



# S.I.G.T.O.

## Base de Datos 2

### ShovelWare

Rol	Apellido	Nombre	C.I	Email	Telefono
Coordinator	Villero	Ian	5.639.694-0	ianvillero@gmail.com	098333955
Sub-Coordinator	Rodriguez	Mathias	5.576.322-5	mathiasrodriguez0507@gmail.com	097 700 830
Member 1	Abrines	Santiago	5.628.130-9	santiabrinesm@gmail.com	092727953
Member 2	García	Franco	5.400.435-1	francogpersonal@gmail.com	095313490

**Docente: Carámbula, Leonardo**

Fecha de culminación: 5/11/2024

**Tercera Entrega**

**ShovelWare**

**ISBO**

**3BF**

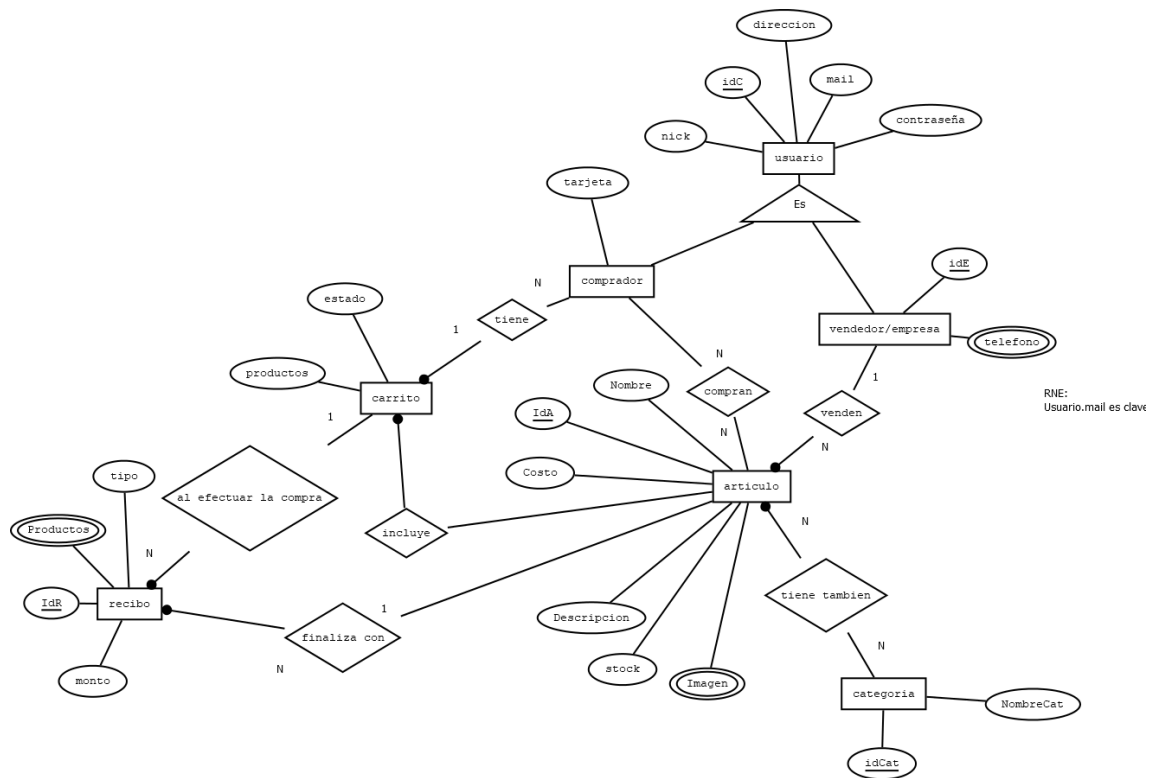


## Índice:

<b>Modelo conceptual (D.E.R.) 2da. Versión.....</b>	<b>3</b>
<b>Esquema Relacional.....</b>	<b>4</b>
Entidades:.....	4
Relaciones:.....	5
Normalización a 3ra forma:.....	5
<b>Diccionario de Datos:.....</b>	<b>7</b>
<b>Declaraciones SQL:.....</b>	<b>10</b>



## Modelo conceptual (D.E.R.) 2da. Versión.





# Esquema Relacional.

## Entidades:

Usuario: (IdC, mail, nick, contraseña, dirección)

Comprador: (IdC, mail, dirección, tarjeta)

IdC → usuario

mail → usuario

dirección → usuario

vendedorEmpresa: (IdC, mail, IdE, dirección, telefono\*)

IdC → usuario

mail → usuario

dirección → usuario

Artículo: (IdA, idCat, idE, nombre, descripción, imagen\*, costo, stock)

idCat → categoría

Categoría: (IdCat, NombreCat)

Recibo: (IdR, IdA, IdC, tipo, montoTotal, Productos\*)

IdA → artículo

Carrito(idC, idA, estado, productos)

idC → Comprador

**ShovelWare**

**ISBO**

**3BF**



## Relaciones:

Compran: (IdC, IdA)

Idc → Comprador

mail → Comprador

IdA → Artículo

~~al efectuar la compra~~ por totalidad

~~tiene también~~ por totalidad

~~Contiene~~ por totalidad

~~finaliza con~~ por totalidad

~~venden~~ por totalidad

## Normalización a 3ra forma:

Usuario: (IdC, mail, nick, contraseña)\*\*\*

Comprador: (IdC, mail, dirección, tarjeta)\*\*\*

IdC → usuario

mail → usuario

dirección → usuario

vendedorEmpresa: (IdC, mail, IdE, telefono)\*\*\*

IdC → usuario

mail → usuario

**ShovelWare**

**ISBO**

**3BF**



ArticuloId: (IdA, nombre, descripción, costo, stock)\*\*\*

Articulo: (IdA, imagen, idCat, idE)\*\*\*

Recibo: (IdR, Productos, IdA, IdC, tipo, montoTotal)\*\*\*

IdA → artículo

idC → carrito

CarritoArt(idA, productos)\*\*\*

idA → Artículo

CarritoCuenta(idC, estado)\*\*\*

idC → Comprador

Compran: (idC, mail, IdA)\*\*\*

Idc → Comprador

mail → Comprador

IdA → Artículo



## Diccionario de Datos:

Tabla	Atributos	Tipo	Largo	Restricciones	Descripción
Artículo	IdA	auto-increment	-	Clave primaria NOT NULL	identificador de artículo
	idCat	int	-	Clave foránea (categoría) NOT NULL	identificador de categoría
	idE	int	-	Clave foránea (vendedorEmpresa) NOT NULL	identificador de empresa
	imagen	varchar	150	-	Imagen a utilizar en el programa (se referencia, no se descarga)
Articuloid	IdA	int	-	Clave primaria Clave foránea(artículo)	identificador del artículo
	costo	int	-	NOT NULL	precio del producto
	descripción	varchar	150	-	descripción del producto
	nombre	varchar	30	NOT NULL	nombre del producto
	stock	int	-	NOT NULL	stock del producto
CarritoArt	idA	int	-	clave primaria clave foránea(Artículo)	identificador del artículo
	productos	varchar	150	-	productos en el carrito
CarritoCuenta	idC	int	-	clave primaria clave foránea(comprador) NOT NULL	identificador de cuenta



	estado	enum	2	-	estado del carrito (pendiente,en proceso)
Categoría	idCat	Autoincrement	-	Clave primaria	identificador de la categoría
	NombreCat	varchar	50	NOT NULL	nombre de la categoría
Comprador	idC	int	-	clave primaria clave foránea (usuario) NOT NULL	identificador de comprador
	mail	varchar	50	clave primaria clave foránea(usuario) NOT NULL	dirección mail de la cuenta
	dirección	varchar	100	NOT NULL	dirección real de la cuenta (ciudad)
	tarjeta	int	-	-	tarjeta que se utilizara (los números, no se guardará o se encriptará)
Compran	idC	int	-	Clave primaria clave foránea (comprador)	identificador de la cuenta
	mail	varchar	50	Clave primaria clave foránea (comprador)	mail de la cuenta que efectuo la compra
	idA	int	-	Clave primaria clave foránea (artículo)	id del artículo que va a comprar (en este caso, añadir al carrito)
Recibo	IdR	autoincrement	-	clave primaria	identificador del recibo
	productos	varchar	200	clave primaria	productos que hay en el recibo
	IdA	int	-	Clave	identificador de

**ShovelWare**

**ISBO**

**3BF**





				foránea(artículo)	artículo
	IdC	int	-	clave foránea(carrito)	identificador de la cuenta
	tipo	ENUM	2	NOT NULL	tipo del recibo (nota de devolución o E.Ticket)
	montoTotal	int	-	-	monto total de la compra (en pesos uruguayos o dólares)
vendedorEmpresa	IdC	int	-	clave primaria clave foránea (usuario)	identificador de cuenta de la empresa
	mail	varchar	50	clave primaria clave foránea (usuario)	mail de la empresa
	IdE	autoincrement	-	clave primaria	identificador de la empresa
	telefono	varchar	30	clave primaria	telefono de la empresa
Usuario	idc	autoincrement	-	clave primaria	identificador de la cuenta
	mail	varchar	50	clave primaria	mail de la cuenta
	nick	varchar	50	NOT NULL	nick de la cuenta
	contraseña	varchar	50	NOT NULL	contraseña de la cuenta



## Declaraciones SQL:

```
CREATE DATABASE CompraMaster;
```

```
USE CompraMaster;
```

```
-- Tabla Categoría
```

```
CREATE TABLE IF NOT EXISTS categoria (
```

```
    idCat INT AUTO_INCREMENT,
```

```
    nombreCat VARCHAR(50),
```

```
    PRIMARY KEY(idCat)
```

```
);
```

```
-- Tabla Usuario con mail único
```

```
CREATE TABLE IF NOT EXISTS usuario (
```

```
    idC INT AUTO_INCREMENT,
```

```
    mail VARCHAR(50) UNIQUE NOT NULL, -- Correo único
```

```
    nick VARCHAR(50) NOT NULL,
```

```
    contraseña VARCHAR(50),
```

```
    dirección VARCHAR(100), -- Dirección movida aquí para su uso tanto en Comprador como en  
    VendedorEmpresa
```

```
    PRIMARY KEY(idC)
```

```
);
```

```
-- Tabla Comprador, sin repetir dirección y con tarjeta como VARCHAR(16)
```

```
CREATE TABLE IF NOT EXISTS comprador (
```

```
    idC INT,
```

**ShovelWare**

**ISBO**

**3BF**



```
tarjeta VARCHAR(16) NOT NULL, -- Tarjeta con longitud estándar

FOREIGN KEY (idC) REFERENCES usuario(idC),

PRIMARY KEY(idC)

);

-- Tabla VendedorEmpresa con dirección eliminada y teléfono añadido

CREATE TABLE IF NOT EXISTS vendedorEmpresa (

    idC INT NOT NULL,

    idE INT AUTO_INCREMENT,

    telefono VARCHAR(15), -- Asumido como VARCHAR en lugar de INT para cubrir múltiples
formatos

    FOREIGN KEY (idC) REFERENCES usuario(idC),

    PRIMARY KEY (idE) -- Clave primaria única para idE

);

-- Tabla Artículo separada por información de artículo y la imagen como VARCHAR

CREATE TABLE IF NOT EXISTS articulo (

    idA INT AUTO_INCREMENT,

    idCat INT NOT NULL,

    idE INT NOT NULL,

    imagen VARCHAR(150), -- Ruta a imagen almacenada como VARCHAR

    FOREIGN KEY (idCat) REFERENCES categoria(idCat),

    FOREIGN KEY (idE) REFERENCES vendedorEmpresa(idE),

    PRIMARY KEY(idA)

);
```



-- Tabla ArticuloId con detalles del artículo

```
CREATE TABLE IF NOT EXISTS articuloId (  
    idA INT,  
    nombre VARCHAR(30) NOT NULL,  
    descripcion VARCHAR(150),  
    costo DECIMAL(10, 2) NOT NULL, -- Costo como decimal  
    stock INT NOT NULL,  
    FOREIGN KEY (idA) REFERENCES articulo(idA),  
    PRIMARY KEY(idA)  
);
```

-- Tabla CarritoArt para los artículos y productos en carrito

```
CREATE TABLE IF NOT EXISTS carritoArt (  
    idA INT,  
    productos TEXT, -- Lista de productos (VARCHAR podría ser suficiente aquí, pero puedes  
    cambiar a TEXT si esperas cadenas largas)  
    FOREIGN KEY (idA) REFERENCES articulo(idA),  
    PRIMARY KEY (idA)  
);
```

-- Tabla CarritoCuenta para el estado de cada carrito

```
CREATE TABLE IF NOT EXISTS carritoCuenta (  
    idC INT NOT NULL,  
    estado ENUM ('pendiente', 'en proceso', 'completado', 'cancelado') DEFAULT 'pendiente',  
    FOREIGN KEY (idC) REFERENCES comprador(idC),  
    PRIMARY KEY (idC)
```



```
);
```

```
-- Tabla Compran, conexión entre compradores y artículos
```

```
CREATE TABLE IF NOT EXISTS compran (
```

```
idC INT,
```

```
idA INT,
```

```
FOREIGN KEY (idC) REFERENCES comprador(idC),
```

```
FOREIGN KEY (idA) REFERENCES articulo(idA),
```

```
PRIMARY KEY(idC, idA)
```

```
);
```

```
-- Tabla Recibo con tipo de recibo y montoTotal, productos como TEXT para mayor flexibilidad
```

```
CREATE TABLE IF NOT EXISTS recibo (
```

```
idR INT AUTO_INCREMENT,
```

```
productos TEXT, -- Cambiado a TEXT para listas largas de productos
```

```
idA INT,
```

```
idC INT,
```

```
tipo ENUM ('Nota de devolución', 'E.Ticket'),
```

```
montoTotal DECIMAL(10, 2) NOT NULL, -- Monto total del recibo
```

```
FOREIGN KEY (idA) REFERENCES articulo(idA),
```

```
FOREIGN KEY (idC) REFERENCES carritoCuenta(idC),
```

```
PRIMARY KEY (idR)
```

```
);
```



**ShovelWare**

**ISBO**

**3BF**