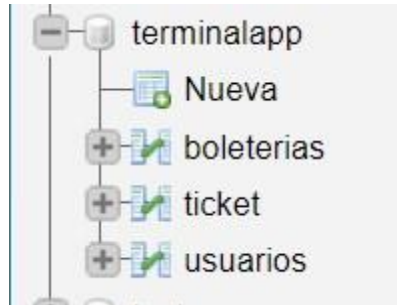


Manual del terminal APP

La aplicación web se trabajo con la base de datos Mariadb – Wampserver, Local host 3307



Tiene tres conexiones.

1.- Conexión de la base de datos terminalapp con la tabla boletería – Modulo boletería.

```
$connect = mysqli_connect("localhost:3307", "root", "", "terminalapp");
```

La función **mysqli_connect()** intenta abrir una conexión al servidor MySQL que se está ejecutando en *equipo_afitrión* el cual puede ser el nombre de un equipo o una dirección IP.

La cual nos permite llamar los datos de la base de datos terminalapp en este caso es para llamar los datos de las boleterías por medio de las id.

```
$query = "SELECT * FROM boleterias ORDER BY id_boleteria ASC";
```

```
$result = mysqli_query($connect, $query);
```

2.- Conexión de la base de datos terminalapp con la tabla Usuarios – Login

La segunda conexión nos permite acceder a la tabla usuarios de la base de datos terminalapp

```
<?php
$server = 'localhost:3307';
$username = 'root';
$password = '';
$databse = 'terminalapp';
try {
    $conn = new PDO("mysql:host=$server;dbname=$databse;", $username, $password);
} catch (PDOException $e) {
    die('Connection Failed: ' . $e->getMessage());
}
?>
```

Usamos require para por que los Scripts no deberían ejecutarse si faltan archivos necesarios o están mal nombrados.

```
require '../../php/database.php';
```

llamamos a la tabla usuario por la Id

```
if (isset($_SESSION['user_id'])) {
    $records = $conn->prepare('SELECT * FROM usuarios WHERE Id = :id');
    $records->bindParam(':id', $_SESSION['user_id']);
    $records->execute();
    $results = $records->fetch(PDO::FETCH_ASSOC);

    $user = null;

    if (count($results) > 0) {
        $user = $results;
    }
}
```

Login

Usamos la sentencia if para cumplir la expresión condicional que en este caso es el correo y password que vamos a confirmar de su existencia en la tabla usuarios buscando su correo y password una vez encontrados se obtiene la id y el tipo de usuario para poder iniciar sesión ya sea en el rol administrador o rol usuario.

```
require 'database.php';
if (!empty($_POST['correo']) && !empty($_POST['password'])) {
    $records = $conn->prepare('SELECT * FROM usuarios WHERE correo = :correo AND password = :password');
    $records->bindParam(':correo', $_POST['correo']);
    $records->bindParam(':password', $_POST['password']);
    $records->execute();
    $results = $records->fetch(PDO::FETCH_ASSOC);

    if (count($results) > 0) {
        $_SESSION['user_id'] = $results['Id'];
        $_SESSION['rol'] = $results['tipo'];
        switch ($_SESSION['rol']) {
            case "admin":
                header('Location: ../paginas/admin/index2.php');
                break;

            case "cliente":
                header('Location: ../paginas/cliente/index3.php');
                break;

            default:
        }
    }
}
```

```

    } else {
        $message = 'Sorry, those credentials do not match';
    }

```

3.- conexión para poder insertar y consultar de la tabla usuarios

Esta conexión se encuentra en la carpeta "Modelo", se llama conexión.php la cual trabaja con usuarios.php que trabaja con una clase llamada Usuario y declara dos métodos el primer método es para consultar y el segundo método es para poder ingresar valores a la tabla usuarios. En el trabajadorController lo llame así por que va a estar tanto el ingresar y consultar de la lista de usuarios y también están los trabajadores del terminal con rol Admin, se llamara a usuario.php y la clase Usuario para poder ingresar de valores los dos métodos antes mencionados, en este caso usamos un switch.

Esta conexión la usamos en el registro de usuarios que esta en la carpeta vista/Usuario new.html

Modelo de menú de opciones.

El modelo se trabajó con estilos Bootstrap.css, se encuentra en todas las paginas.

```

<link rel="stylesheet" href="../../css/bootstrap.css">

```

Modelo de los tickets

```

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" />

```

Rutas

Para las rutas usamos el iframe de Google maps

```

<iframe src="https://www.google.com/maps/embed?pb=!1m14!1m12!1m3!1d63795.66151378121!2d-79.93978692089843!3d-2.0643840000000058!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!5e0!3m2!1ses!2sec!4v1612419979749!5m2!1ses!2sec" width="1100" height="600" frameborder="0" style="border:0;" allowfullscreen="" aria-hidden="false" tabindex="0"></iframe>

```

Uso de Terminalapp

Página de inicio de sesión

**Por favor inicie sesion para
navegar en nuestra app**

Iniciar Sesion

Tenemos la opción de ingresar los datos registrados o crear un nuevo usuario

Bienvenido al terminal terrestre de Guayaquil

Inicie Sesion

Ingrese su correo

Ingrese su contraseña

submit

Registro

En el registro el usuario tendrá que llenar todos los datos necesarios para poder crear su usuario en tal caso que se haya equivocado de selección anterior, cree un botón Login para que pueda retroceder.

Registro de Usuario

Nombre

Apellido

Telefono

Correo

Cedula

Usuario

Password

Guardar

Login

Una vez iniciado sesión el usuario estará en el inicio de la pagina en este caso e elegido de inicio al módulo noticias.

Terminal Terrestre

	Noticias	Bienvenido, samuel	Ticket	Rutas	Carro de compras	Boletería	
--	----------	--------------------	--------	-------	------------------	-----------	--

Bienvenido al terminal terrestre de Guayaquil



El módulo de ticket tenemos todos los tickets que se encuentran disponibles, podemos seleccionar la cantidad de ticket que necesitamos y agregar al carro.

Terminal Terrestre



Noticias

Bienvenido. samuel

Rutas

Carro de compras

Boleteria

Ticket

Destino

Pedro Carbo

Nombre de la Cooperativa

Pedro Carbo



Precio: \$ 2.00

Tiempo de viaje: 1h
30min

Agregar al carro

Destino

Pedro Carbo

Nombre de la Cooperativa

Piedad



Precio: \$ 2.00

Tiempo de viaje: 1h
40min

Agregar al carro

Destino

Pajan

Nombre de la Cooperativa

Pajan



Precio: \$ 4.00

Tiempo de viaje: 2h

Agregar al carro

Destino

Babahoyo

Destino

Villamil Playas

Destino

Montanita

En rutas tenemos el mapa

Terminal Terrestre

Noticias

Bienvenido. samuel

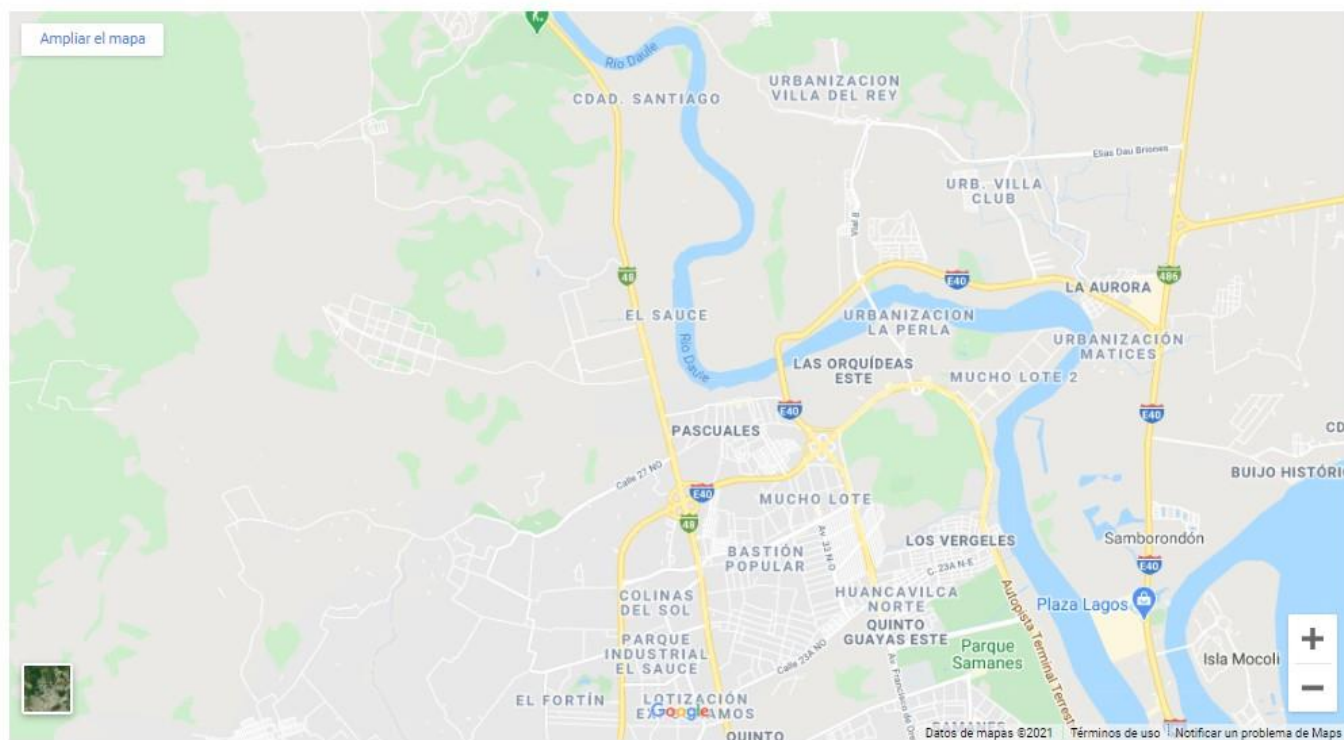
Ticket

Rutas

Carro de compras

Boletería

Mapa



Carro de compras – Tenemos la opción de eliminar el boleto

Terminal Terrestre

Ticket

Rutas



Boleteria

Carro de compras

Detalle de la Tickets

Destino	Cantidad	Precio	Total	
Pedro Carbo	4	\$ 2.00	\$ 8.00	Eliminar
Pedro Carbo	3	\$ 2.00	\$ 6.00	Eliminar
Pajan	4	\$ 4.00	\$ 16.00	Eliminar
		Total	\$ 30.00	

En boleterías podemos ver los estados actuales de las boleterías, si están habilitadas o Inactivas esta función va a ser de mucha ayuda a los pasajeros dado que algunas personas llegan a los terminales y las boleterías no están en funcionamiento, esto evitara que los usuarios lleguen por gusto o pierdan tiempo.

Terminal Terrestre



Estamos trabajando

<p>Nombre de la boleteria Pedro Carbo</p>  <p>Estado Habilitada Tiempo de viaje: 3h 40min</p>	<p>Nombre de la boleteria Piedad</p>  <p>Estado Inactiva Tiempo de viaje: 1h 60min</p>	<p>Nombre de la boleteria Pajan</p>  <p>Estado Habilitada Tiempo de viaje: 3h 50min</p>
<p>Nombre de la boleteria Quito</p>  <p>Estado Habilitada</p>	<p>Nombre de la boleteria Playas</p>  <p>Estado Cerrada</p>	<p>Nombre de la boleteria Monta^{ita}</p>  <p>Estado Habilitada</p>

En este modulo esta el fin de inicio de sesión una ves el cliente haya seleccionado esta opción, inmediatamente saldrá de la pagina



Código para poder ingresar el nevo ticket a la base de datos

Destino: Modelo/conexión.php

```
public function insertar_ticket($tabla,$datos)
{
    try {
        $this->conectar();
        $sql = "INSERT INTO $tabla(";
        foreach($datos as $clave=>$valor)
        {
            $sql .=$clave.", ";
        }
        $sql = substr ($sql, 0, strlen($sql) - 1).") VALUES(";
        foreach($datos as $clave=>$valor)
        {
            $sql .=":". $clave.", ";
        }
        $sql = substr ($sql, 0, strlen($sql) - 1).")";
        $stmt = $this->dbconn->prepare($sql);
        foreach($datos as $clave=>$valor)
        {$clave=':'.$clave;
            $stmt->bindValue($clave, $valor);
        }
    }
}
```

```

    }
    // execute the insert statement
    $stmt->execute();
    return "Ticket insertados...";
} catch (Exception $e) {
    $this->error= $e->getMessage();
    return "Error al insertar el ticket... ".$this->error;
}
}

```

Destino: Modelo/usuario.php

```

public function nuevo_ticket($datos)
{
    $conexion=new Conexion;
    $ticket=$conexion->insertar_ticket('ticket',$datos);
    return $ticket;
}

```

Destino: Controlador/TrabajadorController.php

```

case 'ingresar_ticket':
    $datos['name']=$_POST['name'];
    $datos['tiempo']=$_POST['tiempo'];
    $datos['cooperativa']=$_POST['cooperativa'];
    $datos['image']=$_POST['image'];
    $datos['price']=$_POST['price'];

    if($objUsuario->nuevo_ticket($datos))
    {
        echo "Ticket ingresado";
    }
    else
    {
        echo "Error al registrar Ticket".$objUsuario->
geterror();
    }
    break;

```

Interfaz del Administrador

Inicio: El inicio del Admin será el de las noticias

Terminal Terrestre

Noticias

Bienvenido Admin.
AdminMario

Ticket

Rutas

Ingresar Ticket

Bienvenido Admin



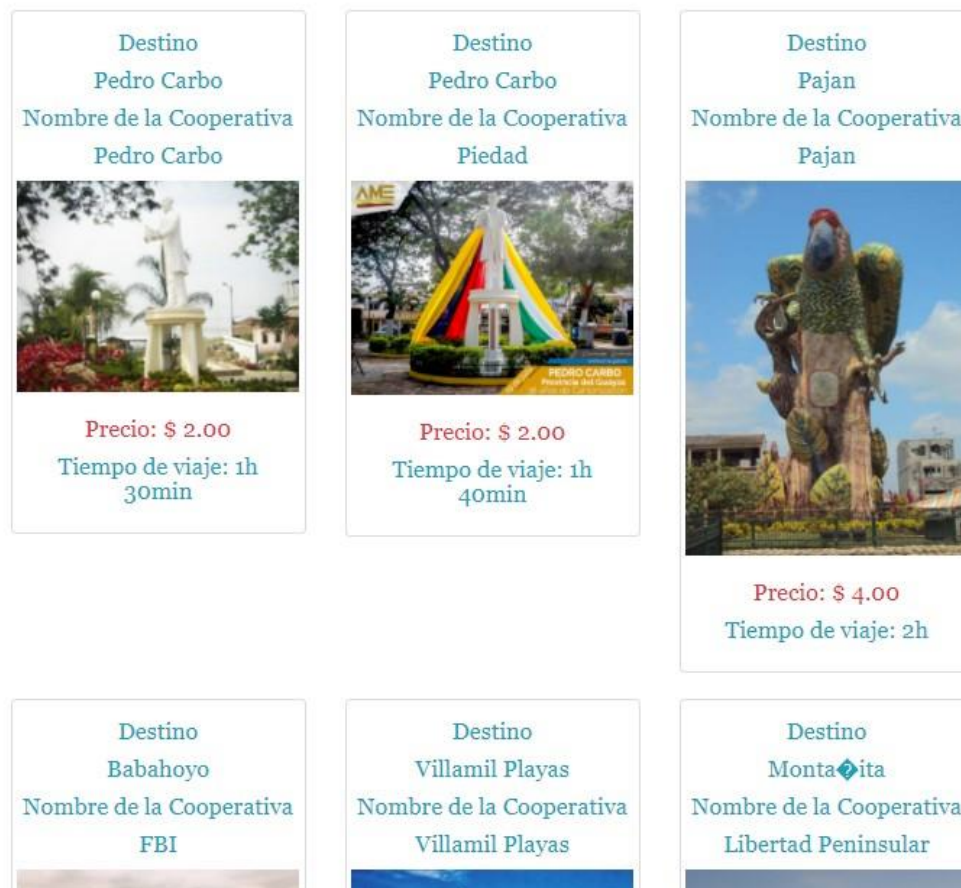
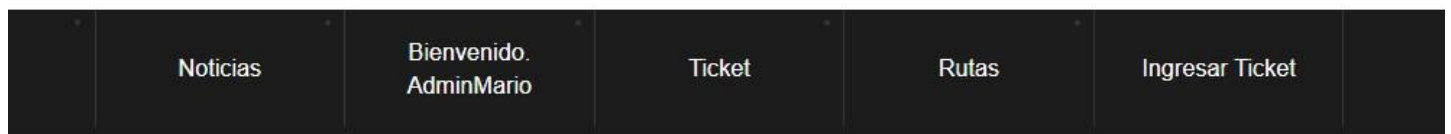
Termina

Para finalizar sesión el Admin tiene que dar clic en el icono.



El Admin puede ver los ticket que a creado

Terminal Terrestre



Tiene la opción del mapa disponible.

Terminal Terrestre

Noticias

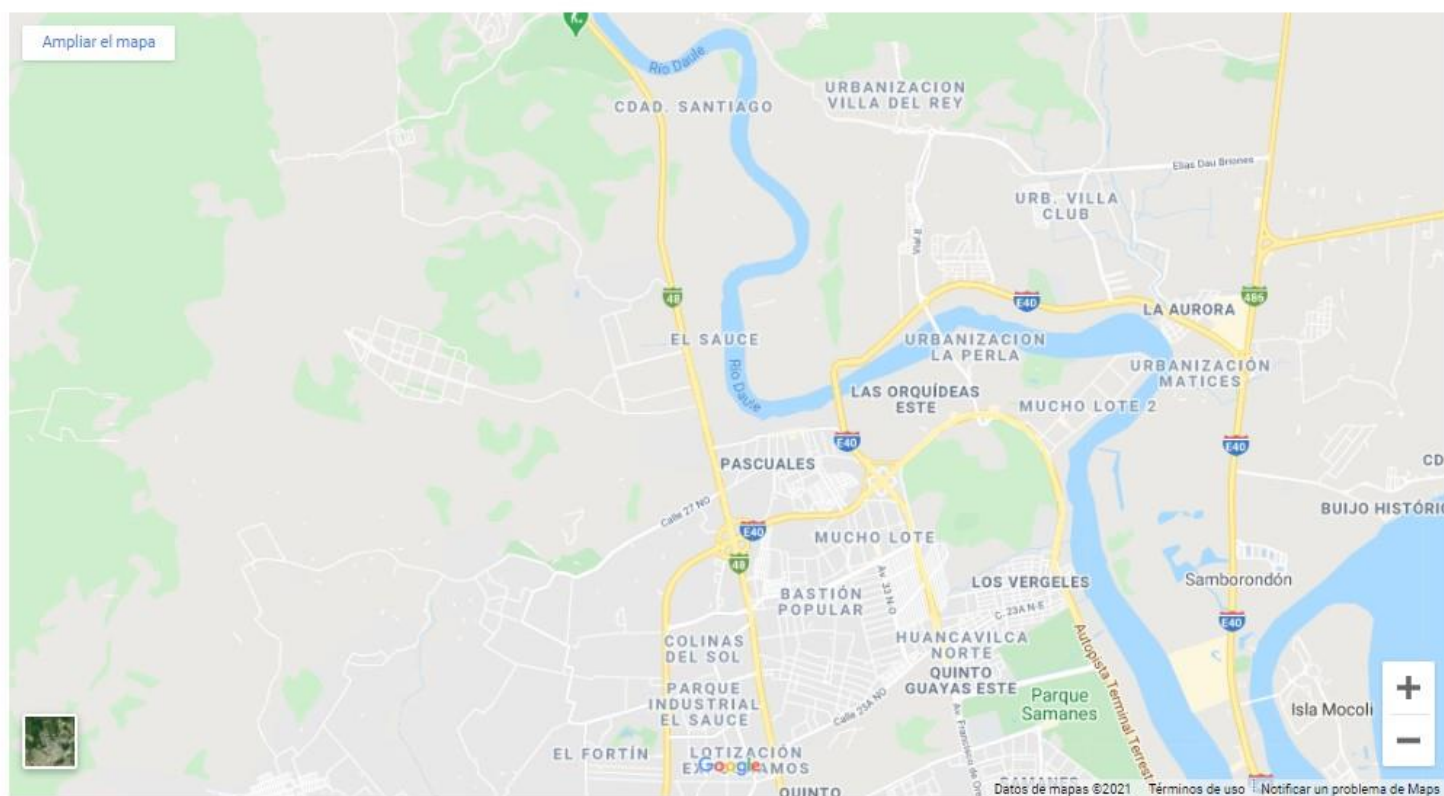
Bienvenido.
AdminMario

Ticket

Rutas

Ingresar Ticket

Mapa



También tiene el acceso para crear nuevos tickets.

Terminal Terrestre

	Noticias	Bienvenido Admin. AdminMario	Ticket	Rutas	Ingresar Ticket	
--	----------	---------------------------------	--------	-------	-----------------	--

Registro de Ticket

Nombre
Tiempo
Cooperativa
Imagen "Link de la imagen"
Precio
Guardar