



Ingeniería de Sistemas
Universidad Popular del Cesar



Patrones de diseño

Integrantes:

Juan David Ayala

Daniel Novoa Castellar

Brayan Javier Acuña

Carlos Lindarte Santos

Universidad Popular del Cesar

Facultad de Ingenierías y Tecnológicas: Ingeniería de sistemas

Patrones de diseño Electiva de profundización

Docente: Jairo Seoanes

Valledupar

12 de marzo de 2025



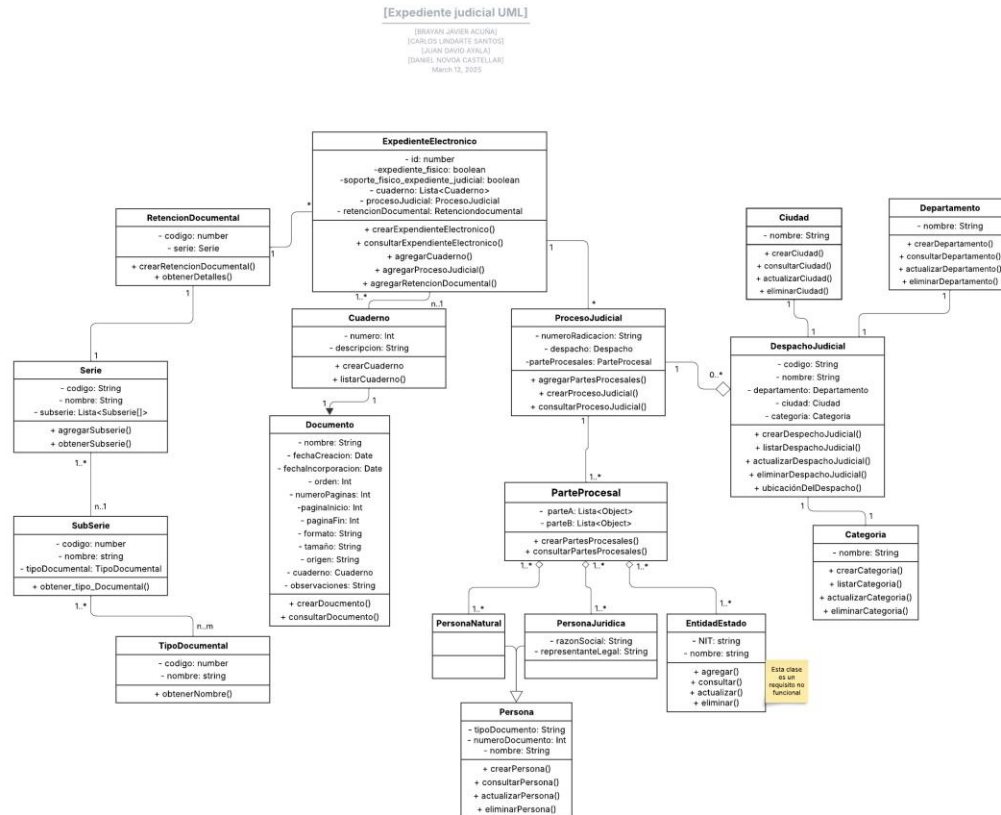
Taller nro 1: Introducción a UML y Diseño Orientado a Objetos

Desarrollo de la actividad:

- Diagrama UML
 - Desarrollo de clases y relaciones
- Link del repositorio con el código alojado
- Capturas de pantalla de la estructura del archivo arrojado en Excel
- Herramientas usadas para llevar a cabo la actividad
- Conclusiones



Diagrama UML



Un diagrama UML normalmente es una herramienta utilizada en el diseño de software para representar la estructura y el comportamiento de un sistema. Permite visualizar clases, atributos, métodos y las relaciones entre ellas, facilitando la organización y comprensión de los componentes de un sistema complejo.

En este caso, nuestro diagrama se aplica al diseño de un sistema de gestión de expedientes judiciales, donde se estructuran entidades clave como los expedientes, documentos, procesos judiciales y participantes. Este modelo permite la organización de información legal teniendo en cuenta los patrones de diseño para el desarrollo del mismo.



Clases y Relaciones Principales

Expediente Electrónico

Representa el archivo digital de un caso judicial.

Se relaciona con Cuaderno (1..n), que agrupa los documentos del expediente.

Tiene una relación con Proceso Judicial (1..1), indicando el trámite legal asociado.

Vinculado a Retención Documental (1..1), que gestiona la conservación de documentos.

Proceso Judicial

Define el trámite legal y sus procedimientos.

Se asocia a un Despacho Judicial (1..1), que gestiona el proceso.

Contiene múltiples Partes Procesales (0..), los sujetos involucrados en el caso.

Parte Procesal y Persona

Parte Procesal representa a los actores de un proceso judicial.

Está asociada a la clase Persona (1..), que puede ser:

Persona Natural (individuos).

Persona Jurídica (empresas con razón social y representante legal).



Entidad del Estado (instituciones con NIT).

Documento y Cuaderno

Documento almacena información dentro del sistema.

Está vinculado a un Cuaderno (1..1), que organiza los documentos dentro de un expediente.

Despacho Judicial y su Ubicación

Representa las oficinas judiciales encargadas de procesos.

Relacionado con Ciudad y Departamento (1..1 cada uno).

Posee una Categoría (1..1), que puede indicar el tipo de despacho.

Link del repositorio:

<https://github.com/misterio562/Expediente>

Capturas de pantalla:

[illegible]

Herramientas utilizadas:

Diseño del diagrama UML: lucidapp

Lenguaje de programación: Python



Conclusiones

Trabajar en este diagrama UML fue más que solo interpretar una estructura; requirió investigar sobre el funcionamiento real de un sistema de expedientes judiciales. No basta con conectar clases al azar, sino que fue necesario entender cómo se relacionan en el mundo legal los expedientes, documentos, procesos y participantes.

Uno de los mayores aprendizajes fue notar que ciertas relaciones no son tan obvias al principio. Por ejemplo, la conexión entre Despacho Judicial y Ciudad/Departamento parece simple, pero tiene implicaciones importantes en la jurisdicción y manejo de los casos. Lo mismo ocurre con la relación entre Parte Procesal y Persona, donde es clave diferenciar entre personas naturales, jurídicas y entidades estatales.

Además, se evidencia que un Expediente Electrónico no es solo una colección de documentos, sino un sistema estructurado con reglas claras sobre almacenamiento, organización y retención documental. En la práctica, estos detalles pueden definir la eficiencia y seguridad del sistema.

En resumen, este ejercicio no fue solo de modelado, sino también de aprendizaje sobre el manejo de información legal y cómo se estructura en un sistema digital. A veces, entender un diagrama UML significa adentrarse en el dominio real del problema, y este fue un claro ejemplo de ello.