

# **MANUAL TECNICO**

**SOFTWARE PARA EL MANEJO DE USUARIOS Y LA REALIZACION DE RESERVAS  
PERSONALIZADAS PARA EL CONJUNTO RESIDENCIAL RECODO DE CEDRO SUBA**

**JOSEPH NICOLAS VARON VARGAS**

**JUAN SEBASTIAN CASTRO MARTINEZ**

**FICHA: 2771440V2**

**SERVICIO NACIONAL DE APRENDIZAJE SENA**

**CENTRO DE ELECTRICIDAD, ELECTRONICA Y TELECOMUNICACIONES CEET**

**TECNICO EN PROGRAMACION DE SOFTWARE**

**BOGOTA D.C., SEPTIEMBRE 2012**

## INTRODUCCION

Este manual describe una solución tecnológica para mejorar el proceso de agendamiento en el Conjunto Residencial Recodo de Cedro Suba. Actualmente, se utilizan métodos anticuados como hojas sueltas y libros de registro, lo que causa problemas de organización y eficiencia.

El nuevo sistema propuesto es una aplicación digital que centralizará toda la información de agendamientos. Sus principales características incluyen:

- Una base de datos unificada
- Una interfaz fácil de usar
- Acceso para múltiples usuarios desde diferentes dispositivos

Con esta herramienta, los administradores podrán crear y modificar citas, registrar asistencias y generar informes automáticamente. Esto ayudará a reducir errores, ahorrar tiempo y facilitar el análisis de datos.

La implementación se realizará gradualmente, empezando por instalar el software y trasladar la información existente. Luego, se capacitará al personal y se hará una prueba antes de usarlo definitivamente.

## CONTENIDO

INTRODUCCION .....	2
1. REQUERIMIENTOS TÉCNICOS.....	5
2. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO .....	6
2.1.1 PHP.....	6
2.2.2 DISEÑO WEB.....	6
2.2.3 HTML.....	6
2.2.4 CSS.....	6
2.2.5 JAVASCRIPT.....	6
2.2.6 MYSQL .....	7
2.2.7 XAMPP.....	7
2.2.8 AZURE JUNTO A MYSQL WORKBENCH.....	8
3. CASOS DE USO.....	9
4. DIAGRAMA DE DESPLIEGUE .....	21
5. DIAGRAMA DE CLASES .....	22
6. MODELO RELACIONAL .....	23
7. SENTENCIAS DDL .....	24
8. SENTENCIAS DML .....	25
9. DICCIONARIO DE DATOS .....	26
10. EJECUCION DE SENTENCIAS EN WORKBENCH .....	28
10.1 EJECUCION DE SENTENCIAS EN WORKBENCH .....	28
10.2 EJECUCION DE SENTENCIAS EN WORKBENCH .....	29

## TABLAS CU

1. TABLA 1 INICIAR SESION .....	10
2. TABLA 2 CERRAR SESION .....	10
3. TABLA 3 RECUPERAR CONTRASEÑA .....	10
4. TABLA 4 REGISTRAR ROL .....	11
5. TABLA 5 CONSULTAR ROL .....	11
6. TABLA 6 CONSULTAR ROL POR ID .....	11
7. TABLA 7 ELIMINAR ROL .....	12
8. TABLA 8 ACTUALIZAR ROL .....	12
9. TABLA 9 REGISTRAR CASA .....	13
10. TABLA 10 CONSULTAR CASA .....	13
11. TABLA 11 CONSULTAR CASA POR ID .....	13
12. TABLA 12 ELIMINAR CASA .....	14
13. TABLA 13 ACTUALIZAR CASA .....	14
14. TABLA 14 REGISTRAR USUARIO .....	15
15. TABLA 15 CONSULTAR USUARIO .....	15
16. TABLA 16 CONSULTAR USUARIO POR ID .....	15
17. TABLA 17 ELIMINAR USUARIO .....	16
18. TABLA 18 ACTUALIZAR USUARIO .....	16
19. TABLA 19 REGISTRAR LUGAR .....	17
20. TABLA 20 CONSULTAR LUGAR .....	17
21. TABLA 21 ACTUALIZAR LUGAR .....	17
22. TABLA 22 ELIMINAR LUGAR .....	18
23. TABLA 23 REGISTRAR RESERVA .....	18
24. TABLA 24 CONSULTAR RESERVA .....	19
25. TABLA 25 APROBAR RESERVA .....	19
26. TABLA 26 RECHAZAR RESERVA .....	20
27. TABLA 27 ELIMINAR RESERVA .....	20

# 1. REQUERIMIENTOS TÉCNICOS.

## - REQUERIMIENTOS TÉCNICOS

### REQUERIMIENTOS MÍNIMOS DE HARDWARE

- ✓ Procesador: Intel Pentium 4 o equivalente en Amd
- ✓ Memoria RAM: Mínimo: 1 Gigabyte (GB)
- ✓ Disco Duro: 500 GB de espacio libre

### REQUERIMIENTOS MÍNIMOS DE SOFTWARE

- ✓ Sistema Operativo: Windows 7 o posterior
- ✓ Navegador web: Google Chrome (versión 50 o posterior), Mozilla Firefox (versión 45 o posterior), o Microsoft Edge

### Notas adicionales:

- Se recomienda una conexión a Internet de al menos 1 Mbps para un rendimiento óptimo.
- Aunque Gmail puede funcionar con especificaciones inferiores, estos requisitos aseguran una experiencia de usuario fluida.
- Para sistemas operativos más antiguos como los mencionados (Windows NT/98/Me/2000/2003/XP/Vista), es posible que algunas funciones de Gmail no estén disponibles o el rendimiento sea inferior.

## 2. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO

### 2.1 PHP

PHP es un lenguaje de programación diseñado para crear páginas web dinámicas. Se integra fácilmente con HTML, permitiendo a los desarrolladores combinar código PHP dentro del marcado HTML.

Las ejecuciones de PHP se realizan en el servidor, y el cliente (navegador web) recibe solo los resultados de estas ejecuciones. Cuando un cliente hace una petición, el servidor procesa el código PHP y genera contenido dinámico que se envía de vuelta al cliente.

Una de las principales ventajas de PHP es su capacidad para conectarse con diversos sistemas de gestión de bases de datos, incluyendo MySQL, Oracle, PostgreSQL y SQL Server. Esto permite crear aplicaciones web robustas y escalables.

PHP puede ejecutarse en la mayoría de los sistemas operativos modernos y es compatible con los servidores web más populares. Su versatilidad lo hace ideal para desarrollar desde sitios web simples hasta complejas aplicaciones empresariales, sistemas de gestión de contenidos (CMS) y plataformas de comercio electrónico.

Además, PHP cuenta con un amplio ecosistema de frameworks y bibliotecas que facilitan el desarrollo rápido y eficiente de aplicaciones web modernas.

### 2.2 DISEÑO WEB

#### 2.2.1 HTML (HyperText Markup Language):

- **Propósito:** Es el lenguaje de marcado principal para estructurar el contenido en la web.
- **Características:** Utiliza etiquetas para definir elementos como encabezados, párrafos, enlaces, listas y otros componentes de la página. HTML establece la estructura básica de una página web.

#### 2.2.2 CSS (Cascading Style Sheets):

- **Propósito:** Se usa para estilizar y diseñar la apariencia de los elementos HTML.
- **Características:** Permite aplicar estilos como colores, fuentes, márgenes y layout a los elementos HTML. CSS ayuda a hacer que las páginas web sean visualmente atractivas y coherentes con el diseño deseado.

### 2.2.3 JavaScript:

- **Propósito:** Es un lenguaje de programación que se usa para agregar interactividad y funcionalidad dinámica a las páginas web.
- **Características:** Permite manipular el contenido HTML y CSS, responder a eventos del usuario (como clics y desplazamientos), y realizar operaciones más avanzadas como validación de formularios y manipulación de datos.

### 2.2.4 MYSQL WORKBENCH

MySQL Workbench es una herramienta gráfica desarrollada por Oracle que se utiliza para el diseño, desarrollo y administración de bases de datos MySQL. Proporciona una interfaz visual para gestionar y modelar bases de datos, crear y ejecutar consultas SQL, y administrar usuarios y permisos. También permite la administración de instancias de MySQL, la realización de copias de seguridad y la restauración de datos, así como la generación de informes. Su objetivo es simplificar y agilizar las tareas relacionadas con la gestión de bases de datos MySQL.

### 2.2.5 XAMPP

XAMPP es una solución integral de software libre diseñada para simplificar el desarrollo web local. Incluye los siguientes componentes clave:

**Apache:** Un servidor web de código abierto que permite gestionar y servir contenido web. Es ampliamente utilizado debido a su flexibilidad y robustez.

**MySQL/MariaDB:** Sistemas de gestión de bases de datos relacionales. MySQL es conocido por su rendimiento y fiabilidad, mientras que MariaDB es una bifurcación de MySQL que ofrece mejoras y características adicionales.

XAMPP agrupa estos componentes en un solo paquete, facilitando la instalación y configuración de un entorno de desarrollo web completo en la computadora del usuario. Esto permite a los desarrolladores crear, probar y depurar aplicaciones web en un entorno local antes de su implementación en servidores de producción. La interfaz gráfica de XAMPP también simplifica la administración de los servicios, como iniciar y detener Apache o MySQL.

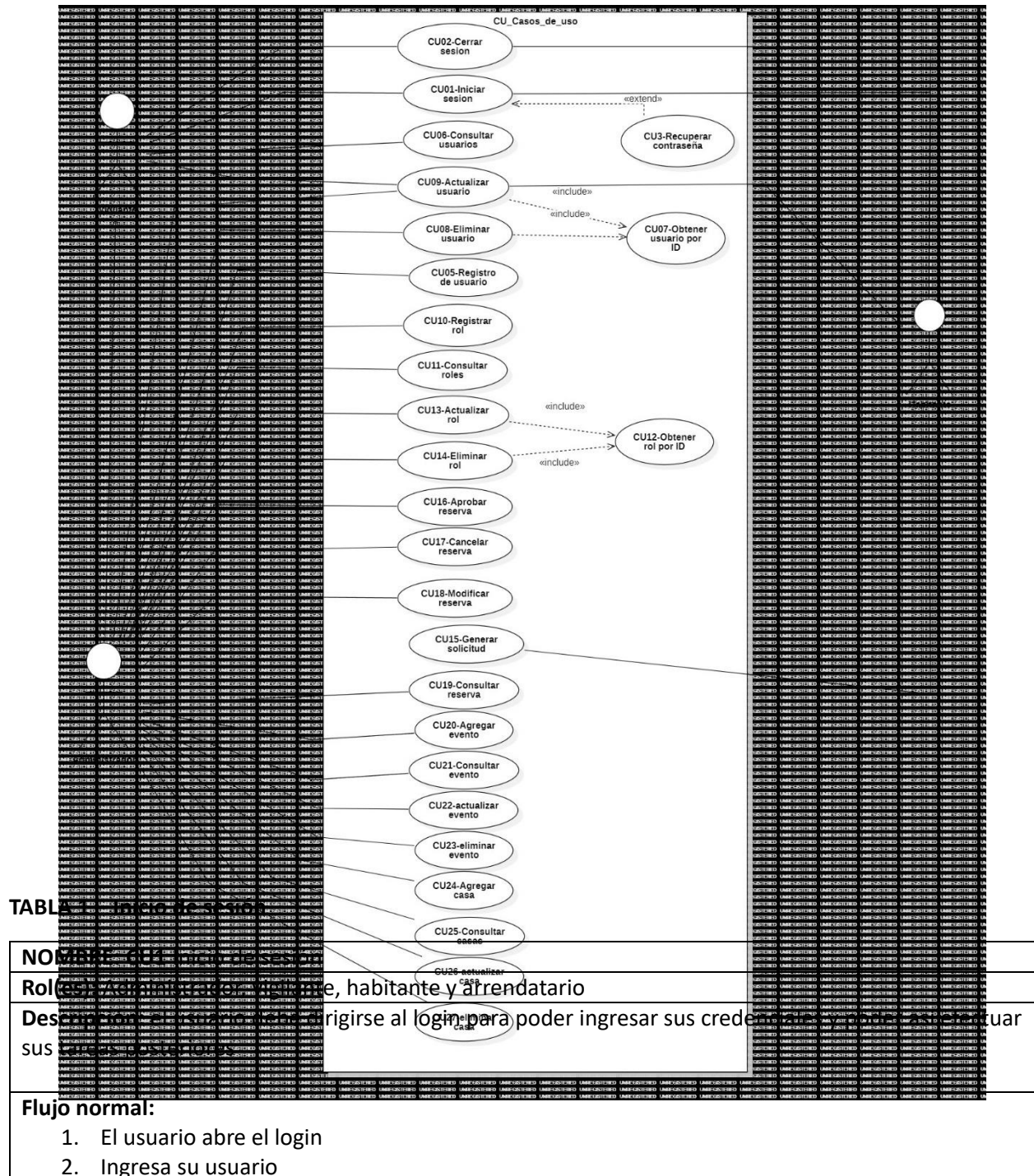
### **2.2.6 AZURE JUNTO A MYSQL WORKBENCH**

Azure es la plataforma de servicios en la nube de Microsoft, que ofrece una amplia gama de servicios en la nube, incluyendo almacenamiento, cómputo, análisis y bases de datos. Uno de los servicios disponibles en Azure es Azure Database for MySQL, que es una base de datos relacional basada en MySQL completamente administrada y escalable.

MySQL Workbench es una herramienta gráfica para el diseño, desarrollo y administración de bases de datos MySQL. Puedes usar MySQL Workbench para conectarte y gestionar bases de datos MySQL que se ejecutan en Azure. Aquí te explico cómo se conectan ambos:



En el desarrollo se realizaron 4 roles, uno de estos mas complementario que funcional puesto que cumple las mismas funciones que otro. Este rol es el de arrendatario ya que va a poder realizar las mismas funciones que un habitante del conjunto.



3. Ingresar su contraseña
4. Inicia sesión
<b>Pos condición:</b>
1. Sesión iniciada

**TABLA 2 – Cerrar sesión**

<b>NOMBRE:</b> CU2 cerrar sesión
<b>Rol(es):</b> Administrador, vigilante, habitante y arrendatario
<b>Descripción:</b> el usuario debe dirigirse al logout para poder salir de su sesión
<b>Flujo normal:</b>
1. El usuario se dirige al botón de cerrar sesión
2. Presiona cerrar sesión
<b>Pos condición:</b>
1. Sesión cerrada

**TABLA 3 – Recuperar/cambiar contraseña**

<b>NOMBRE:</b> CU3 recuperar o cambiar contraseña
<b>Rol(es):</b> Administrador, vigilante, habitante y arrendatario
<b>Descripción:</b> el usuario debe dirigirse al login para poder ubicar el botón de recuperar contraseña.
<b>Flujo normal:</b>
1. El usuario abre el login
2. Ubica el botón de recuperar contraseña
3. Ingresar su código de usuario junto a su nueva contraseña
4. Inicia sesión de nuevo
<b>Pos condición:</b>
2. Contraseña recuperada

**TABLA 4 – Registrar rol**

<b>NOMBRE:</b> CU4 registro de roles
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador debe registrar los roles esto dirigiéndose a la barra lateral con el nombre de roles

<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. El administrador entra al módulo de roles</li> <li>2. Llena los datos del rol a registrar (nombre del rol)</li> <li>3. Guarda el rol</li> </ol>
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. rol registrado</li> </ol>

**TABLA 5 – consultar roles**

<b>NOMBRE:</b> CU5 Consultar roles
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todos los roles registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de roles y se ubica el campo de consultar rol</li> <li>2. Se visualizan los roles</li> <li>3. Se encuentra el botón de actualizar o eliminar rol</li> </ol>
<b>Flujo alternativo:</b> Debe haber roles registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan los roles</li> </ol>

**TABLA 6 – consultar roles por id**

<b>NOMBRE:</b> CU6 Consultar roles por id
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todos los roles registrados por id para hacer una búsqueda específica de los mismos
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de roles y se ubica el campo de consultar roles</li> <li>2. Se visualizan los roles</li> <li>3. Se encuentra el botón de actualizar o eliminar roles</li> </ol>
<b>Flujo alternativo:</b> Debe haber roles registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan los roles</li> </ol>

**TABLA 7 – eliminar roles**

<b>NOMBRE:</b> CU7 Eliminar roles
-----------------------------------

<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá eliminar los roles a registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de roles y se ubica el campo de consultar rol</li> <li>2. Se visualizan los roles</li> <li>3. Se encuentra el botón de eliminar rol</li> <li>4. Se oprime y se elimina el rol</li> </ol>
<b>Flujo alternativo:</b> Debe haber roles registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan los roles</li> <li>2. Se elimina el rol</li> </ol>

**TABLA 8 – actualizar roles**

<b>NOMBRE:</b> CU8 Actualizar roles
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá actualizar los roles a registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de roles y se ubica el campo de consultar rol</li> <li>2. Se visualizan los roles</li> <li>3. Se encuentra el botón de actualizar rol</li> <li>4. Se oprime y se envía a la vista de actualizar el rol</li> </ol>
<b>Flujo alternativo:</b> Debe haber roles registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualiza la información del rol</li> <li>2. Se actualiza el rol</li> </ol>

**TABLA 9– Registrar casa**

<b>NOMBRE:</b> CU9 registro de casas
<b>Rol(es):</b> Administrador

<b>Descripción:</b> el administrador debe registrar las casas esto dirigiéndose a la barra lateral con el nombre de casas
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. El administrador entra al módulo de casas</li> <li>2. Llena los datos de la casa y registra la registra (nombre de la casa)</li> <li>3. Guarda la casa</li> </ol>
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. casa registrada</li> </ol>

**TABLA 10– consultar casas**

<b>NOMBRE:</b> CU10 Consultar casas
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todas las casas registradas
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de casas y se ubica el campo de consultar casa</li> <li>2. Se visualizan las casas</li> <li>3. Se encuentra el botón de actualizar o eliminar casa</li> </ol>
<b>Flujo alternativo:</b> Debe haber casas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan las casas</li> </ol>

**TABLA 11 – consultar casas por id**

<b>NOMBRE:</b> CU10 Consultar casas por id
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todas las casas registradas por id y poder consultarlas mejor
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>4. Se abre el módulo de casas y se ubica el campo de consultar casa</li> <li>5. Se visualizan las casas</li> <li>6. Se encuentra el botón de actualizar o eliminar casa</li> </ol>
<b>Flujo alternativo:</b> Debe haber casas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>2. Se visualizan las casas</li> </ol>

**TABLA 12 – eliminar casas**

<b>NOMBRE: CU12</b> Eliminar casas
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá eliminar las casas registradas
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>5. Se abre el módulo de casas y se ubica el campo de consultar casa</li> <li>6. Se visualizan las casas</li> <li>7. Se encuentra el botón de eliminar casa</li> <li>8. Se oprime y se elimina la casa</li> </ol>
<b>Flujo alternativo:</b> Debe haber casas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>3. Se visualizan las casas</li> <li>4. Se elimina las casas</li> </ol>

**TABLA 13 – actualizar casas**

<b>NOMBRE: CU13</b> Actualizar casas
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá actualizar las casas registradas
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>5. Se abre el módulo de casas y se ubica el campo de consultar casa</li> <li>6. Se visualizan las casas</li> <li>7. Se encuentra el botón de actualizar casa</li> <li>8. Se oprime y se envía a la vista de actualizar la casa</li> </ol>
<b>Flujo alternativo:</b> Debe haber casas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>3. Se visualiza la información de la casa</li> <li>4. Se actualiza la casa</li> </ol>

**TABLA 14 – registrar usuarios**

<b>NOMBRE: CU14</b> registro de usuarios
<b>Rol(es):</b> Administrador

<b>Descripción:</b> el administrador debe registrar los usuarios esto dirigiéndose a la barra lateral con el nombre de usuarios
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. El administrador entra al módulo de usuarios</li> <li>2. Llena los datos del usuario a registrar (rol, casa, nombres, apellidos, id...)</li> <li>3. Se le asigna un correo y contraseña</li> </ol>
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>2. Usuario registrado</li> </ol>

**TABLA 15 – consultar usuarios**

<b>NOMBRE:</b> CU15 Consultar usuarios
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todos los usuarios registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>3. Se abre el modulo de usuarios y se ubica el campo de consultar usuarios</li> <li>4. Se visualizan los usuarios</li> <li>5. Se encuentra el botón de actualizar o eliminar usuario</li> </ol>
<b>Flujo alternativo:</b> Debe haber usuarios registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>2. Se visualizan los usuarios</li> </ol>

**TABLA 16 – consultar usuarios por id**

<b>NOMBRE:</b> CU16 Consultar usuarios por id
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todos los usuarios registrados por id para hacer una búsqueda específica de los mismos
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>4. Se abre el módulo de usuarios y se ubica el campo de consultar usuarios</li> <li>5. Se visualizan los usuarios</li> <li>6. Se encuentra el botón de actualizar o eliminar usuario</li> </ol>
<b>Flujo alternativo:</b> Debe haber usuarios registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>2. Se visualizan los usuarios</li> </ol>

**TABLA 17 – eliminar usuarios**

<b>NOMBRE: CU17</b> Eliminar usuarios
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá eliminar los usuarios a registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de usuarios y se ubica el campo de consultar usuarios</li> <li>2. Se visualizan los usuarios</li> <li>3. Se encuentra el botón de eliminar usuario</li> <li>4. Se oprime y se elimina el usuario</li> </ol>
<b>Flujo alternativo:</b> Debe haber usuarios registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan los usuarios</li> <li>2. Se elimina el usuario</li> </ol>

**TABLA 18 – actualizar usuario**

<b>NOMBRE: CU18</b> Actualizar usuario
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá actualizar los usuarios a registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>9. Se abre el módulo de usuarios y se ubica el campo de consultar usuarios</li> <li>10. Se visualizan los usuarios</li> <li>11. Se encuentra el botón de actualizar usuario</li> <li>12. Se oprime y se envía a la vista de actualizar el usuario</li> </ol>
<b>Flujo alternativo:</b> Debe haber usuarios registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>5. Se visualiza la información del usuario</li> <li>6. Se actualiza el usuario</li> </ol>

**TABLA 19 – agregar lugar**

<b>NOMBRE: CU19</b> agregar evento/lugar
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá realizar el registro de algún lugar especial para las reservas



<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. se abre el módulo de lugares</li> <li>2. se registra en el campo de agregar lugar</li> </ol>
<b>Flujo alternativo:</b> N/A
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualiza el modulo de lugares</li> <li>2. Se agrega el lugar</li> </ol>

**TABLA 20 – consultar lugares**

<b>NOMBRE:</b> CU20 Consultar lugares
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todos los lugares registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de lugares y se ubica el campo de consultar lugar</li> <li>2. Se visualizan los lugares</li> <li>3. Se encuentra el botón de actualizar o eliminar lugar</li> </ol>
<b>Flujo alternativo:</b> Debe haber lugares registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan los lugares</li> </ol>

**TABLA 21 – actualizar lugar**

<b>NOMBRE:</b> CU21 Actualizar lugar
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá actualizar los lugares registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de lugares y se ubica el campo de consultar lugar</li> <li>2. Se visualizan los lugares</li> <li>3. Se encuentra el botón de actualizar lugar</li> <li>4. Se oprime y se envía a la vista de actualizar el lugar</li> </ol>
<b>Flujo alternativo:</b> Debe haber lugares registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualiza la información del lugar</li> <li>2. Se actualiza el lugar</li> </ol>

**TABLA 22 – eliminar lugares**

<b>NOMBRE:</b> CU22 Eliminar lugares
--------------------------------------

<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá eliminar los lugares a registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de lugares y se ubica el campo de consultar lugar</li> <li>2. Se visualizan los lugares</li> <li>3. Se encuentra el botón de eliminar lugar</li> <li>4. Se oprime y se elimina el lugar</li> </ol>
<b>Flujo alternativo:</b> Debe haber roles registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan los roles</li> <li>2. Se elimina el rol</li> </ol>

**TABLA 23 – agregar reserva**

<b>NOMBRE:</b> CU23 agregar reserva
<b>Rol(es):</b> Administrador, vigilante, habitante y arrendatario
<b>Descripción:</b> los usuarios podrán realizar sus reservas
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de las reservas</li> <li>2. Se llenan los datos predispuestos como fecha, lugar y código de usuario</li> </ol>
<b>Flujo alternativo:</b> debe haber lugares y usuarios registrados
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualiza el módulo de reservas</li> <li>2. Se agrega la reserva</li> </ol>

**TABLA 24 – consultar reservas**

<b>NOMBRE:</b> CU24 Consultar reserva
<b>Rol(es):</b> Administrador, vigilante
<b>Descripción:</b> el administrador y vigilante podrán revisar todas las reservas registradas
<b>Flujo normal:</b>

<ol style="list-style-type: none"> <li>1. Se abre el módulo de reservas y se ubica el campo de consultar reserva</li> <li>2. Se visualizan las reservas</li> <li>3. Se encuentra el botón de aprobar o rechazar la reserva</li> </ol>
<b>Flujo alternativo:</b> Debe haber reservas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualizan las reservas registradas</li> </ol>

**TABLA 25 – aprobar reserva**

<b>NOMBRE:</b> CU25 Aprobar reserva
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá aprobar o rechazar la reserva. Esto actualizando el campo de aprobado o rechazado
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de reservas y se ubica el campo de consultar reserva</li> <li>2. Se visualizan las reservas</li> <li>3. Se encuentra el botón de aprobar</li> <li>4. Se oprime y se actualiza a aprobada la reserva</li> </ol>
<b>Flujo alternativo:</b> Debe haber reservas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualiza la información de la reserva</li> <li>2. Se aprueba la reserva</li> </ol>

**TABLA 26 – rechazar reserva**

<b>NOMBRE:</b> CU26 Rechazar reserva
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá aprobar o rechazar la reserva. Esto actualizando el campo de aprobado o rechazado

<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de reservas y se ubica el campo de consultar reserva</li> <li>2. Se visualizan las reservas</li> <li>3. Se encuentra el botón de aprobar</li> <li>4. Se oprime y se actualiza a rechazada la reserva</li> </ol>
<b>Flujo alternativo:</b> Debe haber reservas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>1. Se visualiza la información de la reserva</li> <li>2. Se rechaza la reserva</li> </ol>

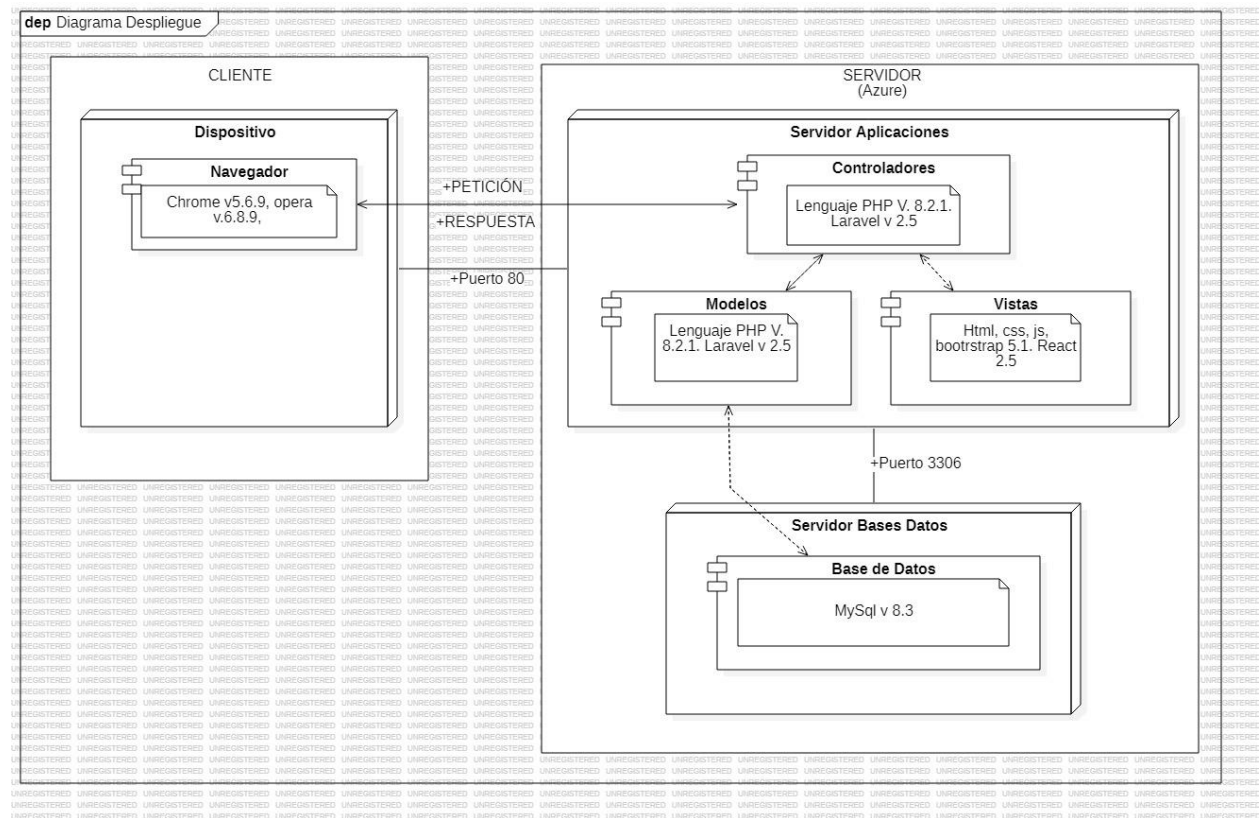
**TABLA 27 – eliminar reserva**

<b>NOMBRE:</b> CU27 Eliminar reservas
<b>Rol(es):</b> Administrador
<b>Descripción:</b> el administrador podrá eliminar las reservas a registrados
<b>Flujo normal:</b> <ol style="list-style-type: none"> <li>1. Se abre el módulo de lugares y se ubica el campo de consultar reserva</li> <li>2. Se visualizan las reservas</li> <li>3. Se encuentra el botón de eliminar reserva</li> <li>4. Se oprime y se elimina la reserva</li> </ol>
<b>Flujo alternativo:</b> Debe haber reservas registradas
<b>Pos condición:</b> <ol style="list-style-type: none"> <li>3. Se visualizan las reservas</li> <li>4. Se elimina la reserva</li> </ol>

#### 4. DIAGRAMA DE DESPLIEGUE

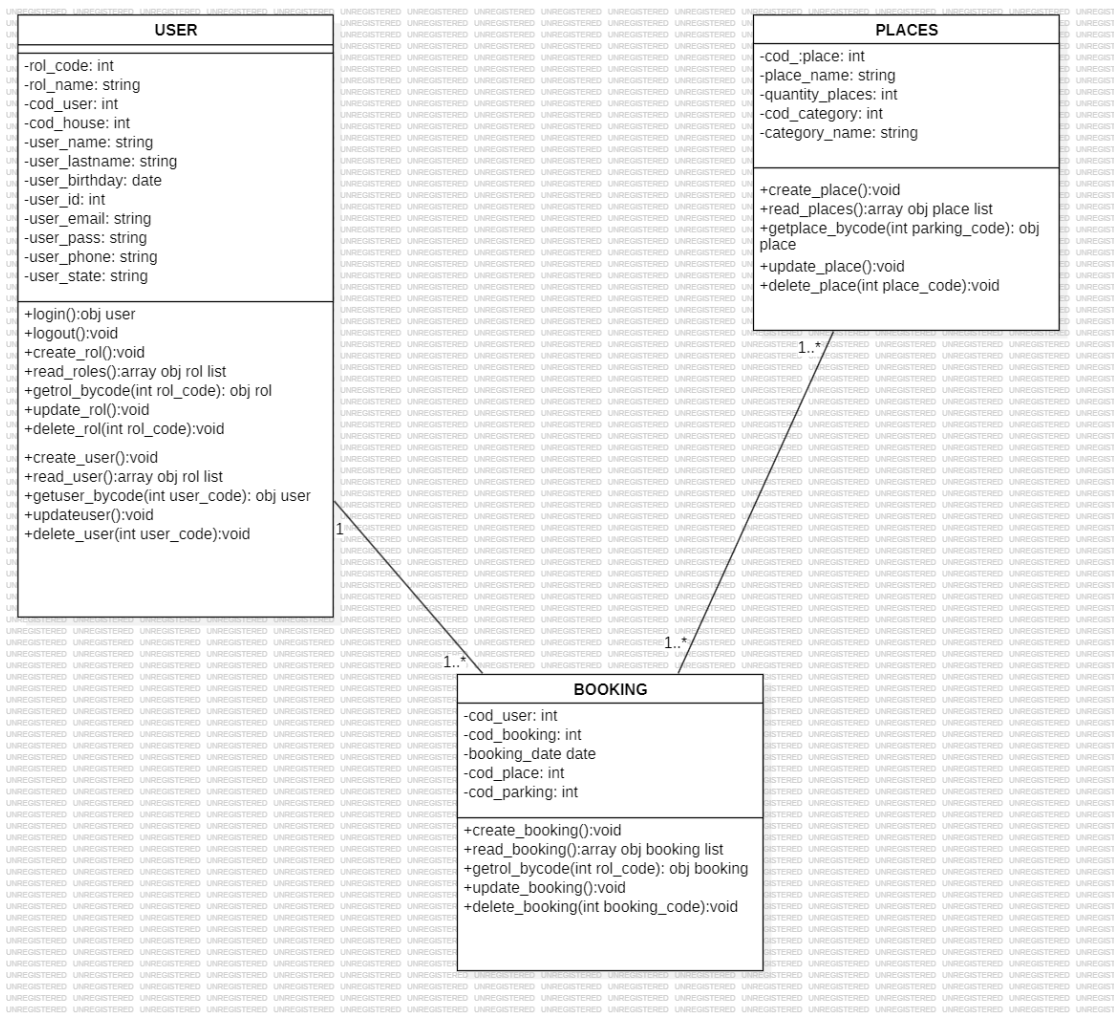
El diagrama de despliegue es una representación visual que muestra cómo los componentes de un sistema de software están distribuidos en el hardware y cómo interactúan entre sí. Permite entender la disposición física de los elementos del sistema, las conexiones entre ellos y la infraestructura necesaria

para su funcionamiento. Es útil para planificar, implementar y gestionar el sistema en su entorno real, ayudando a asegurar un despliegue eficiente y efectivo.



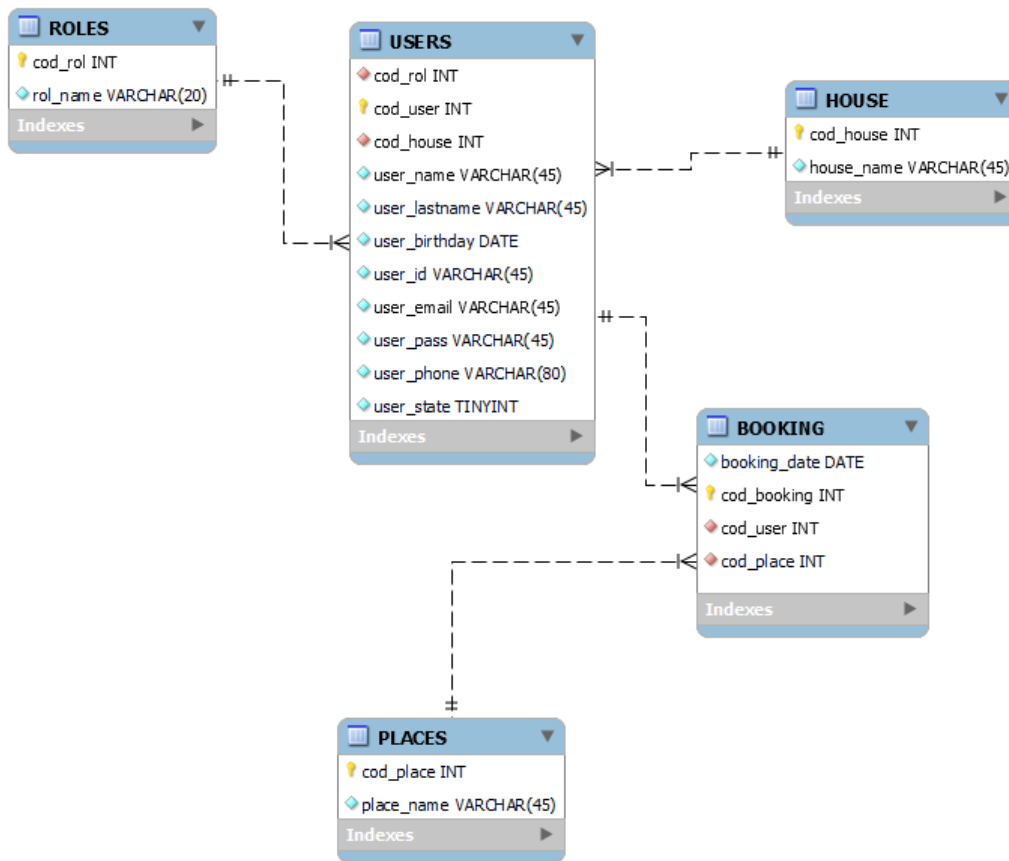
## 5. DIAGRAMA DE CLASES

El diagrama de clases es una representación visual que ilustra la estructura estática de un sistema de software, mostrando las clases, sus atributos, métodos y las relaciones entre ellas. Es esencial para entender y diseñar la arquitectura del sistema, facilitando la comunicación y la documentación del diseño de software.



## 6. MODELO RELACIONAL

el modelo relacional realizado en MySQL Workbench proporciona una visión estructurada y visual de cómo se organizan los datos en una base de datos, facilitando el diseño, la implementación y el mantenimiento del sistema de gestión de bases de datos.



## 7. SENTENCIAS DDL

Las sentencias DDL se utilizan para definir y modificar la estructura de las bases de datos y sus objetos (como tablas, índices y vistas). Estas sentencias afectan la estructura y el esquema de la base de datos, y

suelen tener efectos permanentes que no se pueden deshacer fácilmente sin realizar una operación específica de reversión. En este caso se puede usar nuestra base de datos como referencia.

## CREACION DE LAS TABLAS ROLES, USUARIOS, CASAS, LUGARES Y RESERVAS

```

17 CREATE SCHEMA IF NOT EXISTS `conjuntos_reservas` DEFAULT CHARACTER SET utf8 ;
18 USE `conjuntos_reservas` ;
19
20 -----
21 -- Table `conjuntos_reservas`.`ROLES`
22 -----
23 CREATE TABLE IF NOT EXISTS `conjuntos_reservas`.`ROLES` (
24   `cod_rol` INT NOT NULL AUTO_INCREMENT,
25   `rol_name` VARCHAR(20) NOT NULL,
26   PRIMARY KEY (`cod_rol`))
27 ENGINE = InnoDB;
28
29
30 -----
31
32 -----
33 CREATE TABLE IF NOT EXISTS `conjuntos_reservas`.`HOUSE` (
34   `cod_house` INT NOT NULL AUTO_INCREMENT,
35   `house_name` VARCHAR(45) NOT NULL,
36   PRIMARY KEY (`cod_house`))
37 ENGINE = InnoDB;
38
39
40 -----
41
42 -----
43 -- Table `conjuntos_reservas`.`USERS`
44 -----
45
46 -----
47 CREATE TABLE IF NOT EXISTS `conjuntos_reservas`.`USERS` (
48   `cod_rol` INT NOT NULL,
49   `cod_user` INT NOT NULL AUTO_INCREMENT,
50   `cod_house` INT NOT NULL,
51   `user_name` VARCHAR(45) NOT NULL,
52   `user_lastname` VARCHAR(45) NOT NULL,
53   `user_birthday` DATE NOT NULL,
54   `user_id` VARCHAR(45) NOT NULL,
55   `user_email` VARCHAR(45) NOT NULL,
56
57   -----
58   -- Table `conjuntos_reservas`.`PLACES`
59   -----
60
61   -----
62   CREATE TABLE IF NOT EXISTS `conjuntos_reservas`.`PLACES` (
63     `cod_place` INT NOT NULL AUTO_INCREMENT,
64     `place_name` VARCHAR(45) NOT NULL,
65     PRIMARY KEY (`cod_place`))
66   ENGINE = InnoDB;
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975

```

```

81 table 'conjuntos_reservas', 'BOOKING'
82
83 CREATE TABLE IF NOT EXISTS 'conjuntos_reservas', 'BOOKING' (
84     'booking_date' INT64 NOT NULL,
85     'cod_booking' INT64 NOT NULL,
86     'cod_user' INT64 NOT NULL,
87     'cod_place' INT64 NOT NULL,
88     'booking_status' ENUM('approved', 'pending', 'rejected') NOT NULL DEFAULT 'pending',
89     PRIMARY KEY ('cod_booking'),
90     INDEX 'cod_user_idx' ('cod_user' ASC) VISIBLE,
91     INDEX 'cod_place_idx' ('cod_place' ASC) VISIBLE,
92     CONSTRAINT 'cod_user'
93     FOREIGN KEY ('cod_user')
94     REFERENCES 'conjuntos_reservas', 'USERS' ('cod_user')
95     ON DELETE CASCADE
96     ON UPDATE CASCADE,
97     CONSTRAINT 'cod_place'
98     FOREIGN KEY ('cod_place')
99     REFERENCES 'conjuntos_reservas', 'PLACES' ('cod_place')
100     ON DELETE CASCADE
101     ON UPDATE CASCADE)
102 ENGINE = InnoDB;
103
104
105

```

## 8. SENTENCIAS DML

Las sentencias DML se utilizan para manipular los datos dentro de las tablas. Estas sentencias afectan los datos almacenados en las tablas y permiten realizar operaciones como insertar, actualizar, eliminar y consultar datos.



### Datos de la tabla casa, roles, usuarios y lugares

```

1      INSERT INTO `house` (`cod_house`, `house_name`) VALUES
2      (1, 'CASA UNO'),
3      (2, 'CASA DOS'),
4      (3, 'CASA TRES'),
5      (4, 'CASA CUATRO'),
6      (5, 'NO TIENE CASA');

INSERT INTO `roles` (`cod_rol`, `rol_name`) VALUES
(1, 'ADMIN'),
(2, 'VIGILANTE'),
(3, 'HABITANTE'),
(4, 'ARRENDATARIO');

INSERT INTO `places` (`cod_place`, `place_name`) VALUES
(1, 'BBQ'),
(2, 'SALON COMUNAL');

INSERT INTO `users` (`cod_rol`, `cod_user`, `cod_house`, `user_name`, `user_lastname`, `user_birthday`, `user_id`, `user_email`, `user_pass`, `user_phone`, `user_state`) VALUES
(1, 1, 5, 'maribel', 'perez', '1986-02-15', '1001010110', 'maribel@gmail.com', '8cb2237d0679ca88db6464eac60da96345513964', '31022541678', 1),
(2, 2, 5, 'Luis', 'Parra', '1977-10-06', '99884568', 'parra@gmail.com', '7c4a8d09ca3762af61e59520943dc26494f8941b', '3022548972', 1),
(3, 3, 1, 'joseph', 'varon', '2002-09-06', '1022524565', 'joseph@gmail.com', '7c4a8d09ca3762af61e59520943dc26494f8941b', '123456', 1),
(4, 4, 4, 'juan', 'lopez', '1999-09-27', '15151515', 'juan@gmail.com', '7c4a8d09ca3762af61e59520943dc26494f8941b', '3222156874', 1),
(3, 5, 2, 'luisa', 'martinez', '2001-01-01', '1111111111', 'luisa@gmail.com', '7c4a8d09ca3762af61e59520943dc26494f8941b', '123456', 1);

```

## 9. DICCIONARIO DE DATOS

Un diccionario de datos es una herramienta crucial para la gestión y diseño de bases de datos. Proporciona una descripción detallada de los datos, sus estructuras, relaciones y reglas, facilitando el

BOOKING								
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI Default Comment
booking_date	DATE		✓					
cod_booking	INT	✓	✓				✓	
id_booking	INT							

diseño, desarrollo, mantenimiento y comprensión del sistema de datos. Es esencial para asegurar una gestión eficiente y coherente de la información en cualquier sistema de base de datos.

users										
Column name	Data Type	PK	NN	UQ	BIN	UN	ZE	AI	Default	Comment
cod_rol	INT(11)		✓							
cod_user	INT(11)	✓	✓							
cod_house	INT(11)		✓							
user_name	VARCHAR(45)		✓							
user_lastname	VARCHAR(45)		✓							
user_birthday	DATE		✓							
user_id	INT(11)		✓							
user_email	VARCHAR(45)		✓							
user_pass	VARCHAR(45)		✓							
user_phone	VARCHAR(80)		✓							
user_state	TINYINT(4)		✓							

rol_name	VARCHAR(20)		✓							
----------	-------------	--	---	--	--	--	--	--	--	--

## 10. EJECUCION DE SENTENCIAS EN WORKBENCH

En esta ejecución de las sentencias tanto ddl y dml buscamos poder agilizar el proceso de la carga de base de datos. Esto para que el administrador tenga un usuario universal registrado y poder así ingresar

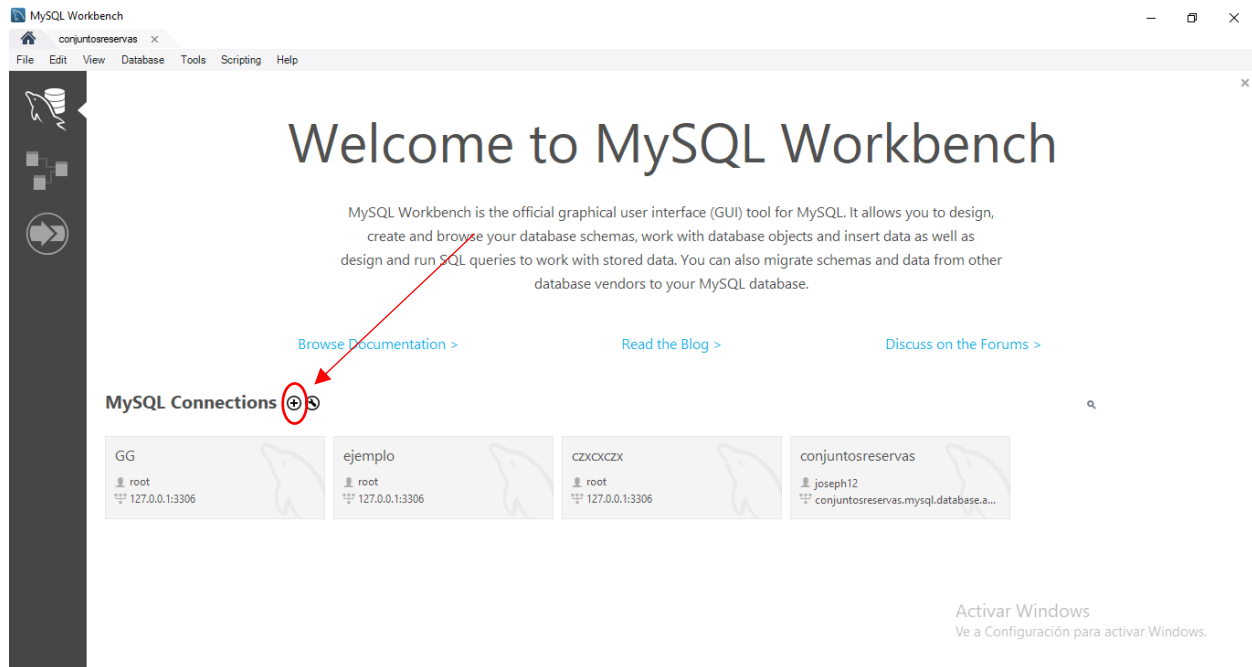
### 10.1 Instalación del repositorio de git

- Nos dirigimos a nuestro navegador de preferencia y entramos al repositorio con este url [https://github.com/joseph12n/conjuntos\\_residenciales-main](https://github.com/joseph12n/conjuntos_residenciales-main) y clonamos el proyecto usando la herramienta de git con su comando de:

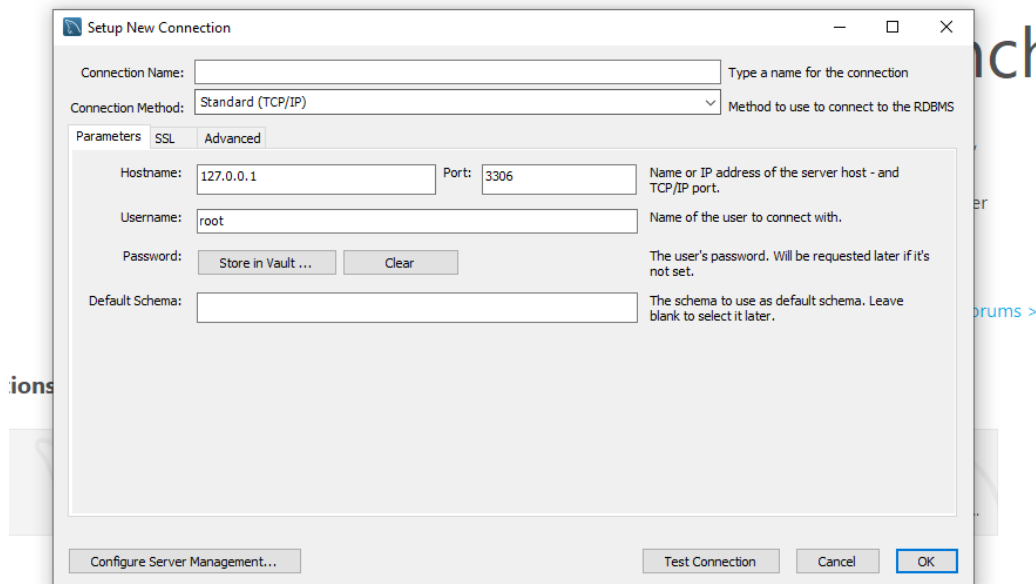
git clone https://github.com/joseph12n/conjuntos\_residenciales-main.

## 10.2 CONEXIÓN EN MYSQL WORKBENCH V8.0CE

- Iniciamos el programa e ingresamos al icono del + para poder registrar una nueva conexión.



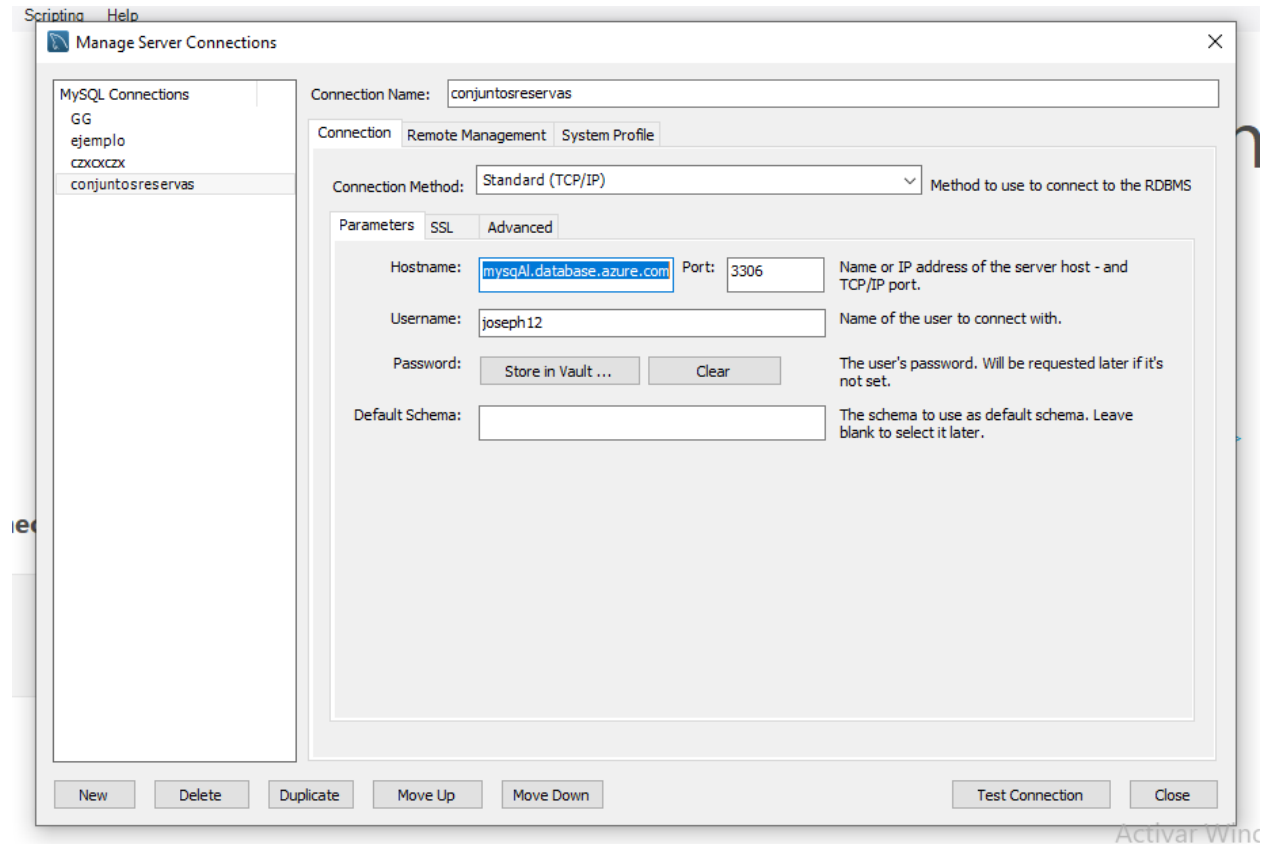
- Al darle clic se nos abrirá el modulo para realizar la conexión con. Esto ya que la base de datos esta alojada en Azure.



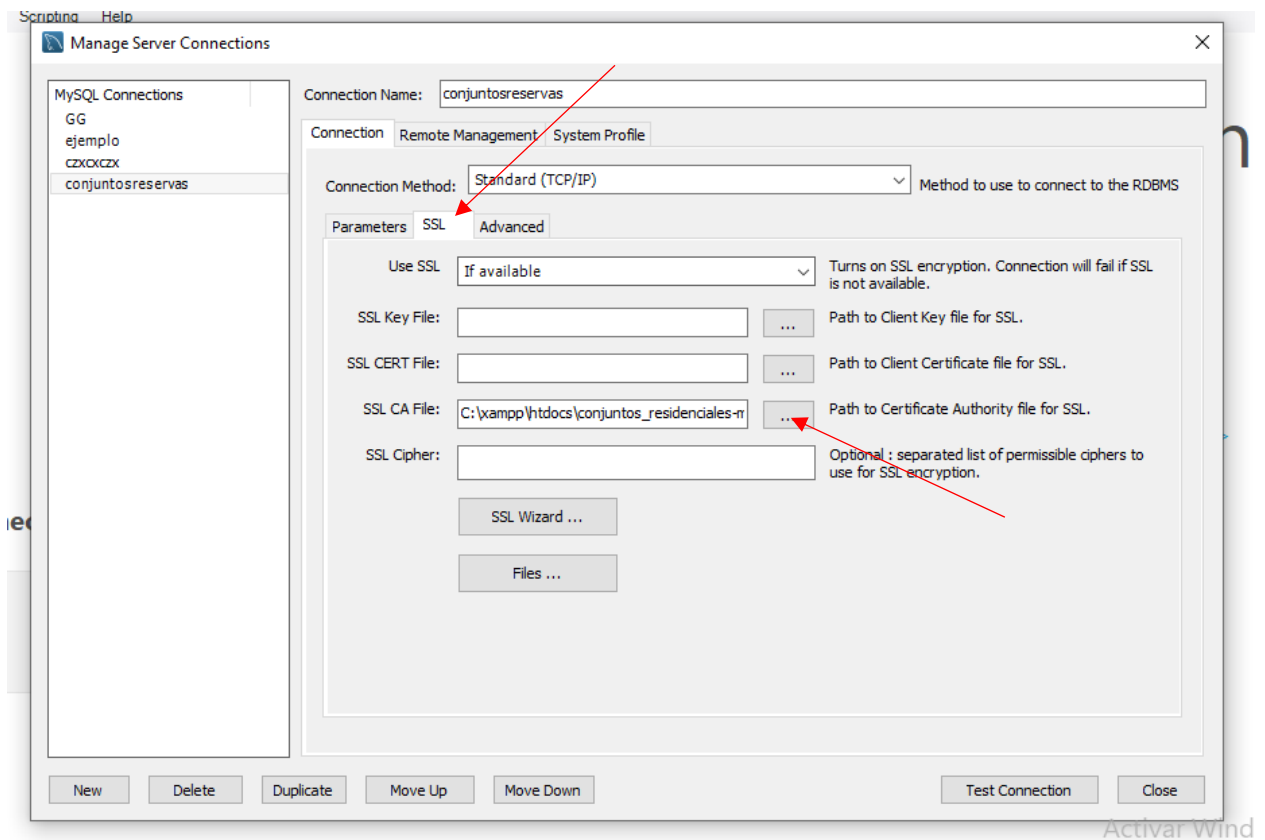
- Inicio de la conexión. Necesitaremos ingresar un nombre a la conexión a gusto del que vaya a iniciar la conexión.
- El método para usar se deja en Standart(TPC/IP). Es el que viene por defecto.
- En el hostname ingresaremos la siguiente url conjuntosreservas.mysql.database.azure.com

- El port se deja el que viene por defecto. En este caso. Port 3306
- En user name ingresamos el usuario admin. En este caso es joseph12
- En password ingresamos al botón que dice store in vault e ingresamos la contraseña que es conjuntosR12

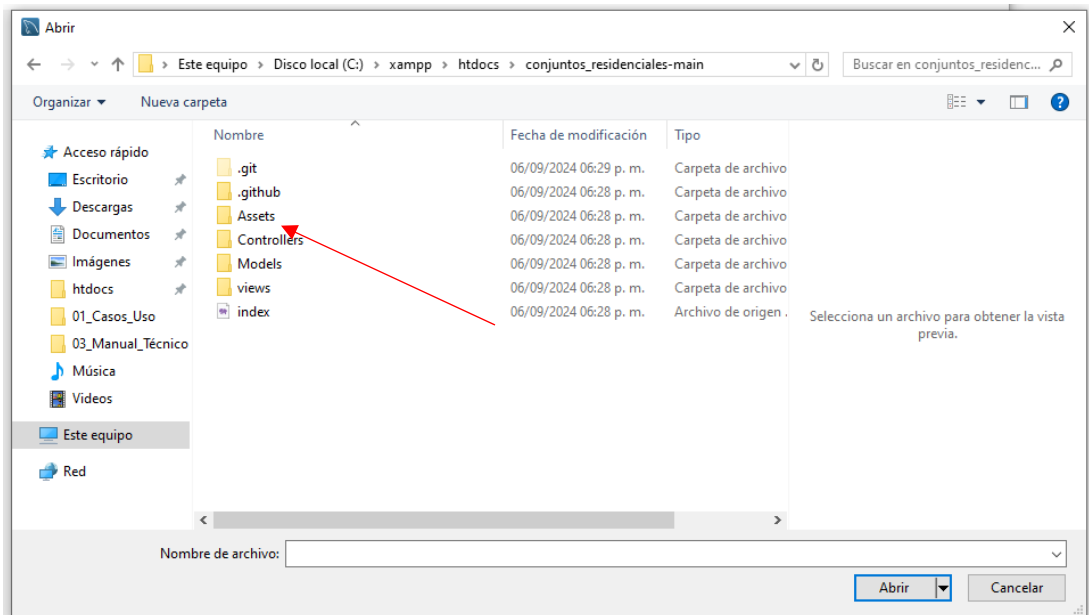
Así debería quedarnos por el momento.



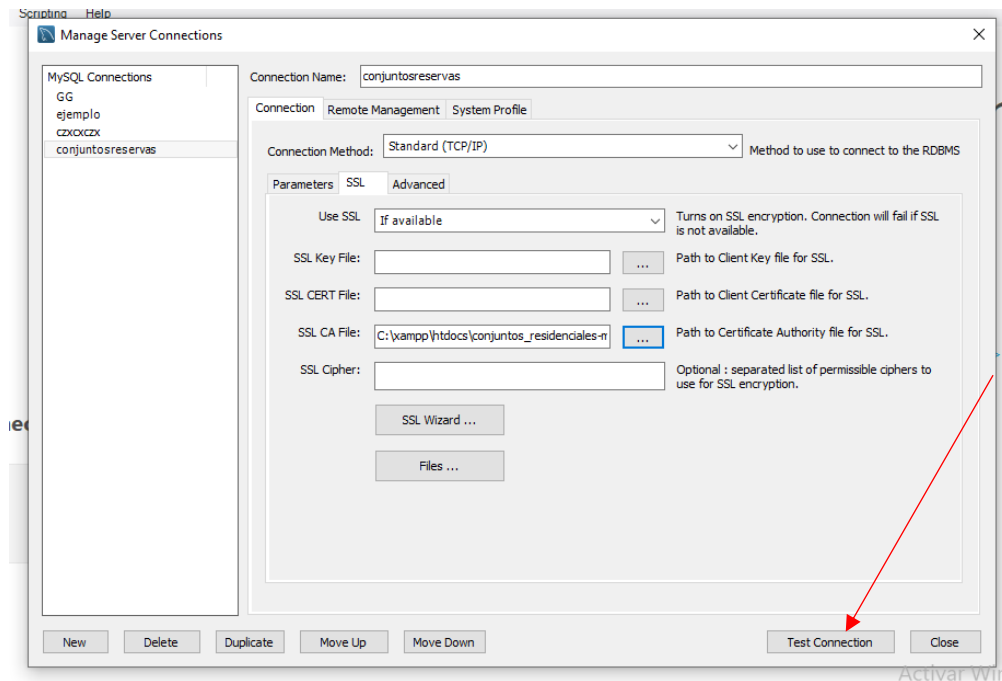
- Ahora ingresamos ahí dentro a la pestaña SSL y ubicamos el campo que dice SSL CA File.



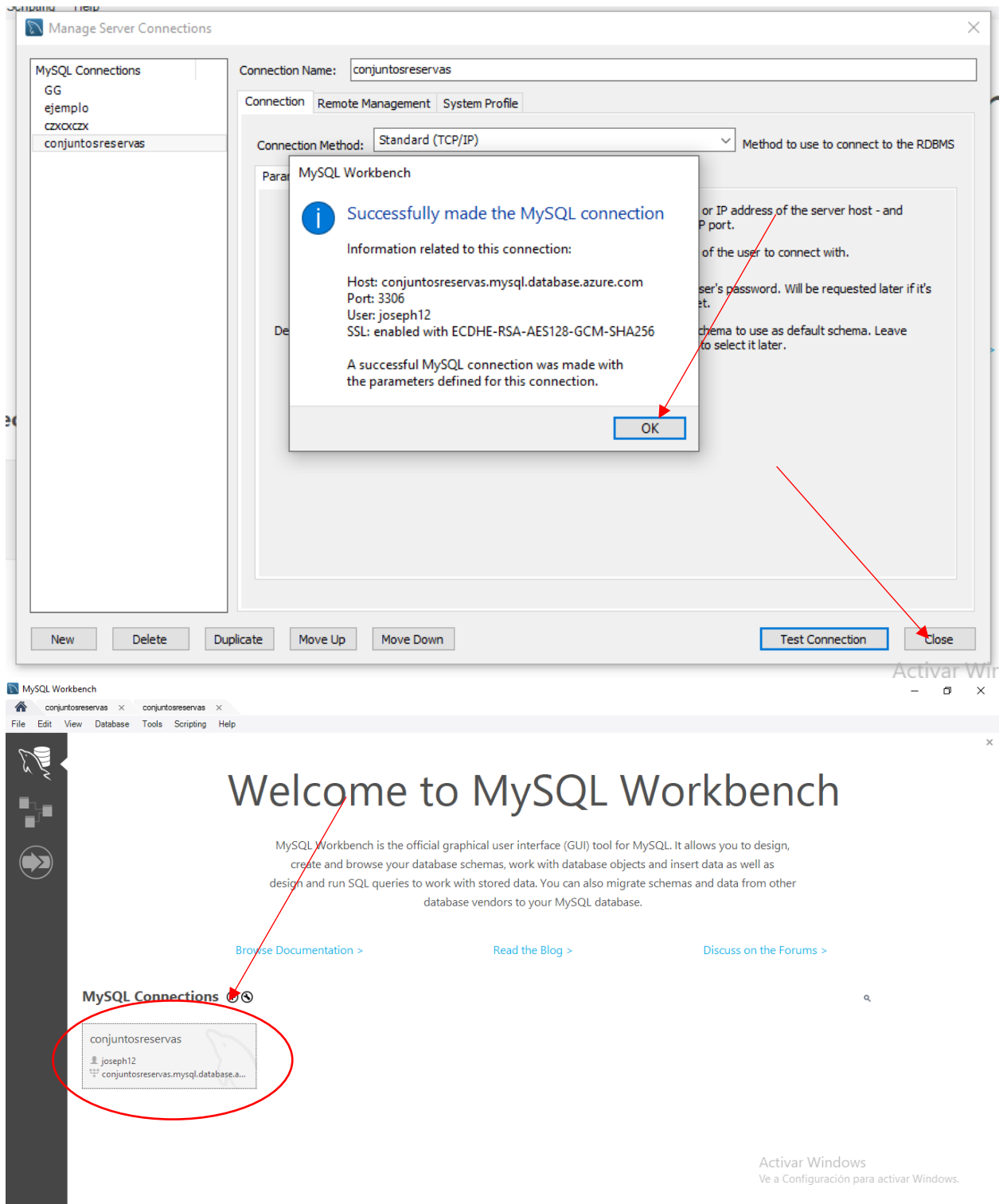
- Al ingresar debemos ubicar el repositorio clonado. Al ubicarlo ingresamos y deberíamos encontrar la carpeta llamada assets.



- Ingresamos y deberíamos encontrar un archivo llamado DigiCertGlobalRootCA.crt y lo seleccionamos. Ese es el archivo que debemos alojar en SSL CA File.
- Ahora se oprime el botón de test connection y esperamos a que se valide. Posibles errores concurrentes pueden ser por el nombre que dedicamos al nombre de la conexión o hacia la contraseña ingresada, pero siguiendo todos los pasos debería ser una conexión exitosa.

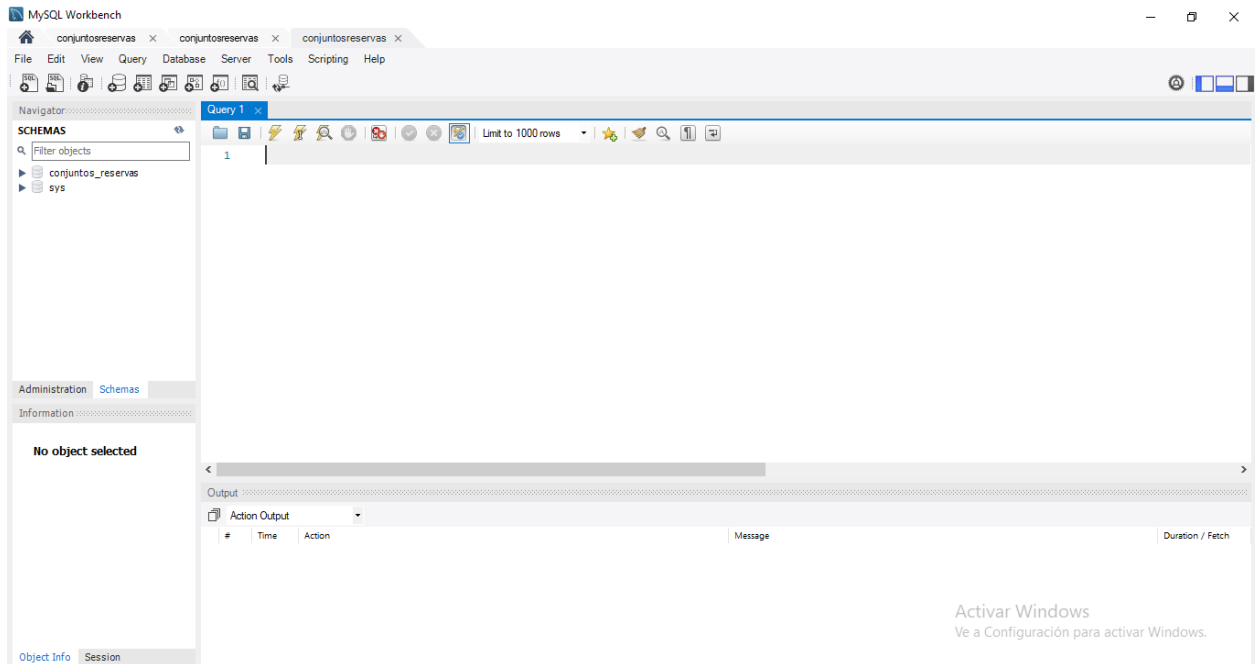


- Nos debería aparecer este mensaje. Seleccionamos ok y cerramos. Ahora la conexión existe y ya deberíamos ingresar. Nos pedirá la contraseña de nuevo, pero podemos dejar checkeado el mensaje para que no nos la pida de nuevo

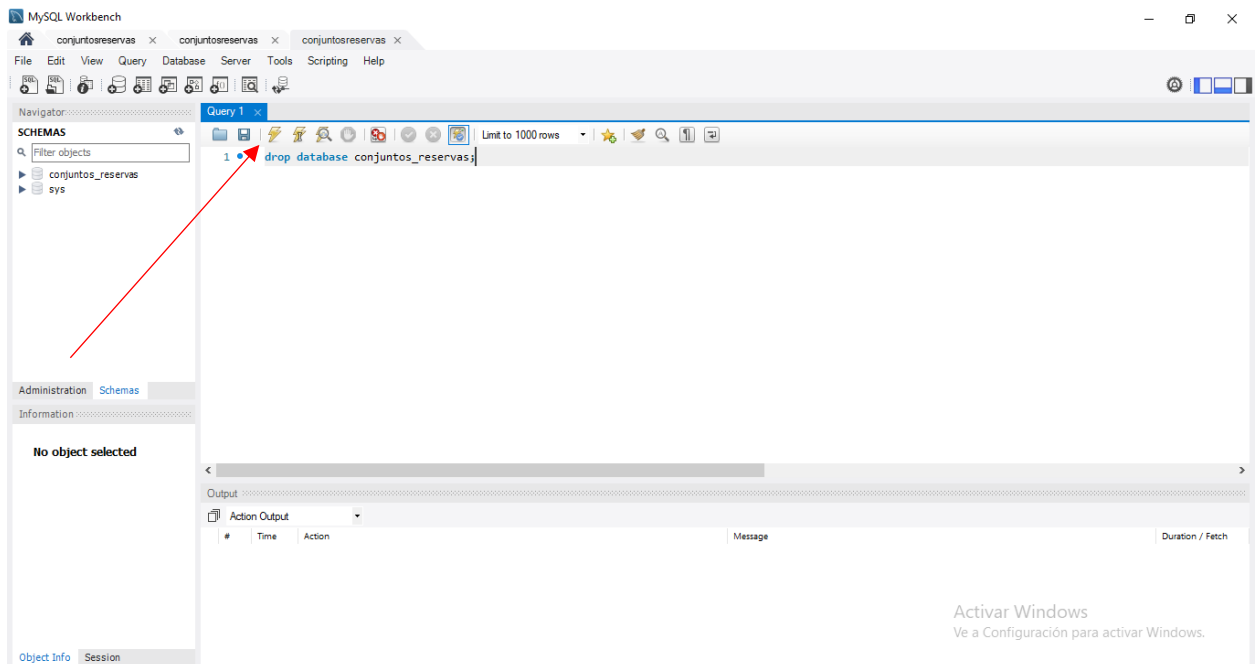


- Ingresamos y nos aparecerá la siguiente interfaz

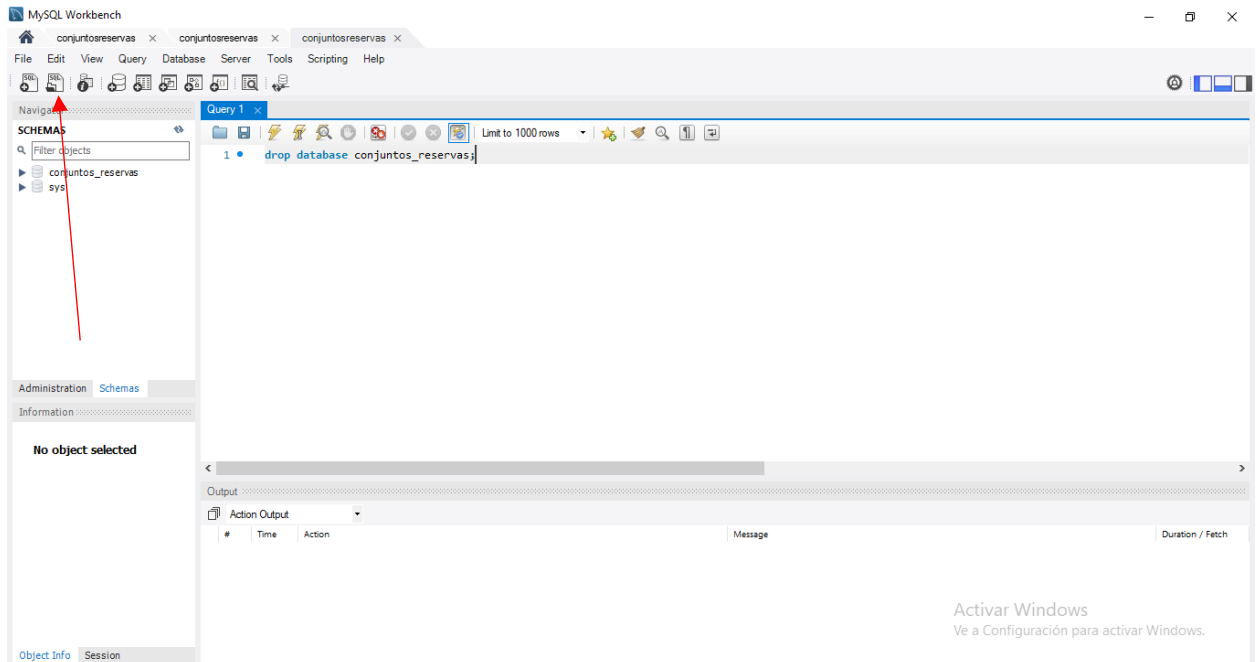




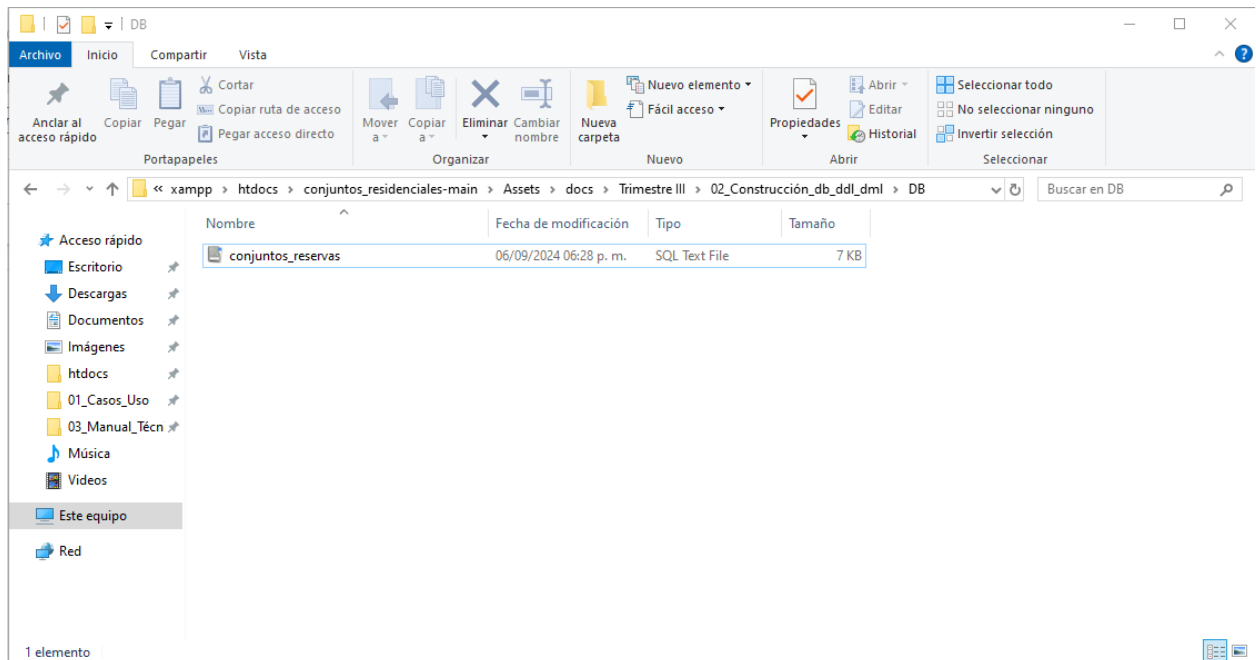
- Ingresamos la primera consulta para poder borrar la base de datos. Esto se hace por comodidad, en caso de urgencia en la que se deba eliminar. Esto se ejecutaría con `drop database conjuntos_reservas;` (para ejecutarlo debe estar la base de datos creada).
- Cuando se quiera ejecutar, simplemente se oprime el primer rayo que nos aparece en la parte superior izquierda. Así debería quedar.



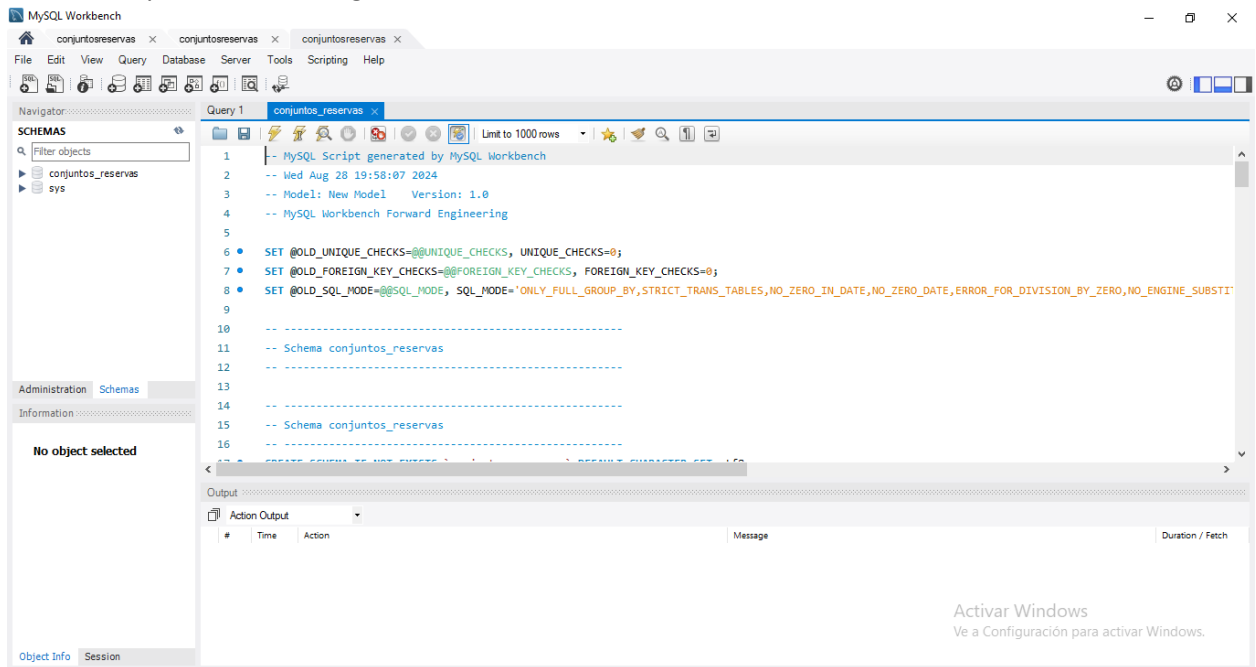
- Ya teniendo esa consulta ya iniciaríamos a cargar las sentencias ddl y dml en el icono que contiene una carpeta y el nombre sql.



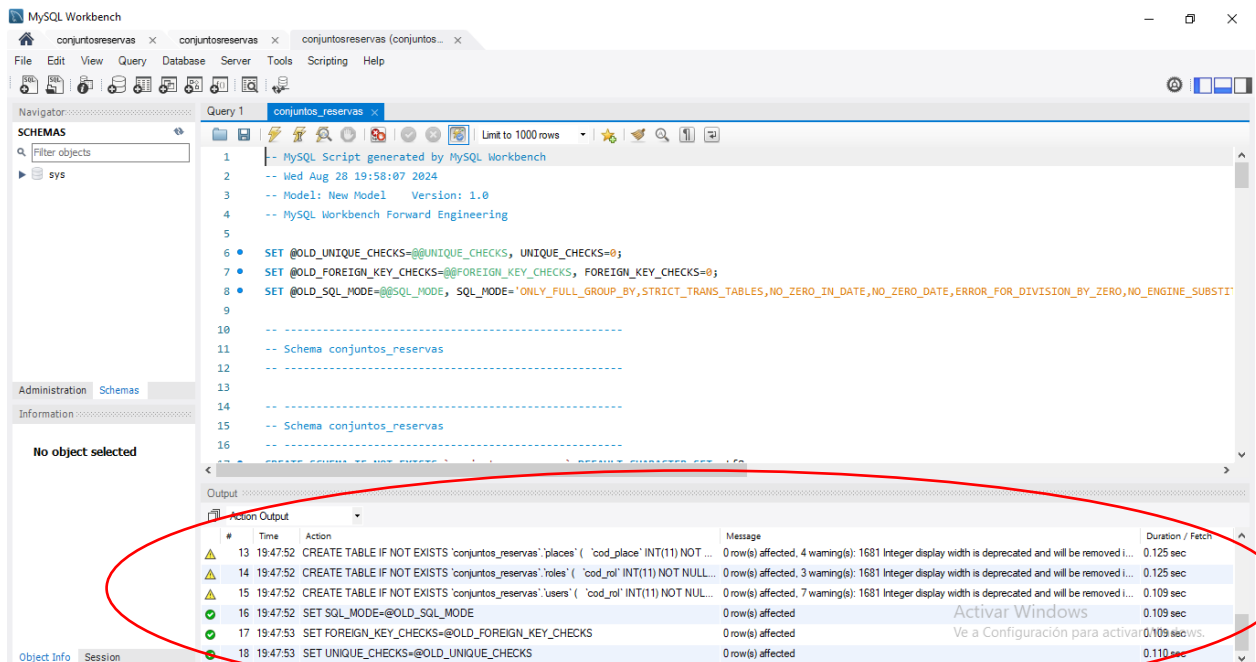
- Estas están alojadas en el mismo repositorio previamente clonado. Nos dirigimos a assets/docs/trimestre III y seleccionamos la carpeta que se llama 02\_Construcción\_db\_ddl\_dml y seleccionamos la carpeta que se llama DB Y abrimos el archivo que se llama conjuntos\_reservas.



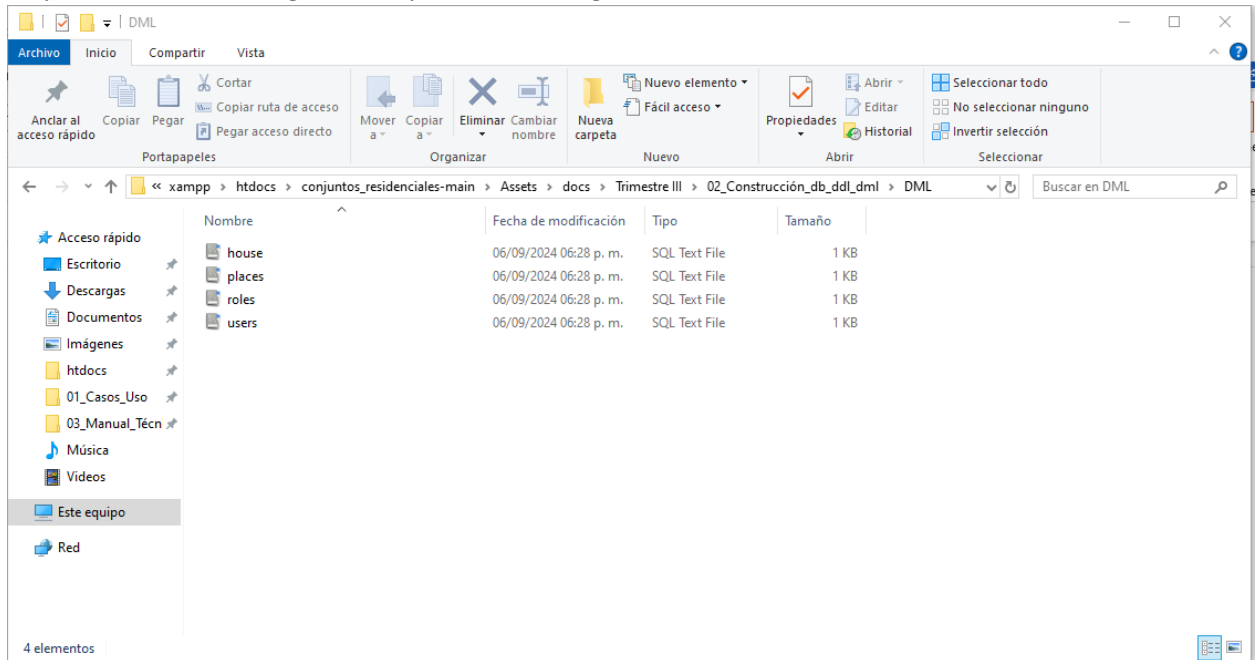
- Lo abrimos y se los verá lo siguiente.



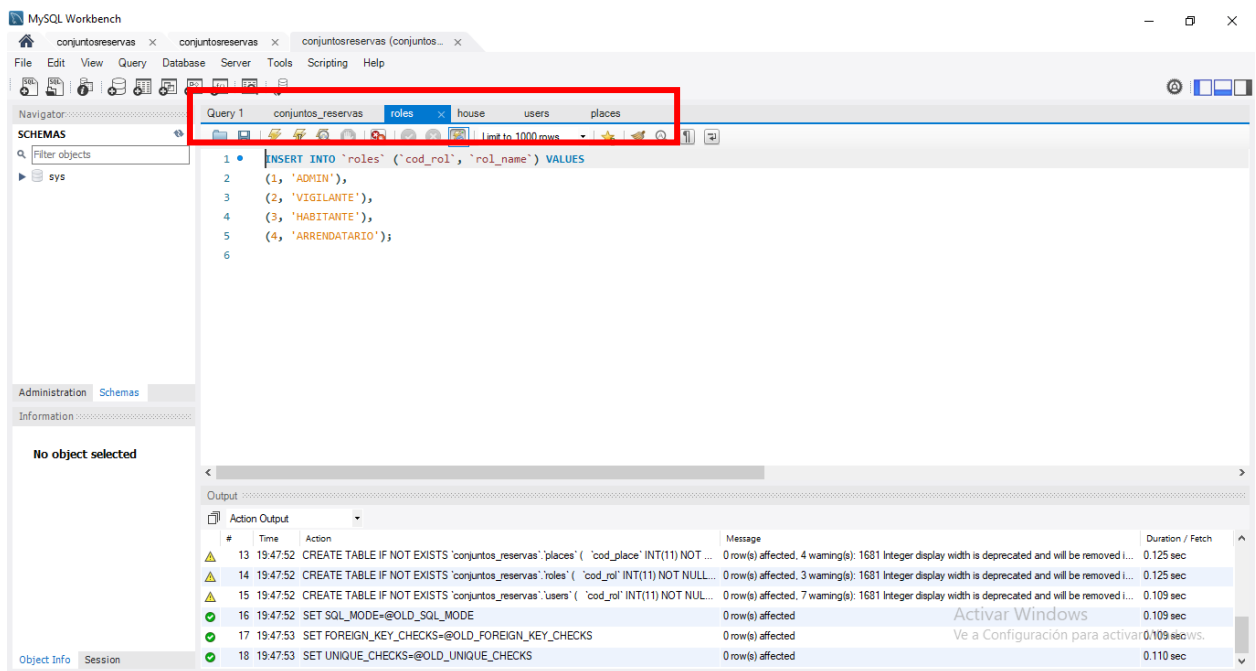
- Ya tendríamos la base de datos cargada. Para que se ejecute en azure simplemente le oprimimos al rayo y en la consola de la parte de abajo debería aparecer todos los iconos en verde y amarillo. Si hay uno en rojo lo mas probable es que se exporto mal la base de datos, se recomienda volverla a exportar.



- Ya tendríamos la bd cargada en azure. Ahora seria cargar los datos para estas tablas o bien conocido como sentencias dml. El proceso es el mismo con el que se cargo la base de datos solo que nos tendríamos que devolver y ubicar ahí mismo en 02\_Construcción\_db\_ddl\_dml la carpeta llamada dml, ingresamos y veríamos lo siguiente



- Realizamos el mismo proceso que con la carga de base de datos, abrimos uno por uno las carpetas en su preferencia en el orden de roles, house, users y places para no tener confusiones. Esto se debe ya que no se pueden registrar usuarios si no hay casas ni roles. Nos debería quedar así



- Ejecutamos en ese orden y ya deberíamos tener los datos llenos de cada tabla. En la tabla de reservas si se hace de manera manual en la vista ya que esta hecho para los usuarios. Este seria el paso a paso para la ejecución de la base de datos.