**Arquitectura de 3 niveles**

Es una arquitectura de aplicaciones de software consolidada que organiza las aplicaciones en tres niveles informáticos lógicos y físicos: el nivel de presentación; el nivel de aplicación, y el nivel de datos.

**Nivel de presentación**

El nivel de presentación es la interfaz de usuario y la capa de comunicación de la aplicación, donde el usuario final interactúa con la aplicación.

Su objetivo principal es mostrar información al usuario y recopilar información del usuario.

Se puede ejecutar en un navegador web, como una aplicación de escritorio o una interfaz gráfica de usuario.

Los niveles de presentación web normalmente se desarrollan utilizando HTML, CSS y JavaScript.

**Nivel de aplicación**

También conocido como el nivel lógico o el nivel medio, es el núcleo de la aplicación.

Se procesa la información recopilada en el nivel de presentación, a veces contra otra información en el nivel de datos, utilizando la lógica empresarial, un conjunto específico de reglas de negocio.

El nivel de aplicación también puede añadir, suprimir o modificar datos en el nivel de datos.

El nivel de aplicación normalmente se desarrolla utilizando Python, Java, Perl, PHP o Ruby, y se comunica con el nivel de datos utilizando llamadas de API.

**Nivel de datos o back-end**

Almacena y gestiona la información procesada por la aplicación. Puede ser un sistema de gestión de bases de datos relacional como o en un servidor de bases de datos

Toda la comunicación pasa por el nivel de aplicación. El nivel de presentación y el nivel de datos no pueden comunicarse directamente entre sí.

**Ventajas**

El beneficio de la arquitectura de tres niveles es su separación lógica y física de la funcionalidad. Cada nivel puede ejecutarse en un sistema operativo y una plataforma de servidor independientes. Cada nivel se ejecuta en al menos un hardware de servidor dedicado o un servidor virtual, por lo que los servicios de cada nivel se pueden personalizar y optimizar sin que afecte a los demás niveles.

Otras ventajas incluyen:

**Desarrollo más rápido:**

Como cada nivel puede ser desarrollado simultáneamente por diferentes equipos, una organización puede lanzar una aplicación al mercado más rápido, y los programadores pueden utilizar los más recientes y mejores lenguajes y herramientas para cada nivel.

**Escalabilidad mejorada:**

Cualquier nivel se puede escalar de forma independiente de los demás según sea necesario.

**Fiabilidad mejorada:**

Una interrupción en un nivel es menos probable que afecte a la disponibilidad o al rendimiento de los otros niveles.

**Seguridad reforzada:**

Debido a que el nivel de presentación y el nivel de datos no se pueden comunicar directamente, un nivel de aplicación bien diseñado puede funcionar como una especie de cortafuegos interno, lo que impide las inyecciones de SQL y otras vulnerabilidades maliciosas.