

УДК 004.43

ПРЕИМУЩЕСТВА И НЕДОСТАТКИ ФРЕЙМВОРКОВ ДЛЯ РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЙ В ЦЕЛЯХ ЦИФРОВИЗАЦИИ ЭКОНОМИКИ

Чулин К.В., Белов Ю.С., Гришунов С.С.

*Московский государственный технический университет имени Н.Э. Баумана, филиал,
Калуга, e-mail: fn1-kf@mail.ru*

Веб-приложения непрерывно завоевывают всё большую популярность и в значительной мере вытесняют классические. Их высокая гибкость, широкий выбор языков программирования и независимость от операционной системы клиента легко объясняют такой успех. На сегодняшний день актуальной является разработка веб-приложений, связанных с цифровой трансформацией предприятия. В результате цифровизации повышаются производительность каждого сотрудника и уровень удовлетворенности клиентов, а компания приобретает репутацию прогрессивной и современной организации. Технологии цифровизации позволяют организовать максимально персонализированное взаимодействие, которое предпочитает большинство клиентов. Стандарты веб-разработки постоянно растут вместе со сложностью современных технологий. Есть два пути разработки веб-приложений: с использованием фреймворков и без них. Некоторые веб-разработчики считают, что веб-платформа настолько развита, что разработчику не нужны дополнительные API-интерфейсы для упрощения создания приложений. Другие же говорят о том, что фреймворки значительно упрощают и ускоряют разработку. Целью исследования является проведение анализа фреймворков Django, Spring MVC, Laravel и Ruby on Rails для разработки веб-приложения. Рассмотрены их преимущества и недостатки. Описаны особенности фреймворков. Показаны различия исследуемых фреймворков. Осу- ществлен выбор фреймворка в зависимости от требований к разрабатываемому проекту.

Ключевые слова: веб-приложения, фреймворк, цифровизация, цифровая трансформация, Django, Laravel, Spring, Ruby on Rails

ADVANTAGES AND DISADVANTAGES OF FRAMEWORKS FOR WEB APPLICATION DEVELOPMENT IN ORDER TO DIGITALIZE THE ECONOMY

Chulin K.V., Belov Yu.S., Grishunov S.S.

Bauman Moscow State Technical University, branch, Kaluga e-mail: fn1-kf@mail.ru

Web applications are constantly gaining more and more popularity and are largely replacing classic ones. Their high flexibility, wide choice of programming languages and independence from the client's operating system easily explain this success. Today, the development of web applications related to the digital transformation of an enterprise is relevant. As a result of digitalization, the productivity of each employee and the level of customer satisfaction increase, and the company gains a reputation for being a progressive and modern organization. Digitalization technologies allow you to organize the most personalized interaction that most customers prefer. Web development standards are constantly growing along with the complexity of modern technologies. There are two ways to develop web applications: with and without frameworks. Some web developers believe that the web platform is so advanced that the developer does not need additional APIs to simplify the creation of applications. Others say that frameworks make development much easier and faster. The aim of the study is to analyze the Django, Spring MVC, Laravel and Ruby on Rails frameworks for developing a web application. Their advantages and disadvantages are considered. Features of frameworks are described. The differences of the studied frameworks are shown. The choice of the framework has been made depending on the requirements for the project being developed.

Keywords: web applications, framework, digitalization, digital transformation, Django, Laravel, Spring, Ruby on Rails

Веб-приложения беспрерывно завоевывают всё большую популярность. Их высокая гибкость, широкий выбор языков программирования и независимость от операционной системы клиента легко объясняют такой успех. Сама идея клиент-серверных приложений, будучи уже весьма зрелой, захватила мир программного обеспечения. Стандарты веб-разработки постоянно растут вместе со сложностью современных технологий. Сегодня актуальной является разработка веб-приложений, связанных с цифровой трансформацией предприятия. В результате цифровизации повышаются производительность каждого сотрудника и уровень удовлетворенности клиентов,

и компания приобретает репутацию прогрессивной и современной организации.

Есть два пути разработки веб-приложений: с использованием фреймворков и без них. Некоторые веб-разработчики поддерживают идею отказа от фреймворков для разработки веб-приложений. В частности, речь идет о том, что веб-платформа настолько развита, что разработчику не нужны дополнительные API-интерфейсы для упрощения создания приложений. Разработка веб-приложений без использования фреймворков означает согласие не извлекать выгоду из работы групп разработчиков фреймворков, которые часто тесно сотрудничают с создателями браузеров. Другая часть раз-

работчиков говорит о том, что фреймворки значительно упрощают и ускоряют разработку.

Цель исследования: провести анализ фреймворков Django, Spring MVC, Laravel и Ruby on Rails для разработки веб-приложения. Рассмотреть их недостатки и преимущества. Описать особенности фреймворков. Показать различия исследуемых фреймворков. Произвести выбор фреймворка в зависимости от требований к разрабатываемому проекту.

Язык программирования Python считается одним из самых популярных и востребованных для создания веб-приложений. Он способен решать различные задачи, в том числе реализовывать веб-приложения, которые используют миллионы пользователей. «Дзен Python» – это собственная философия языка, на основе которой работает Python.

Фреймворк Django лучше всего подходит для работы с самыми высокими уровнями трафика [2].

Недостатки Django: использование шаблона маршрутизации с URL, Django является слишком монолитным, все основано на ORM Django, компоненты разворачиваются вместе, необходимо освоить всю систему для работы.

Чтобы ускорить рабочий процесс, можно установить индивидуальную конфигурацию. Раньше разработка кода требовала большого количества времени, а теперь появилась возможность значительно упростить этот процесс.

Laravel – это бесплатный PHP-фреймворк с открытым исходным кодом, предназначенный для разработки веб-приложений, использующий архитектурную модель MVC; он был выпущен в 2011 году, но благодаря

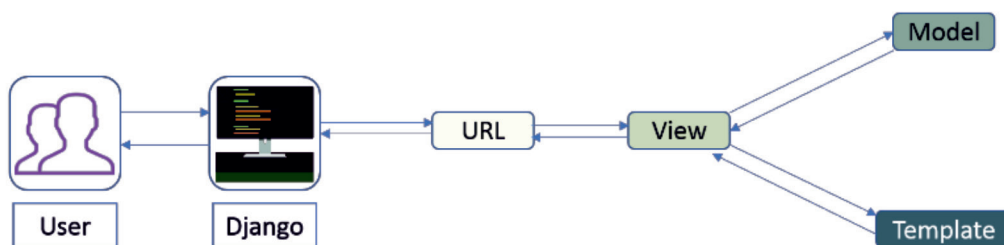


Рис. 1. Шаблон MVC в Django

Django – это веб-фреймворк Python, использующий шаблон проектирования MVC (рис. 1). Django был выпущен в 2005 году и постепенно стал одним из лучших фреймворков для веб-разработки. Возможности фреймворка позволяют сразу приступить к разработке [1].

Django обладает рядом преимуществ. Django был разработан, чтобы помочь разработчикам создавать приложения как можно быстрее. Этот подход предполагает создание идеи, разработку и выпуск проекта, где Django экономит время и ресурсы на каждом из этих этапов. Таким образом, данный фреймворк можно назвать идеальным решением для разработчиков, для которых вопрос дедлайна стоит на первом месте. Django работает с огромным количеством дополнительных функций, которые значительно упрощают реализацию аутентификации пользователей, помогают с картами сайта, администрированием контента, RSS и др. Во время работы вы получаете защиту от ошибок, связанных с безопасностью, таких как SQL-инъекции, clickjacking и т.д., которые ставят под угрозу проект.

быстрым темпам развития и огромному количеству поклонников сейчас это один из самых популярных PHP-движков [3].

К числу преимуществ Laravel можно отнести документацию. Документация Laravel очень структурирована. В ней есть отдельные статьи для каждой конструкции и каждого процесса. Поскольку у этой PHP-платформы много последователей по всему миру, вы можете найти множество различных сообществ, форумов и пользовательских переводов статей в Сети. Структура кода соответствует популярному шаблону проектирования MVC (рис. 2). MVC позволяет сделать код более читаемым, процесс разработки становится комфортнее, дифференцируя работу frontend- и backend-разработчиков [4]. Artisan – это консоль Laravel, которая позволяет использовать команды для работы с миграциями, контроллерами, моделями, авторизацией и другими базовыми компонентами фреймворка. Blade – это настраиваемый шаблонизатор с набором собственных директив. После установки фреймворка Laravel в распоряжении разработчика есть файлы app.js

и `app.css`, которые представляют собой скомпонованные и минимизированные `jQuery` и `Bootstrap` самых последних версий на момент выхода релиза `Laravel`. `Laravel` предоставляет механизм для регистрации и авторизации пользователей. В `Laravel` есть конструкции, позволяющие проводить валидацию данных на основе различных готовых правил. В `Laravel` можно создавать собственные правила, сообщения об ошибках и настраиваемые валидаторы. `Eloquent ORM` – это технология программирования, разработанная для упрощения работы с базами данных, предоставляя методы API для типичных операций (выборка, добавление, обновление, удаление и т.д.). `Laravel` предоставляет набор методов для создания и управления задачами, выполняемыми с помощью планировщика задач `Cron` [5].

К недостаткам `Laravel` можно отнести нарушение обратной совместимости между разными версиями `Laravel`; наличие «синтаксического сахара», к которому разработчики очень быстро привыкают и забывают о том, как правильно писать чистые функции и запросы; нелогичное расположение файлов и каталогов в проекте; требуется хороший уровень владения английским языком: большинство полезных статей и официальной документации написано на английском языке, поэтому разработчики, не владеющие английским языком на уровне чтения

технической документации, не смогут работать с этим фреймворком.

`Spring Framework` – мощный фреймворк для разработки приложений. В более широком смысле `Spring Framework` – это хорошо разработанный инструмент, который поддерживает несколько веб-приложений, использующих `Java` в качестве языка программирования. Фреймворк был замечен на рынке благодаря модульности его базового функционала. Его можно разделить на разные модули, каждый из которых выполняет свою функцию.

Одной из важных особенностей `Spring Framework` является его легкость. Данный фреймворк очень легкий с точки зрения размера и функциональности благодаря реализации `POJO`, которая не требует наследования каких-либо классов и реализации каких-либо интерфейсов. Управление транзакциями используется для объединения нескольких API-интерфейсов управления транзакциями и координации транзакций объектов `Java`. `Spring` проектирует и управляет жизненным циклом и настройками объектов приложения. Внедрение зависимостей позволяет разрабатывать слабосвязанные приложения. Это значительно упрощает выполнение их модульного тестирования. `Spring` обеспечивает интеграцию фреймворков, решающих более сложные проблемы. Например, `IBATIS`, `Hibernate`, `Toplink` и т.д. [6].

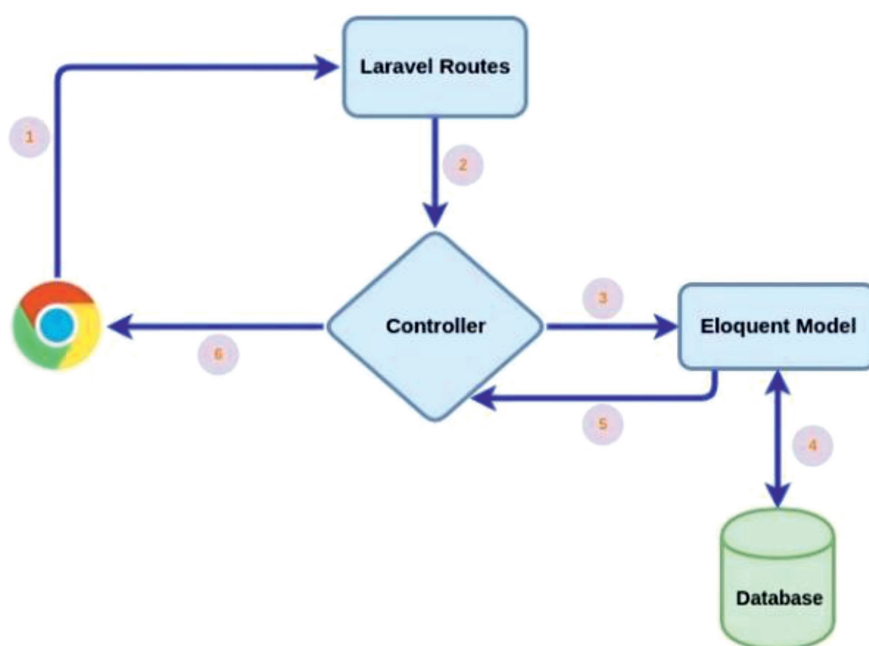


Рис. 2. Архитектура приложения на `Laravel`

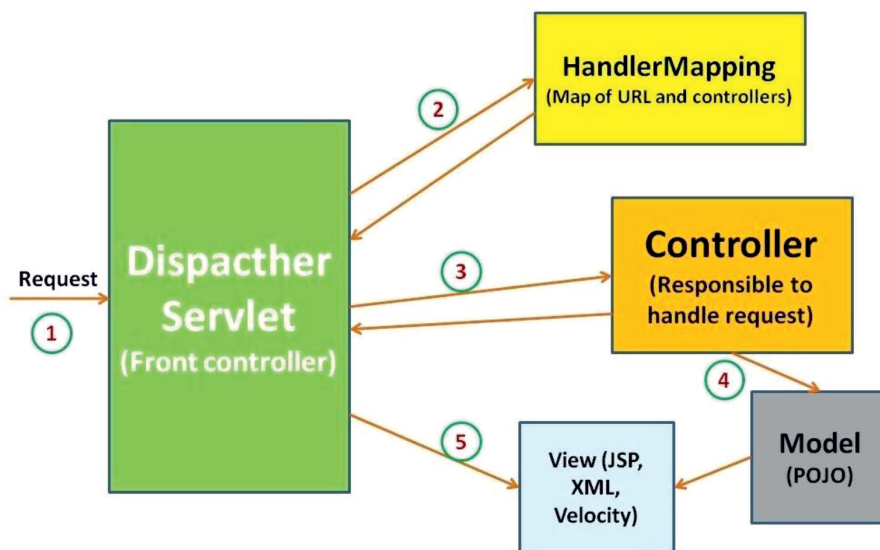


Рис. 3. Шаблон MVC в Spring MVC

Spring Web MVC, также известный как Spring MVC, является веб-средой Spring. Spring MVC позволяет создавать различные приложения: от небольших веб-сайтов до сложных веб-сервисов [7].

Вот так выглядит MVC в Spring (рис. 3):

Model содержит данные, которые требуется отобразить на веб-странице. Однако данные полностью независимы от HTML, это простые объекты Java, из которых состоит приложение.

View будет HTML-шаблоном, который является каркасом для вашей HTML-страницы, написанной с помощью определенных библиотек шаблонов. Эти библиотеки позволяют включать заполнители в шаблоны, которые позволяют получить доступ к данным модели, например имени пользователя.

Controller будет аннотированным методом `@Controller`, который отвечает на HTTP-запрос и знает, как преобразовать HTTP-запрос в объекты Java, а также объекты Java в ответ HTML.

К достоинствам Spring MVC однозначно можно отнести саму концепцию ориентированности на запросы: когда разработчик четко понимает, какие запросы и куда отправляются со стороны клиента, это приводит к более рациональному дизайну приложения, к более оптимизированному и быстрому коду. Поддерживать приложение на Spring MVC легко, потому что все прозрачно. Нет зависимости от сторонних библиотек. Для создания визуальных компонентов и реализации определенной

логики можно использовать библиотеки, известные всем веб-разработчикам, такие как JQuery, Twitter Bootstrap и т.д. [8].

Главным недостатком является длительный срок разработки. Если вам нужно сделать систему в кратчайшие сроки, вы можете брать уже готовые компоненты и не беспокоиться о дизайне. Тогда вполне можно отказаться от Spring MVC.

Ruby on Rails – это фреймворк для веб-разработки, написанный на языке программирования Ruby. Ruby постоянно развивается и окружен большим сообществом. Количество веб-сайтов, созданных с использованием этой платформы, быстро растет.

Философия Ruby on Rails включает два важных ведущих принципа. Don't Repeat Yourself – это принцип разработки программного обеспечения, который гласит, что «каждая часть знания должна иметь единственное, непротиворечивое и авторитетное представление в рамках системы, авторитетное представление в системе». Convention Over Configuration: у Rails есть соглашения о наилучших способах выполнить множество задач в веб-приложении, и эти соглашения выставлены по умолчанию.

Разработка на Ruby on Rails имеет ряд преимуществ для проектов. Первое, что нужно учитывать при выборе правильной серверной среды, – статус Open Source. Это означает, что Ruby on Rails может использоваться бесплатно. Поскольку Ruby on Rails является открытым исходным кодом, есть возможность внести свой вклад в раз-

вите фреймворка. В Ruby on Rails огромное сообщество разработчиков. Множество программистов разрабатывает огромное количество бесплатных дополнений, которые можно интегрировать в приложения [9].

Обширная экосистема делает Ruby on Rails превосходным по сравнению со многими другими фреймворками. RubyGems – сервис хостинга «драгоценных камней» сообщества Ruby, предоставляет доступ к тысячам различных «гемов», которые могут принимать форму дополнений, библиотек или фрагментов программного обеспечения. Драгоценные камни представляют собой готовые решения для различных проблем, которые упрощают процесс разработки. MVC (формат модель-представление-контроллер) – неотъемлемая часть инфраструктуры Ruby on Rails (рис. 4). Ruby on Rails допускает параллельную разработку и позволяет программистам ускорить данный процесс в три раза.

кументируется. Это ускоряет процесс разработки, так как команде разработчиков не нужно выписывать отдельную документацию. В Ruby on Rails есть три среды по умолчанию, а именно: производство, разработка и тестирование. Весь цикл разработки оптимизирован. Есть возможность протестировать продукт, который разрабатывается на каждом этапе. Это приводит к меньшему количеству ошибок и ошибок, о которых вы должны знать и отлаживать.

Несмотря на то что у Ruby on Rails есть много преимуществ, у него есть несколько недостатков, которые необходимо учитывать, прежде чем принимать решение о том, подходит ли он. Одним из главных недостатков является сложность поиска документации и ее нехватка. Скорость выполнения часто называют главным аргументом против Ruby on Rails. При сравнении скорости исполнения Ruby on Rails против Go – Ruby on Rails отстает. Тем не менее, когда

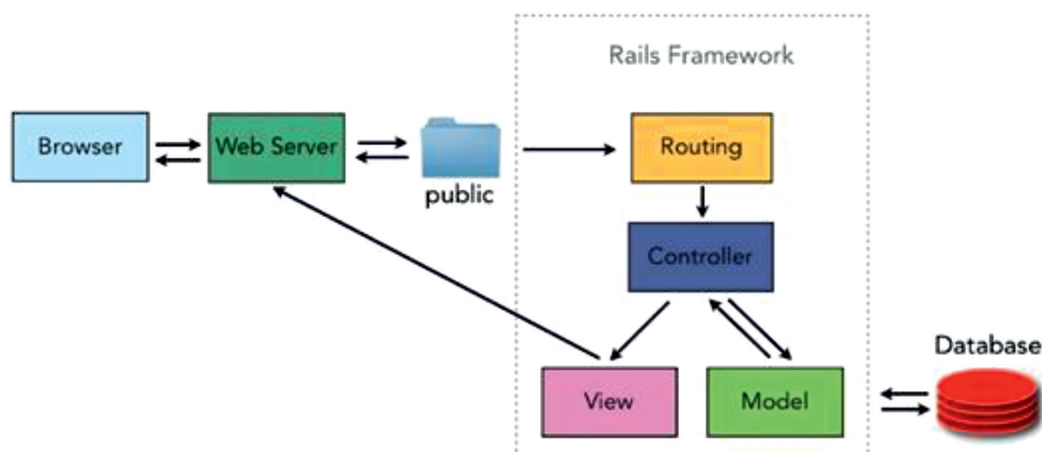


Рис. 4. Шаблон MVC в Ruby on Rails

Инфраструктура Rails предоставляет готовые корзины для разделения бизнес-логики приложения, поэтому компания-разработчик Ruby on Rails может сэкономить время за счет их использования. Разработчики Ruby on Rails могут использовать готовые части кода, что упрощает реализацию многих функций. В результате код приложения становится чистым и имеет высокую читаемость. Приложение, построенное на Ruby on Rails, может быть масштабировано для обработки тысяч запросов в секунду, отправленных несколькими пользователями. В инфраструктуру встроены ориентированные на безопасность функции, которые делают приложения защищенными от SQL-инъекций и XSS-атак. Код Ruby самодо-

мы рассматриваем Java-фреймворк Spring, Ruby on Rails побеждает в этой битве. И, наконец, не все хосты поддерживают Ruby on Rails. Фреймворк требует гораздо больше ресурсов, чем PHP, и не все хостинг-провайдеры могут обеспечить необходимые вычислительные мощности.

Заключение

Таким образом, веб-фреймворк – это воплощение определенных шаблонов, интегрированного набора технологий, облегчающих разработку и поддержку веб-приложений. Чтобы выбрать фреймворк, нужно его проанализировать, изучить возможности и составить список того, что важно в зависимости от поставленной задачи.

Необходимо иметь в виду, что список функций фреймворка – это лишь одна из его характеристик, и больше – не значит лучше.

При рассмотрении проекта с близким дедлайном правильным выбором для решения проблемы будет использование Django. Если вам нужно создать нетривиальный интерфейс, веб-приложение, которое будет работать очень быстро, вам следует выбрать Spring MVC. Нужно сделать выбор в пользу Laravel, если необходимо значительно упростить процесс создания веб-приложения. Если необходимо сконцентрироваться на методах и логике, создать компактное приложение, а не работать с большим объемом кода, то надо выбирать Ruby on Rails, но следует помнить, что язык программирования Ruby – не для новичков. Порог входа высок, поэтому нужно понимать принципы программирования и веб-разработки в целом.

Список литературы

1. Django documentation. [Electronic resource]. URL: <https://docs.djangoproject.com/en/3.1/> (date of access: 03.12.2020).
2. Рябова К.М. Фреймворк Django: архитектура и возможности // Современные технологии: актуальные вопросы, достижения и инновации: сб. статей. Пенза, 2018. С. 118–120.
3. Laravel documentation. [Electronic resource]. URL: <https://laravel.com/docs/8.x> (date of access: 03.12.2020).
4. Brotherton C. The Most Popular PHP Frameworks to Use in 2020. [Electronic resource]. URL: <https://kinsta.com/blog/php-frameworks/> (дата обращения: 03.12.2020).
5. Stauffer M. Laravel: Up and Running: A Framework for Building Modern PHP Apps – O'Reilly Media, 2016. 464 p.
6. Spring. Learn. [Electronic resource]. URL: <https://spring.io/learn> (date of access: 03.12.2020).
7. Web MVC framework. [Electronic resource]. URL: <https://docs.spring.io/spring-framework/docs/3.2.x/spring-framework-reference/html/mvc.html> (date of access: 03.12.2020).
8. Krzywiec W. How to start with Spring MVC. [Electronic resource]. URL: <https://wkrzywiec.medium.com/how-to-start-with-spring-mvc-309dec3c59fd> (date of access: 03.12.2020).
9. Фултон Х., Арко А. Путь Ruby – ДМК Пресс, 2016. 666 с.