

Patrón Modelo Vista Controlador

Patrones de diseño

➤ Describen los distintos problemas "estándares" que nos podemos encontrar en el desarrollo de cualquier aplicación, junto con la correspondiente solución a aplicar a través de la tecnología Java EE

➤ Entre sus principales ventajas:

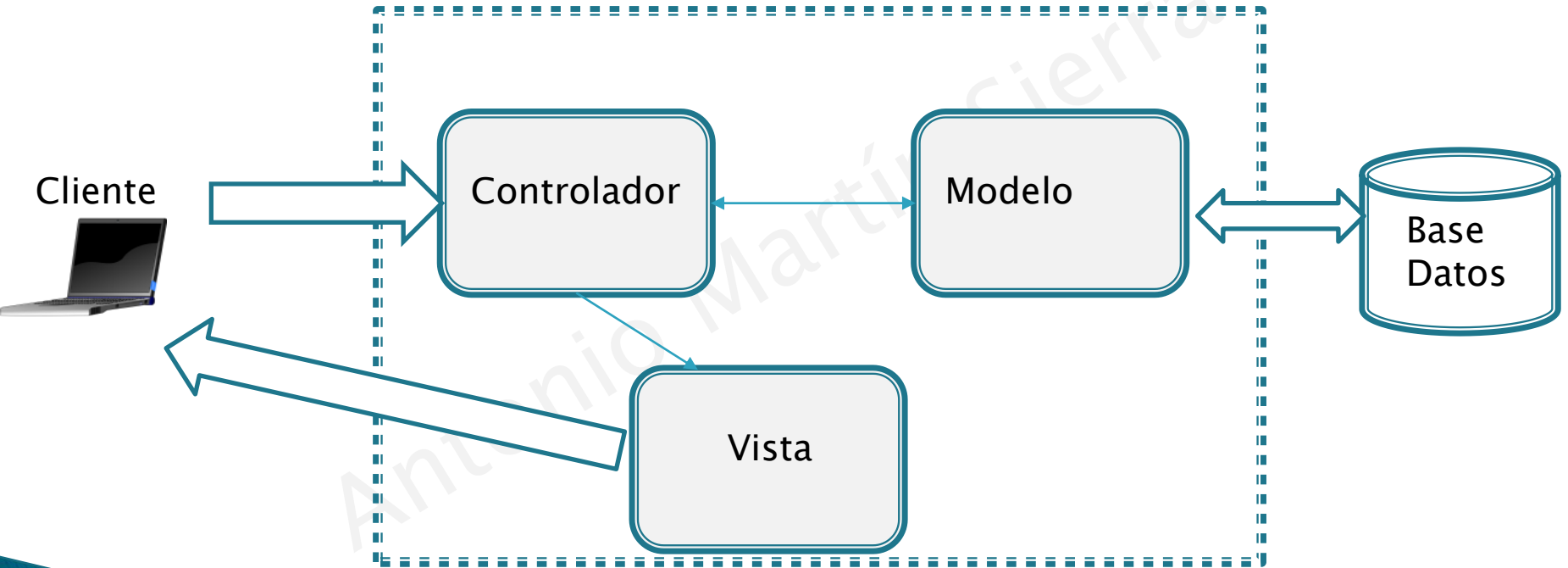
- Rapidez de desarrollo
- Optimización de código
- Menos errores
- Aplicaciones escalables

Patrón Modelo Vista Controlador

- Define la manera en la que se debe abordar el desarrollo de una aplicación
- Propone separar la aplicación en tres grandes bloques:
 - **Modelo.** Encapsula la lógica de negocio de una aplicación, incluyendo el acceso a los datos
 - **Vista.** Encargada de la interfaz gráfica de la aplicación, incluyendo la generación de las respuestas
 - **Controlador.** Actúa de coordinador del resto de los bloques, centralizando la llegada de peticiones del cliente.

Esquema general

Capa intermedia



El modelo

- Encapsula la lógica de negocio de la aplicación
- Exponen la funcionalidad al exterior a través de métodos
- Puede ser implementado en:
 - Clases estándares Java
 - Enterprise Java Beans (EJBs)
 - Beans de Spring

La vista

- Generan dinámicamente las respuestas, a partir de datos proporcionados por el controlador
- Implementadas mediante páginas JSP (Scriptlet Java / JSTL-EL)
- Las vistas estáticas se implementan mediante HTML

Antonio Martín Sierra

El controlador

- Gestiona las peticiones procedentes de la capa cliente, actuando sobre los bloques controlador y vista
- Implementado mediante servlets
- Se divide en dos subbloques: Front controller y Controladores de acción:

