

TALLER DE BASE DE DATOS DIURNO 2-2024

Proyecto semestral

Ayudantes: Nicolás Rojas /Pablo Macuada

Profesores: Matías Calderón

Enunciado Laboratorios

Entrega: 11 de noviembre de 2024

Problema

El objetivo es diseñar y desarrollar un sistema de base de datos relacional para un E-commerce, donde se gestionen productos, órdenes de compra, clientes y el inventario en tiempo real. Debes garantizar que cuando un cliente realice una compra, el stock de productos se actualice automáticamente, y si el stock de un producto llega a 0, este se marque como “agotado”.

Requerimientos del sistema:

1. Gestión de Productos:

- Cada producto tiene un nombre, descripción, precio, stock disponible y categoría.
- Se debe llevar un control del stock de los productos. Si el stock de un producto llega a 0, el producto debe marcarse como “agotado”.

2. Gestión de Clientes:

- Cada cliente tiene un nombre, dirección, correo electrónico y teléfono.
- Se requiere registrar la información personal de los clientes y su historial de compras.

3. Gestión de Órdenes:

- Un cliente puede realizar una o más órdenes de compra. Cada orden contiene una fecha y el estado de la orden (pendiente, pagada, enviada).
- Cada orden puede tener uno o más productos. Se debe registrar la cantidad de cada producto en la orden y el precio total.

4. Actualización automática del inventario:

- Al registrar una nueva orden, el sistema debe disminuir automáticamente el stock de los productos involucrados.
 - Si la cantidad de un producto llega a 0, debe cambiar su estado a “agotado” automáticamente.
-

Modelo Relacional:

1. Tabla producto:

Gestiona los productos que se venden en el sistema.

Campo	Tipo de dato	Descripción
id_producto	SERIAL (PK)	Identificador único del producto
nombre	VARCHAR(255)	Nombre del producto
descripcion	TEXT	Descripción del producto
precio	DECIMAL(10, 2)	Precio del producto
stock	INT	Cantidad disponible del producto
estado	VARCHAR(50)	Estado del producto ("disponible", "agotado")
id_categoria	INTEGER (FK)	Clave foránea que referencia a la categoría

2. Tabla categoría:

Contiene las categorías de los productos.

Campo	Tipo de dato	Descripción
id_categoria	SERIAL (PK)	Identificador único de la categoría
nombre	VARCHAR(100)	Nombre de la categoría

3. Tabla cliente:

Almacena los datos de los clientes.

Campo	Tipo de dato	Descripción
id_cliente	SERIAL (PK)	Identificador único del cliente
nombre	VARCHAR(255)	Nombre completo del cliente
direccion	VARCHAR(255)	Dirección del cliente
email	VARCHAR(100)	Correo electrónico del cliente
telefono	VARCHAR(20)	Teléfono de contacto

4. Tabla orden:

Contiene las órdenes de compra realizadas por los clientes.

Campo	Tipo de dato	Descripción
id_orden	SERIAL (PK)	Identificador único de la orden
fecha_orden	TIMESTAMP	Fecha en que se realizó la orden
estado	VARCHAR(50)	Estado de la orden ("pendiente", "pagada", "enviada")
id_cliente	INTEGER (FK)	Clave foránea que referencia al cliente
total	DECIMAL(10, 2)	Total a pagar por la orden

5. Tabla detalle_orden:

Contiene los detalles de cada producto en las órdenes de compra.

Campo	Tipo de dato	Descripción
id_detalle	SERIAL (PK)	Identificador único del detalle de la orden
id_orden	INTEGER (FK)	Clave foránea que referencia a la orden
id_producto	INTEGER (FK)	Clave foránea que referencia al producto
cantidad	INT	Cantidad del producto en la orden

Campo	Tipo de dato	Descripción
precio_unitario	DECIMAL(10, 2)	Precio unitario del producto

Relaciones:

- Un **cliente** puede realizar múltiples **órdenes** (relación 1 entre cliente y orden).
- Cada **orden** puede contener múltiples **productos** a través de la tabla detalle_orden (relación 1 entre orden y detalle_orden).
- Un **producto** puede aparecer en múltiples **detalles de órdenes** (relación 1 entre producto y detalle_orden).
- Cada **producto** pertenece a una **categoría** (relación 1 entre producto y categoría).

Tecnologías

La aplicación a desarrollar utilizará las siguientes tecnologías:

- Base de datos PostgreSQL + PostGIS.
- Servicio REST desarrollado con Java Spring, debe conectarse a la BD utilizando la librería Sql2o.
- Frontend desarrollado con Vue, versión 3.x
- Demostraciones de servicios REST pueden realizarse con aplicación Postman o similar.

Formato de Presentación

Los grupos deben realizar una presentación de no más de 15 minutos donde explican la solución realizada.

Las diapositivas de la presentación deben contener:

- Introducción: presentación del proyecto, roles de equipo.
- Descripción de avance del proyecto: Funcionalidades solicitadas con porcentaje de avance.
- Diagrama de arquitectura de software: Diagrama de alto nivel mostrando las capas, componentes principales y la interconexión entre las capas.
- Fragmentos de código: La presentación debe incluir fragmentos de código que muestran el flujo de datos desde que se obtiene una petición REST hasta que se realiza la consulta en la base de datos por SQL. Estos fragmentos se deben incluir para las funcionalidades solicitadas y para una llamada representativa del CRUD.
- Demostración de funcionalidades solicitadas: Se debe mostrar el comportamiento de la funcionalidad y presentar brevemente el código que la realiza, en funciones que tienen solamente backend, se pueden presentar las llamadas mediante Postman u otra aplicación similar.

- Contribuciones: Se debe mostrar las contribuciones de integrantes del grupo al código fuente mediante Github.
- Conclusiones: Similar a un Sprint Review, deben explicar desafíos y cómo los resolvieron, además, se debe analizar el modelo utilizado, indicando posibles cambios según los requerimientos solicitados.

Entregables

Cada grupo debe subir a la carpeta designada:

- El archivo con diapositivas la presentación (formato PDF).
- Enlace al repositorio con número del grupo en el nombre.

Evaluación

Las funcionalidades individuales las pueden encontrar en la [pauta detallada](#).

	ID	ítem	Pts.
Diapositivas	1	Introducción	2
	2	Descripción avance de proyecto	2
	3	Diagrama de arquitectura	5
	4	Demostración de funcionalidades solicitadas	10
	5	Fragmentos de código	5
	6	Contribuciones	2
	7	Conclusiones	5
Entregables	7	Presentación de máximo 15 minutos	5
	8	Enlace a repositorios	5
	9	Archivo con diapositivas de la presentación	5
Repositorio código fuente	10	Script de creación de la BD	4
	11	Dump con datos de prueba	5
	12	Archivo README.md con instrucciones y versiones de instalación	5
Aplicación	13	Creación de sistema consistente en servicio REST, aplicaciones web y BD PostgreSQL	20
	14	Creación de CRUD para todas las tablas del modelo de BD	20
Funcionalidades	15	Total de funcionalidades individuales	100
		Total	200