

УДК 004.5

ОПИСАНИЕ И ПРИМЕНЕНИЕ ТЕХНОЛОГИИ ASP.NET MVC 5 ДЛЯ СОЗДАНИЯ WEB-ПРИЛОЖЕНИЙ

Вишняков В.В., студента группы ИТб-132, курс IV
Научный руководитель – Алексеева Г.А.
Кузбасский государственный технический университет
имени Т.Ф. Горбачева
г. Кемерово

ASP.NET MVC является фреймворком для создания веб-приложений и сайтов.

MVC Framework не предполагает использование обычных web-форм и web-элементов управления. MVC Framework подразумевает использование URL-mapping и архитектуру REST в качестве модели запросов, что оказывает положительное влияние на поисковую оптимизацию web-проектов. Сравнительный анализ классического ASP.NET и ASP.NET MVC приведен в таблице 1. [1]

Таблица 1

Наименование	Классический ASP.NET	MVC
Модель запросов	postback	REST
Модель данных	через код cs-файла страницы (code-behind)	определяется паттерном MVC
Разработка интерфейса	web-controls	html, MVC UI Helpers
Авто-генерация id	присутствует	отсутствует
ViewState	присутствует	отсутствует
URL mapping	отсутствует	присутствует

MVC Framework предложил программистам ASP.NET новый стиль, позволяющий значительно улучшить качество кода со стороны клиентской части. Сгенерированный MVC код страниц не содержит ничего автоматически созданного. Код страницы представляет собой чистый, написанный самим разработчиком HTML. В эпоху прочно укоренившегося понятия web 2.0 абсолютный контроль над страницей со стороны клиента – это безоговорочный залог успеха любого web-проекта.

Концепция паттерна (шаблона) MVC (model - view - controller) предполагает разделение приложения несколько важных компонентов:

1. Контроллер (controller) представляет класс, обеспечивающий связь между пользователем и системой, представлением и хранилищем данных. Он получает введенные пользователем данные и занимается их

обработкой. И в зависимости от результатов обработки отправляет пользователю определенный вывод, например, в виде представления.

2. Представление (view) – есть визуальная часть, html-страница, которую пользователь может наблюдать, зайдя на сайт.

3. Модель (model) представляет класс, описывающий логику используемых данных.

Общая схема взаимодействия этих компонентов представлена на рисунке 1.



Рисунок 1 – Паттерн MVC в ASP.NET

В этой схеме модель является независимым компонентом – любые изменения контроллера или представления не затрагивают модель. Контроллер и представление являются относительно независимыми компонентами, и нередко их можно изменять отдельно друг от друга. [4]

Благодаря этому реализуется концепция разделение ответственности, в связи, с чем легче построить работу над отдельными компонентами. Кроме того, вследствие этого приложение обладает лучшей тестируемостью и отказоустойчивостью. И если нам, допустим, важна визуальная часть или frontend, то мы можем тестировать представление независимо от контроллера. Либо мы можем сосредоточиться на backend`е и тестировать controller.

Конкретные реализации и определения данного паттерна могут различаться, но в силу своей гибкости и простоты он стал очень популярным в последнее время, особенно в сфере web-разработки.

Свою реализацию паттерна представляет платформа ASP.NET MVC. 2013 год ознаменовался выходом новой версии ASP.NET MVC - MVC 5, а также релизом Visual Studio 2013, которая предоставляет инструментарий для работы с MVC5.

Хотя во многих аспектах MVC 5 не слишком сильно будет отличаться от MVC 4, многое из одной версии можно применить к другой, но в то же время есть и серьезные отличия:

В MVC 5 изменилась концепция аутентификации и авторизации. Вместо SimpleMembershipProvider была внедрена система ASP.NET Identity, которая использует компоненты OWIN и Katana.

Для создания адаптивного и расширяемого интерфейса в MVC 5 используется css-фреймворк Bootstrap упрощающий верстку.

Добавлены фильтры аутентификации, а также появилась функциональность переопределения фильтров.

В MVC 5 также добавлены различные атрибуты маршрутизации.

Это наиболее важные нововведения в MVC 5. Кроме того, есть еще ряд менее значимых, например, использование по умолчанию Entity Framework 6, некоторые изменения при создании проекта (концепция One ASP.NET), дополнительные компоненты и т.д.

А это означает, что все знания, имеющиеся по MVC 4 можно успешно применять при использовании MVC 5, не забывая о нововведениях.

Для написания веб-приложения с использованием MVC 5 достаточно базовых знаний C#, CSS, HTML, JavaScript и понимания работы Entity Framework и Bootstrap.

Примеров сайтов, написанных с использованием данной технологии, представлены на рисунках 2 и 3 [2, 3].

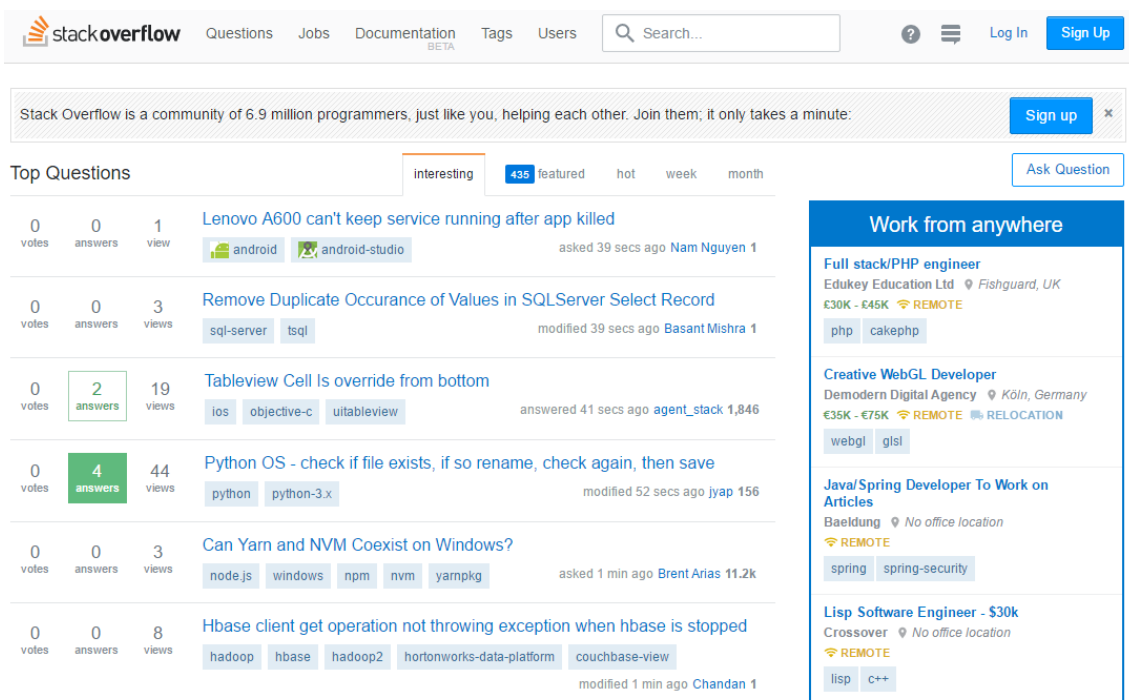


Рисунок 2 – Сайт вопросов и ответов для программистов
<http://stackoverflow.com/>

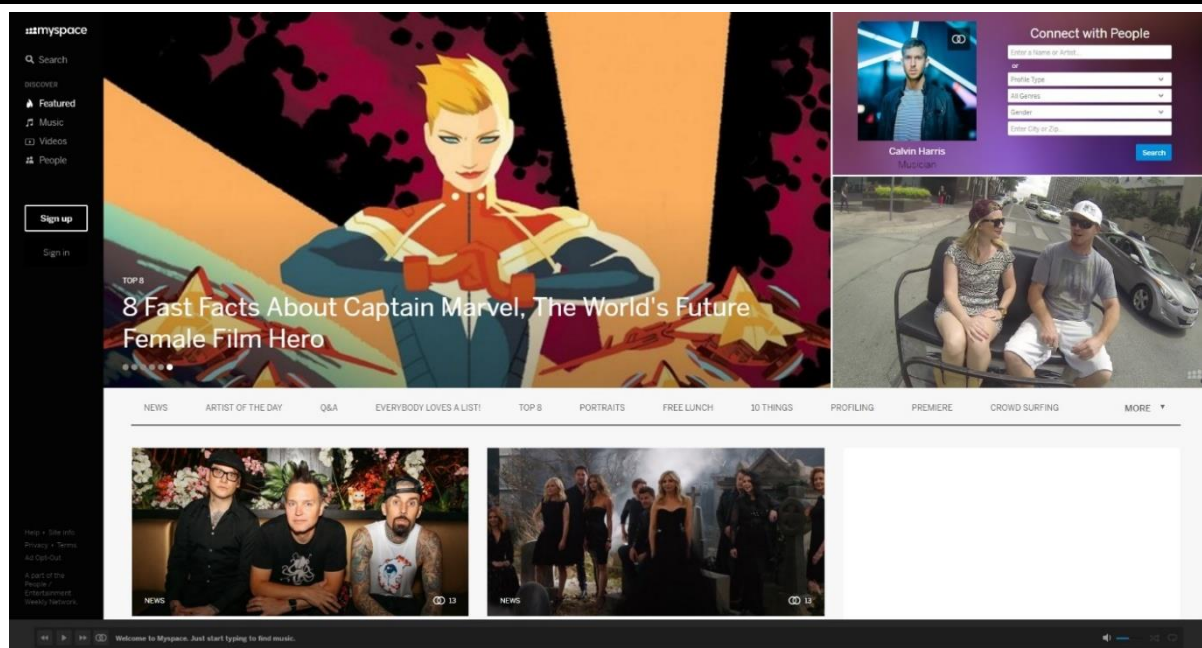


Рисунок 3 – Featured Content on Myspace <https://myspace.com/>

Список литературы:

1. Материалы сайта «Metanit» [Электронный ресурс]. – Режим доступа: <https://metanit.com/>
2. Материалы сайта «StackOverflow» [Электронный ресурс]. – Режим доступа: <http://stackoverflow.com/>
3. Материалы сайта «MySpace» [Электронный ресурс]. – Режим доступа: <https://myspace.com/>
4. Адам Фримен, ASP.NET MVC 5 с примерами на С# для профессионалов 5-е издание – 2013 – № 1 (6). – С. 218–224.