DESARROLLO DE LA SEGUNDA TAREA\_V02

CENTRO GERONTOLÓGICO | UTVM-UNIVERSIDAD TECNOLOGICA DEL VALLE DEL MEZQUITAL

CENTRO GERONTOLÓGICO

SAYGON HERNÁNDEZ CATALÁN

2023

Contenido

[1. TAREA A DESARRLLAR 2](#_Toc148010673)

[2. DATOS PROPORCIONADOS A ANALIZAR Y DESARROLLAR 2](#_Toc148010674)

[3. MODELO ENTIDAD-RELACIÓN 4](#_Toc148010675)

[4. DIAGRAMA E-R DE BASE DE DATOS 5](#_Toc148010676)

[5. DICCIONARIO DE DATOS 6](#_Toc148010677)

[6. CODIGO 10](#_Toc148010678)

[7. PRUEBA UNITARIA 10](#_Toc148010679)

[Ilustración 1 Modelo E-R 4](file:///D:\10%20CUATRIMESTRE\DULCE%20SUGEY\TAREA_V02\05102023_DOR_DICCIONARIO_CG_2023_V02.docx#_Toc148010667)

[Ilustración 2 Modelado E-R 6](#_Toc148010668)

[Tabla 1 Tareas Asignadas 2](#_Toc148037469)

[Tabla 2 Diagrama E-R 5](file:///H:\10%20CUATRIMESTRE\DULCE%20SUGEY\TAREA_V02\05102023_DOR_DICCIONARIO_CG_2023_V02.docx#_Toc148037470)

[Tabla 3 Diccionario de datos usuario 7](#_Toc148037471)

[Tabla 4 Diccionario de datos tutor 8](#_Toc148037472)

[Tabla 5 Diccionario de datos talleres 9](#_Toc148037473)

[Tabla 6 Diccionario de datos servicios 9](#_Toc148037474)

[Tabla 7 Diccionario de datos administrador 10](#_Toc148037475)

# TAREA A DESARRLLAR

|  |  |  |  |
| --- | --- | --- | --- |
| **Tarea** | **Descripción** | **Tiempo Estimado** | **ESTATUS** |
| Modelo de la data base | Crear el modelo de la base de datos con la información | 2 días | FINALIZADO |
| Diagrama de la data base | Crear el diagrama de la base de datos | 2 días | FINALIZADO |
| Diccionario de datos | Diccionario de datos | 1 día | EN PROCESO |
| Código | Código documentado y versionado | 1 día | EN PROCESO |
| Prueba unitaria | Prueba del proyecto desarrollado | 1 día | EN PROCESO |

Tabla 1 Tareas Asignadas

# DATOS PROPORCIONADOS A ANALIZAR Y DESARROLLAR

“En esta información tenemos a un administrador el cual el tendrá el rol de hacer todo sin tener que tener un **registro** ni **login**”

Datos que deberá llevar el modelo y el diagrama:

Datos de la persona (Mayor de Edad)

* Nombre
* Apellidos
* (Normalizar BD)
* Teléfono
* Correo

Datos tutor

* Nombre de la persona responsable
* Teléfono de responsable
* Dirección

Talleres

* **Danza** (Huapango y Danzón) días **Martes y jueves.**
* **Pintura** (en tela) días **Lunes.**
* **Cocina** días **Miércoles y viernes**.

Servicios

* Atención Medica
* Atención Psicológica
* Activación Física

Generar reportes

# MODELO ENTIDAD-RELACIÓN

Ilustración 1 Modelo E-R

Reporte

pdf

ADMIN

SERVICIO

Nombre

Hora

Días

Registra

TALLER

Días

Hora

Nombre

Usuario

Nombre

Ape\_Pat

Ape\_Mat

Teléfono

Correo

Tiene

TUTOR

Nombre

Dirección

Teléfono

Genera

# DIAGRAMA E-R DE BASE DE DATOS

Usuario

Id\_usuario

Id\_Taller

Id\_Servicio

Nombre

Apellido

Teléfono

Email

Nombre\_tutor

Teléfono\_tutor

Address

PK

FK

FK

+

+

+

+

+

+

+

Taller

Id\_taller

Id\_usuario

Fecha

Hora

FK

FK

+

+

Servicio

Id\_servicio

Id\_usuario

Fecha

Hora

Id\_doctor

FK

+

+

+

+

Administrador

PK

+

+

+

+

+

Id\_Admin

Nombre

Ape\_Pat

Ape\_Mat

Teléfono

Correo

Tabla 2 Diagrama E-R

# DICCIONARIO DE DATOS

Las tablas siguientes proporcionan una descripción completa de cada campo de la base de datos. Esta descripción incluye:

* + Nombre del campo.
  + Clave.
  + La Clave Primaria (**PK**) es una columna o conjunto de columnas que identifica de forma unívoca los datos restantes de una fila cualquiera de la tabla.
  + La Clave Foránea (**FK**) es una columna de una tabla que es una clave primaria de otra tabla; por tanto, cualquier dato contenido en una columna de clave foránea debe tener datos correspondientes en la otra tabla donde esa columna es la clave primaria. En el caso de las bases de datos, esta correspondencia se denomina integridad referencial.
  + El valor NOT NULL (**NN**) significa que la columna debe tener un valor en cada fila. Si se utiliza NULL, esa columna se puede dejar vacía en una fila determinada.
  + Tipo de datos.
  + Tamaño del campo.
  + Una descripción de la finalidad de cada campo.

**Tabla 1: Diccionario de datos de usuario**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TITULO** | **Datos cliente** | | **Fecha de elaboración** | **05/10/2023** |
| **Nombre de**  **campo** | **Clave** | **Tipo de**  **dato** | **Tamaño del campo**  **(Bytes)** | **Descripción del campo** |
| ID\_Usuario | PRIMARY  KEY | VARCHAR | 30 | Identificador del usuario, llave primaria. |
| Nombre |  | VARCHAR | 30 | Nombre del usuario. |
| Apellidos |  | VARCHAR | 20 | Guarda los Apellidos |
| Teléfono |  | VARCHAR | 20 | Guarda el teléfono |
| Email |  | VARCHAR | 30 | Guarda el Email |
| Nombre\_tutor |  | VARCHAR | 250 | Guarda el nombre del tutor |
| Telefono\_tutor |  | VARCHAR | 15 | Guarda el teléfono del tutor |
| Address |  | VARCHAR | 250 | Guarda la dirección |

Tabla 3 Diccionario de datos usuario

**Código SQL**

CREATE TABLE usuario s(

ID\_Usuario VARCHAR(30) PRIMARY KEY,

Nombre VARCHAR(30),

apellidos VARCHAR(20),

Teléfono VARCHAR(20),

Email VARCHAR(30),

Nombre\_tutor VARCHAR(250),

Telefono\_tutor VARCHAR(15),

Address VARCHAR(250)

);

**Tabla 3: Diccionario de datos de talleres**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TITULO** | **Datos cliente** | | **Fecha de elaboración** | **05/10/2023** |
| **Nombre de**  **campo** | **Clave** | **Tipo de**  **dato** | **Tamaño del campo**  **(Bytes)** | **Descripción del campo** |
| ID\_Taller | PRIMARY  KEY | VARCHAR | 30 | Identificador del taller, llave primaria. |
| Fecha |  | DATE | 30 | Guarda la fecha |
| Hora |  | TIME | 30 | Guarda la hora |
| FOREIGN KEY(ID) REFERENCES usuarios(ID) )ENGINE=INNODB; | | | | |

Tabla 5 Diccionario de datos talleres

**Código SQL**

CREATE TABLE talleres (

ID\_Taller VARCHAR(30) PRIMARY KEY,

Fecha DATE,

Hora TIME,

FOREIGN KEY (ID\_taller) REFERENCES usuario(ID\_usua)

);

**Tabla 4: Diccionario de datos de servicios**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TITULO** | **Datos cliente** | | **Fecha de elaboración** | **05/10/2023** |
| **Nombre de**  **campo** | **Clave** | **Tipo de**  **dato** | **Tamaño del campo**  **(Bytes)** | **Descripción del campo** |
| ID\_Servicios | PRIMARY  KEY | VARCHAR | 30 | Identificador del servicio, llave primaria. |
| Fecha |  | DATE | 30 | Guarda la fecha |
| Hora |  | TIME | 30 | Guarda la hora |
| FOREIGN KEY(ID) REFERENCES usuarios(ID) )ENGINE=INNODB; | | | | |

Tabla 6 Diccionario de datos servicios

**Código SQL**

CREATE TABLE servicios (

ID\_Servicios VARCHAR(30) PRIMARY KEY,

Fecha DATE,

Hora TIME,

FOREIGN KEY (ID\_servicios) REFERENCES usuarios(ID\_usua)

);

**Tabla 5: Diccionario de datos de administrador**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TITULO** | **Datos cliente** | | **Fecha de elaboración** | **05/10/2023** |
| **Nombre de**  **campo** | **Clave** | **Tipo de**  **dato** | **Tamaño del campo**  **(Bytes)** | **Descripción del campo** |
| ID\_Admin | PRIMARY  KEY | VARCHAR | 30 | Identificador del servicio, llave primaria. |
| Fecha |  | DATE | 30 | Guarda la fecha |
| Hora |  | TIME | 30 | Guarda la hora |
| FOREIGN KEY(ID) REFERENCES usuarios(ID) )ENGINE=INNODB; | | | | |

Tabla 7 Diccionario de datos administrador

CREATE TABLE administrador(

ID\_Admin VARCHAR(30) PRIMARY KEY,

Nombre VARCHAR(30),

Apellidos VARCHAR(20),

Teléfono VARCHAR(20),

Email VARCHAR(30)

);

# CODIGO

* **Pantalla principal**

Código que se utilizó para la funcionalidad de transición y animación de las imágenes para la pantalla principal del sistema utilizando **<sript>**.

<!-- Initialize Swiper -->

    <script>

      var swiper = new Swiper(".mySwiper", {

        effect: "coverflow",

        grabCursor: true,

        centeredSlides: true,

        slidesPerView: "auto",

        coverflowEffect: {

          rotate: 15,

          stretch: 0,

          depth: 300,

          modifier: 1,

          slideShadows: true,

        },

        loop: true,

        autoplay: {

          delay: 3000, // 3000 milisegundos (3 segundos) entre diapositivas

          disableOnInteraction: false, // Permite que el autoplay continúe después de la interacción del usuario

        },

      });

    </script>

* **Registro Formulario**

Código que se utilizó para guardar los datos del formulario

Register.html

<form action="registrar.php" method="post" onsubmit="return validarFormulario()"> <!-- Funcionalidad para guardar los datos con el metodo post-->

Código que se utilizó para validar los checkbox que se utilizan para guardar datos.

Register.html

<script>

function validarFormulario() {

var checkboxes = document.querySelectorAll('input[type="checkbox"]');

var noneCheckbox = document.getElementById('preferencia4');

var alMenosUnCheckboxMarcado = false;

checkboxes.forEach(function (checkbox) {

if (checkbox.checked) {

alMenosUnCheckboxMarcado = true;

}

});

if (!alMenosUnCheckboxMarcado && !noneCheckbox.checked) {

alert('Debes seleccionar al menos un taller o marcar "Ninguna" antes de guardar.');

return false; // Evita que el formulario se envíe

}

return true; // Permite que el formulario se envíe si al menos un checkbox está marcado

}

// Agrega un evento para validar las selecciones de talleres antes del envío del formulario

var checkboxes = document.querySelectorAll('input[type="checkbox"]');

var noneCheckbox = document.getElementById('preferencia4');

noneCheckbox.addEventListener('change', function() {

if (noneCheckbox.checked) {

// Si "Ninguna" se marca, desmarca las otras opciones

checkboxes.forEach(function(checkbox) {

if (checkbox !== noneCheckbox) {

checkbox.checked = false;

}

});

}

});

checkboxes.forEach(function(checkbox) {

checkbox.addEventListener('change', function() {

if (checkbox !== noneCheckbox && checkbox.checked) {

// Si se marca otra opción, desmarca "Ninguna"

noneCheckbox.checked = false;

}

});

});

</script>

Codigo php que se utilizo para la conexión a la base de datos y guardar los datos con el método **POST** en un archivo llamado **registrar.php**

<?php

// ... Código de conexión a la base de datos ...

$servername = "localhost";

$username = "root";

$password = "App-Adult-May";

$dbname = "centro\_gerontologico";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect\_error) {

    die("Conexión fallida: " . $conn->connect\_error);

}

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $nombre = $\_POST["nombre"];

    $apellido = $\_POST["apellido"];

    $telefono = $\_POST["telefono"];

    $email = $\_POST["email"];

    $nombre\_tutor = $\_POST["nombre\_tutor"];

    $telefono\_tutor = $\_POST["telefono\_tutor"];

    $address = $\_POST["address"];

    // Verifica si se seleccionó un taller

    $preferencia1 = isset($\_POST["preferencia1"]) ? 1 : 0;

    $preferencia2 = isset($\_POST["preferencia2"]) ? 1 : 0;

    $preferencia3 = isset($\_POST["preferencia3"]) ? 1 : 0;

    $preferencia4 = isset($\_POST["preferencia4"]) ? 1 : 0;

    // Prepara la consulta SQL para insertar los datos

    $sql = "INSERT INTO usuarios (nombre, apellido, telefono, email, nombre\_tutor, telefono\_tutor, address, preferencia1, preferencia2, preferencia3, preferencia4)

    VALUES ('$nombre', '$apellido', '$telefono', '$email', '$nombre\_tutor', '$telefono\_tutor', '$address', $preferencia1, $preferencia2, $preferencia3, $preferencia4)";

if ($conn->query($sql) === TRUE) {

    $message = "Registro exitoso";

    // Redirige al usuario al formulario con un parámetro GET para indicar éxito

    header("Location: Register.html?success=1");

    exit; // Asegura que el script termine después de la redirección

} else {

    $message = "Error: " . $sql . "<br>" . $conn->error;

}

// Imprime el mensaje en JavaScript para mostrarlo como un mensaje flotante

echo "<script>

  var messageContainer = document.getElementById('message-container');

  messageContainer.innerHTML = '$message';

  messageContainer.style.display = 'block';

  setTimeout(function() {

    messageContainer.style.display = 'none';

  }, 5000); // 5000 milisegundos = 5 segundos

</script>";

}

// Cierra la conexión a la base de datos

$conn->close();

?>

* **Login usurario**

**Loginusuario.html**: Se utilizó para guardar los datos mediante el método **POST**.

<form action="loginusuario.php" method="post">

**Loginusuario.php**: Se utilizó para la conexión a la base de datos donde se valida los datos que utilizamos para poder iniciar la sesión mas segura.

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $telefono = $\_POST["telefono"];

    // Conectar a la base de datos (reemplaza estos valores con los tuyos)

    $servername = "localhost";

    $username = "root";

    $password = "App-Adult-May";

    $database = "centro\_gerontologico";

    $conn = new mysqli($servername, $username, $password, $database);

    if ($conn->connect\_error) {

        die("Conexión fallida: " . $conn->connect\_error);

    }

    // Consulta SQL para verificar el número de teléfono

    $sql = "SELECT \* FROM usuarios WHERE telefono = '$telefono'";

    $result = $conn->query($sql);

    if ($result->num\_rows == 1) {

        // Inicio de sesión exitoso, obtén el nombre del usuario

        $row = $result->fetch\_assoc();

        $user\_name = $row["nombre"];

        // Inicia una sesión para mantener al usuario conectado

        session\_start();

        $\_SESSION["user\_name"] = $user\_name;

        // Redirige al usuario a la página de perfil

        header("Location: perfil.php");

        exit();

    } else {

        // Usuario no válido, redirige al formulario de inicio de sesión con un mensaje de error

        header("Location: login.html?error=1");

        exit();

    }

    // Cierra la conexión a la base de datos

    $conn->close();

}

?>

**Perfil.php**: Se utiliza para mantener la sesión antes mencionada donde aparecerá el perfil del usuario con su nombre y las opciones para actualizar datos y cerrar sesión.

<?php

session\_start();

if (!isset($\_SESSION["user\_name"])) {

    // Si el usuario no está conectado, redirige al formulario de inicio de sesión

    header("Location: login.html");

    exit();

}

$user\_name = $\_SESSION["user\_name"];

?>

<div class="profile">

<div class="profile-name">Usuario: <?php echo $user\_name; ?></div>

      <div class="floating-menu">

        <ul class="profile-options">

          <li><a href="editarperfil.php">Actualizar datos</a></li>

          <li><a href="cerrarsesion.php">Cerrar Sesión</a></li>

      </ul>

</div>

</div>

**editarperfil.php**: Se utilizó para poder actualizar los datos del perfil ya guardados en la base de datos.

<?php

session\_start();

if (!isset($\_SESSION["user\_name"])) {

    // Si el usuario no está conectado, redirige al formulario de inicio de sesión

    header("Location: login.html");

    exit();

}

// Obtén el nombre de usuario del usuario actual

$user\_name = $\_SESSION["user\_name"];

// Aquí debes conectar a tu base de datos y recuperar los datos del usuario

// basado en el nombre de usuario y asignarlos a las variables correspondientes

// Reemplaza con tu propio código de conexión y consulta SQL

$mysqli = new mysqli("localhost", "root", "App-Adult-May", "centro\_gerontologico");

if ($mysqli->connect\_error) {

    die("La conexión a la base de datos ha fallado: " . $mysqli->connect\_error);

}

$query = "SELECT nombre, apellido, telefono, email, nombre\_tutor, telefono\_tutor, address, danza, pintura, cocina, ninguno FROM usuarios WHERE nombre = '$user\_name'";

$result = $mysqli->query($query);

if ($result->num\_rows > 0) {

    $row = $result->fetch\_assoc();

    $nombre = $row["nombre"];

    $apellido = $row["apellido"];

    $telefono = $row["telefono"];

    $email = $row["email"];

    $nombre\_tutor = $row["nombre\_tutor"];

    $telefono\_tutor = $row["telefono\_tutor"];

    $address = $row["address"];

    $danza\_actual = $row["danza"];

    $pintura\_actual = $row["pintura"];

    $cocina\_actual = $row["cocina"];

    $ninguno\_actual = $row["ninguno"];

}

$mysqli->close();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="styles.css">

    <link rel="stylesheet" href="assets/css/registro.css">

    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/swiper@9/swiper-bundle.min.css" />

    <link rel="shortcut icon" href="assets/iconos/icono.png">

</head>

<body>

    <!-- Formulario de edición de perfil -->

    <div class="container">

        <form action="procesar\_actualizacion.php" method="POST" onsubmit="return validarFormulario()">

            <h2 class="title">Editar Perfil</h2>

            <img src="assets/iconos/icono.png" alt="Logo" class="logo1">

            <div class="column">

                <h4>Datos del adulto mayor</h4>

                <div class="nombre">

                    <div class="row">

                        <input type="text" id="nombre" name="nombre" pattern="[A-Za-záéíóúÁÉÍÓÚüÜñÑ\s]+" required placeholder="Nombre(s)" value="<?php echo $nombre; ?>">

                    </div>

                    <div class="row">

                        <input type="text" id="apellido" name="apellido" pattern="[A-Za-záéíóúÁÉÍÓÚüÜñÑ\s]+" required placeholder="Apellidos" value="<?php echo $apellido; ?>">

                    </div>

                </div>

                <div class="datos">

                    <div class="row">

                        <input type="tel" id="telefono" name="telefono" pattern="[0-9]{10}" placeholder="Teléfono (10 dígitos)" value="<?php echo $telefono; ?>">

                    </div>

                    <div class="row">

                        <input type="email" id="email" name="email" required placeholder="Correo electrónico" value="<?php echo $email; ?>">

                    </div>

                </div>

            </div>

            <div class="tutor">

                <div class="column">

                    <h4>Datos del tutor</h4>

                    <div class="row">

                        <input type="text" id="nombre\_tutor" name="nombre\_tutor" pattern="[A-Za-záéíóúÁÉÍÓÚüÜñÑ\s]+" required placeholder="Nombre del tutor" value="<?php echo $nombre\_tutor; ?>">

                    </div>

                    <div class="row">

                        <input type="tel" id="telefono\_tutor" name="telefono\_tutor" pattern="[0-9]{10}" placeholder="Teléfono del tutor (10 dígitos)" value="<?php echo $telefono\_tutor; ?>">

                    </div>

                    <div class="row">

                        <input type="text" id="address" name="address" pattern="[A-Za-0-9-záéíóúÁÉÍÓÚüÜñÑ\s]+" required placeholder="Dirección" value="<?php echo $address; ?>">

                    </div>

                </div>

                <div class="checkbox-row">

                    <h4>¿Gusta inscribirse a un taller?</h4>

                    <br>

                    <label for="danza"><input type="checkbox" id="danza" name="danza" <?php if ($danza\_actual) echo 'checked'; ?>> Danza</label><br>

                    <label for="pintura"><input type="checkbox" id="pintura" name="pintura" <?php if ($pintura\_actual) echo 'checked'; ?>> Pintura</label><br>

                    <label for="cocina"><input type="checkbox" id="cocina" name="cocina" <?php if ($cocina\_actual) echo 'checked'; ?>> Cocina</label><br>

                    <label for="ninguno"><input type="checkbox" id="ninguno" name="ninguno" <?php if ($ninguno\_actual) echo 'checked'; ?>> Ninguna</label>

                </div>

            </div>

            <button type="submit" class="submit-button">Guardar Cambios</button>

        </form>

    </div>

    <div id="success-message" class="message">¡Actualización exitosa!</div>

    <div id="error-message" class="message">¡Hubo un error!</div>

    <script>

    var urlParams = new URLSearchParams(location.search);

    if (urlParams.get('success') === '1') {

        var successMessage = document.getElementById('success-message');

        successMessage.style.display = 'block';

        setTimeout(function() {

            successMessage.style.display = 'none';

        }, 5000); // 5000 milisegundos = 5 segundos

    } else if (urlParams.get('success') === '0') {

        var errorMessage = document.getElementById('error-message');

        errorMessage.style.display = 'block';

        setTimeout(function() {

            errorMessage.style.display = 'none';

        }, 5000); // 5000 milisegundos = 5 segundos

    }

    function validarFormulario() {

        var checkboxes = document.querySelectorAll('input[type="checkbox"]');

        var noneCheckbox = document.getElementById('ninguno');

        var alMenosUnCheckboxMarcado = false;

        checkboxes.forEach(function (checkbox) {

            if (checkbox.checked) {

                alMenosUnCheckboxMarcado = true;

            }

        });

        if (!alMenosUnCheckboxMarcado && !noneCheckbox.checked) {

            alert('Debes seleccionar al menos un taller o marcar "Ninguna" antes de guardar.');

            return false; // Evita que el formulario se envíe

        }

        return true; // Permite que el formulario se envíe si al menos un checkbox está marcado

    }

    // Agrega un evento para validar las selecciones de talleres antes del envío del formulario

    var checkboxes = document.querySelectorAll('input[type="checkbox"]');

    var noneCheckbox = document.getElementById('ninguno');

    noneCheckbox.addEventListener('change', function() {

        if (noneCheckbox.checked) {

            // Si "Ninguna" se marca, desmarca las otras opciones

            checkboxes.forEach(function(checkbox) {

                if (checkbox !== noneCheckbox) {

                    checkbox.checked = false;

                }

            });

        }

    });

    checkboxes.forEach(function(checkbox) {

        checkbox.addEventListener('change', function() {

            if (checkbox !== noneCheckbox && checkbox.checked) {

                // Si se marca otra opción, desmarca "Ninguna"

                noneCheckbox.checked = false;

            }

        });

    });

</script>

</body>

</html>

**Cerrarsesion.php**: Se utilizo para que la sesión del perfil se cierre asi como la base de datos.

<?php

session\_start();

session\_unset();

session\_destroy();

header("Location: login.html");

exit();

?>

# PRUEBA UNITARIA