**Министерство образования и науки Российской Федерации**

Федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

|  |  |
| --- | --- |
| **Институт** | *Кибернетики* |
| **Направление** | *Информационные системы и технологии* |
| **Кафедра** | *Информационных систем и технологий* |
|  |  |

**ТВОРЧЕСКИЙ ПРОЕКТ**

|  |
| --- |
| Разработка Web-сайта с применением современных Web-технологий |
|  |
| *Название творческого проекта* |

**Студенты:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *8И52* |  | *Смирнов Павел Олегович* |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| *группа* |  | *ФИО* |  | *подпись* |

**Руководитель:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Шерстнев Владислав Станиславович,  доцент кафедры ИСТ ИК ТПУ |  |  |  |  |  |  |
| *ФИО должность,*  *уч. степень, звание* |  | оценка |  | *дата* |  | *подпись* |

**Итоговая оценка**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Комиссия**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *ФИО должность, уч. степень, звание* |  | *дата* |  | *подпись* |
|  |  |  |  |  |
|  |  |  |  |  |

Томск 2017 г.

Оглавление

[Введение 3](#_Toc483159405)

[Цель и постановка задачи 3](#_Toc483159406)

[1. Структура сайта 4](#_Toc483159407)

[2. Языки программирования 5](#_Toc483159408)

[2.1 Язык гипертекстовой разметки (HTML) 5](#_Toc483159409)

[2.2 Таблиц каскадных стилей (CSS) 7](#_Toc483159410)

[2.3 JavaScript 8](#_Toc483159411)

[2.4 PHP 9](#_Toc483159412)

[3. Работа с базой данных 10](#_Toc483159413)

[3.1 Интерфейс веб-приложения phpMyAdmin 10](#_Toc483159414)

[3.2 Описание таблиц 12](#_Toc483159415)

[4. Работа с FTP-клиентом 14](#_Toc483159416)

[5. Подключение CMS 15](#_Toc483159417)

[5.1 Интерфейс CMS WordPress 16](#_Toc483159418)

[6. Работа с веб-сайтом 19](#_Toc483159419)

[Заключение 24](#_Toc483159420)

[Список литературы 25](#_Toc483159421)

Введение

Интернет развивается довольно быстро. Интернет превращается из большой игрушки для интеллектуалов в полноценный источник разнообразной полезной информации для любой категории пользователей.

Каждый из нас в настоящее время является пользователем сети и каждый может внести свой вклад в развитие Интернета. Так, например, любой пользователь может создать свой веб-сайт и разместить его в Интернете. Но стоит понимать, что в общем для создания веб-сайта требуется задействовать множество различных web–технологий, поэтому с процесса разработки и до его размещения происходит множество этапов.

На сегодняшний день существует множество разных языков программирования и систем управления для создания и работы разнотипных веб-сайтов и веб-приложений. Важно определиться с технологиями, которые будут применяться в ходе работы.

Цель и постановка задачи

Создать веб-сайт, который будет предназначен для хранения и быстрого предоставления информации, касательно внутри учебных дел группы. Так же веб-сайт будет объединять информацию в сфере информационных технологий.

Получение знаний и навыков в сфере Web-программирования.

Были поставлены следующие задачи:

* изучение методического материала необходимого для создания веб-сайта;
* изучение необходимых языков программирования;
* программная реализация функциональной части веб-сайта;
* проектирование и создание базы данных для работы с файловой системой на сервере и хранения необходимой информации;

1. Структура сайта

Любой сайт, если не с самого начала, то со временем, становится все более сложным, увеличивается число страниц и усложняются их взаимосвязи. Сайт уже не укладывается в одну или несколько равнозначных страничек, а, следовательно, приобретает некую внутреннюю структуру. Под **структурой сайта** обычно понимают либо иерархическую (в виде дерева) организацию его страниц, либо блок-схему переходов со странички на страничку по имеющимся ссылкам. Соответственно, можно различать два вида структуры сайта - **файловую структуру** и **гипертекстовую структуру**.

Файлы, содержащие страницы сайта, размещаются в иерархической структуре каталогов - так с ними удобнее работать, проще находить и легче добавлять новые страницы. Это и есть **файловая структура сайта**. Целесообразно организовывать эту структуру в соответствии с логикой организации разделов сайта: первый уровень каталогов должен соответствовать основным разделам сайта, на втором уровне размещаются каталоги, соответствующие подразделам, и так далее. Файловая структура обычно изображается в виде дерева или в виде структуры разделов.

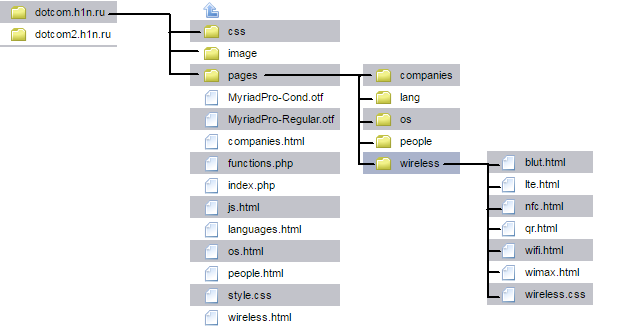


Рисунок 1 – Фрагмент структуры сайта

В файловой структуре могут присутствовать каталоги и подкаталоги, которые не соответствуют каким-либо разделам сайта и либо вообще не отображаются на страницах сайта, либо присутствуют только в служебных ссылках. Например, подкаталог с файлами изображений и видео или подкаталог с файлами программ. В общем, при планировании структуры сайта нужно продумать, где будут храниться все необходимые файлы сайта.

2. Языки программирования

При выборе идеального языка программирования для разработки сайта, важно, понимать, что идеального языка не существует. Это просто вопрос выбора языка, который наилучшим образом отвечает необходимым потребностям. Перед тем как определиться, какой язык следует использовать, нужно учесть следующие факторы:

* серверная платформа
* серверное программное обеспечение
* база данных, выбранная для серверной части

Ниже приведены языки программирования, выбранные для создания веб-сайта.

2.1 Язык гипертекстовой разметки (HTML)

HTML — теговый язык разметки [документов](https://ru.wikipedia.org/wiki/HTML-%D0%B4%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82). Любой документ на языке HTML представляет собой набор [элементов](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D1%8B_HTML), причём начало и конец каждого элемента обозначается специальными пометками — [тегами](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%B3_(%D1%8F%D0%B7%D1%8B%D0%BA%D0%B8_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8)). Элементы могут быть пустыми, то есть не содержащими никакого текста и других данных. В этом случае обычно не указывается закрывающий тег. Кроме того, элементы могут иметь атрибуты, определяющие какие-либо их свойства. Атрибуты указываются в открывающем теге.

Для начала нужно ознакомиться с основными тегами, которые применяются при создании любого веб-сайта. Подробнее о всех можно узнать в интернете, а сейчас разберём только необходимые теги для нашего сайта и этих знаний будет вполне достаточно для начала. Их нужно уметь видеть и понимать предназначение, тогда не будет проблем с изменениями и дополнениями на странице.

<html> - этот тег стоит в самом начале любой страницы, и объявляет кодировки и саму страницу.   
**</**html**>** - этим тегом обязательно заканчиваются все страницы, точнее обьявляют конец страницы.  
**<**head**> </**head**>** - между этими тегами находится невидимая часть сайта, документация.  
**<**title**><.**title**>** - между этими тегами пишется название вашего сайта.  
**<**body**> </**body **>** - между этими тегами пишется весь контент сайта - наполнение страницы.  
**<**h**> ... </**h**>   <**p**>...</**p**>   <**li**>...</**li**>** - теги для написания заголовков, параграфов, списков для вставки текста на страницу.  
**<**div**></**div**>** - блочный элемент, на котором можно построить весь сайт.  
<img src="адрес" width="ширина" height="высота" alt="название" > - так вставляется изображение.  
<a href="# адрес"> Страница 1 </a> - так делается ссылка на страницу.

Как правило страница строится из трех основных частей:  
1) хедер (шапка сайта) - в котором находятся все кодировки, стили, название страницы, основные логотипы, ключевые слова.  
2) контент - здесь все наполнение, другими словами все что нужно показать в браузере, это тексты, изображения и т.д.  
3) футер (подвал) - здесь обычно счетчики, копирайты, возможно меню, реклама, или что-то другое.

Ниже представлен фрагмент кода, написанного на HTML, необходимый для создания любой страницы.

Необходимый листинг любой страницы

**<! DOCTYPE\_ \_>** обязательная часть  
**<html>** объявление языка написания страницы  
**<head>**  
**<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />** - кодировки сайта  
**<title>**Здесь название страницы, отображаемое в верхнем левом углу браузера**</title>**  
**</head>**  
**<body>**  
--- здесь располагают ХЕДЕР и меню сайта  
-------- дальше контентная часть  
это уже веб-страница которую посмотреть в браузере.  
--- внизу обычно ФУТЕР  
**</body>**  
**</html>** - объявление окончания данной страницы

2.2 Таблицы каскадных стилей (CSS)

CSS используется создателями [веб-страниц](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0) для задания [цветов](https://ru.wikipedia.org/wiki/%D0%A6%D0%B2%D0%B5%D1%82), [шрифтов](https://ru.wikipedia.org/wiki/%D0%A8%D1%80%D0%B8%D1%84%D1%82), расположения отдельных блоков и других аспектов представления внешнего вида этих веб-страниц. Основной целью разработки CSS являлось разделение описания логической структуры веб-страницы (которое производится с помощью [HTML](https://ru.wikipedia.org/wiki/HTML) или других [языков разметки](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D1%80%D0%B0%D0%B7%D0%BC%D0%B5%D1%82%D0%BA%D0%B8)) от описания внешнего вида этой веб-страницы (которое теперь производится с помощью [формального языка](https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) CSS).

Правила CSS пишутся на [формальном языке](https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) CSS и располагаются в таблицах стилей, то есть таблицы стилей содержат в себе правила CSS. Эти таблицы стилей могут располагаться как в самом [веб-документе](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%B4%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82), внешний вид которого они описывают, так и в отдельных [файлах](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB), имеющих [формат](https://ru.wikipedia.org/wiki/%D0%A4%D0%BE%D1%80%D0%BC%D0%B0%D1%82_%D1%84%D0%B0%D0%B9%D0%BB%D0%B0) CSS. (По сути, формат CSS — это обычный [текстовый файл](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB). В файле .css не содержится ничего, кроме перечня правил CSS и комментариев к ним.)

Ниже показан пример подключения файла style.css к сайту.

<link rel="stylesheet" href="style.css" type="text/css" media="screen, projection" />

<div id="primer" align="center" >пример с id</div><br>

<div class="primer\_2" align="center">пример с class</div>

-- и описание стилей --

#primer { width:400px; height:20px; border:1px solid #0000FF; background:#33FF33;

position:relative; float:none;}

.primer\_2 { width:500px; height:30px; border:1px solid #00FF00; background:#666600;

position:relative; float:none;}

Применение стилей к тегам

2.3 JavaScript

**JavaScript** - предназначен для написания сценариев для активных HTML-страниц. Язык JavaScript не имеет никакого отношения к языку **Java**.

JavaScript не предназначен для создания автономных приложений. Программа на JavaScript встраивается непосредственно в исходный текст HTML-документа и интерпретируется брaузером по мере загрузки этого документа. С помощью JavaScript можно динамически изменять текст загружаемого HTML-документа и реагировать на события, связанные с действиями посетителя или изменениями состояния документа или окна.

Важная особенность JavaScript - объектная ориентированность. Программисту доступны многочисленные объекты, такие, как документы, гиперссылки, формы, фреймы и т.д. Объекты характеризуются описательной информацией (свойствами) и возможными действиями (методами).

Сценарий JavaScript встраивается в HTML-документ с помощью тега **<script>**.

**jQuery** — javascript библиотека (framework), использование которой делает разработку javascript кода намного быстрее, проще и приятнее.

Так с помощью языка программирования JavaScript реализован слайдер. Благодаря ему происходит смена изображений на главной странице веб-сайта.

Листинг слайдера

<script type="text/javascript">

var chet = 1;

window.onload = function () { setTimeout(frout, 5000); }

function frout() {

chet++;

if (chet>3) {chet=0;}

switch(chet) {

case 1:

document.getElementById('img').src = "http://cloudberrystudio.ru/ff/4/rip\_turk1672\_steve\_jobs\_fon\_apple\_stiv\_dzhobs\_1920x1200.jpg";

break;

case 2:

document.getElementById('img').src = "https://tproger2.azureedge.net/wp-content/uploads/2016/02/hello.png";

break;

case 3:

document.getElementById('img').src = "http://www.zastavki.com/pictures/1920x1200/2012/Creative\_Wallpaper\_Information\_Technologies\_034938\_.jpg";

break;

}

setTimeout(frout, 5000);

}

</script>

2.4 PHP

PHP предназначен для программирования на стороне сервера, что означает, что библиотека языка специализирована для задач, которые выполняются неоднократно. PHP также часто взаимодействует с HTML кодом что дает ему огромное преимущество, таким образом позволяя вам смешивать верстку с программированием.

PHP является одним из наиболее широко используемых языков для веб-разработки. PHP - это встраиваемый в HTML скриптовый язык, который используется для того чтобы своевременно сформировать динамические страницы. PHP, позволяет очень быстро и легко расширить веб-приложение.

PHP код может быть непосредственно размещен в HTML коде, без использования внешнего файла.

3. Работа с базой данных

База данных - это хранилище, в которое можно сохранять данные, а позже делать по ним поиск и загружать их. Почти любой интерактивный сайт использует БД.

В нашей базе данных храниться информация о пользователях сайта, написанных ими сообщениях, а также хранимые ими данные.

При разработке веб-сайта использовалась база данных, поддерживающая язык SQL. Для ее создания и управления применялась СУБД – MySQL.

Как правило сам сервер базы не имеет своего интерфейса и каких-то окон, кнопок, чтобы с ним взаимодействовать. Управление базой данных делается с помощью запуска программы-клиента, который подсоединяется к серверу, пересылает ему SQL запросы и выводит полученные ответы. Для этого было задействовано веб-приложение phpMyAdmin, представляющее собой веб-интерфейс для администрирования [СУБД](https://ru.wikipedia.org/wiki/%D0%A1%D0%A3%D0%91%D0%94) [MySQL](https://ru.wikipedia.org/wiki/MySQL). PHPMyAdmin позволяет через [браузер](https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80) и не только осуществлять администрирование сервера MySQL, запускать команды [SQL](https://ru.wikipedia.org/wiki/SQL) и просматривать содержимое таблиц и баз данных.

3.1 Интерфейс веб-приложения phpMyAdmin

Интерфейс веб-приложения изображен на рисунке 2.

В правом углу находятся все созданные базы данных, прикрепленные к веб-сайту (рисунок 2.1). Посредине – доступная информация по выбранной таблице (рисунок 2.2). В верхнем углу – панель редактирования таблицы (рисунок 2.3).

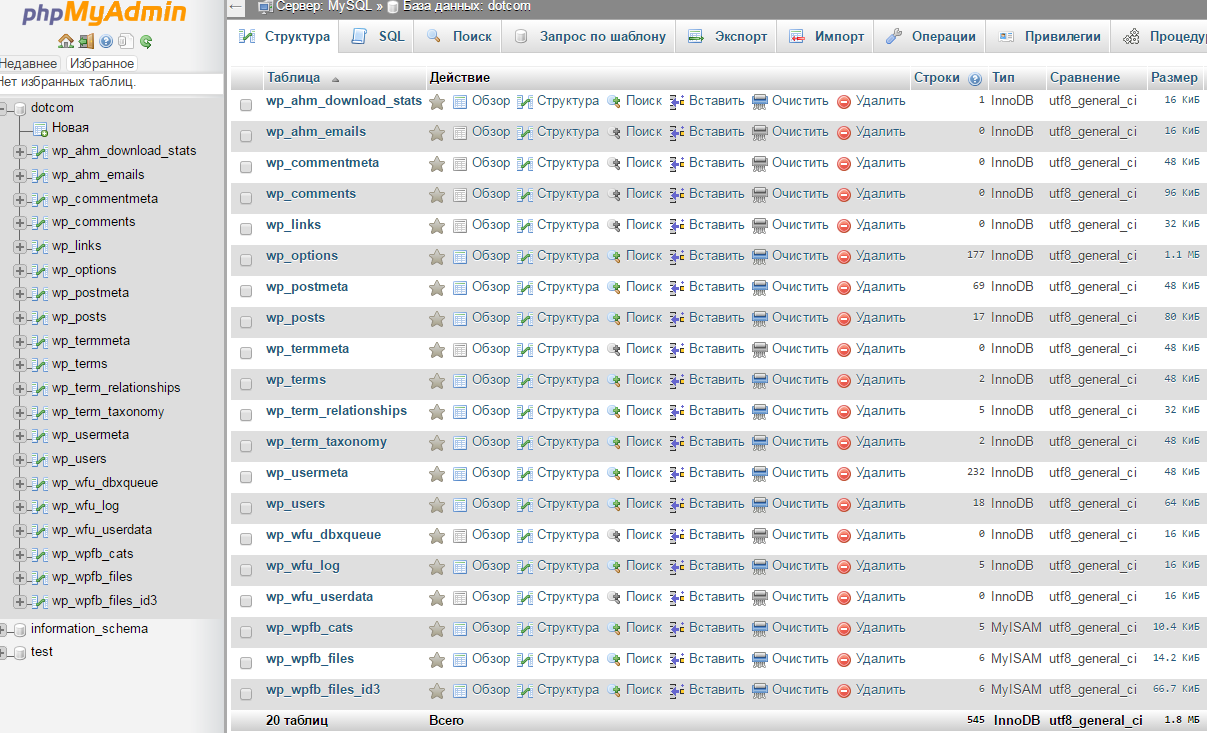
****

Рисунок 2 - Интерфейс phpMyAdmin

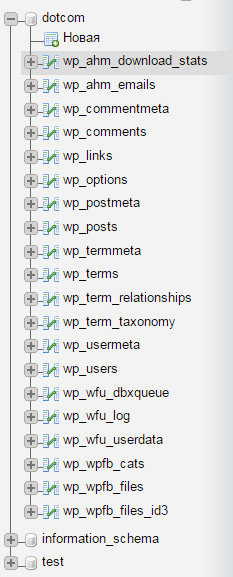


Рисунок 2.1 – Прикрепленные базы данных

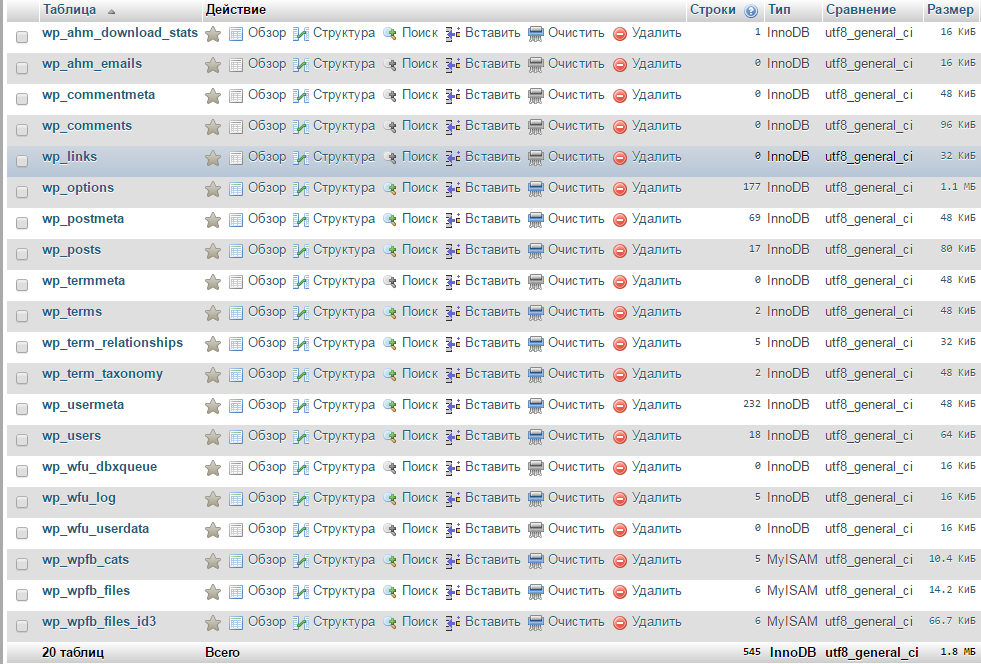


Рисунок 2.2 – Информация по таблицам

C:\Users\User\Desktop\творческий картинки\Снимок12.PNG

Рисунок 2.3 – Панель редактирования таблицы

3.2 Описание таблиц

База данных хранит данные в таблицах. Таблицы создает разработчик, и обычно каждая из них предназначается для своей сущности - например, таблица со списком пользователей, таблица тем на форуме, таблица сообщений на форуме. Таблица состоит из колонок, каждая из которых имеет определенных тип (число, строка). Таблица для хранения информации о пользователях веб-сайта выглядит так:

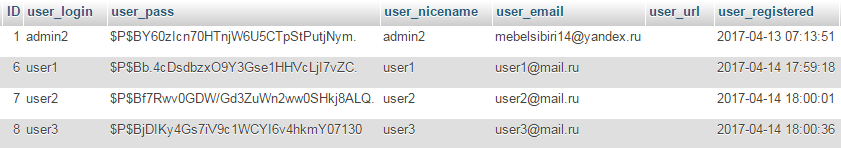


Рисунок 3 – Таблица пользователей

При регистрации скрипт добавляет в нее информацию о новом пользователе, а при логине - проверяет введенные email и пароль.

Колонки таблицы пользователей:

* ID – уникальный идентификатор в качестве ключа
* User\_login – логин пользователя
* User\_pass – пароль пользователя
* User\_email – почтовый адрес пользователя
* User\_registred – дата регистрации пользователя

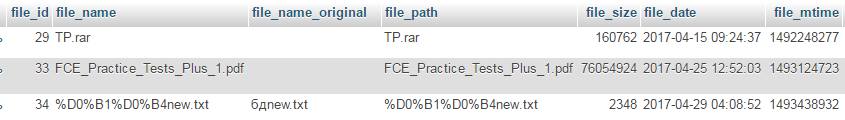


Рисунок 4 – Таблица загруженных файлов

Колонки таблицы загруженных файлов:

* File\_id – уникальный идентификатор в качестве ключа
* File\_name – название файла
* File\_name\_original – описание файла
* File\_size – размер файла в битах
* File\_date – дата загрузки файла

4. Работа с FTP-клиентом

 File Transfer Protocol — протокол передачи файлов, который применяется для обмена файлами по TCP/IP сетям между двумя компьютерами (клиент и сервер).

Данный протокол применяет различные сетевые соединения для передачи команд и файлов между клиентом и сервером. FTP сервер, представляет собой компьютер с установленным на него специальным программным обеспечением и ожидающим внешнего подключения от других компьютеров.

FTP клиент является программой, которая делает попытку соединится с серверным компьютером. После успешного подключения к FTP серверу, можно совершать всевозможные операции над данными располагающимися на нем, в частности, просмотреть содержимое каталогов, загружать и скачивать файлы с FTP сервера, переименовывать, назначать права доступа, удалять файлы с сервера и прочее.

В ходе работы был выбран ftp-клиент FileZilla, т.к. он легок в освоении, обладает кроссплатформенностью, поддерживает множество языков, а также гибок в настройке.

На рисунке 5 изображен интерфейс программы.

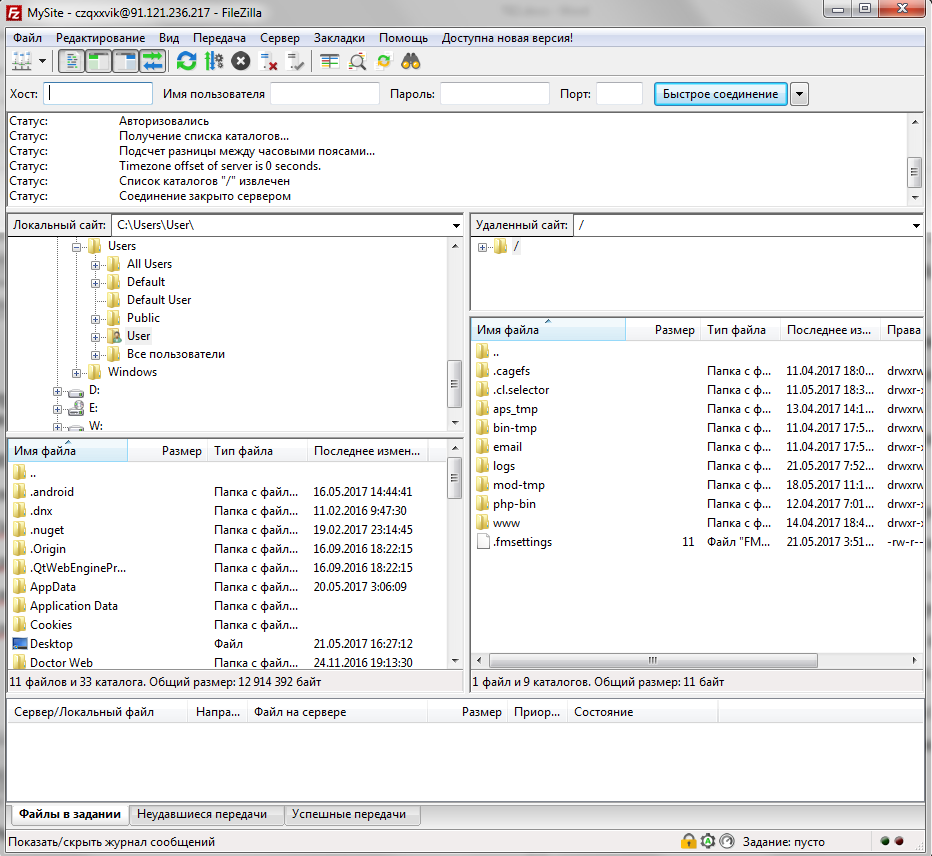


Рисунок 5 – Интерфейс программы FileZilla

5. Подключение CMS

Система управления содержимым (Content management system, CMS) – информационная система, которая используется для обеспечения и организации создания, редактирования и управления содержимым на сайте.

В система управления содержимым хранится множество различных данных: документы, пользовательские файлы, статьи и так далее. Так же в ней задействован контроль версий, что позволяет вести работу с ней сразу нескольким пользователям.

В ходе работы над сайтом была выбрана CMS WordPress, так как она включает в себя все необходимые для данного веб-сайта функции:

* Предоставление инструментов для создания содержимого и организация совместной работы над ним;
* Управление содержимым: хранение, контроль версий, соблюдение режима доступа, управление потоком документов;
* Публикация содержимого;
* Предоставление информации в виде, удобном для навигации и поиска;
* Гибкость в настройка, а также поддержка установки дополнительных плагинов;

5.1 Интерфейс CMS WordPress

Интерфейс основного меню работы с системой изображен на рисунке 6. Сверху находится строка пользователя, позволяющая редактировать информацию о пользователе (рисунок 6.1). В левой части – панель работы с консолью, с помощью которой можно добиться гибкой настройки веб-страниц, а также в нем находится панель настроек CMS (рисунок 6.2).

На данной панели доступны следующие вкладки:

* Пользователи – в которой доступна информация о пользователях, прошедших регистрацию на веб-сайте, а также присутствует возможность ее редактирования (рисунок 6.3).
* Плагины – где доступна информация о уже установленных плагинах и возможность добавления дополнительной функциональности веб-сайту (рисунок 6.4).
* WP-Filebase – основная панель работы с базой данных веб-сайта. В ней доступно управление файлами, хранимые пользователями и основная информация о них (рисунок 6.5).

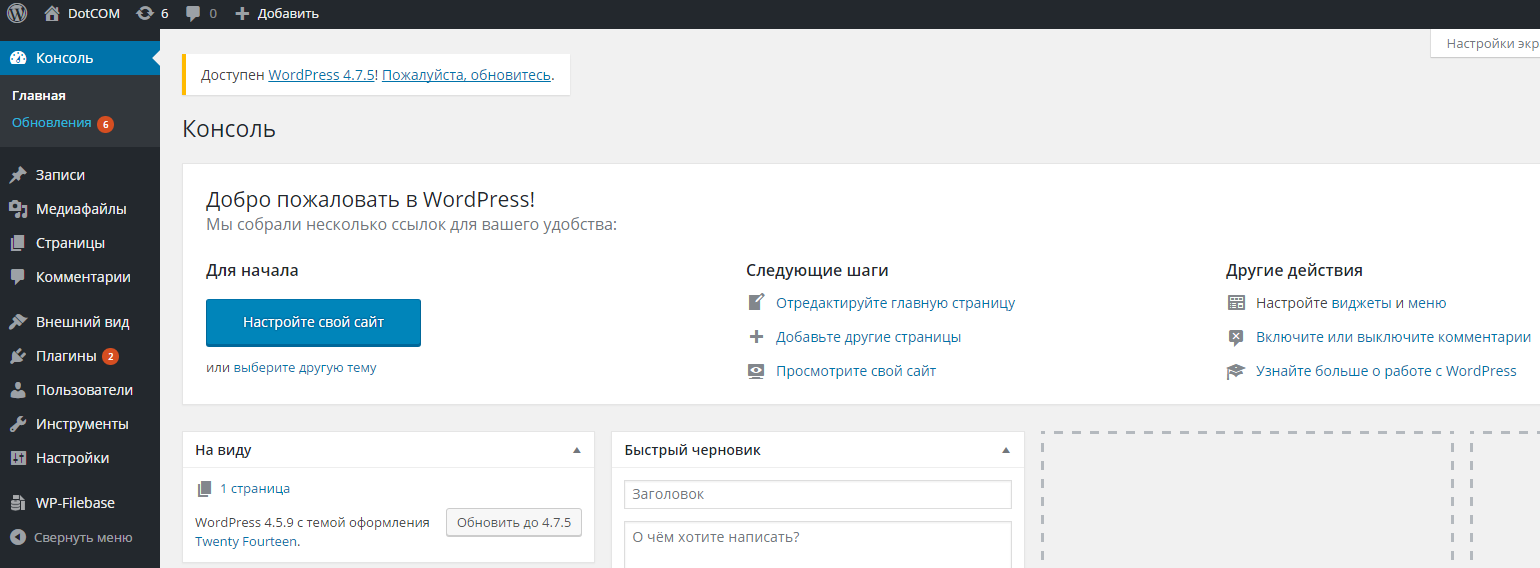


Рисунок 6 – Основное меню CMS

C:\Users\User\Desktop\творческий картинки\Снимок17.PNG

Рисунок 6.1 – Строка пользователя

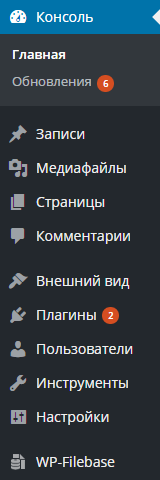


Рисунок 6.2 – Панель работы с консолью

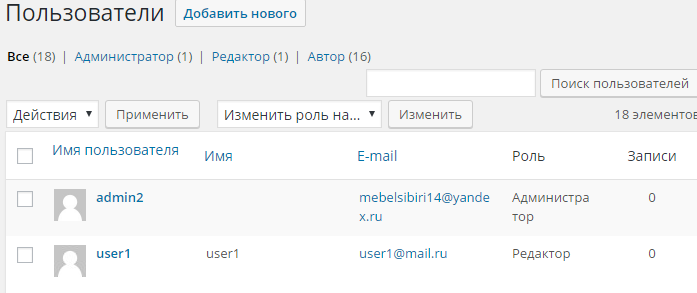


Рисунок 6.3 – Вкладка “Пользователи”

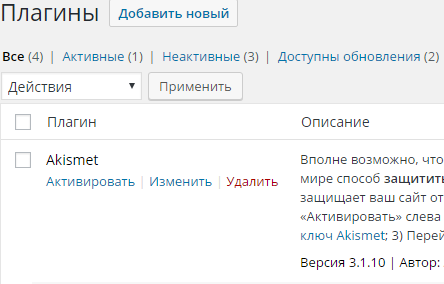


Рисунок 6.4 – Вкладка “Плагины”

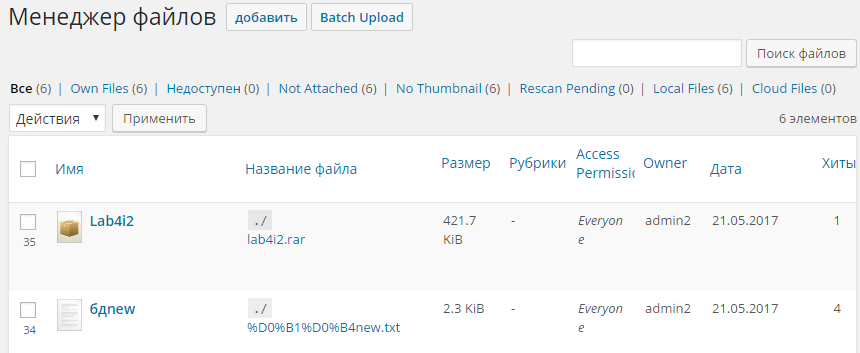


Рисунок 6.5 – Вкладка “WP-Filebase”

6. Работа с веб-сайтом

На рисунке 7 можно видеть главную страницу веб-сайта.

Интерфейс выполнен на интуитивно понятном уровне с учётом минимизации объектов на экране. Основная область контентного меню заполнена слайд-баром, где происходит смена слайдов.

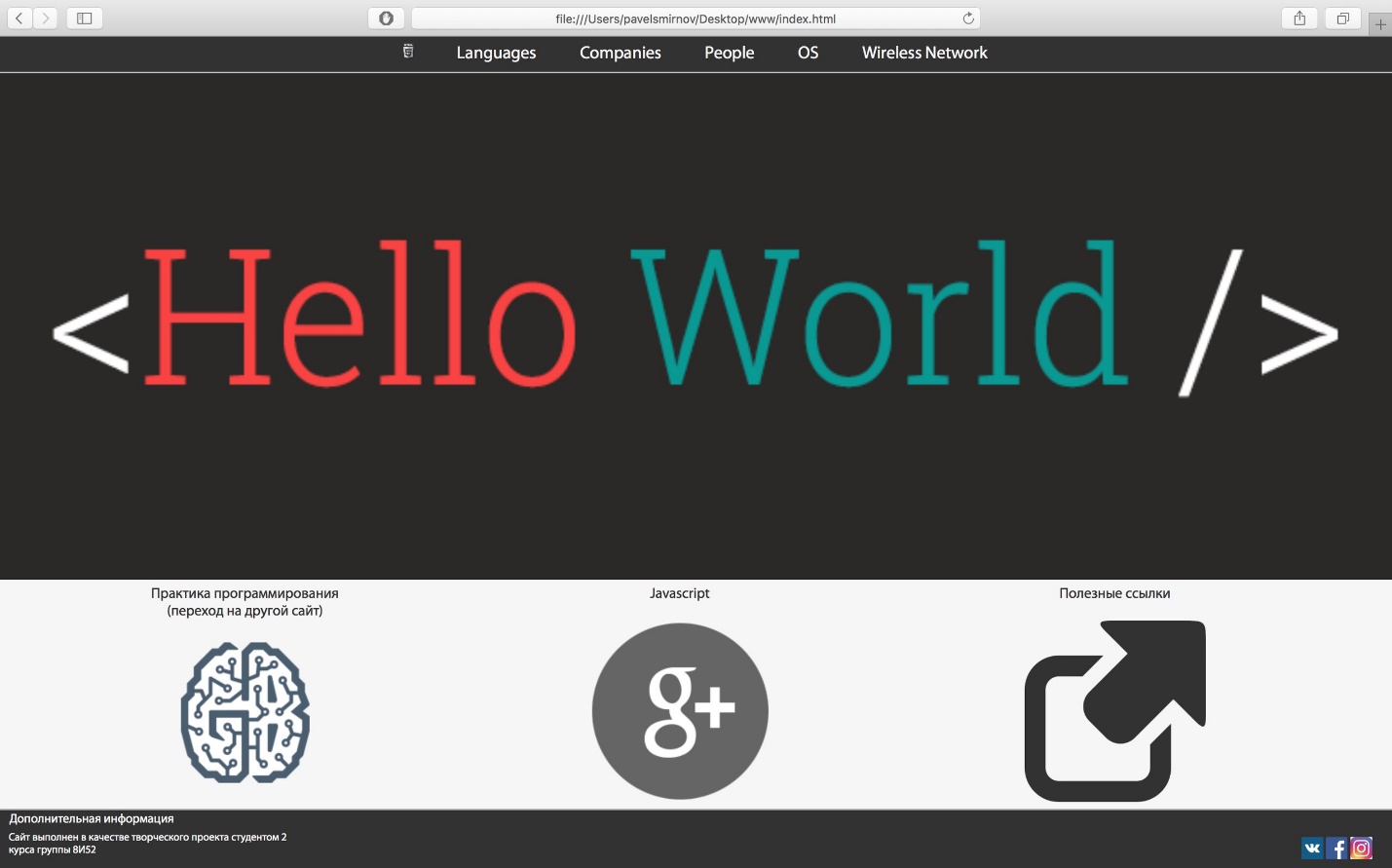


Рисунок 7 - Главная страница сайта

Сверху находится панель меню, которая служит для навигации и переходу по страницам веб-сайта, при клике на определенной вкладке появляется страница с панелью выбора интересующей информации.

Ну и на рисунке 8 изображена страница, содержащая основную информацию по данной теме.

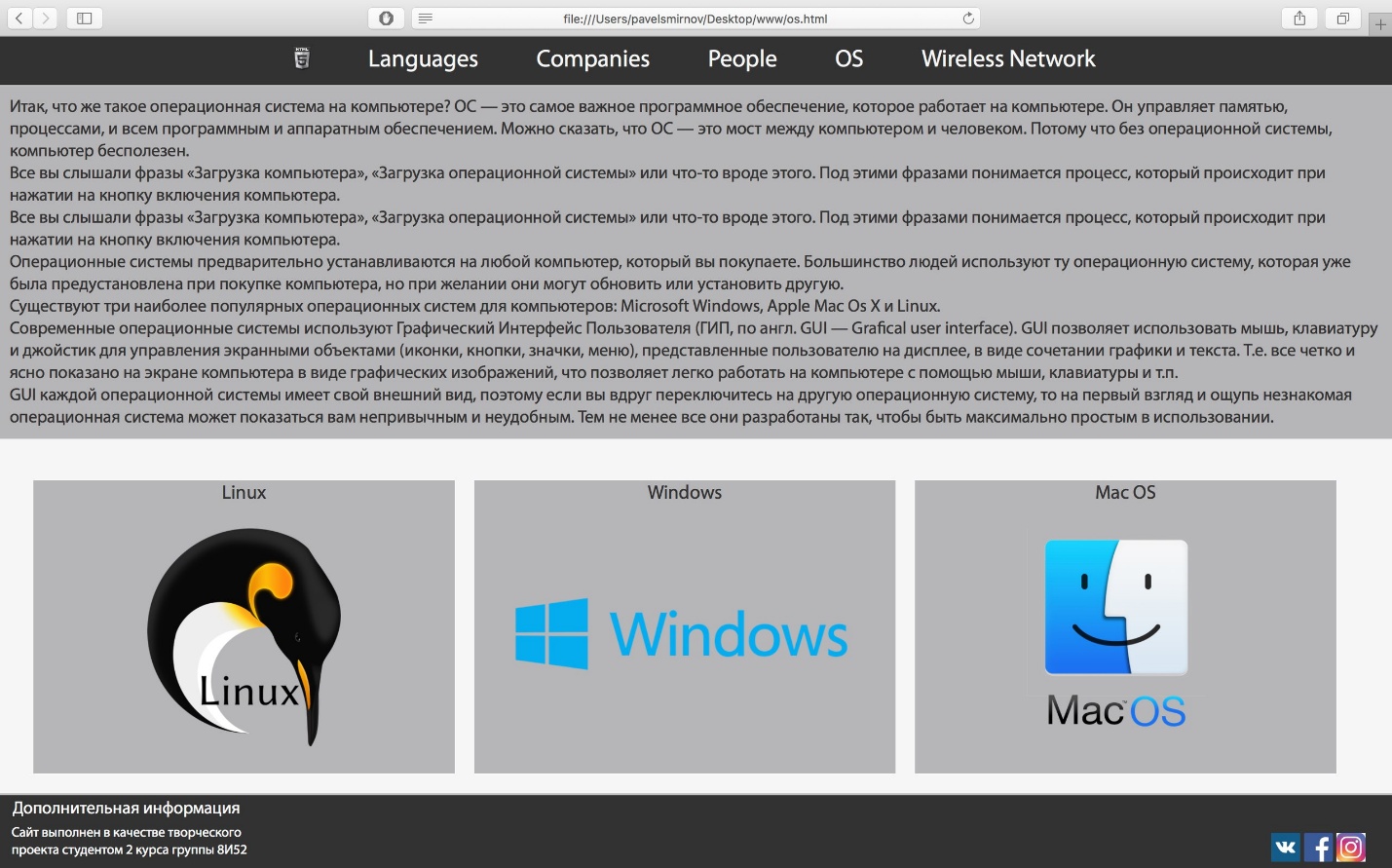


Рисунок 8 - Панель выбора

Была проведена частичная адаптивность сайта для диагонали телефона iPhone 5. На рисунке 9 изображено как выглядит веб-страница, сделанная с данного телефона в вертикальном положении экрана.

На главной странице веб-сайта находится вкладка “Облако”, при нажатии которой происходит переход на страницу работы с файловой системой сайта, где каждый пользователь может добавлять, хранить и редактировать необходимые файлы.

На рисунке 10 изображена страница управления пользовательскими файлами.



Рисунок 9 - Скриншот сайта с телефона

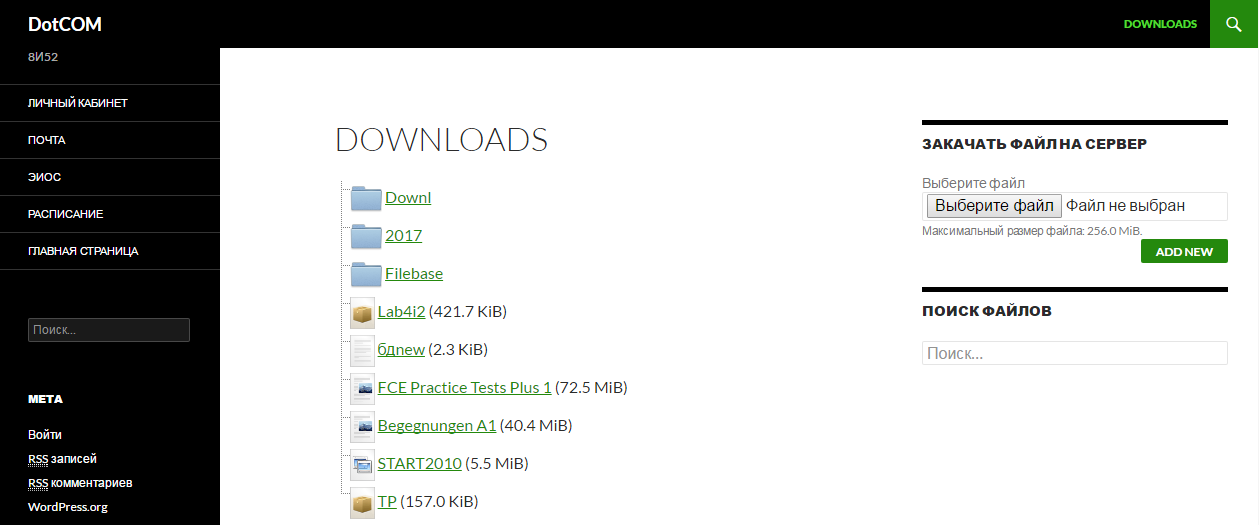


Рисунок 10 – Страница управления файлами

Слева находится панель с вкладками (рисунок 10.1), при нажатие которых пользователь переходит на соответствующие веб-страницы.

Для работы с файлами пользователю необходимо авторизоваться, иначе функционал сайта недоступен. На рисунке 10.2 и 10.3 изображен процесс авторизации пользователя.

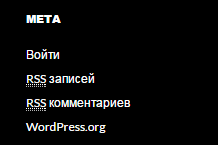
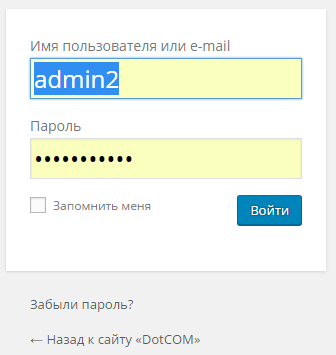
 

Рисунок 10.2 – Панель входа Рисунок 10.3 – Панель входа

На веб-сайте реализована система, которая ограничивает права доступа к некоторым функциям. Система управления правами доступа доступна администратору веб-сайта.

После авторизации пользователю доступно вести работу с файловой системой (рисунок 10.4). Функции доступные пользователю с обычными правами:

* Добавление файла
* Создание папок и подкаталогов
* Редактирование файла
* Удаление файла

Добавление файла на сервер происходит по нажатию “Выберите файл” (рисунок 10.5).

Также любой пользователь может найти доступные ему файлы в строке “Поиск”. Загрузка происходит при нажатии на соответствующий файл.

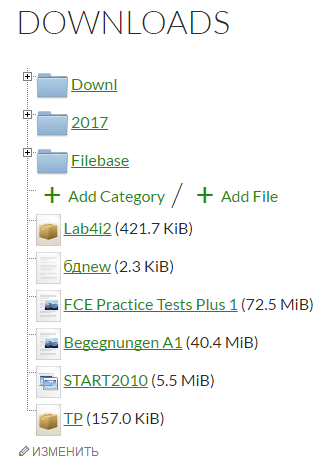


Рисунок 10.4 – Панель управления файлами

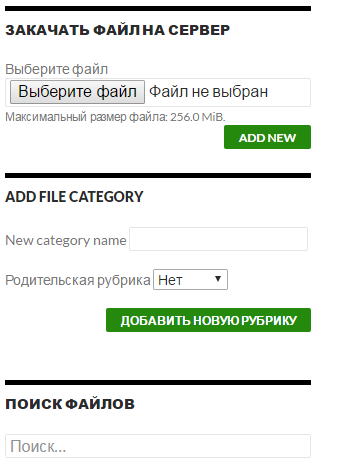


Рисунок 10.5 – Панель управления файлами

Заключение

В результате выполнения проекта были получены опыт и навыки работы с web-технологиями. Были выполнены все поставленные задачи.

Была изучен методический материал по необходимый для создания веб-сайта. В процессе работы над творческим проектом, были изучены основы языков программирования HTML, CSS, JavaScript, PHP. Для работы с базой данных, работающий с СУБД MySQL было изучено веб-приложение phpMyAdmin, а для более удобной передачи файлов между клиентом и сервером был использован FTP-клиент FileZilla.

Для реализации хранения данных пользователей и управления ими, была установлена CMS WordPress, которая с помощью плагина реализовала необходимые задачи с работой файловой системы веб-сайта.

Созданный веб-сайт является конечным продуктом, нужной информацией и размещен на хостинге, в настоящий момент работоспособен.

Список литературы

1. Бланшет Ж., Саммерфилд М. Работа с базой данных под управлением MySQL. — М.: КУДИЦ-ПРЕСС, 2008. – 736 с.
2. Липпман С. Б. Основы программирования на PHP. – М.: Вильямс, 2014. — 1118 с.
3. <http://htmlbook.ru/>
4. <https://learn.javascript.ru/>
5. <https://jquery.com/>
6. <https://ru.wordpress.org/>
7. <http://filezilla.ru/>