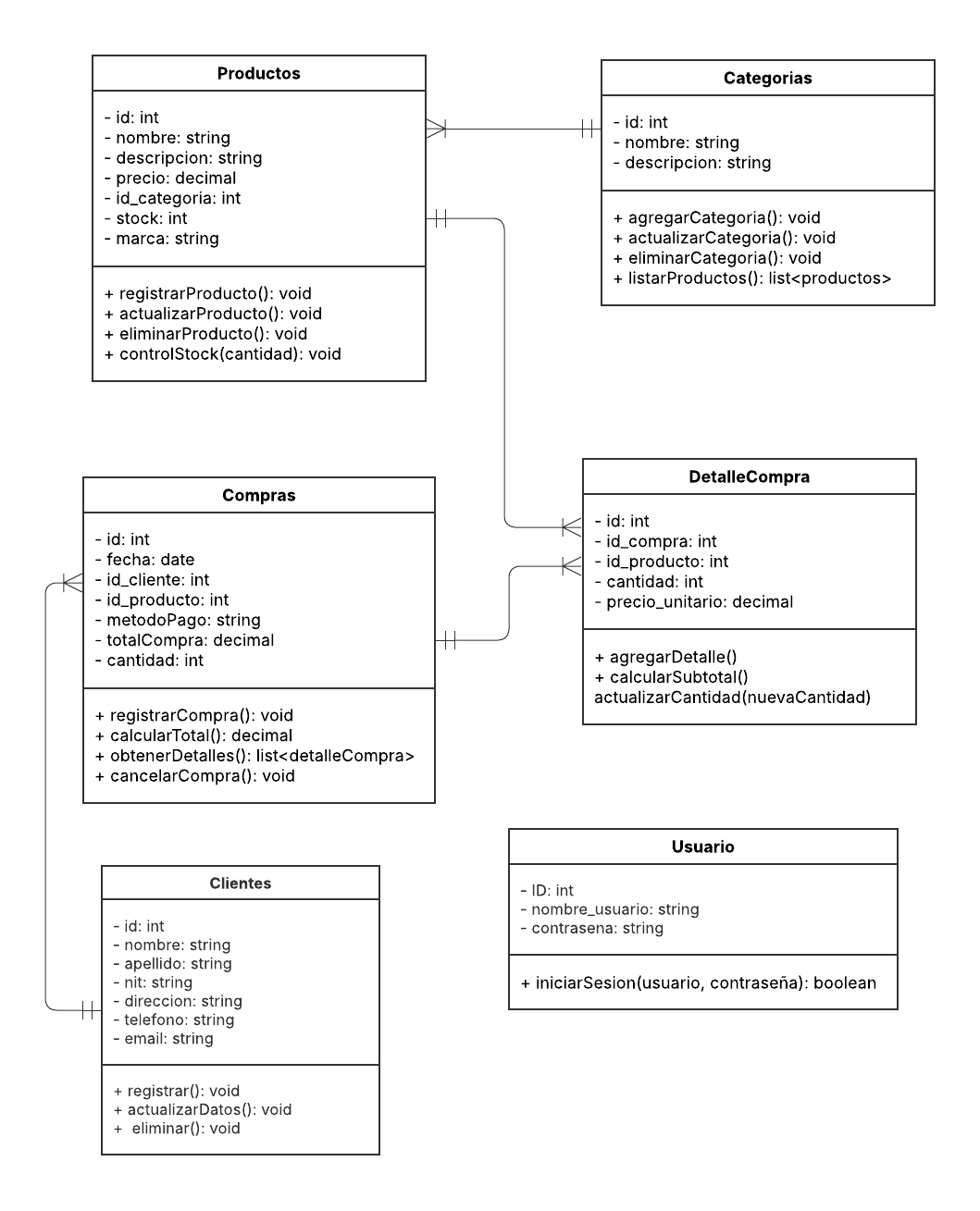
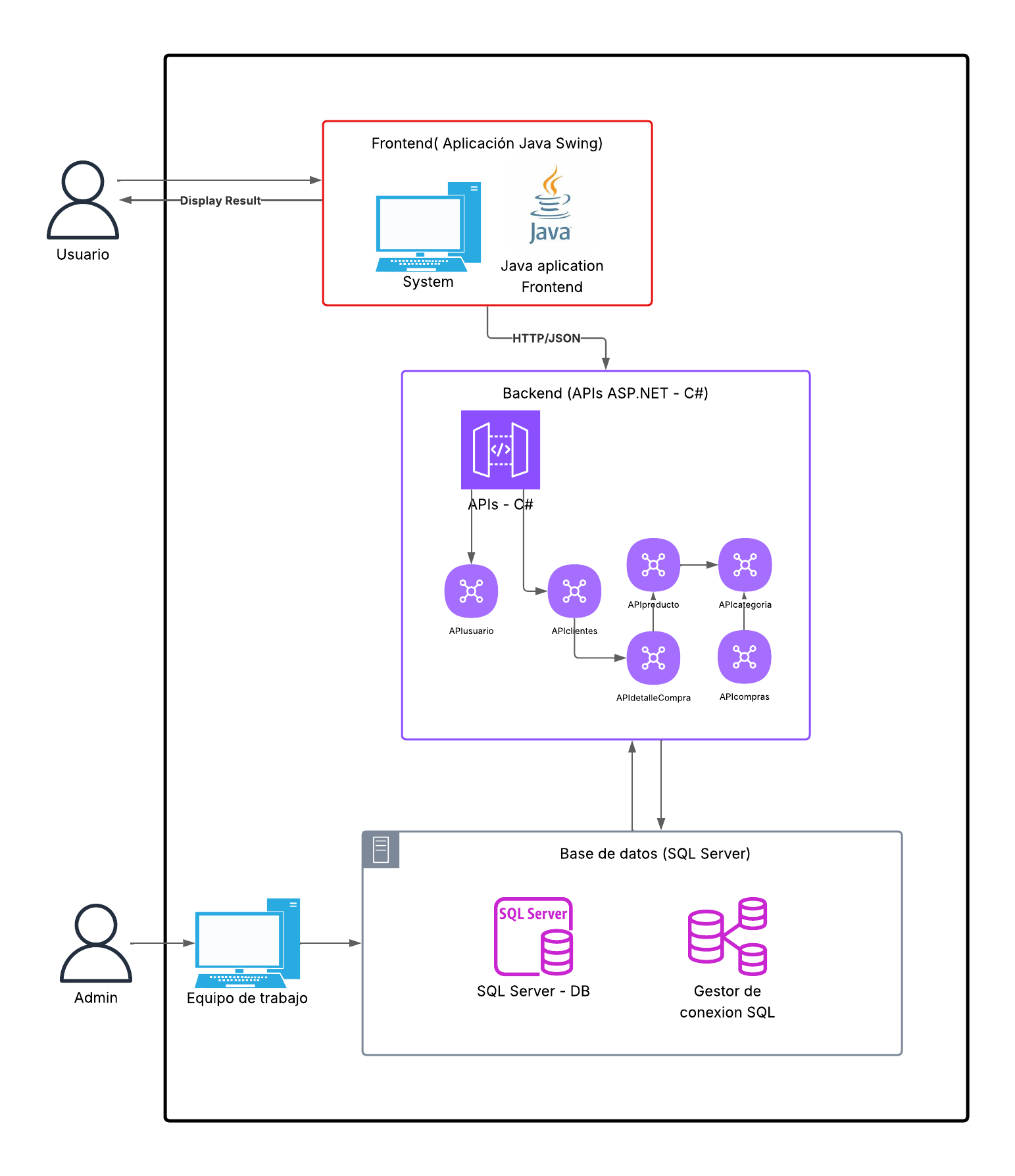
**Diagramas del proyecto**

**Diagrama de clases**



**Diagrama de arquitectura general del sistema**



**Metodología de desarrollo “Cascada”**

**Justificación del uso de la Metodología en Cascada**

Para el desarrollo del sistema de ventas, que incluye los módulos de Gestión de Productos y Gestión de Clientes, se optó por emplear la metodología de desarrollo en cascada. Esta decisión fue tomada considerando las características del proyecto, la estabilidad de los requisitos y la organización del equipo de trabajo.

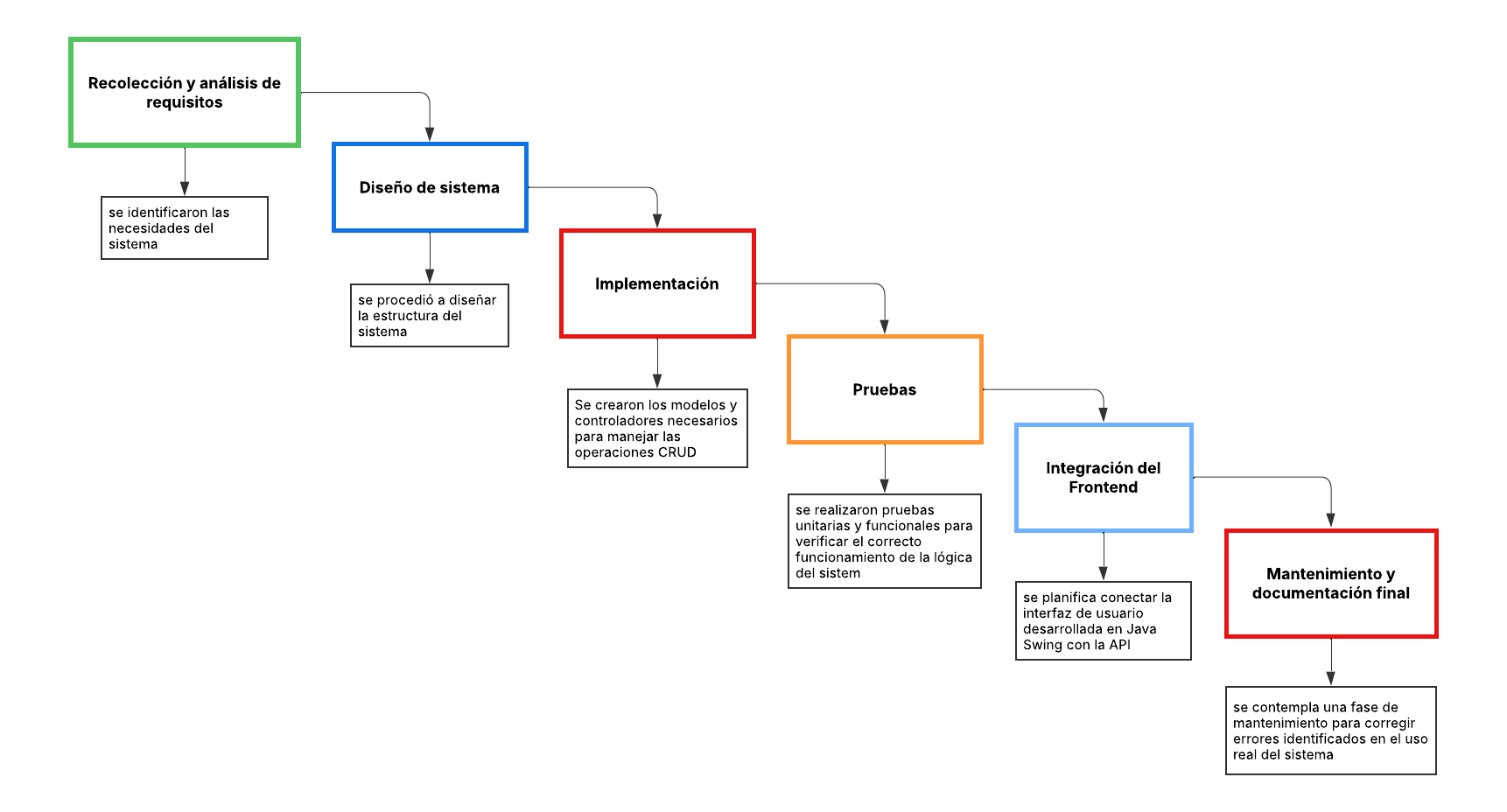
A continuación, se detallan las razones principales por las que se seleccionó esta metodología:

1. **Claridad en los requisitos desde el inicio**: desde la fase de análisis, los requerimientos funcionales del sistema fueron definidos de forma clara, concreta y sin ambigüedades. Esto permitió estructurar un plan de desarrollo donde cada fase tuviera objetivos delimitados y resultados esperados, tal como lo propone el modelo en cascada.
2. **Proceso secuencial ordenado y definidas**: la naturaleza lineal de esta metodología favorece la planificación y ejecución ordenada del proyecto. En este caso, se comenzó por la documentación y análisis, se continuó con el diseño de la base de datos y la creación de la API (backend), y posteriormente se desarrollará la interfaz gráfica con Java Swing (frontend). Esta estructura secuencial es coherente con el enfoque en cascada.
3. **Énfasis en la documentación técnica**: el enfoque en cascada prioriza una documentación exhaustiva antes de iniciar cada fase. Esto resultó especialmente útil para nuestro equipo, ya que se elaboraron especificaciones técnicas que facilitaron el desarrollo y permitieron una comprensión integral del sistema por parte de todos los miembros.
4. **Facilidad de gestión y control del proyecto**: el modelo en cascada permite llevar un control riguroso del avance del proyecto, con entregables definidos al finalizar cada etapa. Esto ha permitido una asignación eficiente de tareas, mayor control de tiempos y una verificación clara del cumplimiento de los objetivos.
5. **Adecuado para Proyectos Académicos o de Corto Alcance**: dado que se trata de un sistema de ventas con funcionalidades bien definidas desde el inicio y con bajo riesgo de cambios a mitad del desarrollo, la metodología en cascada resulta adecuada. Además, su aplicación es común en entornos académicos o proyectos con objetivos concretos y plazos definidos.

**Descripción del proceso de desarrollo aplicado**

A continuación, se describe cada una de estas etapas y cómo fueron aplicadas en el proyecto:

1. **Fase de Recolección y Análisis de Requisitos**: en esta primera etapa se identificaron las necesidades del sistema, a través de reuniones de grupo y análisis de casos similares. Se definieron los módulos principales (Gestión de Productos y Gestión de Clientes), así como las funcionalidades necesarias, como el CRUD de productos y clientes, el control de stock, y el registro de compras.
2. **Fase de Diseño del Sistema**: con los requisitos claros, se procedió a diseñar la estructura del sistema. Se elaboraron los diagramas entidad-relación para la base de datos, definiendo tablas como **categoría, productos, clientes, compras, detalleCompra y usuarios.** También se diseñó la arquitectura de la API y su integración futura con la interfaz gráfica.
3. **Fase de Implementación**: en esta etapa se desarrolló la base de datos en SQL Server y se programó la API utilizando un lenguaje backend ( ASP.NET – C#, según lo utilizado por el grupo). Se crearon los modelos y controladores necesarios para manejar las operaciones CRUD de productos y clientes.
4. **Fase de Pruebas**: una vez finalizada la implementación del backend, se realizaron pruebas unitarias y funcionales para verificar el correcto funcionamiento de la lógica del sistema, validación de datos, y manejo de errores. Esto aseguró la integridad de la base de datos y la correcta respuesta de la API a distintas peticiones.
5. **Fase de Integración del Frontend**: en esta fase se planifica conectar la interfaz de usuario desarrollada en Java Swing con la API previamente construida. Esta conexión permitirá realizar las operaciones desde una interfaz gráfica, como la gestión de productos, clientes y visualización de registros de compras.
6. **Fase de Mantenimiento y Documentación Final**: finalmente, se contempla una fase de mantenimiento para corregir errores identificados en el uso real del sistema, así como la entrega de la documentación técnica completa que detalle la arquitectura del sistema, los endpoints de la API, y las instrucciones de uso del software.

**Diagrama de modelo de cascada:**