**Arquitectura De 3 Capas**

La programación por capas es un modelo de desarrollo software en el que el objetivo primordial es la separación (desacoplamiento) de las partes que componen un sistema software o también una arquitectura cliente-servidor: lógica de negocios, capa de presentación y capa de datos. De esta forma, por ejemplo, es sencillo y mantenible crear diferentes interfaces sobre un mismo sistema sin requerirse cambio alguno en la capa de datos o lógica.

La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, solo afectará al nivel requerido sin tener que revisar entre el código fuente de otros módulos, dado que se habrá reducido el Acoplamiento Informático hasta una interfaz de paso de mensajes.

Además, permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.

En el diseño de [sistemas informáticos](https://es.wikipedia.org/wiki/Sistema_inform%C3%A1tico) actual se suelen usar las [arquitecturas multinivel](https://es.wikipedia.org/wiki/Arquitectura_software) o programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El más utilizado actualmente es el diseño en tres niveles (o en tres capas).

**Capas y niveles**

* Capa de presentación: la que ve el usuario (también se la denomina «capa de usuario»), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como [interfaz gráfica](https://es.wikipedia.org/wiki/Interfaz_gr%C3%A1fica) y debe tener la característica de ser «amigable» (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.
* Capa de negocio: es donde residen los [programas](https://es.wikipedia.org/wiki/Programa_(computaci%C3%B3n)) que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de [base de datos](https://es.wikipedia.org/wiki/Base_de_datos) almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.
* Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único ordenador, si bien lo más usual es que haya una multitud de ordenadores en donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más [ordenadores](https://es.wikipedia.org/wiki/Ordenador). Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio.

Si, por el contrario, fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o más ordenadores que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de negocio, y otra serie de ordenadores sobre los cuales corre la base de datos.

En una arquitectura de tres niveles, los términos «capas» y «niveles» no significan lo mismo ni son similares.

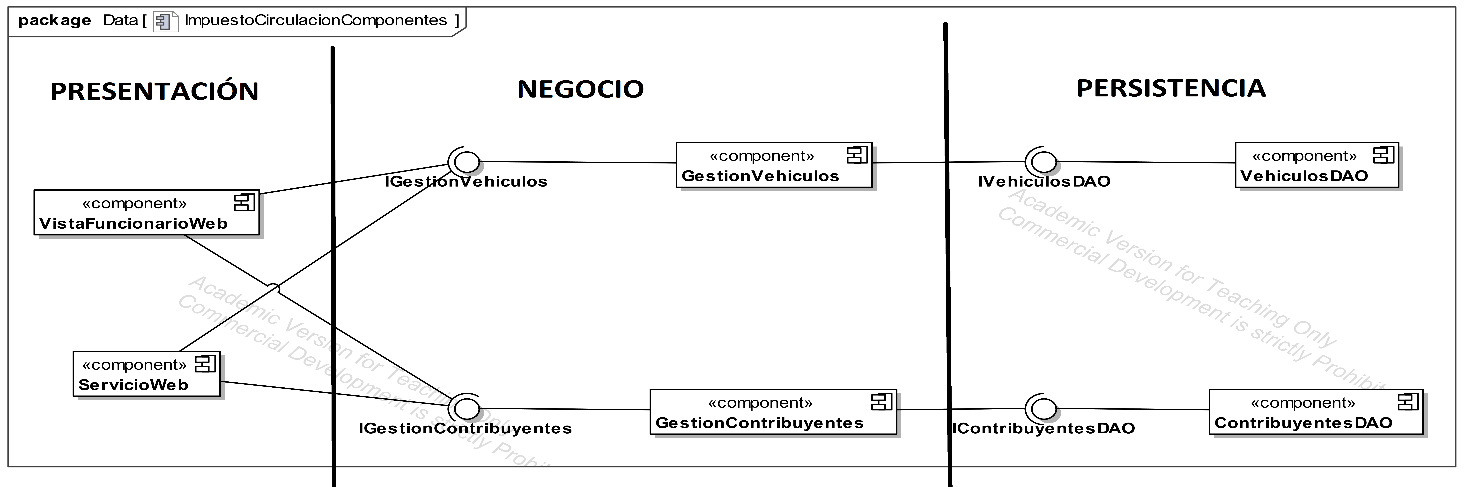
El término «capa» hace referencia a la forma como una solución es segmentada desde el punto de vista lógico:

* Presentación. (Conocida como capa web en aplicaciones web o como capa de usuario en aplicaciones nativas)
* Lógica de negocio. (Conocida como capa aplicativa)
* Datos. (Conocida como capa de base de datos)

En cambio, el término «nivel» corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo:

* Una solución de tres capas (presentación, lógica del negocio, datos) que residen en un solo ordenador (Presentación + lógica + datos). Se dice que la arquitectura de la solución es de tres capas y un nivel.
* Una solución de tres capas (presentación, lógica del negocio, datos) que residen en dos ordenadores (Presentación + lógica por un lado; lógica + datos por el otro lado). Se dice que la arquitectura de la solución es de tres capas y dos niveles.

**¿Cuándo se debe utilizar el modelo tres capas?**

Se suele utilizar en sistemas que implementan un modelo de negocio como por ejemplo una tienda online, una app para gestionar ciertos datos, etc. Sin embargo, no es recomendable usarla para sistemas de tiempo real como los de los coches o los aviones.  


**¿Qué son las capas en el modelo cliente servidor?**

En la arquitectura de tres capas (en general, en la arquitectura multicapa) el cliente implementa la lógica de presentación (cliente fino), el servidor(es) de aplicación implementan la lógica de negocio y los datos residen en uno (o varios) servidor(es) de bases de datos.

**Arquitecturas de tres capas:**

Una generación más moderna de la arquitectura Cliente/Servidor añade una capa intermedia. En la arquitectura de tres capas (en general, en la arquitectura multicapa) el cliente implementa la lógica de presentación (cliente fino), el servidor(es) de aplicación implementan la lógica de negocio y los datos residen en uno (o varios) servidor(es) de bases de datos.

Una arquitectura multicapa se define por tanto por las siguientes tres capas de componentes:

* Un componente front-end que es el responsable de proporcionar la lógica de presentación.
* Un componente back-end que proporciona acceso a servicios dedicados, tales como un servidor de bases de datos.
* Un componente que hace las funciones de capa intermediaria (middl tier) que permite a los usuarios compartir y controlar la lógica de negocio mediante su aislamiento de la aplicación real.

**Ejemplo concreto:**

Servicio de correo electrónico: Funcionamiento del correo:

El correo electrónico o e-mail es una forma de enviar mensajes entre ordenadores conectados a través de Internet.

Como la mayoría de los servicios de Internet el correo se basa en la arquitectura cliente/servidor.

Cuando alguien envía un correo, primero llega a su servidor de correo que lo envía al servidor del destinatario, donde el mensaje queda almacenado en el buzón del destinatario. Cuando el destinatario se conecte al servidor, éste le enviará todos sus mensajes pendientes. Por esto da igual que el destinatario esté conectado o no a Internet en el momento que se le envía un mensaje.

