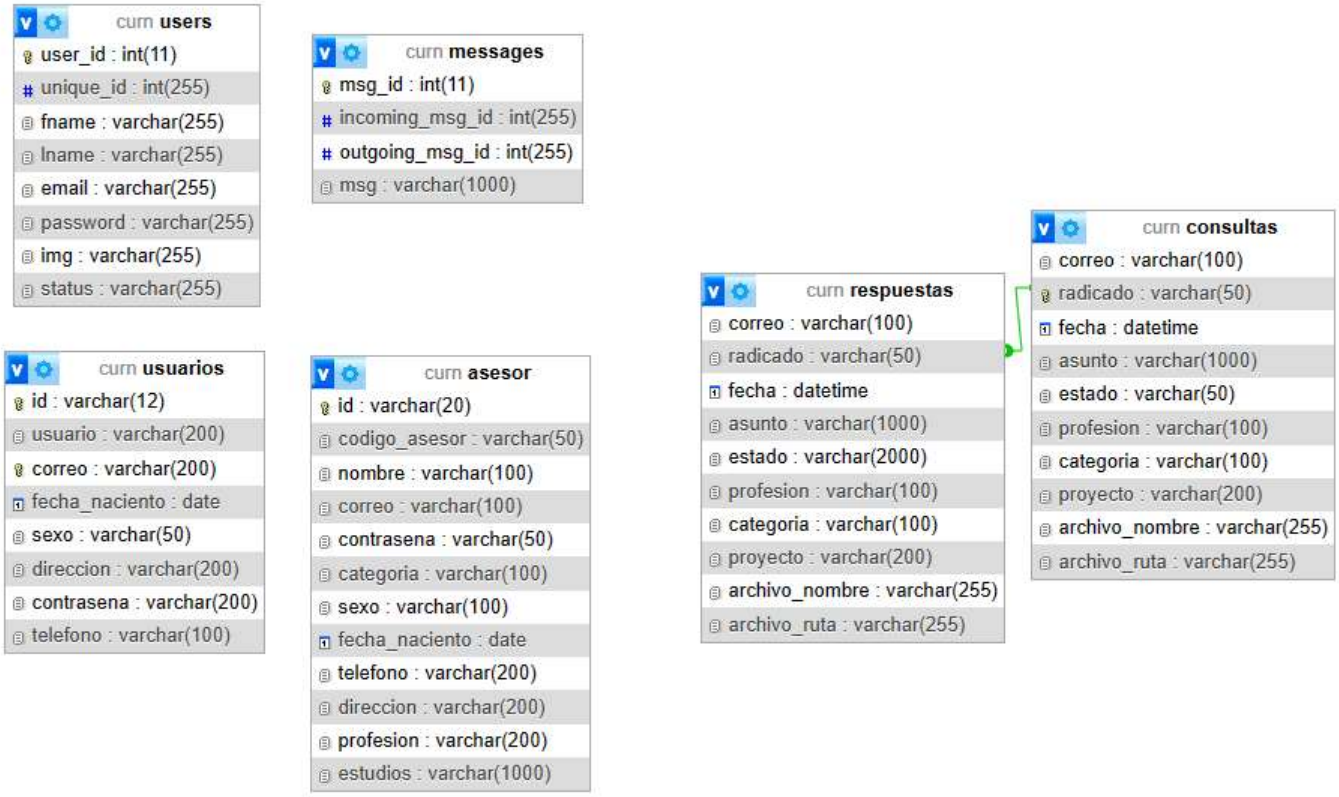
---------- ESTRUCTURA ----------

TABLA BASE DE DATOS

TABLAS Y TRIGGER



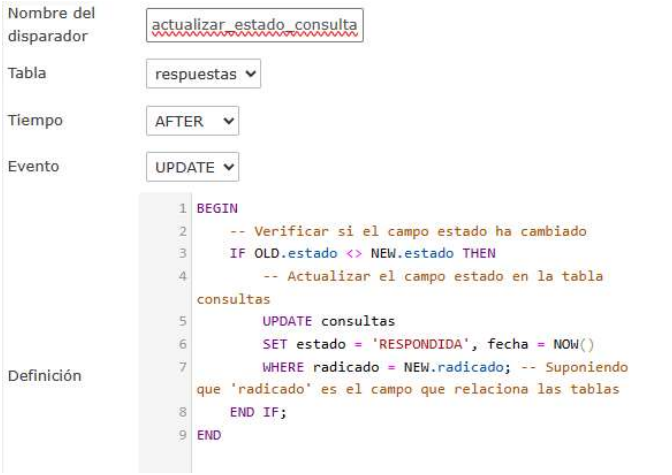
Podemos apreciar que nuestra base de datos tiene 6 tablas, y que solamente dos de ellas tienen relación entre sí, que son las tablas **Consultas** y **Respuestas**.

Dado que un usuario puede realizar múltiples consultas y, por lo tanto, el mismo correo puede aparecer en varias filas de la tabla consultas, no podemos establecer una relación directa de clave externa entre el correo en consultas y la tabla usuarios.
Para lograr toda la lógica de eliminación, inserción y actualización entre las tablas, por ejemplo: si se elimina un usuario, deberían desaparecer todas sus consultas, y si se elimina un registro, debería desaparecer esa consulta en todas las tablas.
Para lograr todo esto, utilizamos disparadores o triggers que se encargan de realizar acciones específicas.

CREACIÓN DE TRIGGER

Este trigger está diseñado para activarse después de insertar una fila en la tabla consultas e insertar una fila correspondiente en otra tabla llamada respuestas. De esta manera, estamos agregando datos a la tabla respuestas, a la que posteriormente los asesores tendrán acceso para responder las consultas.

Este trigger se activa después de que se actualiza el estado en la tabla respuestas. Su propósito principal es mantener sincronizados los estados entre las tablas respuestas y consultas, asegurando que cuando se actualice el estado de una respuesta en la tabla respuestas, se actualice automáticamente el estado correspondiente en la tabla consultas.



---------- ESTRUCTURA ----------

CREACIÓN DE TRIGGER

Este trigger se activará antes de eliminar una fila de la tabla usuarios. Eliminará todas las filas de la tabla consultas que tengan el mismo correo que el usuario que está siendo eliminado (OLD.correo). De esta manera, cuando elimines un usuario de la tabla usuarios, todas las consultas asociadas a ese usuario también se eliminarán automáticamente.

Entidad-relación (ER) para las tablas usuarios y consultas

Entidades:

Usuario: Representa a los usuarios registrados en el sistema.
Consulta: Representa las consultas realizadas por los usuarios.

Relaciones: Un usuario puede realizar varias consultas (relación uno a muchos) Cada consulta está asociada a un único usuario.

Codigo Frontend

```
<form class="mt-8 space-y-6" action="php\login\registro.php" method="POST">
    <div>
        <label for="email" class="block mb-2 text-sm font-medium text-gray-900 dark:text-white">Tu Correo</label>
        <input type="email" name="user" id="email" class="bg-gray-50 border border-gray-300 text-gray-900 text-sm rounded-lg focus:ring-blue-500
        focus:border-blue-500 block w-full p-2.5 dark:bg-gray-700 dark:border-gray-600 dark:placeholder-gray-400 dark:text-white dark:focus:ring-blue-5
        dark:focus:border-blue-500" placeholder="name@company.com" required />
    </div>
    <div>
        <label for="password" class="block mb-2 text-sm font-medium text-gray-900 dark:text-white">Tu Contraseña</label>
        <input type="password" name="pass" id="password" placeholder="••••••••" class="bg-gray-50 border border-gray-300 text-gray-900 text-sm rounded-lg
        focus:ring-blue-500 focus:border-blue-500 block w-full p-2.5 dark:bg-gray-700 dark:border-gray-600 dark:placeholder-gray-400 dark:text-white
        dark:focus:ring-blue-500 dark:focus:border-blue-500" required />
    </div>
    <div class="flex items-start">
        <div class="flex items-center h-5">
            <input id="remember" aria-describedby="remember" name="remember" type="checkbox" class="w-4 h-4 border-gray-300 rounded bg-gray-50 focus:ring-3
            focus:ring-blue-300 dark:focus:ring-blue-600 dark:ring-offset-gray-800 dark:bg-gray-700 dark:border-gray-600" required />
        </div>
        <div class="ms-3 text-sm">
        <label for="remember" class="font-medium text-gray-500 dark:text-gray-400">Autorizar tratamiento de datos</label>
        </div>
        <a data-modal-target="authentication-modal" data-modal-toggle="authentication-modal" style="color: #00a99e;" class="ms-auto text-sm font-medium text-write
        hover:underline cursor-pointer">Olvido su contraseña?</a>
    </div>
    <button type="submit" class="w-full px-5 py-3 text-base font-medium text-center text-white rounded-lg  focus:ring-4 focus:ring-blue-300 sm:w-auto"
    style="background-color: orange;" >Acceder</button>
    <button data-modal-target="authentication-modalAsesor" data-modal-toggle="authentication-modalAsesor" class="w-full px-5 py-3 text-base font-medium
    text-center text-white rounded-lg  focus:ring-4 focus:ring-blue-300 sm:w-auto" style="background-color: #00a99e;" >Asesor</button>

    <div class="text-sm font-medium text-gray-900 dark:text-white">
    ¿Todavía no estás registrado?  <a href="php\usuario\userRegistrer.php" style="color: orange;" class=" hover:underline cursor-pointer" >Crea una cuenta</a>
    </div>
</form>
```

Este es el código de inicio de sesión para acceder a la plataforma, el cual tiene un **action y método:** El formulario tiene un atributo action que apunta a un archivo PHP llamado "registro.php", y un atributo method establecido como "POST", lo que significa que cuando se envíe el formulario, los datos se enviarán a través del método HTTP POST a ese archivo PHP para su procesamiento.

Campos de entrada: Hay dos campos de entrada dentro del formulario:

- Uno para el correo electrónico (user), con un identificador email.
- Otro para la contraseña (pass), con un identificador password.

Este formulario es parte de un sistema de inicio de sesión y registro para usuarios, con funcionalidades adicionales para los asesores. Los datos ingresados por el usuario se envían a través de POST a un archivo PHP para su procesamiento, como el inicio de sesión o el registro de nuevos usuarios. Todos los datos son procesados y validados en el archivo registro.php, el cual se mostrará a continuación.

Codigo backend

```php
<?php
session_start();
include_once("../../bd/conexion.php");
// Obtener las variables del formulario
$user = $_POST['user'];
$pass = $_POST['pass'];

// Consulta SQL para verificar las credenciales del usuario
$query = "SELECT * FROM usuarios WHERE correo='$user' AND contrasena='$pass'";
$result = $db_connection->query($query);

// Verificar si el usuario y la contraseña son correctos
if ($result->num_rows > 0) {
    // Guardar las variables del usuario en sesiones
    $usuario = $result->fetch_assoc(); // Obtener el usuario de la base de datos
    $_SESSION['usuario'] = $usuario['usuario'];
    $_SESSION['correo'] = $usuario['correo'];

    // Redireccionar a principal.php
    header("Location: ../usuario\consultorio.php");
    exit();
} else {
    // Si las credenciales son inválidas, redireccionar a formulario.html
    header("Location: ../../index.php?error=1");
    exit();
}

// Cerrar la conexión
$db_connection->close();
```

Este es el código del backend para el inicio de sesión en la plataforma. En primer lugar, verificamos si hay una sesión activa. Si no hay una sesión activa, se redireccionará a la página de inicio de sesión. Si hay una sesión activa, recuperamos el usuario y la contraseña de la sesión. Luego verificamos las credenciales y, si son válidas, recuperamos la información del usuario, representada por las variables $user o $usuario, y redireccionamos al usuario a la página del consultorio, donde se mostrará la información respectiva.

En caso de que por algún motivo las credenciales de usuario o contraseña sean incorrectas o no se encuentren en la base de datos, se redireccionará a la página de inicio de sesión mostrando un mensaje de error.

Descripción basica del codigo frontend y backend

El código frontend mostrará al usuario la interfaz de usuario en la cual podrá iniciar sesión, crear una cuenta y cambiar la contraseña. Una vez que el usuario haya ingresado sus credenciales, entra en acción el código backend, el cual se encarga de verificar la sesión de las cookies. Una vez verificada la sesión, procederá a validar las credenciales de usuario y contraseña para poder redireccionar al usuario, ya sea al consultorio o a la página de inicio de sesión si algo sale mal.