

## Особенности использования современных PHP-фреймворков для реализации корпоративных веб-порталов

*Козич Полина Александровна*

*Приамурский государственный университет имени Шолом-Алейхема*

*Студент*

### Аннотация

Язык программирования PHP известен легкостью, с которой он позволяет писать скрипты, исполняемые на веб-сервере. Из-за низкого порога вхождения, обусловленного простотой языка и возможностью писать в процедурном стиле существует большое количество проектов, программный код которых далек от высокого качества. С выходом версии PHP 7, в языке появилась возможность использовать принципы объектно-ориентированного программирования. Так же PHP было создано большое количество фреймворков, представляющие собой программную платформу, которая определяет структуру программной системы, облегчающая разработку и объединение разных компонентов большого программного проекта. Обзор данных фреймворков и их особенности и будут рассмотрены в данной статье.

**Ключевые слова:** Язык программирования, PHP, фреймворк, MVC, ORM, паттерны проектирования, объектно-ориентированное программирование (ООП).

### Features of the use of modern PHP-frameworks for the implementation of corporate web portals

*Kozich Polina Alexandrovna*

*Sholom-Aleichem Priamursky State University*

*Student*

### Abstract

The PHP programming language is known for the ease with which it allows you to write scripts that are executed on a web server. Due to the low entry threshold, due to the simplicity of the language and the ability to write in a procedural style, there are a large number of projects, the code of which is far from high quality. With the release of PHP 7, the language has the ability to use the principles of object-oriented programming. PHP also created a large number of frameworks, which are a software platform that defines the structure of the software system, which facilitates the development and integration of different components of a large software project. The review of these frameworks and their features will be discussed in this article.

**Keyword:** PHP, framework, MVC, ORM, design patterns, object-oriented programming (OOP).

Один из популярнейших языков программирования на сегодняшний день является PHP. По рейтингу TIOBE на май 2018 года PHP занимает 7 место в мире [1]. Он прошел свой путь развития от языка для написания небольших скриптов до возможности писать хорошие с архитектурной точки зрения приложения, которые легко масштабируются и дорабатываются. В этих достижениях не последнюю роль играют различные фреймворки, предлагающие изначально архитектуру приложения, шаблоны и набор инструментов, позволяющий ускорить разработку.

Цель данного исследования: обозначить особенности использования современных PHP-фреймворков для реализации корпоративных веб-порталов.

PHP-фреймворки в своих работах рассматривали А.К. Романов, С.С. Гришунов в своей работе сделали обзор PHP-фреймворков [2], Н.В. Шабров, А.А. Забуга в своей работе описали выбор критериев для сравнения mvc-фреймворков [3], О.В. Николаева в своей работе рассматривает возможности yii framework [4], Б.В. Палаш в своей работе производит анализ современных веб-фреймворков [5].

Для примера рассмотрим пару популярных фреймворков и проанализируем, чем они могут помочь при разработке. Для данной цели были выбраны Laravel и Yii.

Оба данных фреймворка используют архитектуру MVC – это архитектурный паттерн (шаблон), цель которого разделить приложение на три слоя: Model, View, Controller, отсюда и название MVC. Сделано это для разделения ответственности и осуществления возможности внесения изменения в один слой, не влияя на другой.

Взаимодействие слоев можно увидеть на рисунке 1.

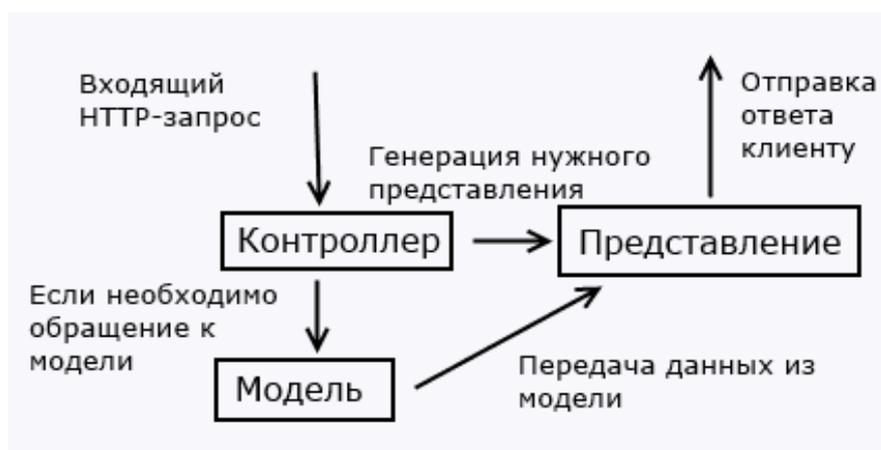


Рисунок 1 – Взаимодействие слоев в архитектуре MVC

Назначение слоев:

- Model (Модель) – данный слой отвечает за структуру классов, модулей, которые описывают данные, используемые в приложении. Часто модель связана с базой данных, используемой в системе.

- View (Представление) – визуальная часть, пользовательских интерфейс.
- Controller (Контроллер) – обеспечивает связь между пользователем и системой, получает и обрабатывает входные данные.

Такое разделение на слои позволяет легче тестировать ту или иную часть приложения, например, отдельно представления, отдельно модель. Так же в силу независимости слоев, позволяет их заменить на альтернативный. Например, есть возможность заменить интерфейс (слой представления), оставив прежним, модель и контроллеры.

С накоплением опыта, разработчик понимает, что одни и те же действия повторяются из проекта в проект. Для представления изначально пишется одна и та же разметка, в контроллеры как по умолчанию добавляются одни и те же методы и т.д. Одним из преимуществ фреймворков является различные инструменты или генераторы, которые одной командой позволяют сгенерировать тот или иной код или осуществить действие с файлами программного проекта. Примерами могут послужить CRUD генератор из Yii. Для Laravel так же есть подобный генератор. Или генератор контроллеров и представлений, которой также есть и в том и в другом фреймворке.

Другой отличительной особенностью современной разработке на PHP и не только, является работа с базами данных с помощью ORM (Object-Relational Mapping) – объектно-реляционное отображение, позволяющее наладить взаимодействие между базой данных и ООП. ORM позволяет работать с элементами базы данных как с объектами, упрощая работу. К примеру, программист может работать с SQL-базой и не написать ни одного SQL-запроса, благодаря использованию ORM. Другими словами, ORM – это посредник между объектной моделью и базой данных (рисунок 2).

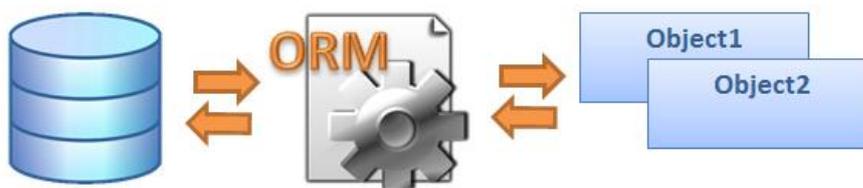


Рисунок 2 – ORM как посредник между базой данных и объектами

В рассматриваемых фреймворках примером ORM может служить Eloquent ORM и Yii DAO для Laravel и Yii соответственно.

В обоих фреймворках поддерживается функционал миграций. Данный функционал позволяет отслеживать изменения в базе данных, применять миграции или откатывать уже примененные. Необходимо это, если в процессе разработки есть несколько версий исходного кода и несколько версий состояния БД и с помощью данного функционал можно поддерживать соответствие кода и базы данных.

Несмотря на описанные инструменты двух фреймворках, сами по себе они отличаются друг от друга. Чуть ниже приведена таблица сравнения этих двух фреймворков (таблица 1). Ну и стоит отметить, что фреймворки – это не решение всех проблем. Если разработчик не привык писать хороший, структурированный код, то он и при использовании данных инструментов, способен создать трудно дорабатываемый продукт.

Таблица 1. Сравнение фреймворков

Параметр сравнения	Yii	Laravel
<b>Требования</b>	PHP 5.4 и выше	PHP 5.4 и выше MCrypt PHP extension PHP JSON extension
<b>MVC</b>	Да	Да
<b>Расширения</b>	Поддерживаются	Поддерживаются
<b>ORM (объектно-реляционное отображение)</b>	Data Access Objects (DAO) Active Record Pattern (ActiveRecord) Doctrine 2 через плагины	DAO ActiveRecord Pattern (Eloquent ORM) Doctrine 2 через плагины
<b>Тестирование</b>	PHPUnit Codeception	PHPUnit. Компоненты Symfony для тестирования (HttpKernel, DomCrawler и BrowserKit)
<b>Отладка</b>	Подробная консоль отладки из коробки Логирование разных уровней Логирование запросов к БД	Логирование Консоль отладки из коробки со стеком вызовов и расширением laravel-debugbar
<b>Миграция</b>	Инструмент для миграций (класс Migration)	Миграции (класс Migration) Инструмент наполнения данными (класс Seeder)
<b>Безопасность</b>	Богатый правилами Access Control Filter (ACF) Role Based Access Control (RBAC) на основе NIST RBAC model Расширения для авторизации по OpenID, Oauth или Oauth2 и пр.	RBAC ACL плагины Фильтры (функции) контроля доступа

<b>Шаблонизаторы, работа с шаблонами</b>	Официальные Twig и Smarty пакеты Asset'ы для подключения JS, CSS и пр. Плагины	На базе PHP Blade Asset'ы для подключения JS, CSS и пр. Плагины Все шаблоны по умолчанию кэшируются
<b>Кэширование</b>	APC Database File Memcached Redis WinCache Xcache Zend Data Cache	APC Database File Memcached (в 5 версии) Redis (в 5 версии)
<b>REST API</b>	Поддержка JSON, JSONP и XML Роутинг в соответствии с REST-запросами Поддержка HATEOAS Кэширование запросов Ограничение скорости и пр.	Настройка роутинга REST-запросов Поддержка JSON, JSONP и пр.
<b>Локализация приложения</b>	Присутствует	Присутствует
<b>Валидация форм</b>	Присутствует	Присутствует

В качестве заключения можно сделать вывод: обозначены некоторые особенности использования современных PHP-фреймворков. Данное исследование будет полезно при изучении дисциплин web-программирования.

### Библиографический список

1. Рейтинг языков программирования TIOBE URL: <https://www.tiobe.com/tiobe-index/> (дата обращения: 27.01.2019).
2. Романов А.К., Гришунов С.С. Обзор php-фреймворков // Электронный журнал: наука, техника и образование. 2017. № СВ1 (11). С. 107-111.
3. Шабров Н.В., Забуга А.А. Выбор критериев для сравнения mvc-фреймворков // В сборнике: Наука. Технологии. Инновации. Материалы всероссийской научной конференции молодых ученых в 10 частях. 2013. С. 56-58.
4. Николаева О.В. Возможности Yii framework // В сборнике: Информационно-телекоммуникационные системы и технологии. Материалы Всероссийской научно-практической

- конференции. 2014. С. 273-274.
5. Палаш Б.В. Анализ современных веб-фреймворков // Постулат. 2018. № 6 (32). С. 109.
  6. Сравнение PHP фреймворков: Yii и Laravel URL: <https://magora-systems.ru/yii-vs-laravel/> (дата обращения: 27.01.2019).