

UNIVERSIDAD LUTERANA SALVADOREÑA
FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN



PROYECTO:

Sistema de ventas (Tienda la Bendición)

EQUIPO DESARROLLADOR:

SOTO ALDANA, JUAN GABRIEL	SA01134315
PEREZ FUENTEZ, GUILLERMO ANTONIO	PF01132156
HERNÁNDEZ, ALDEMARO ANDRÉS	HD01134237
GARCÍA VILLAFRANCO, MOISÉS ISAI	GV01134159

CÁTEDRA: BASE DE DATOS II

HORARIO: VIERNES 7:00 a.m. a 9:30 a.m.

FACILITADOR: Lic. Rigoberto Israel Orellana

INDICE

INTRODUCCION.....	3
ANALISIS DEL PROYECTO.....	4
TEMA.....	4
PROBLEMÁTICA A RESOLVER.....	4
JUSTIFICACION.....	4
OBJETIVOS.....	5
General:.....	5
Específicos:.....	5
Información de la base de datos.....	6
Entidades y atributos:.....	6
Relaciones y sus identidades.....	7
DICCIONARIO DE DATOS.....	8
Desarrollo.....	9
Metadatos.....	10
Consultas útiles.....	12
Conclusiones.....	14
Recomendaciones.....	14
Bibliografía.....	15

INTRODUCCION

El presente trabajo pretende mejorar la condición de un negocio pequeño, de forma que facilite el uso y manejo de inventario, reportes, compra y venta de productos; ya que el negocio es una tienda, el dueño es la única persona encargada de estar ahí, por ende, le planteamos crear el proyecto de forma amigable y simple de manera que facilite el entendimiento. y que de esta manera posea un respaldo de sus datos.

Creando un sistema capaz de desempeñar un funcionamiento adecuado para lo que se requiere en este caso como lo es un sistema de ventas lo cual se desarrollara de manera podamos cumplir con los objetivos y poder presentarle un sistema amigable facil de usar y funcional de manera le facilite el tener el control de las ventas que realiza durante el dia, facilitando su buen funcionamiento en el sentido que tendra un mejor control.

ANALISIS DEL PROYECTO

TEMA

Desarrollo de un sistema de venta para la tienda “Variedades la bendicion”, Santiago Texacuangos, San Salvador.

PROBLEMÁTICA A RESOLVER

Se desarrollara un sistema del cual le brindara un beneficio de manera el encargado tenga un mejor control de sus productos, dando efectividad aun inventario, el sistema esta dirigido para el uso del propietario o ecargado de dicho establecimiento.

JUSTIFICACION

Nace la necesidad de crear una base de datos, en “Variedades la bendición”, por el hecho de llevar sus registros “compra-venta” de una manera manual, ya que en el pasado se perdieron los datos recabados en 3 meses, gracias a una pequeña inundación en la empresa; dicho libro de registros, afectaron el balance económico de la pequeña empresa, por ende el dueño decide, llevar registro “compra-venta” de manera digital, con el objetivo de salvaguardar su información de cualquier percance que el local pueda recibir.

Hoy en día, hasta las pequeñas empresas tienen la necesidad de manejar todo su inventario de forma digital, con el que garantizan una transparencia económica en sus bienes empresariales. Por ende “Variedades la bendicion”; decidió actualizarse a la era tecnológica.

OBJETIVOS

General:

Diseñar una base de datos para un sistema de venta para la tienda “Variedades la bendicion”, Santiago Texacuangos, San Salvador, con esto pueda llevar un mejor control de las ventas que realiza.

Específicos:

- Organizar cada tabla de la base de datos según sea su relación con las otras tablas.
- Detallar el modelo de la base de datos a diseñar para todas las necesidades y campos que necesitara para desarrollar el sistema de ventas.
- Explicar el uso de la base de datos, el porque de dichos campos.
- Establecer consultas personalizadas.

Información de la base de datos

Nombre de la base de datos: tienda

Entidades y atributos:

proveedor: idprovee, nombre, direccion, gmail, telefono

contacto_proveesor: idcon_pro, nombre, apellido, gmail, telefono

categoria_productos: idcat_produc, nombre

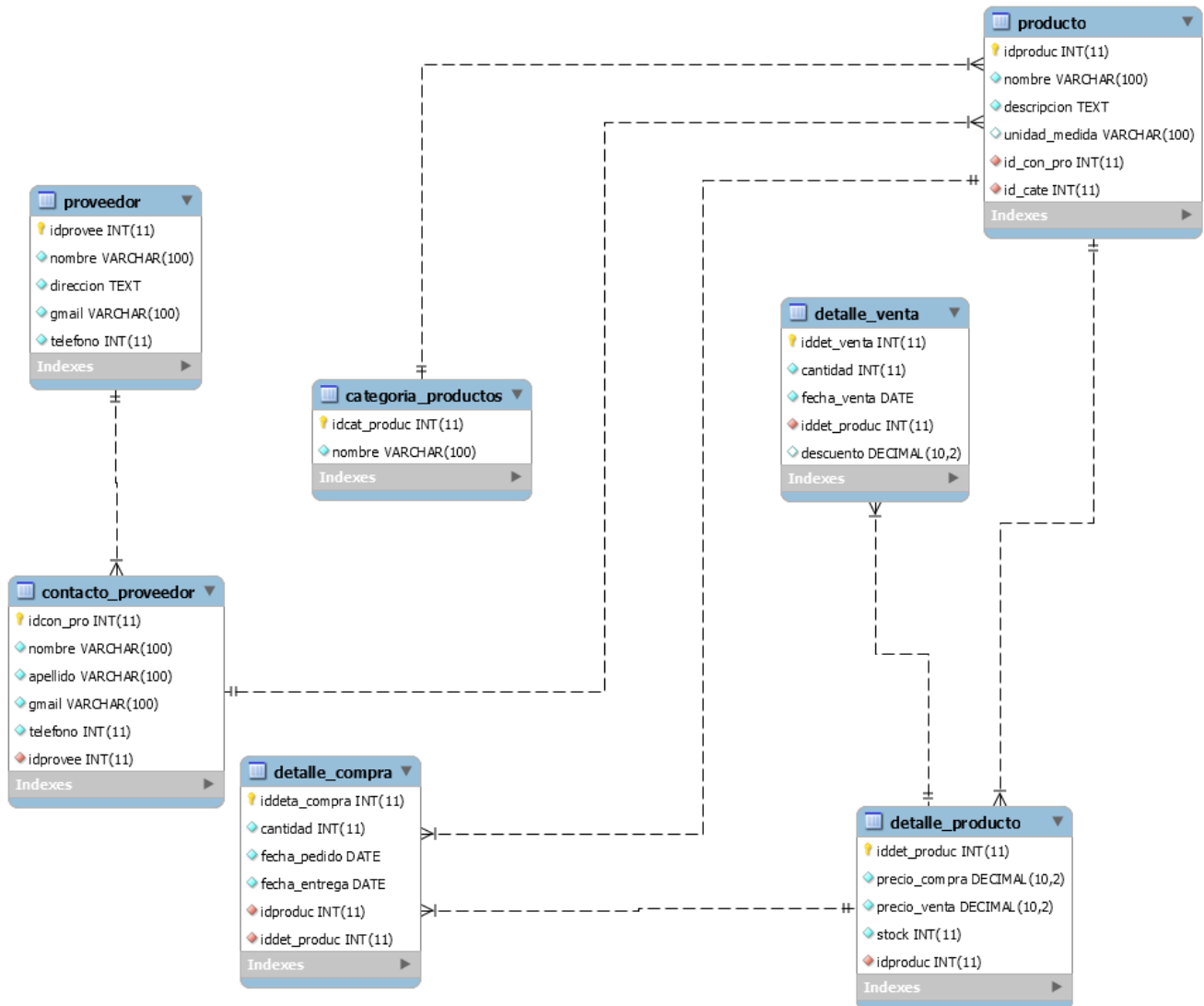
producto: idproduc, nombre, descripcion, unidad_medida, id_con_pro, id_cate

detalle_producto: iddet_produc, precio_compra, precio_venta, stock, idproducto

detalle_compra: iddete_compra, cantidad, fecha_pedido, fecha_entrega, idproduc, iddet_produc

detalle_venta: iddet_venta, cantidad, fecha_venta, iddet_produc, descuento

Relaciones y sus identidades



DICCIONARIO DE DATOS

1.- proveedor

Idprovee = { Caracter numerico }
nombre = {Caracter " "}
direccion = {Caracter " "}
gmail = {Caracter " "}
telefono = { Caracter numerico }
PRIMARY KEY = { Caracter numerico }

2.- contacto_proveedor

idcon_pro = { Caracter numerico }
nombre = {Caracter " "}
apellido = {Caracter " "}
gmail = {Caracter " "}
telefono = { Caracter numerico }
idprovee = { Caracter numerico }

3.- categoria_productos

idcat_produc = { Caracter numerico }
nombre = {Caracter " "}

4.- producto

idproduc = { Caracter numerico }
nombre = {Caracter " "}
descripcion TEXT NOT NULL,
unidad_medida = {Caracter " "}
id_con_pro = { Caracter numerico }
id_cate = { Caracter numerico }

5.- detalle_producto

iddet_produc = { Caracter numerico }
precio_compra = { Caracter numerico }
precio_venta = { Caracter numerico }
stock = { Caracter numerico }
idproduc = { Caracter numerico }

6.- detalle_compra

iddeta_compra = { Caracter numerico }
cantidad = { Caracter numerico }
fecha_pedido = { Caracter numerico }
fecha_entrega = { Caracter numerico }
idproduc = { Caracter numerico }
iddet_produc = { Caracter numerico }

7.- detalle_venta

iddet_venta = { Caracter numerico }
cantidad = { Caracter numerico }
fecha_venta = { Caracter numerico }
iddet_produc = { Caracter numerico }
descuento = { Caracter numerico }

Desarrollo

Para este proyecto se escogió sistema de venta para la tienda “Variedades la bendicion” para poder hacer una base de datos y que de esta forma los el dueño de dicho lugar, pudiera llevar un mejor control y de esa forma teniendo un buen orden, puedan llevar el control de las ventas en una base de datos, teniendo un mejor control.

Con ello se evita la perdida de datos, el lenguaje a usarse para crear esta base de datos sera POSTSQL, que es un buen lenguaje de desarrollo de base de datos, facilitando su uso y manipulación para la misma.

También se realizará una serie de consultas que servirán al dueño de la tienda para ciertas actividades que necesiten(consultas de la base de datos personalizadas).

Después de elaborar la base de datos, se hará un sistema con programación estructura en el lenguaje php, en el cual se hará conexión con dicha base de datos para su ingreso, modificación, listar y eliminación de datos.

Metadatos

table_catalog	table_schema	table_name	table_type	self_referencing_column_name	reference_generation	user_defined_type_catalog	user_defined_type_schema	user_defined_type_name	is_inset
tienda	public	proveedor	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES
tienda	public	categoria_productos	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES
tienda	public	contacto_proveedor	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES
tienda	public	producto	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES
tienda	public	detalle_compra	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES
tienda	public	detalle_producto	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES
tienda	public	detalle_venta	BASE TABLE	NULL	NULL	NULL	NULL	NULL	YES

7.4.1.1

table_name	column_name	data_type	column_default	is_nullable
columns	table_catalog	character varying	NULL	YES
pg_type	typarray	oid	NULL	NO
pg_statio_all_sequences	reloid	oid	NULL	YES
pg_class	relkind	"char"	NULL	NO
pg_language	lanplcallfoid	oid	NULL	NO
pg_user	useconfig	ARRAY	NULL	YES
pg_timezone_abbrevs	abbrev	text	NULL	YES
udt_privileges	udt_catalog	character varying	NULL	YES
pg_index	indisreplident	boolean	NULL	NO
element_types	object_type	character varying	NULL	YES
table_constraints	constraint_type	character varying	NULL	YES
pg_roles	rolinherit	boolean	NULL	YES
pg_stat_xact_all_tables	n_tup_upd	bigint	NULL	YES
collation_character_set_applicability	character_set_name	character varying	NULL	YES
pg_stat_all_tables	autovacuum_count	bigint	NULL	YES
_pg_foreign_table_columns	attname	name	NULL	YES
pg_depend	classid	oid	NULL	NO
pg_database	datdba	oid	NULL	NO
domain_constraints	domain_catalog	character varying	NULL	YES
pg_attribute	atttypid	oid	NULL	NO
pg_shseclabel	label	text	NULL	NO
pg_transform	trflang	oid	NULL	NO
columns	collation_name	character varying	NULL	YES
routines	maximum_cardinality	integer	NULL	YES
pg_attribute	atthasdef	boolean	NULL	NO
pg_type	typbyval	boolean	NULL	NO
pg_database	dattablespace	oid	NULL	NO
pg_amproc	amproc	regproc	NULL	NO
pg_replication_slots	datoid	oid	NULL	YES
pg_user_mappings	srvname	name	NULL	YES
element_types	numeric_precision_radix	integer	NULL	YES
routines	sql_data_access	character varying	NULL	YES
pg_class	reloftype	oid	NULL	NO
pg_index	indisprimary	boolean	NULL	NO
sequences	minimum_value	character varying	NULL	YES
pg_collation	collnamespace	oid	NULL	NO
pg_file_settings	sourceline	integer	NULL	YES
pg_am	amkeytype	oid	NULL	NO

constraint_name	table_name	column_name
contacto_proveedor_idprovee_fkey	contacto_proveedor	idprovee
producto_id_cate_fkey	producto	id_cate
producto_id_con_pro_fkey	producto	id_con_pro
detalle_producto_idproduc_fkey	detalle_producto	idproduc
detalle_compra_iddet_produc_fkey	detalle_compra	iddet_produc
detalle_compra_idproduc_fkey	detalle_compra	idproduc
detalle_venta_iddet_produc_fkey	detalle_venta	iddet_produc

constraint_name	table_name	column_name
detalle_venta_iddet_produc_fkey	detalle_producto	iddet_produc
detalle_compra_idproduc_fkey	producto	idproduc
detalle_compra_iddet_produc_fkey	detalle_producto	iddet_produc
detalle_producto_idproduc_fkey	producto	idproduc
producto_id_con_pro_fkey	contacto_proveedor	idcon_pro
producto_id_cate_fkey	categoria_productos	idcat_produc
contacto_proveedor_idprovee_fkey	proveedor	idprovee
detalle_venta_pkey	detalle_venta	iddet_venta
detalle_compra_pkey	detalle_compra	iddeta_compra
detalle_producto_pkey	detalle_producto	iddet_produc
producto_pkey	producto	idproduc
categoria_productos_pkey	categoria_productos	idcat_produc
contacto_proveedor_pkey	contacto_proveedor	idcon_pro
proveedor_pkey	proveedor	idprovee

Consultas útiles

Consulta para crear una vista, para el reporte de ventas.

```
47
48
49
50 create view reporte_venta as
51 select dp.iddet_produc as id CODIGO_PRODUCTO,
52        dp.precio_venta as cantidad_de_la_venta,
53        pr.idproduc as CODIGO_PRODUCTO,
54        concat(pr.nombre, ' ', pr.unidad_medida) as DETALLE_VENTA,
55        dv.iddet_venta as ID_DETALLE_VENTA,
56        concat(dv.cantidad, ' $', dv.descuento, ' __ ', dv.fecha_venta) as cantidad_descuento_fecha
57 from detalle_venta as dv
58 inner join detalle_producto as dp
59 on dp.iddet_produc = dv.iddet_produc
60 inner join producto as pr
61 on pr.idproduc = dp.idproduc;
62
63
```

Consulta para crear una vista, para el detalle de la compra.

```
24
25
26 create view detalle_compra as
27 select p.idproduc as CODIGO_PRODUCTO,
28        concat(p.nombre, ' ', p.descripcion) as PRODUCTO,
29        dp.iddet_produc as CODIGO_DETALLE_PRODUCTO,
30        dp.precio_compra as DETALLE_COMPRA,
31        dc.iddeta_compra as id detalle compra,
32        concat(dc.cantidad, ' ', dc.fecha_pedido, ' ', dc.fecha_entrega) as detalle_compra,
33        pr.idprovee as PROVEEDOR,
34        concat(pr.nombre, ' ', pr.telefono) as proveedor,
35        cp.idcat_produc as id_categoria_producto,
36        cp.nombre as categoria_productos
37 from detalle_compra as dc
38 inner join producto as p
39 on p.idproduc = dc.idproduc
40 inner join detalle_producto as dp
41 on dp.iddet_produc = dc.iddet_produc
42 inner join proveedor as pr
43 on pr.idprovee = dc.idprovee
44 inner join categoria_productos as cp
45 on cp.idcat_produc = p.id_cate;
46
47
```

Consulta para crear una vista, para el reporte de productos.

```
1
2 create view reporte_producto as
3 select p.idproduc as CODIGO_PRODUCTO,
4        p.nombre as PRODUCTO,
5        cpr.idcon_pro as CODIGO_PROVEEDOR,
6        concat(cpr.nombre,' ',cpr.apellido,' ',cpr.telefono) as contacto_proveedor,
7        pr.idprovee as PROVEEDOR,
8        concat(pr.nombre,' ', pr.telefono) as proveedor,
9        cp.idcat_produc as codigo_categoria_producto,
10       cp.nombre as categoria_productos
11 from detalle_producto as dp
12 inner join producto as p
13 on p.idproduc = dp.idproduc
14 inner join contacto_proveedor as cpr
15 on cpr.idcon_pro = p.id_con_pro
16 inner join proveedor as pr
17 on pr.idprovee = cpr.idprovee
18 inner join categoria_productos as cp
19 on cp.idcat_produc = p.id_cate;
20
21
```

Funcion para insertar datos a una tabla.

```
63
64
65 CREATE FUNCTION insertProvedor (idprovee integer, nombre character, direccion text, gmail character, telefono integer )
66 RETURNS void
67 AS
68 'INSERT INTO proveedor VALUES (idprovee, nombre, direccion, gmail, telefono);'
69 language sql;
70
71 select insertprovedor (123, 'antonio', 'sant ana', 'antonio123@gmail.com', '72390154');
72
73
74
75
```

Conclusiones

Atraves del desarrollo de esta base de datos hemos podido notar que es una forma importante de poder hacer las cosas en la vida diaria mucho más fáciles, más sin embargo la lógica para poder establecer un buen funcionamiento de una base de datos es muy importante, ya que se trabaja principalmente con la lógica a través del desarrollo de la base de datos.

El tener datos almacenados en una base de datos eso asegura que nuestros datos estén de manera segura y que podemos visualizar lo que queramos según la consulta que realicemos a través del código.

Aprendimos de una manera diferente a como trabajar a través de postgresql las bases de datos lo cual se hace más a través de líneas de comandos y utilizando la lógica es como podemos ser capaces de poder consultar lo que queramos según a lo que tenemos en nuestra base de datos.

Recomendaciones

Las recomendaciones serían tener mucho cuidado con algunas líneas de consultas ya que si se modifican esto no funcionará o si no se está llamando de manera adecuada de igual manera no funcionará ya que las consultas deben tener el nombre adecuado según la tabla a la que estamos consultando.

Bibliografía

PHP: ¿Qué es PHP? - Manual

URL: <http://php.net/manual/es/intro-what-is.php>

Postgresql.org

URL: <https://www.postgresql.org>

PostgreSQL

URL: <http://php.net/manual/es/book.pgsql.php>