



Manual de Sistema

Sistema de Reportes Sala CICENT

Introducción

El presente manual sirve de soporte y ayuda para los usuarios sobre información del desarrollo del sistema, para posibles mejoras, reparaciones o ampliación del sistema, incluye información acerca del lenguaje usado para el desarrollo del sistema, modelo de la base de datos que tiene el sistema, diccionario de datos del sistema, requerimientos para su instalación e información de los programas que se utilizaron para su desarrollo.

Información Técnica.

Para el desarrollo de las interfaces del sistema se usó el lenguaje de Programación HTML, PHP, JavaScript, Css y JQuery. Para la Base de datos se utilizó el Gestor de Base de datos MySQL y el framework para PHP llamado Bootstrap, y el servidor de alojamiento del sistema se eligió Apache.

HTML, siglas de **HyperText Markup Language** («lenguaje de marcado hipertextual»), hace referencia al lenguaje de marcado predominante para la elaboración de páginas Web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo, JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML. Para más información puede acceder a la página <http://www.w3schools.com/html/>

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo Web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. Para tener más información puede acceder a la página <http://www.php.net/>

JavaScript: es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript es una marca

registrada de la empresa Sun Microsystems, para más información puede acceder a la página <http://www.w3schools.com/js/>

Css: es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (también llamados "*documentos semánticos*"). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes. Para más información pueden acceder a la página <http://www.w3schools.com/css/default.asp>

Jquery: Jquery se encuentra escrito en JavaScript, un lenguaje de programación muy rico y expresivo.

El capítulo está orientado a personas sin experiencia en el lenguaje, abarcando conceptos básicos y problemas frecuentes que pueden presentarse al trabajar con el mismo. Por otro lado, la sección puede ser beneficiosa para quienes utilicen otros lenguajes de programación para entender las peculiaridades de JavaScript. Para más información pueden acceder a la página <http://www.w3schools.com/jquery/default.asp>

MySQL: es un sistema de gestión de bases de datos relacionales, multihilo y multiusuario con más de seis millones de instalaciones MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual. Para más información pueden acceder a la página <http://www.mysql.com/>

Bootstrap: es una colección de herramientas de software libre para la creación de sitios y aplicaciones web. Contiene plantillas de diseño basadas en HTML y CSS con tipografías, formularios, botones, gráficos, barras de navegación y demás componentes de interfaz, así como extensiones opcionales de JavaScript. Para más información pueden acceder a la página <http://twitter.github.io/bootstrap/index.html>

Apache: es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTP 1.3, pero más tarde fue reescrito por completo. Para más información pueden acceder a la página <http://httpd.apache.org/>

Wamp: Para el alojamiento del sistema se usó el programa **WampServer** (acrónimo formado por Windows, Apache, MySQL y PHP) es un entorno de desarrollo web con el que podrás instalar fácilmente un servidor

Apache son soporte PHP y gestión de bases de datos MySQL en tu computadora.

Se instala automáticamente y no requiere configuraciones avanzadas para su funcionamiento. Una vez instalado, deberás colocar tus sitios web en la carpeta “www” (ubicado por defecto en C:\wamp\www). Para acceder al servidor escribir la dirección <http://localhost> en cualquier navegador web: Firefox, Internet Explorer, Chrome, etc. Para más información pueden acceder a la página <http://www.wampserver.com/en/>

Características de WampServer:

Con este programa podrás:

- Pre visualizar sitios web localmente
- Instalar un servidor web en Windows
- Administrar configuraciones de servidores Apache
- Crear aplicaciones web
- Gestionar bases de datos MySQL
- Utilizar lenguaje PHP
- Ejecutar archivos .php localmente
- Realizar pruebas con sitios PHP antes de subirlos a Internet

Requerimientos de WampServer:

Configuraciones mínimas de software y hardware:

- Windows NT/98/Me/2000/2003/XP/Vista
- Privilegios de administrador

Para la edición de código HTM, PHP, Css, JavaScript y JQuery se usó el programa **NotePad++** que es un editor de texto y de código fuente libre con soporte para varios lenguajes de programación. De soporte nativo a Microsoft Windows.

Se parece al Bloc de notas en cuanto al hecho de que puede editar texto sin formato y de forma simple. No obstante, incluye opciones más avanzadas que pueden ser útiles para usuarios avanzados como desarrolladores y programadores. Para más información pueden acceder a la página <http://notepad-plus-plus.org/>

Modelamiento de la Base de Datos

Para el modelamiento de datos se utilizó el Programa MySQL Workbench que es una herramienta visual de diseño de bases de datos que integra desarrollo de software, Administración de bases de datos, diseño de bases de datos, creación y mantenimiento para el sistema de base de datos MySQL. Para más información pueden acceder a la página <http://www.mysql.com/products/workbench/>

Diagrama de base de datos relacional para un sistema de gestión de viajes y vehículos. El diagrama muestra 15 tablas con sus respectivos campos, tipos de datos e índices. Las relaciones entre las tablas se representan mediante líneas con guiones de cardinalidad.

Tablas y sus campos:

- usuario**: ID_Usuario INT(11), Nombre VARCHAR(50), Apellido VARCHAR(50), Telefono VARCHAR(50), Email VARCHAR(50), Login VARCHAR(50), Clave VARCHAR(50), Rol_ID_Rol INT(11), Departamento_ID_Dpto INT(11), Pregunta VARCHAR(50), Respuesta VARCHAR(50). *Indexes*
- rol**: ID_Rol INT(11), Nombre_Rol VARCHAR(50). *Indexes*
- departamento**: ID_Dpto INT(11), Nombre_Dpto VARCHAR(50), Sede VARCHAR(50). *Indexes*
- conductor**: ID_Conductor INT(11), Nombre VARCHAR(50), Apellido VARCHAR(20), Telefono VARCHAR(20), foto VARCHAR(200). *Indexes*
- km_cond**: Fecha_Km DATE, Km_Cond VARCHAR(45), conductor_ID_Conductor INT(11), Num_Desp INT(11). *Indexes*
- viajes**: Fecha_Viaje DATE, Hora_Salida TIME, Num_Despacho INT(11), Usuario_ID_Usuario INT(11), CH_CD_Conductor_ID_Conductor INT(11), CH_CD_Chuto_Placal_Chuto VARCHAR(50), Cliente_ID_Cli INT(11), id_conductor VARCHAR(20), nombre_cliente VARCHAR(100), tipo INT(11). *Indexes*
- ch_cd**: Conductor_ID_Conductor INT(11), Chuto_Placal_Chuto VARCHAR(50). *Indexes*
- viajes_Velocidad**: Fecha_Viaje DATE, Hora_Salida TIME, Velocidad INT(11), Ubicacion VARCHAR(200), Latitud VARCHAR(20), Longitud VARCHAR(20), id_conductor VARCHAR(20), tipo INT(11), ch_cd_Conductor_ID_Conductor INT(11), ch_cd_Chuto_Placal_Chuto VARCHAR(50). *Indexes*
- chuto**: Placal_Chuto VARCHAR(50), Serial_Carroceria VARCHAR(50), Serial_Motor VARCHAR(50), Modelo VARCHAR(50), foto_chuto VARCHAR(250), Km_recorridos DOUBLE, Observaciones VARCHAR(250), Unidad VARCHAR(10), Marca_Chuto_ID_Marca INT(11), Condicion_Chuto_ID_Condicion INT(11), Flota_ID_Flota INT(11), Ubicacion_ID_Ubicacion INT(11), Estatus_ID_Estatus INT(11), num_mant INT(11). *Indexes*
- flota**: ID_Flota INT(11), Nombre_Flota VARCHAR(50). *Indexes*
- condicion_chuto**: ID_Condicion INT(11), Nombre_Cond VARCHAR(20). *Indexes*
- estatus**: ID_Estatus INT(11), Nombre_Estatus VARCHAR(100). *Indexes*
- cliente**: ID_Cli INT(11), Nombre_Cli VARCHAR(50), RIF_Cli VARCHAR(50), Razon_Social_Cli VARCHAR(50), Direccion_Cli VARCHAR(100), Telefono_Cli VARCHAR(20), Distancia_Cli DOUBLE, Tiempo_Cli DOUBLE, Tipo_Cliente_ID_Tipo INT(11), Distrito_ID_Dtto INT(11), Mayorista_ID_Mayorista INT(11), Estado_ID_Edo INT(11), Zona_Com_ID_Zona INT(11), Ciudad_ID_Gudad INT(11), Municipio_ID_Mcpio INT(11), Sede_ID_Sede INT(11), Lat VARCHAR(20), Lon VARCHAR(20). *Indexes*
- distrito**: ID_Dtto INT(11), Nombre_Dtto VARCHAR(50). *Indexes*
- tipo_cliente**: ID_Tipo INT(11), Nombre_Tipo VARCHAR(50). *Indexes*
- municipio**: ID_Mcpio INT(11), Nombre_Mcpio VARCHAR(50). *Indexes*
- ubicacion**: ID_Ubicacion INT(11), Nombre_Ubicacion VARCHAR(50). *Indexes*
- ciudad**: ID_Gudad INT(11), Nombre_Gudad VARCHAR(50). *Indexes*
- estado**: ID_Edo INT(11), Nombre_Edo VARCHAR(50). *Indexes*
- marca_chuto**: ID_Marca INT(11), Nombre_Marca VARCHAR(50). *Indexes*
- zona_com**: ID_Zona INT(11), Nombre_Zona VARCHAR(50). *Indexes*
- sede**: ID_Sede INT(11), Nombre_Sede VARCHAR(50). *Indexes*
- mayorista**: ID_Mayorista INT(11), Nombre_May VARCHAR(100). *Indexes*

Relaciones:

- usuario (1) a rol (M)
- usuario (1) a departamento (M)
- usuario (1) a viajes (M)
- rol (1) a departamento (M)
- conductor (1) a km_cond (M)
- conductor (1) a viajes (M)
- conductor (1) a ch_cd (M)
- km_cond (1) a viajes_Velocidad (M)
- viajes (1) a viajes_Velocidad (M)
- viajes (1) a ch_cd (M)
- viajes (1) a chuto (M)
- viajes (1) a cliente (M)
- viajes (1) a ubicacion (M)
- viajes (1) a ciudad (M)
- viajes (1) a estado (M)
- viajes (1) a marca_chuto (M)
- viajes (1) a zona_com (M)
- viajes (1) a sede (M)
- viajes (1) a mayorista (M)
- ch_cd (1) a viajes_Velocidad (M)
- ch_cd (1) a chuto (M)
- chuto (1) a flota (M)
- chuto (1) a condicion_chuto (M)
- chuto (1) a estatus (M)
- chuto (1) a ubicacion (M)
- chuto (1) a ciudad (M)
- chuto (1) a estado (M)
- chuto (1) a marca_chuto (M)
- chuto (1) a zona_com (M)
- chuto (1) a sede (M)
- chuto (1) a mayorista (M)
- flota (1) a condicion_chuto (M)
- flota (1) a estatus (M)
- flota (1) a ubicacion (M)
- flota (1) a ciudad (M)
- flota (1) a estado (M)
- flota (1) a marca_chuto (M)
- flota (1) a zona_com (M)
- flota (1) a sede (M)
- flota (1) a mayorista (M)
- condicion_chuto (1) a estatus (M)
- condicion_chuto (1) a ubicacion (M)
- condicion_chuto (1) a ciudad (M)
- condicion_chuto (1) a estado (M)
- condicion_chuto (1) a marca_chuto (M)
- condicion_chuto (1) a zona_com (M)
- condicion_chuto (1) a sede (M)
- condicion_chuto (1) a mayorista (M)
- estatus (1) a ubicacion (M)
- estatus (1) a ciudad (M)
- estatus (1) a estado (M)
- estatus (1) a marca_chuto (M)
- estatus (1) a zona_com (M)
- estatus (1) a sede (M)
- estatus (1) a mayorista (M)
- ubicacion (1) a ciudad (M)
- ubicacion (1) a estado (M)
- ubicacion (1) a marca_chuto (M)
- ubicacion (1) a zona_com (M)
- ubicacion (1) a sede (M)
- ubicacion (1) a mayorista (M)
- ciudad (1) a estado (M)
- ciudad (1) a marca_chuto (M)
- ciudad (1) a zona_com (M)
- ciudad (1) a sede (M)
- ciudad (1) a mayorista (M)
- estado (1) a marca_chuto (M)
- estado (1) a zona_com (M)
- estado (1) a sede (M)
- estado (1) a mayorista (M)
- marca_chuto (1) a zona_com (M)
- marca_chuto (1) a sede (M)
- marca_chuto (1) a mayorista (M)
- zona_com (1) a sede (M)
- zona_com (1) a mayorista (M)
- sede (1) a mayorista (M)

El Script asociado a ese modelo es el siguiente:

```
-- phpMyAdmin SQL Dump
-- version 3.3.9
-- http://www.phpmyadmin.net
--
-- Servidor: localhost
-- Tiempo de generaci3n: 06-06-2013 a las 13:30:30
-- Versi3n del servidor: 5.5.8
-- Versi3n de PHP: 5.3.5

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

/*!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 utf8 */;

--
-- Base de datos: `gtr`
--
-----
--
-- Estructura de tabla para la tabla `chuto`
--
CREATE TABLE IF NOT EXISTS `chuto` (
  `Placal_Chuto` varchar(50) NOT NULL,
  `Serial_Carroceria` varchar(50) DEFAULT 'NULL',
  `Serial_Motor` varchar(50) DEFAULT 'NULL',
  `Modelo` varchar(50) DEFAULT 'NULL',
  `foto_chuto` varchar(250) DEFAULT 'NULL',
  `Km_recorridos` double DEFAULT NULL,
  `Observaciones` varchar(250) DEFAULT NULL,
  `Unidad` varchar(10) DEFAULT NULL,
  `Marca_Chuto_ID_Marca` int(11) NOT NULL,
  `Condicion_Chuto_ID_Condicion` int(11) NOT NULL,
  `Flota_ID_Flota` int(11) NOT NULL,
  `Ubicacion_ID_Ubicacion` int(11) NOT NULL,
  `Estatus_ID_Estatus` int(11) NOT NULL,
  `num_mant` int(11) DEFAULT NULL,
  PRIMARY KEY (`Placal_Chuto`),
  KEY `Chuto_Condicion_Chuto_FK` (`Condicion_Chuto_ID_Condicion`),
  KEY `Chuto_Estatus_FK` (`Estatus_ID_Estatus`),
  KEY `Chuto_Flota_FK` (`Flota_ID_Flota`),
  KEY `Chuto_Marca_Chuto_FK` (`Marca_Chuto_ID_Marca`),
  KEY `Chuto_Ubicacion_FK` (`Ubicacion_ID_Ubicacion`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `ch_cd`
--
```



```
CREATE TABLE IF NOT EXISTS `ch_cd` (  
  `Conductor_ID_Conductor` int(11) NOT NULL,  
  `Chuto_Placal_Chuto` varchar(50) NOT NULL,  
  PRIMARY KEY (`Conductor_ID_Conductor`,`Chuto_Placal_Chuto`),  
  KEY `CH_CD_Chuto_FK` (`Chuto_Placal_Chuto`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
```

```
-- Estructura de tabla para la tabla `ciudad`
```

```
--
```

```
CREATE TABLE IF NOT EXISTS `ciudad` (  
  `ID_Ciudad` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Ciudad` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Ciudad`),  
  UNIQUE KEY `Nombre_Ciudad` (`Nombre_Ciudad`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=45 ;
```

```
--
```

```
-- Estructura de tabla para la tabla `cliente`
```

```
--
```

```
CREATE TABLE IF NOT EXISTS `cliente` (  
  `ID_Cli` int(11) NOT NULL,  
  `Nombre_Cli` varchar(50) DEFAULT 'NULL',  
  `RIF_Cli` varchar(50) DEFAULT 'NULL',  
  `Razon_Social_Cli` varchar(50) DEFAULT 'NULL',  
  `Direccion_Cli` varchar(100) DEFAULT 'NULL',  
  `Telefono_Cli` varchar(20) DEFAULT 'NULL',  
  `Distancia_Cli` double DEFAULT NULL,  
  `Tiempo_Cli` double DEFAULT NULL,  
  `Tipo_Cliente_ID_Tipo` int(11) DEFAULT NULL,  
  `Distrito_ID_Dtto` int(11) DEFAULT NULL,  
  `Mayorista_ID_Mayorista` int(11) DEFAULT NULL,  
  `Estado_ID_Edo` int(11) DEFAULT NULL,  
  `Zona_Com_ID_Zona` int(11) DEFAULT NULL,  
  `Ciudad_ID_Ciudad` int(11) DEFAULT NULL,  
  `Municipio_ID_Mcpio` int(11) DEFAULT NULL,  
  `Sede_ID_Sede` int(11) DEFAULT NULL,  
  `Lat` varchar(20) DEFAULT NULL,  
  `Lon` varchar(20) DEFAULT NULL,  
  PRIMARY KEY (`ID_Cli`),  
  KEY `Cliente_Ciudad_FK` (`Ciudad_ID_Ciudad`),  
  KEY `Cliente_Distrito_FK` (`Distrito_ID_Dtto`),  
  KEY `Cliente_Estado_FK` (`Estado_ID_Edo`),  
  KEY `Cliente_Mayorista_FK` (`Mayorista_ID_Mayorista`),  
  KEY `Cliente_Municipio_FK` (`Municipio_ID_Mcpio`),  
  KEY `Cliente_Sede_FK` (`Sede_ID_Sede`),  
  KEY `Cliente_Tipo_Cliente_FK` (`Tipo_Cliente_ID_Tipo`),  
  KEY `Cliente_Zona_Com_FK` (`Zona_Com_ID_Zona`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
```

```
-- Estructura de tabla para la tabla `condicion_chuto`
```

```
--
```

```
CREATE TABLE IF NOT EXISTS `condicion_chuto` (  
  `ID_Condicion` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Cond` varchar(20) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Condicion`),  
  UNIQUE KEY `Nombre_Cond` (`Nombre_Cond`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;  
  
--  
-- Estructura de tabla para la tabla `conductor`  
--  
  
CREATE TABLE IF NOT EXISTS `conductor` (  
  `ID_Conductor` int(11) NOT NULL,  
  `Nombre` varchar(50) DEFAULT 'NULL',  
  `Apellido` varchar(20) DEFAULT 'NULL',  
  `Telefono` varchar(20) DEFAULT 'NULL',  
  `foto` varchar(200) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Conductor`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
--  
-- Estructura de tabla para la tabla `departamento`  
--  
  
CREATE TABLE IF NOT EXISTS `departamento` (  
  `ID_Dpto` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Dpto` varchar(50) DEFAULT 'NULL',  
  `Sede` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Dpto`),  
  UNIQUE KEY `Nombre_Dpto` (`Nombre_Dpto`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;  
  
--  
-- Estructura de tabla para la tabla `distrito`  
--  
  
CREATE TABLE IF NOT EXISTS `distrito` (  
  `ID_Dtto` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Dtto` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Dtto`),  
  UNIQUE KEY `Nombre_Dtto` (`Nombre_Dtto`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;  
  
--  
-- Estructura de tabla para la tabla `estado`  
--  
  
CREATE TABLE IF NOT EXISTS `estado` (  
  `ID_Edo` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Edo` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Edo`),  
  UNIQUE KEY `Nombre_Edo` (`Nombre_Edo`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=25 ;  
  
--  
-- Estructura de tabla para la tabla `estatus`  
--
```



```
CREATE TABLE IF NOT EXISTS `estatus` (  
  `ID_Estatus` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Estatus` varchar(100) DEFAULT NULL,  
  PRIMARY KEY (`ID_Estatus`),  
  UNIQUE KEY `Nombre_Estatus` (`Nombre_Estatus`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;  
  
--  
-- Estructura de tabla para la tabla `flota`  
--  
  
CREATE TABLE IF NOT EXISTS `flota` (  
  `ID_Flota` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Flota` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Flota`),  
  UNIQUE KEY `Nombre_Flota` (`Nombre_Flota`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;  
  
--  
-- Estructura de tabla para la tabla `km_cond`  
--  
  
CREATE TABLE IF NOT EXISTS `km_cond` (  
  `Fecha_km` date NOT NULL,  
  `Km_Cond` varchar(45) DEFAULT NULL,  
  `conductor_ID_Conductor` int(11) NOT NULL,  
  `Num_Desp` int(11) NOT NULL,  
  PRIMARY KEY (`Num_Desp`),  
  KEY `fk_Km_Cond_conductor1_idx` (`conductor_ID_Conductor`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
--  
-- Estructura de tabla para la tabla `marca_chuto`  
--  
  
CREATE TABLE IF NOT EXISTS `marca_chuto` (  
  `ID_Marca` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Marca` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Marca`),  
  UNIQUE KEY `Nombre_Marca` (`Nombre_Marca`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;  
  
--  
-- Estructura de tabla para la tabla `mayorista`  
--  
  
CREATE TABLE IF NOT EXISTS `mayorista` (  
  `ID_Mayorista` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_May` varchar(100) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Mayorista`),  
  UNIQUE KEY `Nombre_May` (`Nombre_May`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=13 ;  
  
--  
-- Estructura de tabla para la tabla `municipio`  
--  
  
CREATE TABLE IF NOT EXISTS `municipio` (  
  `ID_Mcpio` int(11) NOT NULL AUTO_INCREMENT,
```

```
`Nombre_Mcpio` varchar(50) DEFAULT 'NULL',  
PRIMARY KEY (`ID_Mcpio`),  
UNIQUE KEY `Nombre_Mcpio` (`Nombre_Mcpio`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=31 ;
```

```
--  
-- Estructura de tabla para la tabla `rol`  
--
```

```
CREATE TABLE IF NOT EXISTS `rol` (  
  `ID_Rol` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Rol` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Rol`),  
  UNIQUE KEY `Nombre_Rol` (`Nombre_Rol`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

```
--  
-- Estructura de tabla para la tabla `sede`  
--
```

```
CREATE TABLE IF NOT EXISTS `sede` (  
  `ID_Sede` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Sede` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Sede`),  
  UNIQUE KEY `Nombre_Sede` (`Nombre_Sede`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;
```

```
--  
-- Estructura de tabla para la tabla `tipo`  
--
```

```
CREATE TABLE IF NOT EXISTS `tipo` (  
  `Id_Tv` int(11) NOT NULL,  
  PRIMARY KEY (`Id_Tv`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Estructura de tabla para la tabla `tipo_cliente`  
--
```

```
CREATE TABLE IF NOT EXISTS `tipo_cliente` (  
  `ID_Tipo` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Tipo` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Tipo`),  
  UNIQUE KEY `Nombre_Tipo` (`Nombre_Tipo`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

```
--  
-- Estructura de tabla para la tabla `ubicacion`  
--
```

```
CREATE TABLE IF NOT EXISTS `ubicacion` (  
  `ID_Ubicacion` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_Ubicacion` varchar(50) DEFAULT NULL,  
  PRIMARY KEY (`ID_Ubicacion`),  
  UNIQUE KEY `Nombre_Ubicacion` (`Nombre_Ubicacion`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;

--
-- Estructura de tabla para la tabla `usuario`
--

CREATE TABLE IF NOT EXISTS `usuario` (
  `ID_Usuario` int(11) NOT NULL,
  `Nombre` varchar(50) DEFAULT 'NULL',
  `Apellido` varchar(50) DEFAULT 'NULL',
  `Telefono` varchar(50) DEFAULT 'NULL',
  `Email` varchar(50) DEFAULT 'NULL',
  `Login` varchar(50) DEFAULT 'NULL',
  `Clave` varchar(50) DEFAULT 'NULL',
  `Rol_ID_Rol` int(11) NOT NULL,
  `Departamento_ID_Dpto` int(11) NOT NULL,
  `Pregunta` varchar(50) DEFAULT 'NULL',
  `Respuesta` varchar(50) DEFAULT 'NULL',
  PRIMARY KEY (`ID_Usuario`),
  KEY `Usuario_Departamento_FK` (`Departamento_ID_Dpto`),
  KEY `Usuario_Rol_FK` (`Rol_ID_Rol`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `viajes`
--

CREATE TABLE IF NOT EXISTS `viajes` (
  `Fecha_Viaje` date NOT NULL,
  `Hora_Salida` time NOT NULL,
  `Tipo_Id_Tv` int(11) DEFAULT NULL,
  `Num_Despacho` int(11) NOT NULL DEFAULT '0',
  `Velocidad` int(11) DEFAULT NULL,
  `Ubicacion` varchar(200) DEFAULT NULL,
  `Latitud` varchar(20) DEFAULT NULL,
  `Longitud` varchar(20) DEFAULT NULL,
  `Usuario_ID_Usuario` int(11) DEFAULT NULL,
  `CH_CD_Conductor_ID_Conductor` int(11) DEFAULT NULL,
  `CH_CD_Chuto_Placal_Chuto` varchar(50) DEFAULT NULL,
  `Cliente_ID_Cli` int(11) DEFAULT NULL,
  `id_conductor` varchar(20) DEFAULT NULL,
  `nombre_cliente` varchar(100) DEFAULT NULL,
  `tipo` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`Num_Despacho`,`tipo`),
  KEY `Viajes_CH_CD_FK`
  (`CH_CD_Conductor_ID_Conductor`,`CH_CD_Chuto_Placal_Chuto`),
  KEY `Viajes_Cliente_FK` (`Cliente_ID_Cli`),
  KEY `Viajes_Tipo_FK` (`Tipo_Id_Tv`),
  KEY `Viajes_Usuario_FK` (`Usuario_ID_Usuario`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Estructura de tabla para la tabla `viajes2`
--

CREATE TABLE IF NOT EXISTS `viajes2` (
  `Fecha_Viaje` date NOT NULL,
  `Hora_Salida` time NOT NULL,
  `Velocidad` int(11) DEFAULT NULL,
```

```
`Ubicacion` varchar(200) COLLATE utf8_unicode_ci DEFAULT NULL,  
`Latitud` varchar(20) COLLATE utf8_unicode_ci DEFAULT NULL,  
`Longitud` varchar(20) COLLATE utf8_unicode_ci DEFAULT NULL,  
`CH_CD_Conductor_ID_Conductor` int(11) DEFAULT NULL,  
`CH_CD_Chuto_Placal_Chuto` varchar(50) COLLATE utf8_unicode_ci DEFAULT  
NULL,  
`id_conductor` varchar(20) COLLATE utf8_unicode_ci DEFAULT NULL,  
`tipo` int(11) NOT NULL DEFAULT '0',  
KEY `Viajes2_CH_CD_FK`  
(`CH_CD_Conductor_ID_Conductor`,`CH_CD_Chuto_Placal_Chuto`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;  
  
--  
-- Estructura de tabla para la tabla `zona_com`  
--  
CREATE TABLE IF NOT EXISTS `zona_com` (  
  `ID_Zona` int(11) NOT NULL AUTO_INCREMENT,  
  `Nombre_zona` varchar(50) DEFAULT 'NULL',  
  PRIMARY KEY (`ID_Zona`),  
  UNIQUE KEY `Nombre_zona` (`Nombre_zona`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;  
  
--  
-- Filtros para la tabla `chuto`  
--  
ALTER TABLE `chuto`  
  ADD CONSTRAINT `Chuto_Condicion_Chuto_FK` FOREIGN KEY  
(`Condicion_Chuto_ID_Condicion`) REFERENCES `condicion_chuto`  
(`ID_Condicion`),  
  ADD CONSTRAINT `Chuto_Estatus_FK` FOREIGN KEY (`Estatus_ID_Estatus`)  
REFERENCES `estatus` (`ID_Estatus`),  
  ADD CONSTRAINT `Chuto_Flota_FK` FOREIGN KEY (`Flota_ID_Flota`) REFERENCES  
`flota` (`ID_Flota`),  
  ADD CONSTRAINT `Chuto_Marca_Chuto_FK` FOREIGN KEY (`Marca_Chuto_ID_Marca`)  
REFERENCES `marca_chuto` (`ID_Marca`),  
  ADD CONSTRAINT `Chuto_Ubicacion_FK` FOREIGN KEY (`Ubicacion_ID_Ubicacion`)  
REFERENCES `ubicacion` (`ID_Ubicacion`);  
  
--  
-- Filtros para la tabla `ch_cd`  
--  
ALTER TABLE `ch_cd`  
  ADD CONSTRAINT `CH_CD_Chuto_FK` FOREIGN KEY (`Chuto_Placal_Chuto`)  
REFERENCES `chuto` (`Placal_Chuto`),  
  ADD CONSTRAINT `CH_CD_Conductor_FK` FOREIGN KEY (`Conductor_ID_Conductor`)  
REFERENCES `conductor` (`ID_Conductor`);  
  
--  
-- Filtros para la tabla `cliente`  
--  
ALTER TABLE `cliente`  
  ADD CONSTRAINT `Cliente_Ciudad_FK` FOREIGN KEY (`Ciudad_ID_Ciudad`)  
REFERENCES `ciudad` (`ID_Ciudad`),  
  ADD CONSTRAINT `Cliente_Distrito_FK` FOREIGN KEY (`Distrito_ID_Dtto`)  
REFERENCES `distrito` (`ID_Dtto`),  
  ADD CONSTRAINT `Cliente_Estado_FK` FOREIGN KEY (`Estado_ID_Edo`) REFERENCES  
`estado` (`ID_Edo`),  
  ADD CONSTRAINT `Cliente_Mayorista_FK` FOREIGN KEY  
(`Mayorista_ID_Mayorista`) REFERENCES `mayorista` (`ID_Mayorista`),
```

```
ADD CONSTRAINT `Cliente_Municipio_FK` FOREIGN KEY (`Municipio_ID_Mcpio`)
REFERENCES `municipio` (`ID_Mcpio`),
ADD CONSTRAINT `Cliente_Sede_FK` FOREIGN KEY (`Sede_ID_Sede`) REFERENCES
`sede` (`ID_Sede`),
ADD CONSTRAINT `Cliente_Tipo_Cliente_FK` FOREIGN KEY
(`Tipo_Cliente_ID_Tipo`) REFERENCES `tipo_cliente` (`ID_Tipo`),
ADD CONSTRAINT `Cliente_Zona_Com_FK` FOREIGN KEY (`Zona_Com_ID_Zona`)
REFERENCES `zona_com` (`ID_Zona`);

--
-- Filtros para la tabla `km_cond`
--
ALTER TABLE `km_cond`
ADD CONSTRAINT `fk_Km_Cond_conductor1` FOREIGN KEY
(`conductor_ID_Conductor`) REFERENCES `conductor` (`ID_Conductor`) ON DELETE
NO ACTION ON UPDATE NO ACTION;

--
-- Filtros para la tabla `usuario`
--
ALTER TABLE `usuario`
ADD CONSTRAINT `Usuario_Departamento_FK` FOREIGN KEY
(`Departamento_ID_Dpto`) REFERENCES `departamento` (`ID_Dpto`),
ADD CONSTRAINT `Usuario_Rol_FK` FOREIGN KEY (`Rol_ID_Rol`) REFERENCES `rol`
(`ID_Rol`);

--
-- Filtros para la tabla `viajes`
--
ALTER TABLE `viajes`
ADD CONSTRAINT `Viajes2_CH_CD_FK` FOREIGN KEY
(`CH_CD_Conductor_ID_Conductor`, `CH_CD_Chuto_Placal_Chuto`) REFERENCES
`ch_cd` (`Conductor_ID_Conductor`, `Chuto_Placal_Chuto`),
ADD CONSTRAINT `Viajes_CH_CD_FK` FOREIGN KEY
(`CH_CD_Conductor_ID_Conductor`, `CH_CD_Chuto_Placal_Chuto`) REFERENCES
`ch_cd` (`Conductor_ID_Conductor`, `Chuto_Placal_Chuto`),
ADD CONSTRAINT `Viajes_Cliente_FK` FOREIGN KEY (`Cliente_ID_Cli`)
REFERENCES `cliente` (`ID_Cli`),
ADD CONSTRAINT `Viajes_Tipo_FK` FOREIGN KEY (`Tipo_Id_Tv`) REFERENCES
`tipo` (`Id_Tv`),
ADD CONSTRAINT `Viajes_Usuario_FK` FOREIGN KEY (`Usuario_ID_Usuario`)
REFERENCES `usuario` (`ID_Usuario`);
```

Diccionario de Datos

Tabla	Columna	Admite_Nulos	Tipo_Datos	Se_Autoincrementa	Llave	Comentario
cliente	Zona_Com_ID_Zona	YES	int(11)		MUL	Es la clave foránea asociada a la tabla Zona_Com
	Ciudad_ID_Ciudad	YES	int(11)		MUL	Es la clave foránea asociada a la tabla Ciudad
	Municipio_ID_Mcpio	YES	int(11)		MUL	Es la clave foránea asociada a la tabla Municipio
	Sede_ID_Sede	YES	int(11)		MUL	Es la clave foránea asociada a la tabla Sede
	Lat	YES	varchar(20)			Es el campo donde se guarda la latitud del cliente
	Lon	YES	varchar(20)			Es el campo donde se guarda la longitud del cliente
condicion_chuto	ID_Condicion	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código de condición
	Nombre_Cond	YES	varchar(20)		UNI	Es el campo donde se guarda el nombre de la condición
conductor	ID_Conductor	NO	int(11)		PRI	Es el campo donde se guarda la cedula del conductor
	Nombre	YES	varchar(50)			Es el campo donde se guarda el nombre del conductor
	Apellido	YES	varchar(20)			Es el campo donde se guarda el apellido del conductor
	Telefono	YES	varchar(20)			Es el campo donde se guarda el teléfono del conductor
	foto	YES	varchar(200)			Es el campo donde se guarda el foto del conductor
departame	ID_Dpto	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del

Tabla	Columna	Admite_Nulos	Tipo_Datos	Se_Autoincrementa	Llave	Comentario
nto						departamento
	Nombre_Dpto	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre del departamento
	Sede	YES	varchar(50)			Es el campo donde se guarda el nombre de la sede del departamento
distrito	ID_Dtto	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del distrito
	Nombre_Dtto	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre del distrito
estado	ID_Edo	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del estado
	Nombre_Edo	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre del estado
estatus	ID_Estatus	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del estatus
	Nombre_Estatus	YES	varchar(100)		UNI	Es el campo donde se guarda el nombre del estatus
flota	ID_Flota	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código de la flota
	Nombre_Flota	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre de la flota
km_cond	Fecha_km	NO	date			Es el campo donde se guarda la fecha en que el conductor hizo un viaje
	Km_Cond	YES	varchar(45)			Es el campo donde se guarda los km recorridos por un conductor
	conductor_ID_Conductor	NO	int(11)		MUL	Es el campo asociado a la cedula de un conductor
	Num_Desp	NO	int(11)		PRI	Es el campo donde se guarda el número de despacho de un viaje

Tabla	Columna	Admite_Nulos	Tipo_Datos	Se_Autoincrementa	Llave	Comentario
marca_chuto	ID_Marca	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código de la marca de chuto
	Nombre_Marca	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre de la marca de chuto
mayorista	ID_Mayorista	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del mayorista
	Nombre_May	YES	varchar(100)		UNI	Es el campo donde se guarda el nombre del mayorista
municipio	ID_Mcpio	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del municipio
	Nombre_Mcpio	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre del municipio
rol	ID_Rol	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del rol
	Nombre_Rol	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre del rol
sede	ID_Sede	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código de la sede
	Nombre_Sede	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre de la sede
tipo_cliente	ID_Tipo	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código del tipo de cliente
	Nombre_Tipo	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre del tipo de cliente
ubicacion	ID_Ubicacion	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código de la ubicacion

Tabla	Columna	Admite_Nulos	Tipo_Datos	Se_Autoincrementa	Llave	Comentario
	Nombre_Ubicacion	YES	varchar(50)		UNI	Es el campo donde se guarda el código de la ubicacion
usuario	ID_Usuario	NO	int(11)		PRI	Es el campo donde se guarda el código del usuario
	Nombre	YES	varchar(50)			Es el campo donde se guarda el nombre del usuario
	Apellido	YES	varchar(50)			Es el campo donde se guarda el apellido del usuario
	Telefono	YES	varchar(50)			Es el campo donde se guarda el telefono del usuario
	Email	YES	varchar(50)			Es el campo donde se guarda el email del usuario
	Login	YES	varchar(50)			Es el campo donde se guarda el login del usuario
	Clave	YES	varchar(50)			Es el campo donde se guarda la clave del usuario
	Rol_ID_Rol	NO	int(11)		MUL	Es el campo donde se guarda el rol del usuario
	Departamento_ID_Dpto	NO	int(11)		MUL	Es el campo donde se guarda el departamento del usuario
	Pregunta	YES	varchar(50)			Es el campo donde se guarda la pregunta de seguridad del usuario
	Respuesta	YES	varchar(50)			Es el campo donde se guarda la respuesta de seguridad del usuario
viajes	Fecha_Viaje	NO	date			Es el campo donde se guarda la fecha de salida del viaje
	Hora_Salida	NO	time			Es el campo donde se guarda la hora de salida del viaje
	Tipo_Id_Tv	YES	int(11)		MUL	Es el campo donde se guarda el tipo del viaje

Tabla	Columna	Admite_Nulos	Tipo_Datos	Se_Autoincrementa	Llave	Comentario
	Num_Despacho	NO	int(11)		PRI	Es el campo donde se guarda el número de despacho del viaje
	CH_CD_Conductor_ID_Conductor	YES	int(11)		MUL	Es el campo donde se guarda el conductor del viaje
	CH_CD_Chuto_Placal_Chuto	YES	varchar(50)			Es el campo donde se guarda el chuto del viaje
	Cliente_ID_Cli	YES	int(11)		MUL	Es el campo donde se guarda el cliente del viaje
	nombre_cliente	YES	varchar(100)			Es el campo donde se guarda el cliente del viaje
	tipo	NO	int(11)		PRI	Es el campo donde se guarda el tipo del viaje
viajes2	Fecha_Viaje	NO	date			Es el campo donde se guarda la fecha de salida del viaje
	Hora_Salida	NO	time			Es el campo donde se guarda la hora de salida del viaje
	Velocidad	YES	int(11)			Es el campo donde se guarda la velocidad del viaje
	Ubicacion	YES	varchar(200)			Es el campo donde se guarda la ubicacion del viaje
	Latitud	YES	varchar(20)			Es el campo donde se guarda la latitud del viaje
	Longitud	YES	varchar(20)			Es el campo donde se guarda la longitud del viaje
	CH_CD_Conductor_ID_Conductor	YES	int(11)		MUL	Es el campo donde se guarda el conductor del viaje
	CH_CD_Chuto_Placal_Chuto	YES	varchar(50)			Es el campo donde se guarda el chuto del viaje

Tabla	Columna	Admite_Nulos	Tipo_Datos	Se_Autoincrementa	Llave	Comentario
	id_conductor	YES	varchar(20)			Es el campo donde se guarda el conductor del viaje
	tipo	NO	int(11)			Es el campo donde se guarda el tipo del viaje
zona_com	ID_Zona	NO	int(11)	auto_increment	PRI	Es el campo donde se guarda el código de la zona comercial
	Nombre_zona	YES	varchar(50)		UNI	Es el campo donde se guarda el nombre de la zona comercial

