MANUAL TECNICO

APLICATIVO ORIENTADO A LA DISTRIBUIDORA DISTRIANDINA DEDICADA A SURTIR PAPELEIRAS

MARIA PAULA CASTRO TELLEZ

MARIA ALEJANDRA RUIZ GONZALEZ

LUIS MIGUEL TORRES RODRIGUEZ

FICHA 1193354



BOGOTA D.C

**CONTENIDO**

1. **INTRODUCCION**
2. **OBJETIVOS**

2.1 **GENERAL**

* 1. **ESPECIFICOS**

1. **DICCIONARIO DE DATOS**
2. **MAPA DE PROCESOS**

5. **DIAGRAMAS UML**

4.1 **DIAGRAMA DE CASO DE USOS**

4.2 **DIAGRAMA DE CLASES**

* 1. **DIAGRAMA RELACIONAL**
  2. **DIAGRAMA DE DISTRIBUCION**

6. **DESCRIPCION DE ERRORES**

7. **DESCRIPCION DE FORMULARIOS**

8. **DESCRIPCION DE BOTONES**

9. **HERRAMIENTAS PARA IMPLEMENTACION DEL APLICATIVO**

* 1. PHP
  2. MYSQL
  3. APACHE

10**. BASE DE DATOS**

1. **INTRODUCCION**

En este manual encontrara detalladamente documentación técnica acerca del aplicativo orientado a una tienda online.

El manual servirá de soporte para el desarrollador en el momento en el cual se requiera un mantenimiento o implementación de diferentes mejoras.

**2.OBJETIVOS**

**2.1 OBJETIVO GENERAL**

Brindar la información acerca de la creación e implementación del software a quien pueda interesar.

**2.2 OBJETIVOS ESPECIFICOS**

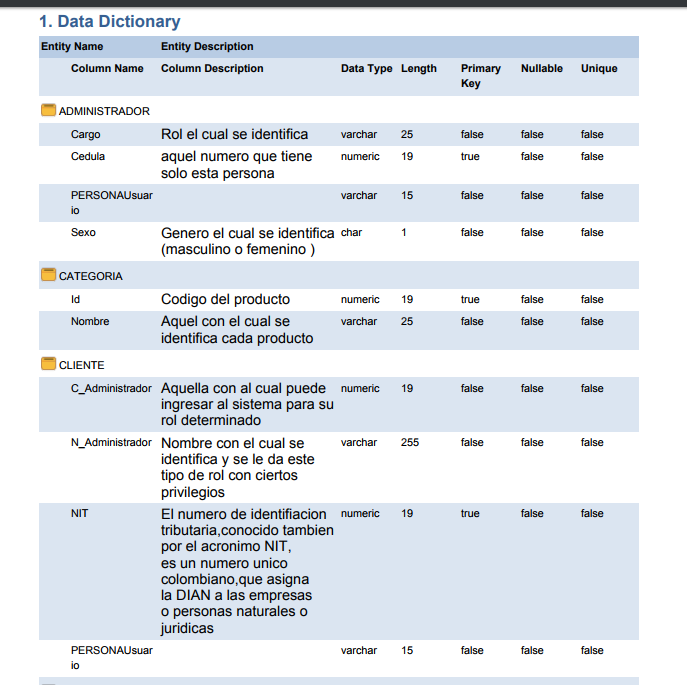
## Documentación de código

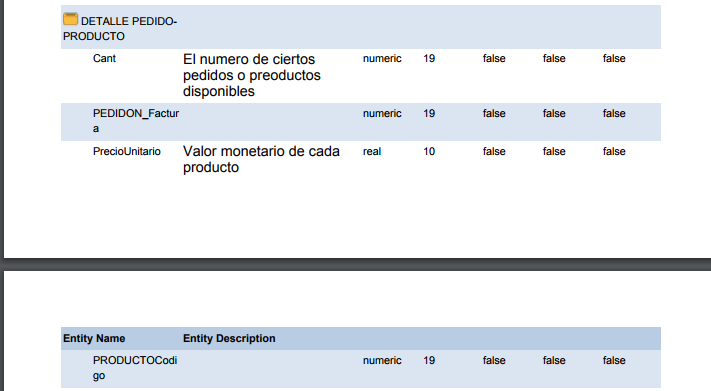
## Definir procedimientos del aplicativo

# Describir las herramientas usadas para el desarrollo del aplicativo

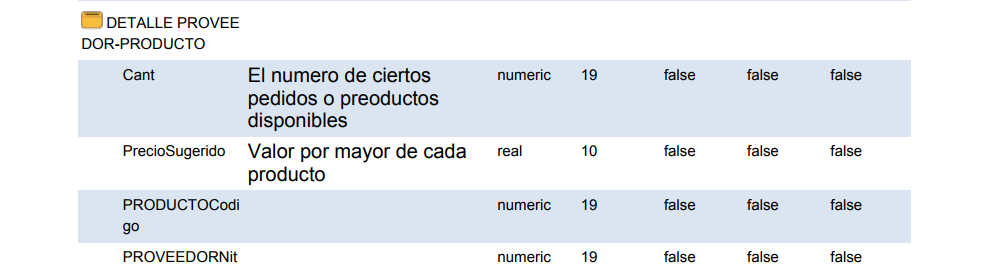
**3.DICCIONARIO DE DATOS**

**ADMINISTRADOR**

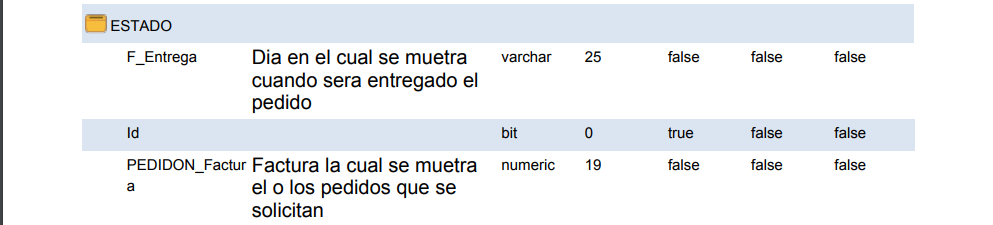
****

**DETALLE PEDIDO-PRODUCTO**

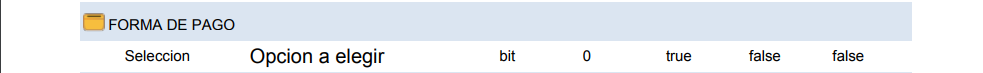
**DETALLE PROVEEDOR-PRODUCTO**

****

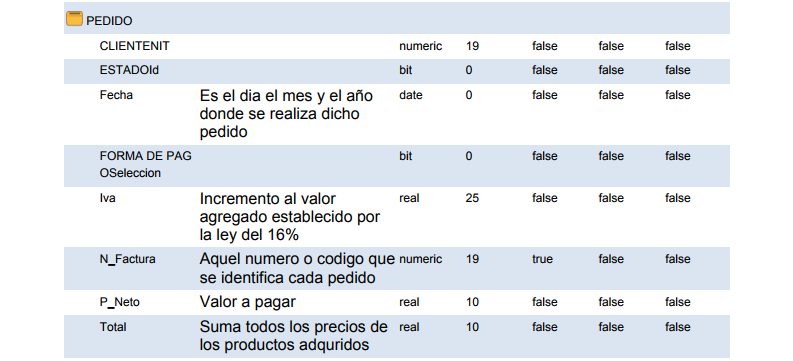
**ESTADO**

****

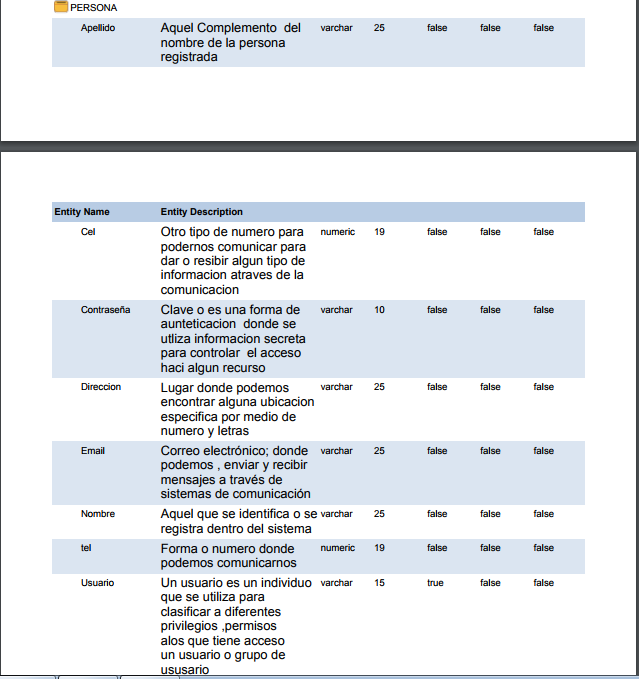
**FORMA DE PAGO**

****

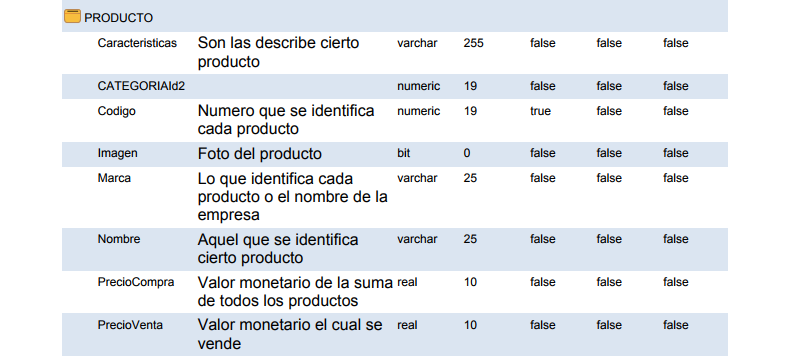
**PEDIDO**

****

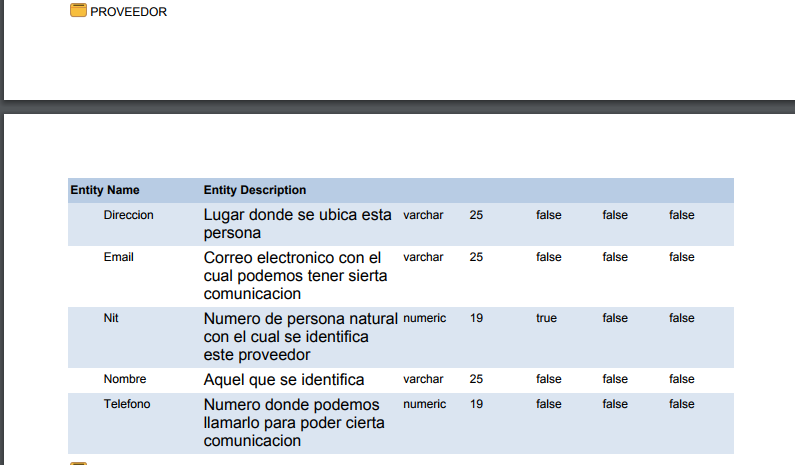
**PERSONA**

****

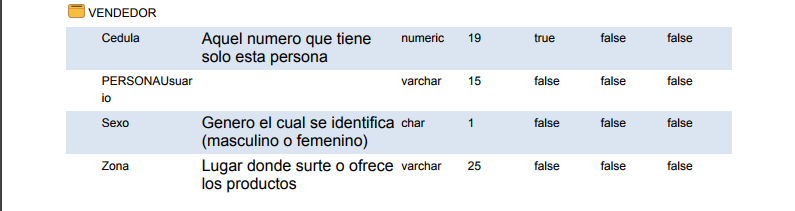
**PRODUCTO**

****

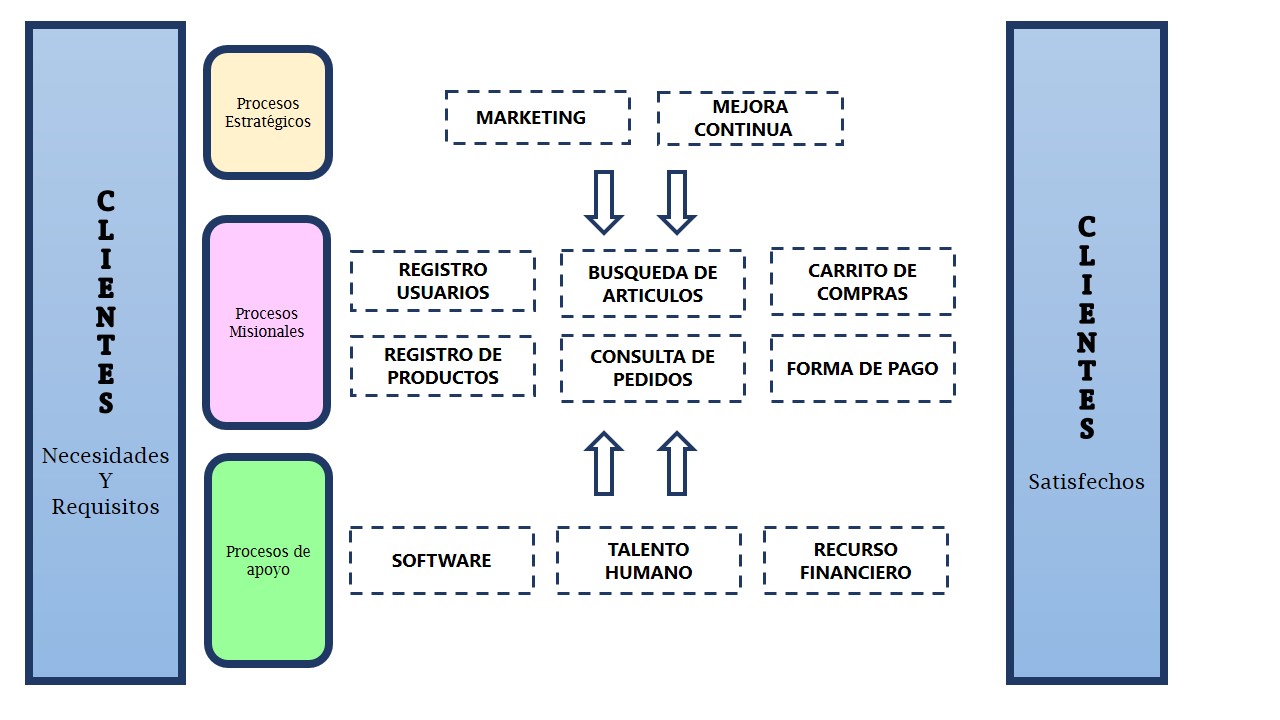
**PROVEEDOR**

****

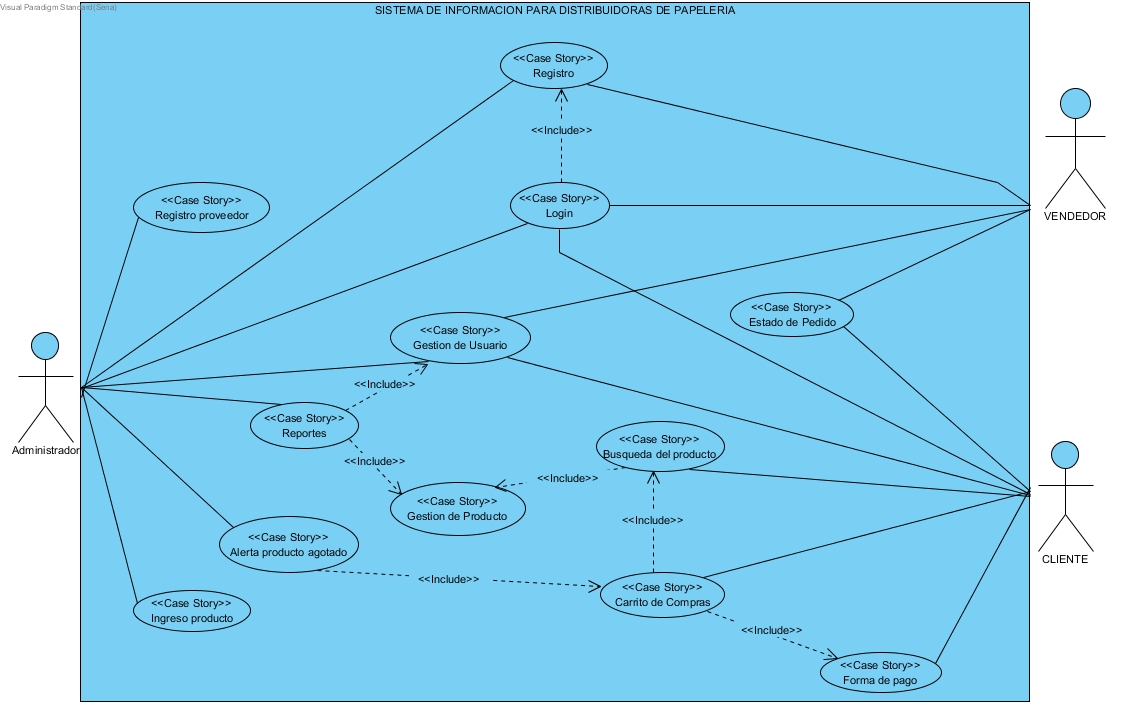
**VENDEDOR**

****

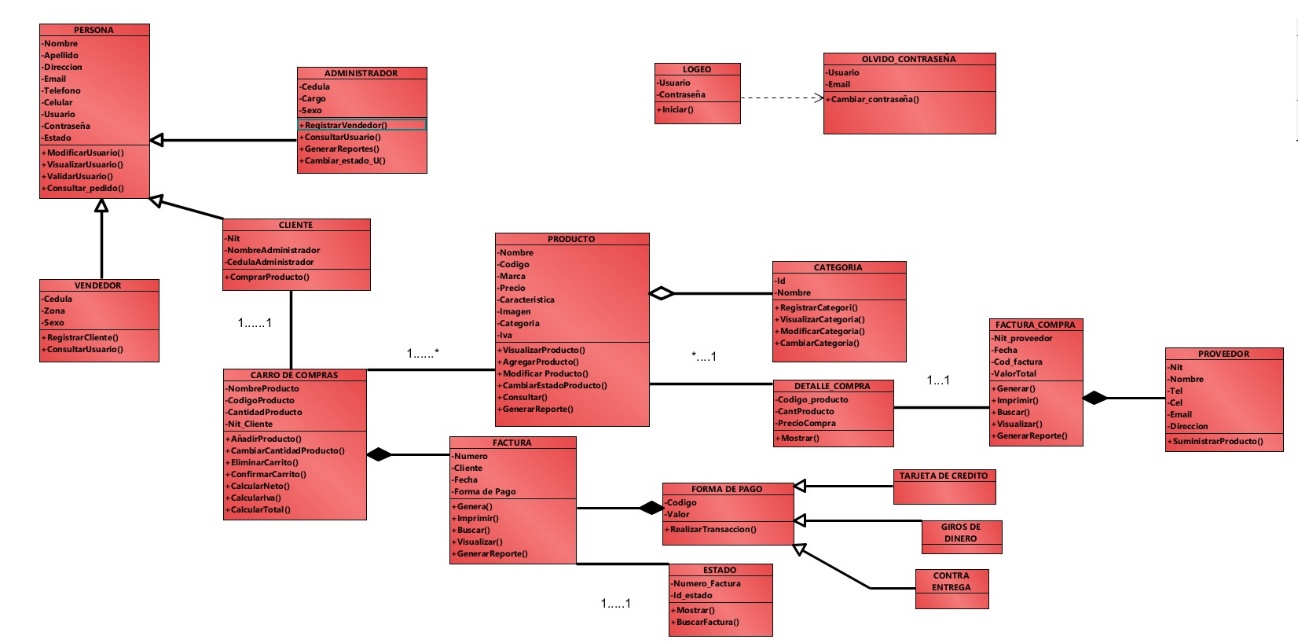
1. **DESCRIPCION DE PROCESOS**



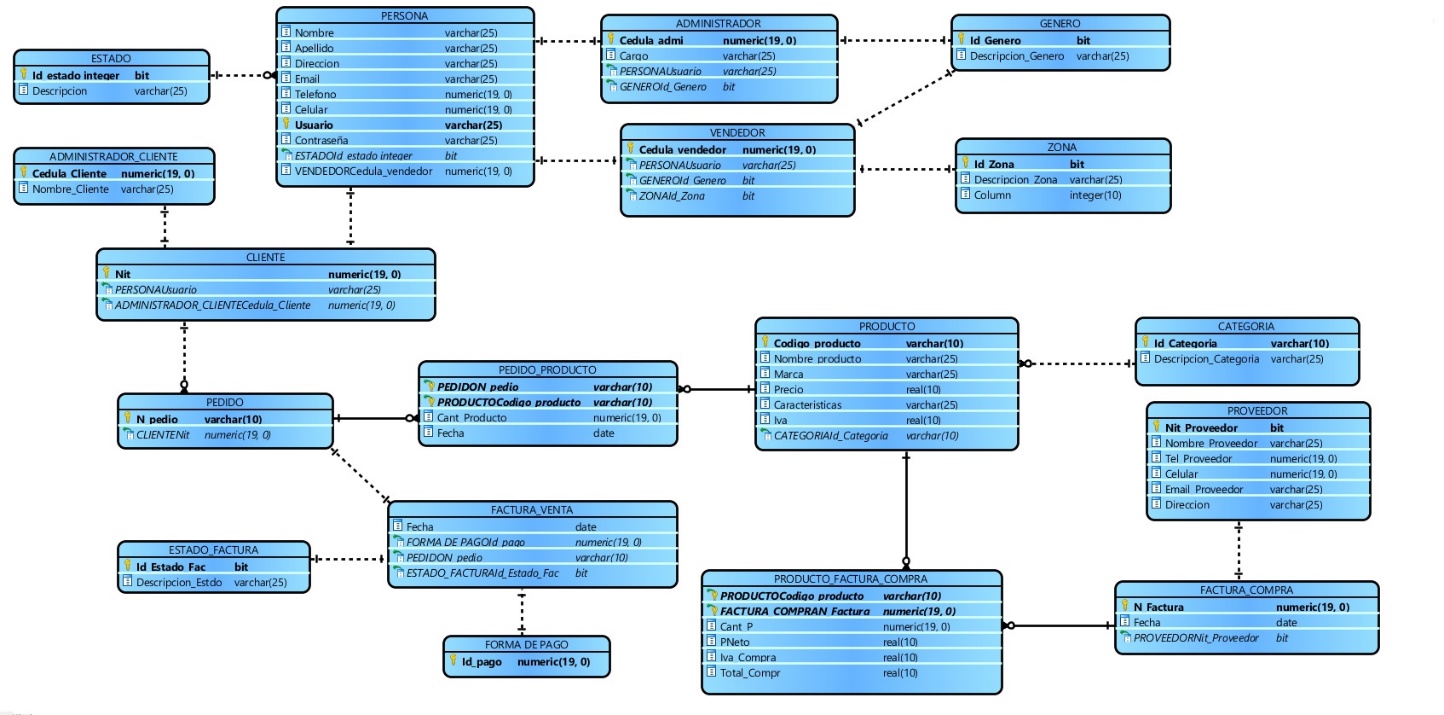
1. **DIAGRAMAS UML** 
   1. **DIAGRAMA DE CASOS DE USO**

****

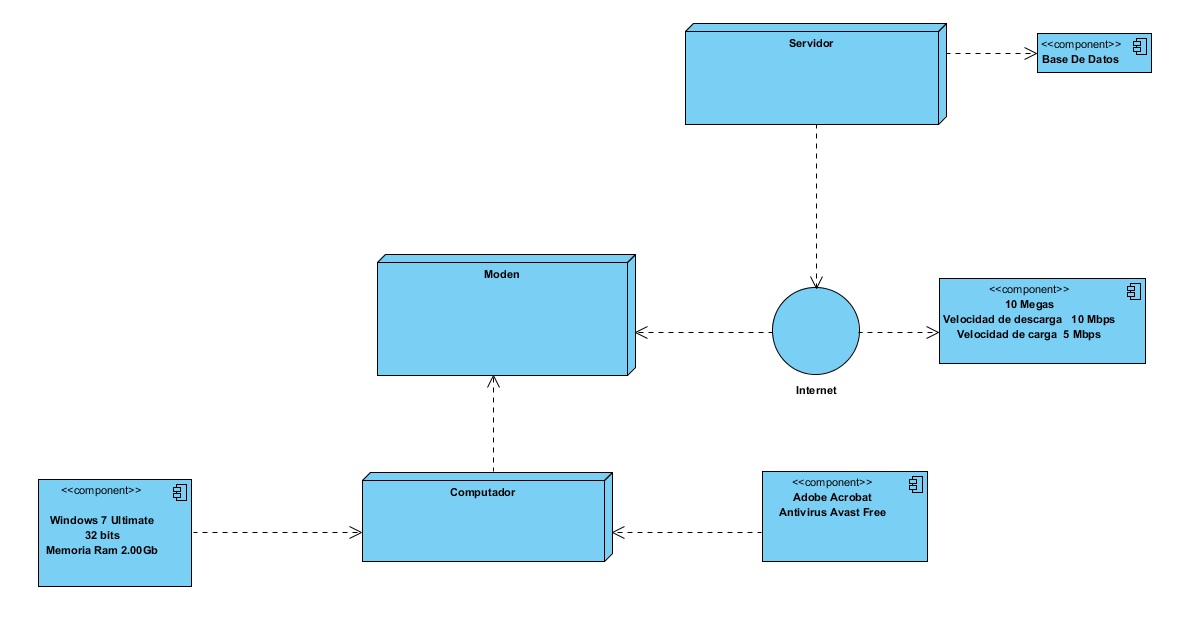
* 1. **DIAGRAMA DE CLASES**

****

* 1. **DIAGRAMA RELACIONAL**



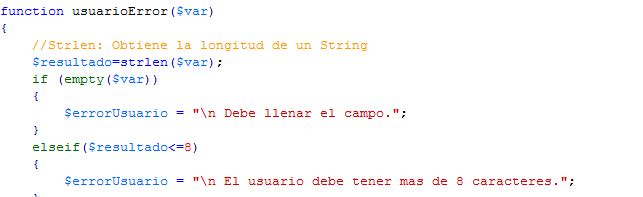
* 1. **DIAGRAMA DE DISTRIBUCION**



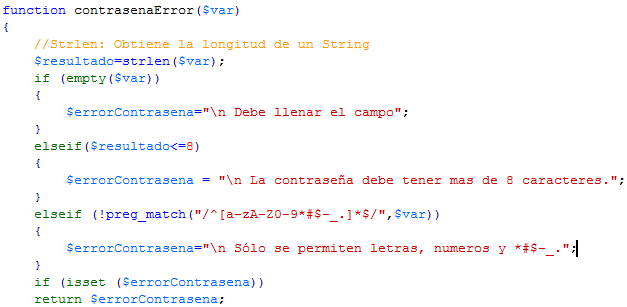
**DESCRIPCION DE ERRORES CODIGO**

Se validan que los datos ingresados estén correctamente digitados según los parámetros establecidos

UsuarioErrorfunción para verificar si el usuario tiene más de 8 caracteres

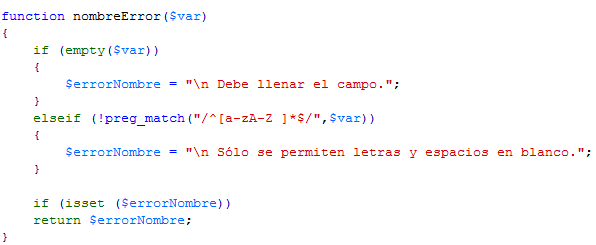
|

La función contrasenaError contiene la validación de que una contraseña cumpla los parámetros para dar seguridad al aplicativo



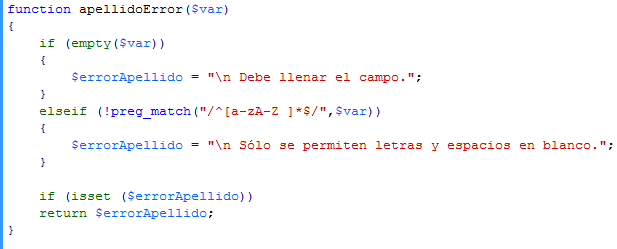
La función nombreError valida que los caracteres ingresados sean de tipo String (letra)

Y no permite que se ingresen otros tipos de datos

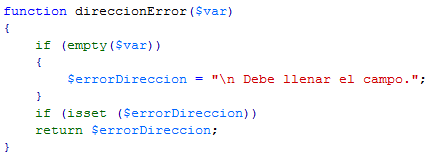


La función apellidoError valida que los caracteres ingresados sean de tipo String (letra)

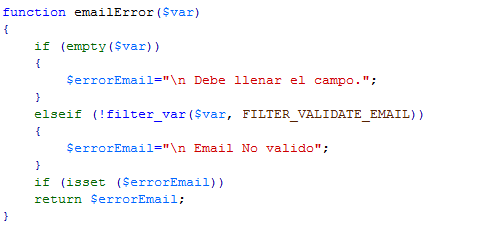
Y no permite que se ingresen otros tipos de datos



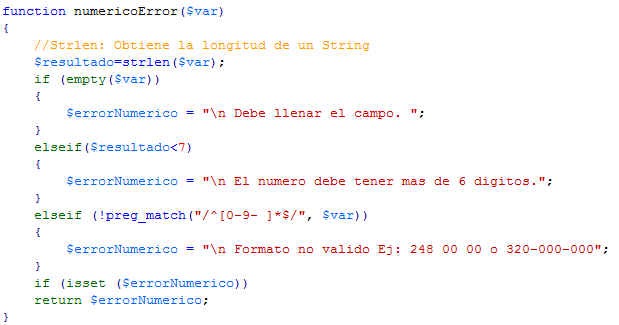
La función direcciónError valida que el usuario digite valores en el campo



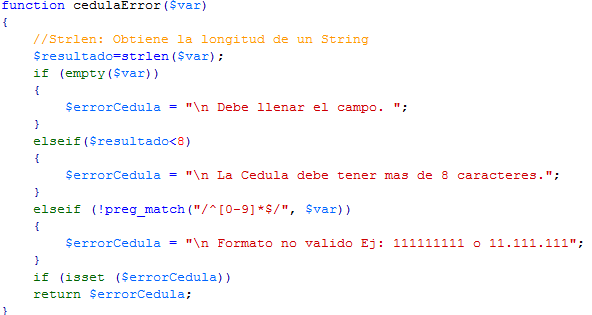
La función emailError valida que el usuario digite un correo electrónico para ello se lee un método FILTER\_VALIDATE\_EMAIL el cual determina los parámetros ingresados por un correo los cuales son @ seguido de unos caracteres y el .com



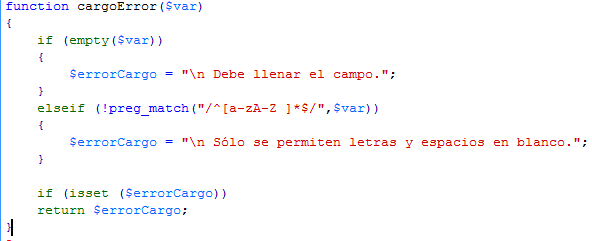
La vfuncion numéricoError valida que se ingresen datos numéricos



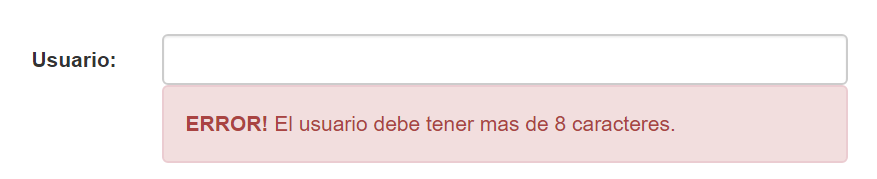
la función cedulaError valida que los datos ingresados contengan más de 8 caracteres

****

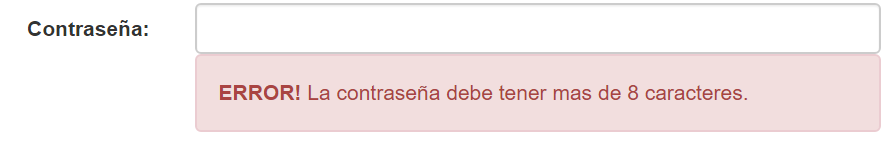
la función cargoError valida que se llene el campo y que los caracteres ingresados sean de tipo Sting (letras).

****

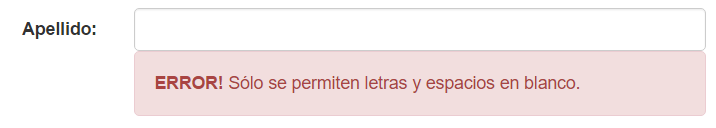
Cuando se ingresa un usuario con menos de 8 caracteres el sistema muestra por pantalla un mensaje de ERROR! Que indica que se deben digitar un usuario con mas de 8 caracteres.

****

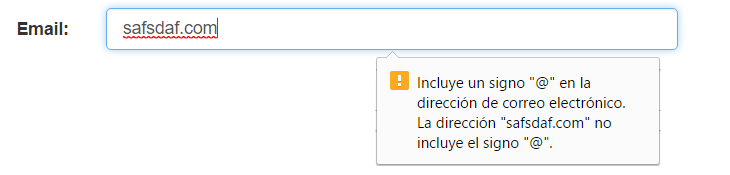
Si se ingresan menos de 8 caracteres en el campo contraseña el sistema muestra por pantalla un mensaje de ERROR! Indicandp que la contraseña debe ser de mas de 8 caracteres.

****

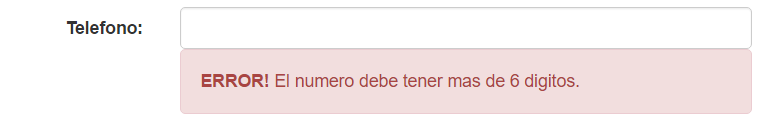
El sistema verifica que en el campo apellido solo se ingresen letras si se hace lo contrario muestra el mensaje de error y le advierte al usuario que debe digitar solo letras y espacios en blanco

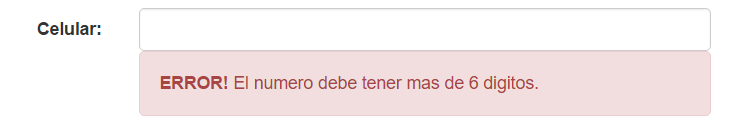
****

la funcion FILTER\_VALIDATE\_EMAIL por defecto nos muestra por ventana emergente el mensaje de validación

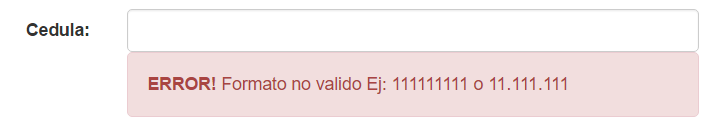
****

Si el sistema detecta que no se digitaron más de 6 dígitos ene l campo teléfono muestra el mensaje de error al usuario para que digite un número que contenga 6 dígitos

**** Si el sistema detecta que no se digitaron más de 6 dígitos ene l campo teléfono muestra el mensaje de error al usuario para que digite un número que contenga 6 dígitos



El sistema valida el formato en el cual se ingresan los datos a campo cedula si es incorrecto muestra el formato en el cual se deben ingresar los campos



**Errores para registrar un producto**

Se validan los campos y si se a generado algún error el sistema muestra por mensaje lo que esta ocurriendo y como llenar debidamente los campos para seguir con la ejecución

if(strtolower($existe['codigob']) == strtolower($\_POST['codigo']))

{

$errorCodigo="\n El CODIGO ya existe. ";

}

else

{

$codigo=dataForm($\_POST['codigo']);

}

}

else

$errorCodigo="\n Solo se permiten letras y numeros.";

}

//nombre (empty=vacio)

if (empty($\_POST['nombre']))

{

$errorNombre = "\n Debe ingresar el nombre.";

} elseif (!preg\_match("/^[a-zA-Z0-9 ]\*$/",$nombreValido))

{

$errorNombre = "\n Sólo se permiten letras, numeros y espacios en blanco.";

} else

{ //Caso verdadero obtenemos datos.

$nombre = dataForm($\_POST['nombre']);

}

//marca (empty=vacio)

if (empty($\_POST['marca']))

{

$errorMarca = "\n Debe ingresar la marca.";

} elseif (!preg\_match("/^[a-zA-Z ]\*$/",$marcaValido))

{

$errorMarca = "\n Sólo se permiten letras y espacios en blanco.";

} else

{ //Caso verdadero obtenemos datos.

$marca = dataForm($\_POST['marca']);

}

//precio

if (empty($\_POST['precio']))

{

$errorPrecio = "\n Debe ingresar el precio. ";

} elseif (!preg\_match("/^[0-9- ]\*$/", $precioValido))

{

$errorPrecio = "\n Precio no valido";

} else

{ //Caso verdadero obtenemos datos.

$precio = dataForm($\_POST['precio']);

}

//iva

if (empty($\_POST['iva']))

{

$errorIva = "\n Debe ingresar el iva. ";

} elseif (!preg\_match("/^[0-9- ]\*$/", $ivaValido))

{

$errorIva = "\n Iva no valido";

} else

{ //Caso verdadero obtenemos datos.

$iva = dataForm($\_POST['iva']);

}

//caracteristicas

if (empty($\_POST['caracteristicas']))

{

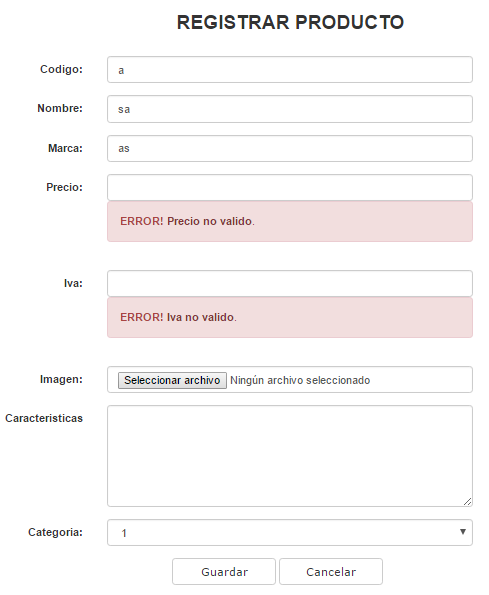
$errorCaracteristicas = "\n Debe ingresar las caracteristicas. ";

} else

{ //Caso verdadero obtenemos datos.

$caracteristicas = dataForm($\_POST['caracteristicas']);

}

****

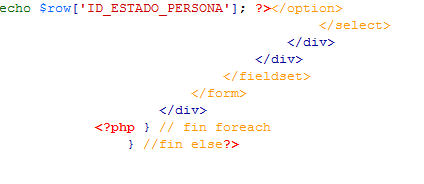
**FORMULARIOS**

**VER PERFIL**

En el siguiente código se llama todos los campos que se encuentran en la base de datos para imprimirlos en un formulario

****

****

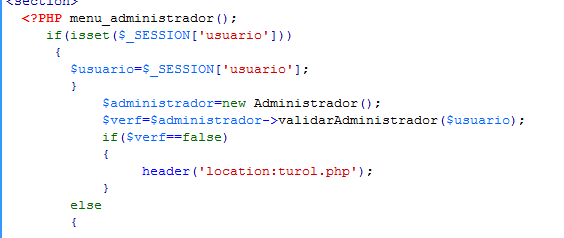
****

Después de ejecutar la consulta en el sistema se mostrara un formulario que contenga los campos con la información del usuario con el cual está ingresando

****

**MODIFICAR PERFIL**

El sistema verifica el usuario que ingresa al sistema



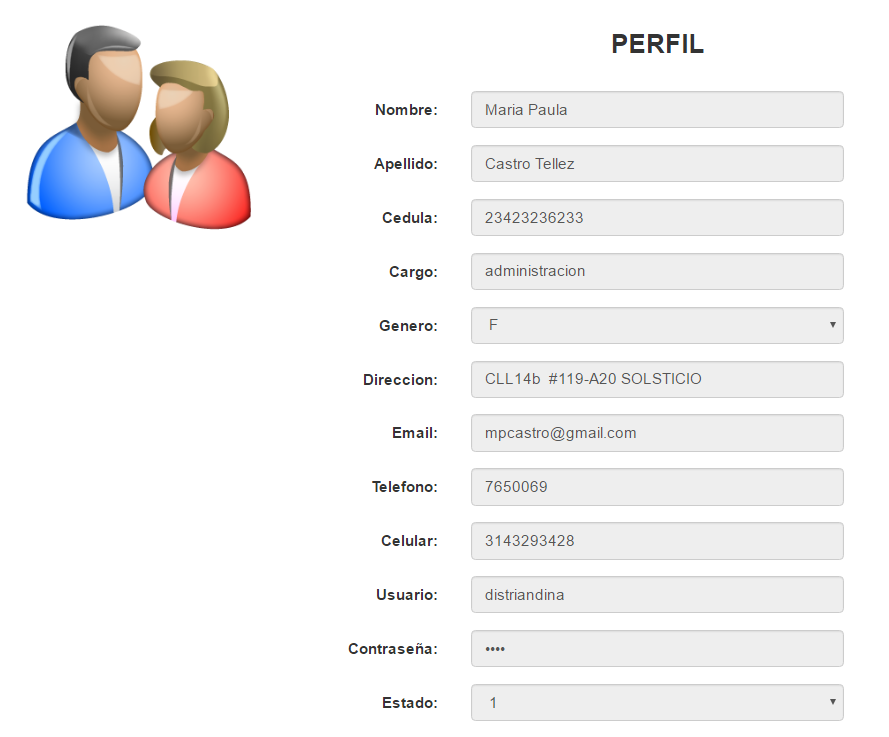
El sistema muestra por pantalla los datos del usuario ingresado al sistema



El sistema modifica los campos que se cambiaron en el formulario y almacena en la base de datos cambiando los datos solicitados



El sistema redireccióna a ver perfil y muestra el nuevo perfil con los datos cambiados



**CAMBIAR CONTRASEÑA**

****

**REGISTRO DE UN VENDEDOR**

El formulario se crea con los campos

Usuario Contraseña Nombre Apellido Dirección Email Teléfono Celular Cedula Genero Zona

Se leen los campos por el métodos $\_POST y se almacenan en variables para usarlas en el código PHP



El isset verifica que los daros ingresados sean correctos y de esta manera se crea un objeto de la clase vendedor para ingresar los datos y así guardarlos en la base de datos

if ((isset($usuario)) && (isset($contrasena)) && (isset($nombre)) && (isset($apellido)) && (isset($email)) && (isset($direccion)) && (isset($telefono)) && (isset($celular)) && (isset($cedula)))

{

$vendedor=new Vendedor;

$msg=$vendedor->registrarVendedor($\_POST['usuario'],$\_POST['contrasena'],$\_POST['nombre'],$\_POST['apellido'],$\_POST['direccion'],$\_POST['email'],$\_POST['telefono'],$\_POST['celular'],'1',$\_POST['cedula'],$\_POST['zona'],$\_POST['genero']);

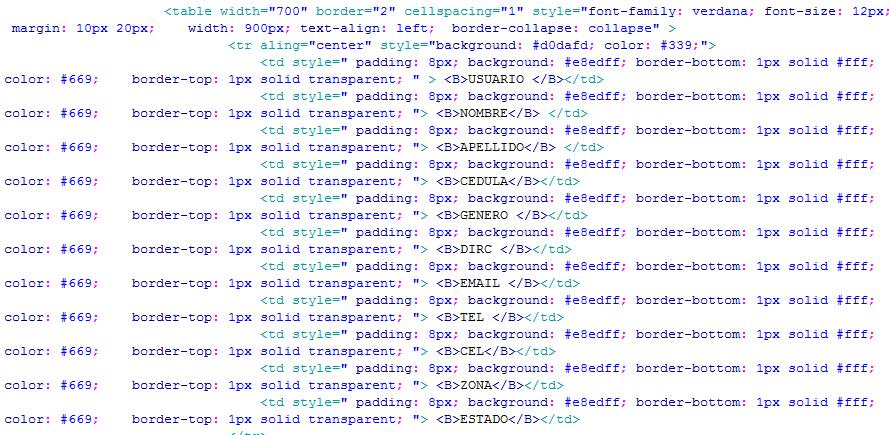
}

} ?>

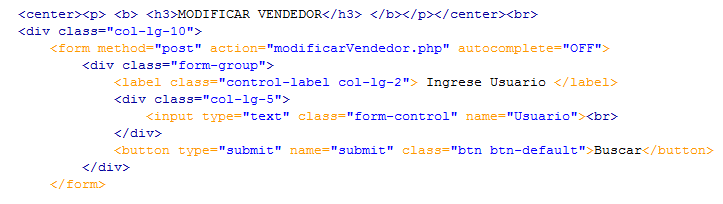
**+**

**MODIFICAR UN VENDEDOR**

El sistema muestra un formulario de todos los usuarios que se encuentran registrados



El sistema muestra un formulario para buscar el usuario pidiendo dijitar el formulario



****

**FOMULARIO PARA REGISTRAR UN PRODUCTO**

****

****

****

**BOTONES**

GUARDAR = se almacena en la Base de datos el vendedor registrado con los campos

Usuario

Contraseña

Nombre

Apellido

Dirección

Email

Teléfono

Celular

Cedula

Genero

Zona

CANCELAR =se da por terminada la operación sin ejecutar ningún cambio

**9 HERRAMIENTAS PARA IMPLEMENTACION DEL APLICATIVO**

**9.1 PHP**

Es un Lenguaje de Programación para trabajar páginas WEB ofreciendo la ventaja de mezclarse con HTML. Las ejecuciones son realizadas en el Servidor y el cliente es el encargado de recibir los resultados de la ejecución. Si el cliente realiza una petición, se ejecuta el intérprete de PHP y se genera el contenido de manera dinámica. Permite conexión con varios tipos de Bases de Datos como: MySql, Oracle, Postgress, SQL Server, etc. permitiendo aplicaciones robustas sobre la WEB. Este lenguaje de programación puede ser ejecutado en la gran mayoría de sistemas operacionales y puede interactuar con Servidores WEB populares

**9.2 MYSQL**

Es un manejador de Bases de Datos, el cual permite múltiples hilos y múltiples usuarios, fue desarrollado como software libre. Aunque se puede usar sobre varias plataformas es muy utilizado sobre LINUX. Es libre para uso en Servidores WEB. Ofrece ventajas tales como fácil adaptación a diferentes entornos de desarrollo, Interacción con Lenguajes de Programación como PHP, Java Script y fácil Integración con distintos sistemas operativos

**9.3 APACHE**

Es un Servidor WEB desarrollado por el grupo Apache. Su código fuente se puede distribuir y utilizar de forma libre. Está disponible para diferentes plataformas de Sistemas Operativos entre otros Windows, Linux, Mac y NetWare. Ofrece ventajas tales como independencia de plataforma, haciendo posible el cambio de plataforma en cualquier momento; creación de contenidos dinámicos, permitiendo crear sitios mediante lenguajes PHP. Además de ser libre su soporte técnico es accesible ya que existe una comunidad que está disponible en foros, canales IRC y servidores de noticias, donde hay gran cantidad de usuarios disponibles para cuando surge algún problema.

**10. BASE DE DATOS**

**CREATE TABLE PERSONA**

**(**

**USUARIO VARCHAR(25) NOT NULL,**

**CONTRASENA VARCHAR(25) NOT NULL,**

**NOMBRE VARCHAR(25) NOT NULL,**

**APELLIDO VARCHAR(25) NOT NULL,**

**DIRECCION VARCHAR(50) NOT NULL,**

**EMAIL VARCHAR(50) NOT NULL,**

**TELEFONO NUMERIC(19,0) NOT NULL,**

**CELULAR NUMERIC(19,0) NOT NULL,**

**ID\_ESTADO\_PERSONA VARCHAR(5) NOT NULL**

**);**

**CREATE TABLE ADMINISTRADOR**

**(**

**USUARIO VARCHAR(25) NOT NULL,**

**CEDULA\_ADMI NUMERIC(19,0) NOT NULL,**

**CARGO VARCHAR(25) NOT NULL,**

**ID\_GENERO\_ADMI VARCHAR(5) NOT NULL**

**);**

**CREATE TABLE VENDEDOR**

**(**

**USUARIO VARCHAR(25) NOT NULL,**

**CEDULA\_VENDEDOR NUMERIC(19,0) NOT NULL,**

**ZONA\_VENDEDOR VARCHAR(5) NOT NULL,**

**ID\_GENERO\_VENDEDOR VARCHAR(5) NOT NULL**

**);**

**CREATE TABLE GENERO**

**(**

**ID\_GENERO VARCHAR(5) NOT NULL,**

**DESCRIPCION\_GENERO VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE ZONA**

**(**

**ID\_ZONA VARCHAR(5) NOT NULL,**

**DESCRIPCION\_ZONA VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE CLIENTE**

**(**

**USUARIO VARCHAR(25) NOT NULL,**

**NIT NUMERIC(19,0) NOT NULL,**

**ADMINISTRADOR NUMERIC (19,0) NOT NULL**

**);**

**CREATE TABLE ADMINISTRADOR\_CLIENTE**

**(**

**CEDULA\_CLIENTE NUMERIC(19,0) NOT NULL,**

**NOMBRE\_CLIENTE VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE ESTADO**

**(**

**ID\_ESTADO VARCHAR(5) NOT NULL,**

**DESCRIPCION VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE PEDIDO**

**(**

**N\_PEDIDO VARCHAR(10) NOT NULL,**

**CLIENTE VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE PEDIDO\_PRODUCTO**

**(**

**NUMERO\_PEDIDO VARCHAR(10) NOT NULL,**

**COD\_PRODUCTO VARCHAR(10) NOT NULL,**

**CANT\_PRODUCTO DECIMAL(19,0) NOT NULL,**

**FECHA DATE NOT NULL**

**);**

**CREATE TABLE FACTURA\_VENTA**

**(**

**FECHA DATE NOT NULL,**

**N\_PEDIDO VARCHAR(10) NOT NULL,**

**ID\_ESTADO\_FACT BIT NOT NULL,**

**ID\_PAGO\_FACT NUMERIC(19,0) NOT NULL**

**);**

**CREATE TABLE ESTADO\_FACTURA**

**(**

**ID\_ESTADO\_FAC BIT NOT NULL,**

**DESCRIPCION\_ESTADO VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE FORMA\_DE\_PAGO**

**(**

**ID\_PAGO NUMERIC(19,0) NOT NULL,**

**DESCRIPCION\_PAGO VARCHAR(25) NOT NULL**

**);**

**CREATE TABLE PRODUCTO**

**(**

**CODIGO\_PRODUCTO VARCHAR(10) NOT NULL,**

**NOMBRE\_PRODUCTO VARCHAR(25) NOT NULL,**

**MARCA VARCHAR(25) NOT NULL,**

**PRECIO REAL NOT NULL,**

**CARACTERISTICAS VARCHAR(255) NOT NULL,**

**IVA REAL NOT NULL,**

**ID\_CATEGORIA\_PROD VARCHAR(10) NOT NULL,**

**ID\_ESTADO\_PROD VARCHAR(5) NOT NULL**

**);**

**CREATE TABLE CATEGORIA**

**(**

**ID\_CATEGORIA VARCHAR(10) NOT NULL,**

**DESCRIPCION\_CATEGORIA VARCHAR(25) NOT NULL,**

**ID\_ESTADO\_CAT VARCHAR(5) NOT NULL**

**);**

**CREATE TABLE PRODUCTO\_FACTURA\_COMPRA**

**(**

**CODIGO\_PRODUCTO\_FAC VARCHAR(10) NOT NULL,**

**N\_FACTURA NUMERIC(19,0) NOT NULL,**

**CANT\_P NUMERIC(19,0) NOT NULL,**

**PNETO REAL NOT NULL,**

**IVA\_COMPRA REAL NOT NULL,**

**TOTAL\_COMPRA REAL NOT NULL**

**);**

**CREATE TABLE FACTURA\_COMPRA**

**(**

**N\_FACTURA NUMERIC(19,0) NOT NULL,**

**FECHA DATE NOT NULL,**

**FAC\_NIT\_PROVEEDOR NUMERIC(19,0) NOT NULL**

**);**

**CREATE TABLE PROVEEDOR**

**(**

**NIT\_PROVEEDOR NUMERIC(19,0) NOT NULL,**

**NOMBRE\_PROVEEDOR VARCHAR(25) NOT NULL,**

**TEL\_PROVEEDOR NUMERIC(19,0) NOT NULL,**

**CEL\_PROVEEDOR NUMERIC(19,0) NOT NULL,**

**EMAIL\_PROVEEDOR VARCHAR(25) NOT NULL,**

**DIRECCION\_PROVEEDOR VARCHAR(25) NOT NULL**

**);**

**ALTER TABLE `persona` ADD PRIMARY KEY(`USUARIO`);**

**ALTER TABLE `administrador` ADD PRIMARY KEY( `USUARIO`);**

**ALTER TABLE `administrador\_cliente` ADD PRIMARY KEY( `CEDULA\_CLIENTE`);**

**ALTER TABLE `categoria` ADD PRIMARY KEY( `ID\_CATEGORIA`);**

**ALTER TABLE `cliente` ADD PRIMARY KEY( `USUARIO`);**

**ALTER TABLE `estado` ADD PRIMARY KEY( `ID\_ESTADO`);**

**ALTER TABLE `estado\_factura` ADD PRIMARY KEY( `ID\_ESTADO\_FAC`);**

**ALTER TABLE `factura\_compra` ADD PRIMARY KEY( `N\_FACTURA`);**

**ALTER TABLE `factura\_venta` ADD PRIMARY KEY( `N\_PEDIDO`);**

**ALTER TABLE `forma\_de\_pago` ADD PRIMARY KEY( `ID\_PAGO`);**

**ALTER TABLE `genero` ADD PRIMARY KEY( `ID\_GENERO`);**

**ALTER TABLE `pedido` ADD PRIMARY KEY( `N\_PEDIDO`);**

**ALTER TABLE `pedido\_producto` ADD PRIMARY KEY( `NUMERO\_PEDIDO`, `COD\_PRODUCTO`);**

**ALTER TABLE `producto` ADD PRIMARY KEY( `CODIGO\_PRODUCTO`);**

**ALTER TABLE `producto\_factura\_compra` ADD PRIMARY KEY( `CODIGO\_PRODUCTO\_FAC`, `N\_FACTURA`);**

**ALTER TABLE `proveedor` ADD PRIMARY KEY( `NIT\_PROVEEDOR`);**

**ALTER TABLE `vendedor` ADD PRIMARY KEY( `USUARIO`);**

**ALTER TABLE `zona` ADD PRIMARY KEY( `ID\_ZONA`);**

**-- LLAVES FORANEAS---**

**ALTER TABLE `administrador` ADD FOREIGN KEY (`USUARIO`) REFERENCES `persona`(`USUARIO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `administrador` ADD FOREIGN KEY (`ID\_GENERO\_ADMI`) REFERENCES `genero`(`ID\_GENERO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `cliente` ADD FOREIGN KEY (`USUARIO`) REFERENCES `persona`(`USUARIO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `cliente` ADD FOREIGN KEY (`ADMINISTRADOR`) REFERENCES `administrador\_cliente`(`CEDULA\_CLIENTE`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `vendedor` ADD FOREIGN KEY (`USUARIO`) REFERENCES `persona`(`USUARIO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `vendedor` ADD FOREIGN KEY (`ID\_GENERO\_VENDEDOR`) REFERENCES `genero`(`ID\_GENERO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `vendedor` ADD FOREIGN KEY (`ZONA\_VENDEDOR`) REFERENCES `zona`(`ID\_ZONA`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `persona` ADD FOREIGN KEY (`ID\_ESTADO\_PERSONA`) REFERENCES `estado`(`ID\_ESTADO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `pedido` ADD FOREIGN KEY (`CLIENTE`) REFERENCES `cliente`(`USUARIO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `factura\_venta` ADD FOREIGN KEY (`N\_PEDIDO`) REFERENCES `pedido`(`N\_PEDIDO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `factura\_venta` ADD FOREIGN KEY (`id\_estado\_fact`) REFERENCES `estado\_factura`(`id\_estado\_fac`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `factura\_venta` ADD FOREIGN KEY (`id\_pago\_fact`) REFERENCES `forma\_de\_pago`(`id\_pago`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `pedido\_producto` ADD FOREIGN KEY (`NUMERO\_PEDIDO`) REFERENCES `pedido`(`N\_PEDIDO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `pedido\_producto` ADD FOREIGN KEY (`COD\_PRODUCTO`) REFERENCES `producto`(`CODIGO\_PRODUCTO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `producto` ADD FOREIGN KEY (`ID\_CATEGORIA\_PROD`) REFERENCES `categoria`(`ID\_CATEGORIA`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `producto` ADD FOREIGN KEY (`ID\_ESTADO\_PROD`) REFERENCES `estado`(`ID\_ESTADO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `categoria` ADD FOREIGN KEY (`ID\_ESTADO\_CAT`) REFERENCES `estado`(`ID\_ESTADO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `producto\_factura\_compra` ADD FOREIGN KEY (`CODIGO\_PRODUCTO\_FAC`) REFERENCES `producto`(`CODIGO\_PRODUCTO`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `producto\_factura\_compra` ADD FOREIGN KEY (`N\_FACTURA`) REFERENCES `factura\_compra`(`N\_FACTURA`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**ALTER TABLE `factura\_compra` ADD FOREIGN KEY (`FAC\_NIT\_PROVEEDOR`) REFERENCES `proveedor`(`NIT\_PROVEEDOR`) ON DELETE RESTRICT ON UPDATE RESTRICT;**

**INSERT INTO `distriandina`.`genero` (**

**`ID\_GENERO` ,**

**`DESCRIPCION\_GENERO`**

**)**

**VALUES (**

**'M', 'MASCULINO'**

**), (**

**'F', 'FEMENINO'**

**);**

**INSERT INTO `distriandina`.`estado` (**

**`ID\_ESTADO` ,**

**`DESCRIPCION`**

**)**

**VALUES (**

**'1', 'ACTIVO'**

**), (**

**'0', 'INACTIVO'**

**);**

**Creación de las tablas.**

**Insertar llaves primarias a las tablas.**

**Insertar llaves foráneas a las tablas.**

**Insertar selección múltiple a campos de las tablas.**