

ОБЗОР MVC WEB-ФРЕЙМВОРКОВ

Ногербек Н.Д.

Научный руководитель: Мальчуков А.Н.

Томский политехнический университет

nogerbeknurzhan@gmail.com

Введение

Современным способом разработки интернет ресурсов является применение MVC фреймворков. Фреймворк является архитектурой разрабатываемого ресурса, который гарантирует стандартную структуру программ и их поведение по умолчанию. Данные в MVC отделены от реализации функционала, а тот, в свою очередь, – от представления (внешнего вида) [1].

В общем случае веб-фреймворк состоит из следующих компонентов [2]:

- Шаблонизатор. Отвечает за независимость вёрстки от программного кода.
- Роутер. Распознаёт URL, по которому произошло обращение к серверу.
- Модуль доступа к базе данных.
- Модуль кэширования. Ускоряет загрузку страниц.
- Модуль безопасности. Аутентификация и авторизация пользователей.
- Файлы конфигурации.

Фреймворки также могут управлять сессиями, вести логи, упрощать использование технологии Ajax и многое другое.

Таким образом, делая сайт на базе фреймворка, мы получаем доступ к большому количеству готовых функций. Все базовые функции – от доступа к базе данных до отдачи готовой страницы браузеру – уже написаны создателями фреймворка.

Среди популярных современных веб-фреймворков можно отметить следующие: Zend Framework, Yii, Symfony2, Laravel, CakePHP (PHP), Django (Python), Ruby on Rails (Ruby) [2].

Схема проектирования MVC

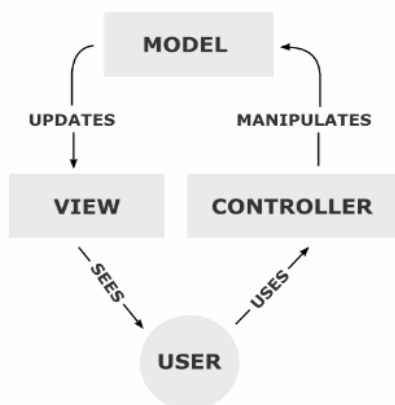


Рисунок. Шаблон проектирования

Model View Controller (MVC, «модель-представление-контроллер», «модель-вид-контроллер») – схема использования нескольких шаблонов проектирования, с помощью которых модель приложения, пользовательский интерфейс и взаимодействие с пользователем разделены на три отдельных компонента таким образом, чтобы модификация одного из компонентов оказывала минимальное воздействие на остальные (см. рис.). Данная схема проектирования часто используется для построения архитектурного каркаса, когда переходят от теории к реализации в конкретной предметной области [1].

Наиболее известные MVC фреймворк системы

Анализ возможностей некоторых распространённых веб-фреймворков приведен в таблице [3].

Таблица. Сравнения некоторых фреймворков.

Критерий	ASP.NET MVC	Zend	Cake	Symfony	ROR	Struts
Язык	ASP.NET	PHP	PHP	PHP	Ruby	JAVA
Шаблон проектирования MVC	ДА	ДА	ДА	ДА	ДА	ДА
Применение ORM	ORM-независима	ДА	ДА	ДА	ДА	ДА
Механизмы интернационализации и локализации	----	ДА	ДА	ДА	ДА	ДА
Применение шаблонов при создании пользовательских интерфейсов	ДА	ДА	ДА	----	ДА	ДА (Jakarta Tiles)
Создание и проверка форм	ДА	ДА	ДА	ДА	ДА	ДА (Jakarta Validator)
Управление доступом на основе ролей	ДА	ДА	ДА	-----	ДА	НЕТ
Использование AJAX	ДА	ДА	ДА	-----	ДА	ДА
ЧПУ (Friendly URL)	ДА	ДА	ДА	ДА	ДА	НЕТ
Модульное тестирование (unit-testing)	ДА	ДА	ДА	----	ДА	ДА
Система кэширования	ДА	ДА	ДА	ДА	ДА	НЕТ

В таблице сравнительного анализа не приведен фреймворк Django на языке Python. Фреймворк Django – open source фреймворк, использующий шаблон проектирования MVC. В целом очень похож на Ruby On Rails и, фактически, является главным конкурентом RoR [4].

Проект поддерживается организацией Django Software Foundation и огромным количеством разработчиков со всего мира, работающих над

устранением уязвимостей и добавлением нового функционала в арсенал фреймворка Django.

Сайты в Django строятся из так называемых "приложений", которые необходимо разрабатывать модульно, то есть независимыми друг от друга и подключаемыми. Это одно из существенных архитектурных отличий Django от некоторых других, как на Python, так и на других языках программирования (например, Ruby on Rails). Один из главных принципов фреймворка — Don't repeat yourself [5].

Также, в отличие от других фреймворков, обработчики URL в Django конфигурируются явно при помощи регулярных выражений, а не выводятся автоматически из структуры моделей контроллеров, предоставляя улучшенный контроль над адресами.

Для работы с базой данных Django использует собственный ORM, в котором модель данных описывается классами Python, и по ней генерируется схема базы данных. ORM полностью не зависит от выбора базы данных, поэтому проект легко переносится как с SQLite на Postgres, так и с MySQL на Oracle [6].

Достоинства фреймворка Django

Достоинством использования Python в качестве языка программирования является то, что он достаточно прост с синтаксической точки зрения, что снижает порог вхождения для начала написания первых программ. К тому же, Python имеет обширную библиотеку классов, хорошую документацию и достаточно компактный и интуитивно понятный синтаксис [7].

В фреймворке Django имеется подробная и исчерпывающая документация. Множество примеров, объяснений, и самое главное — открытый исходный код, который написан качественно.

Так же имеется встроенный ORM (Object-relational mapper), который позволяет в большинстве случаев не использовать SQL-синтаксиса в выражениях, что снижает риск появления SQL-injection уязвимости.

Одной из уникальных особенностей Django является автоматически генерируемая страница для администратора. Другие фреймворки с такой функцией нам не известны. Помимо того, что данная функциональность позволяет значительно сократить время на написание нужного интерфейса и страницы администратора, она еще и дает возможность клиентам сразу начать работать с интернет ресурсом еще на начальных этапах его разработки. Фактически, достаточно спроектировать нужные модели, и можно сразу показывать сайт клиенту, и уже интерактивно с

ним обсуждать бизнес-логику, не отвлекаясь на дизайн.

В фреймворке Django имеется поддержка MTV (Model-Template-View). Данный паттерн проектирования очень близок к классическому MVC, и самое главное, что он позволяет отделять реализацию функционала от интерфейса.

Высокая скорость работы фреймворка Django является её лучшей чертой. Несмотря на то, что Python имеет сравнительно малую скорость, в целом Django работает достаточно быстро. Он может выдерживать высокую нагрузку, к тому же имеет встроенные возможности кэширования и распределения нагрузки [6].

Заключение

В данной статье рассмотрены возможности и особенности современных MVC фреймворков, которые при правильном подходе к разработке являются архитектурой веб ресурса. Приведенный сравнительный анализ наиболее популярных веб-фреймворков наглядно показывает их особенности и текущее положение развития. По результатам обзора можно сделать вывод, что наиболее предпочтительным из перечисленных фреймворков является Django.

Список литературы

7. Model View Controller // Wikipedia // URL: <https://ru.wikipedia.org/wiki/Model-View-Controller> (дата обращения: 23.10.2015)
8. Современные веб-фреймворки // Pirsipy // URL: <http://blog.pirsipy.com/sovriemiennyye-vieb-freimvorki/> (дата обращения: 21.10.2015)
9. «Веб-фреймворки и с чем их едят // Сайт для веб-мастера // URL: <http://iwsn.ru/blog/show/veb-freymvorki-i-s-chem-ih-edyat> (дата обращения: 23.10.2015)
10. Rails vs Django Battle of frameworks // Diogo Nunus — Персональный сайт // URL: <http://www.diogonunes.com/blog/rails-vs-django-vs-play-framework/> (дата обращения 20.10.2015)
11. Сравнение веб-фреймворков Django и Ruby on Rails // Все о науке // URL: <http://www.science.kuzstu.ru/wp-content/Events/Conference/RM/2015/RM15/pages/Articles/IITMA/23.pdf> (дата обращения: 21.10.2015)
12. Причины выбора Django для разработки сайта // Habrahabr // URL: <http://www.habrahabr.ru/post/75131/> (дата обращения: 18.10.2015)
13. Choosing a Python Web Framework // Software Micro-Consulting // URL: <https://www.airpair.com/python/posts/django-flash-pyramid> (дата обращения: 19.10.2015)