



SoftWave

Proyecto final

BASE DE DATOS

ANDRADE SALAZAR, IGNACIO

GUERRERO SANTANA, EDGAR GEOVANNY

ORTIZ JIMENEZ, VLADIMIR

Base de datos - Punto de venta

Contenido

Definición del problema	2
Objetivos	3
Objetivo General:	3
Objetivos específicos:	3
Especificación de Entidades y atributos	4
Diagrama Entidad-Relación	13
Modelo Relacional	14
Tablas	15
Normalización	19
Enunciados para las sentencias SQL	27
Conclusiones	31
Referencias	31
Avances del sistema	32

Definición del problema

El presente proyecto pretende desarrollar una herramienta con el uso de la tecnología que hoy en día se encuentra en todos lados y en cualquier actividad que hagamos, en la administración de los negocios ha resultado muy favorable el uso de ella para el control y manejo de recursos, así como para la reducción de problemas, es por ello que este proyecto está dirigido a los pequeños y medianos negocios que busquen una herramienta que les ayude a administrar sus entradas y salidas de mercancías, así como el cálculo neto de su utilidad. Al ser el manejo de recursos el principal problema a resolver nos permite proporcionar soluciones a otras situaciones que puedan presentarse como el robo hormiga, la pérdida de tiempo en inventariar y la facilitación de información de tu cartera de clientes y sus productos favoritos al momento.



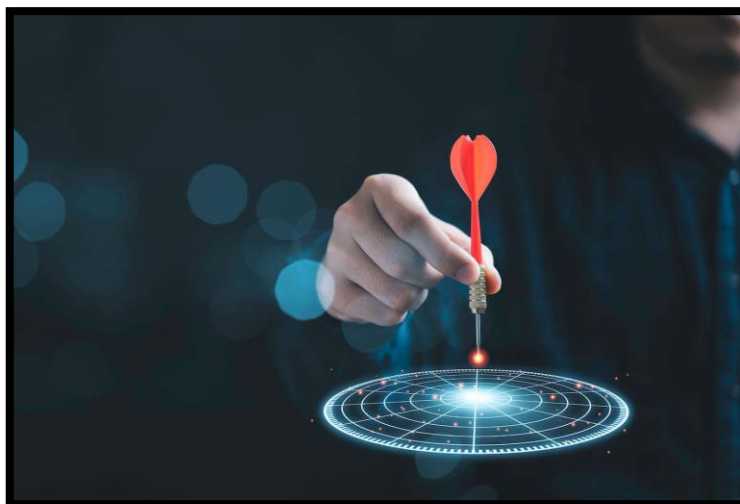
Objetivos

Objetivo General:

Proporcionar al usuario una estructura de almacenamiento de datos correctamente estructurada para la administración de mercancías de su negocio.

Objetivos específicos:

- Proporcionar al usuario la información necesaria para conocer la situación actual de su negocio, por ejemplo:
 - Ventas totales en un periodo de tiempo determinado
 - Cantidad en stock de los productos
 - Mercancías a punto de agotarse
 - Top de productos más vendidos o menos vendidos
 - Utilidad neta en un periodo de tiempo determinado
 - Clientes potenciales
 - Proveedores con mejores precios
 - Entre otras.
- Evitar robos por parte del personal o posibles alteraciones en los inventarios.
- Facilitación en el cálculo de utilidades y otros conceptos contables.



Especificación de Entidades y atributos

VENTAS		
Descripción	Almacena los datos del cliente y el monto total de la venta.	
Atributos	Descripción	Tipo de dato
id	Es la llave primaria.	Int autoincrementable (PK)
clientes_id	Es la llave foránea que relaciona a la entidad clientes.	(FK)
estatus_compras_v entas_id	Es la llave foránea que relaciona a la entidad estatus_compras_ventas.	(FK)
categoria_facturas	Es la llave foránea que relaciona la entidad categoría_facturas.	(FK)
monto_total	Almacena el valor monetario total de la venta.	Decimal(10,2)
descuentos_id	Es la llave foránea que relaciona la entidad con los descuentos.	(FK)

COMPRAS		
Descripción	Almacena los datos del proveedor y el monto total de la compra.	
Atributos	Descripción	Tipo de dato
id	Es la llave primaria.	Int autoincrementable(PK)
proveedores_id	Es la llave foránea que relaciona a la entidad proveedores.	(FK)
estatus_compras_ventas_id	Es la llave foránea que relaciona a la entidad estatus_compras_ventas.	(FK)
monto_total	Almacena el valor monetario total de la compra.	Decimal(10,2)
descuentos_id	Es la llave foránea que relaciona la entidad descuentos.	(FK)

CLIENTES		
Descripción	Entidad que almacena los datos personales del cliente y sus datos para la aplicación móvil.	
Atributos	Descripción	Tipo de dato
id	Es la llave primaria.	Int autoincrementable(PK)
nombre	Guarda el nombre completo del cliente.	String(99)
telefono	Guarda el teléfono del cliente 10 dígitos	String(10)
calle	Guarda el nombre de la calle de la dirección del cliente.	String(99)
numero	Guarda el número del domicilio del cliente.	Int
codigo_postal	Guarda el código postal de la dirección del cliente.	Int
email	Guarda un email.	String(45)
password	Guarda una contraseña generada por el cliente para acceder a la aplicación de pedidos y se guarda encriptada.	String(99)

PROVEEDORES		
Descripción	Entidad que almacena la información de contacto del proveedor.	
Atributos	Descripción	Tipo de dato
id	Es la llave primaria.	Int auto_increment(PK)
nombre	Guarda el nombre completo del proveedor.	String(99)
telefono	Guarda el teléfono del proveedor 10 dígitos	String(10)
calle	Guarda la dirección del proveedor.	String(99)
numero	Guarda el número del domicilio del proveedor.	Int
codigo_postal	Guarda el código postal de la dirección del proveedor.	Int
email	Guarda un email.	String(45)

INGRESOS		
Descripción	Entidad que contiene los ingresos provenientes de las ventas realizadas al público	
Atributo	Descripción	Tipo de dato
id	Es la llave primaria.	INT autoincrement(PK)
ingreso_total	total, del ingreso	DECIMAL(10, 2)
fecha	fecha en la que se registró el ingreso	TIMESTAMP(2)

EGRESOS		
Descripción	Almacena el egreso total asociado al número de compra correspondiente.	
Atributo	Descripción	Tipo de dato
id	Es la llave primaria.	INT autoincrement(PK)
egreso_total	total, del egreso	DECIMAL(10, 2)
fecha	fecha en la que se registró el egreso	TIMESTAMP(2)

UTILIDADES		
Descripción	Almacena la utilidad bruta diaria restando los ingresos y los egresos.	
Atributo	Descripción	Tipo de dato
id	Es la llave primaria.	INT autoincrement(PK)
utilidad	total, del egreso	DECIMAL(10, 2)
fecha_cierre	Fecha del cierre de caja y cálculo de utilidad bruta	TIMESTAMP(2)
ingresos_id	Es la llave foránea que relaciona la entidad ingresos con utilidades	FK
egresos_id	Es la llave foránea que relaciona la entidad egresos con utilidades	FK

NOTA: Solo se tendrá un registro de utilidad por día ya que esta entidad está relacionada uno a uno con ingresos y egresos.

ÓRDENES DE VENTA		
Descripción	Relaciona la entidad ventas con productos y almacena la cantidad de cada producto asociado a un número de venta.	
Atributo	Descripción	Tipo de dato
id	identificador de la venta	INT(PK)
productos_id	identificador de los productos que fueron vendidos en cierto número de venta	FK
ventas_id	identificador del número de venta que cuenta con sus datos	FK
cantidad	cantidad total de la venta	INT
ÓRDENES DE COMPRA		
Descripción	Relaciona la entidad compras con productos y almacena la cantidad de cada producto asociado a un número de compra.	
Atributo	Descripción	Tipo de dato
id	identificador de la compra	INT(PK)
productos_id	identificador de los productos que fueron comprados en cierto número de compra	FK
ventas_id	identificador del número de compra que cuenta con sus datos	FK
cantidad	cantidad total de la compra	INT

PRODUCTOS		
Descripción	Entidad que almacena la información de cada producto, podría considerarse como la tabla más importante de la base de datos, almacena tanto costo como precio de cada producto.	
Atributo	Descripción	Tipo de dato
id	número identificador del producto	INT autoincrement(PK)
categorias_productos_id	Es la llave foránea que relaciona la entidad productos con categoria_productos	FK
nombre_producto	nombre del producto	VARCHAR(45)
precio	precio al público	DECIMAL(10, 2)
costo	costo que tuvo con proveedor	DECIMAL(10, 2)
fecha_alta	fecha en la que se dio de alta el producto	TIMESTAMP(2)

INVENTARIOS		
Descripción	Relaciona la entidad productos con almacenes y almacena la cantidad en stock de cada producto y su valor monetario.	
Atributo	Descripción	Tipo de dato
id	número identificador del inventario	INT autoincrement(PK)
productos_id	Es la llave foránea que relaciona la entidad productos con la de inventarios	FK
almacenes_id	Es la llave foránea que relaciona la entidad almacenes con la de inventarios	FK
stock	Cantidad en unidades inventariadas	INT
valorMonetario	Valor en dinero del producto inventariado	DECIMAL(10, 2)
fecha_registro	fecha en la que se registró el inventario	TIMESTAMP(2)
fecha_caducidad	Fecha de caducidad del lote de productos inventariado	TIMESTAMP(2)

ALMACENES		
Descripción	Guarda un conjunto de almacenes virtuales que sirven para identificar el estado del producto, por ejemplo, el almacén 0001 es el almacén de productos disponibles, mientras que el almacén 0002 es el almacén de productos próximos a caducar.	
Atributo	Descripción	Tipo de dato
id	número identificador del inventario	INT autoincrement(PK)
almacen	Nombre del almacén	VARCHAR(45)
descripcion	Características que tienen los productos almacenados aquí.	VARCHAR(255)

CATEGORIAS PRODUCTOS		
Descripción	Define la clasificación de productos para su mejor manejo e identificación.	
Atributo	Descripción	Tipo de dato
id	identificador de la categoría	INT autoincrement(PK)
categoria_productos	nombre de la categoría	VARCHAR(45)

ESTATUS COMPRA-VENTA		
Descripción	Define el estado actual de la compra o venta, se utiliza para llevar un control de solicitudes finalizadas, canceladas, en proceso, etc.	
Atributo	Descripción	Tipo de dato
id	identificador del estatus	INT autoincrement(PK)
estatus	estado de la compra/venta	VARCHAR(45)

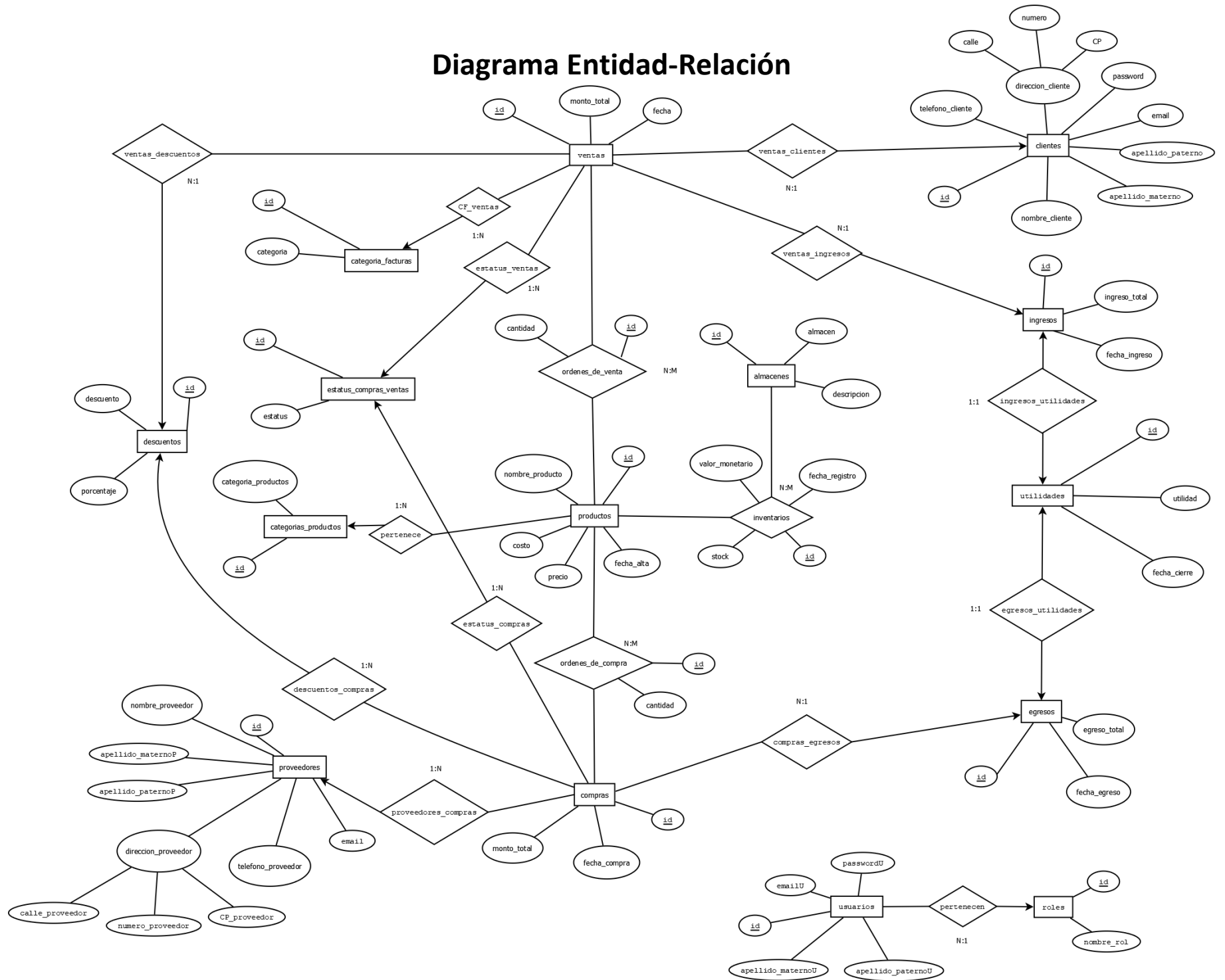
DESCUENTOS		
Descripción	Define la categoría de descuentos que aplicamos a los clientes o que los proveedores nos apliquen y el porcentaje aplicado del mismo.	
Atributo	Descripción	Tipo de dato
id	identificador del descuento	INT autoincrement(PK)
descuento	categoría del descuento	VARCHAR(45)
porcentaje	porcentaje que representa el descuento del monto total	DECIMAL(10,2)

CATEGORÍAS FACTURA		
Descripción	Define una serie de categorías que nos permite emitir facturas a los clientes según sea el concepto o motivo de venta.	
Atributo	Descripción	Tipo de dato
id	identificador de la categoría	INT autoincrement(PK)
categoria	estado de la compra/venta	VARCHAR(45)

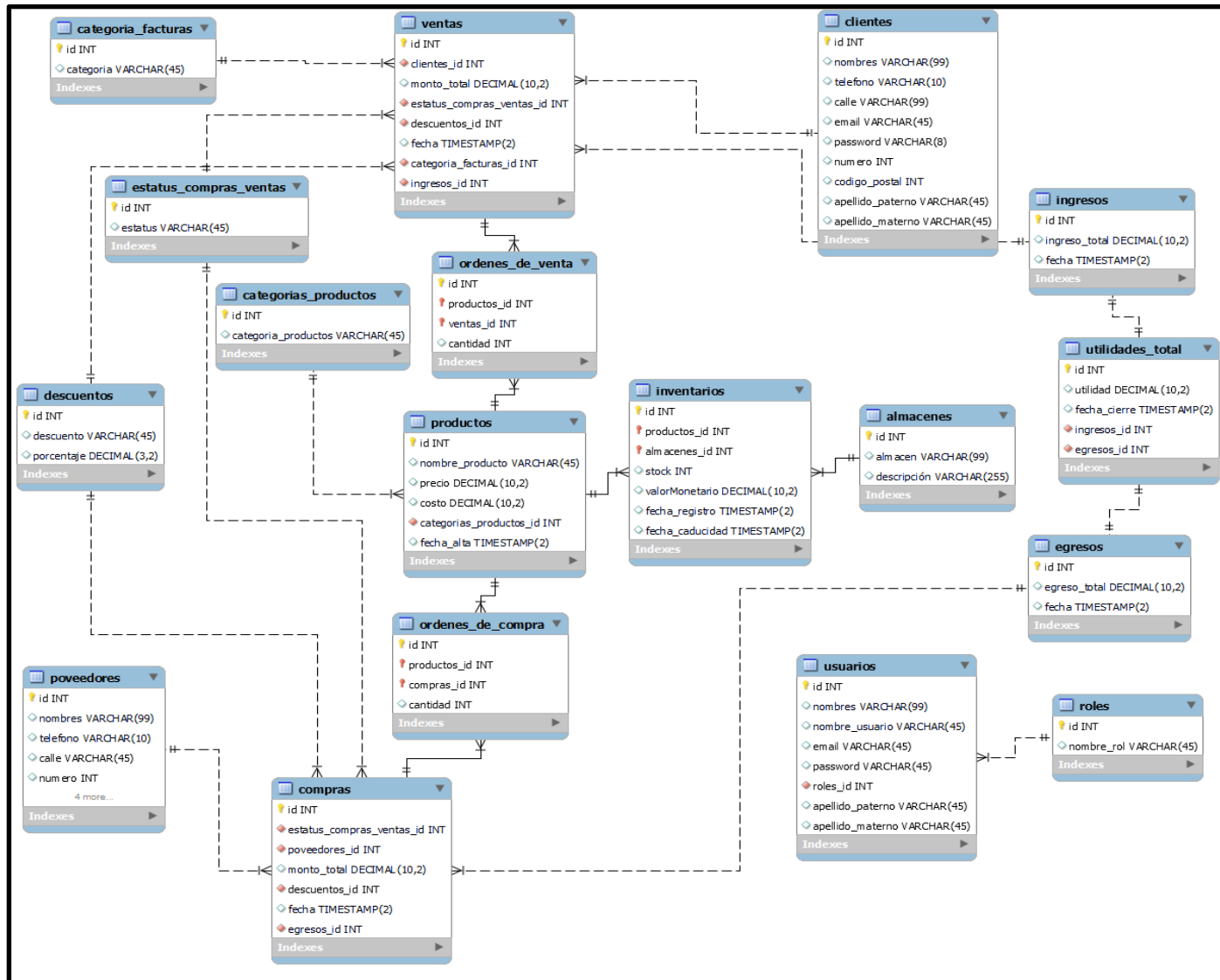
USUARIOS		
Descripción	Son los encargados del uso del sistema final, podrán realizar actividades de registro, modificación, consultas a otras entidades según sea su rol de usuario.	
Atributo	Descripción	Tipo de dato
id	identificador del usuario	INT autoincrement(PK)
nombre	nombre real del usuario	VARCHAR(45)
nombre_usuario	username o alias del usuario se utiliza para entrar al sistema	VARCHAR(45)
email	correo electronico	VARCHAR(45)
password	contraseña	VARCHAR(45)
roles_id	identificador del rol al que pertenece el usuario	FK

ROLES		
Descripción	Define la jerarquía de privilegios de la entidad usuarios.	
Atributo	Descripción	Tipo de dato
id	identificador del rol	INT autoincrement(PK)
nombre_rol	nombre representativo del rol	VARCHAR(45)

Diagrama Entidad-Relación



Modelo Relacional



Tablas

ventas								
id (pk)	clientes _id(fk)	montoTotal	estatus_compras _ventas_id(fk)	categoria _facturas _id(fk)	descuentos _id(fk)	fecha	categorias _facturas_id(fk)	ingresos_id(fk)

descuentos		
id (pk)	descuento	porcentaje

Categoria_facturas	
id(pk)	categoria

clientes						
id (pk)	nombre_cliente	telefono	calle	password	numero	codigo_postal

ingresos		
id(pk)	ingreso_total	fecha

estatus_compras_ventas	
id(pk)	estatus

utilidades_total				
id(pk)	utilidad	fecha_cierre	ingresos_id(fk)	egresos_id(fk)

egresos		
id(pk)	egreso_total	fecha

ordenes_de_venta			
id(pk)	productos_id(fk)	ventas_id(fk)	cantidad

categoria_productos	
id(pk)	categoria_productos

productos					
id (pk)	nombre_producto	precio	costo	categorias_producto_id(fk)	fecha_alta

ordenes_de_compra			
id(pk)	productos_id(fk)	compras_id(fk)	cantidad

compras						
id (pk)	estatus_compras_ventas(fk)	proveedores(fk)	monto_total	descuentos_id(fk)	fecha	egresos_id(fk)

usuarios					
id (pk)	nombre	nombre_usuario	email	password	roles_id(fk)

roles	
id(pk)	nombre_rol

inventarios					
id (pk)	productos_id(fk)	almacenes_id(fk)	stock	valorMonetario	fecha_registro

almacenes		
id(pk)	almacen	descripcion

proveedores						
id (pk)	nombre_proveedor	telefono	calle	numero	codigo_postal	email

Normalización

Las siguientes 6 tablas (almacenes, categorias_facturas, categorías_productos, estatus_compra_venta, descuentos y roles).

Cumplen con la primera forma normal ya que tiene un valor simple en los atributos, en descripción de la tabla almacenes como el nombre lo dice es una descripción así que es válido.

Cumplen con la segunda forma normal ya que los atributos son dependientes de la llave.

Cumplen con la tercera forma normal ya que no hay relación de los atributos entre ellos por lo que no habrá que hacer otra tabla.

Almacenes

id	almacen	descripcion
1	Disponibles	En este almacén se guardan todos los productos lis...
2	Sin stock	En este almacén se guardan todos los productos con...
3	Preventas	En este almacén se registraran productos que tendr...
4	Proximos a caducar	En este almacén se guardan todos los productos que...
5	Prestamos	En este almacén se guardaran los productos que ven...
6	Productos reacondicionados	En este almacén se guardan todos los productos que...

categorias_facturas

id	categoria_factura
1	Adquisición de mercancías
2	Devoluciones, descuentos o bonificaciones
3	Gastos en general
4	Construcciones
5	Mobiliario y equipo de oficina por inversiones
6	Equipo de transporte
7	Equipo de cómputo y accesorios
8	Dados, troqueles, moldes, matrices y herramental
9	Comunicaciones telefónicas
10	Comunicaciones satelitales
11	Otra maquinaria y equipo
12	Pagos
13	Nómina
14	Sin Efectos Fiscales

categorias_productos

id	categoria_productos
1	otra
2	Abarrotes
3	Lacteos
4	Carnes
5	Frutas y Verduras
6	Limpieza

estatus_compras_ventas

id	estatus
1	Creada
2	En Preparación
3	En camino
4	Finalizada
5	Pendiente de pago

descuentos

id	descuento	porcentaje
1	No aplica	0.00
2	cliente frecuente	10.00
3	temporada baja	5.00
4	trabajador	7.00
5	cliente nuevo	3.00
6	buen fin	5.00

roles

id	rol
1	Administrador
2	Usuario_ventas
3	Usuario_inventarios
4	Usuario_contabilidad

Tabla clientes

Tiene un valor simple en los atributos así que cumple con la primera forma normal, la calle cuenta como un valor.

Cumplen con la segunda forma normal ya que los atributos son dependientes de la llave.

No existe relación de los atributos entre ellos por lo que no habrá que hacer otra tabla, por lo tanto, cumple con la tercera forma.

clientes

id	nombres	apellido_paterno	apellido_materno	telefono	calle	numero	codigo_postal	email	password
1	Público en general	No aplica	No aplica	No aplica	No aplica	0	0	No aplica	No aplica
2	Juan	de la Barrera	Gomez	3335698462	13 de septiembre de 1810	93	46554	juanito@gmail.com	No aplica
3	Juan	de la Barrera	Gonzalez	333569889	13 de septiembre de 1810	91	46554	juanito97@gmail.com	No aplica
4	Juana	de la Barrera	Gomez	3335698479	13 de septiembre de 1810	95	46554	juanita@gmail.com	No aplica
5	Joaquin	Camacho	Smith	3155698462	16 de septiembre de 1810	93	46556	Smith@gmail.com	No aplica

La tabla Ventas y Compras

Las tablas ventas y compras son una relación con otras entidades, como las id dentro de las tablas ventas se pueden repetir entonces se agrega un id de ventas para que no se repitan, así se está cumpliendo la primera forma normal, cumple la segunda forma normal ya que las entidades tienen dependencia con la llave primaria, cumple la tercera forma normal porque las entidades no tienen relación con otra llave que no sea la primaria.

ventas

id	monto_total	id_clientes	id_estatus	id_categoria_factura	id_descuentos	id_ingresos
1	40.00	1	4		1	1
2	40.00	1	4		1	2
3	20.00	1	4		1	3
4	30.00	1	4		1	4
5	10.00	1	4		1	4

compras

id	monto_total	id_estatus	id_proveedores	id_descuentos	id_egresos
1	20.00	4		1	1
2	20.00	4		1	2
3	10.00	4		1	3
4	15.00	4		1	4
5	5.00	4		1	4

La tabla productos

Tiene un único dato dentro de las entidades (1FN), las entidades tienen dependencia con la llave primaria(2FN), las entidades no tienen relación entre ellas o con la llave foránea(3FN).

productos

id	producto	precio	id_categorias	costo
1	otro	0.00	1	0.00
2	Pure de tomate	20.00	2	10.00
3	Azúcar	20.00	2	10.00
4	Cereal de maíz	20.00	2	10.00
5	Huevo	20.00	2	10.00
6	Pastas	20.00	2	10.00
7	Frijol	20.00	2	10.00
8	Leche 1L	10.00	3	5.00
9	Leche 2L	20.00	3	10.00
10	Yogurt 500ml	10.00	3	5.00

Las tablas Utilidades, ventas, compras, inventarios y son una relación con otras entidades, como las id dentro de las tablas ventas se pueden repetir entonces se agrega un id de ventas para que no se repitan, así se está cumpliendo la primera forma normal, cumple la segunda forma normal ya que las entidades tienen dependencia con la llave primaria, la tercera forma normal se cumple porque las entidades no tienen relación con otra llave que no sea la primaria.

utilidades

id	id_ingresos	id_egresos	utilidad
1	1	1	20.00
2	2	2	20.00
3	3	3	10.00
4	4	4	20.00
5	5	5	0.00

ordenesdecompra

id	id_productos	id_compras	cantidad
1	2	1	1
2	3	1	1
3	4	1	1
4	5	2	1
5	6	2	1
6	7	3	1
7	8	4	1
8	9	4	1
9	10	5	1

ordenesdeventa

id	id_productos	id_ventas	cantidad
1	2	1	1
2	3	1	1
3	4	1	1
4	5	2	1
5	6	2	1
6	7	3	1
7	8	4	1
8	9	4	1
9	10	5	1

inventarios

id	stock	monto	id_productos	id_almacenes	fecha_caducidad
1	1	10.00	2	1	2024-01-01
2	1	10.00	3	1	2024-01-01
3	1	10.00	4	1	2024-01-01
4	1	10.00	5	1	2024-01-01
5	1	10.00	6	1	2024-01-01
6	1	10.00	7	1	2024-01-01
7	1	5.00	8	1	2024-01-01
8	1	10.00	9	1	2024-01-01
9	1	5.00	10	1	2024-01-01

La tabla proveedores

La tabla proveedores tiene un único valor dentro de los atributos, la calle cuenta como un único elemento ya que se trata del mismo dato por lo tanto es válido(1FN), los atributos tienen dependencia con la llave primaria(2FN), los atributos no son dependientes de otros atributos por lo que separar la tabla no es necesario(3FN).

proveedores

id	nombres	apellido_paterno	apellido_materno	telefono	calle	numero	codigo_postal	email
1	Otro	No aplica	No aplica	No aplica	No aplica	0	0	No aplica
2	Jessica Noemí	Gallardo	Valencia	4213569846	Rinconcito en el cielo	102	55465	Jessyjoey@gmail.com
3	Carnes	del Norte SA	No aplica	4213569566	Desierto	10	55489	carnesdelnorte@gmail.com
4	Maestro	Limpio SRL	No aplica	4783569846	Insurgentes	26	85465	lipiamas@gmail.com
5	Joaquín	Lopez	Doriga	8913569846	Revolución	20	55475	noticieros@gmail.com

La tabla users tiene un único valor dentro de los atributos, created_at y updated_at el valor que almacenan cuenta como un único elemento(1FN),.

Los atributos tienen dependencia con la llave primaria(2FN).

Los atributos no son dependientes de otros atributos ni de la llave foránea (roles_id) por lo que separar la tabla no es necesario(3FN).

Users

id	username	nombres	apellido_paterno	apellido_materno	email	password	roles_id
1	Admin	Admin	No aplica	No aplica	admin@gmail.com	\$2y\$10\$hPa3pvGK/86UeiyTgUFzSOErRsk5noRhqQh6F75EUX...	1

La tabla ingresos

La tabla ingresos tiene un único valor almacenado dentro de los atributos(1FN), las tres entidades tienen dependencia con la llave primaria(2FN), no hay una relación fuerte entre los atributos como para hacer otra tabla(3FN).

ingresos

id	ingreso
1	40.00
2	40.00
3	20.00
4	40.00
5	0.00

Enunciados para las sentencias SQL

1. Un nuevo día comienza, llega nuestro primer cliente y con él, nuestra primera venta. Procedemos a crear la nueva venta y agregar cada uno de los productos dentro de la orden asignada a la venta.

```
1 INSERT INTO `ordenesdeventa` (`id`, `id_productos`, `id_ventas`, `cantidad`, `created_at`, `updated_at`)
VALUES (NULL, '2', '1', '1', '2023-04-04 09:40:03', NULL), (NULL, '7', '1', '2', '2023-04-04 09:41:03',
NULL);
```

```
1 INSERT INTO `ordenesdeventa` (`id`, `id_productos`, `id_ventas`, `cantidad`, `created_at`, `updated_at`)
VALUES (NULL, '10', '1', '2', '2023-04-04 09:43:25', NULL), (NULL, '18', '1', '1', '2023-04-04 09:44:25',
NULL);
```

					id	id_productos	id_ventas	cantidad	created_at	updated_at
<input type="checkbox"/>		Editar		Copiar		Borrar	3	2	1	1 2023-04-04 09:40:03 NULL
<input type="checkbox"/>		Editar		Copiar		Borrar	4	7	1	2 2023-04-04 09:41:03 NULL
<input type="checkbox"/>		Editar		Copiar		Borrar	5	10	1	2 2023-04-04 09:43:25 NULL
<input type="checkbox"/>		Editar		Copiar		Borrar	6	18	1	1 2023-04-04 09:44:25 NULL

						id	monto_total	id_clientes	id_estatus	id_categoria_factura	id_descuentos	id_ingresos	created_at	updated_at
<input type="checkbox"/>		Editar		Copiar		Borrar	1	56.12	1	1	1	1	1 NULL	NULL

2. Un nuevo cliente ha llegado, le ofrecemos el ser cliente registrado y este acepta, procedemos a registrarlo en el sistema

```
1 INSERT INTO `clientes` (`id`, `nombres`, `apellido_paterno`, `apellido_materno`, `telefono`, `calle`, `numero`, `codigo_postal`,
`email`, `password`, `created_at`, `updated_at`) VALUES (NULL, 'Maria', 'Gil', 'Guerrero', '3332313034', 'Lopez Cotilla', '43',
'46600', 'lopezRossy94@example.com', '$2y10%&In3nBh4/D.48eeF0ozK', NULL, NULL);
```

3	Maria	Gil	Guerrero	3332313034	Lopez Cotilla	43	46600	lopezRossy94@example.com	\$2y10%&In3nBh4/D.48eeF0ozK	2023-04-10 09:33:39	NULL
---	-------	-----	----------	------------	---------------	----	-------	--------------------------	-----------------------------	---------------------	------

3. El gerente del supermercado necesita una reimpresión del ticket de compra de un cliente ya registrado llamado Juan, el cliente dice que hizo la compra el día 10 de abril del 2023

```
SELECT nombres, apellido_paterno, producto, precio FROM clientes, productos,
ordenesdeventa, ventas WHERE ventas.created_at LIKE '%2023-04-10%' AND
ordenesdeventa.id_ventas = ventas.id AND productos.id = ordenesdeventa.id_productos AND
clientes.id = ventas.id_clientes;
```

☐ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25

Opciones extra

nombres	apellido_paterno	producto	precio
Juan	de la Barrera	Huevo	20.00
Juan	de la Barrera	Pastas	20.00

4. Ha llegado el fin de mes, el súper requiere de abastecimiento, necesitamos los datos de uno de nuestros proveedores para contactarlo y realizar el pedido

```
SELECT * FROM `proveedores` WHERE nombres LIKE 'Miguel';
```

☐ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Opciones extra

	id	nombres	apellido_paterno	apellido_materno	telefono	calle	numero	codigo_postal	email	created_at	updated_at
<input type="checkbox"/>	2	Miguel	Santana	Hernandez	1234567890	Corregidora	534	810	abarrosesMSH@example.com	2023-04-01 12:13:54	NULL

5. para una documentación necesaria para un reporte mensual necesitamos la cantidad total generada por las ventas dentro del mes de abril del 2023

```
SELECT SUM(monto_total) FROM `ventas` WHERE created_at LIKE '%2023-04%';
```

☐ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

Opciones extra

SUM(monto_total)

1374.52

6. Un cliente quiere abrir una tienda de abarrotes y quiere que nosotros seamos su proveedor. Le entregaremos nuestro catálogo de productos dentro de la categoría de abarrotes.

```
SELECT categoria_productos, producto, precio FROM productos, categorias_productos WHERE
categorias_productos.categoria_productos = 'Abarrotes' AND productos.id_categorias =
categorias_productos.id;
```

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Número de filas: 25 ▾

Opciones extra

categoria_productos	producto	precio
Abarrotes	Pure de tomate	20.00
Abarrotes	Azúcar	20.00
Abarrotes	Cereal de maíz	20.00
Abarrotes	Huevo	20.00
Abarrotes	Pastas	20.00
Abarrotes	Frijol	20.00

7. Uno de nuestros proveedores nos recomendó un producto que ha estado en tendencia estas últimas semanas, después de verificarlo nos ha convencido, procederemos a hacer el registro de este

```
1 INSERT INTO `productos` (`id`, `producto`, `precio`, `id_categorias`, `costo`, `created_at`, `updated_at`) VALUES (NULL, 'Limpiador
XLR8', '32.50', '6', '25.00', '2023-04-10 12:40:47', NULL);
```

id	producto	precio	id_categorias	costo	created_at	updated_at
52	Limpiador XLR8	32.50	6	25.00	2023-04-10 12:40:47	NULL

8. A la semana siguiente que nuestro proveedor que nos recomendó el producto nos trae el primer cargamento para ver cómo se comportan las ventas respecto a este, haremos la compra y el inventariado de este

```
1 INSERT INTO `compras` (`id`, `monto_total`, `id_estatus`, `id_proveedores`, `id_descuentos`, `id_egresos`, `created_at`, `updated_at`)
VALUES ('2', '500.00', '4', '2', NULL, '1', '2023-04-09 13:02:57', NULL);
```

```
1 INSERT INTO `ordenesdecompra` (`id`, `id_productos`, `id_compras`, `cantidad`, `created_at`, `updated_at`) VALUES (NULL, '52', '2',
'20', '2023-04-16 12:58:36', NULL);
```

```
1 INSERT INTO `inventarios` (`id`, `stock`, `monto`, `id_productos`, `id_almacenes`, `fecha_caducidad`, `created_at`, `updated_at`)
VALUES (NULL, '20', '32.5', '52', '1', '2025-04-30', '2023-04-16 13:04:43', NULL);
```

id	stock	monto	id_productos	id_almacenes	fecha_caducidad	created_at	updated_at
2	20	32.50	52	1	2025-04-30	2023-04-16 13:04:43	NULL












9. Ha pasado temporada vacacional y requerimos saber solamente la cantidad de ingresos que hemos obtenido en las dos semanas referentes a semana santa y pascua

```
SELECT monto_total, created_at FROM `ventas` WHERE created_at BETWEEN '2023-04-02' AND '2023-04-16' ORDER BY created_at;
```

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Ordenar según la clave:

Opciones extra

			monto_total	created_at
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	394.50
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	562.30
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	134.60
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	56.12

10. Un producto está próximo a caducar, por tanto, procedemos a agregarlo a dicho almacén

```
1 INSERT INTO `inventarios` (`id`, `stock`, `monto`, `id_productos`,  
  `id_almacenes`, `fecha_caducidad`, `created_at`, `updated_at`) VALUES (NULL,  
  '5', '10.00', '10', '4', '2023-04-17', '2023-04-11 11:32:25', NULL);
```

```
SELECT producto, almacen, descripcion FROM almacenes, inventarios, productos WHERE  
inventarios.id_productos = 10 AND almacenes.id = inventarios.id_almacenes AND productos.id  
= inventarios.id_productos;
```

☐ Perfilando [[Editar en línea](#)] [[Editar](#)] [[Explicar SQL](#)] [[Crear código PHP](#)] [[Actualizar](#)]

☐ Mostrar todo | Número de filas: 25

Opciones extra

producto	almacen	descripcion
Yogurt 500ml	Proximos a caducar	En este almacén se guardan todos los productos que...

Conclusiones

El presente proyecto nos brinda las herramientas necesarias para la correcta administración de entradas y salidas de mercancías, a nivel base de datos, es decir se tiene la estructura relacionada del proyecto mas no se ha concluido la programación del sistema como tal.

Durante el desarrollo de esta nos encontramos con problemas como por ejemplo el almacenamiento masivo de registros en la tabla de ventas, ya que por cada producto se almacenaba un registro, por lo anterior se optó por incluir una tabla llamada orden de ventas donde se almacenarán todos los productos de una venta en específico, evitando así la saturación de información en la tabla ventas y mejor organización en la Base de datos.

Por decisión del equipo se optó por elaborar una base de datos que comprenda un punto de venta y un sistema de inventarios, proporcionando información tanto de compras como de ventas, y teniendo en cuenta estos dos factores se pueden obtener utilidades brutas.

Este proyecto tiene como finalidad desarrollar varios sistemas que se complementen entre sí, proporcionando a los usuarios un control total de sus recursos, ya sean mercancías, personal, maquinaria, equipos, etcétera.

Referencias

- [1] A. E. y. S. B. Navathe, Sistemas de bases de datos, 5ta Ed Addison Wesley, 2007.
- [2] H. F. K. y. A. Silverschatz, Fundamentos de bases de datos, 5ta Ed McGrawHill, 2006.
- [3] T. M. C. y. C. E. Begg, Sistemas de bases de datos, 4ta Ed Pearson, 2005.

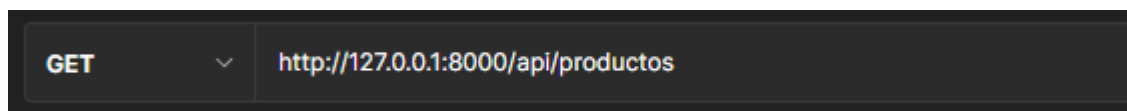
Avances del sistema

El sistema de punto de venta se encuentra en desarrollo actualmente el proyecto estará desarrollado en un framework de php llamado laravel en su versión 9, esta es una poderosa librería que nos ofrece un gran número de herramientas para el desarrollo web además de sistemas criptográficos que ayudan a la seguridad, se optó por usar este framework para el desarrollo backend y utilizar un framework de JavaScript conocido como vue para el desarrollo frontend permitiendo la interacción del usuario con el sistema en tiempo real sin necesidad de recargar página después de una petición a la base de datos.

Además de un sistema web, contaremos con una aplicación móvil para el cliente la cual le brindará la facilidad de realizar pedidos desde la comodidad de su hogar, esto pensado para negocios de comida rápida mas que nada.

Para poder comunicar vue con la base de datos se desarrollaron apis con laravel las cuales permiten el acceso a la base de datos por medio de peticiones get y post, añadiendo a esta un Jason web token para mayor seguridad en el manejo de datos.

Actualmente se tienen algunas apis desarrolladas por ejemplo la siguientes:



Este api nos arroja todos los productos registrados en la tabla productos, con un formato json como se muestra a continuación:

```

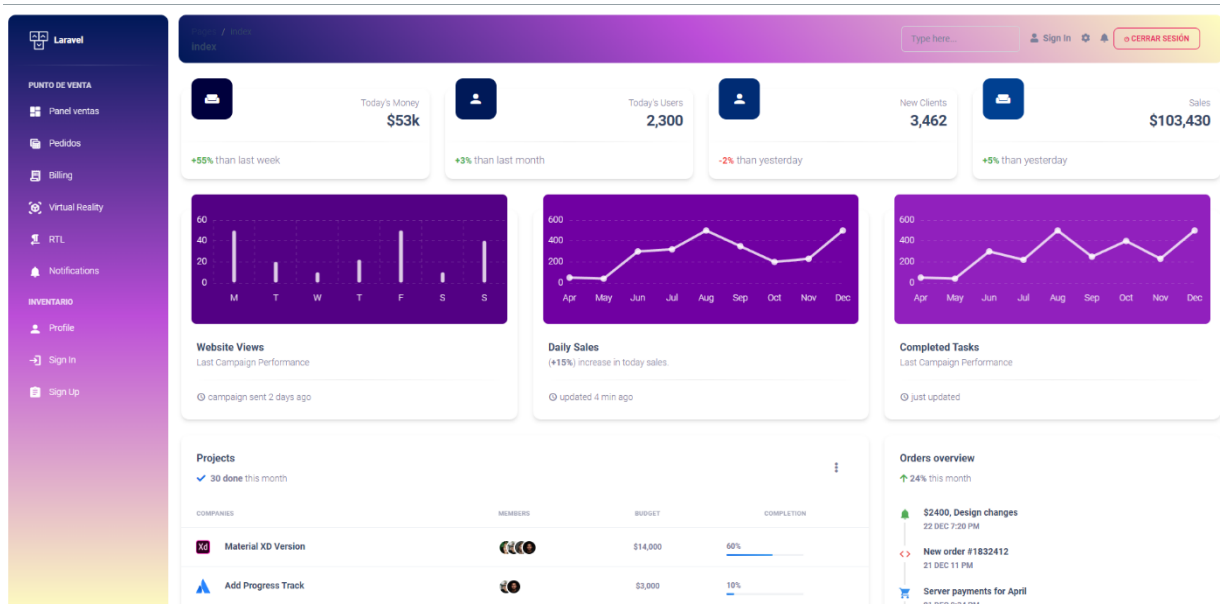
1  [
2  {
3      "id": 1,
4      "producto": "otro",
5      "precio": "0.00",
6      "id_categorias": 1,
7      "costo": "0.00",
8      "created_at": null,
9      "updated_at": null,
10     "categorias_productos_productos__rel": {
11         "id": 1,
12         "categoria_productos": "otra"
13     }
14 },
15 {
16     "id": 2,
17     "producto": "Queerat.",
18     "precio": "6.64",
19     "id_categorias": 4,
20     "costo": "2.55",
21     "created_at": "2023-03-02T22:14:54.000000Z",
22     "updated_at": "2023-03-02T22:14:54.000000Z",
23     "categorias_productos_productos__rel": {
24         "id": 4,
25         "categoria_productos": "Carnes"
26     }
27 },
28 {
29     "id": 3,
30     "producto": "Sed ipsa.",
31     "precio": "7.61",
32     "id_categorias": 2,
33     "costo": "3.70",

```

Como podemos observar gracias al ORM de php llamado eloquent el cual se utilizó para desarrollar las sentencias sql con código php podemos traer el producto con su relación a la tabla categorías_productos con su id y su nombre, lo que permite un manejo mas simple a la hora de mostrar datos al usuario.

Este es el modelo que se utilizo para todas las tablas, permite un mejor acomodo de los datos, la comunicación de vue con la base de datos y una doble protección de las peticiones gracias al Json Web Token.

Por parte del frontend solo se ha trabajado en la implementación de una platilla para el sistema y una estructura de bienvenida para el usuario:



Se pretende mostrar información importante al usuario al inicio de sesión dependiendo del rol que desempeñe en el negocio, por ejemplo, si es un usuario de ventas les mostraríamos estadísticas del producto que este en oferta, de los productos que falten ventas entre otras, si se tratará el caso de un usuario de inventarios, podríamos mostrarle estadísticas de productos próximos a agotarse, próximos a caducar entre otros.

The dashboard displays a sales panel with a table of sales data. The table has columns for AUTHOR, FUNCTION, TECHNOLOGY, and EMPLOYER. The data is as follows:

AUTHOR	FUNCTION	TECHNOLOGY	EMPLOYER
John Michael john@creative-tim.com	Manager Organization		23/04/18 Edit
Alexa Liras alex@creative-tim.com	Programator Developer		11/01/19 Edit
Laurent Perrier laurent@creative-tim.com	Executive Projects		19/09/17 Edit
Michael Levi michael@creative-tim.com	Programator Developer		24/12/08 Edit
Richard Gran richard@creative-tim.com	Manager Executive		04/10/21 Edit
Miriam Eric miriam@creative-tim.com	Programator Developer		14/09/20 Edit

At the bottom of the dashboard, there's a footer with copyright information: '© 2023, hecho por SoftWaves' and a list of names: 'José, Ignacio, Miguel, Vladimir, Geovanny'.

Dentro del panel de ventas nos aparecerán las ordenes de venta que han sido encargadas a través de la aplicación o las ventas que aun no tienen estatus de terminada, en el botón rojo podremos dar clic para realizar una nueva venta, la cual permitirá registrar productos por nombre, código de barras, id, etc.