МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Кафедра прикладной математики**

**КУРСОВАЯ РАБОТА**

**РАЗРАБОТКА WEB-РЕСУРСА ПО ОРГАНИЗАЦИИ ПРАЗДНИЧНЫХ МЕРОПРИЯТИЙ**

Работу выполнила \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Т.Е. Кулиш

(подпись, дата) (инициалы,фамилия)

Факультет компьютерных технологий и прикладной математики курс 3

Направление 02.03.03 Математическое обеспечение и администрирование информационных систем

Научный руководитель,

преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.Х. Арутюнян

(подпись, дата) (инициалы, фамилия)

Нормоконтролер,

магистр педагогики \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Г.В. Калайдина

(подпись, дата) (инициалы, фамилия)

Краснодар 2017

СОДЕРЖАНИЕ

Введение 3

1 Основы Web-программирования 5

1.1 Описание предметной области 5

1.2 Актуальность разработки web-ресурса 8

2 Разработка web-ресурса 10

2.1 Техническая часть написания web-страницы 10

2.2 Реализация собственного входа 14

2.3 Реализация собственной регистрации на сайте 17

2.4 Оформление страниц 20

Заключение 22

Список использованных источников 23

ПРИЛОЖЕНИЕ А Login\_page.php 24

ПРИЛОЖЕНИЕ Б Authorization.php 25

ПРИЛОЖЕНИЕ B Гиперссылки с использованием изображения 30

ВВЕДЕНИЕ

Интернет стал наиболее эффективным средством продвижения товаров и услуг, удовлетворяющих потребности интернет-пользователя, а также является одним из важных элементов современной цивилизации. Поэтому создание web-ресурсов предоставляет отличную возможность заявить о своем предприятии.

Web-сайт – совокупность веб-страниц, объединенных одним доменным именем, имеющих общую тематику, дизайн, а также взаимосвязанных при помощи гиперссылок. Страницы представлены в текстовом формате на языках программирования: HTML, JAVA, PHP, CSS, MYSQL, которые позволяют изменять интерфейс.

Web-сервер – это компьютер, который предоставляет в сеть один или множество web-сайтов. Хостинг – услуга, которая предоставляет ресурсы для размещения информации на сервере, постоянно находящемся в сети. В пакет его обслуживания входит предоставление места для баз данных, файлового хранилища на специально выделенном файл-сервере, поддержание функционирования соответствующих сервисов и т.п. Адрес ресурса, который вводится пользователем при обращении к сайту называется доменным именем(домен).

Рассматривая вопрос о том, какие преимущества имеют онлайн порталы, стоит сказать, что создание сайтов позволяет добиться ряда важнейших факторов:

* Возможность быстрой подачи информации;
* Проведение онлайн опросов;
* Простой способ поддержания связи через интернет с пользователями других регионов;
* Привлечение и обеспечение обратной связи с клиентами;

С появлением глобальной сети каждый человек получил интерактивный инструмент, позволяющий сообщить миру об услугах и товарах компании, что ярко подчеркивает актуальность создания web-ресурсов.

Цель курсовой работы – разработки web-ресурсов по организации праздничных мероприятий.

Задача курсовой работы – изучение современных технологий разработки web-ресурсов.

Курсовая работа состоит из двух глав, введения, заключения и списка используемой литературы.

В первой главе представлено описания предметной области, которые используются для размещения web-ресурса в сеть интернет, а также актуальность разработки web-ресурса.

Вторая глава повествует о технической разработки web-сайта. В данной главе представлены основные аспекты взаимодействия установленного сервера с разработанным web-ресурсом, который предоставляет информацию по организации праздничных мероприятий. Можно ознакомиться с реализацией собственного входа и формой регистрации. А также в данной главе представлено оформление страниц, что показывает уникальные возможности данного web-сайта.

Разработанный web-ресурс предоставляет услуги по организации праздничных мероприятий, доступные для любого возраста и слоя населения. Удобное оформление и креативный дизайн привлекает взгляды интернет-пользователей. Ознакомившись с данным проектом, можно получить более подробную информацию по реализации сайта.

В заключении подведены основные итоги написания курсовой работы, а также представлен результат, с которым можно ознакомится.

1 Основы Web-программирования

1.1 Описание предметной области

Стремительное развитие Интернета за последних несколько лет существенно увеличило количество новых онлайн-сервисов. Такое оперативное продвижение в крупную глобальную сеть с большим объемом различных сервисов, технологических решений начиналось с первоначальных характеристик и основных сервисов, которые были доступны пользователям (почта и web-ресурсы).

Услуги, которые предоставляет интернет в современной деятельности, открывают огромные возможности для решения технологических вопросов, вне зависимости от того, в каком уголке мира находятся интернет-пользователи. Доступная интернет-информация располагается на компьютерах (web-серверах), значительная часть которой организована в виде web-сайтов.

Говоря о создании сайтов, затрагивается тема web-программирования. Web-сайт объединяются под одним адресом (доменным именем или IP-адресом) совокупность документов частного лица или организации. По умолчанию подразумевается, что сайт располагается в сети Интернет. В совокупности все web-ресурсы составляют Всемирную паутину. Для прямого доступа клиентов к web-сайтам на серверах был специально разработан протокол HTTP. Web-сайты иначе называют интернет-представительством человека или организации. Когда говорят «своя страничка в Интернет», то подразумевается целый web-сайт или личная страница в составе чужого сайта. Кроме web-сайтов в сети Интернет так же доступны WAP-сайты для мобильных телефонов. Из-за проблем связанных с безопасностью протоколов HTTP и WAP, появилось защищенное HTTPS соединение. Данное соединение использует протокол, похожий HTTP, но обладает высокой степенью безопасности. Такой тип соединения применяется в «личных кабинетах» пользователей интернет-магазинов. Данный метод соединения обретает все большую популярность.

Изначально web-сайты представляли собой совокупности статических документов, имеющий красивый дизайн, интересное и хорошо организованное содержание, удобное и легкое применение. В настоящее время большинству из них свойственна динамичность и интерактивность. Для того, чтобы усовершенствовать страницу, сделать способной реагировать на действия пользователя, уметь собирать и обрабатывать информацию от посетителей сайта необходимо использование программ, так называемые web-разработчики или скрипты. С помощью скриптов можно добавить на сайт голосование, формы обратной связи, поиск данных. Для таких случаев специалисты используют термин web-приложение – готовый программный комплекс для решения задач web-сайта.

Разработчики web-программирования для создания сайтов применяют огромный спектр реализации web-ресурсов. При всем разнообразии средств и языков программирования, которые могут быть использованы в web-строительстве, скрипты разделяются на две группы:

* работа происходит на сервере (т.е. на компьютере, где располагается непосредственно сайт);
* работа происходит на стороне клиента (т.е. на компьютере пользователя, который просматривает данный сайт).

Один и тот же сайт может быть доступен по разным адресам и хранится на разных серверах. Копия оригинального сайта в таком случае называется зеркалом. Существует так же понятие оффлайновая версия сайта – это копия сайта, которая может быть просмотрена на любом компьютере без подключения к компьютерной сети и использования серверного программного обеспечения. При необходимости сбора и хранения информации на сервере используются серверные скрипты. Так, например, осуществляется организация интернет-магазинов, где происходит прием и сохранение отправленных данных пользователем. Скрипты, работающие на стороне клиента, позволяют реагировать на действия пользователя, когда он просматривает уже загруженную в память своего компьютера страницу, изменять её вид и содержимое, не загружая её с сервера снова. Очень часто для обеспечения выполнения некой задачи используются оба вида скриптов.

На момент появления и начала развития интернета уже существовало большое количество популярных и развитых языков программирования , но для создания web-ресурсов необходимы были новые функции, упрощающие работу с типичными данными и элементами web-страниц, а также требовалось специально ограничить функциональность средств программирования для того, чтобы не было возможности с помощью написанных определенных скриптов нанести вред как серверам, на которых расположены сайты, так и компьютерам пользователей. Для решения поставленных задач web-программирования стали усовершенствоваться языки и средства их применения.

После завершения разработки web-ресурса встает вопрос о выборе хостинга, т.е. площадка для размещения сайта в сети интернет. Безусловно, наш веб-сайт может быть размещен на домашнем компьютере, что будет весьма неудобным и невыгодным решением, поскольку необходим постоянно подключенный интернет. По этим причинам, большинство пользователей предпочитают расположение на серверах хостинговых компаний, которые располагаются на специально оборудованных площадках под круглосуточным присмотром специалистов, подключены к мощным каналам, пропускающим огромное количество трафика. Хостинговые компании предоставляют в аренду целый сервер, а также могут и размещать его на своей площадке. Но более востребованной и выгодной услугой является виртуальный хостинг. Большинство сайтов не нуждаются в использование целого сервера, поэтому выделяют определенную часть свободного места на сервере, которая позволит свободно распоряжаться запуском скриптов, загрузкой html-страницы.

В основном в Интернете одному web-сайту соответствует одно доменное имя. Именно по доменным именам сайты идентифицируются в глобальной сети. Возможны иные варианты: один сайт на нескольких доменах или несколько сайтов под одним доменом. Обычно несколько доменов используют крупные сайты (web-порталы), чтобы логически отделить разные виды предоставляемых услуг.

1.2 Актуальность разработки web-ресурса

Безусловно, в настоящее время web-ресурсы являются электронными визитками для любого бизнес-процесса, которые в первую очередь служат для рекламы предприятия. Также они предоставляют возможность проинформировать клиента о бизнес-идеях, услугах, которые оказывает компания, и описывает продукцию, которую изготовляют. Ресурс в интернете будет доступен всем желающим в любое время, что избавит компанию от многих лишних вопросов и заранее простимулирует часть посетителей к сотрудничеству с компанией. Web-сайты позволяют потенциальным клиентам задавать вопросы об интересующих товарах, оставлять свои комментарии и пожелания, по улучшению производительности. Благодаря этому, фирма переходит на новый этап своего развития, что помогает увеличить сегмент потребителей на рынке.

Но для создания эффективного web-ресурса необходимо тщательно продумать каким же должен быть дизайн сайта, его меню, проанализировать конкурентов, их недостатки и преимущества, а также какая информация интересовала бы клиентов в первую очередь. Дизайн сайта должен соответствовать теме сайта и информации, которая будет размещена на нем.

Web-сайт – это лицо предприятия, поэтому он должен быть оригинальным, приятным в использовании и легко доступным. Информация, которая размещается на таком сайте, должна быть всегда достоверной и обновляться.

После того как сайт предприятия уже создан и функционирует, он должен поддерживаться и продвигаться. Для этой работы существуют разного рода аналитики. Их задача следить за сайтом, продвигать его и оптимизировать, анализируя качество трафика, работу поисковых машин и поведения клиентов. Аналитики должны отслеживать маршруты пользователей, их потребности, количество индексированных страниц, наличие сайта в каталогах, наличность как внешних, так и внутренних силок на данный сайт. С помощью такого анализа можно исправить ошибки, максимально улучшить качество данного проекта и закрепить позиции среди конкурентов. Каждый сайт предприятия содержит разные критические данные, которые нужно защищать от взлома, поэтому сайт должен быть надежным.

Благодаря электронной коммерции доступ к web-сайту открыт для уже существующих и потенциальных клиентов круглосуточно, что позволит не только совершать покупки в любое время, но и увеличит количество продаж. Статистика проведенных опросов свидетельствует о том, что такой способ размещения информации о малом бизнесе способствует большему доверию к компании. Кроме того, отзывы покупателей на популярных специализированных сайтах укрепляет значение бренда, что положительно влияет на репутацию компании. С выходом в Интернет ваша компания сразу же расширит свой рынок и сможет обратиться к по-настоящему безграничной аудитории. При этом будет возможность сделать это с меньшими расходами, чем при использовании традиционных средств.

Чтобы клиенты ценили вашу компанию, крайне важно оперативно отвечать на их вопросы и быстро решать возникающие проблемы. Покупатели должны иметь возможность удобно находить интересующую их информацию о вашем бизнесе, получить ответы на часто задаваемые вопросы или обращаться в службу поддержки клиентов в любое удобное им время. Присутствие в сети позволит вам эффективно поддерживать связь с существующими клиентами.

2 Разработка web-ресурса

2.1 Техническая часть написания web-страницы

Знакомство с разработкой web-ресурса следует начать с языков программирования и баз данных. Основный язык программирования, который реализован в нашей программе по созданию интернет-магазина, является PHP (Hypertext Preprocessor), сконструированный для генерирования HTML-страниц на web-сервере и работы с базами данных. В данный момент его активно поддерживают значительное большинство хостинг-провайдеров.

Отличительной особенностью от языка программирования JavaScript является то, что выполнение работы PHP-скриптов происходит непосредственно на сервере. Сгенерированные HTML-страницы отправляются клиенту, который, в свою очередь, получает не только результат выполнения, но и возможность извлечь код вашей программы. Хотя PHP предназначен для работы в среде web-серверов, область его применения не ограничивается только этим.

Далее познакомимся с CSS (Cascading Style Sheets, каскадные таблицы стилей), который определяет отображение HTML-страниц. Стилем или CSS называется набор параметров формирования, который применяется к элементам кода для изменения их внешнего вида. Появление CSS стало революцией в мире web-дизайна. Рассмотрим его преимущества:

* управление отображением множества документов с помощью одной таблицы стилей;
* более точный контроль над внешним видом страниц;
* различные представления для разных носителей информации (экран, печать, и т. д.);
* сложная и проработанная техника дизайна.

Для создателей сайта легкостью и доступностью является то, что цвет, размер текста и многие другие параметры хранятся в определенном месте и приспосабливаются к любому тегу.

Для того, чтобы web-сайт стал доступен многочисленной публике сети Интернет, необходимо сделать следующие действия:

* присвоить ему индивидуальное имя – домен;
* разместить сайт на сервере;
* воспользоваться услугой хостинга.

Как было сказано ранее, web-сервер – это компьютер, который хранит ресурсы сайта и доставляет их на устройство конечного пользователя (веб-браузер и т.д.). Обычно подключен к сети Интернет и является доступным через доменное имя.

С точки зрения программного обеспечения, web-сервер включает в себя некоторые вещи, которые контролируют доступ web-пользователей к размещенным на сервере файлам (HTTP сервера). HTTP сервер – это часть программного обеспечения, которая принимает URL адреса и протоколы, которые использует браузер для просмотра web-страниц.

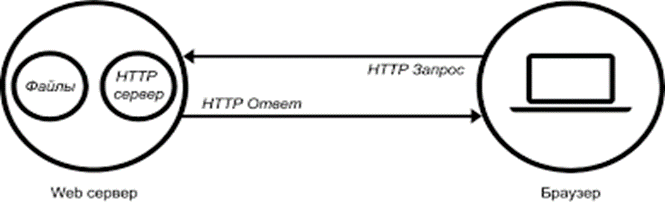


Рисунок 1 – Передача данных через HTTP-протокол

Как видно из рисунка 1, браузер запрашивает нужный файл, размещенный на сервере, через HTTP-протокол. Когда web-сервер получает запрос от пользователя, происходит обработка информации и передача запрашиваемого документа обратно через HTTP-протокол.

Первым шагом для размещения web-ресурса в сеть Интернет является выбор хостинг-провайдера, услуги которого являются наиболее удобными. Рассмотрев различные предложения компаний, в большей степени соответствует потребностям и предложенному обслуживанию крупнейший хостинг-провайдер TimeWeb. Для получения доменного имени и адреса хоста необходимо создать базу данных, как представлено на рисунке 2.

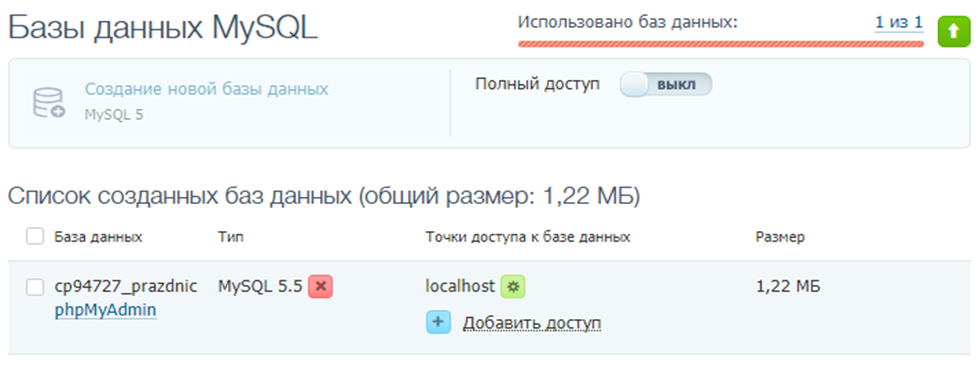


Рисунок 2 – Обзор базы данных

Основные идеи современной информационной технологии базируются на концепции, согласно которой данные должны быть организованы в базы данных. Они создаются и функционируют под управлением специальных программных комплексов, называемых системами управления базами данных (СУБД). Рассмотрим более подробно.

База данных – это организованное хранение информационных ресурсов, которые представляются в виде интегрированной совокупности файлов, обеспечивающей удобное взаимодействие и быстрый доступ к данным. Значения преобразуются при изменении состояния отражаемой предметной областью, под которой понимаются объекты, процессы, которые могут быть представлены в базе данных в полном информационном объеме.

Информация, хранимая в базе, генерируется в единую целостную структуру, что обеспечивает более эффективную работу пользователей с большим объемом данных.

Система управления базой данных (СУБД) – это программное обеспечение, с помощью которого пользователи могут определять, создавать и поддерживать базу данных, а также осуществлять к ней контролируемый доступ.

На рисунке 3 представлено размещение полученных данных на web-сервере.

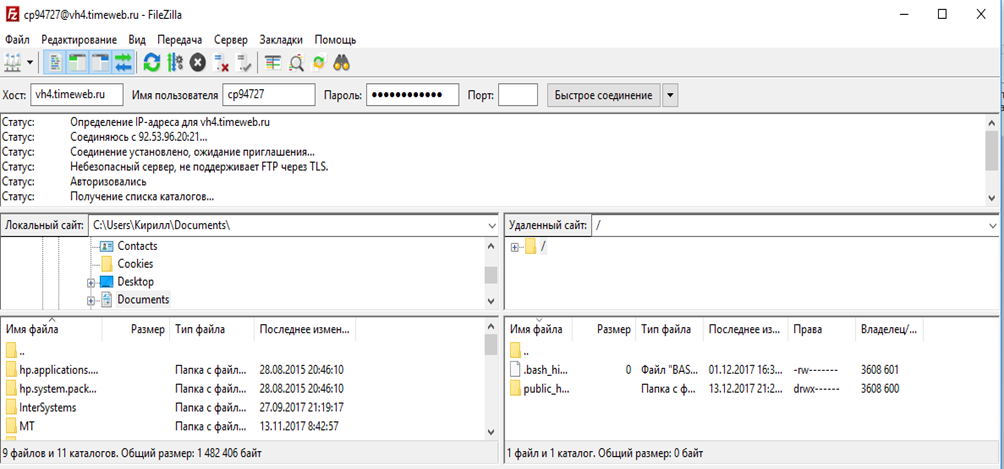


Рисунок 3 – Обзор FTP-клиента(FileZilla)

Визуальное восприятие играет важную роль, поэтому, качественно оформленная страница с товаром является важным элементом успешной стратегии.

Профессионально продуманная web-страница, предоставляющая услуги своей организации посетителю интернет-пользователя, является важным условием успешного бизнеса. Удобное расположение и оформление различных ее элементов поможет получить программное обеспечение WordPress.

Многофункциональная тема WordPress Conica упакована с многочисленным набором настроек, что позволяет создавать и настраивать собственные шаблоны страницы. Для обеспечения доступа к сайту в сети Интернет полученные данные вносим в файл wp-config.php, как представлено на рисунке 4.

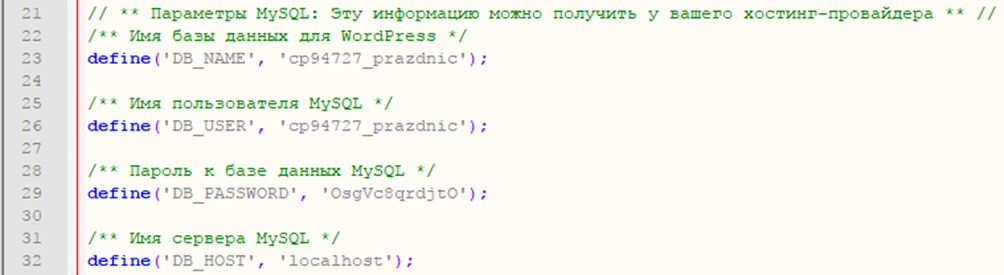


Рисунок 4 – Фрагмент кода в файле wp-config.php

Теперь можно преступить непосредственно к разработке web-странице, которая предоставляет услуги по организации праздничных мероприятий.

2.2 Реализация собственного входа

Для уникальности созданного сайта реализуем свою собственную форму регистрации и входа пользователей. Авторизация предоставляет возможность взаимодействовать непосредственно с данным web-ресурсом, а также оставлять комментарии, которые помогут узнать мнение, пожелания и оценки целевой аудитории, заинтересовавшейся услугами сайта.

Для начала ограничим доступ к области администратора. Фрагмент, отвечающий за вход в панель администрирования только администратору сайта показан на рисунке 5.

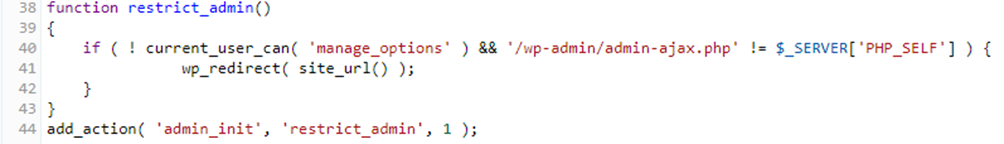


Рисунок 5 – Фрагмент кода из файла function.php

Представленный выше код совершает проверку, является ли пользователь администратором. Такой формат позволяет перенаправлять обычных пользователей сайта на главную страницу.

Теперь встает вопрос о создании новой формы входа для наших посетителей.

Создаем новый PHP-файл Login\_Page.php, полный листинг которого представлен в ПРИЛОЖЕНИИ А, в теме и добавляем в него следующий код, который указан на рисунке 6. Чтобы данная страница выглядела так же, как и все другие страницы, представленные в теме, добавляем get\_header(), get\_sidebar(), get\_footer(). Функция function is wp\_login\_form() получает один параметр, который является массивом аргументов, использующихся для изменения поведения формы. Данный пример демонстрирует изменение URL перенаправления, позволяющий переносить тех пользователей, которые выполнили корректный вход, в специальную область сайта, доступной для чтения.

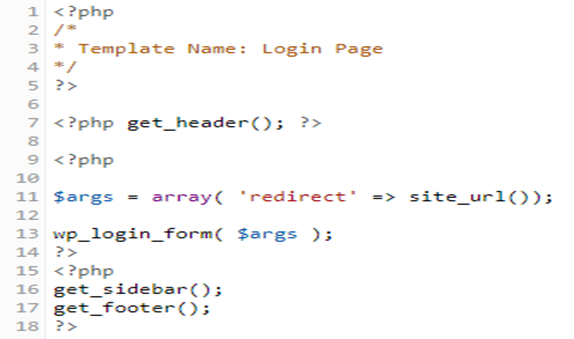


Рисунок 6 – Код для входа пользователей

Не стоит забывать о том, что введенные данные могут не совпадать с данными пользователей, зарегистрированных ранее, или же отсутствие их.

Для начала необходимо обработать событие, которое произойдет, если была совершена неудачная попытка входа, т.е. если был введен некорректный логин или пароль. В WordPress имеется структура wp\_login\_failed, которая помогает привязать функцию для обработки неудачной попытки входа.

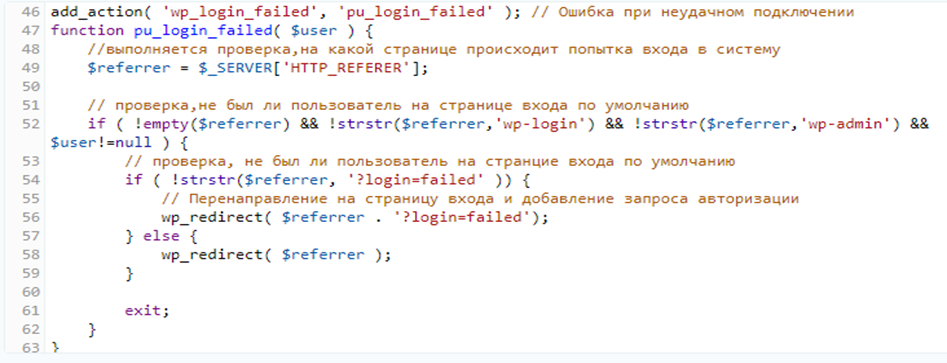


Рисунок 7 – Проверка некорректного ввода данных

Рисунок 7 показывает, что как только попытка входа будет неудачной, необходимо получить URL страницы, и убедиться в том, что это не стандартная страница wp-login.php, после чего происходит добавление строкового параметра неудачного входа и перенаправление с этим параметром на форму входа.

Если пользователь вашего web-ресурса не ввел данные, необходимые для входа, то его перенаправят обратно к стандартной форме входа WordPress. Для решения данного вопроса можно воспользоваться функцией authenticate. Фрагмент кода, который выполняет проверку пустоты, показан на рисунке 8.

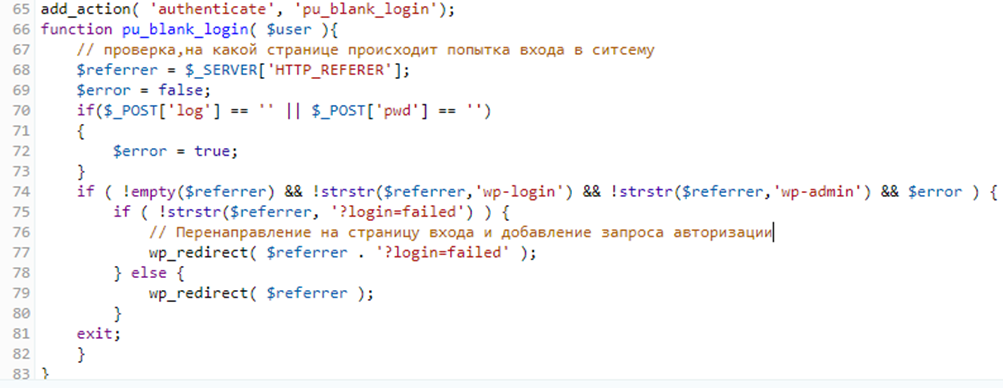


Рисунок 8 – Проверка пустоты данных логина и пароля

На данном этапе можно проверить логин и пароль, убедиться в том, что они не пустые, и в противном случае перенаправить обратно со строкой неудачного входа.

2.3 Реализация собственной регистрации на сайте

Для реализации формы регистрации на web-сайте создадим плагин, который будет запрашивать:

* имя пользователя;
* пароль;
* е-mail;
* имя;
* фамилия;
* информация о себе.

Регистрационная форма находится в верхней панели меню, что позволяет с легкостью перейти на страницу с регистрацией пользователям, которые впервые оказались на данном сайте.

Плагин – это программа или набор функций, которые написаны на языке PHP, добавляющий определенный набор возможностей к разработанному web-ресурсу, и легко объединяются с системой управления и функционалом WordPress при помощи Plugin Application Program Interface (API).

Для написания кода плагина, полный листинг которого представлен в ПРИЛОЖЕНИЕ Б, необходимо начать с того, что стандартный информационный заголовок должен содержать следующий код:

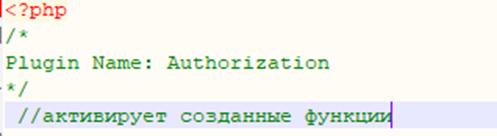


Рисунок 9 – Формат заголовка

На рисунке 9 указан фрагмент начала оформления плагина. Этот заголовок позволяет WordPress понять, что ваш плагин существует, поэтому его возможно добавить в панель управления плагинами, активировать, загрузить и запустить его функции непосредственно на вашем сайте. Минимальная информация, которая должна содержаться в самом начале кода, – Plugin Name (название плагина).

Заполненные поля в регистрационной форме должны быть проверены. Для этого создадим функцию проверки под названием authorization\_validation.

Чтобы облегчить процесс проверки, воспользуемся классом WordPress WP\_Error, который позволяет передавать и обрабатывать ошибки в плагинах и темах.

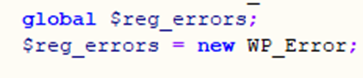


Рисунок 10 – Фрагмент кода плагина

Используем класс WP\_Error и делаем переменную глобальной, чтобы она могла служить в качестве доступа за пределами функции, как показано на рисунке 10.

Как и любой формат заполнения регистрационных полей нужно учитывать количество введенных символов в полях имени пользователя и пароля, правильность данного e-mail и многих других параметров. С проверкой данного запроса можно ознакомиться в ПРИЛОЖЕНИЕ А.

Далее нам нужна функция плагина complete\_authorization(), которая отвечает за обработку регистрации пользователей. А сама регистрация пользователей, на самом деле, осуществляется с помощью функции wp\_insert\_user, которая принимает ряд пользовательских данных.

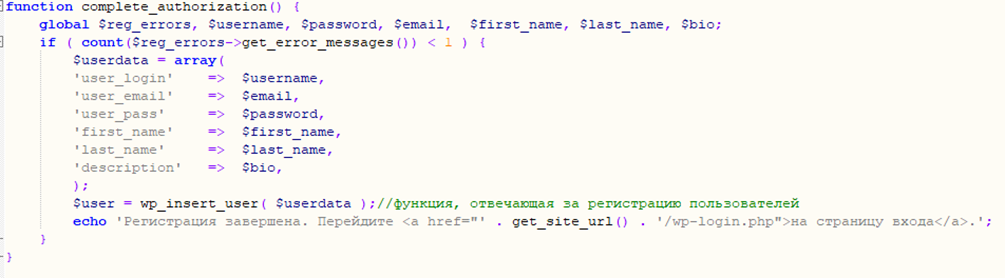


Рисунок 11 – Функция обработки регистрации пользователя

В фрагменте кода, представленный на рисунке 11, создана переменная $reg\_errors WP\_Error и поля формы, которые открывают доступ к переменной в глобальном масштабе.

Затем происходит проверка переменной $reg\_errors, которая обрабатывает запросы и выявляет наличие ошибок. Если данные введены корректно, переходим к заполнению $userdata, вставляем данные пользователя в базу данных WordPress и отображаем сообщение «Регистрация прошла успешно» со ссылкой на страницу входа в систему.

На очереди у нас отличная функция authorization\_function(), которая активирует все созданные нами функции. Проверка установки $\_POST['submit'] показывает, что форма регистрация была завершена и отправлена. После чего используем функцию authorization\_validation для проверки отправленной пользователем формы. C помощью complete\_authorization регистрируем пользователя.

В разработанном данном проекте дизайн страницы регистрации оформлен как показано на рисунке 12.



Рисунок 12 – Дизайн страницы регистрации

2.4 Оформление страниц

Дизайн – это первое, на что обращает внимание пользователь, зайдя на web-ресурс. Разработка дизайна – важная составляющая для любого сайта.

Главная страница является лицом вашей организации. Разработанный сайт по организации праздников на главной странице содержит перечень всех возможных услуг, поэтому так важно, чтобы он был целостный, полезный и предоставлял посетителю вашей страницы удобные и доступные перемещения по разработанному web-ресурсу. Другими словами, человек должен понять: чем занимается ваша организация, что предоставляет в качестве своих услуг и по каким ссылкам переходить, что найти интересующую информацию.

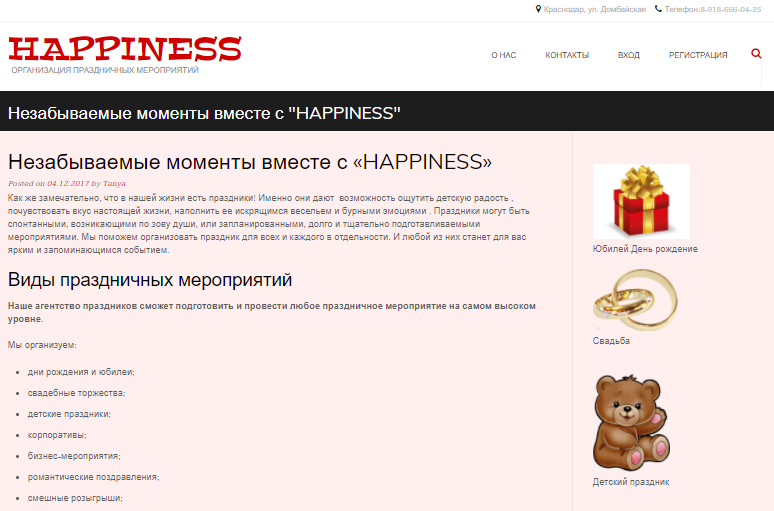


Рисунок 13 – Главная страница разработанного сайта

Ознакомившись с рисунком 13, заметим, что в боковой панели располагаются ссылки, которые позволяют перейти на те или иные страницы.

Код, который можно посмотреть в ПРИЛОЖЕНИИ В, расположен в документе, прикрепленном к разделу виджеты. Это позволяет всегда находится в поле зрения интернет-пользователей, вне зависимости от перемещения относительно этого web-сайта. Выбрав нужный раздел, посетитель может ознакомиться с интересующимся видом поздравления. Для дальнейшего согласования места и времени проведения мероприятия, можно перейти на страницу, отвечающую за контактные данные, как представлено на рисунке 14.

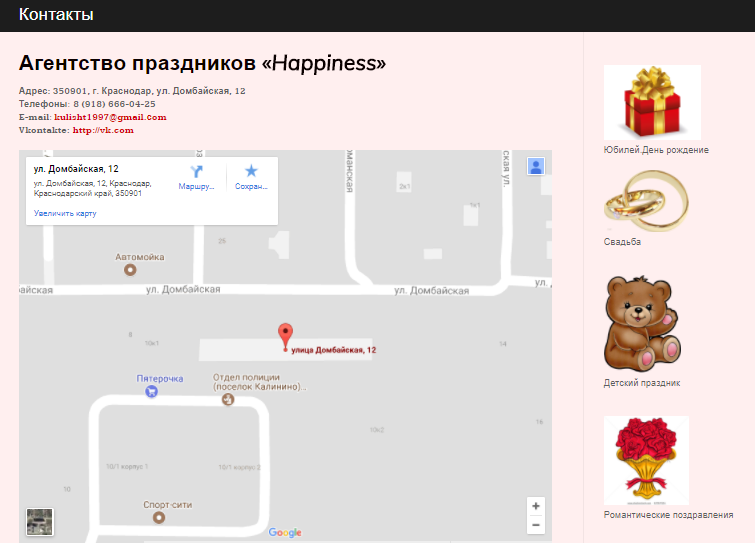


Рисунок 14 – Страница контактной информации

Код, который позволяет показывать место положение вашей организации на карте, представлен на рисунке 15.

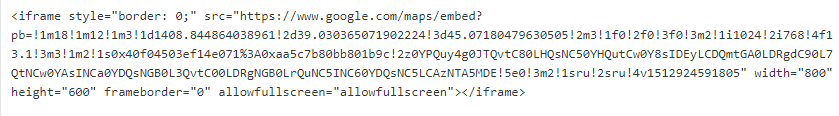


Рисунок 15 – Фрагмент подключения карты города

Для просмотра разработанного web-ресурса о организации праздничных мероприятий можно перейти по ссылке <http://cp94727.tmweb.ru/>.

ЗАКЛЮЧЕНИЕ

Результатом выполнения курсовой работы является разработанный web-сайта по организация праздничных мероприятий, ссылка на который находится по адресу <http://cp94727.tmweb.ru/>. Web-ресурс предоставляет целевой аудитории, заинтересованных данным видом предлагаемых услуг, наиболее интересные и креативные поздравления родных и близких.

Эффектный и яркий дизайн сайта разработан с целью привлечения клиентов. Для удобного перемещения по страницам web-ресурса были реализована навигация в боковой панели меню.

Особое внимание при создании сайта было уделено безопасности пользователей, контролю над корректностью входных данных.

Актуальность разработанного проекта заключается в реализации собственного входа и регистрации, что позволяет сохранять пользователей в базе данных и открывает для них доступ к оформлению заказа.

Разработанный сайт предоставляет пользователю возможность изучить историю различных праздников, предоставляет большое количество идей для необычных, красочных и незабываемых поздравлений.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Написание плагина. URL: https:// codex.wordpress.org /
2. Якоб Нильсен, Хоа Лоранжер. Web-дизайн: удобство использования Web-сайтов — М.: «Вильямс», 2007, 368 с.
3. Виктор Ромашев. CMS Drupal: Система управления содержимым сайта. — Питер, 2010, 255 с.
4. Канев В.С., Шевцова Ю.В. Основы моделирования и управления операционными рисками в электронной коммерции и телекоммуникациях. — М.: Горячая линия - Телеком, 2015, 278 с.
5. Байков В. Д. Интернет: поиск информации и продвижение сайтов. — Санкт-Петербург: «БХВ-Санкт-Петербург», 2000, 288 с.
6. Юрасов А.В. Основы электронной коммерции. — М.: Горячая линия - Телеком, 2008, 480 с.
7. Ральф Ф. Уилсон. Планирование стратегии интернет-маркетинга — М.: Издательский Дом Гребенникова, 2003, 264 с.

ПРИЛОЖЕНИЕ А

Login\_page.php

<?php

/\*

\* Template Name: Login Page

\*/

?>

<?php get\_header(); ?>

<?php

/\*\*

\* Template Name: User Login

\*/

$args = array( 'redirect' => site\_url());

if(isset($\_GET['login']) && $\_GET['login'] == 'failed')

{

?>

<div id="login-error" style="background-color: #FFEBE8;border:1px solid #C00;padding:5px;">

<p>Ошибка входа: вы ввели неправильное имя пользователя или пароль, повторите попытку.</p>

</div>

<?php

}

wp\_login\_form( $args );

?>

<?php

get\_sidebar();

get\_footer();

?>

ПРИЛОЖЕНИЕ Б

Authorization.php

<?php

/\*

Plugin Name: Authorization

\*/

function authorization\_function() {

if (isset($\_POST['submit'])) {

authorization\_validation(

$\_POST['username'],

$\_POST['password'],

$\_POST['email'],

$\_POST['website'],

$\_POST['fname'],

$\_POST['lname'],

$\_POST['bio']

);

global $username, $password, $email, $website, $first\_name, $last\_name, $nickname, $bio;

$username = sanitize\_user($\_POST['username']);

$password = esc\_attr($\_POST['password']);

$email = sanitize\_email($\_POST['email']);

$website = esc\_url($\_POST['website']);

$first\_name = sanitize\_text\_field($\_POST['fname']);

$last\_name = sanitize\_text\_field($\_POST['lname']);

$bio = esc\_textarea($\_POST['bio']);

complete\_authorization(

$username,

$password,

$email,

$first\_name,

$last\_name,

$bio

);

}

authorization\_form(

$username,

$password,

$email,

$first\_name,

$last\_name,

$bio

);

}

function authorization\_form( $username, $password, $email, $first\_name, $last\_name, $bio ) {

echo '

<style>

div {

margin-bottom:2px;

}

.field {clear:both; text-align:right; line-height:25px;}

label {float:left; padding-right:10px;}

.main {float:left}

input{

margin-bottom:4px;

}

</style>

';

echo '

<form action="' . $\_SERVER['REQUEST\_URI'] . '" method="post">

<div class="main">

<div class="field">

<label for="username">Логин<strong>:</strong></label>

<input type="text" name="username" value="' . (isset($\_POST['username']) ? $username : null) . '">

</div>

<div class="field">

<label for="password">Пароль<strong>:</strong></label>

<input type="password" name="password" value="' . (isset($\_POST['password']) ? $password : null) . '">

</div>

<div class="field">

<label for="email">Email <strong>:</strong></label>

<input type="text" name="email" value="' . (isset($\_POST['email']) ? $email : null) . '">

</div>

<div class="field">

<label for="firstname">Имя</label>

<input type="text" name="fname" value="' . (isset($\_POST['fname']) ? $first\_name : null) . '">

</div>

<div class="field">

<label for="website">Фамилия</label>

<input type="text" name="lname" value="' . (isset($\_POST['lname']) ? $last\_name : null) . '">

</div>

<div class="field">

<label for="bio">О себе</label>

<textarea name="bio">' . (isset($\_POST['bio']) ? $bio : null) . '</textarea>

</div>

<input type="submit" name="submit" value="Регистрация"/>

</form>

</div>

';

}

function authorization\_validation( $username, $password, $email, $first\_name, $last\_name, $bio ) {

global $reg\_errors;

$reg\_errors = new WP\_Error;

if ( empty( $username ) || empty( $password ) || empty( $email ) ) {

$reg\_errors->add('field', 'Поле обязательной формы заполнения отсутствует:');

}

if ( strlen( $username ) < 4) {

$reg\_errors->add('username\_length', 'Имя пользователя слишком короткое. Требуется не менее 4 символов');

}

if ( username\_exists( $username ) )

$reg\_errors->add('user\_name', 'Извините, это имя пользователя уже существует!');

if ( !validate\_username( $username ) ) {

$reg\_errors->add('username\_invalid', 'Извините, введенное вами имя пользователя недействительно');

}

if ( strlen( $password ) < 6 ) {

$reg\_errors->add('password', 'Пароль слишком короткий. Длина не должна иметь менее 6 символов');

}

if ( !is\_email( $email ) ) {

$reg\_errors->add('email\_invalid', 'Email не является допустимым');

}

if ( email\_exists( $email ) ) {

$reg\_errors->add('email', 'Этот Email уже занят');

}

if ( is\_wp\_error( $reg\_errors ) ) {

foreach ( $reg\_errors->get\_error\_messages() as $error ) {

echo '<div>';

echo '<strong>ERROR</strong>:';

echo $error . '<br/>';

echo '</div>';

}

}

}

function complete\_authorization() {

global $reg\_errors, $username, $password, $email, $first\_name, $last\_name, $bio;

if ( count($reg\_errors->get\_error\_messages()) < 1 ) {

$userdata = array(

'user\_login' => $username,

'user\_email' => $email,

'user\_pass' => $password,

'first\_name' => $first\_name,

'last\_name' => $last\_name,

'description' => $bio,

);

$user = wp\_insert\_user( $userdata );

echo 'Регистрация завершена. Перейдите <a href="' . get\_site\_url() . '/wp-login.php">на страницу входа</a>.';

}

}

add\_shortcode('cr\_authorization', 'authorization\_shortcode');

function authorization\_shortcode() {

ob\_start();

authorization\_function();

return ob\_get\_clean();

}

ПРИЛОЖЕНИЕ В

Гиперссылки с использованием изображения

<div class="entry">

<a href="http://cp94727.tmweb.ru">

<img class="" style="margin- top: 7px;" src="http://www.images.lesyadraw.ru/2013/12/kak\_narisovat\_podarok.jpg" alt="" width="146" height="112" /></a>

<a href="http://cp94727.tmweb.ru/197-2/">Юбилей.День рождение</a>

<img class="" src="https://img-fotki.yandex.ru/get/6813/2449448.21/0\_10a222\_a656d8e\_orig" alt="" width="127" height="92" />

<a href="http://cp94727.tmweb.ru/205-2/">Свадьба</a>

<a href="http://cp94727.tmweb.ru">

</a><img class="" src="http://detsad-mishutka1984.narod.ru/olderfiles/2/76205046\_large\_0\_506f2\_1c58441\_L\_kopiya.png" alt="" width="115" height="146" />

<a href="http://cp94727.tmweb.ru/209-2/">Детский праздник</a>

<a href="http://cp94727.tmweb.ru">

</a><img class="" src="https://thumb7.shutterstock.com/display\_pic\_with\_logo/1003250/117617224/stock-vector-vintage-vector-bouquet-of-roses-sketchy-illustration-holidays-postcard-red-roses-red-ribbons-117617224.jpg" width="128" height="133" />

<a href="http://cp94727.tmweb.ru/214-2/">Романтические поздравления</a>

<a href="http://cp94727.tmweb.ru">

</a><img class="" src="https://media.istockphoto.com/vectors/yellow-cone-with-party-hat-vector-id487719676" alt="" width="98" height="129" />

<a href="http://cp94727.tmweb.ru/217-2/">Корпоративы</a>

<a href="http://cp94727.tmweb.ru">

</a><img class="" src="http://cdn6.s.kolorado.ru/products/1/19/194/194661/118\_1\_1\_design.png" alt="" width="128" height="148" />

<a href="http://cp94727.tmweb.ru/220-2/">Бизнес-мероприятия</a>

</div>