



UNIVERSIDAD DE CÓRDOBA
"VIGILADA MINEDUCACIÓN"

ENTREGA PROTOTIPO PROYECTO FINAL

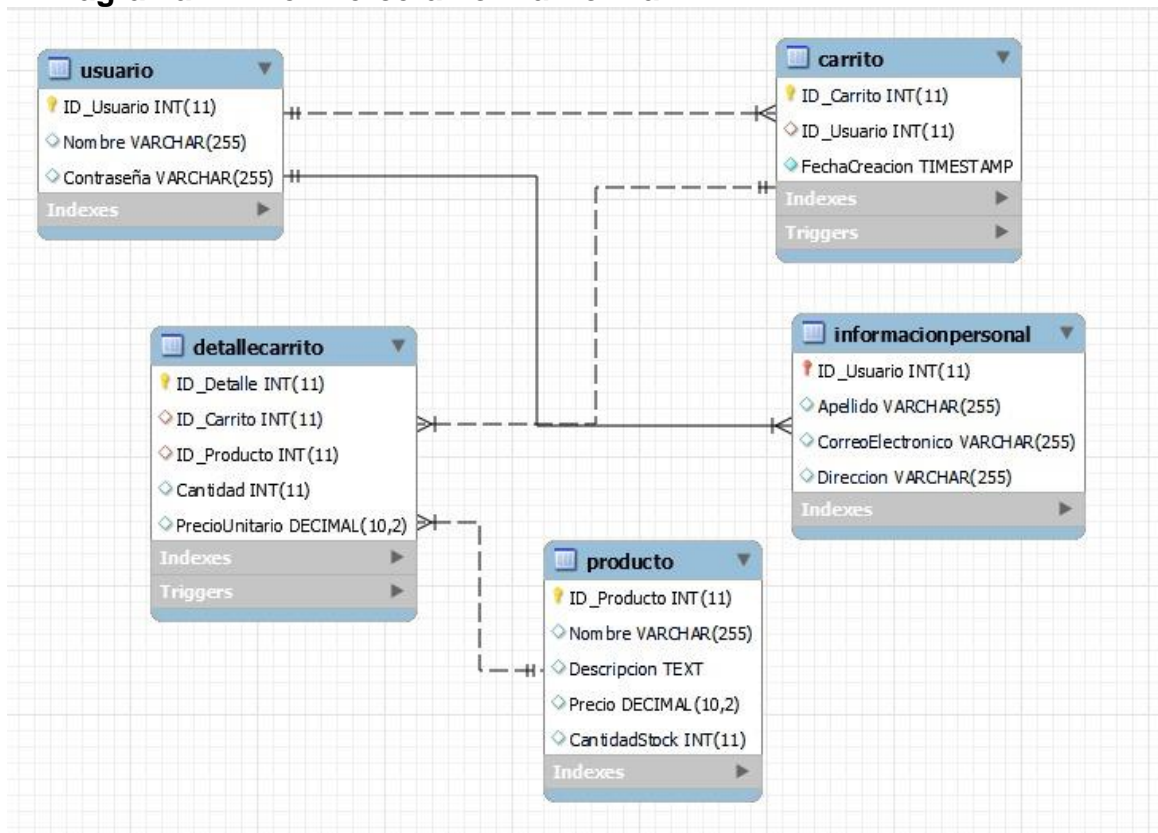
Jhonatan José Pérez rivera
Andrés Javier Orozco Begambre
Rodrigo Espitia Miranda

Rubén enrique Baena navarro

INGENIERÍA DE SISTEMAS, FACULTAD DE INGENIERÍAS

BASE DE DATOS II
MONTERÍA – CÓRDOBA

1. Diagrama MER en Tercera Forma Normal.



2. Creación de Bases de Datos.

```
CREATE TABLE usuario (  
  ID_Usuario INT PRIMARY KEY,  
  Nombre VARCHAR(255),  
  Contraseña VARCHAR(255)  
);
```

```
CREATE TABLE informacionpersonal (  
  ID_Usuario INT PRIMARY KEY,  
  Apellido VARCHAR(255),  
  CorreoElectronico VARCHAR(255) UNIQUE,  
  Direccion VARCHAR(255),  
  FOREIGN KEY (ID_Usuario) REFERENCES  
  usuario(ID_Usuario)  
);
```

```
CREATE TABLE Producto (
  ID_Producto INT PRIMARY KEY,
  Nombre VARCHAR(255),
  Descripcion TEXT,
  Precio DECIMAL(10, 2),
  CantidadStock INT
);
```

```
CREATE TABLE Carrito (
  ID_Carrito INT PRIMARY KEY,
  ID_Usuario INT,
  FechaCreacion TIMESTAMP DEFAULT
CURRENT_TIMESTAMP,
  FOREIGN KEY (ID_Usuario) REFERENCES
Usuario(ID_Usuario)
);
```

```
CREATE TABLE DetalleCarrito (
  ID_Detalle INT PRIMARY KEY,
  ID_Carrito INT,
  ID_Producto INT,
  Cantidad INT,
  PrecioUnitario DECIMAL(10, 2),
  FOREIGN KEY (ID_Carrito) REFERENCES Carrito(ID_Carrito),
  FOREIGN KEY (ID_Producto) REFERENCES
Producto(ID_Producto)
);
```

3. Estructura de Tablas con Datos.

Tabla carrito:

| | | | | |
|---|------------------------|------------------------|------------------------|-------------------------|
| ✓ Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0005 segundos.) | | | | |
| SELECT * FROM `carrito` | | | | |
| <input type="checkbox"/> Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar] | | | | |
| <input type="checkbox"/> Mostrar todo Número de filas: <input type="text" value="25"/> Filtrar filas: <input type="text" value="Buscar en esta tabla"/> | | | | |
| Opciones extra | | | | |
| <div> <div>← T →</div> <div>▼ ID_Carrito ID_Usuario FechaCreacion</div> </div> | | | | |
| <input type="checkbox"/> | Editar | Copiar | Borrar | 1 1 2023-11-14 04:13:11 |
| <input type="checkbox"/> | Editar | Copiar | Borrar | 2 2 2023-11-14 04:13:11 |

Tabla detalle carrito:













| | | | | | | |
|--|---|---|---|---|------------|--|
| ✓ 1 fila afectada. | | | | | | |
| UPDATE `detallecarrito` SET `PrecioUnitario` = '120000' WHERE `detallecarrito`.`ID_Detalle` = 4; | | | | | | |
| [Editar en línea] [Editar] [Crear código PHP] | | | | | | |
| ✓ Mostrando filas 0 - 3 (total de 4, La consulta tardó 0,0003 segundos.) | | | | | | |
| SELECT * FROM `detallecarrito` | | | | | | |
| ☐ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar] | | | | | | |
| ☐ Mostrar todo Número de filas: 25 ▾ Filtrar filas: <input type="text" value="Buscar en esta tabla"/> Ordenar según la cla | | | | | | |
| Opciones extra | | | | | | |
| ← T → ▾ ID_Detalle ID_Carrito ID_Producto Cantidad PrecioUnitario | | | | | | |
| ☐  Editar  Copiar  Borrar | 1 | 1 | 1 | 3 | 1500000.00 | |
| ☐  Editar  Copiar  Borrar | 2 | 1 | 3 | 2 | 35000.00 | |
| ☐  Editar  Copiar  Borrar | 3 | 2 | 2 | 5 | 25000.00 | |
| ☐  Editar  Copiar  Borrar | 4 | 2 | 4 | 1 | 120000.00 | |

Tabla producto

| | | | | | | |
|--|---|---------------|-------------------------|------------|----|--|
| ✓ Mostrando filas 0 - 3 (total de 4, La consulta tardó 0,0005 segundos.) | | | | | | |
| SELECT * FROM `producto` | | | | | | |
| ☐ Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar] | | | | | | |
| ☐ Mostrar todo Número de filas: 25 ▾ Filtrar filas: <input type="text" value="Buscar en esta tabla"/> Ordenar según la clave: Ningun | | | | | | |
| Opciones extra | | | | | | |
| ← T → ▾ ID_Producto Nombre Descripcion Precio CantidadStock | | | | | | |
| ☐  Editar  Copiar  Borrar | 1 | Motocierra | stihl 381 | 1500000.00 | 10 | |
| ☐  Editar  Copiar  Borrar | 2 | Fertilizante | fertilizante para mais | 25000.00 | 30 | |
| ☐  Editar  Copiar  Borrar | 3 | glifosato | matamalezas concentrado | 35000.00 | 10 | |
| ☐  Editar  Copiar  Borrar | 4 | electro bomba | electrobomba de 110V | 120000.00 | 15 | |

Tabla usuario

✓ Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0008 segundos.)

SELECT * FROM `usuario`

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

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

Opciones extra

| | ID_Usuario | Nombre | Contraseña |
|---|------------|--------|-------------|
| <input type="checkbox"/>  Editar  Copiar  Borrar | 1 | Juan | contraseña1 |
| <input type="checkbox"/>  Editar  Copiar  Borrar | 2 | Maria | contraseña2 |

Tabla informacionpersonal







✓ Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0015 segundos.)





SELECT * FROM `informacionpersonal`

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

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

Opciones extra

| | ID_Usuario | Apellido | CorreoElectronico | Direccion |
|---|------------|----------|-------------------|-----------------|
| <input type="checkbox"/>  Editar  Copiar  Borrar | 1 | Perez | juan@example.com | Calle A, Ciudad |
| <input type="checkbox"/>  Editar  Copiar  Borrar | 2 | Lopez | maria@example.com | Calle B, Ciudad |

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

4. Subconsultas:

4.1 Subconsulta para Obtener el Total de Compras de un Usuario:

```
SELECT ID_Usuario, (SELECT COUNT(*) FROM Carrito WHERE ID_Usuario = Usuario.ID_Usuario) AS TotalCompras FROM Usuario;
```

✓ Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0004 segundos.)

SELECT ID_Usuario, (SELECT COUNT(*) FROM Carrito WHERE ID_Usuario = Usuario.ID_Usuario) AS TotalCompras FROM Usuario;

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

Opciones extra

| ID_Usuario | TotalCompras |
|------------|--------------|
| 1 | 1 |
| 2 | 1 |

4.2 Subconsulta para Verificar la Existencia de un Producto en el Carrito:

```
SELECT ID_Carrito FROM DetalleCarrito WHERE ID_Producto = 1  
AND ID_Carrito = (SELECT ID_Carrito FROM Carrito WHERE  
ID_Usuario = 1);
```

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0006 segundos.)

```
SELECT ID_Carrito FROM DetalleCarrito WHERE ID_Producto = 1 AND ID_Carrito = (SELECT ID_Carrito FROM Carrito WHERE ID_Usuario = 1);
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

Opciones extra

| | ID_Carrito |
|---|------------|
| <input type="checkbox"/> Editar Copiar Borrar | 1 |

4.3 Subconsulta para Obtener el Producto Más Caro

```
SELECT *  
FROM Producto  
WHERE Precio = (SELECT MAX(Precio) FROM Producto);
```

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0007 segundos.)

```
SELECT * FROM Producto WHERE Precio = (SELECT MAX(Precio) FROM Producto);
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas:

Opciones extra

| | ID_Producto | Nombre | Descripcion | Precio | CantidadStock |
|---|-------------|------------|-------------|------------|---------------|
| <input type="checkbox"/> Editar Copiar Borrar | 1 | Motocierra | stihl 381 | 1500000.00 | 10 |

5. Procedimientos:

5.1 Procedimiento para Actualizar la Cantidad de Stock después de una Compra:

```
DELIMITER //  
CREATE PROCEDURE ActualizarStock(IN producto_id INT, IN  
cantidad_comprada INT)  
BEGIN  
UPDATE Producto SET CantidadStock = CantidadStock -  
cantidad_comprada WHERE ID_Producto = producto_id;  
END //  
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0024 segundos.)

```
CREATE PROCEDURE ActualizarStock(IN producto_id INT, IN cantidad_comprada INT) BEGIN UPDATE Producto SET CantidadStock = CantidadStock - cantidad_comprada WHERE ID_Producto = producto_id; END;
```

[[Editar en línea](#)] [[Editar](#)] [[Crear código PHP](#)]

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0003 segundos.)

```
SELECT CantidadStock FROM Producto WHERE ID_Producto = 1;
```

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

☐ Mostrar todo | Número de filas: Filtrar filas:

Opciones extra

← T → CantidadStock
☐  Editar  Copiar  Borrar 10

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0094 segundos.)

```
CALL ActualizarStock(1, 5);
```

[[Editar en línea](#)] [[Editar](#)] [[Crear código PHP](#)]

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0004 segundos.)

```
SELECT CantidadStock FROM Producto WHERE ID_Producto = 1;
```

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

☐ Mostrar todo | Número de filas: Filtrar filas:

Opciones extra

← T → CantidadStock
☐  Editar  Copiar  Borrar 5

5.2 Procedimiento para Actualizar la Cantidad de Stock después de una Compra:

```
DELIMITER //
CREATE PROCEDURE CalcularTotalCompra(IN carrito_id INT, OUT
total DECIMAL(10, 2))
BEGIN
SELECT SUM(Cantidad * PrecioUnitario) INTO total
FROM DetalleCarrito WHERE ID_Carrito = carrito_id;
END //
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0169 segundos.)

```
CREATE PROCEDURE CalcularTotalCompra(IN carrito_id INT, OUT total DECIMAL(10, 2)) BEGIN SELECT SUM(Cantidad * PrecioUnitario) INTO total FROM
DetalleCarrito WHERE ID_Carrito = carrito_id; END;
```

[Editar en línea] [Editar] [Crear código PHP]

Inicializamos la variable en cero y llamamos el procedimiento

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0003 segundos.)

```
SET @total_compra = 0;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0007 segundos.)

```
CALL CalcularTotalCompra(1, @total_compra);
```

[Editar en línea] [Editar] [Crear código PHP]

Seleccionamos la variable

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0002 segundos.)

```
SELECT @total_compra AS TotalCompra;
```

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

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

TotalCompra

4570000.00

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

5.3 Procedimiento para Actualizar la Cantidad de Stock después de una Compra:

```
DELIMITER //  
CREATE PROCEDURE EliminarProductoCarrito(IN detalle_id INT)  
BEGIN  
    DELETE FROM DetalleCarrito WHERE ID_Detalle = detalle_id;  
END //  
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0087 segundos.)










```
CREATE PROCEDURE EliminarProductoCarrito(IN detalle_id INT) BEGIN DELETE FROM DetalleCarrito WHERE ID_Detalle = detalle_id; END;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0016 segundos.)

```
CALL EliminarProductoCarrito(3);
```

[Editar en línea] [Editar] [Crear código PHP]

| | | | | ID_Detalle | ID_Carrito | ID_Producto | Cantidad | PrecioUnitario |
|--------------------------|--|--|--|------------|------------|-------------|----------|----------------|
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 1 | 1 | 1 | 3 | 1500000.00 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 2 | 1 | 3 | 2 | 35000.00 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 4 | 2 | 4 | 1 | 120000.00 |

6. Funciones:

6.1 Función para Obtener el Precio Total de un Producto en el Carrito

```
DELIMITER //  
CREATE FUNCTION CalcularPrecioTotal(Cantidad  
INT,PrecioUnitario DECIMAL(10, 2))  
RETURNS DECIMAL(10, 2)  
BEGIN  
    RETURN Cantidad * PrecioUnitario;  
END//  
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0033 segundos.)

```
CREATE FUNCTION CalcularPrecioTotal(Cantidad INT,PrecioUnitario DECIMAL(10, 2)) RETURNS DECIMAL(10, 2) BEGIN RETURN Cantidad * PrecioUnitario; END;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0006 segundos.)

```
SELECT CalcularPrecioTotal(5, 35000.00) AS PrecioTotal;
```

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

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

| PrecioTotal |
|-------------|
| 175000.00 |

6.2 Función para Validar la Existencia de un Usuario:

```
DELIMITER //
CREATE FUNCTION UsuarioExiste(ID_Usuario INT)
RETURNS BOOLEAN
BEGIN
    RETURN EXISTS (SELECT 1 FROM usuario WHERE ID_Usuario =
ID_Usuario);
END //
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0047 segundos.)

```
CREATE FUNCTION UsuarioExiste(ID_Usuario INT) RETURNS BOOLEAN BEGIN RETURN EXISTS (SELECT 1 FROM usuario WHERE ID_Usuario = ID_Usuario); END;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0011 segundos.)

```
SELECT UsuarioExiste(1) AS UsuarioExiste;
```

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

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

| UsuarioExiste |
|---------------|
| 1 |

6.3 Función para Contar Productos Stock Bajo:*

```
DELIMITER //
CREATE FUNCTION ContarProductosStockBajo(umbral_stock INT)
RETURNS INT
BEGIN
    DECLARE cantidad_stock_bajo INT;

    SELECT COUNT(*)
    INTO cantidad_stock_bajo
    FROM Producto
    WHERE CantidadStock < umbral_stock;

    RETURN cantidad_stock_bajo;
END //
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0111 segundos.)

```
CREATE FUNCTION ContarProductosStockBajo(umbral_stock INT) RETURNS INT BEGIN DECLARE cantidad_stock_bajo INT; SELECT COUNT(*) INTO cantidad_stock_bajo FROM Producto WHERE CantidadStock < umbral_stock; RETURN cantidad_stock_bajo; END;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0010 segundos.)

```
SELECT ContarProductosStockBajo(10) AS ProductosStockBajo;
```

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

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

Opciones extra

ProductosStockBajo

2

☐ Mostrar todo | Número de filas: 25 ▼ Filtrar filas:

7. Disparadores:

7.1 Disparador para Actualizar la Fecha de Creación de un Carrito:

```
DELIMITER //
CREATE TRIGGER ActualizarFechaCreacionCarrito
BEFORE INSERT ON Carrito
FOR EACH ROW
BEGIN
    SET NEW.FechaCreacion = NOW();
END;
//
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0075 segundos.)

```
CREATE TRIGGER ActualizarFechaCreacionCarrito BEFORE INSERT ON Carrito FOR EACH ROW BEGIN SET NEW.FechaCreacion = NOW(); END;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ 1 fila insertada. (La consulta tardó 0,0028 segundos.)

```
INSERT INTO carrito (ID_Carrito, ID_Usuario) VALUES (4,1);
```

[Editar en línea] [Editar] [Crear código PHP]

| | | | ID_Carrito | ID_Usuario | FechaCreacion |
|--------------------------|--|--|--|------------|-----------------------|
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 1 | 1 2023-11-14 04:13:11 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 2 | 2 2023-11-14 04:13:11 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 4 | 1 2023-11-13 23:06:20 |

7.2 Disparador para Actualizar el Stock después de una Compra:

DELIMITER //

CREATE TRIGGER ActualizarStockDespuesDeCompra

AFTER INSERT ON DetalleCarrito

FOR EACH ROW

BEGIN

UPDATE Producto SET CantidadStock = CantidadStock -

NEW.Cantidad WHERE ID_Producto = NEW.ID_Producto;

END //

DELIMITER ;

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0059 segundos.)

```
CREATE TRIGGER ActualizarStockDespuesDeCompra AFTER INSERT ON DetalleCarrito FOR EACH ROW BEGIN UPDATE Producto SET CantidadStock = CantidadStock - NEW.Cantidad WHERE ID_Producto = NEW.ID_Producto; END;
```

[Editar en línea] [Editar] [Crear código PHP]

✓ 1 fila insertada. (La consulta tardó 0,0026 segundos.)

```
INSERT INTO detallecarrito (ID_Detalle,ID_Carrito,ID_Producto, Cantidad, PrecioUnitario) VALUES (3, 2, 3, 2, 26500);
```

[Editar en línea] [Editar] [Crear código PHP]

✓ Mostrando filas 0 - 3 (total de 4, La consulta tardó 0,0004 segundos.)

```
SELECT * FROM `detallecarrito`
```

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

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

Opciones extra

| | | | ID_Detalle | ID_Carrito | ID_Producto | Cantidad | PrecioUnitario |
|--------------------------|--|--|--|------------|-------------|----------|----------------|
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 1 | 1 | 1 | 3 1500000.00 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 2 | 1 | 3 | 2 35000.00 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 3 | 2 | 3 | 2 26500.00 |
| <input type="checkbox"/> |  Editar |  Copiar |  Borrar | 4 | 2 | 4 | 1 120000.00 |

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

7.3 Disparador para Validar la Cantidad de Stock antes de una Compra:

```
DELIMITER //
CREATE TRIGGER ValidarStockAntesDeCompra
BEFORE INSERT ON DetalleCarrito
FOR EACH ROW
BEGIN
    IF NEW.Cantidad > (SELECT CantidadStock FROM Producto
WHERE ID_Producto = NEW.ID_Producto) THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'No hay
suficiente stock para este producto.';
    END IF;
END //
DELIMITER ;
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0115 segundos.)

```
CREATE TRIGGER ValidarStockAntesDeCompra BEFORE INSERT ON DetalleCarrito FOR EACH ROW BEGIN IF NEW.Cantidad > (SELECT CantidadStock FROM Producto
WHERE ID_Producto = NEW.ID_Producto) THEN SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'No hay suficiente stock para este producto.'; END IF; END;
```

[[Editar en línea](#)] [[Editar](#)] [[Crear código PHP](#)]

Error

consulta SQL: [Copiar](#)

```
INSERT INTO DetalleCarrito (ID_Detalle, ID_Carrito, ID_Producto, Cantidad, PrecioUnitario)
VALUES (5, 2, 1, 10, 1500000);
```

MySQL ha dicho: ⓘ

#1644 - No hay suficiente stock para este producto.

Prototipo en Figma

1.



A mobile app login screen prototype. At the top, there is a blue user icon placeholder. Below it, the word "USUARIO" is displayed in bold. The form contains five input fields: "ID_Usuario", "Nombre", "Apellido", "Correo electronico", and "Contraseña". Each field is represented by a light gray rounded rectangle with a thin blue border. At the bottom, there is a blue button with the text "Iniciar sesion" in white.

En esta interfaz el usuario se debe registra para dar inicio a la plataforma, al iniciar sesión nos dirige a la interfaz **(2)** llamada “productos”.

2.



En esta interfaz podemos encontrar la zona de productos que al presionar en el botón de “ agregar al carrito” nos dirige a una nueva interfaz llamada “ productos agregados al carrito”. Al igual que presionar en el icono de las barras nos dirige a una lista de productos deseados.

3.



Como le había comentado en la interfaz (2) al darle en el botón “agregar al carrito” nos dirigía al sitio “productos agregados al carrito”.

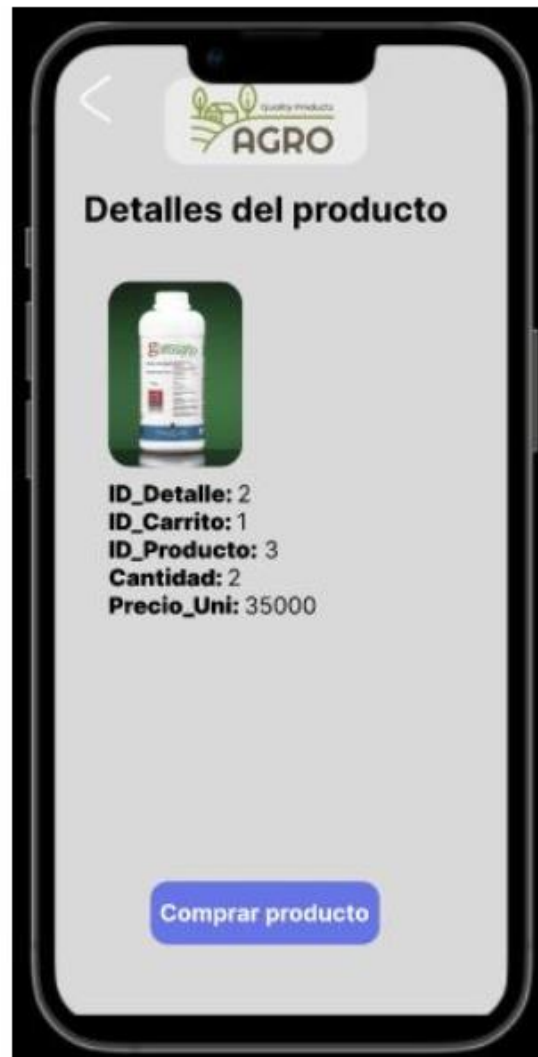
En esta interfaz al darle en el botón “ver detalles” esta petición se guarda en el icono del carrito que se encuentra en la parte superior derecha y al presionar este mismo, nos manda a la interfaz detalles del producto.

4.



Como lo hab a dicho en la interfaz (2) al darle en el espacio de barras este nos dirige a esta interfaz “ lista de deseo”

5.



Como dije en la interfaz **(3)**, al darle en el botón de “ver detalles” nos manda a la interfaz “detalles del producto”.

En esta interfaz al darle en el botón “comprar producto” nos manda a una zona llamada “lista de compras” que es donde estarán todos los productos comprados por ese usuario.

6.



Como lo dije en la interfaz (5), al darle en el botón “comprar producto” nos ubica en esta que se llama “lista de compras”.

Para finalizar la compra le damos en el botón “compra exitosa” y este nos retorna a la interfaz (2) que es la zona de los productos.