

**Д.В. Нефедов** – магистрант кафедры моделирования вычислительных и электронных систем  
**В.С. Павлов** (канд. техн. наук, доц.) – научный руководитель

## ШАБЛОННО-ОРИЕНТИРОВАННЫЙ ПОДХОД В ПРОЕКТИРОВАНИИ ИНТЕРНЕТ-ПРИЛОЖЕНИЙ

При разработке веб-приложений ставится широкий спектр задач. Это эффективная работа с реляционными базами данных, хранение и обработка данных в формате XML, построение гибких систем отображения информации. Такое множество задач делает старые методы разработки приложений крайне неэффективными. Это приводит к необходимости наличия специальных инструментов, которые используются в решении часто возникающих проблем и задач. Такими инструментами становятся шаблонно-ориентированные системы.

Вопросы архитектуры очень сложны по своей сути, но, несмотря на большую контекстную зависимость архитектуры от типа приложения, существуют хорошо изученные и проверенные варианты решения архитектурных проблем. Эти варианты решения носят название шаблонно-ориентированные системы. Шаблонно-ориентированное проектирование может быть применено во всех приложениях. Выявление оптимальных реализаций шаблонов составляет неотъемлемую часть работы над интернет-приложением.

Общая цель разработчиков — разработка и развертывание гибких и легко сопровождаемых приложений. Важным моментом в достижении этой цели является отделение бизнес-логики от логики представления. Разработчики могут использовать шаблонно-ориентированные системы, чтобы сохранить такое разделение. Одной из трудностей при выполнении такого разделения является отсутствие четко определенных критериев для оценки того, что оно действительно выполнено, и насколько хорошо это сделано. Есть стандартные критерии, которые используются в области программного обеспечения. В их число входит наследование (на основе принципов объектно-ориентированного программирования), шаблонизация и порождающее программирование, (в соответствии с принципами MVC разделения).

Одним из самых известных и часто применяемых методов шаблонно-ориентированного проектирования является шаблон MVC, и построенные на основе данного шаблона фреймворк-системы. Для исследования были выбраны три PHP фреймворк-системы, построенные на MVC: CakePHP, CodeIgniter, Yii. Исследование проводилось по следующим параметрам: простота начальной настройки, документация, тестирование на производительность, безопасность.

Далее представлены два теста производительности. Оба теста оценивают минимальные накладные расходы каждого фреймворка. Задача фреймворка – вывести простейшую программу. Основные различия заключаются в том, что в первом случае использовалась конфигурация каждого фреймворка по умолчанию. Был использован паттерн MVC: контроллер устанавливает значение переменной в «Hello World», затем она передается представлению для вывода на основной HTML-странице. Каждый фреймворк содержал следующий код в представлении:

```
<html> <head><title>Greeting</title>
</head> <body>
<h1>Greetings from AAA</h1>
<p>The greetings message is: "<?= $message ?>"</p>
</body></html>
```

Переменная *\$message* предоставляется контроллером, а ее значение — «Hello World». Количество запросов в секунду выяснялось при помощи утилиты ApacheBench с такими параметрами: «ab -t 30 -c 10 URL».

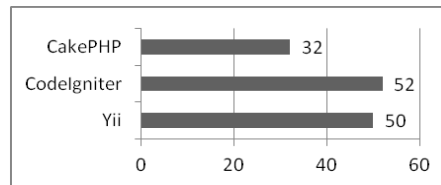


Рис. 1. Результаты теста №1 без APC

#### Тест №2.

APC – акселератор PHP, программа, ускоряющая исполнение сценариев PHP интерпретатором, путём кэширования их байткода.

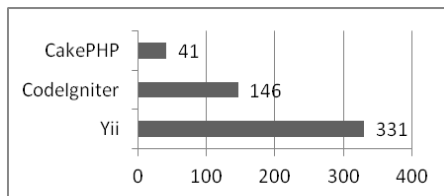


Рис. 2. Результаты теста №2 с APC

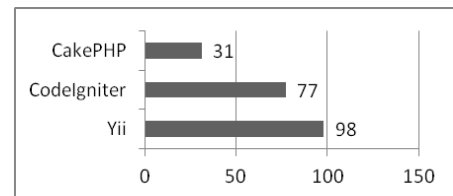


Рис. 3. Результаты теста №2 без APC

Результаты выражены в запросах в секунду (Requests Per Second, RPS), следовательно, большие значения – лучше. В результате тестирования производительности было определено, что Yii и CodeIgniter в плане скорости обработки запросов значительно опережают CakePHP.

При исследовании безопасности системы проводился анализ по следующим показателям: контроль доступа, атаки по словарю, внедрение межсайтовых скриптов, внедрение SQL-кода, подделка межсайтовых запросов, кража куки.

В результате исследования безопасности, было установлено, что фреймворк Yii имеет значительное преимущество в плане безопасности, с лучшей системой контроля доступа и защитой от XSS, внедрения SQL-кода, подделки межсайтовых запросов и краж куки. CakePHP и CodeIgniter уступают в этом плане. Так как CodeIgniter более гибкий, в нем разработчику будет легче добавить требуемые средства защиты.

С фреймворком Yii разработчик может писать свое приложение, не обращая внимания на RBAC (контроль доступа, основанный на ролях), и использовать его только в тех случаях, когда требуется решить специфическую задачу.

На сегодняшний день скорость при создании интернет-приложения важна не меньше качества работы. Оперативность появления на рынке и реакция на изменения имеют высокую стоимость в данной сфере. Соответственно легче использовать готовые шаблонно-ориентированные системы, чем писать каждый раз одни функциональные решения для интернет-приложений.

#### Библиографический список

1. Шлосснейгл, Д. Профессиональное программирование на PHP / Шлосснейгл Д.. М.: Вильямс, 2006. 320 с.
2. Гутманс, Э. PHP5 программирование / Гутманс Э.. М.: Символ-плюс, 2011. 490 с.