

Base de Datos

Proyecto papeleria

Programadores:

Jiménez García María Fernanda

García Cruz Diana Aide

Silva Barrera Brandon

Félix Flores Paul Jaime





Objetivo

Objetivo del proyecto:

“El alumno analizará una serie de requerimientos y propondrá una solución que atienda a los mismos, aplicando los conceptos vistos en el curso”.

Objetivo Base de Datos de una Papelería.

Para el caso de una empresa de ventas de productos de papelería, se desarrolla un sistema de tienda para la gestión de ventas que permita a los clientes registrarse, seleccionar los productos, comprarlos y pagarlos.



Beneficios

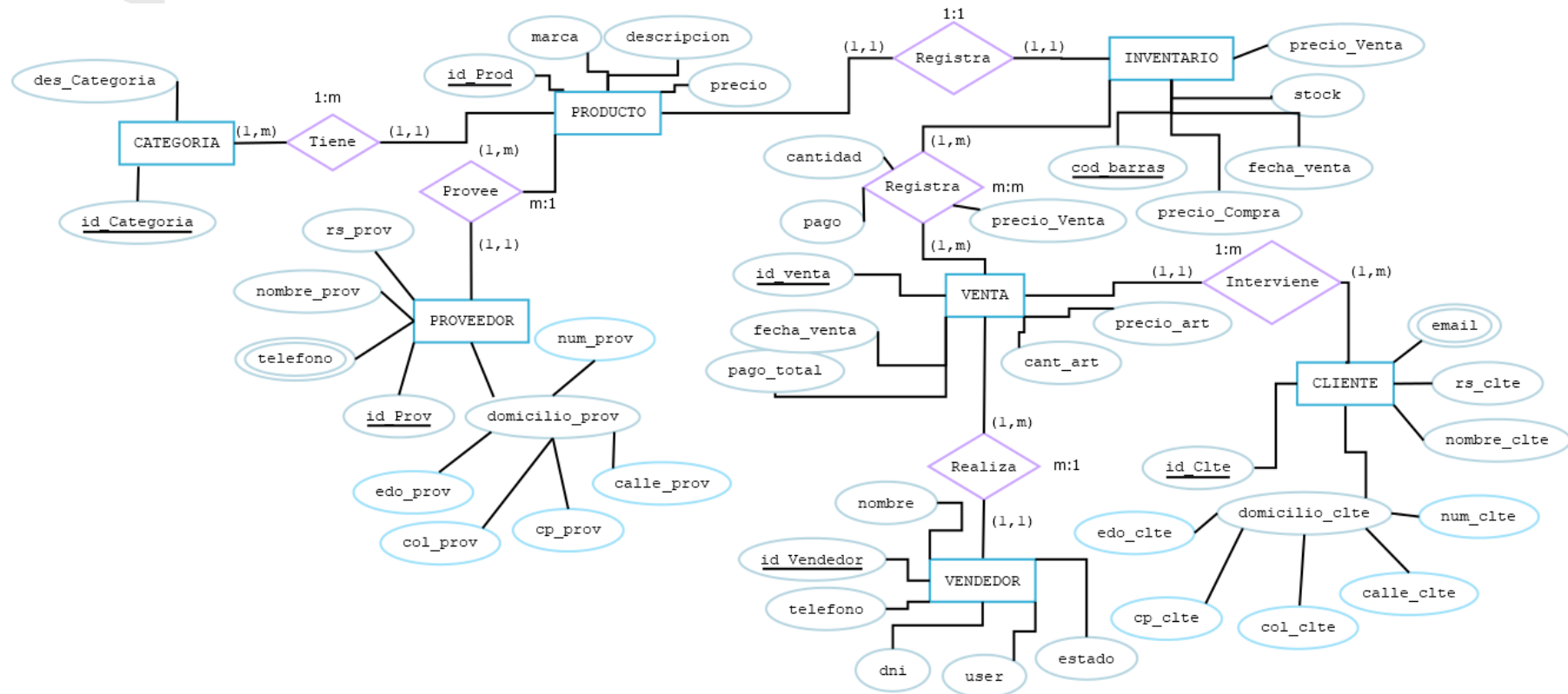
- Manejar un registro de clientes.
- Manejar un registro de ventas.
- Registro y manipulación de inventario.
- Permitir emitir reportes de ventas, clientes y productos actualizados que ayudan a la empresa a tomar mejores decisiones a corto y a largo plazo.



Modelo Entidad Relación.

- Los elementos del modelo entidad – relación a identificar son: entidades, atributos, identificadores y relaciones.
- Entidad: Objeto exclusivo único en el mundo real que se está controlando. Se quiere dar seguimiento.
 1. PROVEEDOR
 2. PRODUCTO
 3. CATEGORÍA
 4. INVENTARIO
 5. VENTA CLIENTE

Modelo Entidad-Relación





Modelo relacional

En el modelo relacional se utiliza un grupo de tablas para representar los datos y las relaciones entre ellos. Cada tabla está compuesta por varias columnas, y cada columna tiene un nombre único.

A partir del análisis del problema, y del diseño planteado mediante el modelo entidad relación obtenemos:



MAPEO DE ENTIDADES

PROVEEDOR {id_Prov (Pk), rs_Prov, nom_Prov, apPat_Prov,apMat_Prov, cp_Pov, col_Prov, calle_Prov num_Prov}

TELEFONO_PROVEEDOR {id_Prov(PK, FK), telefono(PK) }

CLIENTE{ id_Cliente, rs_Cliente, nom_Cliente, edo_Cliente, cp_Cliente, col_Cliente, calle_Cliente, num_Cliente}

EMAIL_CLIENTE {id_Cliente(FK,PK), email}

CATEGORIA {id_Categoria (PK), nom_Categoria}

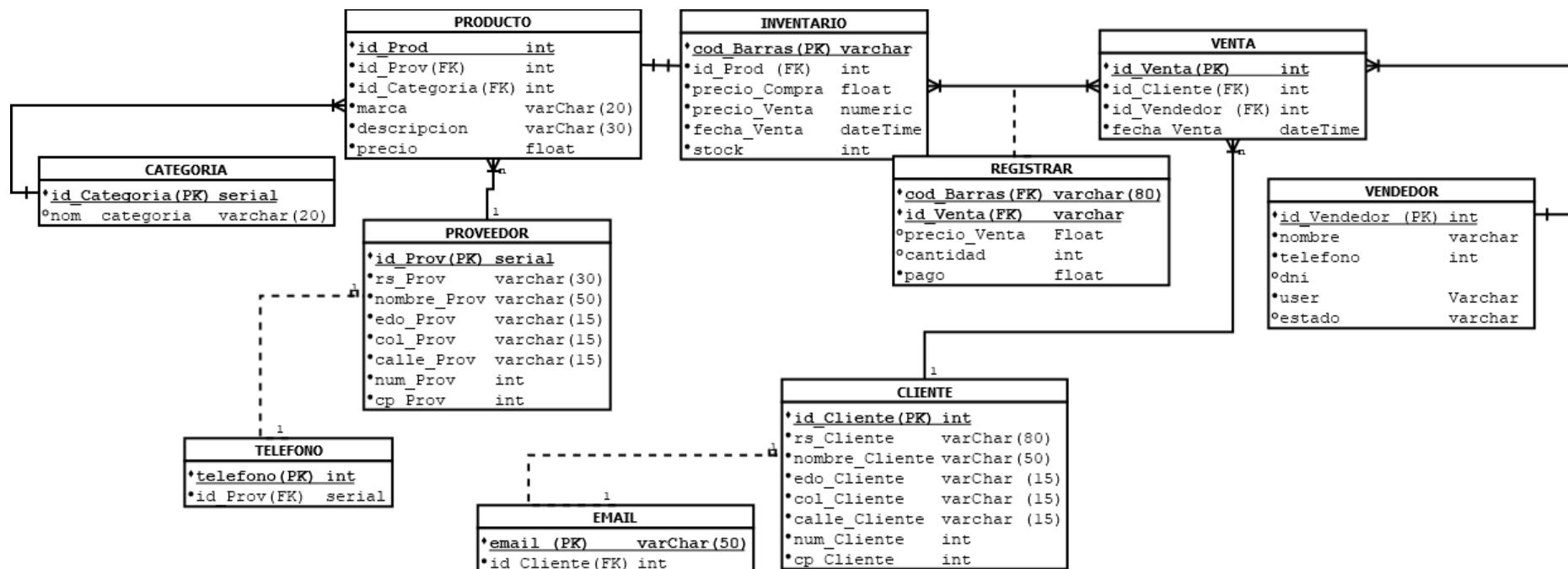
PRODUCTO {id_Prod (PK), id_Prov(FK), id_Categoria (FK), marca, descripcion, precio}

INVENTARIO { cod_Barras (PK), id_Prod (FK), precio_Compra, precio_Venta fecha_Llegada, stock}

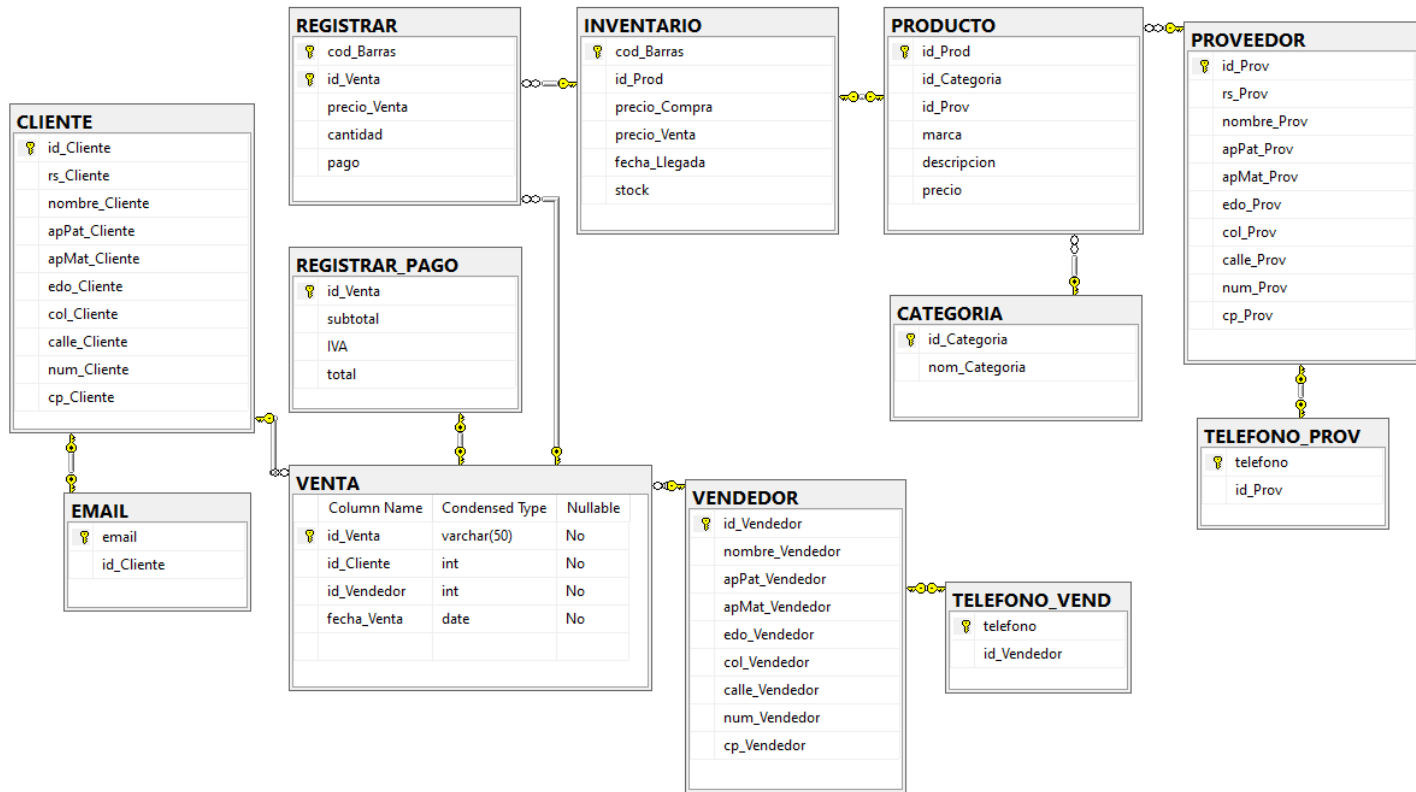
VENTA {id_Venta(PK), id_Cliente (FK), fecha_Venta, pago_total, cant_art, precio_art}

REGISTRAR {[cod_Barras (FK), id_Venta (FK)] PK, precio_Venta, cantidad, pago, total}

MODELO FINAL



BASE DE DATOS



PROGRAMACIÓN DE FUNCIONES EN POSTGRES.



FUNCIÓN UTILIDAD

```
1  /*Funci n que regresa la utilidad dado un c digo de barras*/
2  CREATE OR REPLACE FUNCTION UTILIDAD (varchar(80))
3  RETURNS numeric
4  AS $$
5      declare compra numeric;
6      declare venta numeric;
7      declare utilidad numeric;
8      BEGIN
9      compra:= (SELECT precio_Compra FROM INVENTARIO i WHERE i.cod_Barras = $1);
10     venta := (Select precio_venta FROM INVENTARIO i where i.cod_Barras = $1);
11     utilidad = venta - compra;
12     RETURN utilidad;
13     END;
14 $$LANGUAGE PLPGSQL;
15
16
17  -----EJECUTANDO LA FUNCION UTILIDAD-----
18  SELECT *FROM utilidad ( 'GT28 NYRO ZVT0 3QFV CTFT 3AKV RV8Q' );
```

	utilidad	
	numeric	🔒
1	30.00	

FUNCIÓN VENTA DURANTE UN PERIODO

```
1  /*Dada una fecha de inicio y fin , numero de productos vendidos ,
2  del m s vendido al menos. Organizados por producto.*/
3
4  CREATE OR REPLACE FUNCTION venta_periodo(vARCHAR , VARCHAR)
5  returns TABLE (cod_Barras VARCHAR(80), cantidad BIGINT)
6  as
7  $$
8      declare fin DATE;
9      declare inicio DATE;
10     BEGIN
11         fin = DATE($2);
12         inicio = DATE($1);
13         raise notice 'Fecha inicio: %', inicio;
14         raise notice 'Fecha fin: %', fin;
15         RETURN QUERY SELECT r.cod_barras , sum(r.cantidad)
16                             FROM registrar r
17                             inner join VENTA V ON R.id_Venta = V.id_Venta
18                             GROUP BY R.COD_BARRAS, R.CANTIDAD, V.ID_VENTA
19                             HAVING fecha_Venta>= inicio and fecha_Venta<=fin
20                             ORDER BY R.cantidad DESC;
21     END;
22     $$
23     LANGUAGE PLPGSQL;
24
25     _____REALIZANDO LA CONSULTA _____
26
27     SELECT VENTA_PERIODO( '2015-05-24' , '2018-05-24' );
```

	venta_periodo record
1	("AL47 2753 8452 2UHT 2N9T XYYS HB7N",10)
2	("LV88 TOYX D900 IHXZ 8LK0 0",7)
3	("FR09 2763 8619 10NZ ZAO1 Y3KI D85",5)
4	("FR15 0426 2996 71UI PXE1 T1SE 086",5)
5	("SA32 57IF DFRW XYQU ONLJ GUMS",4)
6	("SK57 2720 0365 1857 8838 3618",4)
7	("FR11 5418 3953 83NK JSQJ ZASG 563",3)
8	("MD02 P997 E0EW ID2V TARL OSQ0",3)
9	("CZ29 3049 8021 4441 1380 1105",2)
10	("GE67 HV16 6467 3257 6999 19",2)
11	("KZ95 931N JR00 571H ZQYK",2)
12	("AL47 2753 8452 2UHT 2N9T XYYS HB7N",1)
13	("BH94 ZAHG LVP5 6040 WSX9 NH",1)
14	("LI54 8712 8ZPZ 00KL JUHZ X",1)
15	("MK74 404S FVVH FAN3 J03",1)
16	("SK54 3772 2355 7482 3156 1717",1)

FUNCIÓN TOTAL DE VENTA DURANTE UN PERIODO


```
1  /*Venta periodo, dada una fecha de inicio y fin, total productos*/
2
3  CREATE OR REPLACE FUNCTION TOTALventa_periodo(v varchar, v varchar)
4  returns int
5  as
6  $$
7      declare fin date;
8      declare inicio date;
9      declare totalProductos int;
10     BEGIN
11         fin = date($2);
12         inicio = date($1);
13         raise notice 'Fecha inicio: %', inicio;
14         raise notice 'Fecha fin: %', fin;
15         totalProductos:= (SELECT SUM(R.cantidad) as CantidadProductosVendidos
16                           FROM Venta V
17                           INNER JOIN Registrar R ON R.id_Venta=v.id_Venta
18                           WHERE fecha_Venta between inicio and fin);
19         RETURN totalProductos;
20     END;
21     $$
22     LANGUAGE PLPGSQL;
23
24     -----REALIZANDO LA CONSULTA-----
25     SELECT totalVENTA_PERIODO( '2015-05-24', '2018-05-24');
```

	totalventa_periodo	
	integer	
1		52

```

1
2  /*-----Vista de una factura-----*/
3  CREATE VIEW FACTURA
4  AS
5  Select* FROM(
6      SELECT 1 AS Factura, 'Datos de Factura:' as PapeleriaBaseDeDatos ,
7      CONCAT('Cliente: ', CAST(C.id_Cliente AS varchar(10))) as Tel55070220 ,
8      CONCAT('Fecha: ',CAST(V.fecha_Venta AS varchar(10))) as AV20DENOVIEMBRE,
9      CONCAT('Venta: ', v.id_Venta) as Num1024, ' ' as Pago
10     FROM CLIENTE C
11     INNER JOIN Venta V ON V.id_Cliente = C.id_Cliente
12     Where v.id_Venta= 'VENT-001'
13  UNION
14     SELECT 2,CONCAT( 'Facturar A: ',C.appat_Cliente , ' ',C.nombre_Cliente),
15     C.rs_Cliente ,CONCAT(C.calle_Cliente , ' #',C.num_Cliente , ' ',
16     Col. ',C.col_Cliente , ' ', CP. ',C.cp_Cliente ),
17     E.email as CorreoElectronico , ' '
18     FROM CLIENTE C
19     INNER JOIN Venta V ON V.id_Cliente = C.id_Cliente
20     INNER JOIN email E ON E.id_Cliente = C.id_Cliente
21     Where v.id_Venta= 'VENT-001'
22  UNION
23     SELECT '3', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '
24  UNION
25     SELECT '3', 'Codigo Barras', 'Producto', 'Precio unitario', 'Cantidad ', ' '
26  UNION
27     SELECT 4, r.cod_Barras , P.descripcion as Producto ,
28     CAST(r.precio_Venta AS varchar(10)),
29     CAST(r.cantidad AS varchar(10)),CAST(r.pago AS varchar(10))
30  From Venta v

```



```

31     inner join registrar r on v.id_Venta = r.id_Venta
32     inner join inventario I on r.cod_Barras= i.cod_Barras
33     inner join producto P on p.id_Prod=i.id_Prod
34     Where v.id_Venta= 'VENT-001'
35 UNION
36     SELECT 5, ' ', ' ', ' ', ' ', 'SUBTOTAL', CAST(SUM(r.pago) AS varchar(10))
37     FROM Venta V
38     inner join registrar r on v.id_Venta = r.id_Venta
39     Where v.id_Venta= 'VENT-001'
40 UNION
41     SELECT 6, ' ', ' ', ' ', ' ', 'IVA 16%', CAST((SUM(r.pago))*0.16 AS varchar(10))
42     FROM Venta V
43     inner join registrar r on v.id_Venta = r.id_Venta
44     Where v.id_Venta= 'VENT-001'
45 UNION
46     SELECT 7, ' ', ' ', ' ', ' ', 'TOTAL', CAST((SUM(r.pago)+(SUM(r.pago))*0.16) AS
47     varchar(10))
48     FROM Venta V
49     inner join registrar r on v.id_Venta = r.id_Venta
50     Where v.id_Venta= 'VENT-001'
51 ) as FACT
52 ORDER BY Factura;
53
54 _____EJECUCION DE FACTURA_____
55 Select * From FACTURA

```



FACTURA

	factura integer	papeleriabasededatos text	tel55070220 text	av20denoviembre text	num1024 text	pago text
1	1	Datos de Factura:	Cliente: 1	Fecha: 2015-05-24	Venta: VENT-001	
2	2	Facturar A: Lenard Randolph	WageWorks, Inc.	Jackson #6516, Col.Welc...	acaizley0@lulu.com	
3	3					
4	3	Codigo Barras	Producto	Precio unitario	Cantidad	
5	4	LV88 TOYX DSO0 IHXZ 8LK0 0	Impresión de carteles	170.16	7	1191.12
6	4	SK54 3772 2355 7482 3156 ...	Oso de peluche	71.15	1	71.15
7	5				SUBTOTAL	1262.27
8	6				IVA 16%	201.9632
9	7				TOTAL	1464.2332



FUNCIÓN STOCK

```
1
2  /*-----CUANDO STOCK SEA MENOR A 3-----*/
3  CREATE OR REPLACE FUNCTION STOCK ()
4  RETURNS TABLE (cod_Barras varchar(80), cantidad int )
5  AS
6      $$
7      BEGIN
8          RETURN query
9              SELECT i.cod_Barras , i.stock
10             FROM INVENTARIO i
11             WHERE stock <= 3;
12      END;
13  $$LANGUAGE PLPGSQL;
14  -----EJECUCION STOCK-----
15  select *from stock ();
```



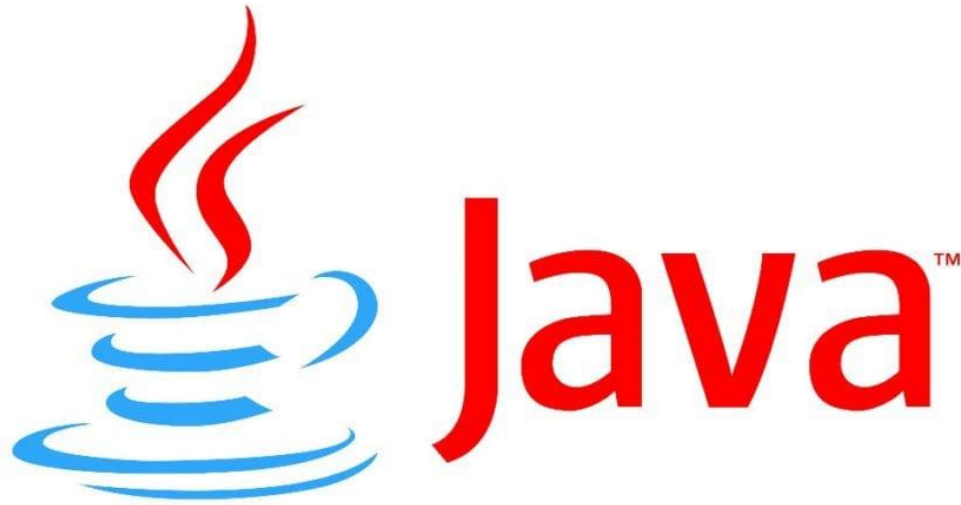
CREACIÓN DE UN ÍNDICE

```
1
2  /* Creación de índice */
3  —El índice se crea en el código de barra, con la finalidad de realizar
4  una búsqueda más rápida al hacer ventas.
5
6  CREATE INDEX Indice_Baras on INVENTARIO(cod_Barras );
```



INTERFAZ DE USUARIO

Se decidió utilizar java y sus bibliotecas gráficas awt y swing





Modificaciones a partir de la interfaz diseñada.

Se agrega una entidad a la base de datos llamada VENDEDOR para tener seguridad en el manejo de los datos. Al iniciar la interfaz se solicita un usuario y su contraseña.

Iniciamos sesión con el usuario hecho en phpMyadmin.



A screenshot of a login window for phpMyAdmin. The window has a purple-to-pink gradient background. At the top center is a black outline icon of a padlock with four dots below it. Below the icon are two input fields. The first is labeled 'USUARIO :' and contains the text 'emp01'. The second is labeled 'PASSWPRD :' and contains six asterisks '*****'. At the bottom is a button labeled 'INGRESAR'.

USUARIO :

emp01

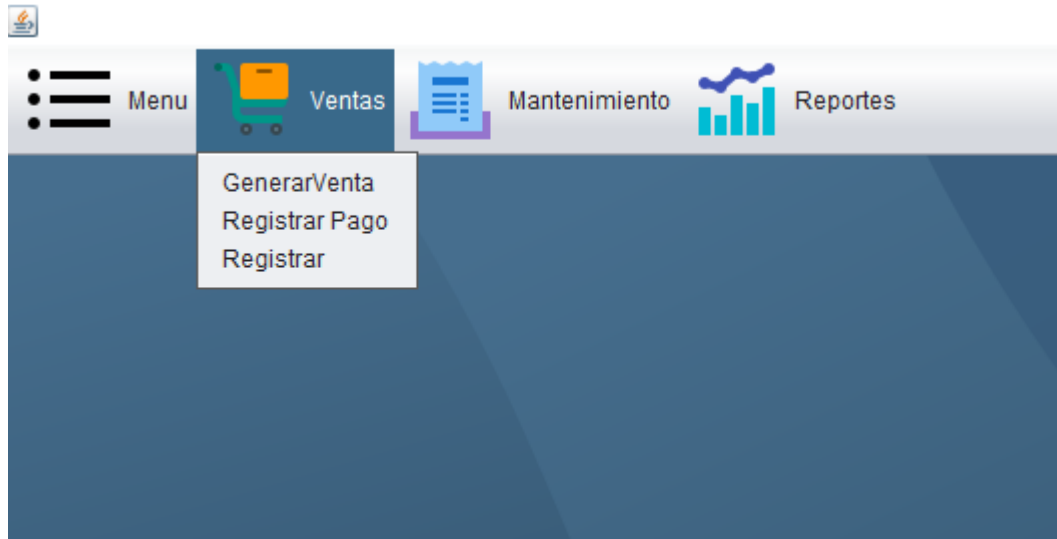
PASSWPRD :

INGRESAR

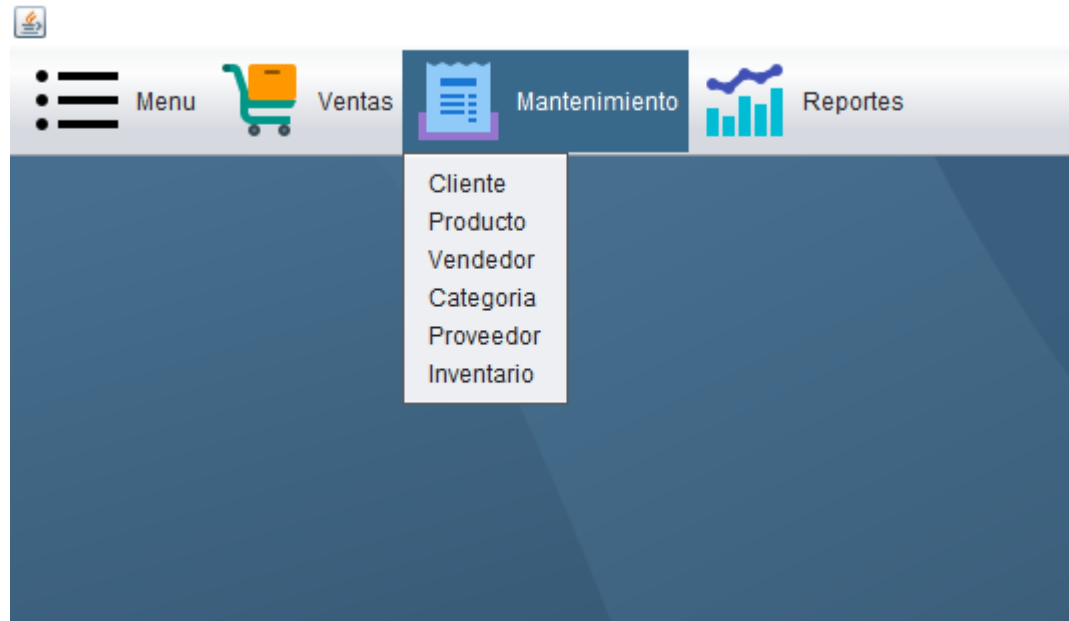
Se mostrará nuestra pantalla principal



Para el menú ventas tendremos estos submenús.



Para el menú mantenimiento tendremos:



Revisaremos nuestros datos en phpMyadmin.

☐ Mostrar todo | Número de filas: 25 ▾ | Filtrar filas: | Sort by key: Ninguna ▾

+ Opciones

   	▼	IdCliente	Dni	Rs	Nombres	EdoCliente	ColCliente	CalleCliente	CpCliente	EmailCliente
<input type="checkbox"/>  Editar  Copiar  Borrar		23	17288	1726382	Laura	Cuerna vaca	Pino suarez	Los azules	1726382	diana@
<input type="checkbox"/>  Editar  Copiar  Borrar		25	7814	17728	Andrew	Sinaloa	Azul	La Viga	17728	an17@gmail.com
<input type="checkbox"/>  Editar  Copiar  Borrar		26	7813	17728	Monse	CDMX	Azul	SailorMoon	17728	mon@gmail.com
<input type="checkbox"/>  Editar  Copiar  Borrar		27	222	17728	Paco	Sinaloa	Azul	La Viga	17728	paco@gmail.com
<input type="checkbox"/>  Editar  Copiar  Borrar		28	768	gg	Pablo	Veracruz	s	Is	8271	v@gmail.com

 ☐ Seleccionar todo | Para los elementos que están marcados:  Editar  Copiar  Borrar  Exportar

Ventana Ventas



El Zorro
Papeleria
www.elzorro.com

IDPROV:

PROVEEDOR:

IDCAT:

CATEGORIA:

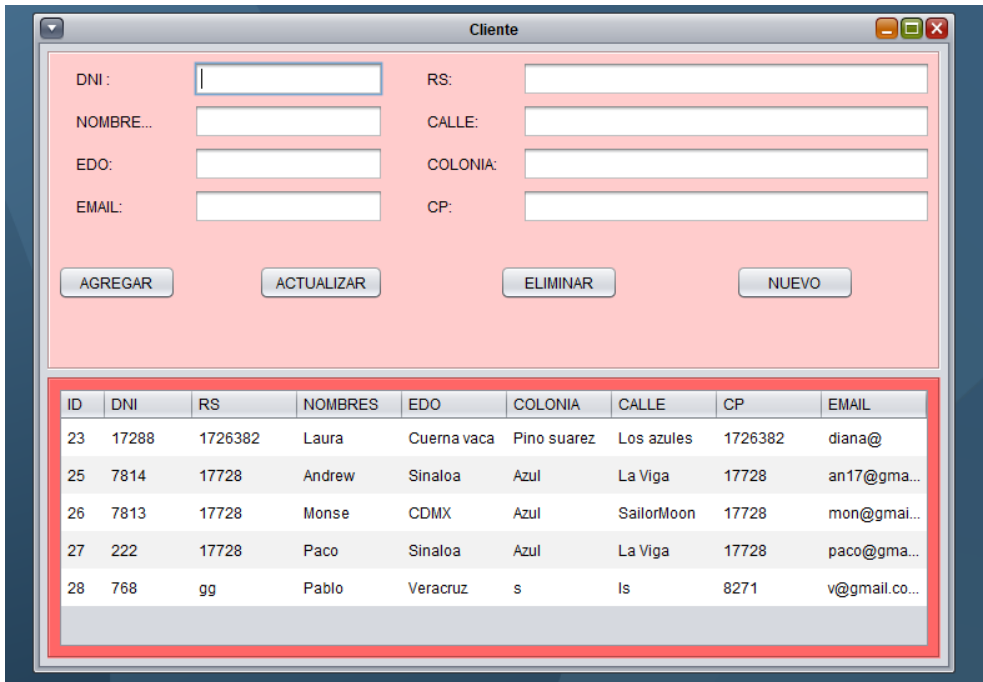
PRECIO:

MARCA:

DESCRIPCION:

Num	IDCAT	IDPROV	MARCA	DESCRIPCION	PRECIO
1	31	49	JELLY ROLL	4	6.0

Ventana Cliente.



The image shows a software window titled "Cliente". It contains a form with two columns of input fields. The first column has fields for DNI, NOMBRE..., EDO, and EMAIL. The second column has fields for RS, CALLE, COLONIA, and CP. Below the form are four buttons: AGREGAR, ACTUALIZAR, ELIMINAR, and NUEVO. At the bottom of the window is a table with 9 columns: ID, DNI, RS, NOMBRES, EDO, COLONIA, CALLE, CP, and EMAIL. The table contains 5 rows of data.

ID	DNI	RS	NOMBRES	EDO	COLONIA	CALLE	CP	EMAIL
23	17288	1726382	Laura	Cuerna vaca	Pino suarez	Los azules	1726382	diana@
25	7814	17728	Andrew	Sinaloa	Azul	La Viga	17728	an17@gma...
26	7813	17728	Monse	CDMX	Azul	SailorMoon	17728	mon@gmai...
27	222	17728	Paco	Sinaloa	Azul	La Viga	17728	paco@gma...
28	768	gg	Pablo	Veracruz	s	Is	8271	v@gmail.co...

Generación de Factura

Al generar una venta se crea automáticamente un archivo PDF con información de la venta.

