

Introducción a los Patrones de Arquitectura

Universitat Jaume I (UJI)
Ingeniería Informática
II27.Ingeniería del Software

Autora: Adriana Venete

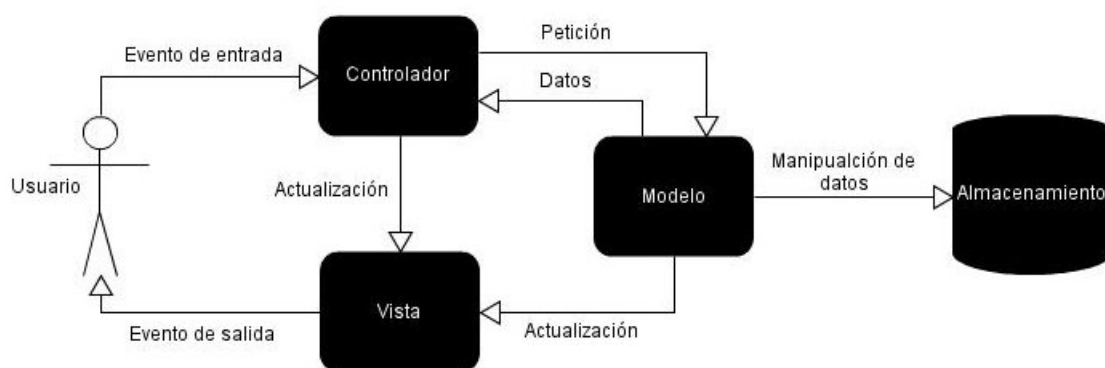
En la arquitectura de software, se utiliza la palabra “arquitectura“, en contraste con el “diseño“, para evocar nociones de codificación, de abstracción, de las normas, de formación (de los arquitectos de software), y de estilo. [2] La Arquitectura del Software es el diseño de más alto nivel de la estructura de un sistema y establece los fundamentos para que analistas, diseñadores, programadores, etc. trabajen en una línea común que permita alcanzar los objetivos del sistema de información, cubriendo todas las necesidades.

Los patrones arquitectónicos, o patrones de arquitectura, son patrones de software que ofrecen soluciones a problemas de arquitectura de software en ingeniería de software. [1] Especifican un conjunto predefinido de subsistemas con sus responsabilidades y una serie de recomendaciones para organizar los distintos componentes.

Resuelven problemas arquitectónicos, adaptabilidad a requerimientos cambiantes, performance, modularidad, acoplamiento, etc. La solución que plantea es la creación de patrones de llamadas entre objetos (similar a los patrones de diseño), decisiones y criterios arquitectónicos, empaquetado de funcionalidad. Este tipo de patrones se utilizan en la fase de desarrollo, en el diseño inicial.

Los beneficios de la utilización de dichos patrones van desde la imposición de decisiones tempranas en el desarrollo hasta la reutilización.

Uno de los patrones de diseño más conocidos y utilizado es el **Modelo-Vista-Controlador** (MVC). Dicho patrón separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.



En la imagen de arriba podemos ver un diagrama de cómo se relacionan los diferentes compo-

nentes entre ellos. Los componentes desempeñan las siguientes tareas:

- *Modelo*: Encapsula los datos y las funcionalidades. El modelo es independiente de cualquier representación de salida y/o comportamiento de entrada.
- *Vista*: Muestra la información al usuario. Pueden existir múltiples vistas del modelo, cada una teniendo asociado un componente controlador.
- *Controlador*: Recibe las entradas, normalmente como eventos. Los eventos son traducidos a solicitudes de servicio, bien para el modelo, bien para la vista.

Bibliografía

- 1 Patrones de arquitectura, Wikipedia
- 2 Foundations for the Study of Software Architecture, artículo por Dewayne E. Perry y Alexander L. Wolf
- 3 Patrón "Modelo-Vista-Controlador"
- 4 Separación modelo vista controlador