

Estilo Arquitectural

En Capas (*N-Layer*)

Descripción

El estilo arquitectural en capas se basa en una distribución jerárquica de los roles y las responsabilidades para proporcionar una división efectiva de los problemas a resolver. Los roles indican el tipo y la forma de la interacción con otras capas y las responsabilidades la funcionalidad que implementan.



Características

- Descomposición de los servicios de forma que la mayoría de interacciones ocurre solo entre capas vecinas.
- Las capas de una aplicación pueden residir en la misma máquina o pueden estar distribuidos entre varios equipos.
- Los componentes de cada capa se comunican con los componentes de otras capas a través de interfaces bien conocidos.
- Cada nivel agrega las responsabilidades y abstracciones del nivel inferior.

<p>Principios Clave</p>	<ul style="list-style-type: none"> • Muestra una vista completa del modelo y a la vez proporciona suficientes detalles para entender las relaciones entre capas. • No realiza ninguna suposición sobre los tipos de datos, métodos, propiedades y sus implementaciones. • Separa de forma clara la funcionalidad de cada capa. • Cada capa contiene la funcionalidad relacionada solo con las tareas de esa capa. • Las capas inferiores no tienen dependencias de las capas superiores. • La comunicación entre capas está basada en una abstracción que proporciona un bajo acoplamiento entre capas.
<p>Beneficios</p>	<ul style="list-style-type: none"> • Abstracción ya que los cambios se realizan a alto nivel y se puede incrementar o reducir el nivel de abstracción que se usa en cada capa del modelo. • Aislamiento ya que se pueden realizar actualizaciones en el interior de las capas sin que esto afecte al resto del sistema. • Rendimiento ya que distribuyendo las capas en distintos niveles físicos se puede mejorar la escalabilidad, la tolerancia a fallos y el rendimiento. • Testeabilidad ya que cada capa tiene una interfaz bien definida sobre la que realizar las pruebas y la habilidad de cambiar entre diferentes implementaciones de una capa. • Independencia ya que elimina la necesidad de considerar el hardware y el despliegue así como las