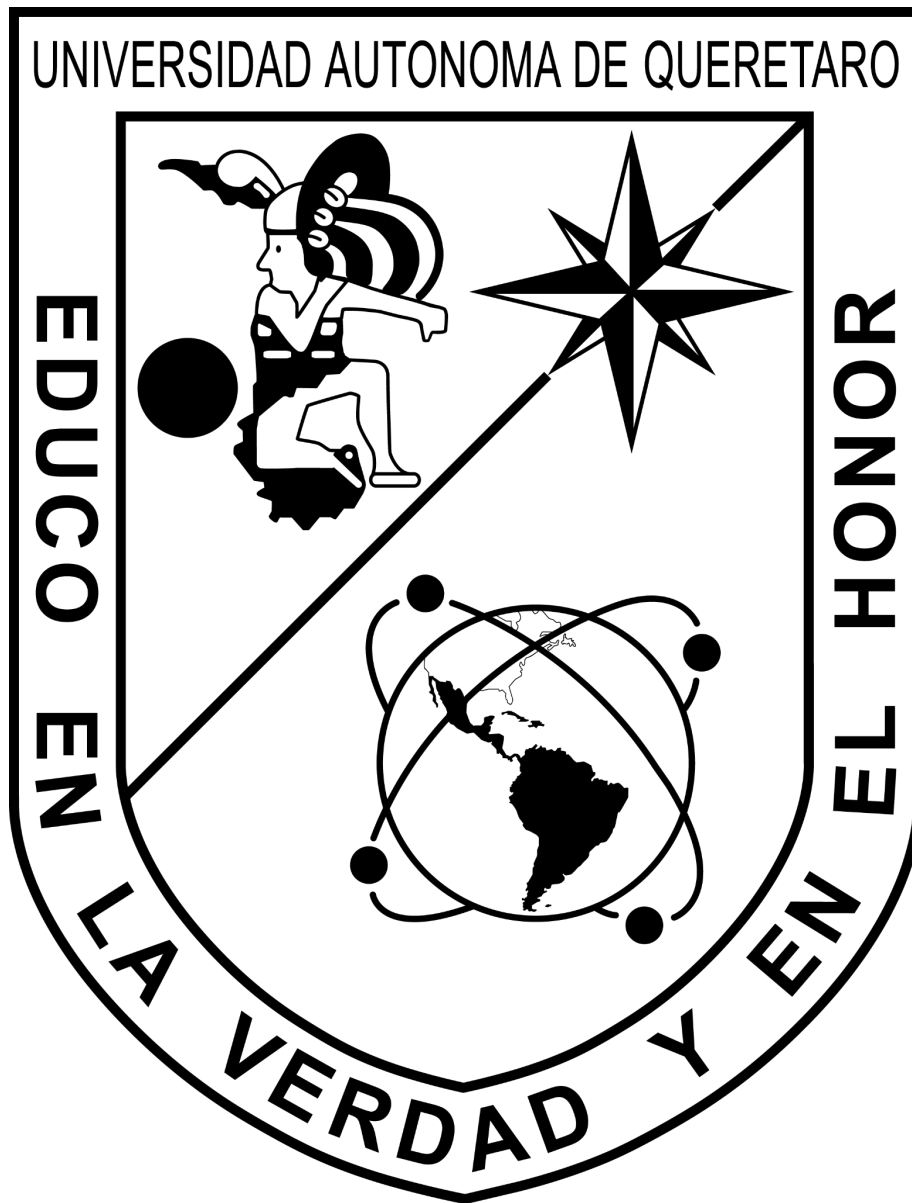


PROYECTO - CONTROL VEHICULAR



INTEGRANTES:

MACAL PALACIO JESÚS SANTIAGO

SALINAS LEDESMA VICTOR ALI

BETAN RAMOS EDGAR

ÍNDICE

INTRODUCCIÓN	3
OBJETIVOS	4
Objetivo 1: Inserción de datos en BD.	4
Objetivo 2: Conexión con BD.	7
Objetivo 3: Eliminación de datos en BD.	8
Objetivo 4: Interfaz visual(Frontend).	9
Objetivo 5: Actualización.	10
Objetivo 6: Implementación de funciones.	13
Objetivo 7: Menús de diferenciación.	15
Objetivo 8: Uso de la librería FPDF.	20
Objetivo 9: Generación de PDF 's.	22
MODELO RELACIONAL DE LA BASE DE DATOS.	29
DIAGRAMA DE COMPONENTES	30
MANUAL DE USUARIO	31
ADMINISTRADOR	34
DELETE	37
UPDATE	42
UPDATE Y DELETE	43
USUARIOS	44
PDFS Y GUARDADO DE ARCHIVOS	45
CONCLUSIONES	47

INTRODUCCIÓN

El proyecto de Control Vehicular consta de utilizar un sistema manejador de base de datos o también conocido como SBMD, en el cual se deberá crear desde cero una base de datos que debe ser funcional al cien por ciento con sus respectivas capacidades de manejo de vehículos/atención variada. Por lo que durante este proyecto se han hecho implementaciones bastante importantes para su desarrollo primero comenzamos creando nuestra base de datos a partir de los documentos que nos proporcionó nuestro profesor al iniciar el semestre para tener en cuenta qué datos serían de bastante relevancia para nuestra implementación en la base de datos, esto nos llevó a diferentes análisis por parte de los compañeros de trabajo para que todo esté correctamente desarrollado. Más que nada lo tenemos que hacer lo más detallado posible para que no tengamos problemas al momento de tener que pasar a partes más avanzadas del proyecto y asegurarnos que una buena parte de la base que es la base de datos funciona como se espera y correctamente. Es así que una vez identificados todos estos parámetros que sí nos serán de completa utilidad empieza la elaboración de la base de datos que usaremos para basar todo este sistema de control vehicular. Una vez que está listo todo se debe crear un código en SQL, para que podamos crear nuestra base de datos, con los tipos de datos más adecuados para manejar correctamente la base de datos. Cuando se crea esto también debemos tener especial cuidado en crear las relaciones que tendrá nuestra base de datos para que funcione correctamente. Una vez que esto está cubierto es esperado que contemos solamente con nueve relaciones ya que son las que tendremos que utilizar para crear un modelo relacional de base de datos correcto.

Cuando esto se haya completado comenzaremos con la creación de archivos que nos sirvan a nosotros como administradores y manipuladores de la base de datos acceso a subir datos a la misma sin estar en la interfaz de subida de datos. Para hacer esto utilizamos el programa XAMPP, que utiliza un servidor local Apache y el motor MySQL, por lo que para no tener que recurrir a utilizar directamente la consola SQL que nos ofrece, creamos una serie de archivos con ayuda del lenguaje de programación HTML y PHP para completar una vista de ingreso de datos básica con HTML y lograr el ingreso de datos a la base de datos con el uso de PHP. Estos archivos utilizan la sentencia de SQL "INSERT" para poder ingresar los datos correctamente.

Durante la implementación del proyecto tenemos en mente hacer algunos cambios generalizados en la parte visual y atractiva de la página web, así mismo lo que nos proponemos algunos cambios leves para que se vea lo más homogéneo posible.

OBJETIVOS

Objetivo 1: Inserción de datos en BD.

Los objetivos de este proyecto son los siguientes:

- Crear una base de datos completamente funcional.
- Crear conexiones backend funcionales.
- Documentar correctamente lo sucedido.
- Mejorar el entendimiento de SQL.

Durante el desarrollo del proyecto algunas de las secciones de objetivos a tratar fueron cumplidas satisfactoriamente, la base de datos fue creada y analizada en caso de cada miembro del equipo y una última analizada para corroborar que todos los datos importantes de la base de datos se encontraban en ella.

Ya que tenemos que generar archivos dinámicos que deberán ser impresos, esto se requiere con la máxima precisión y eficacia al momento de identificar los campos que usaremos para que nuestra base de datos no tenga problemas al momento de utilizarla.

Durante esta implementación es necesario que hagamos uso de archivos para diferentes secciones de la base de datos, siendo que vamos a utilizar diferentes archivos como lo son la Tarjeta de Circulación, Licencias, Multas y Verificaciones. Para los archivos se busca que principalmente estén lo mejor relacionados a la base de datos que estamos manejando. Durante la implementación de la misma utilizando los valores y datos congruentes para el uso del mismo.

Los objetivos que tenemos cumplidos son los siguientes:

- Crear código en HTML y PHP, para la subida de datos a nuestra base de datos.

Primero que nada, estos archivos han sido funcionales desde los inicios del proyecto, son los que nos permiten la subida correcta de los datos a nuestra base de datos para poder tener una mejor visualización de la misma así como otras funcionalidades mejoradas del mismo. El código de HTML que utilizamos es bastante sencillo ya que esta página aún está pendiente en añadir el nuevo estilo que se tiene en mente.

Ejemplo de la tabla Conductores en código HTML.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
```

```
<title>Conductores</title>
</head>
<body>
    <form method="post" action="IConductores.php">
        <label for="IDConductor">Conductores</label>
        <br />

        <label for="IDConductor">ID Conductor</label>
        <input type="number" id="IDConductor" name="IDConductor" />
        <br />

        <label for="Nombre">Nombre</label>
        <input type="text" id="Nombre" name="Nombre" maxlength="50" />
        <br />

        <label for="FechaNacimiento">Fecha de Nacimiento</label>
        <input type="date" id="FechaNacimiento" name="FechaNacimiento"
/>
        <br />

        <label for="Firma">Firma</label>
        <input type="file" id="Firma" name="Firma" />
        <br />

        <label for="Foto">Foto</label>
        <input type="file" id="Foto" name="Foto" maxlength="200" />
        <br />

        <label for="TipoSangre">Tipo de Sangre</label>
        <input type="password" id="TipoSangre" name="TipoSangre"
maxlength="3" />
        <br />

        <label>Donador</label>
        <input type="radio" id="DonadorSi" name="Donador" value="1" />
        <label for="DonadorSi">Sí</label>
        <input type="radio" id="DonadorNo" name="Donador" value="0" />
        <label for="DonadorNo">No</label>
        <br />

        <label for="Licencia">Licencia</label>
        <select name="Licencia" id="Licencia">
            <option value="A">Automovilista</option>
```

```

        <option value="B">Chofer</option>
        <option value="C">Motociclista</option>
    </select>
    <br />

    <input type="submit" value="Enviar" />
</form>
</body>
</html>

```

Así como esta es la interfaz visual que nos permite visualizar qué datos vamos a ingresar a nuestra base de datos necesitamos un archivo PHP que contenga las instrucciones de la sentencia INSERT de SQL.

Ejemplo de Centro Verificaciones.

```

<?php

include("IControlador.php");
$NumeroCentro=$_POST['NumeroCentro'];
$NumerodeLinea=$_POST['NumerodeLinea'];
$Municipio=$_POST['Municipio'];
$Verificacion=$_POST['Verificacion'];

//print("DatosDelAuto= ".$DatosDelAuto."<br>");
//print("NumerodeSerie= ".$NumerodeSerie."<br>");
//print("NumeroVerificacion= ".$NumeroVerificacion."<br>");
//print("Municipio= ".$Municipio."<br>");
//print("Verificacion= ".$Verificacion."<br>");

$SQL="INSERT INTO centrosverificaciones VALUES('$NumeroCentro',
'$NumerodeLinea', '$Municipio', '$Verificacion');";
//print($SQL);

//$SQL="INSERT INTO centrosverificaciones
VALUES('$Verificacion','$DatosDelAuto','$Municipio','$NumerodeSerie','$
NumeroVerificacion');";
//print($SQL);

//Enviar el SMBD
$Con=Conectar();
$ResultSet=Ejecutar($Con,$SQL);

```

```

        if($ResultSet == 1){
            print("1 FILA REGISTRADA");
        }
        else {
            print("ERROR ".mysqli_error($Con));
        }
        Desconectar($Con);
    }
    ?>

```

Objetivo 2: Conexión con BD.

- Crear una forma de conexión a la base de datos.

Una vez que creamos todos nuestros archivos de subida de datos exitosamente para la base de datos el siguiente paso sería crear una forma de poder comunicarnos con la base de datos. De este modo nos aseguramos que exista un archivo con varias funciones que hacen conexiones con la base de datos. Siendo estas la conexión, la función ejecutar, procesar y desconectar.

El código de la conexión es el siguiente, llamado controlador.php

```

<?php
    //Enviar el SMBD
    function Conectar(){
        $Server="127.0.0.1";
        $User="root";
        $Password= "";
        $BD="prueba2";
        $Con=mysqli_connect("$Server", "$User", "$Password",
"$BD");

        return $Con;
    }
    function Ejecutar($Con, $SQL){
        $ResultSet=mysqli_query($Con,$SQL);
        return $ResultSet;
    }

    function Procesar(){

    }

```

```

        function Desconectar($Con) {
            $Valor=mysqli_close($Con);
            return $Valor;
        }
?>

```

Objetivo 3: Eliminación de datos en BD.

- Para esta parte de los objetivos en mente serían utilizar una serie de funciones SQL para poder hacer eliminaciones de la base de datos, siendo estos datos que no estén funcionando correctamente o hayan sido mal ingresados.

Durante esta etapa del proyecto se hizo uso de unos nuevos archivos PHP, que serían los encargados de hacer y manejar las eliminaciones de los datos que fueron ingresados por los usuarios. Generalmente estos archivos son solamente posibles de usar por los administradores, así que con este tipo de archivos los usuarios no tendrán acción alguna, solamente que este tipo de secciones se abarcaran mucho más adelante.

Ejemplo de código PHP.

```

<?php
    include("IControlador.php");
    //RECIBIR EL DATO
    $IDCirculacion=$_GET['IDCirculacion'];

    //IMPRIMIR SENTENCIA SQL
    print("ID de las tarjetas de circulación que se va a
eliminar:".$IDCirculacion);

    $SQL= "DELETE FROM tarjetascirculacion WHERE
IDCirculacion='".$IDCirculacion'";
    print($SQL);

    //Enviar el SMBD
    $Con=Conectar();
    $ResultSet=Ejecutar($Con,$SQL);
    Desconectar($Con);
?>

```

¿Por qué utilizar este tipo de funciones?

Las funciones que nos garantizan la eliminación de elementos dentro de una fila en SQL nos dan varias ventajas, algunas de estas pueden ser que principalmente con

esto genera una mejor administración en la base de datos, así como también una forma mucho más sencilla de poder rehacer las cosas en caso de que algo haya salido mal. Principalmente este archivo funciona ingresando alguno de los datos de la respectiva fila que se va a analizar, en este caso borrar. Una vez que sea la correcta será eliminada por el administrador en caso de que sea necesario. Así mismo es que funciona esta parte del código.

Objetivo 4: Interfaz visual(Frontend).

- A la par que tendremos archivos PHP que funcionaran como eliminadores de información, también tenemos que manejar el cómo es que estos se verán reflejados para las personas que manejan estos documentos.

Una vez que los archivos PHP son funcionales y contiene la sentencia SQL necesaria para que funcione correctamente, ahora toca la parte visual, para que se pueda analizar correctamente lo que tenemos que hacer utilizando estos archivos, estos son llamados en el sistema como “FD” ejemplo “FDCentroVerificación” es así que serán identificados por los administrativos cuando se necesite de su uso.

Código de ejemplo: FDCentroVerificaciones.

```
<html>
  <form method="post" action="DCentroVerificacion.php">
    <label>Eliminar los registros de los centros de
verificación:</label>
    <p></p>

    <label>CentroVerificacion</label>
    <input type="number" id="CentroVerificacion"
name="CentroVerificacion">
    <p></p>

    <input type="submit" value="Enviar" />
  </form>
</html>
```

Como se puede ver se trata de un código HTML bastante corto, por lo que solamente es una entrada visual sencilla para que sea mucho más sencillo que se haga la eliminación del registro esperado, así mismo una vez que se haga se tiene que comprobar que este se haya hecho satisfactoriamente y no tenga una serie de inconvenientes que puede suceder por culpa de como funciona la base de datos. Regresando a como funciona la base de datos al tener varias relaciones entre las tablas, es probable que en algún momento que se necesite realizar alguna eliminación de datos no sea posible debido a que esta tiene que estar relacionada

con alguna otra tabla que tiene alguno de los datos. Por lo que debe revisar el orden de entrada de datos para poder eliminarlos en caso de que sea necesario.

El posible cambio visual sería representado en la presentación del proyecto por culpa de la falta de tiempo.

Objetivo 5: Actualización.

- Así como tenemos la capacidad de eliminar los registros así como instalarlos también es importante que tengamos la oportunidad de actualizar los mismos haciendo uso de la sentencia SQL, UPDATE. Con esta nueva sentencia tendremos que hacer una serie de nuevos archivos PHP para poder hacer que los datos de algunas celdas se actualicen correctamente para que no tengamos problemas al realizar cambios en los datos que están presentes en la base de datos.

Como mencionamos buscamos tener alguna forma de actualización en la base de datos se busca que tengamos una herramienta de actualización en caso de que solo algunos de los campos de información que hayan sido llenados por los usuarios este mal o necesite un cambio, no se elimine toda la información del sistema para que se pueda hacer de una sola consulta para que no tenga problemas con otras secciones diferentes del código. Esto se va a realizar mediante archivos llamados "FU" por ejemplo "FUCentroVerificaciones", mediante estos archivos es que se hará la actualización.

Ejemplo:

```
<?php
$IDConductor = $_GET['IDConductor'];
include("IControlador.php");
$SQL = "SELECT * FROM conductores WHERE IDConductor='$IDConductor'";

$Con = Conectar();
$resultset = Ejecutar($Con, $SQL);
$Fila = mysqli_fetch_row($resultset);
Desconectar($Con);
?>

<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Document</title>
```

```
</head>
<body>
    <form method="post" action="UConductores.php">
        <label>Conductores</label>
        <br />

        <label>IDConductor</label>
        <input type="number" id="IDConductor" name="IDConductor"
value="<?php print($Fila[0]) ?>" />
        <br />

        <label>Nombre</label>
        <input type="text" id="Nombre" name="Nombre" maxlength="50"
value="<?php print($Fila[1]) ?>" />
        <br />

        <label>FechaNacimiento</label>
        <input type="date" id="FechaNacimiento" name="FechaNacimiento"
value="<?php print($Fila[2]) ?>" />
        <br />

        <label>Firma</label>
        <input type="file" id="Firma" name="Firma" value="<?php
print($Fila[3]) ?>" />
        <br />

        <label>Foto</label>
        <input type="file" id="Foto" name="Foto" maxlength="200"
value="<?php print($Fila[4]) ?>" />
        <br />

        <label>TipoSangre</label>
        <input type="password" id="TipoSangre" name="TipoSangre"
maxlength="3" value="<?php print($Fila[5]) ?>" />
        <br />

        <label>Donador</label>
        <input type="radio" id="Donador" name="Donador" value="1" <?php
if($Fila[6] == 1) echo 'checked'; ?>> sí
        <input type="radio" id="Donador" name="Donador" value="0" <?php
if($Fila[6] == 0) echo 'checked'; ?>> no
        <br />
```

```

        <label>Licencia</label>
<select name="Licencia" id="Licencia">
    <option value="A" <?php if(trim($Fila[8]) == 'A') echo 'selected';
?>>Automovilista</option>
    <option value="B" <?php if(trim($Fila[8]) == 'B') echo 'selected';
?>>Chofer</option>
    <option value="C" <?php if(trim($Fila[8]) == 'C') echo 'selected';
?>>Motociclista</option>
</select>
<br />

    <input type="submit" value="Enviar" />
</form>
</body>
</html>

```

Este código lo que espera es actualizar una celda de información que no esté correctamente se espera actualizará, como podemos ver este archivo combina PHP con HTML, por lo que unimos el cómo llamamos a nuestro controlador con la interfaz visual que veremos para que sea mucho más ameno el como se verá reflejado la información a consultar y en caso de que fuera necesario nos mostrará una ventana con los cambios realizados y estará completado esta sección.

Este al ser el HTML de lo que veremos también cuenta con su respectivo PHP, el cual contiene las instrucciones y los campos de la base de datos para realizar la sentencia UPDATE de SQL.

Ejemplo de código CentroVerificaciones.

```

<?php

include("IControlador.php");
$NumeroCentro=$_POST['NumeroCentro'];
$NumerodeLinea=$_POST['NumerodeLinea'];
$Municipio=$_POST['Municipio'];
$Verificacion=$_POST['Verificacion'];

//print("DatosDelAuto= ".$DatosDelAuto."<br>");
//print("NumerodeSerie= ".$NumerodeSerie."<br>");
//print("NumeroVerificacion= ".$NumeroVerificacion."<br>");
//print("Municipio= ".$Municipio."<br>");
//print("Verificacion= ".$Verificacion."<br>");

```

```

$SQL="UPDATE centrosverificaciones SET
NumerodeLinea='$NumerodeLinea',
Municipio='$Municipio',
Verificacion='$Verificacion'
WHERE NumeroCentro='$NumeroCentro'
";
//print($SQL);

//$SQL="INSERT INTO centrosverificaciones
VALUES('$Verificacion','$DatosDelAuto','$Municipio','$NumerodeSerie','$
NumeroVerificacion');";
//print($SQL);

//Enviar el SMBD
$Con=Conectar();
$ResultSet=Ejecutar($Con,$SQL);

if($ResultSet == 1){
print("1 FILA REGISTRADA");
}
else {
    print("ERROR ".mysqli_error($Con));
}
Desconectar($Con);
?>

```

Objetivo 6: Implementación de funciones.

- Una vez teniendo estos archivos, es necesario que se implementen sus funciones en un solo archivo para agilizar el tiempo en caso de tener que hacer alguna sentencia de actualización o eliminación.

Esto se hace principalmente teniendo en cuenta algunos factores como en caso de tener que cambiar solo algunos datos, no estar pasando de un lado a otro con los archivos FU y FD, sino tenerlo todo en un solo apartado funcional, o en caso de que lo desee el operador de la base de datos utilizar los archivos pasados pero la opción de tenerlo junto es posible por lo que tenemos que hacer es una nueva serie de archivos llamados esta vez “C” ejemplo “CCentroVerificaciones” estos archivos cuentan con la capacidad de eliminar registros en la base de datos así como también el actualizarlos.

Ejemplo de CCentroVerificaciones:

```
<html>
```

```

<form>
    <label>Criterio</label>
    <input type = text id="Criterio" name="Criterio">
    <label>Atributo</label><br>
    <input type ="radio" id="Atributo" name="Atributo"
value="NumeroCentro" required> NumeroCentro <br>
    <input type ="radio" id="Atributo" name="Atributo"
value="NumerodeLinea"> NumerodeLinea <br>
    <input type ="radio" id="Atributo" name="Atributo"
value="Municipio"> Municipio <br>
    <input type ="radio" id="Atributo" name="Atributo"
value="Verificacion"> Verificacion <br>
    <input type ="submit">
</form>
</html>

```

```

<?php
    if(isset($_GET['Criterio'])){
        $Criterio = $_GET['Criterio'];
        $Atributo = $_GET['Atributo'];
        include("IControlador.php");
        $Con=Conectar();
        $SQL="SELECT * FROM centrosverificaciones WHERE $Atributo LIKE
'%"$Criterio"";
        $ResultSet=Ejecutar($Con, $SQL);

        $NFilas=mysqli_num_rows($ResultSet);

        print("<table>");
        print("<tr> <table border=1px>");
        print("<th>NumeroCentro</th>");
        print("<th>NumerodeLinea</th>");
        print("<th>Municipio</th>");
        print("<th>Verificacion</th>");
        print("<th>Eliminar</th>");
        print("<th>Actualizar</th>");
        print("</tr>");

        for($i=0; $i<$NFilas; $i++){
            $Fila = mysqli_fetch_row($ResultSet);

```

```

        print("<tr>");
        print("<td>".$Fila[0]."</td>");
        print("<td>".$Fila[1]."</td>");
        print("<td>".$Fila[2]."</td>");
        print("<td>".$Fila[3]."</td>");

        print('<td><a
href="DCentrosVerificaciones.php?DatosdelAuto='.$Fila[0].'">Eliminar</a
></td>');

        print('<td><a
href="FUCentrosVerificaciones.php?DatosdelAuto='.$Fila[0].'">Actualizar
</a></td>');

        print("</tr>");
    }

    print("</table>");

    Desconectar($Con);
}
?>

```

Esto más que nada le agrega la forma de ahorrar tiempo a quien está manejando el sistema para que tenga que utilizar los dos archivos de una vez en lugar de estar yendo de un lado a otro. De esta forma lo que tenemos en cuenta es realizar cambios de forma general para ahorrar tiempo y que sea más cómoda la integración al manejo del sistema.

Objetivo 7: Menús de diferenciación.

- Durante el desarrollo del proyecto se ha hablado sobre funcionalidades de administradores y usuarios, esto se verá manejado por esta sección ya que se espera que mediante cuentas que son creadas para los usuarios que son diferenciados por su función, los administradores tendrán la posibilidad de eliminar registros como de actualizarlos, mientras que los usuarios solo podrán consultar su información en la respectiva tabla que se encuentren.

Durante la implementación del sistema se hace mediante algunos archivos que servirán como método de acceso. El más importante es un archivo PHP llamado “ACCESO” este tendrá la sentencia completa de diferenciación entre usuarios y administradores, para saber que es cada usuario, se crea una tabla nueva en la base de datos donde se añadirán los usuarios dependiendo si son administradores o usuarios con una serie de restricciones que tendrán en las cuentas. Como lo es su

estado, si es activo o no, si están bloqueados o no entre otras. Cuando se quiera acceder a las tablas, tendrán un límite de 3 intentos, si fallan la cuenta se bloqueara y no se podrá acceder

Ejemplo ACCESO.PHP:

```
<?php
    $UserName = $_POST['UserName'];
    $Pwd = $_POST['Pwd'];
    include("IControlador.php");
    $Con = Conectar();
    $SQL = "SELECT * FROM cuentas WHERE UserName='$UserName'";
    $ResultSet = Ejecutar($Con, $SQL);
    $NFilas = mysqli_num_rows($ResultSet);
    if ($NFilas == 1) {
        print("El usuario existe.<br>");
        $DatosCuenta = mysqli_fetch_assoc($ResultSet);
        if ($Pwd == $DatosCuenta['Pwd']) {
            print("Contraseña Correcta <br>");
            if ($DatosCuenta['Status'] == 1) {
                print("Cuenta Activa <br>");
                if ($DatosCuenta['Bloqueado'] == 0) {
                    print("Cuenta no bloqueada. <br>");
                    if ($DatosCuenta['TipoCuenta'] == 'A') {
                        $SQL = "UPDATE cuentas SET intentos = 0 WHERE
UserName = '$UserName'";
                        Ejecutar($Con, $SQL);
                        print("<td>¡¡¡¡¡¡¡¡¡¡¡¡ENTRA COMO
ADMINISTRADOR!!!!!!!!</td><br>");
                        print('<td><a
href="CCentrosverificaciones.php">Entrar a Centros
Verificaciones</a></td><br>');
                        print('<td><a href="CConductores.php">Entrar a
Conductores</a></td><br>');
                        print('<td><a href="CLicencias.php">Entrar a
Licencias</a></td><br>');
                        print('<td><a href="CMultas.php">Entrar a
Multas</a></td><br>');
                        print('<td><a href="CPropietarios.php">Entrar a
Propietarios</a></td><br>');
                        print('<td><a
href="CTarjetascirculacion.php">Entrar a Tarjetas de
Circulacion</a></td><br>');
```



```

        print('<td><a href="CTenencias.php">Entrar a
Tenencias</a></td><br>');
        print('<td><a href="CVehiculos.php">Entrar a
Vehiculos</a></td><br>');
        print('<td><a href="CVerificaciones.php">Entrar
a Verificaciones</a></td><br>');
        print('<center>');
    } else {
        $SQL = "UPDATE cuentas SET intentos = 0 WHERE
UserName = '$UserName'";
        Ejecutar($Con, $SQL);
        print("iiiiiiiiii;ENTRA COMO
USUARIO!!!!!!!!!!<br>");
        print('<td><a
href="MUsuariosPruebaMuestra.php">Entra a
CentroVerificaciones</a></td><br>');
        print('<td><a href="MConductores.php">Entra a
Conductores</a></td><br>');
        print('<td><a href="MLicencias.php">Entra a
Licencias</a></td><br>');
        print('<td><a href="MMultas.php">Entra a
Multas</a></td><br>');
        print('<td><a href="MPropietarios.php">Entra a
Propietarios</a></td><br>');
        print('<td><a
href="MTarjetasCirculacion.php">Entra a Tarjetas de
Circulacion</a></td><br>');
        print('<td><a href="MTenencias.php">Entra a
Tenencias</a></td><br>');
        print('<td><a href="MVehiculos.php">Entra a
Vehiculos</a></td><br>');
        print('<td><a href="MVerificaciones.php">Entra
a Verificaciones</a></td><br>');
    }
    } else {
        print("Cuenta Bloqueada. <br>");
    }
    } else {
        print("Cuenta NO Activa <br>");
    }
    } else {

```

```

        print("Contraseña Incorrecta <br>");
        $intentos = $DatosCuenta['Intentos'] + 1;
        $SQL = "UPDATE cuentas SET intentos = $intentos WHERE
UserName = '$UserName'";
        Ejecutar($Con, $SQL);
        if ($intentos >= 3) {
            $SQL = "UPDATE cuentas SET Bloqueado = 1 WHERE UserName
= '$UserName'";
            Ejecutar($Con, $SQL);
            print("Lo siento, has bloqueado la cuenta. <a
href='MUsuarios.html'>Intentar de nuevo</a>");
        } else {
            print("Contraseña fallida. Intento $intentos de 3. <a
href='MUsuarios.html'>Intentar de nuevo</a>");
        }
    }
} else {
    print("<H1>El usuario no existe</H1><br>");
    print('<center>');
}
Desconectar($Con);
?>

```

Pero este solo es el archivo que se encarga de llamar a los demás y tiene las sentencias de las restricciones. Porque el archivo que es el inicio de sesión que se muestra al usuario es el siguiente.

Ejemplo de código en HTML para el inicio de sesión.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Login Form</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            display: flex;
            justify-content: center;

```

```
        align-items: center;
        height: 100vh;
        margin: 0;
        background-color: #007bff;
    }
    .login-container {
        background-color: white;
        padding: 2rem;
        border-radius: 8px;
        box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
        width: 300px;
    }
    h1 {
        text-align: center;
        color: #333;
    }
    form {
        display: flex;
        flex-direction: column;
    }
    label {
        margin-bottom: 0.5rem;
        color: #666;
    }
    input {
        padding: 0.5rem;
        margin-bottom: 1rem;
        border: 1px solid #ccc;
        border-radius: 4px;
    }
    button {
        padding: 0.5rem;
        background-color: #007bff;
        color: white;
        border: none;
        border-radius: 4px;
        cursor: pointer;
        transition: background-color 0.3s ease;
    }
    button:hover {
        background-color: #0056b3;
    }
</style>
```

```
</head>
<body>
  <div class="login-container">
    <h1>Inicio de Sesion</h1>
    <form method="post" action="Acceso.php">
      <label>Nombre de Usuario:</label>
      <input type="text" id="UserName" name="UserName" required>
      <label>Contraseña:</label>
      <input type="password" id="Pwd" name="Pwd" required>
      <button type="submit">Iniciar Sesion</button>
    </form>
  </div>
</body>
</html>
```

Objetivo 8: Uso de la librería FPDF.

- Realizar uso de la librería FPDF para poder generar archivos PDF, para poder realizar cambios generalizados en caso de necesitar que se cambien de este modo utilizando la libreria podemos generar archivos que se guardaran como archivos de texto no plano. Tambien es posible generar estos pero en un archivo de texto sin formato el clasico TXT.

Durante la implementación de esta parte se necesita que tengamos a la mano la libreria FPDF, que tendremos que descargarla desde la pagina web <http://www.fpdf.org> por lo que una vez descargada la libreria nos dara un archivo comprimido que debemos descomprimir y todos los archivos generados se deben pasar a nuestro repositorio de trabajo siendo este la carpeta dentro de XAMPP. DSI30.

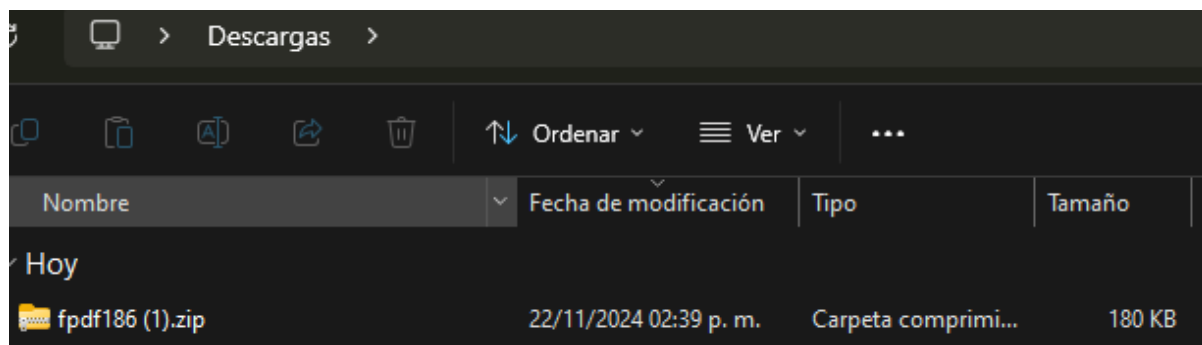
Paso 1:

Entrar a la pagina web y descargar la libreria FPDF.

<http://www.fpdf.org>

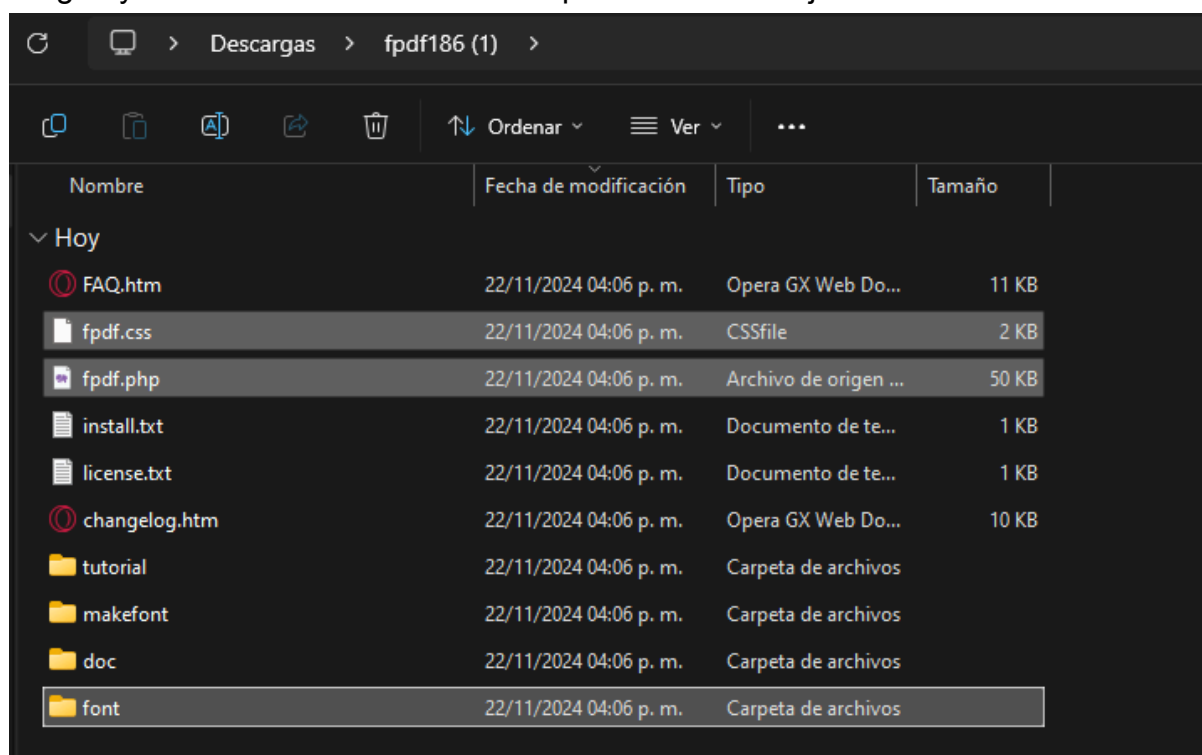
Paso 2:

Descomprimir la libreria, que principalmente se guarda en la carpeta descargas o en la que se haya predefinido por el usuario.



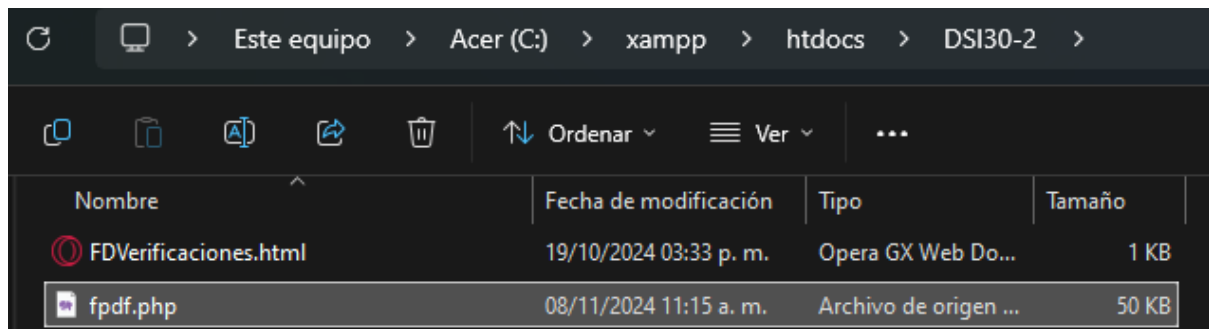
Paso 3:

Una vez descomprimido tomaremos los archivos que están seleccionados en la imagen y iremos a la ruta de nuestro repositorio de trabajo.



Paso 4:

Una vez ya en nuestro repositorio de trabajo, donde están todos los archivos, pegaremos el archivo llamado FPDF.PHP, junto con los demás archivos que serán de necesidad para utilizar correctamente la librería FPDF.



Objetivo 9: Generación de PDF 's.

- Utilizando la librería FPDF, podemos crear como su nombre lo dice archivos PDF utilizando PHP, una vez con estos archivos lo esperado es generar archivos por medio de la base de datos. Generando archivos por medio de vistas en SQL, así como también es esperado generar archivos como por ejemplo. la expedición de licencias, multas... etc.

Con esto lo que haremos es tener una forma de darle al usuario un archivo que puede dejar en formato digital o tenerlo en formato físico con sus respectivos datos que se mostrarán dependiendo del documento que sea requerido. También se puede generar este reporte pero en un archivo de texto plano como lo es el confiable txt. De esta forma nos cercioramos de tener un correcto control de los datos así como lo que le mostramos a los usuarios.

Ejemplo: Expedición de licencias.

```
<?php
///INVOCAR A LOS DATOS

include("IControlador.php");

$Con=Conectar();
$SQL = "SELECT * FROM datoslicencia";
$ResultSet=Ejecutar($Con, $SQL);
$Datos = mysqli_fetch_row($ResultSet);
Desconectar($Con);

////////////////////////////////////

require('fpdf.php'); // Incluye el archivo FPDF
```

```
$pdf = new FPDF('P', 'mm', 'Legal'); // Constructor
$pdf->AddPage();
$pdf->SetFont('Arial', '', 16); // Establece la fuente antes de agregar
texto
$pdf->Image("ESCUDO IMAGEN MX.png", 10, 10, 30);
$pdf->SetXY(45, 10);
$pdf->Cell(120, 10, 'Estados Unidos Mexicanos');

$pdf->SetXY(40, 10);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(40, 15);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(40, 20);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(40, 25);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(40, 30);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(40, 35);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(40, 40);
$pdf->Cell(120, 10, '|');

$pdf->SetXY(45, 20);
$pdf->SetFont('Arial', '', 13);
$pdf->Cell(120, 10, 'Poder Ejecutivo del Estado de Queretaro');

$pdf->SetXY(45, 20);
$pdf->SetFont('Arial', 'B', 15);
$pdf->Cell(120, 35, 'Secretaria de Seguridad Ciudadana');

$pdf->SetXY(45, 27);
$pdf->SetFont('Arial', 'B', 15);
$pdf->Cell(120, 35, 'Licencia para Conducir');

$pdf->SetXY(80, 90);
$pdf->SetFont('Arial', '', 10);
```

```
$pdf->Cell(120, 35, 'No. de Licencia');

$pdf->SetXY(74, 100);
$pdf->SetFont('Arial', 'B', 20);
$pdf->Cell(120, 35, $Datos[0]);

$pdf->SetXY(67, 110);
$pdf->SetFont('Arial', 'B', 13);
$pdf->Cell(120, 35, 'AUTOMOVILISTA');

$pdf->SetXY(186, 150);
$pdf->SetFont('Arial', '', 9);
$pdf->Cell(120, 35, 'Nombre');

// El original era $pdf->SetXY(165, 157);
// Dividimos el nombre completo en partes
$partes = explode(' ', $Datos[1] ?? 'Nombre no disponible');

// Asignamos las partes a variables según la cantidad disponible
$nombre = $partes[0] ?? '';
$nombre1 = $partes[1] ?? '';
$apellido1 = $partes[2] ?? '';
$apellido2 = isset($partes[3]) ? implode(' ', array_slice($partes, 3))
: '';

// Colocamos cada parte en una celda distinta
$pdf->SetFont('Arial', 'B', 25);

// Celda para el primer apellido
$pdf->SetXY(168, 157);
$pdf->SetFont('Arial', '', 25);
$pdf->Cell(60, 35, $apellido1);

// Celda para el segundo apellido
$pdf->SetXY(163, 165);
$pdf->SetFont('Arial', '', 25);
$pdf->Cell(60, 35, $apellido2);

// Celda para el nombre
$pdf->SetXY(100, 175);
$pdf->SetFont('Arial', 'B', 35);
$pdf->Cell(60, 35, $nombre);
```



```
// Celda para el nombre 2
$pdf->SetXY(140, 175);
$pdf->SetFont('Arial', 'B', 35);
$pdf->Cell(60, 35, $nombre1);

$pdf->SetXY(152, 165);
$pdf->SetFont('Arial', '', 25);
// $pdf->Cell(120, 35, 'MARTINEZ');

$pdf->SetXY(76, 175);
$pdf->SetFont('Arial', 'B', 35);
// $pdf->Cell(120, 35, 'FRANCISCO JAVIER');

$pdf->SetXY(176, 185);
$pdf->SetFont('Arial', '', 9);
$pdf->Cell(40, 35, 'Observaciones');

$pdf->SetXY(10, 200);
$pdf->SetFont('Arial', '', 9);
$pdf->Cell(40, 35, 'Fecha de Nacimiento');

$pdf->SetXY(10, 210);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(40, 35, $Datos[2]);

$pdf->SetXY(10, 220);
$pdf->SetFont('Arial', '', 9);
$pdf->Cell(40, 35, 'Fecha de Expedicion');

$pdf->SetXY(10, 230);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(40, 35, $Datos[3]);

$pdf->SetXY(10, 240);
$pdf->SetFont('Arial', 'B', 9);
$pdf->Cell(40, 35, 'Valida hasta');

$pdf->SetXY(10, 250);
$pdf->SetFont('Arial', 'B', 20);
$pdf->Cell(40, 35, $Datos[4]);
```

```
$pdf->SetXY(10, 260);
$pdf->SetFont('Arial', 'B', 9);
$pdf->Cell(40, 35, 'Antigüedad');

$pdf->SetXY(10, 270);
$pdf->SetFont('Arial', 'B', 15);
$pdf->Cell(40, 35, '14');

$pdf->SetXY(90, 256);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(40, 35, 'Firma');

$pdf->SetXY(70, 295);
$pdf->SetFont('Arial', '', 9);
$pdf->Cell(40, 35, 'AUTORIZO PARA QUE LA PRESENTE SEA');

$pdf->SetXY(67, 300);
$pdf->SetFont('Arial', '', 9);
$pdf->Cell(40, 35, 'RECABADA COMO GARANTIA DE INFRACCION');

$pdf->Image( "LETRA A PNG.png", 10, 310, 20);

$pdf->Image( "Chems.png", 85, 280, 20);

$pdf->Image( "Chems.png", 110, 70, 90);

//SEGUNDA PAGÍNA

$pdf->AddPage();
$pdf->SetFont('Arial', '', 16);

$pdf->Image( "URGENCIAS.png", 10, 10, 30);
$pdf->Image( "DENUNCIA.png", 170, 10, 30);

$pdf->SetXY(85, 20);
$pdf->SetFont('Arial', 'B', 20);
$pdf->Cell(42, 10, $Datos[5], 1);

$pdf->SetXY(180, 33);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 5, 'Domicilio');
```

```
// EL ORIGINAL ERA $pdf->SetXY(140, 39);
$pdf->SetXY(170, 39);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, $Datos[10]);

$pdf->SetXY(185, 47);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, 'SN');

// EL ORIGINAL ERA $pdf->SetXY(157, 55);
$pdf->SetXY(170, 55);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, $Datos[10]);

$pdf->SetXY(162, 63);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, 'C.P 76260');

// EL ORIGINAL ERA $pdf->SetXY(150, 71);
$pdf->SetXY(170, 71);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, $Datos[11]);

$pdf->SetXY(10, 110);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Restricciones');

$pdf->SetXY(10, 117);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, 'Ninguna');

$pdf->SetXY(170, 110);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Grupo Sanguineo');

// EL ORIGINAL ERA $pdf->SetXY(178, 117);
$pdf->SetXY(185, 117);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, $Datos[8]);

$pdf->SetXY(165, 125);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Donador de Organos');
```

```
$pdf->SetXY(191, 133);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, $Datos[9]);

$pdf->SetXY(159, 141);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Numero de Emergencias');

$pdf->SetXY(60, 217);
$pdf->SetFont('Arial', 'B', 15);
$pdf->Cell(120, 10, 'MTRQ EN GPA. MIGUEL ANGEL CONTRERAS ALVAREZ');

$pdf->SetXY(107, 225);
$pdf->SetFont('Arial', 'B', 13);
$pdf->Cell(120, 10, 'SECRETARIO DE SEGURIDAD CIUDADANA');

$pdf->SetXY(140, 149);
$pdf->SetFont('Arial', '', 20);
$pdf->Cell(120, 10, '000-442-253-6464');

$pdf->SetXY(10, 240);
$pdf->SetFont('Arial', 'B', 10);
$pdf->Cell(120, 10, 'Fundamento Legal');

$pdf->SetXY(10, 245);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Atributo 19 fraccion X 33 fraccion de la ley Organica del poder ejecutivo del estado de Queretaro. atributo 9');

$pdf->SetXY(10, 250);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'fraccion Xly 55 de la ley de transito del Estado de Queretaro atribuye de ley de procedimientos');

$pdf->SetXY(10, 255);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Administrativo del Estado de Queretaro, atributo 134, 135, 136, 137, 138, 139, 140, 141, 142 y 143 del reglamento de');

$pdf->SetXY(10, 260);
$pdf->SetFont('Arial', '', 10);
```

```

$pdf->Cell(120, 10, 'Transito Estado de Queretaro, atributo 6, fraccion
IV, en 20 fraccion de la ley de la Secretaria de');

$pdf->SetXY(10, 265);
$pdf->SetFont('Arial', '', 10);
$pdf->Cell(120, 10, 'Seguridad Ciudadana');

$pdf->Image( "Chems.png", 160, 175, 40);

$pdf->Image( "SEGURIDAD CIUDADANA.png", 120, 310, 80);

$pdf->Image( "AUTO.png", 10, 80, 30);
$pdf->Image( "AUTO.png", 40, 80, 30);
$pdf->Image( "AUTO.png", 70, 80, 30);
$pdf->Image( "AUTO.png", 100, 80, 30);
$pdf->Image( "AUTO.png", 130, 80, 30);
$pdf->Image( "AUTO.png", 160, 80, 30);

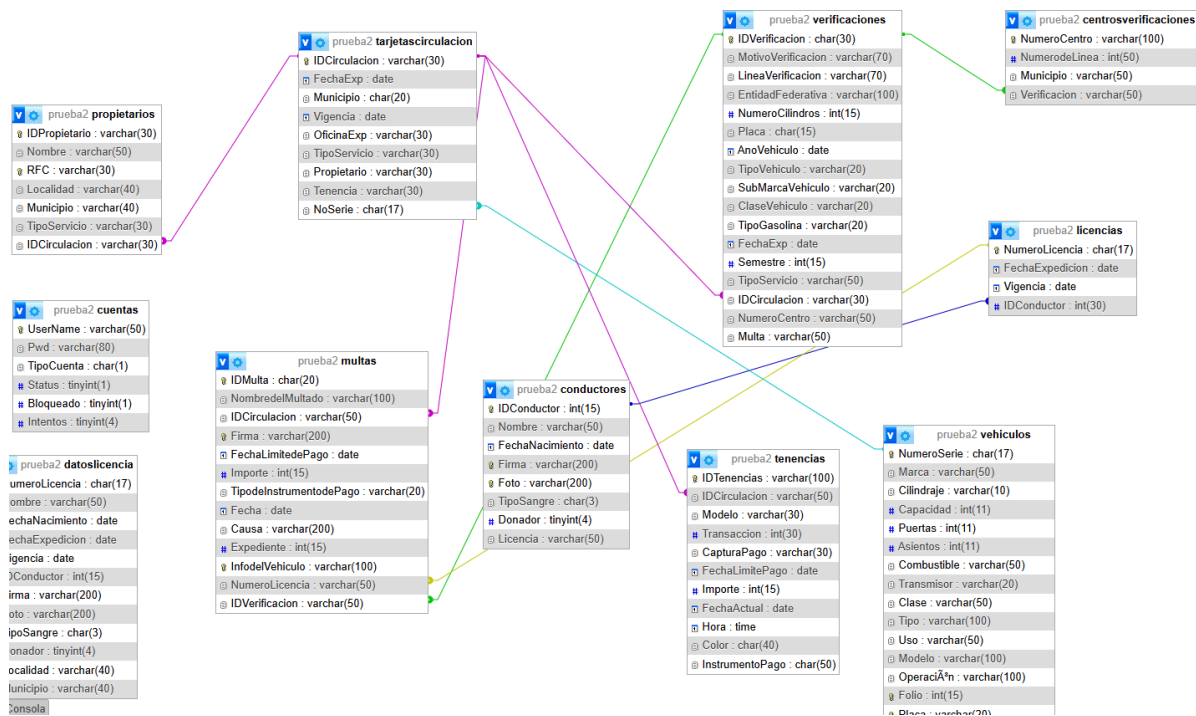
$pdf->Output();
?>

```

MODELO RELACIONAL DE LA BASE DE DATOS.

El modelo relacional de base de datos está conformado por un código en SQL, que busca simular/imitar a máxima eficiencia una credencial, siendo diferente en cada tabla SQL la cual se esté realizando, esto se puede reflejar en los datos que serán presentados en la misma y lo cual también es importante por la necesidad de luego imprimir los documentos necesarios para que sean lo más fieles posibles al original. Nuestra base de datos cuenta con relaciones entre tablas que tienen que ser complementarias las unas con las otras. Siendo que esta parte del proyecto nos da una buena idea de lo que estamos realizando de forma general.

Con nuestro modelo relacional de base de datos esperamos que sea lo más funcional posible relacionado con otras secciones que vamos a utilizar como lo son archivos que estuvimos implementando para que todo funcione correctamente, es así que una vez todo bien realizado procedemos a las siguientes partes del proyecto.



La base de datos cuenta con múltiples relaciones que son usadas a forma de complemento con las demás, tablas esto lo hacemos para primero que nada tener una forma de poder rastrear los datos de un solo usuario cuando éste ingrese datos que son únicamente pertenecientes a él, así mismo con este tipo de completación de datos es más probable que exista una mejor administración de los mismos. Así como también se espera una mejor consulta de los mismos. La base de datos consta de 9 tablas con 9 relaciones. Por lo que una vez esté todo creado podemos seguir avanzando con el desarrollo del proyecto. Mediante lo que vamos a realizar otros rasgos generales.

DIAGRAMA DE COMPONENTES

CREAR CUENTA

Nombre
Tu nombre

Apellido
Tu apellido

Usuario
correo@ejemplo.com

Contraseña
12345678

☐ Acepto los términos y condiciones

CREAR CUENTA

¿Ya tienes una cuenta?
[Iniciar sesión](#)

[Volver al inicio](#)

Enseguida te redirecciona a una página, en donde te indica que la cuenta se creó con éxito.

Presentamos un método de inicio de sesión que se le ofrece a los usuarios, con cuentas existentes, presentándoles de forma sencilla, practica el ingreso de los datos, una interfaz, esta interfaz presenta dos secciones, para ingresar datos pedidos, que son: usuario, contraseña, en caso de que sean correctos los datos ingresados, solo en ese caso, se le permitirá al usuario el ingreso al sistema, solo hay 2 tipos de cuentas, depende del tipo de cuenta al que se trate de ingresar, el apartado será diferente

INICIAR SESIÓN

Usuario
correo@ejemplo.com

Contraseña
12345678

☐ Recordarme [¿Olvidaste tu contraseña?](#)

INGRESAR

¿No tienes una cuenta?
[Crear cuenta](#)

[Volver al inicio](#)

Ingreso de datos

Creacion de cuenta nueva

Administrador:

Bienvenido, Administrador

ENTRA COMO ADMINISTRADOR!!!!!!

Sección Actualizar y Borrar datos

Entrar a Centros Verificaciones

Entrar a Conductores

Entrar a Licencias

Entrar a Multas

Entrar a Propietarios

Entrar a Tarjetas de Circulación

Entrar a Tenencias

Entrar a Vehículos

Entrar a Verificaciones

Sección Ingreso de Datos

Ingreso de datos en Centro Verificaciones

Ingreso de datos en Conductores

Ingreso de datos en Licencias

Ingreso de datos en Multas

Ingreso de datos en Propietarios

Ingreso de datos en Tarjetas de Circulación

Ingreso de datos en Tenencias

Ingreso de datos en Vehículos

Ingreso de datos en Verificaciones

Cerrar sesión

Usuario general:

Bienvenido, Usuario

ENTRA COMO USUARIO!!!!!!!

Entra a Centro Verificaciones

Entra a Conductores

Entra a Licencias

Entra a Multas

Entra a Propietarios

Entra a Tarjetas de Circulación

Entra a Tenencias

Entra a Vehículos

Entra a Verificaciones

Cerrar sesión

En caso de no ingresar los datos de forma correcta, se le indicará al usuario que sus datos ingresados previamente, por el mismo, son erróneos, para proceder a intentar de nuevo.

Contraseña incorrecta.

Pero no sin tomar en cuenta que el usuario solo tiene 3 intentos, para poder ingresar, en cada uno de los intentos se le sumará 1, hasta que se bloquee de forma automática, ya no podría seguir intentando, después del bloqueo, en cada intento que realice, mostrará el mensaje de “Cuenta inactiva o bloqueada” en ese caso, para desbloquear la cuenta, sería necesario hablar con soporte o un administrador de sistema, para permitir nuevamente el acceso a su cuenta


Cuenta inactiva o bloqueada.

ADMINISTRADOR

Cuando el Admin este ingresando a la interfaz, que le permitirá ingresar los datos que necesite a la base de datos le mostrará una interfaz de HTML, donde este debe ingresar datos, algunos debe recordarlos para poder seguir con el ingreso de sus datos en siguientes pestañas.

En este caso de ingresar estos datos, el dato a recordar para ingresar luego, en otra sección es ID Conductor, en todas las secciones, el dato a recordar están hasta arriba, como en este caso y cuando se ingresen los datos, mostrar un mensaje de “Registro exitoso”

Luego pasaremos a Licencias

A screenshot of a web form titled "Registrar Licencia" with a red header and a white body. The form contains five input fields: "Número de Licencia" with the value "666", "Fecha de Expedición" with the value "14/11/2024", "Vigencia" with the value "29/11/2024", and "ID del Conductor" with the value "1212". Each date field has a calendar icon to its right. At the bottom of the form is a red button labeled "Enviar". The entire form is centered on a light red background.

Registrar Licencia	
Número de Licencia	666
Fecha de Expedición	14/11/2024
Vigencia	29/11/2024
ID del Conductor	1212
<button>Enviar</button>	

En este caso se trata de crear una licencia, para el anterior “Conductor” registrado, en este caso el dato a recordar es “Número de Licencia” y en esta tabla se ocupa el primer dato ingresado, del formulario anterior, osea “ID de Conductor”, se presiona el botón de “Enviar” y listo, te mostrará en la parte superior un mensaje, que confirma el ingreso correcto de datos.

Y de tal forma sucesivamente, hasta haber llenado todos los formularios, para completar los datos de toda la persona, cabe recalcar que se tienen que ingresar a los formularios de forma ordenada, de “Conductores”, hasta abajo (“Verificaciones”)



Exceptuando el formulario de “Centro Verificaciones”, en caso de que sea necesario, ya que, crear un nuevo “Centro Verificaciones” para cada usuario, por que varios usuarios pueden usar el ID de un solo “Centro Verificacion”,

DELETE

Si uno de los administradores desea hacer una consulta en la base de datos con el ID o datos del usuario puede hacerlo sin embargo este podrá también hacer algunas modificaciones al mismo registro como lo puede ser una actualización que esto se verá más tarde así como también una eliminación.

¿Por qué hacer una eliminación de registros en la Base de datos?

Esto se hace con el hecho de querer mantener un buen flujo de información en la base de datos, puede que ocurran múltiples situaciones como algún mal ingreso de los datos de un usuario así como posibles duplicaciones del mismo es por esto que al eliminar estos registros se sigue manteniendo un buen flujo en el sistema como también se le da la oportunidad al usuario que este cambie los datos que acaba de ingresar.

La forma de borrado de datos que manejamos en la base de datos es por medio de un archivo llamado FD, estos archivos lo que hacen es llamar a uno de los datos que existan en una fila de información en la base de datos esto proporciona la vista completa de los datos que serán eliminados, una vez ingresado el ID que esté

relacionado a la información que se borrara, esta será eliminada para que se hagan los ajustes necesarios o para que el usuario que ingresó esos datos lo haga.

El proceso para la eliminación de algún dato que desee, el apartado se muestra al inicio, al momento de ingresar como Administrador

Bienvenido, Administrador

iiiiiii¡ENTRA COMO ADMINISTRADOR!!!!!!!

Sección Actualizar y Borrar datos

Entrar a Centros Verificaciones

Entrar a Conductores

Entrar a Licencias

Entrar a Multas

Entrar a Propietarios

Entrar a Tarjetas de Circulación

Entrar a Tenencias

Entrar a Vehículos

Entrar a Verificaciones

Eliminar datos relacionados a la tablas, de abajo hacia arriba,”Verificaciones” a “Conductores”, en otro caso, no te permitirá, mostrará un error.
 Seleccione de qué formulario quieres eliminar los datos de x, usuario, en este caso, como ejemplo seleccionar el de “Conductores”

Consulta Conductores

Ingresa tu criterio de búsqueda:

Selecciona el atributo:

- ☐ ID Conductor
- ☐ Nombre
- ☐ Fecha de Nacimiento
- ☐ Firma
- ☐ Foto
- ☐ Tipo de Sangre
- ☐ Donador
- ☐ Licencia

Consultar

Ya que lo seleccionamos, decidimos en qué dato, de los que ingresamos en Conductores, vamos a usar, para referenciar a los datos, de dicho usuario, escogí “ID Conductor”, le picamos a “Consultar”

Consulta Conductores

Ingresa tu criterio de búsqueda:

1212

Selecciona el atributo:

- ☒ ID Conductor
- ☐ Nombre
- ☐ Fecha de Nacimiento
- ☐ Firma
- ☐ Foto
- ☐ Tipo de Sangre
- ☐ Donador
- ☐ Licencia

Consultar

Y nos muestra todos los datos relacionados, con ese dato ingresado, con la opción de Eliminar

Consulta Conductores

Ingresar tu criterio de búsqueda:

Selecciona el atributo:

☐ ID Conductor

☐ Nombre

☐ Fecha de Nacimiento

☐ Firma

☐ Foto

☐ Tipo de Sangre

☐ Donador

☐ Licencia

Consultar

ID Conductor	Nombre	Fecha de Nacimiento	Firma	Foto
1212	Edgar	2024-11-14	IMG(fotos-firmas)/1212Firma6743f6b3dfe80.png	IMG(fotos-firmas)/1212Foto6743f6b3dfe80.png

Consulta Conductores

Ingresar tu criterio de búsqueda:

Selecciona el atributo:

- ☐ ID Conductor
- ☐ Nombre
- ☐ Fecha de Nacimiento
- ☐ Firma
- ☐ Foto
- ☐ Tipo de Sangre
- ☐ Donador
- ☐ Licencia

Consultar

Foto	Tipo de Sangre	Donador	Licencia	Eliminar	Actualizar
IMG(fotos-firmas)/1212Foto6743f6b3dfe82.png	O+	0	Automovilista	Eliminar	Actualizar

Desplazando la barra, nos saldrán 2 opciones Eliminar/Delete, como es de suponer y como mencione con anterioridad, “Eliminar” funciona para el eliminar, todos los datos mostrados en esa tabla, sobre x usuario

Te mandará a una pantalla, en donde pedirá, el dato principal del mismo formulario que tratas de eliminar, en este caso será “ID de Conductor”

Eliminar Conductores

ID del Conductor a Eliminar

1212

Eliminar

Y en caso de que no esté relacionado con otra tabla, estos datos del usuario, serán borrados, sin problemas, mostrando un mensaje de confirmación.

UPDATE

La sentencia UPDATE le da la libertad al administrativo de que este haga los respectivas actualizaciones al sistema de base de datos en caso de que no sea necesario borrar una de las filas de información del usuario esto se hace principalmente para poder dar la pauta de corregir algo que sea necesario en lugar de tener que rehacer todo de nuevo por culpa de algún o algunos errores es así que una vez que todo esté bien desarrollado.

Estos archivos son llamados FU, se necesita que se ingrese una llave primaria de alguna tabla para que devuelva el registro en cuestión y usando una sentencia SQL UPDATE, vamos a poder cambiar los datos que estén mal en necesidad de que se deba hacer el cambio o un reajuste a la base de datos es así que con esto el administrador tendrá una buena libertad al momento de verificar qué datos deben ser actualizados o cuáles no, esto también queda a consideración en caso de que los datos no se puedan actualizar o se decida eliminar la fila de información.

El proceso es prácticamente el mismo, pero en lugar de seleccionar “Eliminar” seleccionaremos “Actualizar”

Consulta Conductores

Ingresar tu criterio de búsqueda:

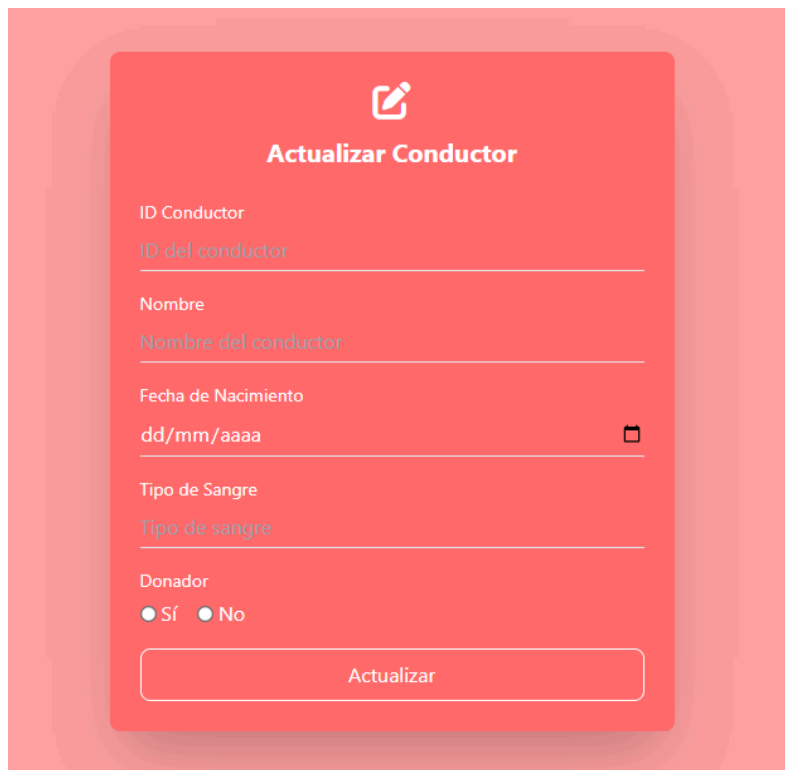
Selecciona el atributo:


- ☐ ID Conductor
- ☐ Nombre
- ☐ Fecha de Nacimiento
- ☐ Firma
- ☐ Foto
- ☐ Tipo de Sangre
- ☐ Donador
- ☐ Licencia

Consultar

Foto	Tipo de Sangre	Donador	Licencia	Eliminar	Actualizar
IMG(fotos-firmas)/1212Foto6743f6b3dfe82.png	O+	0	Automovilista	Eliminar	Actualizar

Te mandara a una página, con los datos que ya estaban ingresados, disponibles, para modificar en cualquier momento.

Un formulario de actualización de conductor con un fondo rojo. El formulario es un recuadro blanco con un icono de lápiz y papel en la parte superior. El título es "Actualizar Conductor". Hay campos de texto para "ID Conductor", "Nombre", "Fecha de Nacimiento" (con un icono de calendario) y "Tipo de Sangre". Hay un campo de selección para "Donador" con opciones "Sí" y "No". Un botón "Actualizar" está al final.



Actualizar Conductor

ID Conductor
ID del conductor

Nombre
Nombre del conductor

Fecha de Nacimiento
dd/mm/aaaa

Tipo de Sangre
Tipo de sangre

Donador
☒ Sí ☐ No

Actualizar

Finalmente picas “Actualizar” y se actualizarán los datos del formulario.

UPDATE Y DELETE


Mencionando anteriormente que el administrador tendrá la libertad de eliminar o actualizar los registros que este necesite, lo puede hacer directamente desde los archivos que están destinados a esa tarea o utilizar una combinación de ambos archivos que le darán la libertad de Actualizar o Eliminar registros, con esto lo que puede tener libertad es a no estar perdiendo tiempo en tener que cambiar de una pestaña con un tipo de archivo a otro sino simplemente cambiando la opción que desea hacer al realizar la consulta.

Estos archivos están llamados como C[Inicial del documento que accede a la tabla] estos archivos contienen de forma general las funciones de DELETE cómo UPDATE, así proporcionando una herramienta más que eficiente a los administrativos que usaran estas herramientas para poder hacer las labores que se tienen en mente al momento de realizar la tarea. Es así que una vez hecho estos cambios se espera que no haya alguna otra implementación en contenido relacionado con la base de datos.

USUARIOS

El sistema en este caso es más sencillo, ya que solo debe de permitir al usuario 1 cosa, visualizar sus datos ingresados en todos los formularios, lo cual hace que necesite mucho menos procesamientos, para esto.

La interfaz mostrará lo siguiente, al momento de iniciar sesión como Usuarios general

A screenshot of a user selection interface. It features a light red background with a central white rounded rectangle. At the top of the white rectangle, the text "Bienvenido, Usuario" is displayed in bold. Below this, a separator line is followed by the text "¡¡¡¡¡¡¡¡¡¡ENTRA COMO USUARIO!!!!!!!!!!". A list of ten red buttons with white text is arranged vertically: "Entra a Centro Verificaciones", "Entra a Conductores", "Entra a Licencias", "Entra a Multas", "Entra a Propietarios", "Entra a Tarjetas de Circulación", "Entra a Tenencias", "Entra a Vehículos", and "Entra a Verificaciones". At the bottom of the list is a smaller red button with white text labeled "Cerrar sesión".

Bienvenido, Usuario

¡¡¡¡¡¡¡¡¡¡ENTRA COMO USUARIO!!!!!!!!!!

Entra a Centro Verificaciones

Entra a Conductores

Entra a Licencias

Entra a Multas

Entra a Propietarios

Entra a Tarjetas de Circulación

Entra a Tenencias

Entra a Vehículos

Entra a Verificaciones

Cerrar sesión

El usuario selecciona cualquier apartado, del cual quiera ver sus datos, en este ejemplo, tomaremos “Conductores”

Ingresamos el ID del conductor

Consulta Conductores

Ingresar tu ID para ver tus datos:

1212

Selecciona el atributo:

☒ ID Conductor

Consultar

Y finalmente le mostrara sus datos al usuario

Consulta Conductores

Ingresar tu ID para ver tus datos:

Selecciona el atributo:


☒ ID Conductor

Consultar

ID Conductor	Nombre	Fecha de Nacimiento	Firma	Foto	Tipo de Sangre	Donador	Licencia
1212	Edgar	2024-11-14	IMG(fotos-firmas)/1212Firma6743f6b3dfe90.png	IMG(fotos-firmas)/1212Foto6743f6b3dfe92.png	O+	0	Automovilista

PDFS Y GUARDADO DE ARCHIVOS














La generación de PDF se hace gracias al uso de la librería de PDF, con esta librería lo que hacemos es generar un documento con base a alguna imagen que nos proporcionó el profesor para que tengamos documentos una vez que tengamos esta información, vamos a utilizar la libreria FPDF, para poder utilizarla debemos instalarla ya que no esta en nuestro editor de código, debemos acudir a la siguiente pagina <http://www.fpdf.org> para poder descargar la libreria.



[Home](#)
[Downloads](#)
[Tutorials](#)
[Documentation](#)
[FAQ](#)
[Scripts](#)
[Forum](#)
[Links](#)

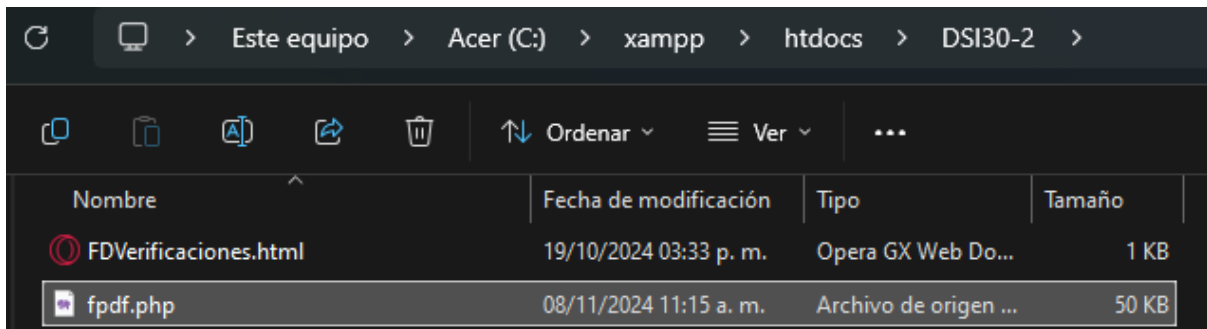
Downloads

You can view the changelog [here](#).

-  An Arabic version of documentation v1.53 is available in [ZIP](#) and [TGZ](#) formats (translation by [Abdullah E.AL-Mehmadi](#))
-  A Catalan version of documentation v1.81 is available in [ZIP](#) and [TGZ](#) formats (translation by [David Gimeno i Ayuso](#))
-  A Hungarian version of documentation v1.81 is available in [ZIP](#) and [TGZ](#) formats (translation by [Csaba Varga](#))
-  An Indonesian version of documentation v1.82 is available in [ZIP](#) and [TGZ](#) formats (translation by [Kang Oedin](#))
-  A Japanese version of documentation v1.52 is available in [ZIP](#) and [TGZ](#) formats (translation by [Matt Ataka](#))
-  A Korean version of documentation v1.6 is available in [ZIP](#) and [TGZ](#) formats (translation by [Kim Taegon](#))
-  A Dutch version of documentation v1.6 is available in [ZIP](#) and [TGZ](#) formats (translation by [Martijn Hogerheijde](#))
-  A Polish version of documentation v1.81 is available in [ZIP](#) and [TGZ](#) formats (translation by [Rafal Lysik](#))
-  A Brazilian Portuguese version of documentation v1.81 is available in [ZIP](#) and [TGZ](#) formats (translation by [Domingos Paraiso](#))
-  A Romanian version of documentation v1.52 is available in [ZIP](#) and [TGZ](#) formats (translation by [Cantemir Mihu](#))
-  A Russian version of documentation v1.53 is available in [ZIP](#) and [TGZ](#) formats (translation by [Olga Smila](#))
-  A Turkish version of documentation v1.86 is available in [ZIP](#) and [TGZ](#) formats (translation by [Hamza Ortatepe](#))
-  A Chinese version of documentation v1.52 is available in simplified ([ZIP](#), [TGZ](#)) and traditional language ([ZIP](#), [TGZ](#)) (translation by [Andrew Choi](#))

v1.86 (2023-06-25)
[ZIP](#) | [TGZ](#)

Una vez descargada la librería se debe enviar los archivos necesarios a nuestro repositorio de archivos para trabajar correctamente.



CONCLUSIONES

Edgar Betan Ramos

En conclusión el sistema recientemente creado, es para una eficaz forma de manejo, de datos en lo que respecta a las credenciales de "Tarjeta de circulación" y "Credencial de conducción", un interfaz sencilla, fácil e intuitiva, tanto para usuarios nuevos, como para viejos, una de las tantas funciones para los usuarios, es la visualización de sus datos ingresados, en cualquier momento que el desee, para verificar, todos, cada uno de los datos.

Victor Alí Salinas Ledesma:

Con este proyecto puedo concluir varias cosas, principalmente fue bastante difícil en ciertos aspectos del backend, vi que en ese campo necesito seguir mejorando aunque principalmente fui yo quien hizo el frontend de todo el proyecto. También me siento satisfecho con el resultado, tal vez tengamos algunas cosas que no puedo pasar por alto, pero para ser un proyecto que ya tocó backend si me gusta el resultado. Con este proyecto espero que en el futuro me ayude para demás proyectos para el futuro.

Jesús Santiago Macal Palacio:

Durante el desarrollo del proyecto pase por varias experiencias, al encargarme de buena parte del desarrollo backend tuve algunos problemas ya sea con los documentos creados en clase así como los cambios que se hicieron en el proyecto de forma general. Durante esta etapa del proyecto algo de los más complicado fue estar de acuerdo a los cambios que se harían al sistema con mi compañero quien se encargó de FrontEnd, por algunos problemas que teníamos en la vista general del proyecto. Pero haber realizado este proyecto fue bastante buena la experiencia que tuve con mis compañeros de trabajo así como también con el maestro. El sistema de control vehicular fue una experiencia bastante completa, aún cuando tenía que buscar algunos elementos de ayuda para implementar algunas secciones del código que no estaban del todo completas. Es por esto que al trabajar en compañía de otros compañeros el proyecto se llevó lo más ameno posible así como también el desarrollo de este documento. De esta forma puedo decir que el proyecto de control vehicular se completó totalmente. Así que no hay cuestión de realizar otros cambios.