

Capas típicas en el modelo de capas

1. Capa de Presentación (UI o View)

- a. Es la capa que interactúa directamente con los usuarios.
- b. Se encarga de mostrar información y recibir las entradas del usuario.
- c. Ejemplos en Java:
 - i. Interfaces gráficas con *Swing*, *JavaFX*, o aplicaciones web con *JSP*, *Thymeleaf*, o *React*.
- d. **Responsabilidad:** Controlar la experiencia del usuario, mostrar datos y recibir eventos.

2. Capa de Lógica de Negocio (Business Logic o Service Layer)

- a. Contiene las reglas de negocio y procesos de la aplicación.
- b. Aquí se realizan las validaciones, cálculos y lógica específica de la aplicación.
- c. Ejemplos en Java:
 - i. Clases y métodos que representan conceptos como "Cliente", "Factura", "Producto", etc.
 - ii. Servicios definidos con frameworks como *Spring* o *Java EE*.
- d. **Responsabilidad:** Procesar la lógica del negocio y actuar como intermediario entre las capas de presentación y acceso a datos.

3. Capa de Acceso a Datos (DAO o Data Layer)

- a. Es la capa responsable de interactuar con las bases de datos u otros sistemas de almacenamiento.
- b. Se encarga de realizar operaciones como *CRUD* (crear, leer, actualizar, eliminar).
- c. Ejemplos en Java:
 - i. Uso de bibliotecas como *JDBC*, *Hibernate*, o *Spring Data JPA*.
- d. **Responsabilidad:** Abstraer las interacciones con la base de datos u otros almacenes de datos.

4. Capa de Persistencia (Database o Repositorio)

- a. Aunque no siempre se considera una capa "lógica", se refiere al almacenamiento físico de los datos.
- b. Ejemplo:
 - i. Bases de datos como MySQL, PostgreSQL, o sistemas de archivos.