usuario.

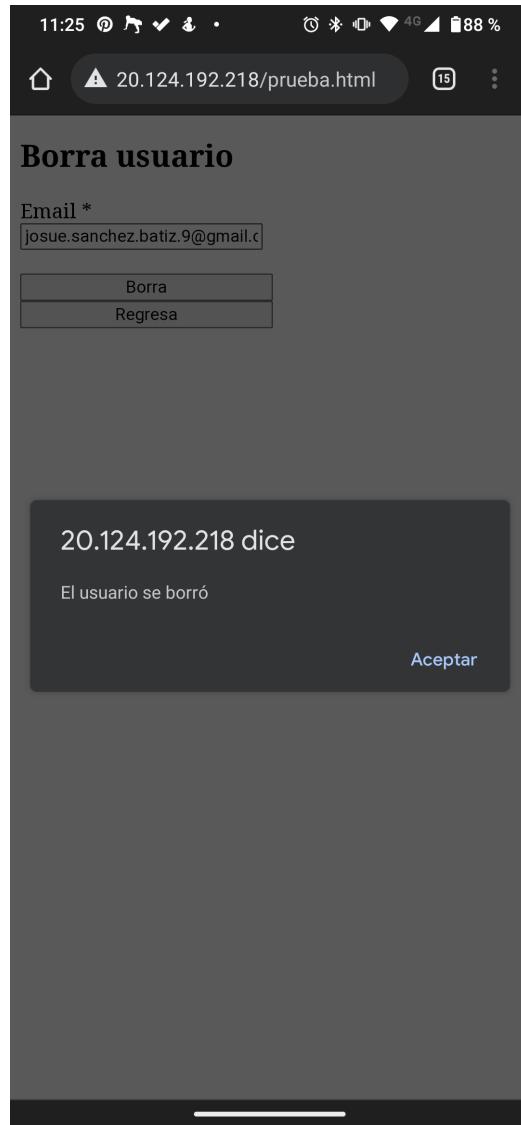


Figura 70: Baja de usuario.

```
edmundojm@LB2019630428-2:~
```

<u>id_usuario</u>	<u>email</u>	<u>nombre</u>	<u>apellido_paterno</u>	<u>apellido_materno</u>	<u>fecha_nacimiento</u>	<u>telefono</u>	<u>genero</u>
1	edmundoeelpro1@gmail.com	Edmundo Josue	Sanchez	Mendez	2020-09-15	5557329497	M
2	prueba	prueba	prueba	prueba	0001-01-01	1	M
4	prueba2	prueba2	prueba2	prueba2	0001-01-01	12	F
5	uwu@uwu.uwu	uwu	uwu	uwu	0005-05-05	555	M
6	a	a	a	a	2021-12-03	5554660155	F
7	cambios	cambios	cambios	cambios	2021-12-01	5555	F
8	PruebaC Nodo0	Ø	Ø	Ø	0001-01-01	Ø	F
9	PruebaC Nodo1	a	a	a	2021-12-03	5554660155	F
11	balanceador@prueba.com	Warwick	Ramirez	Mendez	2010-08-05	7757996125	M
12	josue.sanchez.batiz.9@gmail.com	Prueba	En el	Celular	2016-12-15	551166786	M

10 rows in set (0.00 sec)

```
mysql> select * from usuarios;
```

<u>id_usuario</u>	<u>email</u>	<u>nombre</u>	<u>apellido_paterno</u>	<u>apellido_materno</u>	<u>fecha_nacimiento</u>	<u>telefono</u>	<u>genero</u>
1	edmundoeelpro1@gmail.com	Edmundo Josue	Sanchez	Mendez	2020-09-15	5557329497	M
2	prueba	prueba	prueba	prueba	0001-01-01	1	M
4	prueba2	prueba2	prueba2	prueba2	0001-01-01	12	F
5	uwu@uwu.uwu	uwu	uwu	uwu	0005-05-05	555	M
6	a	a	a	a	2021-12-03	5554660155	F
7	cambios	cambios	cambios	cambios	2021-12-01	5555	F
8	PruebaC Nodo0	Ø	Ø	Ø	0001-01-01	Ø	F
9	PruebaC Nodo1	a	a	a	2021-12-03	5554660155	F
11	balanceador@prueba.com	Warwick	Ramirez	Mendez	2010-08-05	7757996125	M
12	josue.sanchez.batiz.9@gmail.com	Josue	Rendon	Rendon	2016-12-15	551166786	M

10 rows in set (0.00 sec)

```
mysql> select * from usuarios;
```

<u>id_usuario</u>	<u>email</u>	<u>nombre</u>	<u>apellido_paterno</u>	<u>apellido_materno</u>	<u>fecha_nacimiento</u>	<u>telefono</u>	<u>genero</u>
1	edmundoeelpro1@gmail.com	Edmundo Josue	Sanchez	Mendez	2020-09-15	5557329497	M
2	prueba	prueba	prueba	prueba	0001-01-01	1	M
4	prueba2	prueba2	prueba2	prueba2	0001-01-01	12	F
5	uwu@uwu.uwu	uwu	uwu	uwu	0005-05-05	555	M
6	a	a	a	a	2021-12-03	5554660155	F
7	cambios	cambios	cambios	cambios	2021-12-01	5555	F
8	PruebaC Nodo0	Ø	Ø	Ø	0001-01-01	Ø	F
9	PruebaC Nodo1	a	a	a	2021-12-03	5554660155	F
11	balanceador@prueba.com	Warwick	Ramirez	Mendez	2010-08-05	7757996125	M

9 rows in set (0.00 sec)

```
mysql>
```

Figura 71: Cambio en la base de datos después de borrado de usuario.

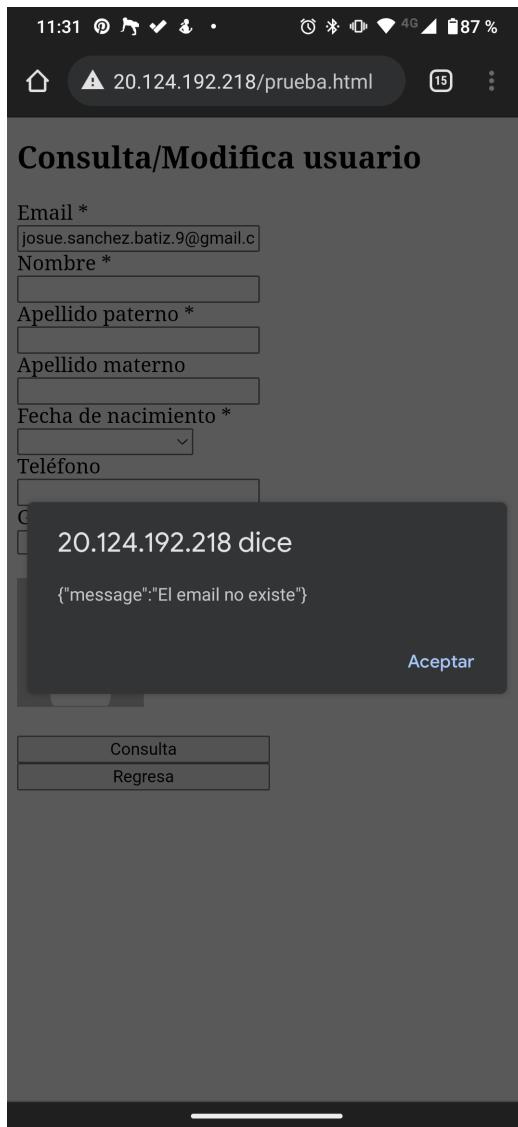


Figura 72: Consulta de usuario después de borrarlo.

Como vemos podemos usar este servicio tanto en una computadora de escritorio o laptop como en un dispositivo inteligente, ya sea una tableta o un celular.

3. Conclusiones

En esta practica pudimos ver el servicio de balanceador de carga de Azure, el cual es muy útil ya que resulta muy sencillo redirigir las peticiones a múltiples servidores utilizando un balanceador de carga, ademas, esto nos permite escalar las aplicaciones e implementar servicios con alta disponibilidad dentro de una zona y a través de diferentes zonas, para esto necesitamos agregar máquinas virtuales al balanceador de carga los sistemas incrementan su capacidad de atender peticiones, así mismo, se evita la interrupción del servicio cuándo uno de los servidores falla. Esta practica es muy útil en la vida laboral ya que cada vez la mayoría de las empresas ocupan servicios en la nube y ofrecer un servicio como lo es el balanceador de carga es demasiado útil para cualquier empresa ya que al poder escalar podemos escalar junto con la empresa que es el objetivo de cualquier empresa.

Referencias

Ramos, J., 2021. Habilitar conexión remota a MySQL en 3 pasos. [online] Programacionymas.com. Available at: <https://programacionymas.com/blog/mysql-habilitar-acceso-remoto> [Accessed 19 December 2021].

AJAX post error : Refused to set unsafe header “Connection”. [online] Stack Overflow. Available at: <https://stackoverflow.com/questions/7210507/ajax-post-error-refused-to-set-unsafe-header-connection> [Accessed 19 December 2021].