

**PROYECTO FINAL DE BASE DE DATOS**  
**ENTREGA FINAL**

**Kevin Andres Bejarano Tello 2067678 - 3743**

**Yessica Fernanda Villa Nuñez 2266301-3743**

**Manuel Antonio Vidales Duran 2155481-3743**

**Karen Jhulieth Grijalba Ortiz 2259623-3743**

**UNIVERSIDAD DEL VALLE SEDE TULUÁ**  
**PROGRAMA ACADÉMICO DE INGENIERÍA DE SISTEMAS**  
**2023**

# 1

## 1. REQUERIMIENTOS FUNCIONALES

### **Gestión de Uniformes:**

Registrar diferentes tipos de uniformes (camisa de diario, camibuso de Ed física, pantalón de diario, sudadera, pantaloneta, jardinera, etc.).

Registrar las características específicas de cada tipo de uniforme, como color, tipo de tela, escudo.

Asociar cada uniforme a los colegios para los que se confeccionan.

### **Gestión de Colegios:**

Registrar información de los colegios, incluyendo nombre.

### **Gestión de Proveedores:**

Registrar información de los proveedores de materias primas, como NIT, nombre, dirección, teléfono y nombre de contacto.

Especificar los productos que suministra cada proveedor.

### **Gestión de Materias Primas:**

Registrar información sobre las materias primas, incluyendo un código único, tipo, descripción, cantidad en existencia y unidad de medida (metros, unidades, tubos, etc.).

### **Gestión de Productos Terminados:**

Registrar productos terminados, asignándoles un código único, descripción, talla, género y precio de venta.

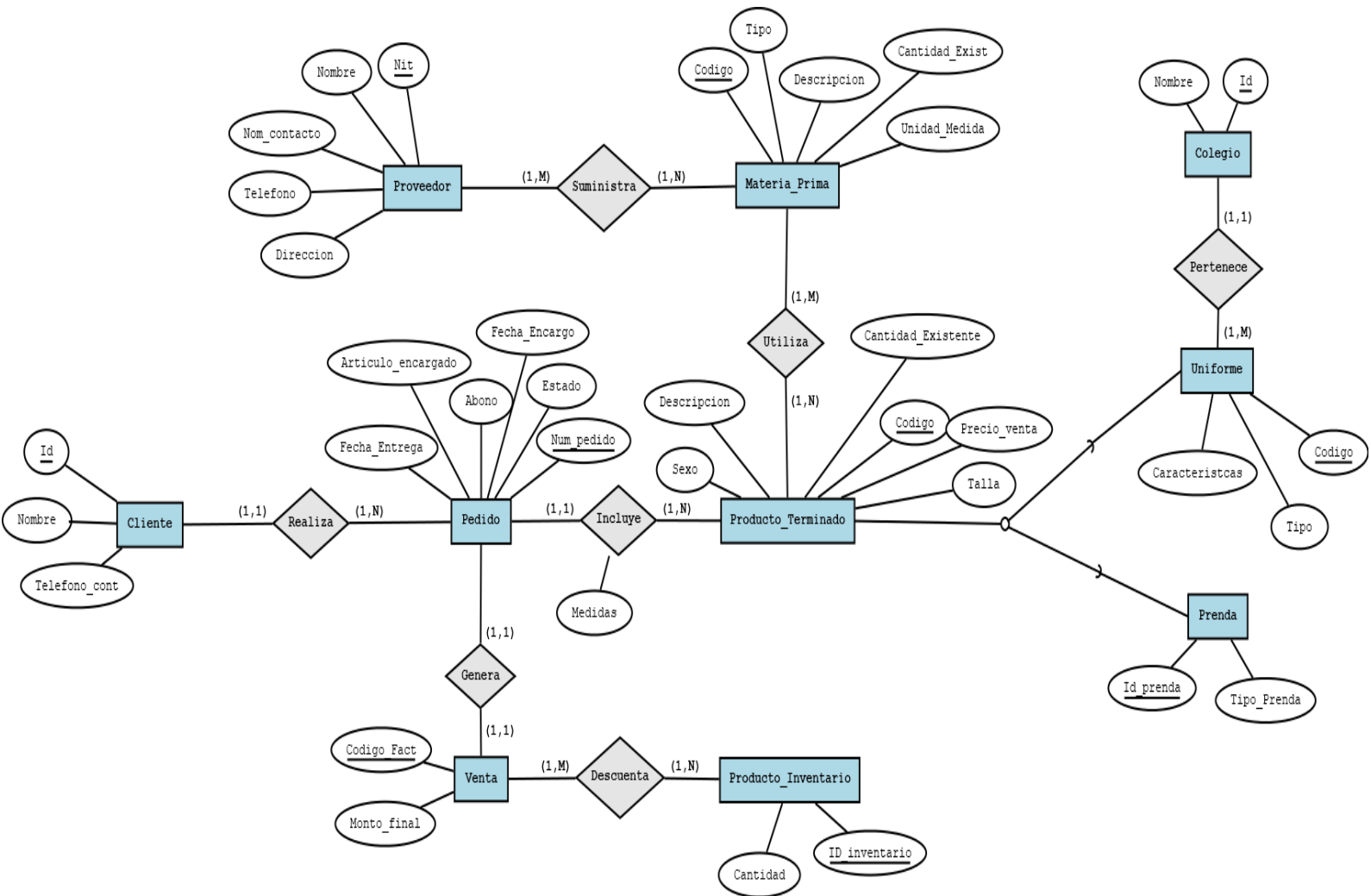
Llevar un seguimiento de la cantidad en existencia de productos terminados en el inventario y en la tabla descuento.

**Gestión de Pedidos:**

- Permitir a los clientes hacer encargos (pedidos) de uniformes o prendas de vestir.
- Registrar información detallada de cada pedido, incluyendo el cliente, el artículo encargado, la fecha del encargo y la fecha probable de entrega.
- Registrar abonos (anticipos) a la cuenta de los clientes.
- Generar un número único para cada pedido.
- Asociar cada pedido con los productos solicitados (uniformes o prendas de vestir).

## 2. MODELO CHEN

Herramienta utilizada para su diseño DIA.



### 3. MODELO RELACIONAL

**CLIENTE**(Id,Nombre,Telefono\_Cont)

**PEDIDO**(Num\_pedido, Id\_Cliente, Estado, Abono, Fecha\_Encargo, Articulo\_encargado, Fecha\_Entrega)  
Id\_Cliente -----> CLIENTE.Id

**VENTA**(Codigo\_Fact, Num\_pedido, Id\_Cliente, Monto\_Final)  
Num\_pedido -----> PEDIDO.Num\_pedido  
Id\_Cliente -----> PEDIDO.iD\_Cliente

**PRODUCTO\_TERMINADO**(Codigo, Num\_pedido, Descripcion, Cantidad\_Existente, Sexo,Precio\_venta, Talla, Medidas)  
Num\_pedido -----> PEDIDO.Num\_pedido

**INVENTARIO**(ID\_inventario, Cantidad)  
Codigo -----> Producto\_Terminado.Codigo  
Cantidad -----> Producto\_Terminado.Cantidad  
Descripcion -----> Producto\_Terminado.Descripcion

**UNIFORME**(Codigo,Codigo\_Prod, Id\_Colegio, Tipo, Caracteristicas)  
Id\_Colegio -----> COLEGIO.Id  
Codigo\_Prod -----> PRODUCTO\_TERMINADO.Codigo

**PRENDA**(Id\_prenda, Codigo\_Prod, Tipo\_prenda)  
Codigo\_Prod -----> PRODUCTO\_TERMINADO.Codigo

**COLEGIO**(Id, Nombre)

**MATERIA\_PRIMA**(Codigo, Tipo, Descripcion, Cantidad\_Exist, Unidad\_Medida)

**PROVEEDOR**(Nit, Nombre, Nom\_contacto, Telefono, Direccion)

**DESCUENTA**(Codigo\_Fact, Codigo)  
Codigo\_Fact -----> VENTA.Codigo\_Fact  
Codigo -----> PRODUCTO\_INVENTARIO.Codigo

**UTILIZA**(Cod\_prod, Cod\_Materia)  
Cod\_prod -----> PRODUCTO\_TERMINADO.Codigo  
Cod\_Materia -----> MATERIA\_PRIMA.Codigo

**SUMINISTRA**(Nit\_Proveedor, Codigo\_Materia)  
     Nit\_Proveedor -----> PROVEEDOR.Nit  
     Codigo\_Materia -----> MATERIA\_PRIMA.Codigo

#### 4. NORMALIZACIÓN

**1FN:** El modelo ya se encuentra en primera forma normal.

**2FN:** El modelo ya se encuentra en segunda forma normal.

**3FN:**

**FECHAS** ( Fecha\_Encargo, Fecha\_Entrega)

**PEDIDO**(Num\_pedido, Id\_Cliente, Estado, Abono, Fecha\_Encargo, Articulo\_encargado)  
     Id\_Cliente -----> CLIENTE.Id  
     Fecha\_Encargo -----> FECHAS.Fecha\_Encargo

**CLIENTE**(Id,Nombre,Telefono\_Cont)

**VENTA**(Codigo\_Fact, Num\_pedido, Id\_Cliente, Monto\_Final)  
     Num\_pedido -----> PEDIDO.Num\_pedido  
     Id\_Cliente -----> PEDIDO.iD\_Cliente

**PRODUCTO\_TERMINADO**(Codigo, Num\_pedido, Descripcion, Cantidad\_Existente, Sexo, Precio\_venta, Talla, Medidas)  
     Num\_pedido -----> PEDIDO.Num\_pedido

**PRODUCTO\_INVENTARIO**(ID\_inventario, Cantidad)

**UNIFORME**(Codigo,Codigo\_Prod, Id\_Colegio, Tipo, Caracteristicas)  
     Id\_Colegio -----> COLEGIO.Id  
     Codigo\_Prod -----> PRODUCTO\_TERMINADO.Codigo

**PRENDA**(Id\_prenda, Codigo\_Prod, Tipo\_prenda)  
     Codigo\_Prod -----> PRODUCTO\_TERMINADO.Codigo

**COLEGIO**(Id, Nombre)

**MATERIA\_PRIMA**(Codigo, Tipo, Descripcion, Cantidad\_Exist, Unidad\_Medida)

**PROVEEDOR**(Nit, Nombre, Nom\_contacto, Telefono, Direccion)

**DESCUENTA**(Codigo\_Fact, ID\_inventario)  
     Codigo\_Fact -----> VENTA.Codigo\_Fact  
     ID\_inventario -----> PRODUCTO\_INVENTARIO.ID\_inventario

**UTILIZA**(Cod\_prod, Cod\_Materia)

Cod\_prod -----> PRODUCTO\_TERMINADO.Codigo

Cod\_Materia -----> MATERIA\_PRIMA.Codigo

**SUMINISTRA**(Nit\_Proveedor, Codigo\_Materia)

Nit\_Proveedor -----> PROVEEDOR.Nit

Codigo\_Materia -----> MATERIA\_PRIMA.Codigo

## **TABLAS**

### **TABLA CLIENTE**

**CREATE TABLE CLIENTE (**

id **INT PRIMARY KEY,**

Nombre **VARCHAR(50),**

Telefono **VARCHAR(15)**

**);**

### **TABLA PEDIDO**

**CREATE TABLE PEDIDO (**

Num\_pedido **INT PRIMARY KEY,**

Id\_Cliente **INT NOT NULL,**

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

Abono **FLOAT NOT NULL,**

Fecha\_Encargo **DATE NOT NULL,**

Fecha\_Entrega **DATE,**

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

**FOREIGN KEY (Id\_Cliente) REFERENCES CLIENTE (ID) ON DELETE  
CASCADE ON UPDATE CASCADE**

**);**

### **TABLACOLEGIO**

**CREATE TABLE COLEGIO (**

Id **INT PRIMARY KEY,**

Nombre **VARCHAR(255) NOT NULL**

**);**

## **TABLA PRODUCTO\_TERMINADO**

```
CREATE TABLE PRODUCTO_TERMINADO (  
    Codigo VARCHAR(50) PRIMARY KEY,  
    Num_pedido INT,  
    Descripcion VARCHAR(255),  
    Cantidad_Existente INT,  
    Sexo VARCHAR(50),  
    Precio_venta FLOAT NOT NULL,  
    Talla VARCHAR(10),  
    Medidas VARCHAR(255),  
    FOREIGN KEY (Num_pedido) REFERENCES PEDIDO (Num_pedido) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

## **TABLA UNIFORME**

```
CREATE TABLE UNIFORME (  
    Codigo VARCHAR(50) PRIMARY KEY,  
    Id_Colegio INT NOT NULL,  
    Tipo VARCHAR(255) NOT NULL,  
    Caracteristicas VARCHAR(255),  
    Codigo_Prod VARCHAR(50) DEFAULT NULL,  
    FOREIGN KEY (Id_Colegio) REFERENCES COLEGIO (Id) ON DELETE  
CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (Codigo_Prod) REFERENCES PRODUCTO_TERMINADO  
(Codigo) ON DELETE CASCADE ON UPDATE CASCADE  
);
```

## **TABLA INVENTARIO**

```
CREATE TABLE INVENTARIO (  
    Codigo VARCHAR(50) PRIMARY KEY,  
    Cantidad_Existente INT,  
    Descripcion VARCHAR(255) NOT NULL,  
    FOREIGN KEY (Codigo) REFERENCES PRODUCTO_TERMINADO (Codigo) ON  
DELETE CASCADE ON UPDATE CASCADE  
);
```



## **TABLA PROVEEDOR**

```
CREATE TABLE PROVEEDOR (  
    Nit INT PRIMARY KEY,  
    Nombre VARCHAR(25),  
    Nom_contacto VARCHAR(25),  
    Telefono VARCHAR(20),  
    Direccion VARCHAR(25)  
);
```

## **TABLA MATERIA**

```
CREATE TABLE MATERIA (  
    Codigo INT PRIMARY KEY,  
    Tipo VARCHAR(50),  
    Descripcion VARCHAR(255),  
    Cantidad_Exist INT,  
    Unidad_Medida VARCHAR(50)  
);
```

## **TABLA VENTA**

```
CREATE TABLE VENTA (  
    Codigo_Fact VARCHAR(20) PRIMARY KEY,  
    Num_pedido INT NOT NULL,  
    Id_Cliente INT,  
    Monto_Final FLOAT,  
    FOREIGN KEY (Num_pedido) REFERENCES PEDIDO (Num_pedido) ON DELETE  
CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (Id_Cliente) REFERENCES CLIENTE (ID) ON DELETE  
CASCADE ON UPDATE CASCADE  
);
```

## **TABLA DESCUENTA**

```
CREATE TABLE DESCUENTA (  
    Codigo_Fact VARCHAR(20),  
    CODIGO VARCHAR(50),  
    FOREIGN KEY (Codigo_Fact) REFERENCES VENTA (Codigo_Fact) ON DELETE  
CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (CODIGO) REFERENCES INVENTARIO ON DELETE  
CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (Codigo_Fact, CODIGO)  
);
```

## **TABLA SUMINISTRA**

```
CREATE TABLE SUMINISTRA (  
    Nit_Proveedor INT,  
    Codigo_Materia INT,  
    FOREIGN KEY (Nit_Proveedor) REFERENCES PROVEEDOR (Nit) ON DELETE  
CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (Codigo_Materia) REFERENCES MATERIA (Codigo) ON  
DELETE CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (Nit_Proveedor, Codigo_Materia)  
);
```

## **TABLA UTILIZA**

```
CREATE TABLE UTILIZA (  
    Cod_prod VARCHAR(50),  
    Cod_Materia INT,  
    FOREIGN KEY (Cod_prod) REFERENCES PRODUCTO_TERMINADO (Codigo)  
ON DELETE CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (Cod_Materia) REFERENCES MATERIA (Codigo) ON DELETE  
CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY (Cod_prod, Cod_Materia)  
);
```

## CONSULTAS

### 1. Listado de Productos encargados pendientes por entregar (ordenados por fecha)

```
SELECT c.ID, p.fecha_encargo, p.Num_pedido, pt.Codigo, pt.Descripcion
FROM CLIENTE c
JOIN PEDIDO p ON c.ID = p.Id_Cliente
JOIN PRODUCTO_TERMINADO pt ON p.Num_pedido = pt.Num_pedido
WHERE p.Estado <> 'Entregado'
ORDER BY p.fecha_encargo ASC;
```

### 2. Por cada cliente, listar los productos encargados que no han sido entregados

```
SELECT c.ID, c.Nombre, c.telefono, p.Num_pedido, pt.Codigo, pt.Descripcion
FROM CLIENTE c
JOIN PEDIDO p ON c.ID = p.Id_Cliente
JOIN PRODUCTO_TERMINADO pt ON p.Num_pedido = pt.Num_pedido
WHERE p.Estado <> 'Entregado'
```

### 4. Dado un colegio las características de su uniforme

```
SELECT c.Nombre, u.Tipo, u.Caracteristicas
FROM COLEGIO c
JOIN UNIFORME u ON c.ID = u.ID_Colegio
WHERE c.Nombre = 'Colegio C';
```

### 5. Calcular el total de productos vendidos por colegio

```
SELECT c.nombre, SUM(pt.cantidad_existente ) AS Total_Productos_Vendidos
FROM Colegio c
JOIN Uniforme u ON c.id = u.Id_colegio
JOIN Pedido p ON u.codigo = p.articulo
JOIN producto_terminado pt ON p.num_pedido = pt.num_pedido
JOIN Venta v ON pt.num_pedido = v.num_pedido
GROUP BY c.nombre;
```

### 6. Total ventas

```
SELECT SUM(Monto_Final) AS Total_Ventas
FROM VENTA;
```

