



JEE5 Patterns

Escuela de arquitectura y programación

Especialidad Java/XML/JEE

[Indra](#) & [Icono Training Consulting](#)

TRAINING CONSULTING



La plataforma JEE

- La plataforma empresarial de Java proporciona una fundación para la creación de sistemas distribuidos (opcionalmente), estructurados en capas construidas en base a componentes predefinidos

Componentes de JEE

- Aplicaciones cliente Java
 - Ejecución remota de aplicaciones Java de escritorio
- Clientes Web
 - Aplicaciones web estándar
- Componentes de negocio
 - Principalmente Enterprise Java Beans (EJBs)

Modelo estructurado en capas I

1. Capa de cliente
2. Capa de presentación
3. Capa de negocio
4. Capa de integración
5. Capa de recursos

Modelo estructurado en capas II

Capa de cliente

Es la capa donde se localizan los diferentes clientes de nuestra aplicación. Los tipos más frecuentes son aplicaciones de escritorio, navegadores web y clientes para dispositivos móviles.

Modelo estructurado en capas III

Capa de presentación

Contiene toda la lógica de interacción entre el usuario y la aplicación. Además está encargada de controlar la interacción entre el usuario y la lógica de negocio, generando las vistas necesarias para mostrar la información en la forma adecuada.

Modelo estructurado en capas IV

Capa de negocio

En ella se localiza el código y las reglas que sirven como núcleo de nuestras aplicaciones empresariales. Debe cumplir una serie de requisitos no funcionales como extensibilidad, mantenibilidad, reutilización, flexibilidad y fácil adopción de tecnologías, etc.

Modelo estructurado en capas V

Capa de integración

Realiza las tareas necesarias para integrar nuestra aplicación con otros sistemas como sistemas de acceso a datos, sistemas legado, motores de workflow, etc.

Esta capa debe ser extensible para que acepte nuevas fuentes sin que esto afecte a la lógica de negocio.

Modelo estructurado en capas VI

Capa de recursos

En ella se encuentran los diferentes Sistemas de Información de nuestra empresa: bases de datos, sistemas de ficheros COBOL, sistemas ERP, CRM, etc.

Patrones de la capa de Integración

1. Data Access Object: encapsular por completo el acceso a datos en su propia capa
2. Service Activator: ejecutar servicios de forma asíncrona
3. Domain Store: mantener la lógica de persistencia separada de los objetos persistentes
4. Web Service Broker: exponer y reutilizar servicios web a través de una interfaz común

Patrones de la capa de negocio

1. **Business Delegate**: presentar una interfaz a los clientes lo más simple posible
2. **Service Locator**: localizar servicios y componentes de una manera uniforme y encapsulada
3. **Session Facade**: exponer a clientes servicios y componentes remotos
4. **Application Service**: centralizar peticiones en un único objeto capaz de delegar en otros
5. **Business Object**: integrar lógica de negocio y persistencia de cara a los clientes
6. **Composite Entity**: uso de beans de entidad para implementar el modelo de dominio
7. **Transfer Object**: agregar datos y transferirlos entre capas
8. **T O Assembler**: agregar transfer objects y otros artefactos
9. **Value List Handler**: facilitar a clientes el recorrido de grandes conjuntos de datos

Patrones de la capa de presentación

1. **Intercepting Filter:** manipular la request antes y después de que se procese
2. **Context Object:** encapsular información global específica del sistema
3. **Front Controller:** centralizar la recepción de la request, delegar en controladores específicos
4. **Application Controller:** centralizar, modularizar y sincronizar vistas con acciones
5. **View Helper:** eliminar de las vistas toda la lógica ortogonal a la de presentación
6. **Composite View:** construir una vista agregando segmentos parciales
7. **Dispatcher View:** manejo limitado de la request y response desde una vista
8. **Service To Worker:** generación dinámica de vistas