

Contenido

1.	Práctica de Examen	2
	Tema: Gestión de citas médicas.....	2
	Equipos: 3 personas	2
2.	Práctica de Examen	3
	Tema: Gestión de Biblioteca.....	3
	Equipos: 3 personas	3
3.	Práctica de Examen	4
	Tema: Gestión de inscripciones a cursos	4
	Equipos: 3 personas	4
4.	Práctica de Examen	5
	Tema: Gestión de alquiler de vehículos.....	5
	Equipos: 3 personas	5
5.	Práctica de Examen	6
	Tema: Gestión de una Agencia de Viajes.....	6
	Equipos: 3 personas	6
6.	Práctica de Examen	7
	Tema: Gestión de Reservas de Hotel	7
	Equipos: 4 personas	7
7.	Práctica de Examen	8
	Tema: Gestión de una tienda de productos electrónicos.....	8
	Equipos: 4 personas	8
8.	Práctica de Examen	9
	Tema: Gestión de Reservas en un Restaurante	9
	Equipos: 4 personas	9
9.	Práctica de Examen	10
	Tema: Gestión de una Clínica Dental	10
	Equipos: 4 personas	10
10.	Práctica de Examen.....	11
	Tema: Gestión de ventas en farmacia	11
	Equipos: 4 personas	11

1. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de citas médicas

Equipos: 3 personas

Enunciado

Una clínica privada requiere una aplicación para la gestión de sus **pacientes, médicos y citas médicas**, asegurando un control eficiente de los registros y facilitando la administración de los servicios médicos.

La aplicación debe permitir la gestión de **pacientes** mediante un formulario donde se puedan registrar, editar y eliminar datos como **ID autogenerado, foto (con ajuste automático de tamaño y forma si es necesario), nombre completo, fecha de nacimiento, teléfono, correo electrónico y dirección**. Además, debe contar con un módulo para la gestión de **médicos**, permitiendo registrar, modificar y eliminar médicos con información como **ID autogenerado, foto (con ajuste automático de tamaño y forma si es necesario), nombre completo, especialidad, teléfono y horario de atención**.

El sistema debe incluir un módulo de **citas médicas**, permitiendo registrar citas con los siguientes datos: **ID autogenerado, selección del paciente y del médico desde listas predefinidas, fecha y hora de la cita, motivo de consulta y estado de la cita (Pendiente, Atendida, Cancelada)**. Debe validarse que un paciente no pueda agendar más de una cita con el mismo médico en el mismo día y que los horarios seleccionados coincidan con la disponibilidad del médico.

Para la gestión eficiente de los datos, se deben desarrollar **procedimientos almacenados** que permitan **insertar, buscar, actualizar y eliminar pacientes, médicos y citas médicas**, así como obtener **la lista de citas programadas para un médico en un día específico y calcular la cantidad de citas atendidas en un período determinado**.

La aplicación debe **mejorar la experiencia del usuario** mediante validaciones para evitar el ingreso de datos incompletos, mensajes de confirmación antes de eliminar registros y notificaciones cuando se intente agendar una cita en un horario no disponible. Debe implementarse una interfaz **MDI (Multiple Document Interface)** con un **menú** para **acceder a la gestión de pacientes, médicos, citas y reportes**.

Finalmente, se debe generar un reporte con **Crystal Reports** que muestre todas las citas médicas programadas en un rango de fechas seleccionado, incluyendo **el nombre del paciente, el médico asignado, la especialidad, la fecha y hora de la cita, y el estado de la cita**. Además, el reporte debe incluir un **resumen del total de citas atendidas en ese período**, permitiendo un mejor análisis del rendimiento de la clínica.

2. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de Biblioteca

Equipos: 3 personas

Enunciado

Una biblioteca municipal necesita una aplicación para la gestión de préstamos y devoluciones de libros, asegurando un control eficiente del inventario y la disponibilidad de ejemplares.

La aplicación debe contar con un módulo para la gestión de **libros**, permitiendo registrar, modificar y eliminar libros con los siguientes datos: **ID autogenerado, foto de portada** (se debe adaptar tamaño de ser necesario), **título, autor, editorial, año de publicación, género y cantidad de ejemplares disponibles**.

También debe incluir un módulo de gestión de **usuarios**, permitiendo registrar, modificar y eliminar usuarios con información como **ID autogenerado, foto** (se debe adaptar tamaño de ser necesario), **nombre completo, número de identificación, dirección, teléfono y correo electrónico**.

El sistema debe permitir la gestión de **préstamos y devoluciones** de libros. Al registrar un préstamo, se debe seleccionar un usuario y un libro disponible desde listas predefinidas, estableciendo la **fecha de préstamo, la fecha de devolución programada y el estado del préstamo (Activo, Devuelto, Vencido)**. Se debe validar que un usuario **no pueda tomar prestados más de tres libros al mismo tiempo** y que solo se pueda prestar un libro si hay ejemplares disponibles.

Para mejorar la administración, se deben desarrollar **procedimientos almacenados** que permitan insertar, buscar, actualizar y eliminar registros de libros, usuarios y préstamos. También se deben crear procedimientos para **consultar el historial de préstamos de un usuario, obtener la lista de libros actualmente en préstamo y calcular la cantidad de préstamos realizados en un período determinado**.

La aplicación debe contar con una interfaz **MDI** (Multiple Document Interface) con un **menú** para **acceder** a la gestión de **libros, usuarios, préstamos y reportes**. Además, debe incluir mensajes de confirmación antes de eliminar registros, validaciones para evitar datos vacíos y notificaciones cuando un usuario intente tomar prestado más libros de los permitidos.

Finalmente, se debe generar un **reporte con Crystal Reports** que muestre **todos los préstamos realizados en un rango de fechas seleccionado**, incluyendo el nombre del usuario, el título del libro, la fecha de préstamo, la fecha de devolución programada y el estado del préstamo. Además, el reporte debe incluir un resumen de **cuántos libros han sido prestados y devueltos en ese período**, permitiendo a la biblioteca optimizar su gestión.

3. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de inscripciones a cursos

Equipos: 3 personas

Enunciado:

Una academia de cursos en línea requiere una aplicación para la **gestión eficiente** de sus **estudiantes, cursos e inscripciones**, asegurando un control adecuado de los registros y optimizando la administración académica.

La aplicación debe permitir la gestión de **estudiantes** mediante un formulario donde se puedan registrar, editar y eliminar datos como **ID autogenerado, foto de perfil con ajuste de tamaño, nombre completo, correo electrónico, teléfono y fecha de inscripción**.

Además, debe incluir un módulo para la administración de **cursos**, permitiendo registrar, modificar y eliminar cursos con información como **ID autogenerado, nombre del curso, descripción, duración en horas y costo**.

También es necesario implementar un sistema de **inscripciones** que permita registrar a los estudiantes en cursos, asegurando que cada inscripción contenga un **ID autogenerado, la selección de un estudiante y un curso desde listas predefinidas, la fecha de inscripción y el estado de la inscripción (Activa, Cancelada, Finalizada)**, validando que un estudiante no pueda inscribirse más de una vez en el mismo curso.

Para la gestión eficiente de los datos, se deben desarrollar **procedimientos almacenados** en que **permitan insertar, buscar, actualizar y eliminar estudiantes, cursos e inscripciones**, obtener la lista de estudiantes inscritos en un curso específico, obtener los cursos en los que está inscrito un **estudiante** y **calcular la cantidad de inscripciones en un período determinado**.

La aplicación debe **mejorar la experiencia del usuario** mediante mensajes de confirmación antes de eliminar registros, validaciones para evitar campos vacíos y notificaciones cuando un estudiante intente inscribirse en un curso en el que ya está registrado. Además, debe contar con una interfaz **MDI (Multiple Document Interface)** que **facilite la navegación** a través de un **menú** para **acceder** a la gestión de **estudiantes, cursos, inscripciones y reportes**.

Finalmente, se **debe generar un reporte** con **Crystal Reports** que muestre todas las **inscripciones realizadas en un período determinado, detallando el nombre del estudiante, el curso inscrito, la fecha de inscripción, el estado de la inscripción y el total de ingresos generados por inscripciones en ese período**, proporcionando así una herramienta útil para la supervisión y toma de decisiones dentro de la academia.

4. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de alquiler de vehículos

Equipos: 3 personas

Enunciado

Una empresa de alquiler de vehículos necesita una aplicación de escritorio para la gestión integral de **vehículos, clientes y alquileres**, que permita optimizar la administración de su flota de autos y proporcionar un seguimiento detallado de las transacciones.

La aplicación debe contar con un módulo de **gestión de vehículos**, que permita registrar, editar y eliminar vehículos de la base de datos, asegurando que cada vehículo tenga datos completos, como **ID autogenerado, foto (con ajuste automático de tamaño y forma si es necesario), marca, modelo, año, tipo de vehículo (Sedán, SUV, Camioneta, etc.), matrícula, precio por día de alquiler, y estado (Disponible, Alquilado, Mantenimiento)**. Además, se debe garantizar que un vehículo solo pueda ser alquilado si su estado es "Disponible".

La **gestión de clientes** debe permitir registrar, editar y eliminar clientes con datos como **ID autogenerado, nombre completo, foto (ajustada de forma automática si es necesario), número de identificación (como cédula o pasaporte), teléfono, y correo electrónico**.

En cuanto a los **alquileres**, la aplicación debe permitir registrar un alquiler asociando un cliente y un vehículo disponibles, estableciendo la **fecha de inicio y fin del alquiler, costo total del alquiler calculado según el precio por día del vehículo y la duración del alquiler, y el estado del alquiler (Activo, Finalizado, Cancelado)**. La aplicación debe evitar que un vehículo sea alquilado si no está disponible, asegurando el control total del inventario de vehículos.

Los **procedimientos almacenados** deben estar implementados para permitir la **gestión de vehículos, clientes y alquileres**, incluyendo la capacidad de insertar, actualizar, eliminar y obtener registros de vehículos, clientes, alquileres, así como el historial de alquileres de un cliente y los ingresos generados por alquileres en un periodo específico.

La aplicación debe incluir **validaciones** para **mejorar la experiencia del usuario**, tales como mostrar mensajes de confirmación antes de eliminar vehículos, clientes o alquileres, validar que los campos requeridos no estén vacíos antes de registrar datos, y notificar al usuario cuando intente alquilar un vehículo que no esté disponible. Además, debe incluir una ventana principal **MDI (Multiple Document Interface)** con un menú de acceso rápido a la **gestión de vehículos, clientes, alquileres y reportes**.

Finalmente, se debe generar un **reporte con Crystal Reports** que detalle todos los alquileres realizados en un **rango de fechas seleccionado**, mostrando la **ID y nombre del cliente, ID y nombre del vehículo, fecha de inicio y finalización del alquiler, costo total del alquiler, y el total de ingresos generados en ese periodo**, lo que permitirá a la empresa obtener un análisis claro de su actividad y optimizar su gestión operativa.

5. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de una Agencia de Viajes

Equipos: 3 personas

Enunciado

Una agencia de viajes requiere una aplicación para gestionar las **reservas de tours, clientes, y pagos**, permitiendo una administración eficiente de los viajes, las actividades y los clientes que los realizan. La aplicación debe permitir el control de disponibilidad de tours, gestión de itinerarios y seguimiento de pagos. La aplicación debe incluir un módulo de gestión de **tours**, donde se pueda registrar, editar y eliminar tours. Cada **tour** debe tener un ID autogenerado, nombre, descripción, fecha de inicio, fecha de fin, destino, precio por persona y disponibilidad (número de plazas disponibles).

Además, debe incluir un módulo de gestión de **clientes**, permitiendo registrar, editar y eliminar datos del cliente, como ID autogenerado, foto con ajuste automático de tamaño y forma si es necesario, nombre completo, teléfono, correo electrónico, dirección, y tipo de documento (DNI, pasaporte, etc.).

El sistema debe permitir la creación de **reservas** para los tours, donde se registre el cliente, el tour reservado, la cantidad de personas, el precio total (calculado a partir del precio por persona y la cantidad de personas), y la fecha de la reserva. La reserva debe tener un estado (confirmada, cancelada, finalizada), y debe actualizar la disponibilidad del tour según el número de personas reservadas.

Los **procedimientos almacenados** deben permitir insertar, buscar, actualizar y eliminar registros de tours, clientes y reservas. Además, deben incluir consultas que permitan obtener los tours más solicitados, los clientes que han reservado más tours, el historial de reservas de cada cliente, y los ingresos generados por las reservas en un período determinado.

Se deben implementar validaciones para asegurarse de que no se realicen reservas en tours sin disponibilidad y que el cliente tenga todos los datos necesarios antes de realizar una reserva.

La aplicación debe tener una interfaz **MDI** (Multiple Document Interface) con un **menú** que permita **acceder** a las diferentes funcionalidades (**gestión de tours, clientes, reservas y reportes**).

Finalmente, se debe generar un reporte con **Crystal Reports** que muestre todas las reservas realizadas en un período determinado, detallando el nombre del cliente, el tour reservado, la fecha de la reserva, el número de personas, el precio total, y el estado de la reserva. Este reporte debe ser útil para la gestión de la agencia de viajes y el seguimiento de los ingresos generados.

6. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de Reservas de Hotel

Equipos: 4 personas

Enunciado:

Un hotel de tamaño medio necesita una aplicación para gestionar las reservas de **habitaciones, clientes y servicios adicionales**. La aplicación debe permitir un control eficiente de las reservas, los pagos y la disponibilidad de las habitaciones, garantizando que la experiencia de los clientes sea óptima.

La aplicación debe contar con un módulo de gestión de **habitaciones**, donde se podrá registrar, editar y eliminar habitaciones. Cada **habitación** debe tener un ID autogenerado, tipo (individual, doble, suite), foto (con ajuste automático de tamaño y forma si es necesario), capacidad, precio por noche y estado (disponible, ocupada, en mantenimiento).

Además, debe permitir gestionar los **clientes**, registrando su información personal como ID autogenerado, foto (con ajuste automático de tamaño y forma si es necesario), nombre completo, teléfono, correo electrónico, dirección y fecha de nacimiento. También debe incluir un módulo para gestionar las **reservas**, permitiendo crear nuevas reservas, modificarlas o cancelarlas. Cada **reserva** debe tener un ID autogenerado, cliente, habitación asignada, fecha de llegada, fecha de salida, número de noches, precio total (calculado con el precio por noche y la cantidad de noches), y el estado de la reserva (confirmada, cancelada, finalizada).

El sistema debe permitir agregar **servicios adicionales**, como desayuno, transporte al aeropuerto, y masajes, los cuales serán opcionales en cada reserva. Estos servicios deberán ser seleccionados por el cliente al hacer la reserva y su costo deberá sumarse al precio total de la misma.

Los **procedimientos almacenados** deben incluir la lógica necesaria para insertar, buscar, actualizar y eliminar habitaciones, clientes, reservas y servicios adicionales. Además, debe permitir consultar las reservas por fecha, los clientes que han realizado reservas y el historial de habitaciones ocupadas.

La aplicación debe incluir validaciones para evitar que se realicen reservas en habitaciones que ya estén ocupadas, que se registre un cliente sin los datos esenciales y que el precio total de la reserva se calcule correctamente sumando los servicios adicionales.

Debe contar con una interfaz **MDI** (Multiple Document Interface) con un **menú** que permita **acceder** a las diferentes funcionalidades de la aplicación (**gestión de habitaciones, clientes, reservas, servicios y reportes**). Finalmente, se debe generar un **reporte con Crystal Reports** que muestre un resumen de las reservas realizadas en un período determinado, incluyendo el nombre del cliente, la habitación reservada, las fechas de la reserva, el total de la factura (con servicios adicionales), y el estado de la reserva. Este reporte debe ser útil para la administración y seguimiento de las reservas del hotel.

7. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de una tienda de productos electrónicos

Equipos: 4 personas

Enunciado:

Una tienda en línea de productos electrónicos necesita una aplicación para la gestión de **productos, ventas, clientes y proveedores**. El sistema debe permitir la administración eficiente de inventarios, el seguimiento de ventas y el control de proveedores.

La aplicación debe contar con un módulo de gestión de **productos**, permitiendo registrar, modificar y eliminar productos con los siguientes datos: **ID autogenerado, foto (con ajuste automático de tamaño y forma si es necesario), nombre del producto, descripción, precio unitario, cantidad en inventario, categoría, y proveedor**. Además, se debe implementar un sistema para registrar las compras de productos, asegurando que se actualice el inventario correctamente después de cada compra. El módulo de gestión de **clientes** debe permitir registrar, modificar y eliminar clientes con información como **ID autogenerado, foto con ajuste automático de tamaño y forma si es necesario, nombre completo, correo electrónico, teléfono, dirección y fecha de registro**.

El sistema debe permitir registrar **ventas** de productos. Al registrar una **venta**, se debe seleccionar un cliente desde una lista predefinida y agregar los productos vendidos, incluyendo **cantidad y precio total**. Se debe garantizar que el sistema no permita vender productos si no hay suficiente stock disponible. Además, cada **venta** debe registrar la **fecha de la venta, el total de la venta y el estado (Pendiente, Pagado, Cancelado)**. Para la gestión de **proveedores**, se debe permitir registrar, modificar y eliminar proveedores con información como **ID autogenerado, nombre del proveedor, contacto, teléfono y dirección**. La aplicación debe permitir realizar un seguimiento de los productos asociados a cada proveedor, y mostrar información sobre los productos que requieren reabastecimiento.

Se deben desarrollar **procedimientos almacenados** para insertar, actualizar y eliminar productos, clientes, proveedores y ventas. Además, se deben crear procedimientos que permitan **consultar el inventario actual de productos, obtener la lista de ventas realizadas en un período de tiempo y calcular el total de ventas en ese período**.

La aplicación debe mejorar la experiencia del usuario mediante **validaciones** que eviten la venta de productos con inventario insuficiente, mensajes de confirmación antes de eliminar registros y alertas cuando un producto se está quedando sin stock.

Finalmente, se debe generar un **reporte con Crystal Reports** que muestre un resumen de las **ventas realizadas en un período determinado**, con los siguientes detalles: **nombre del cliente, productos vendidos, cantidad, precio total de cada venta y estado de la venta**. El reporte también debe incluir un **resumen de los ingresos totales generados** en ese período, lo que permitirá al administrador tomar decisiones sobre reabastecimiento y estrategias de ventas.

8. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de Reservas en un Restaurante

Equipos: 4 personas

Enunciado:

Un restaurante desea implementar un sistema para gestionar las **reservas de mesas**, el **menú de platillos** y los **clientes**. El sistema debe facilitar la administración de las reservas, el control de disponibilidad de mesas, y la gestión de la información de los clientes que realizan las reservas.

La aplicación debe permitir la **gestión de mesas**, donde se pueda registrar, editar y eliminar mesas. Cada mesa debe tener un **ID autogenerado**, un **número de mesa**, una **capacidad de personas** y un **estado** (disponible, ocupada, reservada).

El **menú de platillos** debe ser gestionado de forma que se puedan agregar, editar y eliminar platillos. Cada platillo debe tener un **ID autogenerado**, **nombre**, **descripción**, **foto (con ajuste automático de tamaño y forma si es necesario)**, **precio** y **categoría** (entrante, principal, postre, bebida).

Para las **reservas** de clientes, la aplicación debe permitir registrar la **información del cliente** (nombre completo, foto (con ajuste automático de tamaño y forma si es necesario), correo electrónico, teléfono) y **detalles de la reserva** (mesa seleccionada, fecha y hora de la reserva, número de personas). Además, debe validarse que no se puedan realizar reservas para una mesa que ya esté ocupada en el horario seleccionado. También se debe permitir la cancelación de reservas.

Se deben desarrollar **procedimientos almacenados** que permitan insertar, buscar, actualizar y eliminar mesas, platillos y reservas. Consultar la disponibilidad de mesas en un horario específico. Consultar las reservas de un cliente. Ver las reservas programadas para un día específico. Consultar el total de ingresos generados por las reservas realizadas en un período determinado.

La interfaz de la aplicación debe permitir a los usuarios realizar todas las gestiones necesarias a través de formularios con validaciones para evitar campos vacíos y garantizar la integridad de los datos. Se debe incluir un **menú MDI** que permita navegar entre la gestión de mesas, platillos, reservas y reportes.

Además, la aplicación debe generar un **reporte con Crystal Reports** que detalle todas las reservas realizadas en un período, mostrando la mesa reservada, el nombre del cliente, la fecha y hora de la reserva, el número de personas y el total de ingresos generados por las reservas.

9. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de una Clínica Dental

Equipos: 4 personas

Enunciado

Una clínica dental necesita una aplicación para gestionar el control de **pacientes, citas, tratamientos y pagos**, optimizando la administración y la atención médica.

La aplicación debe incluir un módulo de gestión de **pacientes** donde se puedan registrar, editar y eliminar datos como **ID autogenerado, nombre completo, foto (con ajuste automático de tamaño y forma si es necesario), fecha de nacimiento, teléfono, correo electrónico, dirección e historial de tratamientos**.

También debe contar con un módulo de **citas**, donde los pacientes puedan agendar consultas con los odontólogos disponibles. Cada cita debe tener un **ID autogenerado, paciente, odontólogo, fecha y hora, motivo de la consulta y estado (pendiente, atendida, cancelada)**.

Además, debe permitir el registro de **tratamientos** realizados en cada consulta, especificando el tipo de tratamiento, costo y observaciones del odontólogo.

El sistema de pagos debe registrar los **pagos** de los pacientes, incluyendo ID autogenerado, paciente, tratamiento asociado, monto, método de pago y estado (pagado, pendiente).

Los **procedimientos almacenados** deben permitir insertar, buscar, actualizar y eliminar pacientes, citas, tratamientos y pagos. También se deben implementar consultas para obtener citas por odontólogo, historial de tratamientos de un paciente, y reportes de ingresos por mes.

Se deben **implementar validaciones** para evitar que un paciente agende dos citas a la misma hora, que se registren pagos sin tratamiento asociado y que se muestren alertas para citas canceladas.

La aplicación debe contar con una interfaz **MDI** (Multiple Document Interface) con un **menú** para **acceder** a la gestión de **pacientes, citas, tratamientos y reportes**.

Finalmente, se debe generar un reporte con Crystal Reports que muestre las citas atendidas en un período determinado, detallando nombre del paciente, odontólogo, tratamiento realizado y monto pagado.

10. Práctica de Examen

Herramientas permitidas: Visual Studio Community 2022, SQL Server Express

Tema: Gestión de ventas en farmacia

Equipos: 4 personas

Enunciado

Una farmacia necesita una aplicación para gestionar el inventario de **medicamentos, las ventas, los clientes y los proveedores**, garantizando un control eficiente de los productos y optimizando la administración del negocio.

La aplicación debe incluir un módulo de **gestión de medicamentos**, permitiendo registrar, editar y eliminar datos como ID autogenerado, nombre del medicamento, descripción, categoría (analgésico, antibiótico, etc.), fecha de vencimiento, precio unitario, stock disponible y proveedor asociado.

También debe contar con un módulo de **gestión de clientes**, donde se registren los datos del cliente, incluyendo ID autogenerado, foto con ajuste automático de tamaño y forma si es necesario, nombre completo, teléfono y dirección.

El sistema de **ventas** debe permitir registrar la venta de medicamentos, asignando un ID autogenerado, cliente asociado (opcional si es venta al público), fecha de la venta, lista de medicamentos comprados, cantidad de cada medicamento y el total de la compra. La venta debe actualizar automáticamente el stock de los medicamentos. Además, se debe gestionar a los **proveedores**, registrando ID autogenerado, nombre del proveedor, contacto, teléfono y lista de medicamentos suministrados.

Los **procedimientos almacenados** deben permitir insertar, buscar, actualizar y eliminar medicamentos, clientes, ventas y proveedores. También deben permitir consultas para obtener los medicamentos próximos a vencer, ventas por período, los medicamentos más vendidos y los proveedores más utilizados.

Se deben implementar **validaciones** para evitar que se realicen ventas sin stock suficiente y que no se registren clientes sin datos esenciales.

La aplicación debe contar con una **interfaz MDI (Multiple Document Interface)** con un menú para acceder a la gestión de medicamentos, clientes, ventas, proveedores y reportes.

Finalmente, se debe generar un **reporte con Crystal Reports** que muestre un resumen de las ventas realizadas en un período determinado, detallando nombre del cliente, medicamentos vendidos, cantidad, total de la compra y método de pago. Este reporte será útil para el control de ingresos y toma de decisiones dentro de la farmacia.