

# DISEÑO DE SISTEMAS

## TP3

MAGARIO ULISES

## **Requerimientos funcionales**

### **1. Registro de clientes:**

- El sistema debe permitir a los clientes registrarse e identificarse para cotizar y generar pedidos.

### **2. Consulta de productos:**

- Los clientes deben poder consultar los productos disponibles, junto con sus descripciones, precios por unidad y cantidades por pallet.
- Filtros de búsqueda:
  1. Nombre: Los clientes pueden buscar productos ingresando una palabra clave o el nombre completo del producto.
  2. Hasta precio: Los clientes pueden establecer un valor máximo, y el sistema debe mostrar los productos cuyo precio sea igual o menor a dicho valor.

### **3. Cotización de productos:**

- El sistema debe permitir que los clientes ingresen los metros cuadrados a construir y los tipos de materiales.
- Debe calcular automáticamente la cantidad de materiales necesarios en función de los datos proporcionados por el cliente.
- Debe calcular el importe total de la cotización, incluyendo el detalle de la cantidad de pallets necesarios y el costo total.

### **4. Descuentos por cantidad:**

- El sistema debe gestionar descuentos según la cantidad de productos solicitados.

### **5. Generación de pedidos:**

- El cliente debe poder generar un pedido basado en la cotización realizada.
- El sistema debe solicitar y almacenar la información del domicilio de envío.

### **6. Gestión de envíos:**

- El sistema debe registrar y coordinar el envío de los productos desde la planta C directamente al domicilio del cliente.

### **7. Gestión de pagos:**

- El sistema debe ofrecer distintas formas de pago (transferencia bancaria, tarjetas de crédito, etc.).
- El sistema debe permitir aprobar el pedido una vez que el pago ha sido realizado.

## **Requerimientos no funcionales**

### **1. Rendimiento:**

- El sistema debe permitir realizar las cotizaciones y generar pedidos en tiempo real, sin demoras significativas.

### **2. Escalabilidad:**

- El sistema debe ser capaz de soportar un creciente número de usuarios, especialmente en temporadas de alta demanda.

### **3. Disponibilidad:**

- El sistema debe estar disponible 24/7 para que los clientes puedan consultar productos y realizar pedidos en cualquier momento.

### **4. Seguridad:**

- El sistema debe asegurar la protección de los datos de los clientes y las transacciones de pago mediante mecanismos como cifrado de datos y autenticación segura.

### **5. Mantenibilidad:**

- El sistema debe ser fácilmente mantenible y actualizable para permitir la inclusión de nuevos productos, precios y promociones.

### **6. Usabilidad:**

- La interfaz del sistema debe ser sencilla e intuitiva para que cualquier cliente, sin conocimientos técnicos avanzados, pueda realizar cotizaciones y pedidos fácilmente.

### **7. Compatibilidad con diferentes dispositivos:**

- El sistema debe ser responsive, permitiendo el acceso desde diferentes dispositivos, como computadoras, tablets y teléfonos móviles.

### Iniciar sesión:

Campo	Front-end	Back-end	Base de Datos
usuario	string	string	varchar
contraseña	string	string	varchar

#### Front-end (JavaScript):

```
let credenciales = {  
  usuario: "umagario",      // string  
  password: "dsi.2024"      // string  
};
```

#### Back-end (Node.js):

```
const { usuario, password } = req.body; // string, string
```

#### Base de datos (MySQL):

```
CREATE TABLE clientes (  
  id_cliente INT AUTO_INCREMENT PRIMARY KEY, -- int  
  usuario VARCHAR(50),                        -- string  
  password VARCHAR(255)                      -- string  
);
```

Iniciar sesión

USUARIO

CONTRASEÑA

[¿Has olvidado tu contraseña?](#)


INICIAR SESIÓN

## *Productos disponibles:*


Campo	Front-end	Back-end	Base de Datos
material	string	string	varchar
precio_unitario	number	number	decimal
descripcion	string	string	text
cant_pallet	number	number	int

## PRODUCTOS DISPONIBLES

FILTRAR



**Ladrillo hueco 12x18x33 cm 9 tubos**  
**Precio por unidad:** \$390,00  
**Descripción:** Ladrillo hueco cerámico  
12x18x33 cm 9 tubos  
Ladrillo de cerramiento  
**Cantidad por pallet:** 144 unidades



**Viga 4 mts**  
**Precio por unidad:** \$10619  
**Descripción:** Ladrillo hueco cerámico  
12x18x33 cm 9 tubos  
Ladrillo de cerramiento

COTIZAR

### Front-end (JavaScript):

```
let producto = {  
  material: "Cemento",           // string  
  precio_unitario: 300,          // number  
  descripcion: "Cemento de alta calidad", // string  
  cantidad_pallet: 10            // number  
};
```

### Back-end (Node.js):

```
const { material, precio_unitario, descripcion, cantidad_pallet } = req.body; // string, number, string, number
```

### Base de datos (MySQL):

```
CREATE TABLE productos (  
  id_producto INT AUTO_INCREMENT PRIMARY KEY, -- int  
  material VARCHAR(100),                      -- string  
  precio_unitario DECIMAL(10, 2),             -- number (decimal)  
  descripcion TEXT,                           -- string  
  cantidad_pallet INT                         -- int  
);
```

## Filtrar:


Campo	Front-end	Back-end	Base de Datos
nombre	string	string	varchar
hasta_precio	number	number	decimal

**FILTRAR**

ATRÁS

NOMBRE

HASTA PRECIO



**Ladrillo hueco 12x18x33 cm 9 tubos**  
**Precio por unidad:** \$390,00  
**Descripción:** Ladrillo hueco cerámico  
12x18x33 cm 9 tubos  
Ladrillo de cerramiento  
**Cantidad por pallet:** 144 unidades

COTIZAR



**Viga 4 mts**  
**Precio por unidad:** \$10619  
**Descripción:** Ladrillo hueco cerámico  
12x18x33 cm 9 tubos  
Ladrillo de cerramiento

COTIZAR

### Front-end (JavaScript):

```
let filtros = {  
  nombre: "Cemento",      // string  
  hastaPrecio: 500        // number  
};
```

### Back-end (Node.js):

```
const { nombre, hastaPrecio } = req.query; // string, number
```

### Base de datos (MySQL):

```
SELECT * FROM productos  
WHERE material LIKE '%Cemento%'  
AND precio_unitario <= 500;
```

Cotizar:

Campo	Front-end	Back-end	Base de Datos
nombre	string	string	varchar
largo	number	number	decimal
ancho	number	number	decimal
alto	number	number	decimal
cant_pallets	number	number	int
detalle	string	string	text
descuento	number	number	decimal
importe	number	number	decimal

COTIZACIÓN

NOMBRE: xxxxxxxxxxxxxxxxxxxx

LARGO: xxxx

ANCHO: xxxx

ALTO: xxxx

CALCULAR: xxxxxxxx

DESCUENTO: \$\$\$\$\$\$\$\$\$\$

IMPORTE: \$\$\$\$\$\$\$\$\$\$

Detalle: xxxxxxxxxxxx

COMPRAR

Front-end (JavaScript):

```
let cotizacion = {
  nombre: "Cemento",           // string
  largo: 10,                   // number
  ancho: 5,                    // number
  alto: 3,                     // number
  cantidad_pallets: 10,        // number
  detalle: "Cotización de cemento para proyecto", // string
  descuento: 10.00,            // number (10% de descuento)
  importe: 2000.00             // number (importe total calculado)
};
```

Back-end (Node.js):

```
const { nombre, largo, ancho, alto, cantidad_pallets, detalle, descuento, importe } = req.body;
// string, number, number, number, number, string, number, number
```

## Base de datos (MySQL):

```
CREATE TABLE cotizaciones (  
  id_cotizacion INT AUTO_INCREMENT PRIMARY KEY, -- int  
  nombre VARCHAR(100), -- string  
  largo DECIMAL(10, 2), -- number  
  ancho DECIMAL(10, 2), -- number  
  alto DECIMAL(10, 2), -- number  
  cantidad_pallets INT, -- int  
  detalle TEXT, -- string  
  descuento DECIMAL(5, 2), -- number (porcentaje de descuento)  
  importe DECIMAL(10, 2) -- number (importe total)  
);
```