

16-11-2021

ULTIMA  
MEMORIAS

Cumplimentación proyecto 1



## Contenido

Apartado ANTECEDENTES .....	3
Apartado REQUISITOS FUNCIONALES .....	4
Apartado Prototipo WEB y boceto estructura .....	5
Apartado Guía de estilos .....	6
Apartado Planificación de tareas .....	7
Diseño de base de datos .....	8
• Diseño Entidad Relación de la BBDD .....	8
• Modelo relacional BBDD .....	8
• Script de creación BBDD .....	9
• Consultas .....	19
Validación de formularios .....	30
Proceso de carga .....	31
Jerarquía de directorios .....	32
Contenido directorios .....	32
Apartado Diseño de la interface .....	33
Estructura gráfica de la interface .....	33
IMPLEMENTACIÓN .....	34
REQUISITO 1: Diseño responsive.....	34
Funcionamiento .....	34
Ejemplo de código .....	34
REQUISITO 2: Para que se pueda acceder a la pagina web .....	35
Funcionamiento .....	35
Ejemplo de código .....	35
PRUEBAS.....	36
Metodología de las pruebas.....	36
DESPLIEGUE.....	37
Apartado Herramientas. Por lo menos enumeración .....	39
Apartado Lenguajes. Por lo menos enumeración. ....	39
PRODUCTO .....	40
Página de Inicio .....	40
Página iniciar sesión .....	40
Página de registro.....	41
Página de gestión (tablero) .....	41

Página de editar difunto.....	42
Página de configuración de perfil.....	42
Abre una ventana emergente de un servicio .....	43
Página pedidos .....	43
Apartado Bibliografía. Las que se hayan usado hasta el momento. ....	44

## Apartado ANTECEDENTES

Se trata el proyecto de una página web que presta servicios de mantenimiento de lapidas, panteones y nichos los servicios que presta son oraciones, limpiezas y colocarles flores elegido todo esto al gusto de cliente (es decir tipo de oración según su libro sagrado y cual quiere que se lea y el tipo de flores que quiere que le ponga y la limpieza dependiendo de qué tipo de enterramiento tenga le pondremos un precio u otro es decir más cara o barata por ejemplo no es lo mismo limpiar una lápida que es más pequeña que un panteón). Este proyecto se pensó para melilla que mucha gente vive afuera y se deja su seres queridos aquí claro no va estar todas las veces viajando todo el rato para cuidar donde enterraron sus seres querido ya no solo por tema económico de viajar todo el rato es el tiempo que hoy en día no suele tener mucho tiempo y encima vino una pandemia que con más razón no podían viajar.

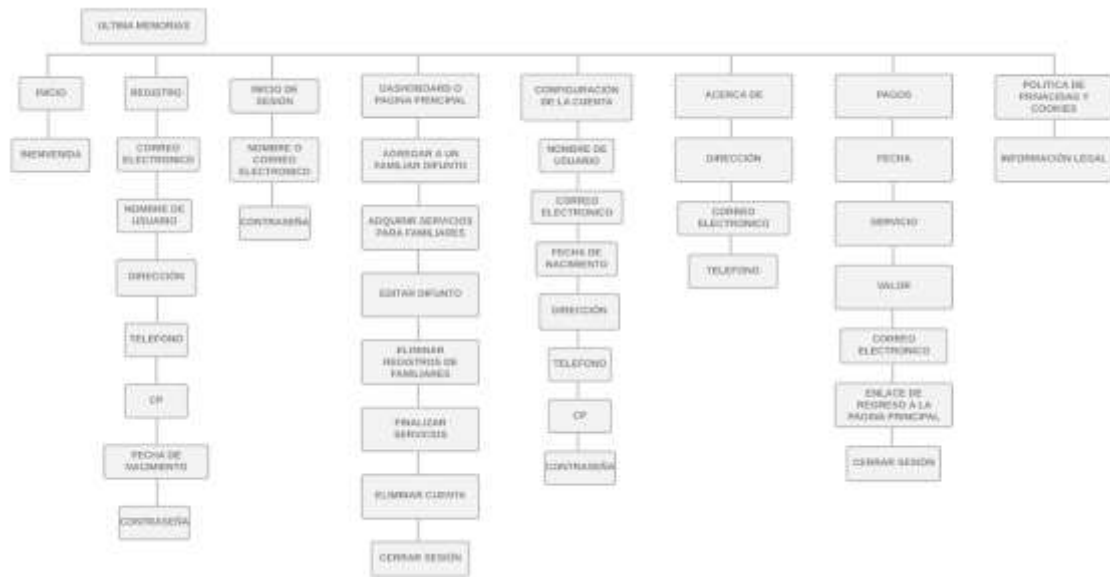
Para hacer el diseño se implementó en papel y la base de datos se pasó a página web mediante la página principal es index.html se añadió el diseño que se hizo anteriormente con sus respectivo Bootstrap que se adapta a cualquier dispositivo, su funciones de pasar la imagen y las letras con javascript y con php me permite registrarme e iniciar sesión modificar los registros del difunto de usuario y también eliminar el difunto y dar de baja el usuario por último el usuario puede elegir entre nuestros servicios para aplicárselos a difunto que son oraciones, colocar las flores y limpieza y tenemos estilo con css para los container.

## **Apartado REQUISITOS FUNCIONALES**

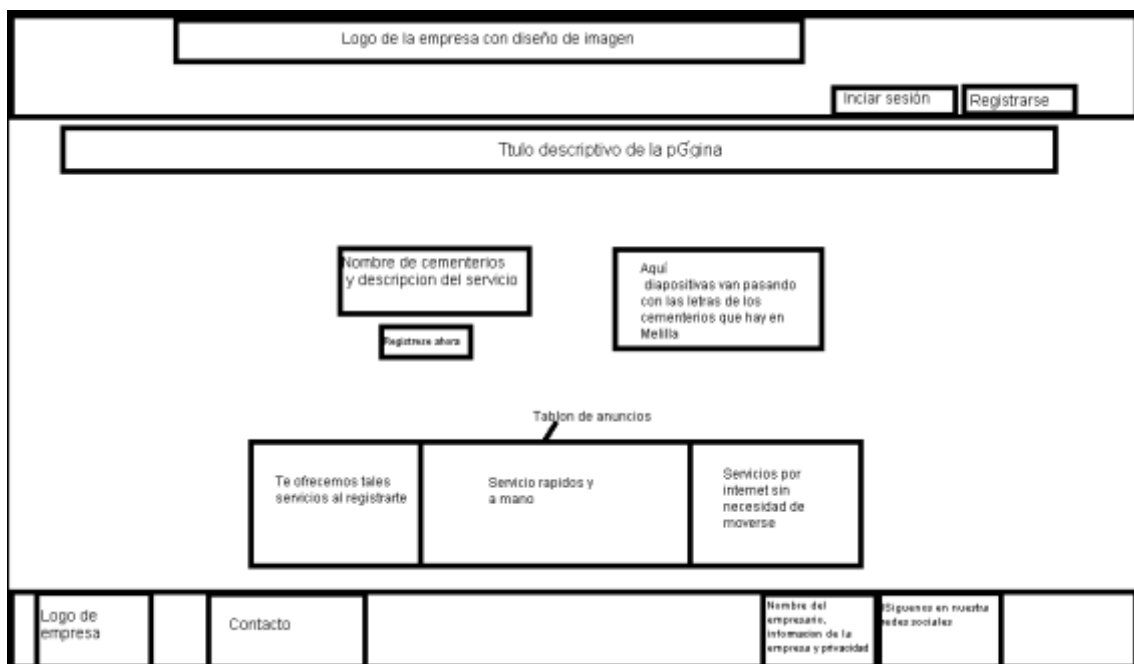
- Permite la autenticación
- Permite modificar y darse de baja los datos del usuario (elegible desde un menú desplegable del nombre del usuario)
- Rellenar, eliminar y modificar los datos del difunto (a través de un botón)
- Rellenar, eliminar y modificar los datos donde está tipo de enterramiento (elegible desde un desplegable)
- Contacta con la empresa(pie de página)
- Cancela el servicio que pidió el usuario (en la página de pedido)
- Implementación del módulo de pagos.

## Apartado Prototipo WEB y boceto estructura.

### Mapa web



### Boceto



## Apartado Guía de estilos

El diseño de la página le puso uno normal y corriente la idea no es que sea lo más llamativo que no estas vendiendo nada si no dando una serie de servicios además siendo donde se tiene que prestar dicho servicios es una cosa muy formal opte por un negro y blanco es lo más clásico y formal para una página de esta categoría y en medio enseñando a los cementerios que prestan esos servicios de Melilla. También soporta formato móviles ya que como he nombrado antes está hecho para las personas que no tiene tiempo pueden conectarse a sin en cualquier momento y donde sea sin depender concretamente de un ordenador (se adapta a cualquier dispositivo gracias a lo estilo Bootstrap en caso de que se habrá en un móvil el encabezado automáticamente le saldrá un botón a la derecha con un submenú para poder acceder fácilmente y queda mucho más visual al iniciar sesión o registrarnos. Para que se aprecia que cementerios hay se le puso una diapositiva en medio div he podido realizar y con un javascript llamando la clase de ese de ese dividir la transición por segundo y que reinicie. A bajo un pie de página con etiquetas semánticas y bootstrap que tiene la información del empresario, el logo de la empresa, contacto de la empresa y términos políticos.

Los estilos con los que están funcionando son los container para el tablón de abajo que dice cómo hacemos los servicios de la empresa y el de medio pero tiene también añadido con la clase myCarousel que pasa una serie de diapositivas y descripciones de las imágenes mediante una función de javascript y por ultimo tiene puesto etiquetas semánticas con estilos de clases de bootstrap tipo para el encabezado de barra de navegación es decir poniendo esto ya cambia al estilo que elegí de encabezado `<nav navbar navbar-light navbar-expand-lg navbar-dark></nav>` con etiquetado hacia arriba header y por último el pie de página que le puse el etiquetado footer y lo dividí los datos con estilo bootstrap `col-xs-12 col-md-3` y etiquetado con div este dividí anda metido también en otro div con clase row.

## Apartado Planificación de tareas

Horas invertidas en total: 245 horas a próximamente de programación e investigación

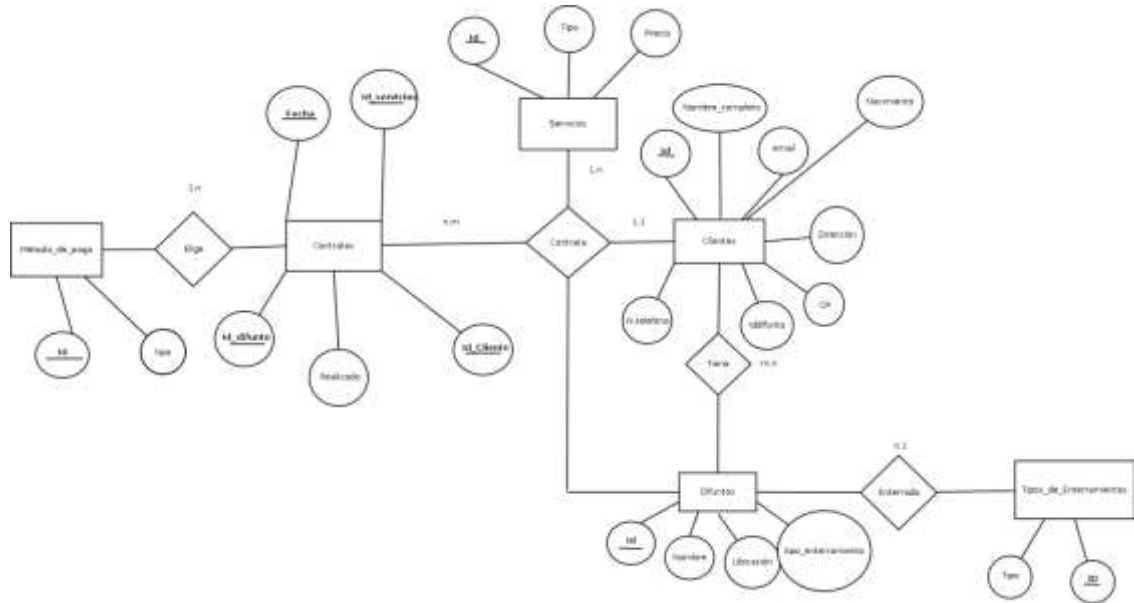
Días empleados en total: 51 (pero horas aplicadas de cada día 5)

Tarea	Días
Elegir el tema del proyecto.	4
Investigación sobre el tema del proyecto.	7
Tutoriales youtube sobre creación de requisitos funcionales para una página web.	4
Diseño de la base de datos.	1
Diseño de la entidad-relación.	1
Tutorial github.	1
Creación de cuenta en github y creación de proyecto.	1
Creación de logotipo.	1
Creación de banner.	1
Corrección de base de datos.	8
Tutorial sobre tabla intermedia en base de datos.	1
Boceto de la página web.	1
Creación de index.html,login.html y register.html.	2
Creación de bd.php,dashboard.php,registrar.php,eliminar.php,grabar.php, Iniciarsesion.php,logout.phpypedidos.php	18

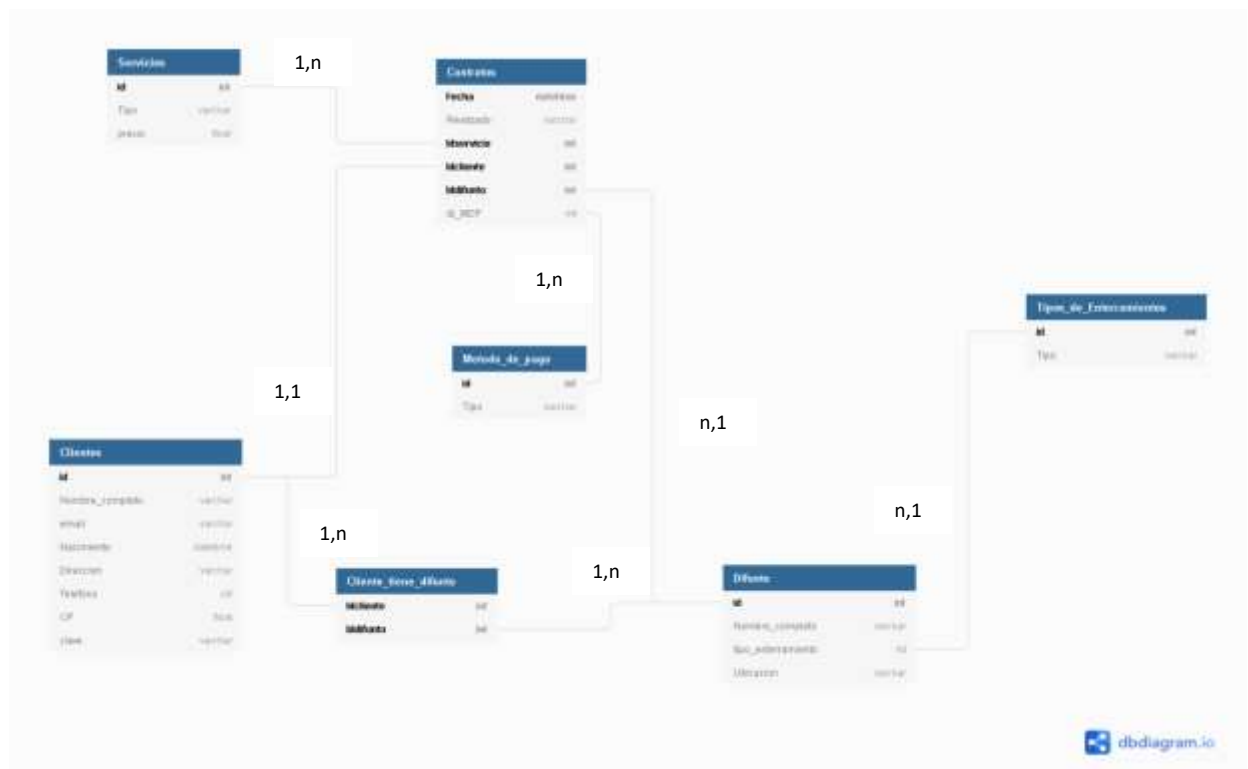


## Diseño de base de datos

- Diseño Entidad Relación de la BBDD



- Modelo relacional BBDD



- **Script de creación BBDD**

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
```

```
START TRANSACTION;
```

```
SET time_zone = "+00:00";
```

```
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
```

```
/*!40101 SET NAMES utf8mb4 */;
```

```
--
```

```
-- Base de datos: `ultimas_memorias`
```

```
--
```

```
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `clientes`
```

```
--
```

```
CREATE TABLE `clientes` (
```

```
  `id` int(11) NOT NULL,
```

```
  `Nombre_completo` varchar(255) DEFAULT NULL,
```

```
  `email` varchar(255) DEFAULT NULL,
```

```
  `Nacimiento` datetime DEFAULT NULL,
```

```
  `Direccion` varchar(255) DEFAULT NULL,
```

```
  `Telefono` int(11) DEFAULT NULL,
```

```
`CP` float DEFAULT NULL,  
`clave` varchar(35) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-----

--

-- Estructura de tabla para la tabla `cliente\_tiene\_difunto`

--

```
CREATE TABLE `cliente_tiene_difunto` (  
  `Idcliente` int(11) NOT NULL,  
  `Iddifunto` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-----

--

-- Estructura de tabla para la tabla `contratos`

--

```
CREATE TABLE `contratos` (  
  `Fecha` datetime NOT NULL DEFAULT current_timestamp(),  
  `Realizado` varchar(255) DEFAULT NULL,  
  `Idservicio` int(11) NOT NULL,  
  `Idcliente` int(11) NOT NULL,  
  `Iddifunto` int(11) NOT NULL,
```

```
`Id_MDP` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-----

--

-- Estructura de tabla para la tabla `difunto`

--

```
CREATE TABLE `difunto` (  
  `id` int(11) NOT NULL,  
  `Nombre_completo` varchar(255) DEFAULT NULL,  
  `Ubicacion` varchar(32) DEFAULT NULL,  
  `tipo_enterramiento` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

-----

--

-- Estructura de tabla para la tabla `metodo\_de\_pago`

--

```
CREATE TABLE `metodo_de_pago` (  
  `Id` int(11) NOT NULL,  
  `Tipo` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
  
-- Volcado de datos para la tabla `metodo_de_pago`  
  
--  
  
INSERT INTO `metodo_de_pago` (`Id`, `Tipo`) VALUES  
(1, 'Paypal');
```

-----

```
--  
  
-- Estructura de tabla para la tabla `servicios`  
  
--  
  
CREATE TABLE `servicios` (  
  `id` int(11) NOT NULL,  
  `Tipo` varchar(255) DEFAULT NULL,  
  `precio` float DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
  
-- Volcado de datos para la tabla `servicios`  
  
--  
  
INSERT INTO `servicios` (`id`, `Tipo`, `precio`) VALUES  
(1, 'Arreglo floral 1', 5),  
(2, 'Arreglo floral 2', 6),  
(3, 'Arreglo floral 3', 7),
```

```
(4, 'Limpieza 1', 5),  
(5, 'Limpieza 2', 6),  
(6, 'Limpieza 3', 7),  
(7, 'Rezcos Coran', 5),  
(8, 'Rezcos Torah', 5),  
(9, 'Rezcos Biblia', 5);
```

```
-- -----
```

```
--
```

```
-- Estructura de tabla para la tabla `tipos_de_enterramientos`
```

```
--
```

```
CREATE TABLE `tipos_de_enterramientos` (  
  `id` int(11) NOT NULL,  
  `Tipo` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--
```

```
-- Volcado de datos para la tabla `tipos_de_enterramientos`
```

```
--
```

```
INSERT INTO `tipos_de_enterramientos` (`id`, `Tipo`) VALUES  
(1, 'Enterramiento en tierra'),  
(2, 'Nicho'),  
(3, 'Panteón');
```

--

-- Índices para tablas volcadas

--

--

-- Indices de la tabla `clientes`

--

ALTER TABLE `clientes`

ADD PRIMARY KEY (`id`);

--

-- Indices de la tabla `cliente\_tiene\_difunto`

--

ALTER TABLE `cliente\_tiene\_difunto`

ADD PRIMARY KEY (`Idcliente`,`Iddifunto`),

ADD KEY `Iddifunto` (`Iddifunto`);

--

-- Indices de la tabla `contratos`

--

ALTER TABLE `contratos`

ADD PRIMARY KEY (`Fecha`,`Idservicio`,`Idcliente`,`Iddifunto`),

ADD KEY `Idcliente` (`Idcliente`),

ADD KEY `Iddifunto` (`Iddifunto`),

ADD KEY `Idservicio` (`Idservicio`),

ADD KEY `Id\_MDP` (`Id\_MDP`) USING BTREE;

```
--  
  
-- Indices de la tabla `difunto`  
  
--  
  
ALTER TABLE `difunto`  
  
  ADD PRIMARY KEY (`id`),  
  
  ADD KEY `tipo_enterramiento` (`tipo_enterramiento`);
```

```
--  
  
-- Indices de la tabla `metodo_de_pago`  
  
--  
  
ALTER TABLE `metodo_de_pago`  
  
  ADD PRIMARY KEY (`Id`),  
  
  ADD UNIQUE KEY `Id_2` (`Id`),  
  
  ADD KEY `Id` (`Id`);
```

```
--  
  
-- Indices de la tabla `servicios`  
  
--  
  
ALTER TABLE `servicios`  
  
  ADD PRIMARY KEY (`id`);
```

```
--  
  
-- Indices de la tabla `tipos_de_enterramientos`  
  
--  
  
ALTER TABLE `tipos_de_enterramientos`  
  
  ADD PRIMARY KEY (`id`);
```



```
--  
  
-- AUTO_INCREMENT de las tablas volcadas  
  
--  
  
--  
  
-- AUTO_INCREMENT de la tabla `clientes`  
  
--  
  
ALTER TABLE `clientes`  
  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=12;  
  
--  
  
-- AUTO_INCREMENT de la tabla `difunto`  
  
--  
  
ALTER TABLE `difunto`  
  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=19;  
  
--  
  
-- AUTO_INCREMENT de la tabla `metodo_de_pago`  
  
--  
  
ALTER TABLE `metodo_de_pago`  
  
    MODIFY `Id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;  
  
--  
  
-- AUTO_INCREMENT de la tabla `servicios`  
  
--  
  
ALTER TABLE `servicios`  
  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=10;
```

```

--

-- AUTO_INCREMENT de la tabla `tipos_de_enterramientos`

--

ALTER TABLE `tipos_de_enterramientos`

    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--

-- Restricciones para tablas volcadas

--

--

--

-- Filtros para la tabla `cliente_tiene_difunto`

--

ALTER TABLE `cliente_tiene_difunto`

    ADD CONSTRAINT `cliente_tiene_difunto_ibfk_1` FOREIGN KEY (`Idcliente`) REFERENCES `clientes` (`id`),

    ADD CONSTRAINT `cliente_tiene_difunto_ibfk_2` FOREIGN KEY (`Iddifunto`) REFERENCES `difunto` (`id`);

--

-- Filtros para la tabla `contratos`

--

ALTER TABLE `contratos`

    ADD CONSTRAINT `contratos_ibfk_1` FOREIGN KEY (`Id_MDP`) REFERENCES `metodo_de_pago` (`Id`),

    ADD CONSTRAINT `contratos_ibfk_2` FOREIGN KEY (`Idcliente`) REFERENCES `clientes` (`id`),

    ADD CONSTRAINT `contratos_ibfk_3` FOREIGN KEY (`Iddifunto`) REFERENCES `difunto` (`id`),

    ADD CONSTRAINT `contratos_ibfk_4` FOREIGN KEY (`Idservicio`) REFERENCES `servicios` (`id`);

```

--

-- Filtros para la tabla `difunto`

--

ALTER TABLE `difunto`

ADD CONSTRAINT `difunto\_ibfk\_1` FOREIGN KEY (`tipo\_enterramiento`) REFERENCES  
`tipos\_de\_enterramientos` (`id`);

COMMIT;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

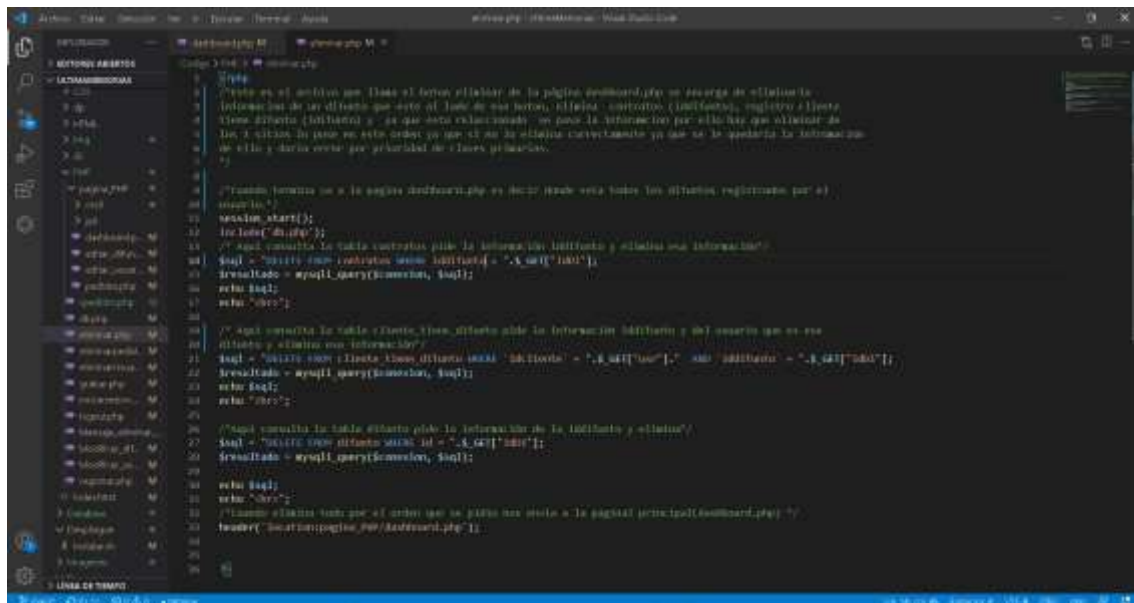
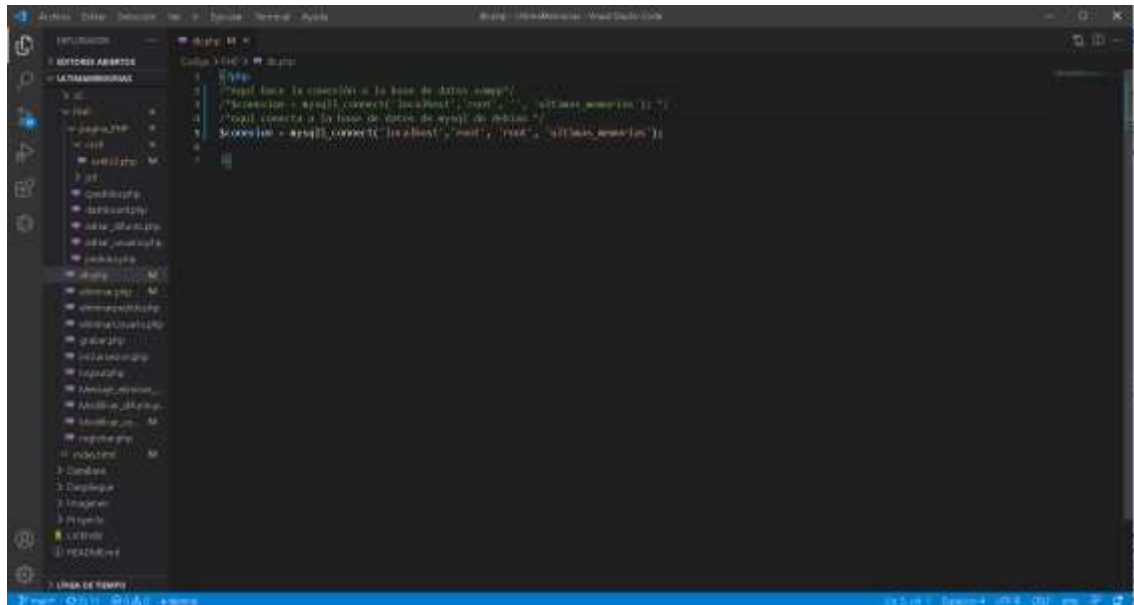
/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

- Consultas

Explico todo y cada una de las consultas y que hacen en los archivos y se puede ver claramente cual es de donde y todo.

## Funciones php:





## Parte2 Eliminar usuario

```

1  // Eliminar usuario
2  // Eliminar usuario
3  // Eliminar usuario
4  // Eliminar usuario
5  // Eliminar usuario
6  // Eliminar usuario
7  // Eliminar usuario
8  // Eliminar usuario
9  // Eliminar usuario
10 // Eliminar usuario
11 // Eliminar usuario
12 // Eliminar usuario
13 // Eliminar usuario
14 // Eliminar usuario
15 // Eliminar usuario
16 // Eliminar usuario
17 // Eliminar usuario
18 // Eliminar usuario
19 // Eliminar usuario
20 // Eliminar usuario
21 // Eliminar usuario
22 // Eliminar usuario
23 // Eliminar usuario
24 // Eliminar usuario
25 // Eliminar usuario
26 // Eliminar usuario
27 // Eliminar usuario
28 // Eliminar usuario
29 // Eliminar usuario
30 // Eliminar usuario
31 // Eliminar usuario
32 // Eliminar usuario
33 // Eliminar usuario
34 // Eliminar usuario
35 // Eliminar usuario
36 // Eliminar usuario
37 // Eliminar usuario
38 // Eliminar usuario
39 // Eliminar usuario
40 // Eliminar usuario
41 // Eliminar usuario
42 // Eliminar usuario
43 // Eliminar usuario
44 // Eliminar usuario
45 // Eliminar usuario
46 // Eliminar usuario
47 // Eliminar usuario
48 // Eliminar usuario
49 // Eliminar usuario
50 // Eliminar usuario
51 // Eliminar usuario
52 // Eliminar usuario
53 // Eliminar usuario
54 // Eliminar usuario
55 // Eliminar usuario
56 // Eliminar usuario
57 // Eliminar usuario
58 // Eliminar usuario
59 // Eliminar usuario
60 // Eliminar usuario
61 // Eliminar usuario
62 // Eliminar usuario
63 // Eliminar usuario
64 // Eliminar usuario
65 // Eliminar usuario
66 // Eliminar usuario
67 // Eliminar usuario
68 // Eliminar usuario
69 // Eliminar usuario
70 // Eliminar usuario
71 // Eliminar usuario
72 // Eliminar usuario
73 // Eliminar usuario
74 // Eliminar usuario
75 // Eliminar usuario
76 // Eliminar usuario
77 // Eliminar usuario
78 // Eliminar usuario
79 // Eliminar usuario
80 // Eliminar usuario
81 // Eliminar usuario
82 // Eliminar usuario
83 // Eliminar usuario
84 // Eliminar usuario
85 // Eliminar usuario
86 // Eliminar usuario
87 // Eliminar usuario
88 // Eliminar usuario
89 // Eliminar usuario
90 // Eliminar usuario
91 // Eliminar usuario
92 // Eliminar usuario
93 // Eliminar usuario
94 // Eliminar usuario
95 // Eliminar usuario
96 // Eliminar usuario
97 // Eliminar usuario
98 // Eliminar usuario
99 // Eliminar usuario
100 // Eliminar usuario

```

Primera parte Grabar(es muy largo el código de consulta entonces dejare la segunda consulta debajo)

```

1  // Grabar usuario
2  // Grabar usuario
3  // Grabar usuario
4  // Grabar usuario
5  // Grabar usuario
6  // Grabar usuario
7  // Grabar usuario
8  // Grabar usuario
9  // Grabar usuario
10 // Grabar usuario
11 // Grabar usuario
12 // Grabar usuario
13 // Grabar usuario
14 // Grabar usuario
15 // Grabar usuario
16 // Grabar usuario
17 // Grabar usuario
18 // Grabar usuario
19 // Grabar usuario
20 // Grabar usuario
21 // Grabar usuario
22 // Grabar usuario
23 // Grabar usuario
24 // Grabar usuario
25 // Grabar usuario
26 // Grabar usuario
27 // Grabar usuario
28 // Grabar usuario
29 // Grabar usuario
30 // Grabar usuario
31 // Grabar usuario
32 // Grabar usuario
33 // Grabar usuario
34 // Grabar usuario
35 // Grabar usuario
36 // Grabar usuario
37 // Grabar usuario
38 // Grabar usuario
39 // Grabar usuario
40 // Grabar usuario
41 // Grabar usuario
42 // Grabar usuario
43 // Grabar usuario
44 // Grabar usuario
45 // Grabar usuario
46 // Grabar usuario
47 // Grabar usuario
48 // Grabar usuario
49 // Grabar usuario
50 // Grabar usuario
51 // Grabar usuario
52 // Grabar usuario
53 // Grabar usuario
54 // Grabar usuario
55 // Grabar usuario
56 // Grabar usuario
57 // Grabar usuario
58 // Grabar usuario
59 // Grabar usuario
60 // Grabar usuario
61 // Grabar usuario
62 // Grabar usuario
63 // Grabar usuario
64 // Grabar usuario
65 // Grabar usuario
66 // Grabar usuario
67 // Grabar usuario
68 // Grabar usuario
69 // Grabar usuario
70 // Grabar usuario
71 // Grabar usuario
72 // Grabar usuario
73 // Grabar usuario
74 // Grabar usuario
75 // Grabar usuario
76 // Grabar usuario
77 // Grabar usuario
78 // Grabar usuario
79 // Grabar usuario
80 // Grabar usuario
81 // Grabar usuario
82 // Grabar usuario
83 // Grabar usuario
84 // Grabar usuario
85 // Grabar usuario
86 // Grabar usuario
87 // Grabar usuario
88 // Grabar usuario
89 // Grabar usuario
90 // Grabar usuario
91 // Grabar usuario
92 // Grabar usuario
93 // Grabar usuario
94 // Grabar usuario
95 // Grabar usuario
96 // Grabar usuario
97 // Grabar usuario
98 // Grabar usuario
99 // Grabar usuario
100 // Grabar usuario

```

## Segunda parte Grabar

```

1 # Seleccionamos los datos de la base de datos de la tabla cliente, clase_difusa y lo vamos a guardar los que voy a escribir los campos del
2 # formulario y devolvemos resultado de la consulta de la base de la petición de consulta.
3 $sql = "SELECT ID, nombre, clase_difusa FROM cliente WHERE ID = '$id'";
4 echo $sql;
5 $resultado = mysql_query($consulta, $sql);
6
7 //Aquí selecciono la tabla cliente, difuso de la columna los datos según se le incluya sesión del cliente en los
8 #para seleccionar y devolvemos resultado de la consulta al crear un objeto se obtiene la información de la tabla seleccionada entre los datos
9 #devueltos al resultado.
10 $sql = "SELECT * FROM cliente WHERE nombre = '$id' AND difuso = '$id'";
11 $resultado = mysql_query($consulta, $sql);
12
13 //Aquí selecciono la tabla difuso de la columna nombre completo y obtengo de la base de datos de la información que se está buscando.
14 #insertar en la base de datos a través de los campos nombre y apellido que son los nombres de los campos de formulario se devolvemos
15 #resultado por la consulta que tiene la consulta del sql a través de la base de datos.
16 $sql = "SELECT * FROM difuso WHERE nombre_completo = '$nombre' AND difuso = '$apellido'";
17 $resultado = mysql_query($consulta, $sql);
18 $count = mysql_num_rows($resultado);
19
20 echo $count;
21
22 //Como voy a resultado si es mayor que cero significa que existen los datos y me los voy a guardar en un resultado si no me devolvemos
23 #error según lo que me da.
24 if($count > 0)
25 {
26     header("location:ingreso.php/actualizar.php");
27 }
28 else
29 {
30     header("location:ingreso.php/actualizar.php/error");
31 }
32
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
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 }

```

[illegible]

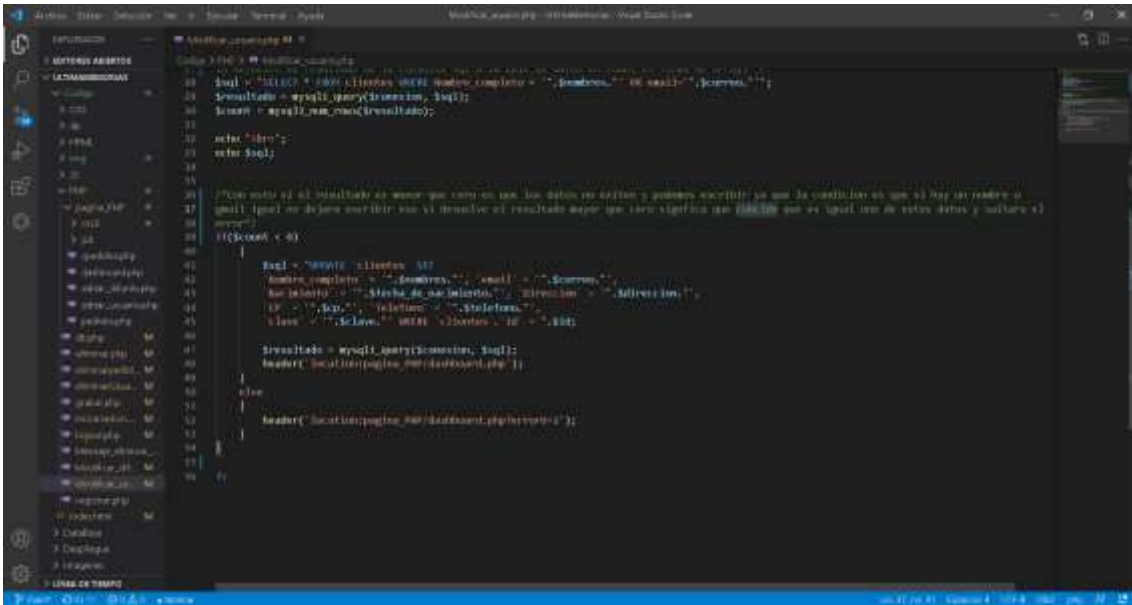
[illegible]

Parte1 modificar usuario(es muy grande el código entonces creo la segunda parte debajo en orden)

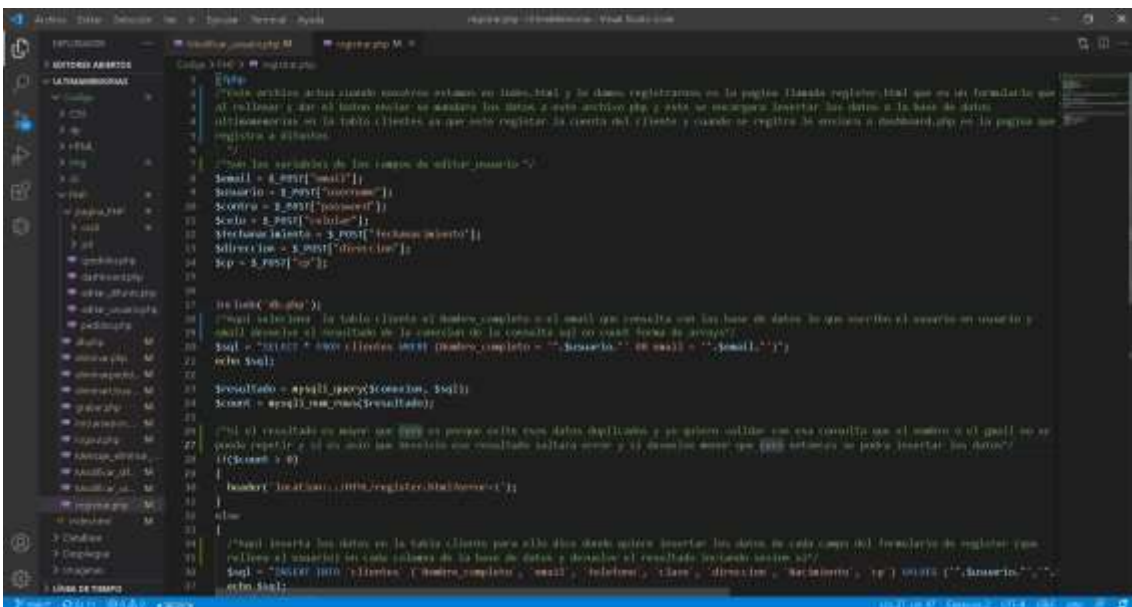
[illegible]



## Parte2 de modificar\_usuario



Parte1 registrar (es muy grande el código entonces creo la segunda parte debajo en orden)



## Parte2 registrar

```

1  <!-- registro.php M. 8 -->
2  <?php
3      // Conexión a la base de datos
4      $conn = mysqli_connect("localhost", "root", "", "bd_clientes");
5      // Verificar la conexión
6      if ($conn === false) {
7          die("Error de conexión: " . mysqli_connect_error());
8      }
9      // Verificar el método de la solicitud
10     if ($_SERVER["REQUEST_METHOD"] == "GET") {
11         // Obtener todos los clientes
12         $sql = "SELECT * FROM clientes";
13         $result = mysqli_query($conn, $sql);
14         if ($result === false) {
15             die("Error de consulta: " . mysqli_error($conn));
16         }
17         // Codificar los datos en JSON
18         $clientes = array();
19         while ($row = mysqli_fetch_assoc($result)) {
20             $clientes[] = $row;
21         }
22         echo json_encode($clientes);
23     } else if ($_SERVER["REQUEST_METHOD"] == "POST") {
24         // Registrar un nuevo cliente
25         $nombre = $_POST['nombre'];
26         $apellido = $_POST['apellido'];
27         $telefono = $_POST['telefono'];
28         $email = $_POST['email'];
29         $direccion = $_POST['direccion'];
30         $ciudad = $_POST['ciudad'];
31         $estado = $_POST['estado'];
32         $pais = $_POST['pais'];
33         // Verificar si el email ya existe
34         $sql = "SELECT * FROM clientes WHERE email = '$email'";
35         $result = mysqli_query($conn, $sql);
36         if ($result === false) {
37             die("Error de consulta: " . mysqli_error($conn));
38         }
39         if (mysqli_num_rows($result) > 0) {
40             echo json_encode(["mensaje" => "El email ya existe."]);
41         } else {
42             // Insertar el nuevo cliente
43             $sql = "INSERT INTO clientes (nombre, apellido, telefono, email, direccion, ciudad, estado, pais) VALUES ('$nombre', '$apellido', '$telefono', '$email', '$direccion', '$ciudad', '$estado', '$pais')";
44             if ($result = mysqli_query($conn, $sql)) {
45                 // Obtener el ID del cliente recién insertado
46                 $id = mysqli_insert_id($conn);
47                 echo json_encode(["mensaje" => "Cliente registrado exitosamente. ID: $id"]);
48             } else {
49                 die("Error de inserción: " . mysqli_error($conn));
50             }
51         }
52     } else {
53         die("Método no permitido");
54     }
55 }

```

```

1 //index.js
2
3 //index.js
4
5 //index.js
6
7 //index.js
8
9 //index.js
10
11 //index.js
12
13 //index.js
14
15 //index.js
16
17 //index.js
18
19 //index.js
20
21 //index.js
22
23 //index.js
24
25 //index.js
26
27 //index.js
28
29 //index.js
30
31 //index.js
32
33 //index.js
34
35 //index.js
36
37 //index.js
38
39 //index.js
40
41 //index.js
42
43 //index.js
44
45 //index.js
46
47 //index.js
48
49 //index.js
50
51 //index.js
52
53 //index.js
54
55 //index.js
56
57 //index.js
58
59 //index.js
60
61 //index.js
62
63 //index.js
64
65 //index.js
66
67 //index.js
68
69 //index.js
70
71 //index.js
72
73 //index.js
74
75 //index.js
76
77 //index.js
78
79 //index.js
80
81 //index.js
82
83 //index.js
84
85 //index.js
86
87 //index.js
88
89 //index.js
90
91 //index.js
92
93 //index.js
94
95 //index.js
96
97 //index.js
98
99 //index.js
100
101 //index.js
102
103 //index.js
104
105 //index.js
106
107 //index.js
108
109 //index.js
110
111 //index.js
112
113 //index.js
114
115 //index.js
116
117 //index.js
118
119 //index.js
120
121 //index.js
122
123 //index.js
124
125 //index.js
126
127 //index.js
128
129 //index.js
130
131 //index.js
132
133 //index.js
134
135 //index.js
136
137 //index.js
138
139 //index.js
140
141 //index.js
142
143 //index.js
144
145 //index.js
146
147 //index.js
148
149 //index.js
150
151 //index.js
152
153 //index.js
154
155 //index.js
156
157 //index.js
158
159 //index.js
160
161 //index.js
162
163 //index.js
164
165 //index.js
166
167 //index.js
168
169 //index.js
170
171 //index.js
172
173 //index.js
174
175 //index.js
176
177 //index.js
178
179 //index.js
180
181 //index.js
182
183 //index.js
184
185 //index.js
186
187 //index.js
188
189 //index.js
190
191 //index.js
192
193 //index.js
194
195 //index.js
196
197 //index.js
198
199 //index.js
200
201 //index.js
202
203 //index.js
204
205 //index.js
206
207 //index.js
208
209 //index.js
210
211 //index.js
212
213 //index.js
214
215 //index.js
216
217 //index.js
218
219 //index.js
220
221 //index.js
222
223 //index.js
224
225 //index.js
226
227 //index.js
228
229 //index.js
230
231 //index.js
232
233 //index.js
234
235 //index.js
236
237 //index.js
238
239 //index.js
240
241 //index.js
242
243 //index.js
244
245 //index.js
246
247 //index.js
248
249 //index.js
250
251 //index.js
252
253 //index.js
254
255 //index.js
256
257 //index.js
258
259 //index.js
260
261 //index.js
262
263 //index.js
264
265 //index.js
266
267 //index.js
268
269 //index.js
270
271 //index.js
272
273 //index.js
274
275 //index.js
276
277 //index.js
278
279 //index.js
280
281 //index.js
282
283 //index.js
284
285 //index.js
286
287 //index.js
288
289 //index.js
290
291 //index.js
292
293 //index.js
294
295 //index.js
296
297 //index.js
298
299 //index.js
300
301 //index.js
302
303 //index.js
304
305 //index.js
306
307 //index.js
308
309 //index.js
310
311 //index.js
312
313 //index.js
314
315 //index.js
316
317 //index.js
318
319 //index.js
320
321 //index.js
322
323 //index.js
324
325 //index.js
326
327 //index.js
328
329 //index.js
330
331 //index.js
332
333 //index.js
334
335 //index.js
336
337 //index.js
338
339 //index.js
340
341 //index.js
342
343 //index.js
344
345 //index.js
346
347 //index.js
348
349 //index.js
350
351 //index.js
352
353 //index.js
354
355 //index.js
356
357 //index.js
358
359 //index.js
360
361 //index.js
362
363 //index.js
364
365 //index.js
366
367 //index.js
368
369 //index.js
370
371 //index.js
372
373 //index.js
374
375 //index.js
376
377 //index.js
378
379 //index.js
380
381 //index.js
382
383 //index.js
384
385 //index.js
386
387 //index.js
388
389 //index.js
390
391 //index.js
392
393 //index.js
394
395 //index.js
396
397 //index.js
398
399 //index.js
400
401 //index.js
402
403 //index.js
404
405 //index.js
406
407 //index.js
408
409 //index.js
410
411 //index.js
412
413 //index.js
414
415 //index.js
416
417 //index.js
418
419 //index.js
420
421 //index.js
422
423 //index.js
424
425 //index.js
426
427 //index.js
428
429 //index.js
430
431 //index.js
432
433 //index.js
434
435 //index.js
436
437 //index.js
438
439 //index.js
440
441 //index.js
442
443 //index.js
444
445 //index.js
446
447 //index.js
448
449 //index.js
450
451 //index.js
452
453 //index.js
454
455 //index.js
456
457 //index.js
458
459 //index.js
460
461 //index.js
462
463 //index.js
464
465 //index.js
466
467 //index.js
468
469 //index.js
470
471 //index.js
472
473 //index.js
474
475 //index.js
476
477 //index.js
478
479 //index.js
480
481 //index.js
482
483 //index.js
484
485 //index.js
486
487 //index.js
488
489 //index.js
490
491 //index.js
492
493 //index.js
494
495 //index.js
496
497 //index.js
498
499 //index.js
500
501 //index.js
502
503 //index.js
504
505 //index.js
506
507 //index.js
508
509 //index.js
510
511 //index.js
512
513 //index.js
514
515 //index.js
516
517 //index.js
518
519 //index.js
520
521 //index.js
522
523 //index.js
524
525 //index.js
526
527 //index.js
528
529 //index.js
530
531 //index.js
532
533 //index.js
534
535 //index.js
536
537 //index.js
538
539 //index.js
540
541 //index.js
542
543 //index.js
544
545 //index.js
546
547 //index.js
548
549 //index.js
550
551 //index.js
552
553 //index.js
554
555 //index.js
556
557 //index.js
558
559 //index.js
560
561 //index.js
562
563 //index.js
564
565 //index.js
566
567 //index.js
568
569 //index.js
570
571 //index.js
572
573 //index.js
574
575 //index.js
576
577 //index.js
578
579 //index.js
580
581 //index.js
582
583 //index.js
584
585 //index.js
586
587 //index.js
588
589 //index.js
590
591 //index.js
592
593 //index.js
594
595 //index.js
596
597 //index.js
598
599 //index.js
600
601 //index.js
602
603 //index.js
604
605 //index.js
606
607 //index.js
608
609 //index.js
610
611 //index.js
612
613 //index.js
614
615 //index.js
616
617 //index.js
618
619 //index.js
620
621 //index.js
622
623 //index.js
624
625 //index.js
626
627 //index.js
628
629 //index.js
630
631 //index.js
632
633 //index.js
634
635 //index.js
636
637 //index.js
638
639 //index.js
640
641 //index.js
642
643 //index.js
644
645 //index.js
646
647 //index.js
648
649 //index.js
650
651 //index.js
652
653 //index.js
654
655 //index.js
656
657 //index.js
658
659 //index.js
660
661 //index.js
662
663 //index.js
664
665 //index.js
666
667 //index.js
668
669 //index.js
670
671 //index.js
672
673 //index.js
674
675 //index.js
676
677 //index.js
678
679 //index.js
680
681 //index.js
682
683 //index.js
684
685 //index.js
686
687 //index.js
688
689 //index.js
690
691 //index.js
692
693 //index.js
694
695 //index.js
696
697 //index.js
698
699 //index.js
700
701 //index.js
702
703 //index.js
704
705 //index.js
706
707 //index.js
708
709 //index.js
710
711 //index.js
712
713 //index.js
714
715 //index.js
716
717 //index.js
718
719 //index.js
720
721 //index.js
722
723 //index.js
724
725 //index.js
726
727 //index.js
728
729 //index.js
730
731 //index.js
732
733 //index.js
734
735 //index.js
736
737 //index.js
738
739 //index.js
740
741 //index.js
742
743 //index.js
744
745 //index.js
7
```

Paginas\_php:

Parte1 editar\_usuario (es muy grande el código entonces creo la segunda parte debajo en orden)

```

1 package com.sistemas.modelo;
2
3 import java.io.Serializable;
4 import java.util.Date;
5
6 public class Usuario implements Serializable, Comparable<Usuario> {
7     private static final long serialVersionUID = 1L;
8     private int id;
9     private String nombre;
10    private String apellido;
11    private String email;
12    private String password;
13    private String telefono;
14    private String direccion;
15    private Date fecha_nacimiento;
16    private Date fecha_registro;
17    private boolean estado;
18    private Date fecha_expiracion;
19
20    // Constructor
21    public Usuario() {
22    }
23
24    // Constructor con parámetros
25    public Usuario(int id, String nombre, String apellido, String email, String password, String telefono, String direccion, Date fecha_nacimiento, Date fecha_registro, boolean estado, Date fecha_expiracion) {
26        this.id = id;
27        this.nombre = nombre;
28        this.apellido = apellido;
29        this.email = email;
30        this.password = password;
31        this.telefono = telefono;
32        this.direccion = direccion;
33        this.fecha_nacimiento = fecha_nacimiento;
34        this.fecha_registro = fecha_registro;
35        this.estado = estado;
36        this.fecha_expiracion = fecha_expiracion;
37    }
38
39    // Getters y Setters
40    public int getId() {
41        return id;
42    }
43    public void setId(int id) {
44        this.id = id;
45    }
46    public String getNombre() {
47        return nombre;
48    }
49    public void setNombre(String nombre) {
50        this.nombre = nombre;
51    }
52    public String getApellido() {
53        return apellido;
54    }
55    public void setApellido(String apellido) {
56        this.apellido = apellido;
57    }
58    public String getEmail() {
59        return email;
60    }
61    public void setEmail(String email) {
62        this.email = email;
63    }
64    public String getPassword() {
65        return password;
66    }
67    public void setPassword(String password) {
68        this.password = password;
69    }
70    public String getTelefono() {
71        return telefono;
72    }
73    public void setTelefono(String telefono) {
74        this.telefono = telefono;
75    }
76    public String getDireccion() {
77        return direccion;
78    }
79    public void setDireccion(String direccion) {
80        this.direccion = direccion;
81    }
82    public Date getFechaNacimiento() {
83        return fecha_nacimiento;
84    }
85    public void setFechaNacimiento(Date fecha_nacimiento) {
86        this.fecha_nacimiento = fecha_nacimiento;
87    }
88    public Date getFechaRegistro() {
89        return fecha_registro;
90    }
91    public void setFechaRegistro(Date fecha_registro) {
92        this.fecha_registro = fecha_registro;
93    }
94    public boolean getEstado() {
95        return estado;
96    }
97    public void setEstado(boolean estado) {
98        this.estado = estado;
99    }
100   public Date getFechaExpiracion() {
101       return fecha_expiracion;
102   }
103   public void setFechaExpiracion(Date fecha_expiracion) {
104       this.fecha_expiracion = fecha_expiracion;
105   }
106
107   // Métodos de validación
108   public boolean validarEmail() {
109       return email != null && email.matches("^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,}$");
110   }
111   public boolean validarPassword() {
112       return password != null && password.matches("^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!@#$%^&*~`-+=~(){}\\[\\]\\\\;:'\".,< /pre>

```

## Parte2 editar\_usuario

```

$ node index.js
? nombre: M
? contraseña: M
? fecha: M
? hora: M
? direccion: M
? telefono: M
? genero: M
? profesion: M
$

```

The terminal window shows the execution of a Node.js script named 'index.js'. The script uses the 'readline' module to interact with the user. It prompts for a name, a password, a birth date, a birth time, a direction, a telephone number, a gender, and a profession. The user's input is displayed in the terminal output.

```

138 <div class="col-md-12">
139     <div class="card">
140         <div class="card-header">
141             <h3>Dashboard</h3>
142         </div>
143         <div class="card-body">
144             <div class="row">
145                 <div class="col-md-4">
146                     <div class="card">
147                         <div class="card-header">
148                             <h4>Usuarios</h4>
149                         </div>
150                         <div class="card-body">
151                             <table>
152                                 <thead>
153                                     <tr>
154                                         <th>ID</th>
155                                         <th>Nombre</th>
156                                         <th>Apellido</th>
157                                         <th>Email</th>
158                                         <th>Telefono</th>
159                                         <th>Fecha de nacimiento</th>
160                                         <th>Sexo</th>
161                                         <th>Estado</th>
162                                         <th>Acciones</th>
163                                     </tr>
164                                 </thead>
165                                 <tbody>
166                                     <tr>
167                                         <td>1</td>
168                                         <td>Juan</td>
169                                         <td>Perez</td>
170                                         <td>juan.perez@ejemplo.com</td>
171                                         <td>3456789012</td>
172                                         <td>1990-01-01</td>
173                                         <td>M</td>
174                                         <td>Activo</td>
175                                         <td><button class="btn btn-primary">Editar</button> <button class="btn btn-danger">Eliminar</button></td>
176                                     </tr>
177                                 </tbody>
178                             </table>
179                         </div>
180                     </div>
181                 </div>
182                 <div class="col-md-4">
183                     <div class="card">
184                         <div class="card-header">
185                             <h4>Productos</h4>
186                         </div>
187                         <div class="card-body">
188                             <table>
189                                 <thead>
190                                     <tr>
191                                         <th>ID</th>
192                                         <th>Nombre</th>
193                                         <th>Precio</th>
194                                         <th>Stock</th>
195                                         <th>Acciones</th>
196                                     </tr>
197                                 </thead>
198                                 <tbody>
199                                     <tr>
200                                         <td>1</td>
201                                         <td>Producto A</td>
202                                         <td>$100</td>
203                                         <td>10</td>
204                                         <td><button class="btn btn-primary">Editar</button> <button class="btn btn-danger">Eliminar</button></td>
205                                     </tr>
206                                 </tbody>
207                             </table>
208                         </div>
209                     </div>
210                 </div>
211                 <div class="col-md-4">
212                     <div class="card">
213                         <div class="card-header">
214                             <h4>Ventas</h4>
215                         </div>
216                         <div class="card-body">
217                             <table>
218                                 <thead>
219                                     <tr>
220                                         <th>ID</th>
221                                         <th>Fecha</th>
222                                         <th>Monto</th>
223                                         <th>Acciones</th>
224                                     </tr>
225                                 </thead>
226                                 <tbody>
227                                     <tr>
228                                         <td>1</td>
229                                         <td>2023-10-26</td>
230                                         <td>$500</td>
231                                         <td><button class="btn btn-primary">Editar</button> <button class="btn btn-danger">Eliminar</button></td>
232                                     </tr>
233                                 </tbody>
234                             </table>
235                         </div>
236                     </div>
237                 </div>
238             </div>
239         </div>
240     </div>
241 </div>

```

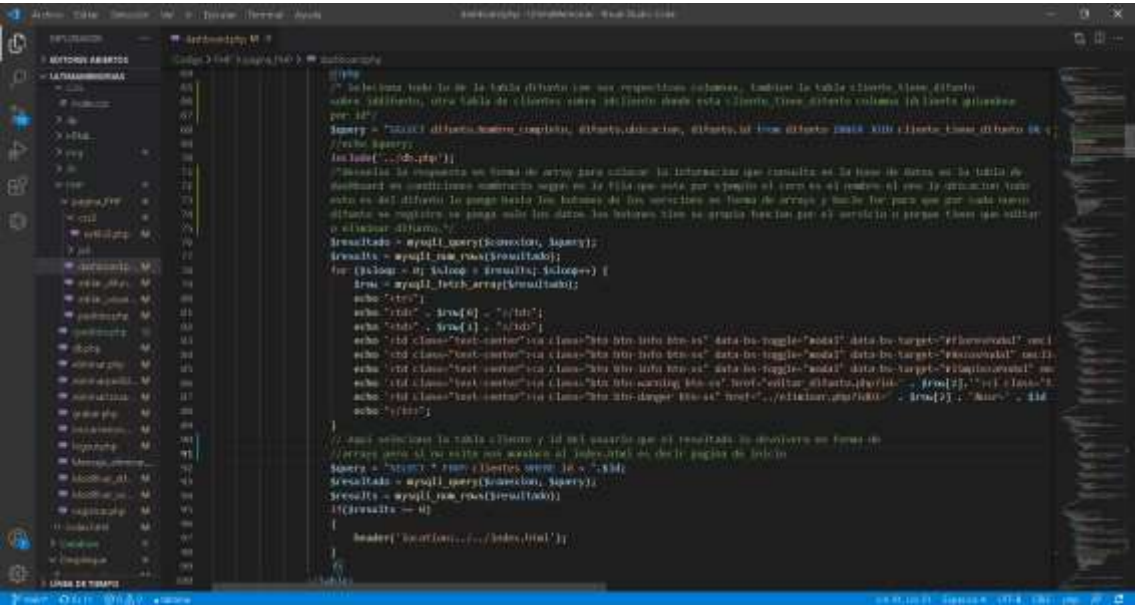
Parte1 dashboard (es muy grande el código entonces creo la segunda parte debajo en orden)

```

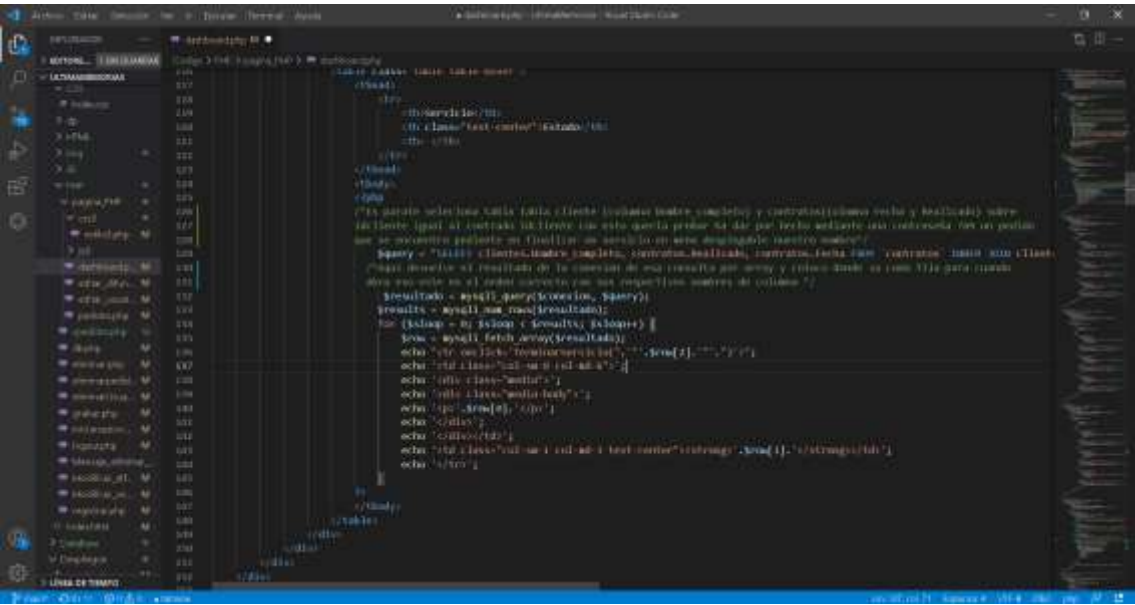
1 <!-- Header -->
2 <div class="header">
3     <div class="container">
4         <div class="row">
5             <div class="col-md-12">
6                 <div class="card">
7                     <div class="card-header">
8                         <h3>Dashboard</h3>
9                     </div>
10                    <div class="card-body">
11                        <div class="row">
12                            <div class="col-md-4">
13                                <div class="card">
14                                    <div class="card-header">
15                                        <h4>Usuarios</h4>
16                                    </div>
17                                    <div class="card-body">
18                                        <table>
19                                            <thead>
20                                                <tr>
21                                                    <th>ID</th>
22                                                    <th>Nombre</th>
23                                                    <th>Apellido</th>
24                                                    <th>Email</th>
25                                                    <th>Telefono</th>
26                                                    <th>Fecha de nacimiento</th>
27                                                    <th>Sexo</th>
28                                                    <th>Estado</th>
29                                                    <th>Acciones</th>
30                                                </tr>
31                                            </thead>
32                                            <tbody>
33                                                <tr>
34                                                    <td>1</td>
35                                                    <td>Juan</td>
36                                                    <td>Perez</td>
37                                                    <td>juan.perez@ejemplo.com</td>
38                                                    <td>3456789012</td>
39                                                    <td>1990-01-01</td>
40                                                    <td>M</td>
41                                                    <td>Activo</td>
42                                                    <td><button class="btn btn-primary">Editar</button> <button class="btn btn-danger">Eliminar</button></td>
43                                                </tr>
44                                            </tbody>
45                                        </table>
46                                    </div>
47                                </div>
48                            </div>
49                            <div class="col-md-4">
50                                <div class="card">
51                                    <div class="card-header">
52                                        <h4>Productos</h4>
53                                    </div>
54                                    <div class="card-body">
55                                        <table>
56                                            <thead>
57                                                <tr>
58                                                    <th>ID</th>
59                                                    <th>Nombre</th>
60                                                    <th>Precio</th>
61                                                    <th>Stock</th>
62                                                    <th>Acciones</th>
63                                                </tr>
64                                            </thead>
65                                            <tbody>
66                                                <tr>
67                                                    <td>1</td>
68                                                    <td>Producto A</td>
69                                                    <td>$100</td>
70                                                    <td>10</td>
71                                                    <td><button class="btn btn-primary">Editar</button> <button class="btn btn-danger">Eliminar</button></td>
72                                                </tr>
73                                            </tbody>
74                                        </table>
75                                    </div>
76                                </div>
77                            </div>
78                            <div class="col-md-4">
79                                <div class="card">
80                                    <div class="card-header">
81                                        <h4>Ventas</h4>
82                                    </div>
83                                    <div class="card-body">
84                                        <table>
85                                            <thead>
86                                                <tr>
87                                                    <th>ID</th>
88                                                    <th>Fecha</th>
89                                                    <th>Monto</th>
90                                                    <th>Acciones</th>
91                                                </tr>
92                                            </thead>
93                                            <tbody>
94                                                <tr>
95                                                    <td>1</td>
96                                                    <td>2023-10-26</td>
97                                                    <td>$500</td>
98                                                    <td><button class="btn btn-primary">Editar</button> <button class="btn btn-danger">Eliminar</button></td>
99                                                </tr>
100                                           </tbody>
101                                       </table>
102                                   </div>
103                               </div>
104                           </div>
105                       </div>
106                   </div>
107               </div>
108           </div>
109       </div>

```

Parte2 dashboard

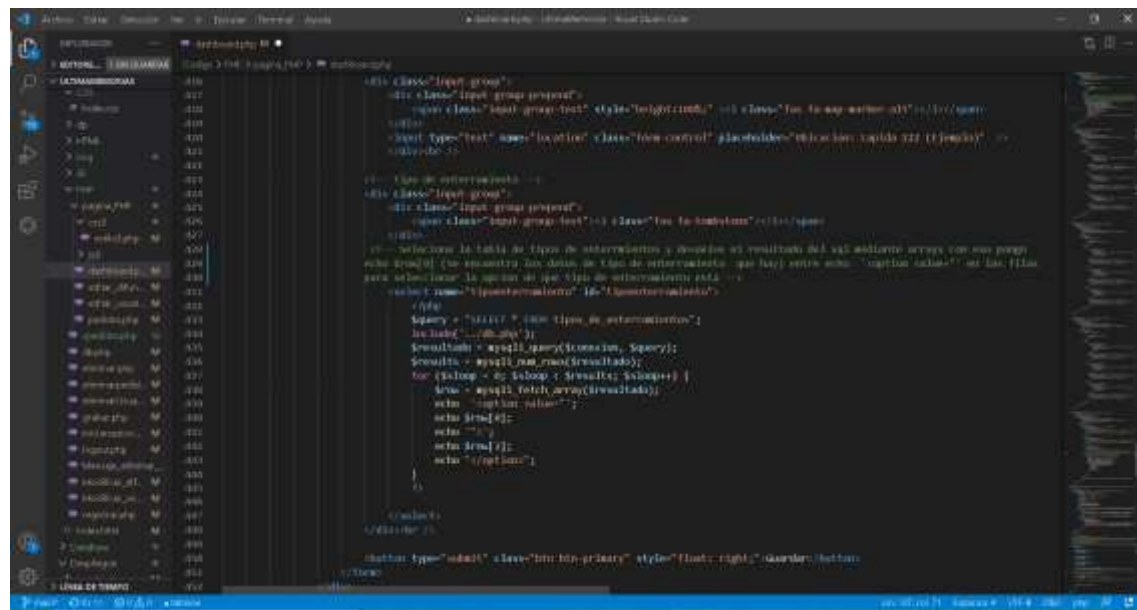


Parte3 dashboard





## Parte3 dashboard



## Validación de formularios

Tenemos cinco formularios en esta página:

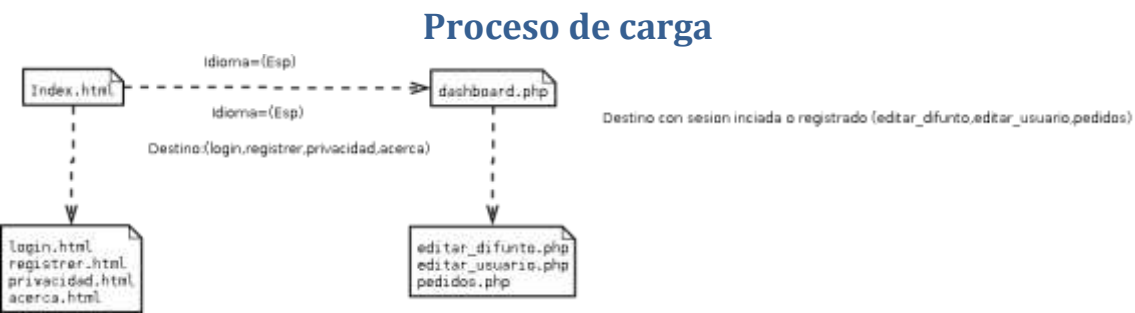
Iniciar sesión-> cuando el usuario le da iniciar con un nombre o contraseña incorrecta le dará un mensaje de error esta parte funciona gracias con esto `getParameterByName` se encuentra en el archivo pone en un archivo js y llama a un parámetro que da un mensaje de alerta que se halla escrito pero claro hay que poner una función de contador (`$count`) claro el contador tiene que ser igual uno significa que si existe hará tal función en mi caso que pueda iniciar y va la página de gestión de difunto del usuario y si no vuelve a la página de iniciar sesión y por ultimo si el usuario se le olvida rellenar el campo los campos tiene dentro este etiquetado para que el campo sea obligatorio `required`.

Registrarse-> cuando el usuario se quiere registrar por primera vez se lo aceptara y se le olvida algún campo se lo pedirá que ese campo es obligatorio de rellenar y si en caso de que ponga un nombre o un gmail que exista le saltara un mensaje de error que están en uso ponga otro para ello utilice `getParameterByName` se encuentra en el archivo pone en un archivo js y llama a un parámetro que da un mensaje de alerta que se halla escrito pero claro hay que poner una función de contador (`$count`) claro el contador tiene que ser igual a 0 significa que si no existe hará tal función es decir registrarme ir a la página de gestión mientras no exista el mismo correo o nombre si existe nos volverá a manda a la página de registro y si se le olvida al usuario rellenar le saltara un mensaje de que requiere rellenar el campo eso se utiliza `required`.

Editar\_usuario-> Cuando un usuario quiere editar sus datos Este formulario lo que hace es cuando el usuario quiere cambiar el nombre por correo hay algo en la base de dato que sea igual a esos dos nombres le sale un error de que ya ese mas el nombre estará en uso está hecho con `getParameterByName` esto funciona de la siguiente manera básicamente llama a un nombre y un parámetro que está eso la función el javascript y saltará con una LED alerta pero claro el nombre y el parámetro se llama con un contador que está en PHP que si básicamente consiste de la siguiente manera el contador en este caso será cero porque básicamente si no están esos datos no existe pues del usuario para modificarlo pero si esos datos no existen el contador está uno entonces significa que salta la alerta nombrada anteriormente decir que si existen ya esos datos no dejar al usuario que lo escriba y puse también que los campos son requerido se decir con `required` por si el usuario se olvida o sin querer deja un campo en blanco.

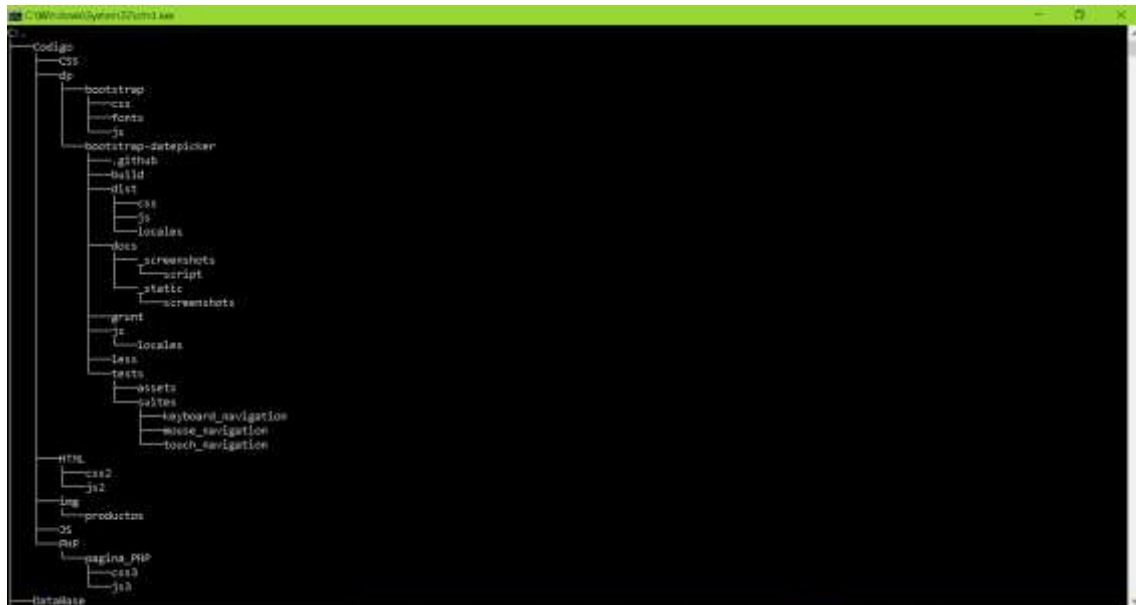
Editar\_difunto-> básicamente es lo mismo método que se utilizó en modificar usuario pero con la diferencia de que los nombres que piden son el nombre y la ubicación porque no van a ser los mismos obviamente no deben coincidir eso dos ya que se puede llamar con el mismo nombre pero no puede tener la misma ubicación también ya que eso es imposible para ello he utilizado lo mismo que el para ello se puso el contador `== 0` si no existe el mismo se aplica como bueno sin mensaje de error y lo envía a la página principal si no le dará error. Y los campos no deben estar vacíos los campos.

Registre nuevo difunto-> Este utiliza el mismo sistema de validación y condiciones que editar difunto.





## Jerarquía de directorios



## Contenido directorios

**BD\_SQL**-> Contiene los archivos de SQL (base de datos) y PDF del cumplimiento de obligatorio del documento del proyecto.

**CSS**-> Contiene los archivos de estilos CSS de Index.HTML

**dp**-> contiene todo el paquete bootstrap de datepicker sirve para poner calendarios es un estilo bootstrap

**html**-> Contiene todos los archivos html y aparte su JavaScript y el CSS de los archivos html y se llaman carpetas que llaman css2(estilo) js2(Javascript).

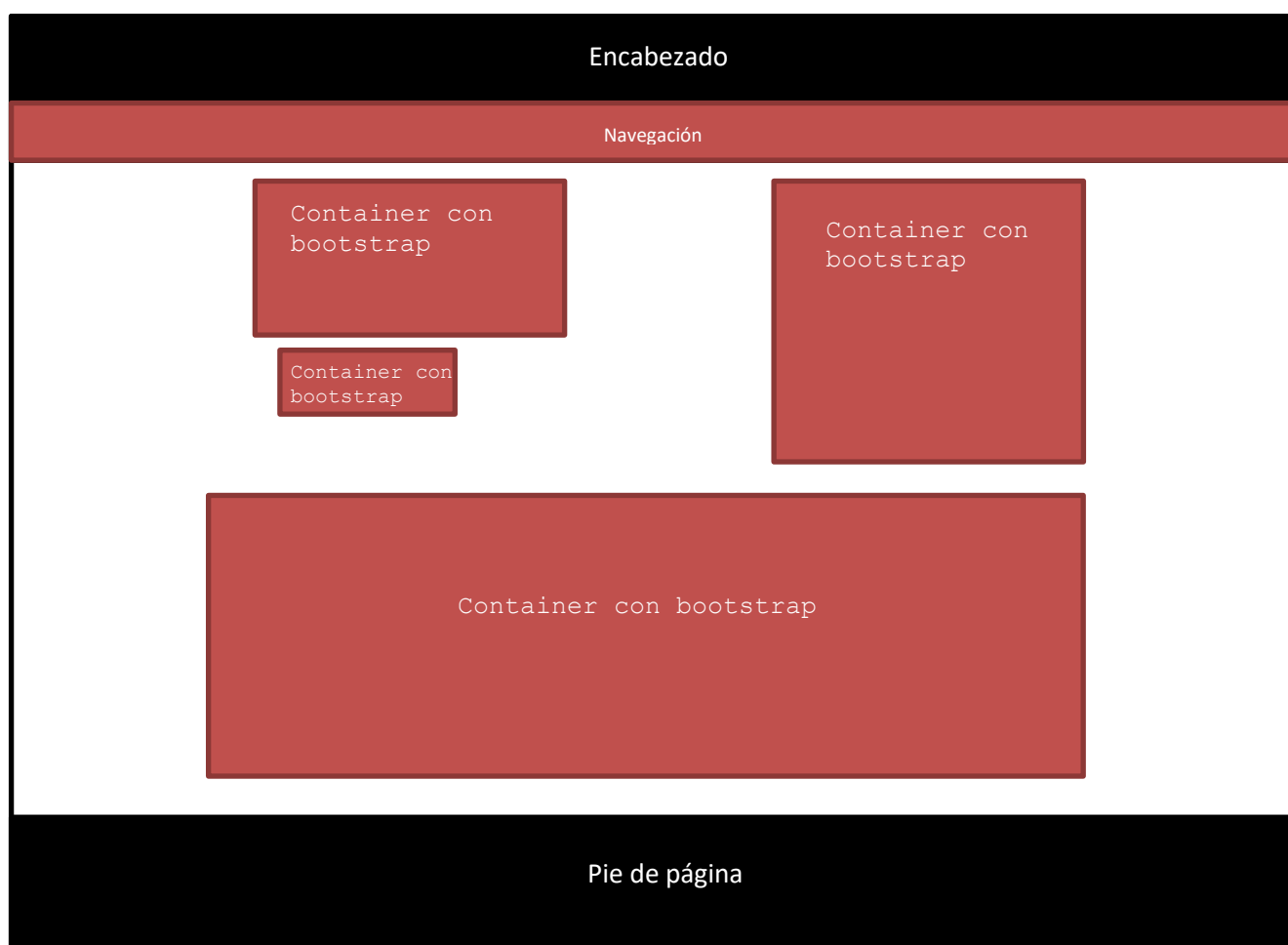
**JS**-> Contiene el JavaScript del Index .HTML

**PHP**-> Contiene todos los archivos de función PHP y dentro de esa carpeta tiene una carpeta llamada pagina\_PHP en esa carpeta están todas las páginas hechas por extensión PHP y dentro tiene la carpeta CSS3 y JS3 que son dos carpetas que contiene el JS3 contiene los JavaScript de las pagina\_PHP que hay ahí y el CSS3 contiene los estilos de la carpeta de las paginas\_PHP

## Apartado Diseño de la interface

El encabezado contiene el menú de navegación en la parte derecha y en la parte izquierda tiene un el logo de la empresa con sus respectivas decoraciones y todo el menú navegación tenemos básicamente iniciar sesión y registrarse luego más para abajo tenemos tres container hecho **por bootstrap básicamente empezare a describir el** container el primero de la izquierda contiene todas las descripción de las imágenes que está **en el container a la derecha y el otro** que tenemos **abajo del primero** básicamente es el botón de registrarse ya abajo tenemos otro container más grande y más extenso está dividido entre dice lo que ofrece la empresa o promete la empresa hacer que es el servicio **más rápido por ejemplo que** lo hace **desde** el móvil sin moverte de casa y por último tenemos el pie de página que nos permitirá contactar con el empresario de diferentes modos con el gmail eso sí está hecho con maíz alto si su extensión apropiada no podrás hacer una CD también se han pensado para los usuarios que no tengan esa extensión poner gema y completo de la empresa también tenemos número de contacto y el logo de la empresa

## Estructura gráfica de la interface



# IMPLEMENTACIÓN

## REQUISITO 1: Diseño responsive

Ya que mi servicio para las personas que no tiene tiempo es más fácil coger un móvil entonces puse ese diseño con bootraps sin apenas tocar css.

### Funcionamiento

Boostraps poniendo sus enlaces correspondiente ya que ponía en la clases el tipo de estilo que se trataba por ejemplo un encabezado lo que conseguí con eso poner todo en un frases todos sus css sin ponerlos y ya cuando era algo muy específico como el menú despegable ese si lo cree mediante css y a su vez bootraps

### Ejemplo de código

```
<header>
```

```
  <nav class="navbar navbar-light navbar-expand-lg navbar-dark" style="background-color: black;">
```

```
    
```

```
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
```

```
      <span class="navbar-toggler-icon"></span>
```

```
    </button>
```

```
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
```

```
    <ul class="navbar-nav mr-auto">
```

```
      <li class="navbar-nav" style="text-align: right;">
```

```
        <a style="color:white;" class="nav-link" href="HTML/login.html?error=0">Iniciar Sesion</a>
```

```
        <a style="color:white;" class="nav-link" href="HTML/register.html?error=0">Registrarse</a>
```

```
      </li>
```

```
    </ul>
```

```
  </div>
```

```
</nav>
```

```
</header>
```

## REQUISITO 2: Para que se pueda acceder a la pagina web

Claro para que el usuario quiera acceder a la pagina para crear nuevos registros de difunto y hacerlos servicios que desee es necesario obligar que se registre el usuario es decir limitamos la a página de inicio y que no pueda hacer nada al menos que se registre o inicié sesión

### Funcionamiento

Para que los datos del difunto que para cada persona es diferente tiene que acceder iniciar sesión o registrarse claro hay puesto en dashboard.php o página principal que no puede acceder sin eso

### Ejemplo de código

```
<?php
    session_start();
    $id = $_SESSION["id"];
    $usuario = $_SESSION["usuario"];
    if ($id == 0 or $usuario == "") {
        header('location:../HTML/login.html');  }

?>
```

## **PRUEBAS**

Importe una mv DebianLAMP.ova que está preparada que tiene instalado un S.O debían que tiene instalado php, apache, ssh y mysql antes de subir nada he hecho varias pruebas de que el apache funcionaba, mysql(entrando con el usuario y contraseña dada), apache (entrando en la página de prueba que tiene desde mi anfitrión) y el php (entrando desde el anfitrión a su página de prueba) y utilice ssh con mi anfitrión conectándome a mv y por ultimo comprobé si la contraseña para entrar root del mv era correcta

### **Metodología de las pruebas**

Para probar que mi proyecto es posible que funcione dentro de estos servicios recree dichos despliegue y conexiones para la base de datos y hice una clonación en git hub para poder hacer mediante comando una clonación en mv en debían que se la hice y comprobé que se descargó donde se indicó los archivos nombrados antes comprobé que funcionaba poniendo la dirección IP 192.168.1.73/Codigo que es la ruta en el que se puede conectar a mi proyecto que esta clonado en mi mv y resulto que salio mi pagina y comprobé que hacia cada uno de los requisitos funcionales nombrados(claro mirando si insertaba y todo en mysql) , las validaciones de formularios que se puso en su debido tiempo y por ultimo si el diseño reponsive funcionaba en otros dispositivos como en un móvil copiando la dirección en chrome de mi móvil funciono.

## DESPLIEGUE

Explico lo que falto por definir en este despliegue y lo que vi correcto lo que se define en los comentarios que ya venían lo deje porque ya lo explica perfectamente no hace falta que añada mas

```
#!/bin/bash
```

```
#
```

```
# SCRIPT DE DESPLIEGUE DE PROYECTO
```

```
#
```

```
# Natalia Otero García
```

```
#
```

```
# Esto es el nombre del usuario y contraseña del mysql de mv conectara hay para dejar la base  
y lo otro es la url de nuestra mv
```

```
USERDB="debianDB"
```

```
PASSDB="debianDB"
```

```
HOST=$(hostname -I)
```

```
WWW="/var/www/html/"
```

```
# Aquí coge el fichero de la base de datos y como se llama la base de datos
```

```
DATOS="ultimas_memorias.sql"
```

```
BBDD="ultimas_memorias"
```

```
# Se toman los parámetros como USER y PASS de la BBDD
```

```
if [ $# = 2 ];
```

```
then
```

```
    USERDB=$1
```

```
    PASSDB=$2
```

```
fi
```

# Copiamos el contenido de la carpeta proyecto a la página html

```
cp -r ../Codigo/ $WWW
```

# Creamos esas base de datos y luego restauramos esa base de datos

```
mysqladmin -u $USERDB -p$USERDB create $BBDD
```

```
mysql -u $USERDB -p$USERDB $BBDD < ../DataBase/$DATOS
```

# Mostramos url de carga

```
echo "http://$HOST/Codigo/Index.php"
```

## **Apartado Herramientas. Por lo menos enumeración**

1. Visual Studio code versión 1.62.2
2. Xampp versión 3.3.0:
3. Chrome versión95.0.4638.69

## **Apartado Lenguajes. Por lo menos enumeración.**

1. PHP
2. Javascript
3. Html
4. css



## PRODUCTO

Muestro aquí todas las páginas que hay en mi página web y describiré brevemente de que archivo pertenece

### Página de Inicio

Este pertenece al archivo index.html



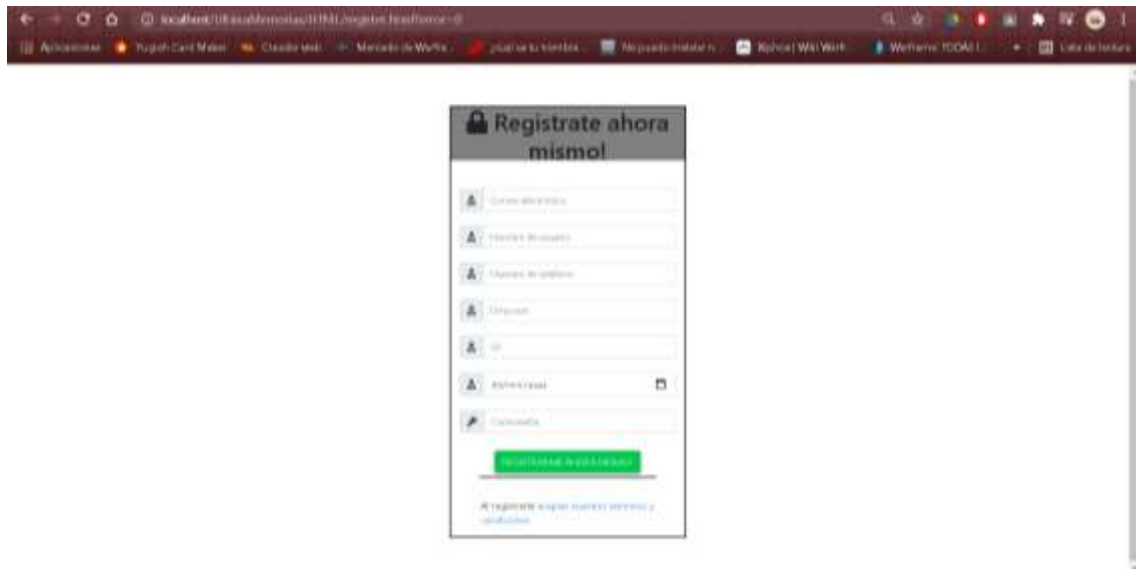
### Página iniciar sesión

Este esta en la carpeta html es el archivo login.html entramos cuando le damos inciar sesión



## Página de registro

La página de registros se accede de dando al registrarse esto está en la carpeta html llamado en un archivo register.HTML



## Página de gestión (tablero)

La página se hace mediante iniciando sesión o registrándose esto se encuentra en la carpeta PHP>pagina\_PHP y se llama el archivo dashboard.php aquí podemos darle al menú que está en nuestro nombre podemos registrar un familiar dándole registro nuevo o podemos configurar nuestra cuenta dándole a configuración, tenemos cerrar sesión y podemos darte bajo a nuestra cuenta que eliminará todo lo que tengamos dentro de nuestra cuenta gestionado y a la cuenta misma. A registrar un nuevo difunto podemos enviarle flores, rezos o limpiar que nos saldrá una ventanita modal que un estilo de bootstrap y le podemos dar el servicio que queremos y pagar y también se puede dar editar que eso lo enseñaré en la siguiente página. Y finalizar servicios es para cuando tengamos un servicio pendiente podemos entrar con contraseña para dar por hecho el servicio en realizado



## Página de editar difunto

Aquí que este archivo esta en PHP/pagina\_PHP se llama editar\_difunto.php cuando quiero editar un difunto le doy editar difunto veo los datos del difunto y los puedo cambiar y guardo los cambios o en caso que no quiera cambiar los datos vuelvo a la pagina dashboard.php



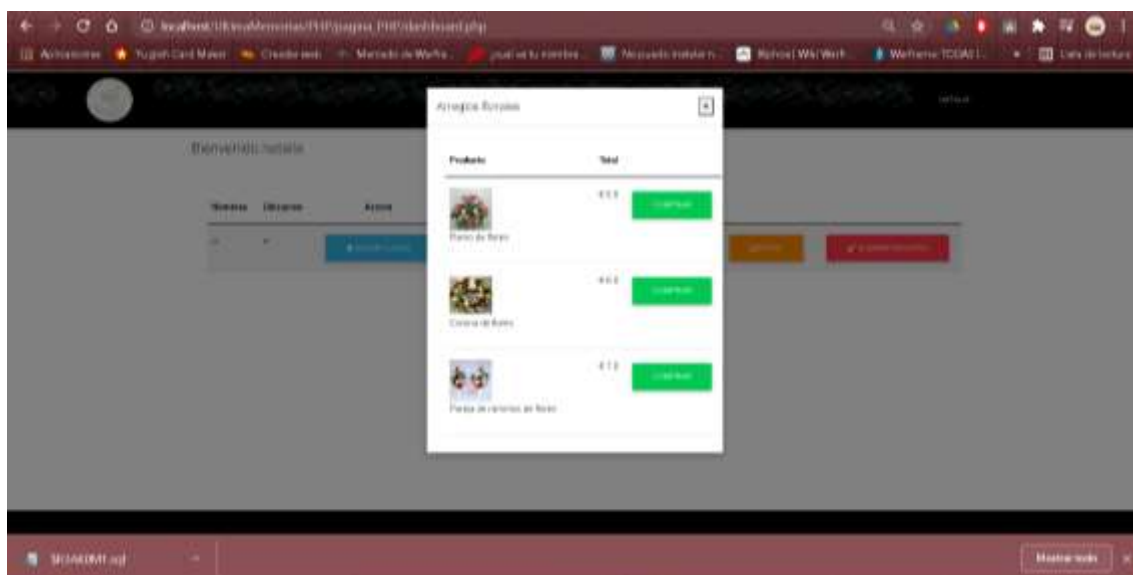
## Página de configuración de perfil

Aquí se entra desde dando nuestro nombre y configuración para entrar a editar un usuario que estemos iniciado para poder cambiar nuestros datos y ver los que pusimos si cambiamos algunos datos le damos guardar cambios y si no queremos cambiar nada le damos volver a la pagina principal es decir a dashboard



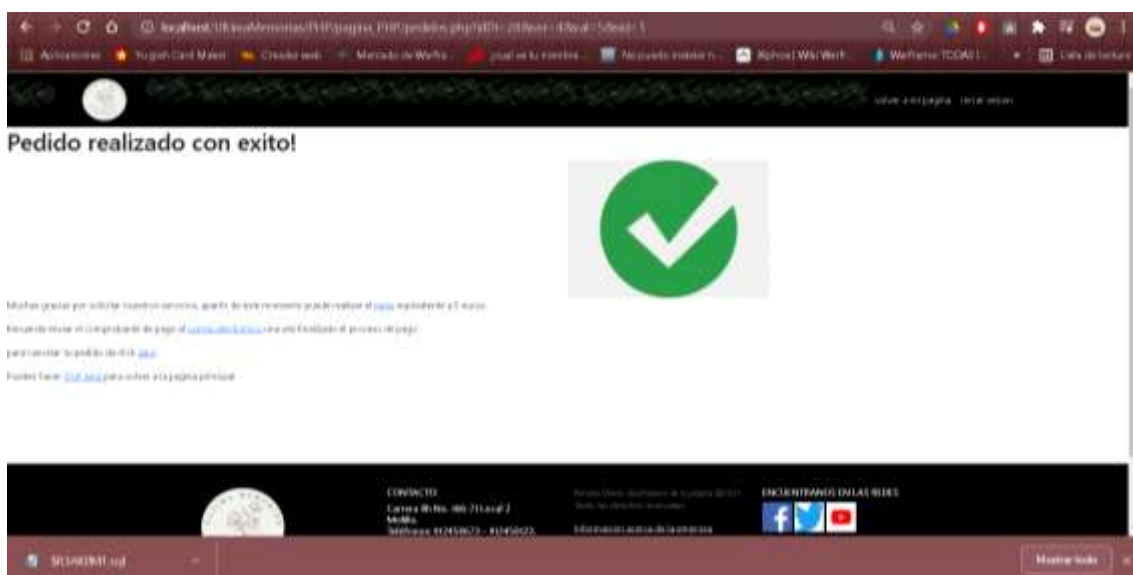
## Abre una ventana emergente de un servicio

Cuando le damos un botón de servicio de flores, limpieza y oraciones nos saldrá esto eligimos que queremos comprar o pagar y luego nos llevara a una página que está en la siguiente imagen de pedios.



## Página pedidos

Cuando le damos algo de la anterior imagen nos lleva a esta parte para realizar el pago o le podemos dar cancelar o volver a otra vez a dashboard.



## **Apartado Bibliografía. Las que se hayan usado hasta el momento.**

<https://www.youtube.com/watch?v=Lv7XbZtnQ6A>

<https://youtu.be/bO18omSzeR4>

[https://youtu.be/L\\_IWQZNhN7w](https://youtu.be/L_IWQZNhN7w)

<https://youtu.be/hWglK8nWh60>

<https://youtu.be/GG4mftkQVrs>

<https://youtu.be/tFr0Vg1q9Eg>

<https://getbootstrap.com/docs/5.0/getting-started/introduction/>

<https://www.w3schools.com/php/>

<https://www.php.net/manual/es/language.functions.php>

<https://getbootstrap.com/>