

MVC – Protectora de Animales

Grupo 1: Elena, Alejandro del Fresno, Rubén Onsurbe

Este esquema es muy claro y para mí el que cumple casi todos los criterios teóricos del patrón. Como mejoras, que además con el formato que se sigue son sencillas de incluir, quitaría la parte final de los controladores y lo unificaría en un index o controlador principal. También se podría mirar aplicando un DAO el automatizar el controlador a uno genérico ajustando los métodos a algo independiente de la tabla u objeto que se trabaja.

Grupo 2: Jorge, Stephany, Alejandro García

La distribución del código está muy clara y ordenada, aplican además el patrón DAO. Utilizan un único controlador donde se unifican todas las vistas y operaciones. Mi única crítica es que al realizar una vista tan automática donde se carga toda la página, se penaliza la separación y algunos de los métodos del controlador se mezclan incluyendo partes que para mí deberían ir en la vista. Interesante revisar el constructor con instancia y carga de contenido en base a la tabla.

Grupo 3: Rocío, Javier, Ricardo

Es uno de los esquemas más diferentes que se han entregado, tanto vistas como modelo son funciones y se utilizan header para cargar las páginas. Tiene un controlador principal que carga todo el modelo. Este controlador es una mezcla de html (que para mí debería ir en una vista), funciones y condicionales. En este caso, la crítica es más una opinión personal, donde me resulta más útil cargar sólo lo que utilizas y evitar las redirecciones con paso de variables dinámicas en la URL si se pueden evitar.

Grupo 4: Lucía, Pedro A, Óscar

Controlador único con condicionales que determinan las acción y tabla y ejecutan la función necesaria y cargan la vista de forma dinámica. Buena propuesta para una aplicación donde todos los objetos realizan lo mismo y quizá no es tan necesario separarlos. Al igual que la vista se dinamiza, también se podría haber hecho para la parte de instancia y llamada a métodos (todo se llama igual). En la parte de las vistas incluyen código con echo que suprimiría por html+php

Grupo 5: Adrián, Juan, Naiara

Porponen un controlador por tabla que incluye una serie de condiciones para cada tipo de operación. Existe un controlador genérico que carga el controlador necesario según la operación. Las instancias están incluidas en las vistas y con el esquema seguido sería mejor incluirlas en el controlador. Sigue un diseño similar al explicado en clase pero siguiendo programación funcional.

Grupo 6: Diego, Marcos, Rubén R

Aunque no funciona, el planteamiento de “DynamicController” tiene sentido. Al ser operaciones comunes, puede ser una opción realizar un único controlador que unifica las operaciones y sea útil para todas las tablas. Tiene sentido en operaciones CRUD con DAO que esto suele ser así. Los métodos, aunque tienen intención de automatizar después se completan por if con tablas repetidas. Las vistas con echo no me parecen la mejor opción y evitaría la instancia en la propia vista también.

Grupo 7: Pedro, Andrés, Christian

Se realizan 2 entregas, la principal (5.4) contiene un controladorOrquestador que es el que se encarga de recibir las operaciones y cargar las diferentes vistas. Según la tabla que aplique, lleva a los diferentes controladores. Estos controladores son una clase con métodos y un conjunto de condicionales al final, quizá estos me parece que se pueden ubicar en el orquestador dejando esto limpio. Es un buen esquema para escalar a aplicaciones más grandes. Creo que el controlador podría reducirse de try-catch que se gestionan ya desde el modelo. La parte de montar las vistas, tiene una estructura buena para el concepto de plantilla pero en mi opinión es mejor no completar el html principal mediante variables.

Muy interesante revisar la versión 4.0. Se incluyen otros conceptos como es el uso de header en el controlador (muy bien integrado y unido). Cuidado con estas rutas y dejar variables abiertas por GET por seguridad. También incluye una vista montada mediante de CURL (veremos esto en el bloque de servicios).

Grupo 8: Nerea, Samuel, Álvaro

Incluye un controlador para cada una de las clases. Con el tipo de controlador realizado tiene más sentido unificar en uno solo ya que tienen una estructura similar y es bastante fácil sacar similitudes. Se incluyen try-catch en la parte de la vista que no son necesarios.