Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

|  |  |  |
| --- | --- | --- |
| УДК 007.51 |  |  |
|  |  |  |
|  |  | *На правах рукописи* |

Михалькевич

Александр Викторович

**Разработка web-ориентированного приложения на**

**php-фрэймворке Laravel с использованием шаблона проектирования HMVC**

**АВТОРЕФЕРАТ**

диссертации на соискание степени

магистра техники и технологии

по специальности 1–39 81 01 Компьютерные технологии проектирования электронных систем

Минск 2019

Работа выполнена на кафедре проектирования информационно-компьютерных систем учреждения образования «Белорусский государ­ственный университет информатики и радиоэлектроники»

|  |  |
| --- | --- |
| Научный руководитель: | **АЛЕКСЕЕВ Виктор Федорович,**  кандидат технических наук, доцент, заместитель заведующего кафедрой проектирования информационно-компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» |
| Рецензент: |  |

Защита диссертации состоится «23» января 2019 г. года в 1000 часов на заседании Государственной экзаменационной комиссии по защите магистерских диссертаций в учреждении образования «Белорусский государственный университет информатики и радиоэлектроники» по адресу: 220013, Минск, ул. П.Бровки, 6, копр. 1, ауд. 415, тел. 293-20-80, e-mail: kafpiks@bsuir.by

С диссертацией можно ознакомиться в библиотеке учреждения образования «Белорусский государственный университет информатики и радиоэлектроники».

**ВВЕДЕНИЕ**

Сегодня современными достижениями науки пользуются не только ради экономии средств, но и ради достижения эффективности на предприятии, в строительстве, сельском хозяйстве, городского и земельного кадастра, промышленности, добычи ископаемых, распределительной логистике, космической робототехники, системах обеспечения безопасности, геоинформационных и прочих системах контроля и управления.

В настоящее время не существует электронного ресурса учебных дисциплин, где преподаватели могли бы создавать дисциплины, а студенты заполнять их курсовыми и лабораторными работами.

Целью данной магистерской работы является создание такого электронного ресурса, со следующим функционалом:

- регистрация и авторизация для преподавателей;

- регистрация и авторизация для студентов;

- создание дисциплин для преподавателей;

- создание курсовых работ для студентов;

- создание лабораторных для студентов;

- создание и управление диссертациями;

- создание и управление публикациями;

С итоговой работой можно ознакомиться по адресу [http://erud.by](http://erud.by/) «Электронный Ресурс Учебных Дисциплин».

**ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ**

**Актуальность темы исследования**

На сегодняшний день Интернет играет огромную роль в жизни каждого отдельного человека и общества в целом.

С его помощью можно не только быстро находить необходимую информацию, но и создавать любой контент. Интернет используют и для коммуникаций. Большой популярностью пользуются социальные сети, программы мгновенного обмена сообщениями.

Число пользователей всемирной паутины растет с каждым днем. Создание Электронного Ресурса Учебных Дисциплин является объективной необходимостью.

**Степень разработанности проблемы**

Процесс создания сайта включает в себя несколько этапов. Для достижения наилучшего результата все этапы следует проходить в строгой последовательности.

Первый этап – определение целей создания сайта и его позиционирования. На этом этапе необходимо определить вид сайта, для чего он создается и каких целей необходимо достичь с его помощью. Также определяются ключевые требования.

На втором этапе осуществляется создание дизайн - макета сайта. Производится выбор цветовой схемы сайта, определяется оформление и расположение элементов на страничке, подготавливается шаблон. Разрабатывается визуальное оформление сайта.

На третьем этапе происходит верстка и программирование сайта. Создаются шаблоны страниц, внедряются интерактивные сервисы. Также на этом этапе разрабатывается структура сайта.

На четвертом этапе готовый сайт наполняется разнообразным содержимым: текстовые материалы, графические материалы, видео, аудиозаписи и т.д. Информация, размещенная на сайте, становится доступной для просмотра.

На пятом этапе происходит тестирование сайта и выкладка в сеть Интернет. Готовый сайт необходимо протестировать, на наличие ошибок. После того, как веб-страничка успешно прошла тестирование, производится выбор и настройка сервера. Когда сервер подготовлен и настроен, сайт можно выкладывать в сеть.

Работа с сайтом не заканчивается размещением ресурса в сети Интернет. Необходимо проводить работу по продвижению сайта, а также улучшать и обновлять его.

**Цель и задачи исследования**

Цель диссертации состоит в разработке Электронного Ресурса Учебных Дисциплин на основе php-фрэймворка Laravel с помощью архитектурного шаблона проектирования HMVC.

Для выполнения поставленной цели в работе были сформулированы **следующие задачи**:

1. Разработать систему регистрации, авторизации и кабинет пользователя для преподавателей.

2. Разработать систему регистрации, авторизации и кабинет пользователя для студентов.

3. Разработать алгоритм взаимодействия студентов и преподавателей.

4. Разработать базу данных.

**Область исследования**

Содержание диссертации соответствует образовательному стандарту высшего образования второй ступени (магистратуры) ОСВО 1–39 81 01-2012 специальности 1–39 81 01 Компьютерные технологии проектирования электронных систем.

**Теоретическая и методологическая основа исследования**

В основу работы лег серверный язык программирования PHP. На стороне клиента были использованы HTML, CSS, JavaScript, jQuery и библиотека Bootstrap.

В разработке был использован php-фрэймворк Laravel*.* Архитектурный шаблон проектирования - HMVC

*Информационная база* исследования сформирована на основе литературы, открытой информации, технических нормативно-правовых актов, сведений из электронных ресурсов, а также материалов научных конференций и семинаров.

**Научная новизна**

*Научная новизна* и значимость полученных результатов работы заключается в разработке мульти-интеграционного приложения.

*Теоретическая значимость* работы заключается в подробном описании процесса разработки клиент-серверного приложения на архитектурном шаблоне проектирования HMVC. Рассмотрены все этапы разработки, начиная с создания макета дизайна и до размещения ресурса на собственном сервере.

*Практическая значимость* диссертации состоит в разработанном Электронном Ресурсе Учебных Дисциплин [http://erud.by](http://erud.by/).

**Основные положения, выносимые на защиту**

1. Архитектурный шаблон проектирования HMVC.

2. Одностраничные и интеграционные приложения.

2. Мульти-интеграционное приложение, или приложение, основанное на взаимодействии или интеграции серверов, например сервер Apache и сервер баз данных MySQL.

3. Маршрутизация проекта.

**Апробация диссертации и информация об использовании ее результатов**

Результаты работы по теме диссертации были представлены на международной научно-практической конференции «Проблемы и перспективы развития эксперементальной науки» (г. Тюмень, Российская Федерация, 26 декабря 2018 г.).

По теме диссертации опубликованы 2 учебных пособия, 5 статей, в которых нашли отражение теоретические принципы и результаты работы.

Отдельные положения диссертации могут быть использованы при преподавании дисциплины «Программирование и проектирование встраиваемых мобильных систем», «Проектирование электронных модулей устройств и систем», «Объектно-ориентированное программирование».

**Публикации**

Основные положение диссертации и результаты исследования изложены в шести опубликованных работах общим объемом более 40 страниц.

**Структура и объем работы**

Диссертация состоит из введения, общей характеристики работы, трех глав с краткими выводами по каждой главе, заключения, библиографического списка и приложений.

**В первой главе** приведен анализ технологий разработки Web-приложений. Произведено обоснование выбора технологий, как серверных, так и клиентских, сравнение ядер операционных систем Linux, Mac и Windows. Обоснование выбора шаблона проектирования Web-приложения HMVC.

**Во второй главе** представлена операционная среда разработки. Рассмотрен системный и серверный инструментарий. А также произведено конфигурирование web-приложения.

**В третьей главе** представлены этапы разработки клиент-серверного web-приложения. Представлен макет главной страницы web-приложения. Программирование серверной и клиенской части web-приложения. Рассматриваются возможные следующие варианты размещения web-приложения:

Хостинг. Стандартный хостинг, предоставляемый большинством хостинговых компаний, подходит для большинства web-приложений. В услуги хостинга входят Apache + PHP + MySQL (PHPMyAdmin), а также системы управления сайтами и самим хостингом.

Выделенный сервер. Выделенный сервер - это тот же компьютер, доступный по ssh, http, https, ftp и другим протоколам.

Собственный сервер. Любой компьютер, у которого есть статический ip-адрес сети может выступать в роли сервера.

**В приложении** представлены публикации автора и акт внедрения.

Общий объем диссертационной работы составляет более 100 страниц.

**ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ**

Во **введении** рассмотрено современное состояние web-программирования.

В **общей характеристике работы** показана актуальность ресурса, степень разработанности проблемы, сформулированы цель и задачи диссертации, обозначена область исследований, научная (теоретическая и практическая) значимость исследований, а также апробация работы.

**В первой главе** приведено сравнение операционных систем Linux, Mac и Windows.

Сообществом программистов принято разделять два направления web-разработки: FrontEnd (или разработка на стороне клиента) и BackEnd (или разработка на стороне сервера). Причем, ранее, на заре зарождения web, ожидалось, что основную нагрузку на себя возьмет серверная часть, т.к. сервера будут мощнее компьютеров пользователей. Оказалось же, что компьютеры пользователей по характеристикам ничем не отличаются от серверов. Поэтому нагрузка решений между сервером и клиентом распределена примерно поровну. Именно такое распределение используется в разрабатываемом проекте.

По способу функционирования web-приложения можно разделить на 3 группы:

Одностраничные web-приложения, у которых вся маршрутизация запросов выполняется в одной загруженной странице.

Интеграционные. Работа этих web-приложений основана на взаимодействии серверов, например сервер PHP и сервер MySQL.

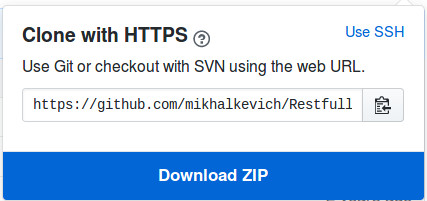
Мульти-инеграционные. Работа этих web-приложении основана на взаимодействии серверных и клиентских API, например технология RestFull API позволяет обойти ограничения SQL-запросов, которые выполняются только на стороне сервера. RestFull API преобразоваывает данные сервера в формат JSON, с которым взаимодействует JavaScript. Формат JSON является единым унифицированным форматом данных как на стороне клиента, так и на стороне сервера.

Мульти-интеграционные программы можно постоянно дописывать и обновлять. Объектно-ориентированные решения в написании кода при построении таких программ предоставляет еще больше гибкости, поскольку классы бибилиотек можно изменять, добавлять и удалять, а объекты, построенные на их основе таких классов могут иметь широкий диапазан характеристик.

Для добавления мульти-инеграционности web-приложению необходимо наличие следующих программ: git, node, а также модуль, преобразовывающий данные из формата MySQL в JSON.

Скачать готовую RestfullApi на Node.js можно по адресу:

https://github.com/mikhalkevich/RestfullApi.git



**Рисунок 1 – Клонирование RestFullApi**

Практически, любое web-приложение можно сделать мульти-интеграционным с помощью RestfullApi. Для этого были созданы модели таблиц базы данных MySQL, которые модуль RestfullApi автоматически преобразовывает в JSON.

**Во второй главе** представлены системный и серверный инструментарий, а также конфигурирование проекта.

Системный инструментарий:

– операционная система Ubuntu;

– встроенный сервер Apache2;

– система контроля версий git;

– удаленный репозиторий github.com;

– интеграционная среда разработки PHPStorm;

– язык программирования JavaScript;

– браузер Firefox;

– клиент обмена мгновенными сообщениями;

– E-mail-клиент;

– FTP-клиент;

– редактор изображений Gimp;

– офисные приложения Libre Office.

Серверный инструментарий

– язык программирования PHP;

– сервер базы данных MySQL;

– серверная технология Node.js;

– менеджер зависимостей для PHP – Copmoser;

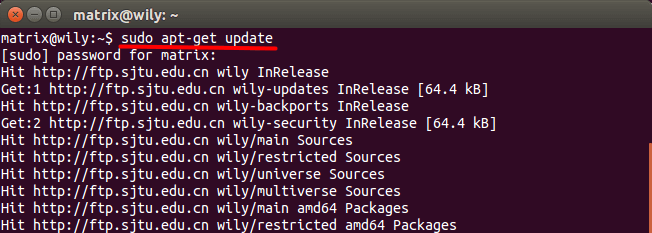
– web-интерфейс для MySQL, система управления базой данных PHPMyAdmin;

– web-интерфейс для управления сервером WebMin.

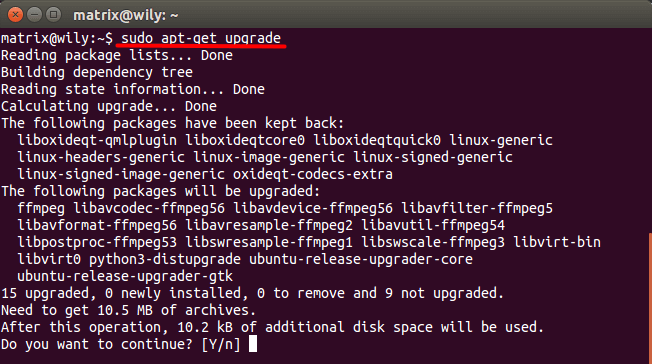
Прежде, чем производить установку программ, необходимо обновить репозитории и пакеты Ubuntu:

*sudo apt-get update*

*sudo apt-get upgrade*



**Рисунок 2 – Обновление репозитория Ubuntu перед установкой программ**



**Рисунок 3 – Обновление пакетов Ubuntu**

В дальнейшем, установка программ будет осуществляться из обновленных репозиториев с помощью команды

apt-get install имя\_программы

**В третьей главе** представлен макет главной страницы web-приложения. Рассмотрено взаимодействие файлов архитектурного шаблона проектирования HMVC. Для разработки web-приложения воспользуемся лучшим на сегодняшний день php-фрэймворком — Laravel.

Официальный сайт laravel – http://laravel.com

У Laravel всего несколько требований к серверу:

- PHP >= 7.2

- OpenSSL PHP Extension

В некоторых операционных системах, например в Linux, может понадобиться ручная установка модулей:

- Mcrypt PHP Extension,

- MbstringPHPExtension.

Laravel не установится без менеджера зависимостей Composer. Поэтому, сперва убедимся в его наличии с помощью команды:

*composer*



**Рисунок 4 – Ответ консоли, свидетельствующий об установленном менеджере зависимости composer**

Если с момента последнего запуска composer прошло более 30 дней, то необходимо обновить composer.

*composer slef-update*

После обновления менеджера зависимостей, можно приступать к установки Laravel.

*Composer create-project laravel/laravel -–prefer-dist*

Laravel установлен.

Папки внутри storage должны быть доступны веб-серверу для записи. Для изменения прав записи, воспользуемся командой chmod

*sudo chmod -R 777 storage*

Для того, чтобы слово public не присутстовало в запросах приложения, создадим файл .htaccess, который будет перенаправлять запрос.

*RewriteEngine On*

*RewriteRule ^(.\*)$ public/$1 [L]*

Далее создадим файл .env на основе существующего примера example.env, который поставляется совместно с Laravel. В Ubuntu это можно сделать с помощью следующих консольных команд:

*cp example.env .env*

*php artisan key:generate*

В файле .env содержатся важные настройки приложения, в том числе и подключение к базе данных.

На данном этапе шаблон установлен.

Laravel поставляется совместно с встроенным программным обеспечением - Artisan. Artisan - название интерфейса командной строки, входящей в состав Laravel. Он предоставляет полезные команды для использования во время разработки вашего приложения. Работает на основе мощного компонента SymfonyConsole.

Чтобы вывести все доступные команды Artisan, воспользуемся командой list:

*php artisan list*

Рассмотрим некоторые команды подробнее. Все они начинаются запросом *php artisan*:

*make:command* — создаёт новый класс команды;

*make:console* — создаёт новую команду Artisan;

*make:controller* — создаёт новый класс контроллера;

*make:event* — создаёт новый класс события;

*make:middleware* — создаёт новый класс промежуточного промежуточного программного обеспечения;

*make:migration* — создаёт новый файл миграции, на основании которого создаются таблицы данных;

*migrate* - выполнение существующих миграций;

make:model — создаёт новый класс модели, опционально и миграцию;

make:provider — создаёт новый класс поставщика услуг;

make:request — создаёт новый класс запроса формы;

event:generate — генерирует пропущенные события и обработчики.

Каждая команда также включает и инструкцию, которая отображает и описывает доступные аргументы и опции для команды. Чтобы её вывести, необходимо добавить слово help перед командой:

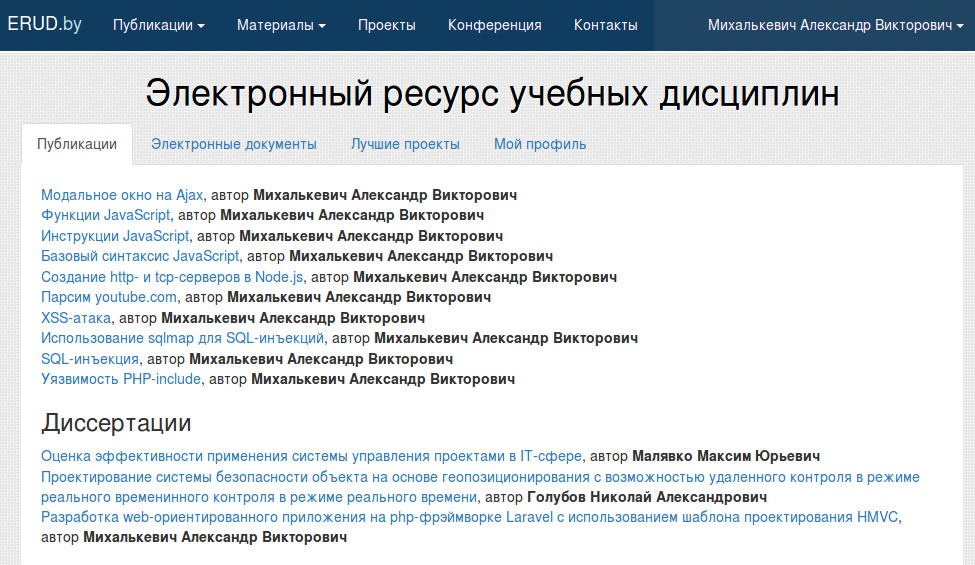
*php artisan help migrate*

Для определения текущей версии Laravel, можно воспользоваться опцией--version

*php artisan —version*

Вся дальнейшая разработка ведется в соответствии с архитектурным шаблоном проектирования HMVC, т.е. производится разделение кода на конроллеры, модели, элементы представления и вспомогательные классы.

Макет главной страницы web-приложения представлен на рисунке 5

**Рисунок 5 – Макет главной страницы erud.by**

В верстке был использован клиентский фрэймворк Bootstrap, который используя современные наработки в области CSS и HTML адаптирует страницу под различные мультимедийные устройства.

**ЗАКЛЮЧЕНИЕ**

В работе представлены: Электронный Ресурс Учебных Дисциплин erud.by. С его помощью преподаватели могут создавать дисциплины, студенты подключаются к дисциплинам, добавляют курсовые и лабораторные работы.

**Основные научные результаты диссертации**

1. Выполнен анализ и сравнение современных операционных систем. Обоснован выбор операционной системы Ubuntu.

2. Разработана система управления публикациями.

3. Научная значимость рработы заключается в подробном описании процесса разработки клиент-серверного приложения на архитектурном шаблоне проектирования HMVC. Рассмотрены все этапы разработки, начиная с создания макета дизайна и до размещения ресурса на собственном сервере.

С полученными результатами работы можно ознакомиться на сайте erud.by.

**Рекомендации по практическому использованию результатов**

Полученные результаты внедрены в учебный процесс на кафедре проектирования информационно−компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» в учебный курс «Программирование и проектирование встраеваемых мобильных систем», «Проектирование электронных модулей устройств и систем», «Объектно-ориентированное программирование», «Разработка программ для операционных систем».

**СПИСОК ПУБЛИКАЦИЙ СОИСКАТЕЛЯ**

1. Михалькевич А.В., Пятосин А.В., Марков А.Н., Боровская О.О. Черепашка для GIT // – Минск : БГУИР, 2018.

2. Михалькевич А.В., Пятосин А.В., Марков А.Н., Боровская О.О. Установка GIT // – Минск : БГУИР, 2018.

3. Михалькевич А.В., Пятосин А.В., Марков А.Н., Боровская О.О. Удаленный репозиторий для GIT // – Минск : БГУИР, 2018.

4. Михалькевич А.В., Пятосин А.В., Марков А.Н., Боровская О.О. Основы GIT // – Минск : БГУИР, 2018.

**РЭЗЮМЭ**

**Мiхалькевiч Аляксандр Вiктаравiч**

**Распрацоўка web-арыентаванага прыкладання на php-фрэймворке Laravel з выкарыстаннем шаблону праектавання HMVC**

***Ключавыя словы***: інтэрнэт, web-дадатак, сістэма кантролю версій, сервер, выдалены рэпазітар

***Мэта працы:*** Распрацоўка web-арыентаванага прыкладання на Laravel з выкарыстаннем шаблону праектавання HMVC

***Атрыманыя вынікі і іх навізна:*** Практычная значнасць дысертацыі складаецца ў распрацаваным электронныя рэсурсы навучальных дысцыплін. З выніковай працай можна азнаёміцца па адрасе http://erud.by. У працэсе напісання дысертацыі, быў выраблены аналіз аперацыйных сістэм. У якасці асноўнай аперацыйнай сістэмы сервера была абраная Ubuntu.

***Ступень выкарыстання:*** вынікі ўкаранёны ў навучальны працэс на кафедры праектавання інфармацыйна-камп'ютэрных сістэм ўстановы адукацыі «Беларускі дзяржаўны універсітэт інфарматыкі і радыёэлектронікі» у навучальны курс «Праграмаванне і праектаванне ўбудавальных мабільных сістэм», «Праектаванне электронных модуляў прылад і сістэм», «Аб'ектна-арыентаванае праграмаванне», «Распрацоўка праграм для аперацыйных сістэм».

***Вобласць ужывання:*** праграмаванне, адукацыя, будаўніцтва, сельская гаспадарка, размеркавальная лагістыка, касмічная робататэхніка, сістэмы кіравання

**РЕЗЮМЕ**

**Михалькевич Александр Викторович**

**Разработка web-ориентированного приложения на Laravel с использованием шаблона проектирования HMVC**

***Ключевые слова*:** интернет, web-приложение, система контроля версий, сервер, удаленный репозиторий.

***Цель работы*:** Разработка web-ориентированного приложения на Laravel с использованием шаблона проектирования HMVC.

***Полученные результаты и их новизна*:** Практическая значимость диссертации состоит в разработанном Электронном Ресурсе Учебных Дисциплин. С итоговой работой можно ознакомиться по адресу [http://erud.by](http://erud.by/). В процессе написания диссертации, был произведен анализ операционных систем. В качестве основной операционной системы сервера была выбрана Ubuntu.

***Степень использования*:** результаты внедрены в учебный процесс на кафедре проектирования информационно-компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» в учебный курс «Программирование и проектирование встраиваемых мобильных систем», «Проектирование электронных модулей устройств и систем», «Объектно-ориентированное программирование», «Разработка программ для операционных систем».

***Область применения*:** программирование, образование, строительство, сельское хозяйство, распределительная логистика, космическая робототехника, системы управления.

**SUMMARY**

**Mikhalkevich Alexander Viktorovich  
Developing a Laravel web-based application using the HMVC design pattern**

***Keywords*:** Internet, web application, version control system, server, remote repository.

***The object of study*:** Developing a Laravel web-based application using the HMVC design pattern

***The results and novelty*:** The practical significance of the thesis consists in the developed Electronic Resource of Educational Disciplines. The final work can be found at http://erud.by. In the process of writing a thesis, an analysis of operating systems was performed. Ubuntu was chosen as the main server operating system.

***Degree of use*:** the results were introduced into the educational process at the department of design of information and computer systems of the educational institution “Belarusian State University of Informatics and Radioelectronics” in the training course “Programming and Designing Embedded Mobile Systems”, “Designing Electronic Modules of Devices and Systems”, "Development of programs for operating systems."

***Sphere of application***: programming, education, construction, agriculture, distribution logistics, space robotics, control systems.