РАЗДЕЛ III. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Тарасов А.А., Удодов В.Н.

Динамический mapping URL при реализации MVC паттерна для web разработки

ФГБОУ ВО «Хакасский государственный университет им. Н.Ф. КАТАНОВА»

(ХГУ ИМ.Н.Ф. КАТАНОВА

(Россия, Абакан)

doi: 10.18411/sr-10-08-2019-03

idsp: sciencerrussia-10-08-2019-03

Аннотация

Разработан новый алгоритм, организующий динамический mapping url на функцию, которая может быть реализована штатными средствами языка, для ускорения разработки web-приложений при использовании web-framework. Разработан также controller для модели MVC, который полностью инкапсулирует mapping URL MVC, отделяет view от models, устанавливая между ними протокол взаимодействия.

Ключевые слова: web-framework, framework, mvc паттерн, web-разработка

В последние годы значительно выросло число web-разработок и число используемых frameworks. В результате возрастают требования, предъявляемые к качеству и скорости разработки программного продукта с использованием framework со стороны разработчиков. Это определяет необходимость поиска новых эффективных способов для ускорения разработки web-приложений и повышения их качества. Эта проблема неоднократно упоминалась на конференциях, посвящённых данному виду программирования [1-2]. Все современные web-frameworks реализованы по типу mapping url на функцию, которая обрабатывает данный url [1-2]. Однако с нашей точки зрения, на сегодняшний день упускается такая возможность как динамический mapping url на функцию, которая может быть реализована штатными средствами языка.

При реализации данного подхода будет использован канонический паттерн проектирования MVC (Model-View-Controller «Модель-Представление-Контроллер», «Модель-Вид-Контроллер») [3-4], что позволяет легко структурировать и реализовать web-framework. Основная идея, предоставляемая паттерном – это разделение логики и интерфейса. В результате, программный продукт легко масштабируется, поддерживается и тестируется, уменьшаются затраты на его сопровождение и программный код структурируется намного проще.

Для реализации подхода предлагаем алгоритм, блок-схема которого изображена на рисунке 1.

Рис. 1. Блок-схема алгоритма

Шаг 0: Парсер URL.

Этот шаг использует обыкновенный парсер строки. На этом этапе мы собираем имена классов, которые соответствуют шаблону класса, сканируя путь к классу. В наших целях ожидается, что шаблон класса будет соответствовать именам контроллера. Имя шаблона класса дополняется списком имен корневых пакетов для поиска классов.

Шаг 1: Создание classname - instance

На этом шаге мы извлекаем name для каждого имени класса, которое мы собрали на шаге 0, и создаем instance с name в качестве ключа и полностью определенным именем класса в качестве значения.

Шаг 2: Создание instance - method

На этом шаге мы вызываем method для каждого имени класса, которое мы собрали на шаге 1, который совпадает с методом HTTP (англ. HyperText Transfer Protocol — «протокол передачи гипертекста») запроса, и создаем request в качестве ответа и отдаем пользователю.

Таким образом, в результате реализации алгоритма спроектирован framework для web-разработки на основе паттерна проектирования MVC.

Разработан MVC-framework, позволяющий ускорить и облегчить web-разработку. Также в результате получен controller для модели MVC, который полностью инкапсулирует mapping URL MVC, отделяет view от models, устанавливая между ними протокол взаимодействия.

1. Laravel: <https://laravel.com/> (дата обращения: 01.06.2019)
2. Django: <https://www.djangoproject.com/> (дата обращения 01.06.2019)
3. MVC: <https://ru.wikipedia.org/wiki/Model-View-Controller> (дата обращения 01.06.2019)
4. Э. Гамма, Р. Хелм, Р. Джонсон, Д. Влиссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования / [пер. с англ.: А. Слинкин науч. ред.: Н. Шалаев]. — Санкт-Петербург [и др.]: Питер, 2014. — 366 с.