



El Patrón MVC

Modelo – Vista – Controlador



MVC: Modelo – Vista – Controlador

Patrón de arquitectura de software

- Separa la lógica de negocio de la interfaz de usuario
- Objetivos
 - Facilitar la evolución por separado de ambos aspectos
 - Incrementa la reutilización y flexibilidad del software

Historia

- Descrito por primera vez en 1979 para Smalltalk
 - ☐ <http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>
- Utilizado en múltiples frameworks
 - ☐ Java Swing
 - ☐ Java Enterprise Edition (J2EE)
 - ☐ XForms (Formato XML estándar del W3C para la especificación de un modelo de proceso de datos XML e interfaces de usuario como formularios web)
 - ☐ GTK+ (escrito en C, toolkit creado por Gnome para construir aplicaciones gráficas, inicialmente para el sistema X Window)
 - ☐ **ASP.NET MVC Framework** (Microsoft)
 - ☐ Google Web Toolkit (GWT, para crear aplicaciones Ajax con Java)
 - ☐ Apache Struts (framework para aplicaciones web J2EE)
 - ☐ Ruby on Rails (framework para aplicaciones web con Ruby)

Descripción del patrón

Se articula en tres secciones

- **Modelo:**
 - ☐ Único.
 - ☐ Implementa toda la lógica de negocio.
 - ☐ Aísla del acceso a datos al resto de la aplicación.
 - ☐ Se estructura en Entidades (una o varias)
- **Vista**
 - ☐ Normalmente son varias, dependiendo de los casos, suelen existir al menos una por cada Entidad del modelo.
 - ☐ Se encarga de la presentación de los datos.
- **Controlador**
 - ☐ Es el encargado de procesar los eventos que se producen en la vista
 - ☐ Es el encargado de obtener del modelo los datos solicitados a través del modelo.
- Esta separación de aspectos ofrecen una gran flexibilidad a los equipos de desarrollo

Flujo de control

Cómo propone el patrón que se debe trabajar

1. El usuario realiza una acción en la interfaz
2. El controlador trata el evento de entrada
 - ☐ Previamente se ha registrado
3. El controlador notifica al modelo la acción del usuario, lo que puede implicar un cambio del estado del modelo (si no es una mera consulta)
4. Se genera una nueva vista. La vista toma los datos del modelo
 - ☐ El modelo no tiene conocimiento directo de la vista
5. La interfaz de usuario espera otra interacción del usuario, que comenzará otro nuevo ciclo

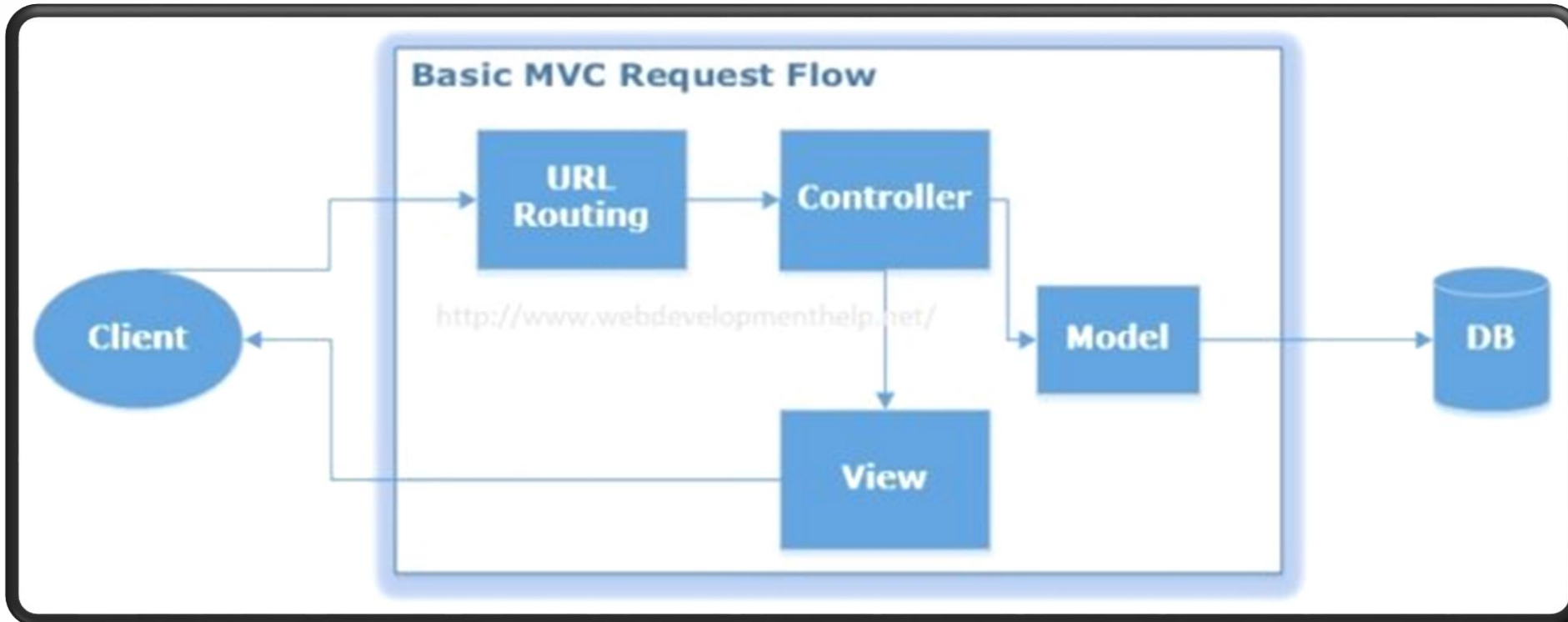
Un Ejemplo de implementación del modelo

MVC en aplicaciones Web

- **Modelo:**
 - ☐ Contiene los datos almacenados en una base de datos, archivos XML o archivos Json.
 - ☐ Contiene las reglas de negocio, que transforma esos datos teniendo en cuenta las acciones de los usuarios.
- **Controlador:**
 - ☐ Código que obtiene los datos procesados por el modelo y genera el contenido HTML
- **Vista:**
 - ☐ La página HTML

Un Ejemplo de implementación del modelo

MVC en aplicaciones Web





Fin de la Presentación

GRACIAS POR SU ATENCIÓN

