

Desarrollo del Sistema de Reservas de Hotel

Software Engineering

Yadiel J Torres Quiles

12 de diciembre del 2024

Universidad Interamericana de Arecibo

Table de Contenido

1. Resumen Ejecutivo	Page 2
2. Objetivos SMART	Page 3
3. Miembros	Page 4
4. Backlog del Producto	Page 5
5. Diseños del Proyecto	Page 6
6. Código Documentado	Page 7
7. Capturas de Pantalla	Page 10
8. Retrospectiva	Page 14

Resumen Ejecutivo

El objetivo de este proyecto es diseñar un sistema que permita a los clientes realizar reservaciones a través de una página web, utilizando sistema distribuido en tres máquinas virtuales (VMs). La primera máquina virtual (VM1) se encarga del hosting de la página web, gestionando la interfaz de usuario y la presentación de datos. La segunda máquina virtual (VM2) se dedica a la lógica de la aplicación, procesamiento de comandos y datos solicitados mientras actuando como intermediaria entre la VM1 y la VM3. Por último, la tercera máquina virtual (VM3) contiene la base de datos del sistema, junto con la lógica necesaria para su gestión y funcionamiento.

Objetivos Smart

1. Específico- Desarrollar un sistema de reservas de hotel que incluya funcionalidades para que los usuarios puedan buscar habitaciones disponibles, registrarse o iniciar sesión, realizar reservas y consultar sus reservas existentes.
2. Medibles- El sistema debe registrar todos los datos de las reservas en una base de datos con exactitud y permitir que los usuarios interactúen con la plataforma en tiempo real.
3. Alcanzables- Se debe implementar una solución que permita la comunicación fluida entre el Front End del sistema de reservas y la base de datos.
4. Relevantes- Garantizar que los usuarios puedan acceder la información correcta sobre la disponibilidad de las habitaciones en tiempo real y completar reservas.
5. Tiempo definido- Se debe completar en el tiempo ofrecido de alrededor de 3 semanas con una tarea completada cada 2 días.

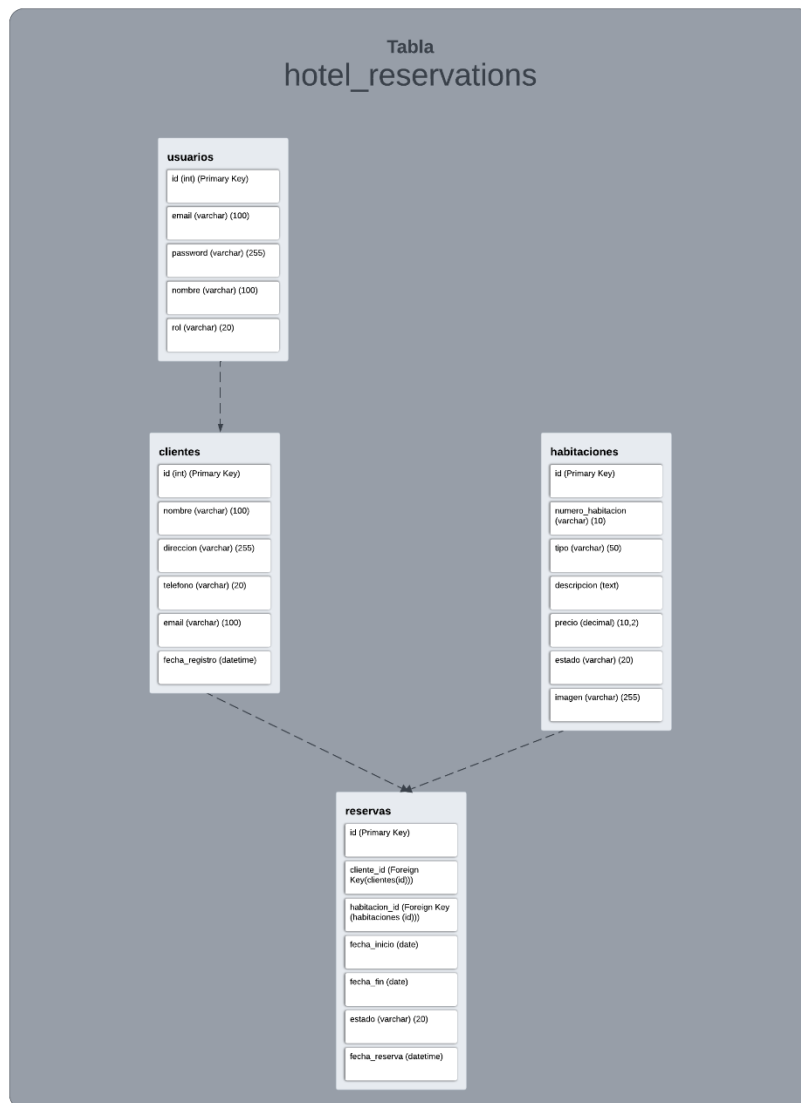
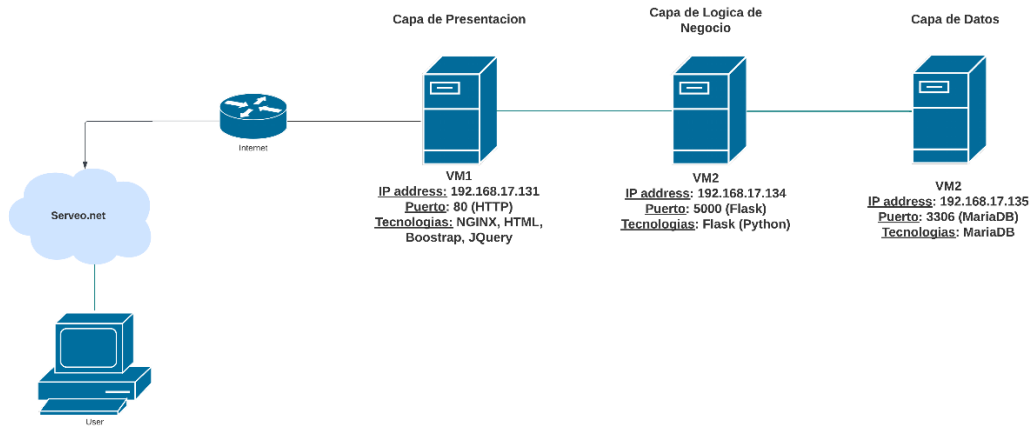
Miembros

- VM1 (Capa de Presentacion)- Yadiel Torres Quiles
 - Responsabilidad- Diseñar y activar el funcionamiento de la pagina de Web utilizando varias tecnologías a través de Serveo.
- VM2(Capa de Lógica)- Yadiel Torres Quiles
 - Responsabilidad- Diseñar y trabajar en el funcionamiento de los scripts y la lógica que se utiliza para que la pagina de web interactúe con la Base de Datos.
- VM3(Capa de Datos) - Yadiel Torres Quiles
 - Responsabilidad- Organizar y agregar los datos necesarios a la base de datos con las propiedades correctas para su almacenamiento.

Backlog del Producto

	Funcionalidad	Prioridad	Estimacion
	Registrar Usuarios Nuevos	Alto	6 Puntos
	inicio de session de Usuarios	Alto	6 Puntos
	Realizacion de reservas	Alto	6 Puntos
	buscar y demostrar datos de habitaciones	Alto	6 Puntos
	demostrar las opciones disponibles a la cuenta del Usuario	Medio	4 Puntos
	Demostracion de Reservas con detalles en la cuenta del usuario	Medio	4 Puntos
	Sistema para modificar detalles en relacion de las reservas	Bajo	2 Puntos

Diseños de Proyecto



Codigo Documentado

Codigo de create_database.sql que es utilizado para crear la base de datos

```
1  -- Crear la base de datos
2  CREATE DATABASE IF NOT EXISTS hotel_reservations;
3  USE hotel_reservations;
4  -- Tabla clientes
5  CREATE TABLE clientes (
6  id INT AUTO_INCREMENT PRIMARY KEY,
7  nombre VARCHAR(100) NOT NULL,
8  direccion VARCHAR(255),
9  telefono VARCHAR(20),
10 email VARCHAR(100) NOT NULL UNIQUE,
11 fecha_registro DATETIME DEFAULT CURRENT_TIMESTAMP
12 );
13 -- Tabla habitaciones
14 CREATE TABLE habitaciones (
15 id INT AUTO_INCREMENT PRIMARY KEY,
16 numero_habitacion VARCHAR(10) NOT NULL UNIQUE,
17 tipo VARCHAR(50),
18 descripcion TEXT,
19 precio DECIMAL(10,2) NOT NULL,
20 estado VARCHAR(20) DEFAULT 'Disponible',
21 imagen VARCHAR(255)
22 );
23 -- Tabla reservas
24 CREATE TABLE reservas (
25 id INT AUTO_INCREMENT PRIMARY KEY,
26 cliente_id INT NOT NULL,
27 habitacion_id INT NOT NULL,
28 fecha_inicio DATE NOT NULL,
29 fecha_fin DATE NOT NULL,
30 estado VARCHAR(20) DEFAULT 'Activa',
31 fecha_reserva DATETIME DEFAULT CURRENT_TIMESTAMP,
32 FOREIGN KEY (cliente_id) REFERENCES clientes(id),
33 FOREIGN KEY (habitacion_id) REFERENCES habitaciones(id)
34 );
35 -- Insertar datos de prueba
36 INSERT INTO clientes (nombre, direccion, telefono, email)
37 VALUES
38 ('Juan Pérez', 'Calle Falsa 123', '555-1234',
39 'juan.perez@example.com'),
40 ('María García', 'Avenida Siempre Viva 456', '555-5678',
```



```

73
5  -- Insertar datos de prueba
6  INSERT INTO clientes (nombre, direccion, telefono, email)
7  VALUES
8  ('Juan Pérez', 'Calle Falsa 123', '555-1234',
9  'juan.perez@example.com'),
10 ('María García', 'Avenida Siempre Viva 456', '555-5678',
11 'maria.garcia@example.com'),
12 ('Carlos López', 'Boulevard de los Sueños 789', '555-9012',
13 'carlos.lopez@example.com');
14 INSERT INTO habitaciones (numero_habitacion, tipo, descripcion, precio, estado, imagen)
15 VALUES
16 ('101', 'Sencilla', 'Habitación sencilla con cama individual',
17 50.00, 'Disponible', '/imagenes_habitaciones/101.jpg'),
18 ('102', 'Doble', 'Habitación doble con dos camas individuales',
19 75.00, 'Disponible', '/imagenes_habitaciones/102.jpg'),
20 ('201', 'Matrimonial', 'Habitación matrimonial con cama king
21 size', 100.00, 'Disponible', '/imagenes_habitaciones/201.jpg'),
22 ('202', 'Suite', 'Suite de lujo con vista al mar', 150.00,
23 'Disponible', '/imagenes_habitaciones/202.jpg'),
24 ('301', 'Presidencial', 'Habitación presidencial con todas las
25 comodidades', 250.00, 'Disponible',
26 '/imagenes_habitaciones/301.jpg');
27 INSERT INTO reservas (cliente_id, habitacion_id, fecha_inicio, fecha_fin, estado)
28 VALUES
29 (1, 101, '2023-01-15', '2023-01-20', 'Activa'),
30 (2, 201, '2023-02-01', '2023-02-05', 'Activa'),
31 (3, 202, '2023-03-10', '2023-03-15', 'Activa');

```

Código de create_usuarios.sql que es utilizado para añadir la información de la table de usuarios a la base de datos

```

Code Blame 7 lines (7 loc) · 176 Bytes
1  CREATE TABLE usuarios (
2  id INT AUTO_INCREMENT PRIMARY KEY,
3  email VARCHAR(100) NOT NULL UNIQUE,
4  password VARCHAR(255) NOT NULL,
5  nombre VARCHAR(100),
6  rol VARCHAR(20) NOT NULL
7  );|

```

Este es el código de HTML utilizado para crear la pagina utilizada para registrar usuarios donde acepta un email, nombre y password para crear una cuenta en la base de datos.

```
Code Blame 71 lines (67 loc) · 2.07 KB Code 55% faster with GitHub Copilot

1 <!-- register.html -->
2 <!DOCTYPE html>
3 <html lang="es">
4 <head>
5   <meta charset="UTF-8">
6   <title>Hotel XYZ - Inicio</title>
7   <link rel="stylesheet" href="assets/css/bootstrap.min.css">
8   <link rel="stylesheet" href="assets/css/styles.css">
9 </head>
10
11 <body>
12   <!-- Header -->
13   <!--#include file="header.html" -->
14   <!-- Contenido principal -->
15   <div class="container">
16     <h1 class="mt-4">Registro de Usuario</h1>
17     <form id="form-registro">
18       <div class="form-group">
19         <label for="nombre">Nombre Completo</label>
20         <input type="text" class="form-control" id="nombre" required>
21       </div>
22       <div class="form-group">
23         <label for="email">Correo Electrónico</label>
24         <input type="email" class="form-control" id="email" required>
25       </div>
26       <div class="form-group">
27         <label for="password">Contraseña</label>
28         <input type="password" class="form-control" id="password" required>
29       </div>
30       <button type="submit" class="btn btn-primary">Registrarse</button>
31     </form>
32   </div>
33   <!-- Footer -->
34   <footer class="py-4 bg-dark text-white-50">
35     <div class="container text-center">
36       <small>© 2023 Hotel XYZ. Todos los derechos reservados.</small>
37     </div>
38   </footer>
39   <!--#include file="footer.html" -->
40   <!-- Enlaces a JavaScript -->
41   <script src="assets/js/jquery.min.js"></script>
42   <script src="assets/js/bootstrap.min.js"></script>
43
44   <script>
45     $('#form-registro').submit(function(event) {event.preventDefault();
46       var nombre = $('#nombre').val();
47       var email = $('#email').val();
48       var password = $('#password').val();
49
50       $.ajax({
51         url: '/api/register',
52         method: 'POST',
53         contentType: 'application/json',
54         data: JSON.stringify({
55           'nombre': nombre,
56           'email': email,
57           'password': password
58         }),
59         success: function(response) {
60           alert('Registro exitoso. Ahora puedes iniciar sesión.');
```

Capturas de Pantalla

Imagen que demuestra la comunicación entre la VM2 (Capa de Lógica) y VM3 (Capa de Datos)

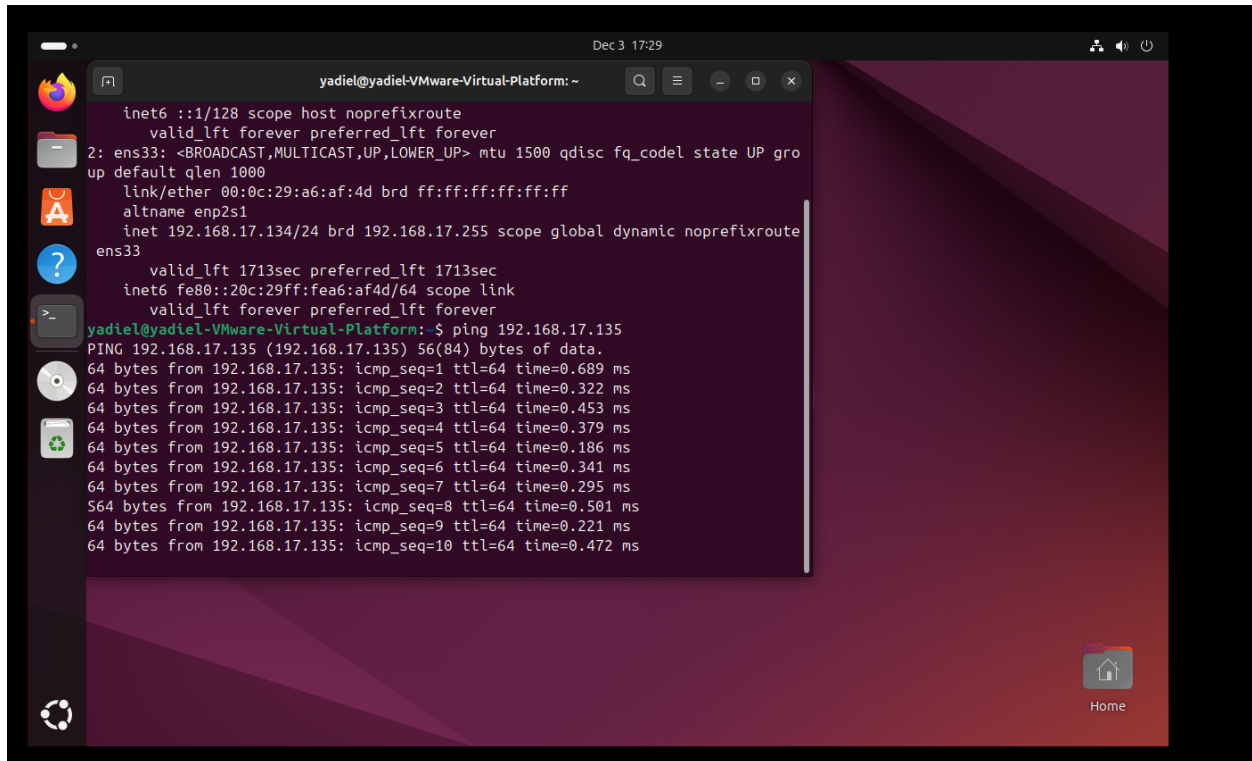
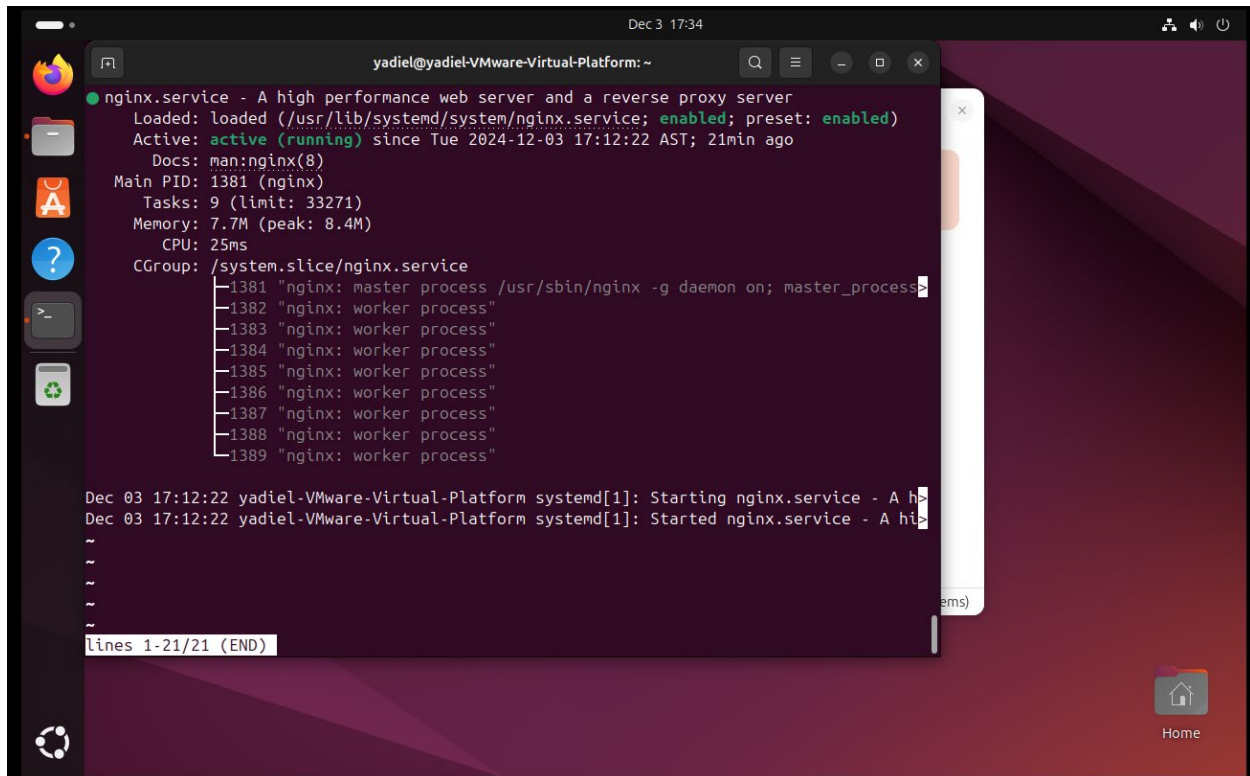


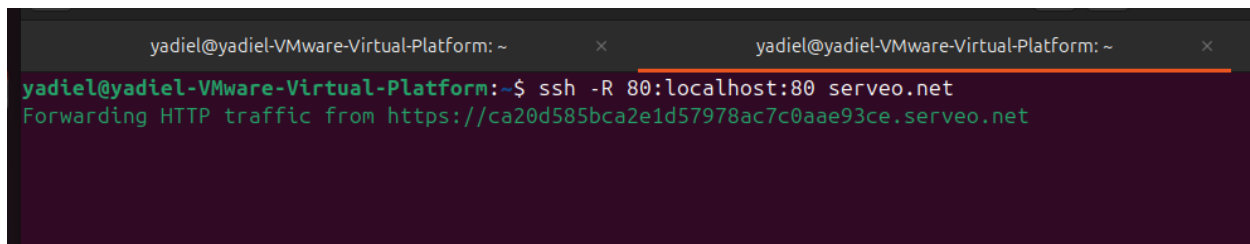
Imagen que demuestra el servidor nginx siendo activo



A terminal window on a Linux desktop environment. The terminal title is 'yadiel@yadiel-VMware-Virtual-Platform: ~'. The output shows the status of the 'nginx.service' as 'active (running)' since December 3, 2024, at 17:12:22 AST. It lists details such as 'Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)', 'Main PID: 1381 (nginx)', 'Tasks: 9 (limit: 33271)', 'Memory: 7.7M (peak: 8.4M)', and 'CPU: 25ms'. A tree view of the CGroup '/system.slice/nginx.service' shows the master process at PID 1381 and eight worker processes at PIDs 1382 through 1389. At the bottom, system logs confirm the start of the service.

```
yadiel@yadiel-VMware-Virtual-Platform: ~  
● nginx.service - A high performance web server and a reverse proxy server  
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)  
   Active: active (running) since Tue 2024-12-03 17:12:22 AST; 21min ago  
     Docs: man:nginx(8)  
  Main PID: 1381 (nginx)  
    Tasks: 9 (limit: 33271)  
  Memory: 7.7M (peak: 8.4M)  
     CPU: 25ms  
   CGroup: /system.slice/nginx.service  
           └─1381 "nginx: master process /usr/sbin/nginx -g daemon on; master_process>  
             └─1382 "nginx: worker process"  
               └─1383 "nginx: worker process"  
                 └─1384 "nginx: worker process"  
                   └─1385 "nginx: worker process"  
                     └─1386 "nginx: worker process"  
                       └─1387 "nginx: worker process"  
                         └─1388 "nginx: worker process"  
                           └─1389 "nginx: worker process"  
  
Dec 03 17:12:22 yadiel-VMware-Virtual-Platform systemd[1]: Starting nginx.service - A hi>  
Dec 03 17:12:22 yadiel-VMware-Virtual-Platform systemd[1]: Started nginx.service - A hi>  
~  
~  
~  
~  
lines 1-21/21 (END)
```

Imagen que demuestra el tráfico corriendo a través del URL proveído de serveo.net



A terminal window showing an SSH command being executed. The terminal title is 'yadiel@yadiel-VMware-Virtual-Platform: ~'. The command is 'ssh -R 80:localhost:80 serveo.net'. The output shows 'Forwarding HTTP traffic from https://ca20d585bca2e1d57978ac7c0aae93ce.serveo.net'.

```
yadiel@yadiel-VMware-Virtual-Platform: ~  
yadiel@yadiel-VMware-Virtual-Platform:~$ ssh -R 80:localhost:80 serveo.net  
Forwarding HTTP traffic from https://ca20d585bca2e1d57978ac7c0aae93ce.serveo.net
```

Imagen de las tablas entradas en la base de datos

```
Dec 3 20:17
yadiel@yadiel-VMware-Virtual-Platform: ~

-> );
Query OK, 0 rows affected (0.004 sec)

MariaDB [hotel_reservations]> SHOW TABLES
-> DESCRIBE clientes;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'DESCRIBE clientes' at line 2
MariaDB [hotel_reservations]> SHOW TABLES;
+-----+
| Tables_in_hotel_reservations |
+-----+
| clientes                      |
| habitaciones                 |
| reservas                     |
+-----+
3 rows in set (0.000 sec)

MariaDB [hotel_reservations]> DESCRIBE clientes
-> ;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(11) | NO | PRI | NULL | auto_increment |
| nombre | varchar(100) | NO | | NULL |
| direccion | varchar(255) | YES | | NULL |
| telefono | varchar(20) | YES | | NULL |
| email | varchar(100) | NO | UNI | NULL |
| fecha_registro | datetime | YES | | current_timestamp() |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.001 sec)

MariaDB [hotel_reservations]> S
```

```
Dec 3 20:18
yadiel@yadiel-VMware-Virtual-Platform: ~

MariaDB [hotel_reservations]> DESCRIBE habitaciones
-> ;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(11) | NO | PRI | NULL | auto_increment |
| numero_habitacion | varchar(10) | NO | UNI | NULL |
| tipo | varchar(50) | YES | | NULL |
| descripcion | text | YES | | NULL |
| precio | decimal(10,2) | NO | | NULL |
| estado | varchar(20) | YES | | Disponible |
| imagen | varchar(255) | YES | | NULL |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

MariaDB [hotel_reservations]> DESCRIBE reservas
-> ;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(11) | NO | PRI | NULL | auto_increment |
| cliente_id | int(11) | NO | MUL | NULL |
| habitacion_id | int(11) | NO | MUL | NULL |
| fecha_inicio | date | NO | | NULL |
| fecha_fin | date | NO | | NULL |
| estado | varchar(20) | YES | | Activa |
| fecha_reserva | datetime | YES | | current_timestamp() |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

MariaDB [hotel_reservations]>
```

```
MariaDB [hotel_reservations]> describe usuarios;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
email	varchar(100)	NO	UNI	NULL	
password	varchar(255)	NO		NULL	
nombre	varchar(100)	YES		NULL	
rol	varchar(20)	NO		NULL	

```
5 rows in set (0.001 sec)
```

```
MariaDB [hotel_reservations]> INSERT INTO clientes (nombre, direccion, telefono, email)
-> VALUES
-> ('Juan Pérez', 'Calle Falsa 123', '555-1234',
-> 'juan.perez@example.com'),
-> ('María García', 'Avenida Siempre Viva 456', '555-5678',
-> 'maria.garcia@example.com'),
-> ('Carlos López', 'Boulevard de los Sueños 789', '555-9012',
-> 'carlos.lopez@example.com');
```

```
Query OK, 3 rows affected (0.001 sec)
```

```
Records: 3 Duplicates: 0 Warnings: 0
```



```
MariaDB [hotel_reservations]>
```

```
MariaDB [hotel_reservations]> INSERT INTO reservas (cliente_id, habitacion_id, fecha_inicio, fecha_fin, estado) VALUES (
1, 101, '2023-01-15', '2023-01-20', 'Activa'), (2, 201, '2023-02-01', '2023-02-05', 'Activa'), (3, 202, '2023-03-10', '2
023-03-15', 'Activa');
```

```
Query OK, 3 rows affected (0.001 sec)
```

```
Records: 3 Duplicates: 0 Warnings: 0
```

```
MariaDB [hotel_reservations]> SHOW GRANTS FOR 'hotel_user'@'%';
```

Grants for hotel_user@%
GRANT USAGE ON *.* TO 'hotel_user'@'%' IDENTIFIED BY PASSWORD '*F31445443BB93ED07F5FAB7744A3FCE47021238F'
GRANT ALL PRIVILEGES ON 'hotel_reservations'.* TO 'hotel_user'@'%'

```
2 rows in set (0.000 sec)
```

```
MariaDB [hotel_reservations]>
```

Retrospectiva

1. ¿Qué funcionó bien en el proyecto?

Se pudieron hacer las maquinas virtuales y establecer una connecion entre ellas para que puedan comunicarse y compartir los datos requeridos para su funcionalidad .

2. ¿Qué desafíos enfrentamos y cómo los superamos?

Tuve el desafio de que no pude estar en un grupo con otros estudiantes a tiempo y empezando un trabajo nuevo, entonces tuve que optimizar mi tiempo en la mejor manera posible para poder hacer el trabajo solo.

3. ¿Qué aprendimos durante el desarrollo?

Se aprendio sobre como se comunican las maquinas con cada una, la distribucion de funcionalidad para encapsular el trabajo y los tipos de tecnologias y trabajos que necesitarian ser utilizados para tomar acargo una tarea como esta.

4. ¿Qué haríamos diferente en futuros proyectos?

Estaria mas determinado y insistente en buscando un grupo de personas el mismo dia de anuncio para prevenir los problemas que ocurrieron con esta tarea.