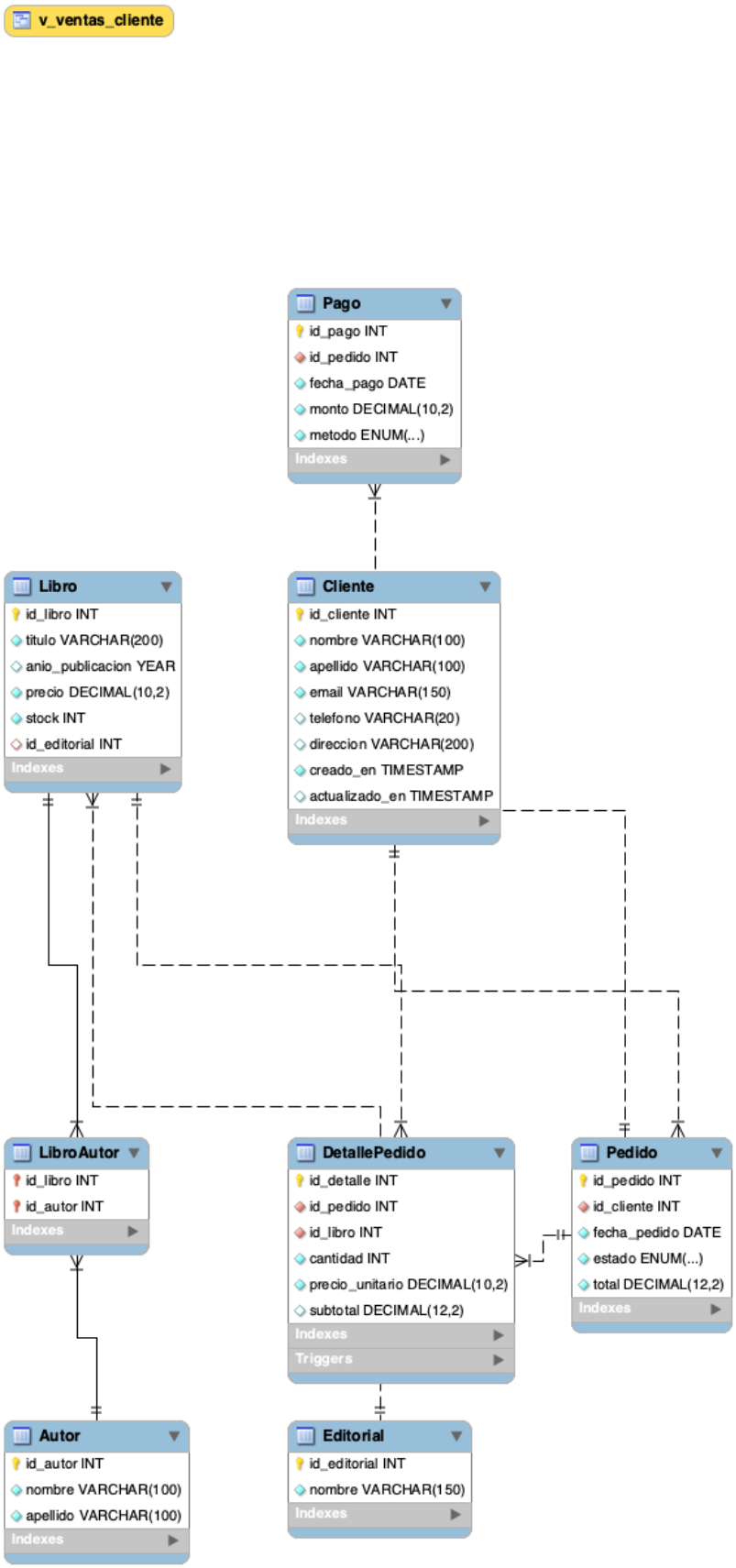
Entrega 2 – Proyecto Final

Alumna: **Noelia Rossi**

1. Resumen de la Entrega 1

En la primera entrega se desarrolló una base de datos para una **librería online**, con tablas para clientes, autores, editoriales, libros, pedidos, detalles de pedidos y pagos. El esquema se diseñó en 3FN y se incluyeron triggers para mantener actualizado el total de cada pedido. Archivo base: **libreriaonlineRossi_Ent1_revisada.sql**.

2. Diagrama Entidad–Relación (E-R)


v_ventas_cliente
Pago
id_pago INT
id_pedido INT
fecha_pago DATE
monto DECIMAL(10,2)
metodo ENUM(...)
Indexes
Libro
id_libro INT
titulo VARCHAR(200)
anio_publicacion YEAR
precio DECIMAL(10,2)
stock INT
id_editorial INT
Indexes
Cliente
id_cliente INT
nombre VARCHAR(100)
apellido VARCHAR(100)
email VARCHAR(150)
telefono VARCHAR(20)
direccion VARCHAR(200)
creado_en TIMESTAMP
actualizado_en TIMESTAMP
Indexes
LibroAutor
id_libro INT
id_autor INT
Indexes
DetallePedido
id_detalle INT
id_pedido INT
id_libro INT
cantidad INT
precio_unitario DECIMAL(10,2)
subtotal DECIMAL(12,2)
Indexes
Triggers
Pedido
id_pedido INT
id_cliente INT
fecha_pedido DATE
estado ENUM(...)
total DECIMAL(12,2)
Indexes
Autor
id_autor INT
nombre VARCHAR(100)
apellido VARCHAR(100)
Indexes
Editorial
id_editorial INT
nombre VARCHAR(150)
Indexes

3. Vistas

Nombre	Objetivo	Tablas
v_ventas_cliente	Resumir cantidad de pedidos y total comprado por cliente.	Cliente, Pedido
v_libros_stock	Listar libros con stock y editorial.	Libro, Editorial
v_pedidos_detallados	Detalle de pedidos con líneas, subtotales y datos de cliente/libro.	Pedido, DetallePedido, Libro, Cliente

4. Funciones (UDF)

Nombre	Objetivo	Tablas
fn_total_pedidos_cliente(id_cliente)	Devuelve el total gastado por un cliente.	Pedido
fn_stock_libro(id_libro)	Devuelve el stock actual de un libro.	Libro
fn_total_pagado_pedido(id_pedido)	Suma los pagos registrados de un pedido.	Pago
fn_saldo_pedido(id_pedido)	Calcula saldo = total del pedido – total pagado.	Pedido, Pago

5. Stored Procedures

Nombre	Objetivo	Tablas
sp_crear_pedido	Crea pedido y agrega una línea de detalle.	Pedido, DetallePedido
sp_agregar_detalle	Agrega una línea a un pedido existente.	DetallePedido
sp_registrar_pago	Registra un pago asociado a un pedido.	Pago
sp_actualizar_estado_pedido	Actualiza el estado de un pedido con validación.	Pedido

6. Triggers

Se definieron tres triggers sobre la tabla DetallePedido: **trg_detallepedido_ai**, **trg_detallepedido_au** y **trg_detallepedido_ad**. Su objetivo es recalcular automáticamente el total del pedido después de operaciones INSERT, UPDATE o DELETE en los detalles.

7. Scripts incluidos

libreriaonlineRossi_Ent1_revisada.sql: esquema base y datos iniciales.
Entrega2_objetos.sql: creación de vistas, funciones, procedimientos y triggers.
Entrega2_inserts.sql: inserción de datos de prueba con sentencias INSERT.
Entrega2_import.sql + archivos CSV: alternativa de carga de datos masiva mediante LOAD DATA.
Entrega2_tests.sql: script opcional de validación de objetos y consultas de prueba.

8. Cómo ejecutar paso a paso

1) Ejecutar **libreriaonlineRossi_Ent1_revisada.sql** (estructura y seed inicial).
2) Ejecutar **Entrega2_objetos.sql** (crea vistas, funciones, SP y triggers).
3) Cargar datos:
  • Opción A: ejecutar **Entrega2_inserts.sql**.
  • Opción B: importar CSV con **Entrega2_import.sql** (requiere habilitar LOCAL INFILE).
4) (Opcional) Ejecutar **Entrega2_tests.sql** para validar funcionamiento.

9. Conclusión

La segunda entrega amplía lo desarrollado en la primera, incorporando vistas, funciones, procedimientos y triggers, además de mecanismos para cargar datos mediante inserts o importación. Se cumplen los requisitos funcionales y técnicos de la consigna.