

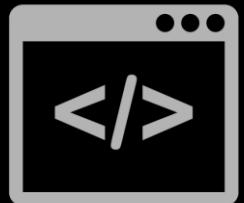
# Diseño de Sistemas



# Agenda

- Introducción a la Arquitectura de Software
- Modelo en Capas

# Introducción a la Arquitectura de Software y de Sistema



# Arquitectura de Software

La arquitectura de software representa la **estructura o estructuras** del sistema que consiste en **componentes de software**, las propiedades **externas** visibles de esos componentes y las **relaciones** entre ellos.

# Arquitectura de Software

Un **componente** es un pieza de software que puede ser código fuente, código binario, un ejecutable, o una biblioteca con una **interfaz definida**.

## ¿Qué es la interfaz de un Componente?

Una interfaz establece las **operaciones externas** de un componente, las cuales determinan una parte del **comportamiento** del mismo.

## ¿Cómo se relacionan con otros componentes?

Se representan las dependencias entre componentes o entre un componente y la interfaz de otro, es decir uno de ellos usa los servicios o funcionalidades del otro.

# Arquitectura de Software - Niveles

**Enterprise**

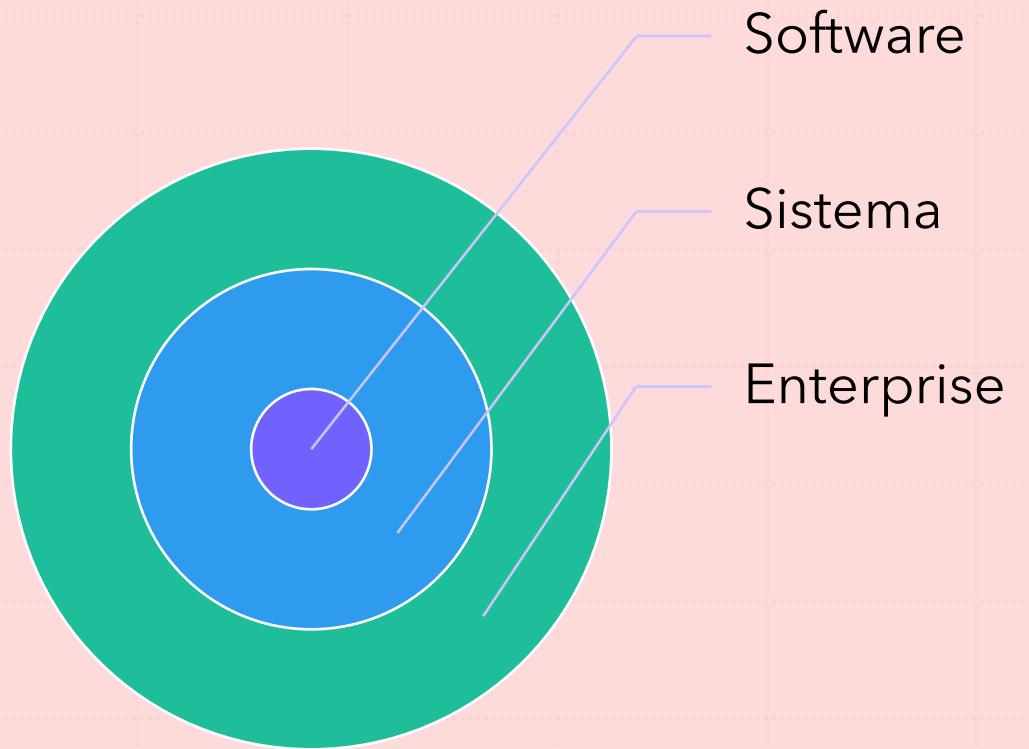
- Define la estrategia tecnológica y de negocio de la organización para el desarrollo de sus Sistemas.

**Sistema**

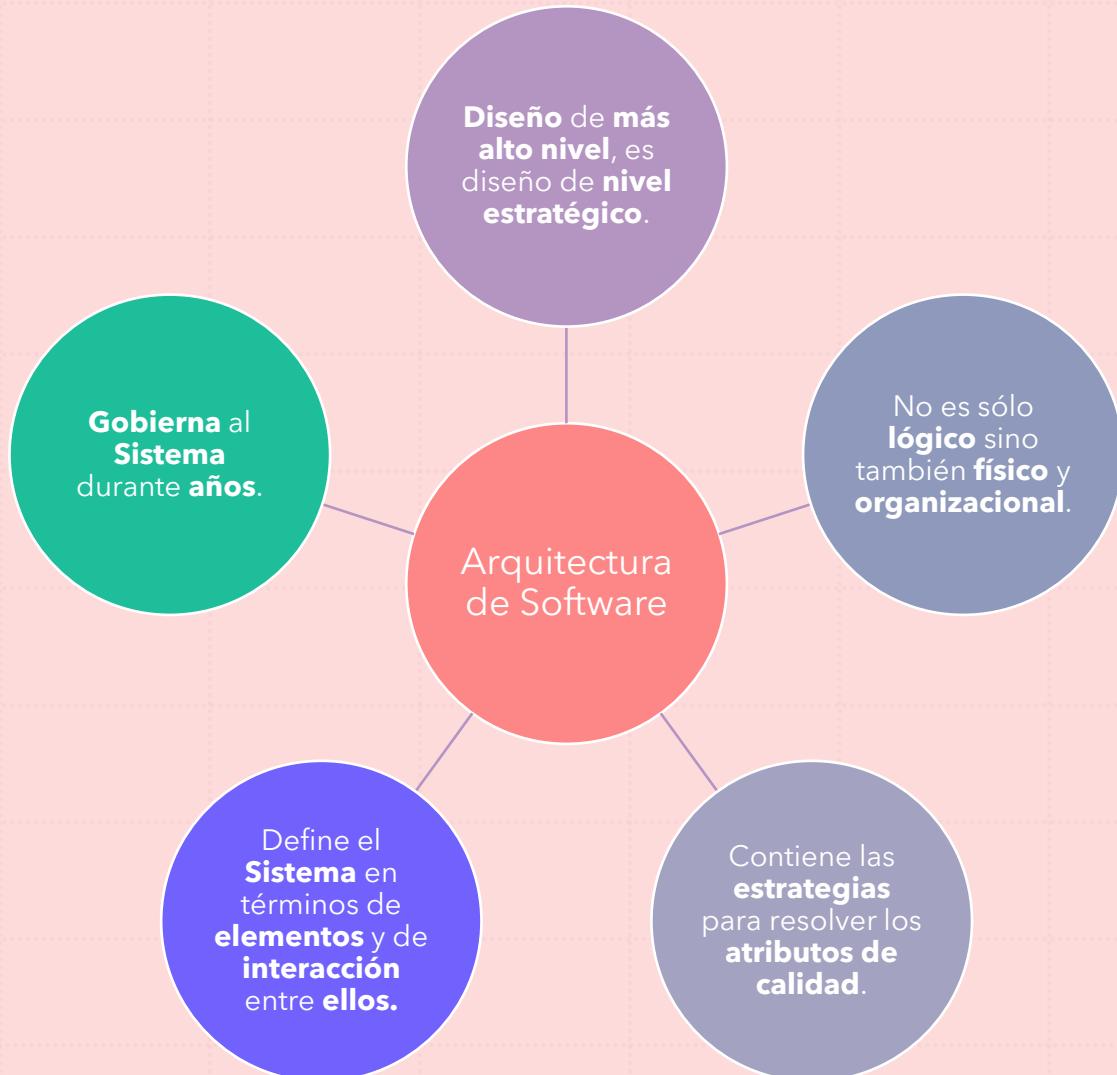
- Arquitectura de Software e Infraestructura.

**Software**

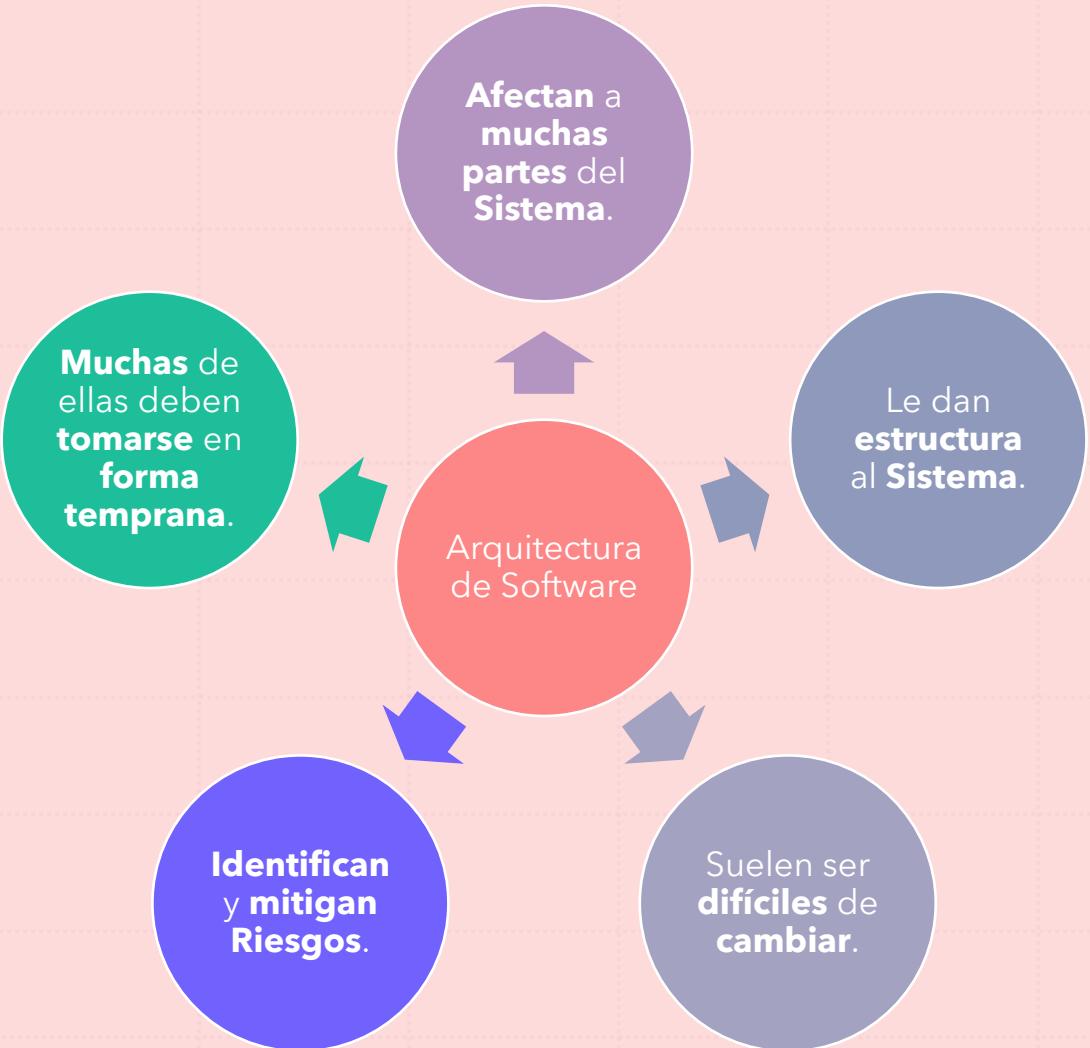
- Arquitectura de Software para una Aplicación o Subsistema.



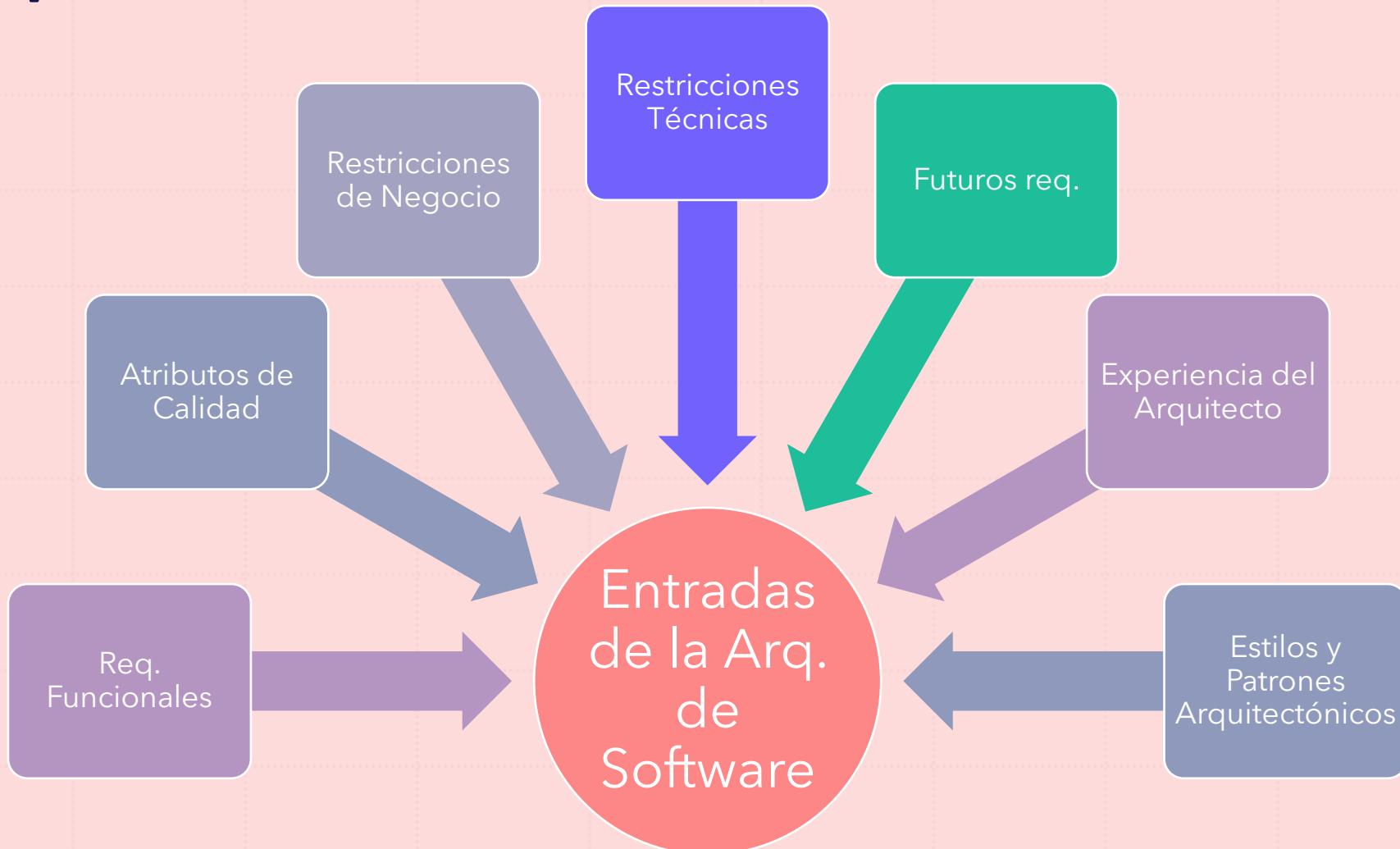
# Arquitectura de Software - Características



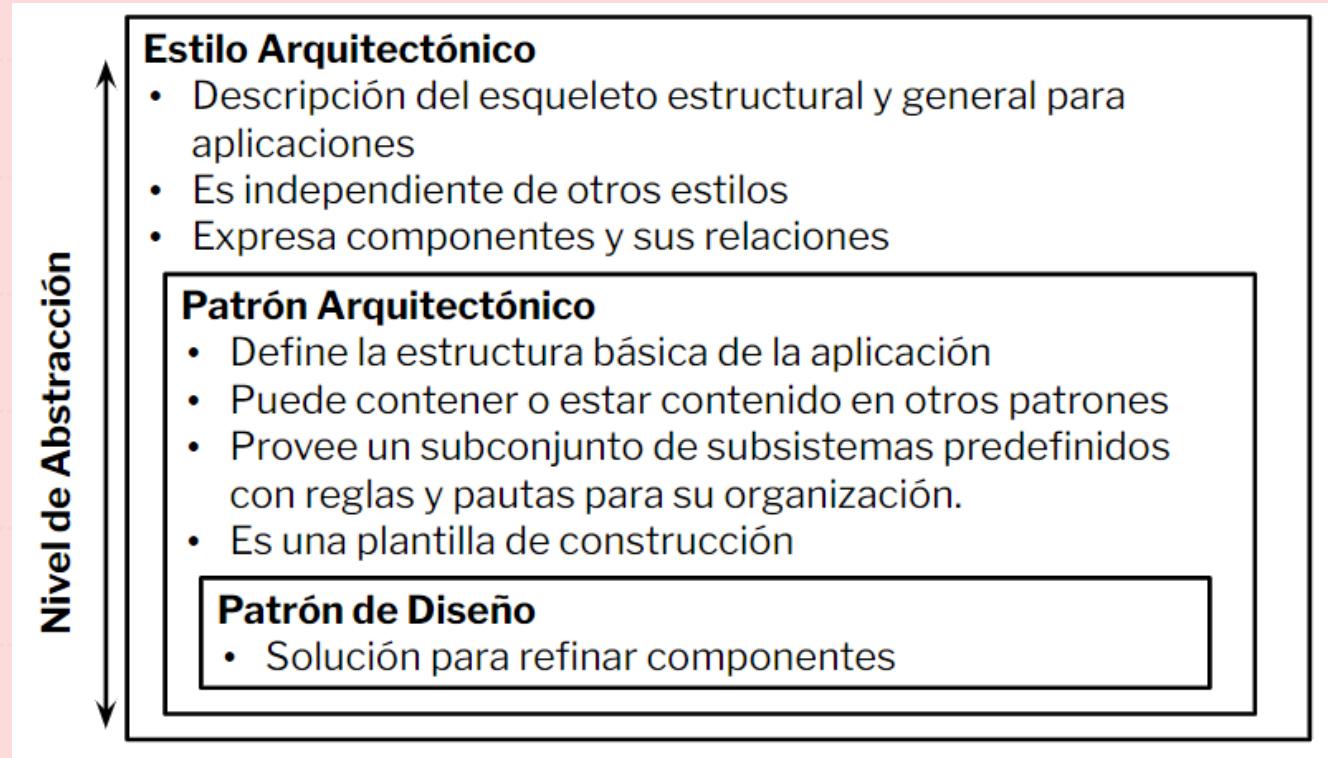
# Arquitectura de Software



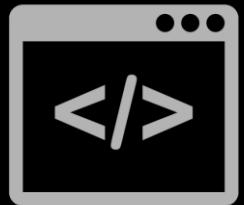
# Arquitectura de Software – Entradas



# Arquitectura de Software – Niveles de Abstracción



# Modelo en Capas (Estilo Arquitectónico)



# Modelo en Capas

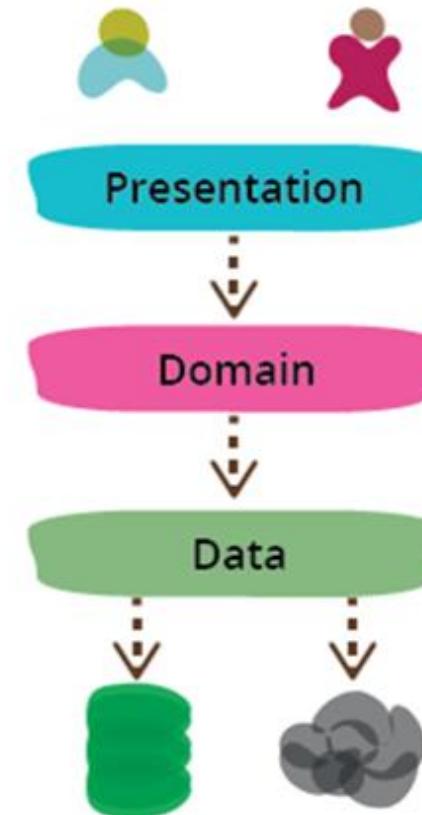
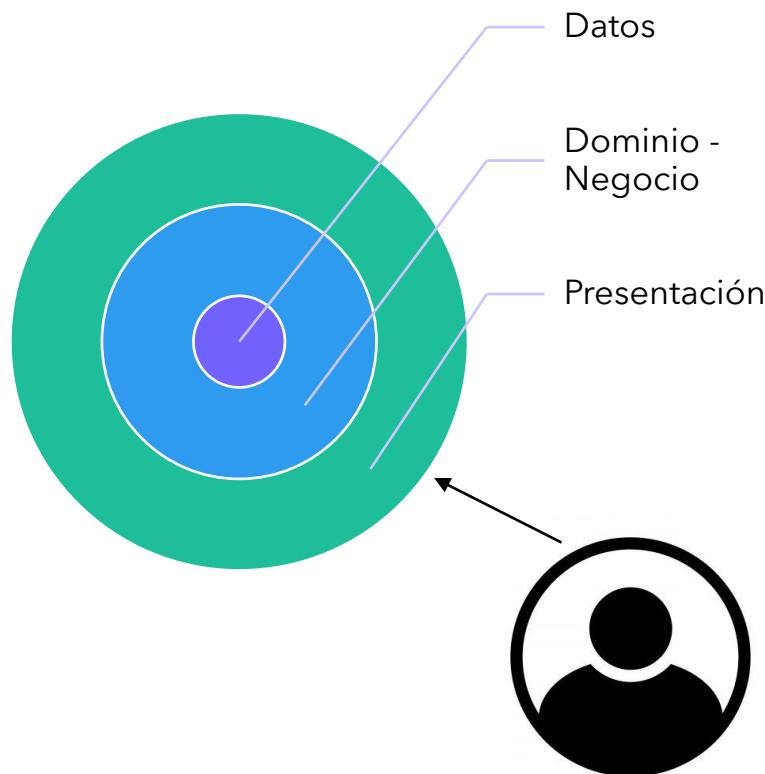
Una analogía: modelo OSI (Redes de Datos)



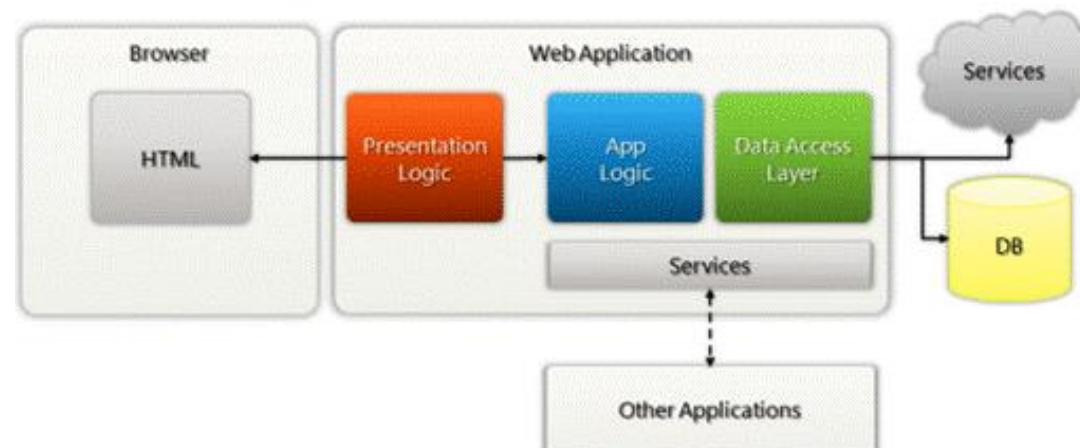
# Modelo en Capas

- Dividir en **capas** (layering) es una de las **técnica más comunes** para resolver un **problema** de **diseño complejo**.
- Cada capa es un **todo coherente** (cohesión), con un **rol único** en el sistema.
- Las **capas superiores usan servicios de las inferiores**, pero no así de forma contraria o saltando niveles.

# Modelo en Capas



# Modelo en Capas



# Modelo en Capas

## Ventajas

- Nos mantiene enfocados en el problema a resolver.
- Esconde el detalle de cómo se llevan a cabo los servicios que expone.
- Puede ser reemplazada la implementación de los servicios siendo transparente para los consumidores.
- Minimiza la dependencia entre componentes.
- Facilita la pruebas.

## Desventajas

- Los cambios pueden generar efecto cascada.
- Demasiadas capas agregan complejidad y afectan negativamente al rendimiento.

# Gracias

