

## ПРОБЛЕМА MVC-КОНТРОЛЛЕРОВ

«MVC-контроллер должен быть тонким».

1. Источником проблемы является плохое понятие архитектурного паттерна MVC. Из-за путанности в триаде MVC, в том, какой компонент за что и где отвечает, возникает главная и довольно долго обсуждаемая проблема – слишком «толстые» контроллеры (fat controllers).

Немного поговорим про паттерн MVC и разберём неточности определения его понятий: Model – не схема БД, не сущность и не ORM, а модель предметной области; View – не анимичный шаблон, а активный инструмент представления; Controller – не место для бизнес-логики и подготовки данных для отображения.

Правильное определение каждого из компонентов триады позволяет создать правильную архитектуру приложения.

2. Недопонимание базовой триады MVC

2.1. Модель внутри Контроллера. Хорошим тоном является вынос бизнес-логики в так называемые Сервисы (Services).

2.2. Представление внутри Контроллера. Следовательно, хорошим тоном будет передавать контроллером только необходимо-минимальные данные для представления.

Резюмируем базовое понятие MVC. Разобрав каждую роль можно выделить следующие две группы – организатор (Контроллер) и исполнители (Модель и Представление). Разобравшись в том, что и какой компонент должен делать, обязанности контроллера сводятся к следующим: выполнить валидацию данных; уведомить модель, или получить от неё ответ; предоставить действительно важные данные представлению.

3. Отвечаем на главный вопрос: «Почему контроллеры должны быть тонкими?» Ответ очень прост – тонкие контроллеры довольно легки в читаемости, тестируемости, простоте поддержки и переиспользовании кода, что в свою очередь следует многим принципам программирования: таким как DRY (Don't Repeat Yourself).

Главной ответственностью у контроллеров должна быть только одна – делегирование работы Модели и Представления.