МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ

ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»



Основная профессиональная образовательная программа

Направление подготовки 09.03.01 Информатика и вычислительная техника

Направленность (профиль) «Технологии разработки программного обеспечения»

форма обучения – очная

**Выпускная квалификационная работа**

Разработка электронного образовательного ресурса «Развертывание современного веб-приложения на Python»

Обучающегося 4 курса

Филатова Ивана Александровича

Научный руководитель:

Кандидат педагогических наук, доцент

Государев Илья Борисович

Санкт-Петербург

2020

Оглавление

[ВВЕДЕНИЕ 3](#_Toc41137224)

[ГЛАВА 1 5](#_Toc41137225)

[1.1 Анализ существующих решений 5](#_Toc41137226)

[1.2 Анализ инструментов и технологий создания ЭОР 9](#_Toc41137227)

[ВЫВОДЫ К ГЛАВЕ 1 20](#_Toc41137228)

[ГЛАВА 2 21](#_Toc41137229)

[2.1 Структура веб-приложения 22](#_Toc41137230)

[2.2 Разработка и развёртывание ЭОР 26](#_Toc41137231)

[ВЫВОДЫ К ГЛАВЕ 2 30](#_Toc41137232)

[ЗАКЛЮЧЕНИЕ 31](#_Toc41137233)

[СПИСОК ЛИТЕРАТУРЫ 32](#_Toc41137234)

# ВВЕДЕНИЕ

На данный момент на рынке образовательных услуг представлено достаточно много различных электронных образовательных ресурсов (далее - ЭОР), которые предоставляют возможность для приобретения практических навыков во многих областях знаний. Кроме того, в таких ресурсах содержится большое количество теоретической информации. Однако, если смотреть в разрезе информатики и вычислительной техники, многие ресурсы предоставляют информацию на платной основе, предварительно сообщая пользователю только общее содержание курса. Другие же представляют из себя, если брать конкретно тему данной работы, чистую документацию, сопровождающую облачные сервисы и платформы. Зачастую, даже не переведённую на русский язык.

Таким образом, **актуальность** данной выпускной квалификационной работыобуславливается необходимостью создания электронного образовательного ресурса с более простым и понятным объяснением материала, который сопровождается демонстрацией на конкретном примере с использованием медиаконтента.

**Предметом** данной работы будет разработка электронного образовательного ресурса.

**Цель:** разработка электронного образовательного ресурса на тему «Развёртывание современного веб-приложения на Python».

Для достижения данной цели необходимо решить следующие задачи:

1. Проанализировать существующие способы создания ЭОР и на основе данного анализа сформировать перечень инструментальных средств, подходящих для реализации ЭОР;
2. Отобрать инструментальные средства, подходящие под требования текущей задачи, для дальнейшей реализации;
3. Спроектировать и реализовать ЭОР, отвечающий требованиям и задачам данной работы;
4. Разработать список рекомендаций по развертыванию современных веб-приложений на Python.

**Результатом** данной выпускной квалификационной работы будет электронный образовательный ресурс, помогающий в освоении темы «развёртывание современного веб-приложения на Python».

# ГЛАВА 1

Для начала необходимо разобраться, что такое «электронный образовательный ресурс». ГОСТ Р 53620-2009 «Информационно-коммуникационные технологии в образовании. Электронные образовательные ресурсы. Общие положения» предлагает следующую трактовку: «электронный образовательный ресурс; ЭОР: Образовательный ресурс, представленный в электронно-цифровой форме и включающий в себя структуру, предметное содержание и метаданные о них» [1]. Иными словами, это практически любой ресурс в интернете, на котором расположена структурированная информация, помогающая пользователю разобраться в той теме, что была рассмотрена в рамках конкретного электронного образовательного ресурса.

Таким образом, можно сделать вывод, что практически любой электронный ресурс, несущий в себе полезную информацию и подходящий под названные выше критерии, можно назвать электронным образовательным ресурсом. Иными словами нет ограничений на использование инструментальных средств для разработки и реализации ЭОР, так как данное описание включает в себя как презентации, созданные в Microsoft Powerpoint, так и образовательные платформы наподобие Coursera и «Открытое образование».

## 1.1 Анализ существующих решений

В настоящем разделе представлен анализ существующих электронных образовательных ресурсов, посвященных развёртыванию современных веб-приложений на Python. Для проведения данного исследования был произведен поиск в сети Интернет с использованием нескольких поисковых систем с отбором всех электронных образовательных ресурсов, позволяющих приобрести знания и навыки по теме «развертывание современного веб-приложения на Python».

Для качественного анализа был разработан следующий перечень критериев оценки ЭОР:

1. Доступность представленной информации;
2. Удобство расположения материала курса на странице;
3. Наличие видео- и аудиоматериалов;
4. Удобство использования ЭОР;
5. Наличие стороннего материала на странице;
6. Достоверность представленной информации.

В итоге были найдены только статьи компаний, представляющих свои программные продукты в сфере облачных технологий, таких как Microsoft Azure (см. Рисунок 1), AWS Amazon (см. Рисунок 2), 8host, а также Heroku, который доступен только на английском языке. Кроме того, были найдены статьи на платформе Habr, в которых рассказывалось о личном опыте автора.



Рисунок Microsoft Azure



Рисунок AWS Amazon

Также был найден электронный учебник Django от компании Mozilla [10], но он представляет из себя пошаговое руководство на примере сайта библиотеки и не включает в себя какой-либо мультимедийный контент (см. Рисунок 3). Такой способ подачи материала может быть сложен для восприятия некоторым типами людей.



Рисунок Руководство от Mozilla Developer

Однако был найден один ресурс, который освещал тему развертывания современного веб-приложения на Python, но провести качественный анализ данного электронного образовательного ресурса не оказалось возможным в виду высокой стоимости самого курса. По состоянию на 10 мая 2020 года, курс на сайте otus.ru «Web-разработчик на Python» стоит 70 тысяч рублей (см. Рисунок 4).



Рисунок – Главное окно ЭОР OTUS

Таким образом можно сделать вывод, что на данный момент тема развертывания современного веб-приложения на Python находится в свободном доступе и освещается лишь в статьях тех авторов, которые делятся личным опытом, а также в документации к различным облачным платформам, что может оказаться сложным для новичков, только начавших изучение веб-разработки на языке программирования Python.

## 1.2 Анализ инструментов и технологий создания ЭОР

На сегодняшний день любому человеку доступен достаточно широкий выбор различных инструментальных средств, позволяющих реализовать электронный образовательный ресурс любой тематики.

После их сравнения и отсеивания, были выбраны следующие инструменты и технологии, подходящие для реализации данного электронного образовательного ресурса:

* Django
* Flask
* Vue
* Bootstrap
* CodeIgniter
* WordPress
* HTML + CSS

Также могут быть использованы сервисы от компании Google, так как они позволяют создавать образовательные ресурсы с мультимедийным контентом, тестами, обратной связью с создателем ЭОР, но при этом они просты в освоении и доступны большинству пользователей и, фактически, требуют только наличие интернета и минимальных навыков обращения с данными сервисами.

Так как большая часть представленных инструментов и технологий являются веб-фреймворками, разберем их назначение подробнее.

Веб-фреймворк – это программный инструмент, предоставляющий возможность создавать и запускать веб-приложения. В результате это упрощает написание кода и ускоряет разработку [8].

На данный момент существует достаточно обширный выбор различных веб-фреймворков, помогающих реализовать практически любое веб-приложение. Однако некоторые из них отличаются высоким порогом вхождения и могут быть использованы только специалистами с достаточно большим опытом работы или владеющим малопопулярным языком программирования, или же изучившим какой-то особый язык программирования, как, к примеру, TypeScript в Angular [9].

Но вернемся к анализу достоинств и недостатков, указанных выше инструментов и технологий.

**Django**

Данный веб-фреймворк основан на языке программирования Python. Основным принципом данного фреймворка можно назвать – DIY (don’t repeat yourself), что переводится как «не повторяйся». Это означает, что главной отличительной чертой данного типа фреймворков является модульность веб-приложений, когда каждая отдельная функция, к примеру вывод текста статьи на экран или ввод сообщений блога на страницу, является отдельным приложением, которое может быть подключено, отключено, либо, при наличии такой необходимости, заменено на другое приложение.

Django реализует в себе архитектуру MTV (Model-Template-View) (см. Рисунок 5), схожую по своей сути с моделью MVC (Model-View-Controller) (см. Рисунок 6). Разница только в том, что Controller в стандартной модели MVC в Django примерно идентичен уровню, который называется View, а логика View из MVC реализуется в Django уровнем Template [11].



Рисунок MTV



Рисунок Модель MVC

Как можно заметить, данные модели действительно крайне похожи друг на друга.

**Достоинства Django:**

* Быстрота – Django обеспечивает высокую скорость разработки благодаря MTV архитектуре и принципу DIY;
* Принцип “Battery inside” – данный веб-фреймворк содержит в себе обширный набор инструментов, которые позволят реализовать необходимое веб-приложение, без подключения дополнительных библиотек;
* Стандартизированная структура веб-приложения – это позволяет новым разработчикам быстрее включиться в процесс разработки, получить помощь от сообщества или же найти готовые решения;
* Приложения – позволяет использовать уже написанные модули других пользователей, которые расположены в свободном доступе на специальном сайте <https://djangopackages.org/>;
* Безопасность – в Django изначально предоставляет механизмы защиты от распространённых атак, таких как межсайтовый скриптинг и внедрение SQL-кода [12];
* Масштабируемость – Django предоставляет возможность развивать веб-приложение, даже если изначально оно было небольшим, без переписывания всего кода.

**Недостатки Django:**

* Django ORM – данная ORM не позволяет использовать базы данных также легко и свободно как SQLAlchemy, которая чаще всего используется, к примеру, в том же Flask, так как автоматизирует множество однотипных взаимодействий с базами данных, а также использует объектно-ориентированный подход, вместо инструкций SQL;
* Скорость развития – Django представляет из себя крупный веб-фреймворк, известный своей надёжностью и монолитностью, которые позволяют сообществу разрабатывать огромное количество универсальных модулей, но именно по этой причине он развивается достаточно медленно, в сравнении с другими веб-фреймворками [13].

**Flask**

Это микро-фреймворк, который так же, как и Django основан на Python. Flask поставляется с меньшим набором вложенных инструментов, представляя из себя конструктор, который может быть сложен из различных деталей – библиотек и модулей. Он появился в противовес таким фреймворкам как Django и Pyramid, так как была потребность в использовании различных компонентов, в то время как ни один из вышеназванных не позволял это делать в таком объеме [14].

**Достоинства Flask:**

* Минималистичность – данный веб-фреймворк предоставляется в минимальной комплектации, а значит, что его можно свободного настроить в соответствии с требованиями текущего проекта;
* Больший контроль в сравнении с Django – более тонкая настройка позволяет получить больший контроль над всем приложением, включая базы данных, которые в Django обрабатываются в автоматическом режиме.

**Недостатки Flask:**

* Сложнее в освоении – из-за модульности данного микро-фреймворка возникает проблема освоения тех библиотек, которые будут необходимы для текущего проекта и, как следствие, появится необходимость в изучении дополнительной информации;
* Изначальное отсутствие административного интерфейса – как и следует ожидать от микро-фреймворка, он не предоставляет стандартной библиотеки для работы административного интерфейса, но его можно подключить с помощью расширения Flask-Admin

Сложно назвать конкретные достоинства и недостатки данного фреймворка, так как его минималистичность позволяет его настраивать под любые нужды и масштабировать под любые проекты и дает контроль над каждой частью веб-приложения, но при этом появляется высокий входной порог, так как увеличивается количество необходимых знаний и навыков для освоения Flask.

Однако стоит отметить, что сам по себе Flask достаточно прост в освоении и позволяет быстро адаптироваться под него опытным пользователям [15].

**Vue**

Данный веб-фреймворк основан на языке JavaScript. Он был запущен в 2013 году и идеально подходит для создания высоко адаптируемых пользовательских интерфейсов и сложных одностраничных приложений [16]. Также данный фреймворк считается реактивным (см. Рисунок 7).



Рисунок Реактивность

**Достоинства Vue:**

* Адаптивность – Vue крайне похоже на React и Angular, что позволяет разработчику быстро освоиться в использовании нового веб-фреймворка;
* Принцип “Battery inside” – Vue поставляется с уже готовыми библиотеками, что позволяет приступить к разработке без установки дополнительных модулей;
* Скорость – он, конечно, не самый быстрый, но запаса производительности хватает для большей части веб-приложений;
* Маленький размер – при весе примерно в 91 КБ он остается производительным и гибким веб-фреймворком, с помощью которого можно реализовать многие проекты [17].

**Недостатки Vue:**

* Малая вовлеченность – при общем удобстве использования Vue, он остается малоиспользуемым веб-фреймворком и, как следствие, имеет более слабую поддержку сообщества, из-за чего некоторые проблемы необходимо будет решать самостоятельно [17].
* Излишняя гибкость – следствие предыдущего пункта, так как сообщество разработчиков, использующих Vue достаточно небольшое, могут возникнуть проблемы при интеграции данного веб-фреймворка в крупные проекты, но судя по тенденциям роста популярности данного решения [17], текущая проблема может перестать быть актуальной в скором времени.

**Bootstrap**

Данный веб-фреймворк был разработан в компании Twitter для внутреннего пользования, однако в дальнейшем был опубликован в открытый доступ. Bootstrap включает в себя набор различных элементов для веб-сайтов, среди которых есть кнопки, поля ввода и веб-формы [18]. Также данный фреймворк основывается на работе с Grid – вертикальные и горизонтальные линии, для более эффективного использования свободного пространства страницы (см. Рисунок 8).



Рисунок Grid

**Достоинства Bootstrap:**

* Адаптивность – данный веб-фреймворк позволяет разрабатывать веб-сайты для их корректного отображения на устройствах с разными экранами;
* Низкий порог вхождения – Bootstrap прост в освоении и позволяет быстро приступить к разработке, не тратя много времени на изучение обучающих материалов, которых существует огромное количество;
* Корректная работа во всех современных браузерах;
* Возможность создать сайт в едином стиле, не выбиваясь из него.

**Недостатки Bootstrap:**

* В старых браузерах веб-сайт, разработанный с помощью данного веб-фреймворка может отображаться не так, как задумывал разработчик по причине постоянных обновлений и использовании современных веб-технологий;
* Шаблонность – часто, веб-сайты разработанные с помощью Bootstrap имеют схожую структуру или элементы.

**Фреймворк CodeIgniter**

CodeIgniter это микро-фреймворк основанный на PHP. Из-за того, что компоненты и процедуры системы выполняются исключительно по запросу, она по умолчанию легкая и достаточно быстрая. Но, несмотря на это, объем отображенных на странице данных находится в прямой зависимости от выделенных ресурсов сервера [19].

Также, как и в Django реализована архитектура MVC

**Достоинства CodeIgniter:**

* Легкий в настройке;
* Обширное сообщество разработчиков, поддерживающих данный веб-фреймворк;
* Предоставляет возможность быстрой разработки веб-ресурсов;
* Слабая связанность компонентов;
* Большое количество различных библиотек.

**WordPress**

Это CMS (Content Management system), написанная с помощью языка программирования PHP и имеющая открытый исходный код. Как следствие, WordPress имеет гибкую структуру, которую можно подстроить под любой проект. Кроме того, для данной системы в открытом доступе находится огромное количество различных шаблонов [20].

**Достоинства WordPress:**

* Открытый исходный код;
* Адаптивность – данную систему можно настроить под любой проект;
* Для расширения возможностей шаблонов можно установить плагины или переписать код шаблона.

**Недостатки WordPress:**

* Высокая нагрузка на сервер при использовании большого количества различных плагинов;
* Низкое качество некоторой части доступных шаблонов в следствии низкого порога вхождения.

**HTML и CSS**

HTML и CSS являются основными частями любого сайта, расположенного в сети Интернет. Они используются для наполнения контентом страниц сайтов и оформления в каком-либо стиле. Часто к ним добавляется скриптовый язык JavaScript для добавления функциональности и придания динамичности.

HTML – это язык гипертекстовой разметки, который обрабатывается браузерами и отображается на странице в видео понятной для человека формы.

CSS – это каскадные стили, то есть формальный способ описания внешнего вида документа, который был написан с использованием языка гипертекстовой разметки HTML.

Главными достоинствами данного тандема являются:

* Поддержка всеми браузерами;
* Большое количество учебников, видеоуроков и курсов в свободном доступе;
* Большое количество различных средств разработки, предоставляемых разработчиками бесплатно;
* Достаточно просты в освоении.

# ВЫВОДЫ К ГЛАВЕ 1

1. На основании проведённого анализа было выявлено отсутствие материалов по развертыванию современного веб-приложения на Python в свободном доступе.
2. Был произведён сбор технологий и инструментальных средств, подходящих под требования данной выпускной квалификационной работы. Произведённый анализ показал, как достоинства, так и недостатки каждой из отобранных технологий.
3. В результате анализа было принято решение о целесообразности использования фреймворка Flask для backend разработки совместно с Bootstrap для frontend как наиболее соответствующих требованиям данной выпускной квалификационной работы.

# ГЛАВА 2

Так как данный электронный образовательный ресурс разрабатывался в виде веб-приложения с использованием веб-фреймворка Flask, стоит упомянуть ГОСТ, который регулирует разработку данного вида приложений - ГОСТ Р ИСО 9241-151-2014.

Он определяет следующее:

* Веб-приложение должно иметь четко поставленную цель, на основании которой ведётся дальнейшая разработка информационного наполнения и функциональных возможностей [21, пункт 6.2];
* Во время определения цели должны быть выявлены целевые группы пользователей, для которых и будет разрабатываться данное веб-приложение [21, пункт 6.3];
* Должны быть проанализированы цели и задачи предполагаемых целевых групп [21, пункт 6.4];
* Цель веб-приложения и цели пользователей должны быть сравнимы или, по крайней мере, соответствовать друг другу, для достижения наилучшего результата [21, пункт 6.5];
* Контент должен соответствовать назначению веб-приложения, пользователям и их задачам [21, пункт 7.1.3];
* При использовании компьютера пользователя для хранения какой-либо информации, должна открыто предоставляться политика использования этих данных и программ [21, пункт 7.2.8.4];
* По возможности использовать только общепринятые стандарты веб-технологий для веб-приложений и задач пользователей, применяемым в соответствии с требованиями стандарта [21, пункт 10.6].

Для разработки данного электронного образовательного ресурса использовалась каскадная модель. Она была выбрана, так как позволяет наиболее полно реализовать процесс разработки и является наиболее строгой – новая стадия разработки не начнётся до завершения предыдущей (см. Рисунок 9).



Рисунок – схема модели разработки «Каскад»

В условиях создания небольшого веб-приложения с изначально известными целью и сроком завершения разработки, такая модель покажет лучший результат, если сравнивать с другими моделями, основывающимися на постоянных корректировках процесса разработки.

## 2.1 Структура веб-приложения

Разработанный электронный образовательный ресурс имеет иерархическую структуру (см. Рисунок 10).



Рисунок – схема структуры веб-приложения

Как видно по схеме, стартовой страницей данного электронного образовательного ресурса является «Главная страница». Сначала пользователь попадает именно на эту страницу (см. Рисунок 11).



Рисунок – скриншот главной страницы

Затем пользователь может зайти как на страницу «Об Авторе», так и на страницу «Курс».

Главное меню, которое находится сверху, позволяет попасть в любую из трёх главных страниц данного веб-приложения.

Страница «Об авторе» содержит информацию о создателе данного электронного образовательного ресурса, а также о целях и технологиях, которые были использованы при разработке (см. Рисунок 12).



Рисунок – скриншот страницы «Об авторе»

Страница «Курс» содержит ссылки на видеоуроки, в которых дается соответствующая информация по теме, описанной в каждой их плиток, расположенных под главным меню. Данная структура расположения материала на странице называется плиточной и позволяет более полно использовать предоставленное пространство страницы (см. Рисунок 13).



Рисунок – Скриншот страницы «Курс»

Данная страница ведёт на страницу с материалами урока, в которые входят:

* Видеоурок;
* Текстовые справочные материалы, соответствующие теме конкретного урока;
* Google форму с тестом по материалу текущего урока.

Google форма требует ввода имени и фамилии студента, а также адрес электронной почты. Это необходимо для идентификации пользователя для дальнейшего информирования о результатах прохождения тестирования.

Google форма собирает ответы пользователей и сводит их в таблицу, на основании которой проходит проверка усвоения студентами материала урока.

## 2.2 Разработка и развёртывание ЭОР

Во время разработки был использован Flask с добавлением пакета SQLAlchemy, который позволил реализовать работу с базой данных и ввести возможность регистрации и входа в свой аккаунт. Для разработки онлайн тестов были использованы возможности Google форм (см. Рисунок 14)



Рисунок

В процессе разработки данного электронного образовательного ресурса была использована интегрированная среда разработки (ИСР) для языка программирования Python – PyCharm.

Данная ИСР позволяет создавать программные продукты на языке Python и поддерживает большое количество различных пакетов, включая Flask, на котором и было написано веб-приложение, и Django, который был проанализирован в первой главе. Кроме того, PyCharm позволяет создавать и редактировать не только файлы .py, в которых содержится и записывается программный код на языке Python, но и другие типы файлов. К примеру, файлы, имеющие расширение .html, в которых записывается HTML-код страниц веб-приложения или же .css, которые используются для хранения каскадных стилей.

Данная интегрированная среда разработки проста в установке, и имеет интуитивно понятный интерфейс (см. Рисунок 15). Интерфейс демонстрируется на основе проекта данного электронного образовательного ресурса.



Рисунок – скриншот пользовательского интерфейса ИСР PyCharm

Для оформления внешнего вида страницы был использован фреймворк Bootstrap. Он позволил в короткие сроки разработать пользовательский интерфейс веб-приложения и сильно упростил задачу в общей разработке.

Так как Bootstrap является по своей сути уже настроенным набором адаптационных шаблонов и стилей, которые находятся в открытом доступе, возможно использовать его для создания любых решений связанных с frontend разработкой. Он использовался не в качестве внутреннего файла стиля, а с BootstrapCDN. Иными словами, это позволяет загружать CSS, JavaScript и изображения удаленно со своих серверов [22].

Также для упрощения разработки была использована возможность веб-фреймворка Flask для создания шаблонов отображения страниц данного приложения.

Таким образом были достигнуты следующие цели:

* Увеличение скорости разработки;
* Повышение качества конечного продукта;
* Упрощение кода для дальнейшего расширения веб-приложения и добавления новых курсов и возможностей;

Была использована облачная PaaS-платформа Heroku для развертывания разработанного веб-приложения (см. Рисунок 16).



Рисунок – страница с настройками веб-приложения, развёрнутого на яHeroku

Данная облачная платформа основана на управляемой контейнерной системе [23]. Она позволяет развертывать и запускать современные веб-приложения.

Платформа обеспечивает защиту как самого приложения, запущенного на данном сервисе, так и защиту данных пользователей. Heroku также включает в себе широкий спектр собственных технологий и модулей.

Однако их не всегда достаточно для успешной реализации проекта. Поэтому на данной платформе существует возможность подключения приложений сторонних разработчиков. Они называются настройками и могут быть использованы в качестве сервисов.

# ВЫВОДЫ К ГЛАВЕ 2

1. На основании требований и выявленных в ходе работы критериев был спроектирован электронный образовательный ресурс в виде веб-приложения на тему «развертывание современного веб-приложения на Python». При разработки использовалась каскадная модель жизненного цикла.
2. Разработанный программный продукт будет представлен кафедре информационных технологий и электронного обучения РГПУ им. А.И. Герцена.

# ЗАКЛЮЧЕНИЕ

Во время написания данной выпускной квалификационной работы были решены следующие задачи:

1. Проведён анализ инструментов для создания электронных образовательных ресурсов;
2. Проанализированы существующие ресурсы по теме исследования и выявлено отсутствие таких ресурсов в свободном доступе и в простом формате изложения материала;
3. Были отобраны технологии и средства разработки веб-приложения;
4. Был разработан электронный образовательный ресурс на тему «развертывание современного веб-приложения на Python».

Данный электронный образовательный ресурс создан на веб-фреймворке Flask, которая позволяет расширить в дальнейшем данное веб-приложение и добавить новые курсы, которые позволят привлечь большее количество пользователей и помочь в изучении, к примеру, языка программирования Python или веб-разработки в целом.

На данный момент этот ресурс может использоваться на кафедре информационных технологий и электронного обучения РГПУ им. А.И. Герцена в качестве дополнительного онлайн курса для обучения студентов.

# СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ Р 53620-2009 Информационно-коммуникационные технологии в образовании. Электронные образовательные ресурсы. Общие положения
2. ГОСТ Р 52653-2006 Информационно-коммуникационные технологии в образовании. Термины и определения
3. Войтович, И.К. Создание электронных образовательных ресурсов на основе сервисов Google / И.К.Войтович // 2014.
4. Чекалина, Т.А. Создание электронных образовательных ресурсов в профессиональных образовательных организациях / Т.А.Чекалина // 2014.
5. Развертывание веб-приложения Python на виртуальной машине в Azure Stack Hub [Электронный ресурс]. [Режим доступа] <https://docs.microsoft.com/ru-ru/azure-stack/user/azure-stack-dev-start-howto-vm-python?view=azs-2002> (Дата обращения: 28.04.2020)
6. Работа с образом Django в Vscale [Электронный ресурс]. Vscale community. 2017. [Режим доступа] <https://community.vscale.io/hc/ru/community/posts/115002064205-%D0%A0%D0%B0%D0%B1%D0%BE%D1%82%D0%B0-%D1%81-%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%BC-Django-%D0%B2-Vscale> (Дата обращения: 28.04.2020)
7. Getting Started on Heroku with Python [Электронный ресурс]. Heroku Dev Center. 2018. [Режим доступа] <https://devcenter.heroku.com/articles/deploying-python> (Дата обращения: 29.04.2020)
8. Ryabtsev, A. Web Frameworks: How To Get Started [Электронный ресурс] / A.Ryabtsev // DjangoStars. [Режим доступа] <https://djangostars.com/blog/what-is-a-web-framework/> (Дата обращения: 05.05.2020)
9. Стариков, Н. Наиболее популярные Javascript фреймворки для быстрой веб разработки: что выбрать? [Электронный ресурс] / Н.Стариков // Stfalcon. 2019. [Режим доступа] <https://stfalcon.com/ru/blog/post/javascript-frameworks-for-software-development> (Дата обращения: 05.05.2020)
10. Веб-фреймворк Django (Python) [Электронный ресурс]. Ресурсы для разработчиков, от разработчиков. 2020. [Режим доступа] <https://developer.mozilla.org/ru/docs/Learn/Server-side/Django> (Дата обращения 08.05.2020)
11. Django — фреймворк для веб-разработки на Python [Электронный ресурс]. Web creator. [Режим доступа] <https://web-creator.ru/articles/django> (Дата обращения: 10.05.2020)
12. Плюсы и минусы Django [Электронный ресурс]. Python 3. [Режим доступа] <https://python-scripts.com/django-obzor> (Дата обращения: 10.05.2020)
13. Tomasz Bak. Pros and Cons of Django as web framework for Python developers [Электронный ресурс]. Medium. 2019. [Режим доступа] <https://medium.com/@tomaszbak/pros-and-cons-of-django-as-a-web-framework-for-python-developers-b3c7ee02548> (Дата обращения: 11.05.2020)
14. Django vs. Flask: что лучше для вашего веб-приложения? [Электронный ресурс]. Tech. 2017. [Режим доступа] <https://techrocks.ru/2017/09/29/django-vs-flask-what-is-the-best-for-your-web-application/> (Дата обращения: 11.05.2020)
15. Flask или Django? Выбираем Python вебфреймворк [Электронный ресурс]. Python 3. [Режим доступа] <https://python-scripts.com/flask-or-django#built-with-flask> (Дата обращения: 11.05.2020)
16. React vs Angular vs Vue.js — What to choose in 2020? (updated in 2020) [Электронный ресурс]. Medium. 2018. [Режим доступа] <https://medium.com/techmagic/reactjs-vs-angular5-vs-vue-js-what-to-choose-in-2018-b91e028fa91d> (Дата обращения: 11.05.2020)
17. Shaumik Daityari. Angular vs React vs Vue: Which Framework to Choose in 2020 [Электронный ресурс] / Daityari Shaumik // codeinwp. 2020. [Режим доступа] <https://www.codeinwp.com/blog/angular-vs-vue-vs-react/> (Дата обращения: 11.05.2020)
18. Плюсы и минусы Bootstrap [Электронный ресурс]. Community by Timeweb. 2018 [Режим доступа] <https://depix.ru/articles/bootstrap_preimuschestva_i_nedostatki> (Дата обращения: 11.05.2020)
19. Фреймворк CodeIgniter [Электронный ресурс]. Unetway. [Режим доступа] <http://unetway.com/blog/codeIgniter-framework-review/> (Дата обращения: 12.05.2020)
20. Какие плюсы и минусы WordPress? [Электронный ресурс]. TemplateMonster. [Режим доступа] <https://www.templatemonster.com/ru/faq/plus-i-minus-wordpress/> (Дата обращения: 12.05.2020)
21. ГОСТ Р ИСО 9241-151-2014 Эргономика взаимодействия человек-система. Часть 151. Руководство по проектированию пользовательских интерфейсов сети Интернет (Переиздание).
22. Wikipedia – BootstrapCDN [Электронный ресурс]. Режим доступа: <https://en.wikipedia.org/wiki/BootstrapCDN> (Дата обращения: 13.05.2020)
23. Heroku [Электронный ресурс]. Национальная библиотека им. Н. Э. Баумана. [Режим доступа] <https://ru.bmstu.wiki/Heroku#.D0.9E.D0.B1.D0.B7.D0.BE.D1.80_.D0.BF.D0.BB.D0.B0.D1.82.D1.84.D0.BE.D1.80.D0.BC.D1.8B> (Дата обращения: 15.05.2020)
24. Маленькие хитрости с Heroku [Электронный ресурс]. JavaRush [Режим доступа] <https://javarush.ru/groups/posts/1987-malenjhkie-khitrosti-s-heroku> (Дата обращения: 15.05.2020).