

Федеральное агентство связи
Федеральное государственное образовательное бюджетное учреждение высшего
профессионального образования
«Сибирский государственный университет телекоммуникаций и информатики»
(ФГОБУ ВПО «СибГУТИ»)

Кафедра ПМиК
Допустить к защите
зав. кафедрой: проф., д.т.н.

_____ Фионов А.Н.

**ВЫПУСКНАЯ
КВАЛИФИКАЦИОННАЯ РАБОТА
БАКАЛАВРА
«РАЗРАБОТКА САЙТА «СТУДИЯ ФЛОРИСТИКИ»**

Пояснительная записка

Студент _____ Чурунова Светлана Николаевна /...../

Факультет _____ ИВТ _____ Группа _____ ПБВ-33

Руководитель _____ Барахнин Владимир Борисович //

Новосибирск 2016 г.

Содержание

Введение	3
1 Общая часть	5
1.1 Постановка задачи. Требования к создаваемому Web-сайту	5
1.2 Формирование разработки Web-сайта. Выбор CMS	6
1.3 Разработка сайта на основе CMS WordPress	8
2 Используемые языки и программное обеспечение	12
2.1 HTML –язык гипертекстовый разметки	13
2.2 CSS (каскадные таблицы стилей)	14
2.3 DHTML – динамический HTML	14
2.4 JavaScript (язык программирования)	15
2.5 PHP (язык программирования)	16
2.6 MySQL	19
3 Анализ, проектирование и реализация сайта «Студия флористики» ..	21
3.1 Обзор и анализ методов проектирования сайта	21
3.2 Анализ ведущих Интернет-сайтов России	25
3.3 Проектирование сайта и классификация пользователей проекта ..	29
3.4 Принципы реализации элементов дизайна сайта	30
3.5 Описание структуры хранения данных	31
4 Описание разработанного приложения	40
4.1 Описание интерфейса пользователя	40
4.2 Описание интерфейса администратора	49
Заключение	52
Список использованных источников	53
Приложение А. Листинг Программы	54

ВВЕДЕНИЕ

Интернет – глобальная сеть, она развивается и стремительно захватывает весь мир. Интернет становится полноценным источником всевозможной информации.

Интернет стал неотъемлемой частью жизни россиян. Количество пользователей интернета в России в 2015 году возросло на 9,2% — до 80,5 миллиона человек, показало исследование, которое представила Российская ассоциация электронных коммуникаций (РАЭК). Согласно прогнозу РАЭК, к концу 2017 года доля пользователей старше 18 лет, пользующихся интернетом ежедневно, возрастет до 80,55%, а к концу 2020 года – 84,48%.

Самые популярные виртуальные услуги это: Интернет-аукционы, Интернет - магазины, виртуальные витрины, электронная почта, группы новостей, файлообменные сети, электронные платёжные системы, Интернет-радио, Интернет-телевидение, поисковые системы, Интернет-реклама и прочее.

Темой дипломной работы является разработка сайта «Студия флористики».

Цель дипломной работы – разработать сайт с удобным, современным и простым интерфейсом. Таким образом, объектом и предметом данной дипломной работы является сайт «Студия флористики».

Флористика, составление композиций из цветов, цветочный дизайн - это искусство создания цветочных композиций в вазах, шарах и корзинах или создание букетов и композиций из срезанных цветов, листьев, декоративных трав и других ботанических материалов. Как правило, флорист komponует цветы в зависимости от сезона и праздника. Что и будет объектом исследования.

Многие потенциальные покупатели не имеют возможности постоянно посещать салоны и магазины, хотя готовы сделать покупку. Им удобнее позвонить по телефону и узнать о новых поступлениях товара или воспользоваться Интернетом. У многих Web-сайтов только одна цель - продать товары или услуги.

Одно из главных преимуществ сайта считается его быстрая загрузка. Если посетителям придется ждать более 20 секунд, то они могут потерять интерес к нему.

Удачный Web-сайт — это эффективный инструмент торговли — он способен привлекать внимание аудитории. Как и любой другой маркетинговый инструмент, основанный на принципе непосредственного отклика, прежде всего он должен заинтересовать посетителя, а затем подвигнуть его на определенные действия. Размещение новостей в сети Интернет гораздо результативнее, поскольку предпочтения покупателей в выборе основываются, в первую очередь на визуальных оценках, затем осуществляется ценовой выбор, в случае удовлетворения всех потребностей – покупается. Однако, многие не принимают во внимание эту особенность главной страницы, что часто приводит к тому, что посетителей не заинтересовывает сайт и они

покидают его, едва зайдя. Такие Web-сайты, пусть даже содержащие иногда большое количество полезной информации, практически никогда не достигают предполагаемого уровня посещаемости, не говоря уже о продажах.

С учетом технического прогресса Internet-технологии развиваются и совершенствуются с каждым днем. Широкое распространение сети Интернет получают даже отдаленные регионы страны. Интернет становится полноценным источником всевозможной полезной информации для любой категории пользователей.

Рассматриваемые вопросы в дипломной работе являются актуальными в области разработки и создания современного Web-сайта.

1 ОБЩАЯ ЧАСТЬ

1.1 Постановка задачи. Требования к создаваемому Web-сайту

Цель работы: Приобретение теоретических и практических знаний по разработке сайта с применением языков программирования HTML, CSS, JavaScript, PHP и реляционная база данных MySQL и разработать сайт «Студия флористики».

Задачи:

- рассмотреть основные теоретические понятия разработки интернет-сайтов;
- рассмотреть структуру разрабатываемого сайта;
- приобрести навыки проектирования и разработки web-сайтов;
- разработать сайт на основе WordPress.

Ожидаемые результаты: Создание интернет-сайта «Студия флористики».

Объектом исследования является создание сайта, посвященный флористике.

Флористика - один из видов визуального искусства, в котором художник вместо красок использует цветочные лепестки, листья, травы, а также специальные искусственные материалы. Это выражение фантазии, творческих порывов и чувств человека посредством живых и сухих даров растительного мира.

Сегодня, проблема разобщенности человека с природой ощущается особенно остро. Живые цветы – самое потрясающее украшение, придуманное природой. Поэтому цветочные композиции являются отличным подарком для любимых и близких, надолго оставляя о вас самые прекрасные воспоминания.

Требования к созданию сайта.

Страницы сайта, выполненные в едином стиле, создают впечатление о целостности сайта.

Эстетическое позитивное впечатление производят страницы сайта, оформленные в духе функционального разумного минимализма.

Рациональное использование площади окна браузера страницами сайта.

Система навигации сайта проста и интуитивно понятна любому пользователю, экономит время и позволяет быстро найти нужную информацию. Наличие поиска по контенту является преимуществом.

Основное содержание должно занимать максимально возможную площадь пространства страницы без ущерба для остальных функциональных элементов сайта. Огромный плюс – это страница сайта, на которой отсутствуют бесполезные элементы для посетителей.

Основная функциональность сервисов сайта, которые важны для пользователей, сохраняется при отключении в браузере Flash, JavaScript и ActiveX.

Весь текстовый контент: основные тексты и комментарии к графическим и другим объектам должны быть доступны пользователям, которые отключают графику или пользуются консольными и речевыми браузерами.

Весь контент сайта свободно индексируется поисковыми машинами: переход между страницами поискового робота не затруднен технологическими ограничениями (например, навигацией JavaScript).

HTML-код страниц сайта должен соответствовать правилам семантической разметки. Например:

- заголовки, в соответствии с иерархией, отмечают тегами `<h1>...<h6>`, но не ``;
- абзацы имеют тег `<p>`, а не `
`;
- таблицы используются исключительно по их прямому назначению;
- классам и идентификаторам названия даются в соответствии со смысловой нагрузкой, а не с их визуальным представлением.

За логическое структурирование контента отвечает HTML-разметка, тогда как на CSS возложено всё управление его представлением (например, внешний вид).

Вне зависимости от версии браузера (Internet Explorer, Chrome, Safari, Opera, Firefox) страницы сайта отображаются без каких-либо существенных отличий.

На страницах сайта предусмотрены простые для пользователя, автоматически генерируемые версии контента для просмотра страниц на экранах мобильных телефонов.

Сайт должен функционировать на мобильных телефонах и планшетах с операционными системами Android и iOS без каких-либо затруднений.

Текст, расположенный на страницах сайта, вселяет уверенность, что автор владеет русским языком и правописанием в совершенстве.

Иллюстрации и фотографии, находящиеся на сайте, не должны вызывать сомнений в технических навыках и таланте автора.

Код разметки соответствует спецификациям HTML5, XHTML 1.1 и XHTML 1.0 Strict. Код таблиц стиля соответствует спецификации CSS 3.0.

Даже если посетитель использует модемный или мобильный доступ в сеть, любая страница сайта должна загружаться не более 10 секунд.

1.2 Формирование разработки Web-сайта. Выбор CMS

Типичный процесс разработки сайта, состоит из упорядоченного процесса действий, который в свою очередь при последовательном планировании, приведет к итоговому запуску сайта.

Если рассматривать визуально этапы планирования Web-сайта, то это выглядит следующим образом, в соответствии с рисунком 1.1.



Рисунок 1.1 - Этапы планирования web-сайта

Этапы планирования сайта:

- первый контакт - постановка цели и дальнейшее её преследование;
- планирование - как, кем, чем и в какие сроки будут проектироваться отдельные части сайта;
- контент - это содержимое сайта, его информационное наполнение;
- дизайн - разработка визуального оформления сайта;
- программирование - сайта на язык PHP, для CMS WordPress;
- запуск.

На сегодняшний день, не все в эффективности сайта решается современным дизайном, великолепной версткой и совершенной программной частью. Поговорим о других важнейших моментах создания успешных сайтов.

Для начала, важно добиться ясности и простоты. Это два кита, на которых обязаны базироваться любые помыслы бизнесмена касаясь разработки или деятельности Web-сайта. Важно понять, что интернет-клиенты не всегда готовы думать, и не следует заставлять их делать это. Клиенты не будут терпеливыми для достижения определенного эффекта. Им даже не захочется разыскивать оптимальные пути решения собственных вопросов, ведь для них они или будут, или нет, и, если понимание, поиск или применение затруднено, ваш сайт будет покинут.

Требуется понять, что никакой потенциальный клиент не будет посещать ресурс для оценки прелестей дизайна или же понимания принципов функционирования. Зачастую пользователь просто нуждается в конкретной информации. Говоря о красивом дизайне, это элемент вспомогательный.

Таким образом, принцип ясности и простоты должен быть прописан в каждом элементе Web-сайта. Каждый из элементов должен оправданно присутствовать на своем месте, без загромождений страниц.

Помимо этого, крайне важно продумывать графику, что способствует простоте восприятия, обеспечивая ощущение стабильности и равновесия, которые крайне важны для всех пользователей. Продумывая форму компонентов сайта и их размещение, важно вспоминать, что расположенные

друг возле друга элементы будут восприниматься пользователями в качестве группы.

Говоря о цветовых решениях, будет важно отметить, что белый и черный цвет относятся к пассивным, и их стоит применять в сочетании со свежими и яркими оттенками, выступающими энергетическими.¹

Продумывая графику, не стоит перебарщивать, достаточно будет остановиться на двух-трех цветах. Также не следует перегружать портал изображениями, которые значительно чаще обращают на себя внимание, чем текстовая информация, но чуткость не менее важна.

На основе всего вышесказанного и умелого применения имеющихся знаний, можно довольно легко научиться управлять вниманием клиентов, с успешным направлением их к определенной цели.

Инструменты управления сайта (CMS) - информационная система или компьютерная программа, используемая для создания, редактирования, организацией обмена информацией и последующим управлением контента [1].

Основные функции CMS:

- наличие инструментов для создания контента - содержимого сайта;
- управление контентом: хранение, контролем версиями, соблюдение доступа для управления документами;
- добавлением материалов, а также их редактирование и удаление;
- представление информации, удобном для навигации по сайту, а также интуитивно понятным поиском;
- пользовательская часть - заполнение информативной наполняемостью сайта, полноценной и достоверной информацией, которая в свою очередь просматривается посетителями сайта;
- интерфейс сайта – это совокупность средств, с помощью которых пользователь общается с визуальным содержимым сайта.

1.3 Разработка сайта на основе CMS Wordpress

До недавнего времени, Web-сайты, основанные на CMS, были последней инновацией в сети. Большинство крупных компании использовали CMS для своих сайтов, потому что они позволяют людям управлять контентом, не зная тонкостей языка программирования. Системы схожие в функциональности, с WordPress, её функции и гибкость, могут стоить десятки или даже сотни тысяч долларов. WordPress является свободным программным обеспечением с открытым исходным кодом, который может быть загружен и установлен в считанные минуты.

WordPress является открытой платформой, он позволяет изменять исходный код системы на свой собственный. Если изменения будут успешными, они могут предложить сообщества для того, чтобы включить их в следующем официальном релизе системы. Но добавление нового кода в

¹ Материалы с сайта: <http://edu.evnts.pw/materials/131/>

ядро платформы данных решений могут осуществить лишь пять основных разработчиков. Кроме того, дополнительные функции могут быть реализованы с помощью модулей расширения или отдельных плагинов.

Плагин - расширение WordPress, который добавляет функции или категории контента, не входящие в него изначально.

Платформа WordPress существует в двух формах: WordPress.org и WordPress.com. WordPress.org Сайт содержит последние версии CMS, плагины, темы и места. WordPress.com это сервисная компания Otomettik, которая была основана с целью предоставления бесплатного хостинга для WordPress. WordPress.com построен на ветке WordPress, под названием WordPress MU, что позволяет использовать только одну установленную копию CMS, но контролировать одновременно не один сайт.

WordPress.com принадлежит и поддерживается Automattic, который также курирует сайт WordPress.org и обеспечивает ее развитие. Компания предоставляет инфраструктуру для поддержки CMS WordPress и предлагает работу для многих веб-разработчиков системы [2].

Установка платформы на серверах хостинга проводится в три этапа:

- скачать файл CMS на хостинг-провайдера;
- разработать базу данных MySQL для этой платформы;
- запустить веб-интерфейс установки WordPress;
- расставить права доступа к директориям.

Загрузка файлов на сервер может быть с помощью веб-интерфейса хостинг-провайдера или любого FTP-клиента. При загрузке файлов CMS должен быть загружен в public_html и папку WWW, потому что это корень будущего сайта. Что также можно установить в отдельные директории сайта, в соответствии с рисунком 1.2.

Below you should enter your database connection details. If you're not sure about these, click here.

Database Name	<input type="text" value="yourwordpresssitename"/>	The name of the database you want to use.
User Name	<input type="text" value="youruserid"/>	Your MySQL username.
Password	<input type="text" value="youruserid"/>	...and MySQL password.
Database Host	<input type="text" value="localhost"/>	99% chance you won't need to change this.
Table Prefix	<input type="text" value="wp_"/>	If you want to run multiple WordPress installations in a single database, change this.

Рисунок 1.2 - Окно установки WordPress

WordPress - система управления контентом в том виде, в котором она применяется, не требует специальных мер по оптимизации и улучшения производительности, хотя можно изменить некоторые настройки, которые лучше всего соответствуют потребностям данного CMS веб-ресурса.

Увеличение производительности WordPress можно разделить на несколько этапов:

- выбор правильных настроек для сайта;
- модерация комментариев и спама;
- контроль дополнительных авторов;
- кэширование и другие корректировки.

Когда создали сайт на базе этой CMS системы управления необходимо ввести оптимальные настройки, которые будут соответствовать требованиям веб-ресурса. Эти требования варьируются от места к месту и характеризуются особенностями задач, которые выполняет ресурс.

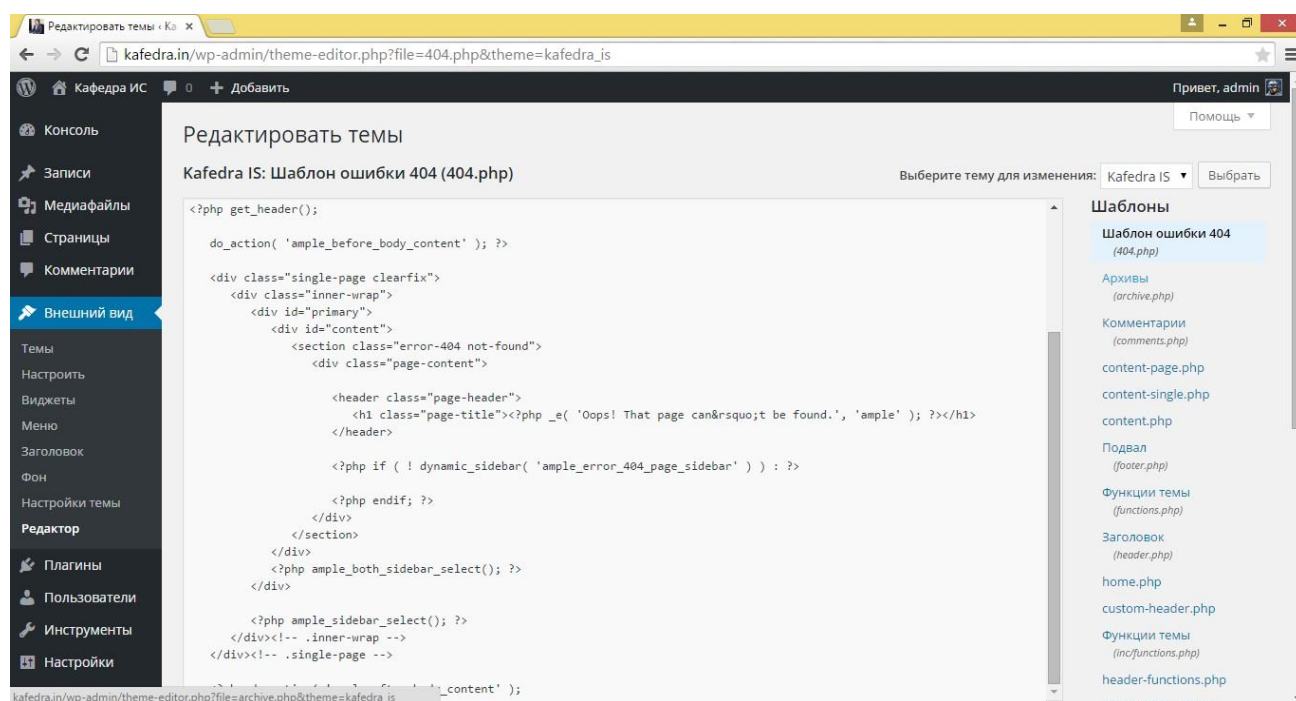


Рисунок 1.3 - Вид системы управления с открытой вкладкой, на которой основные настройки системы

Комментарии могут быть одним из лучших компонентов сайта. Они поддерживают дискуссию и привлекают новых посетителей на ресурс. Но есть и обратная сторона - спам. Для борьбы со спамом два основных способа управления комментариями. Первый способ - умеренность. Перед публикацией комментария приходит уведомление, что новые комментарии требуют утверждения.

Второй способ - использование специализированных плагинов. Они предназначены для борьбы со спамом, которые перехватывают входящие комментарии, если те не удовлетворяют соответствующим критериям, и в

последствии помещают в корзины со спамом.

Многие пользователи являются единственными авторами в блоге или на сайте, и они не должны добавлять пользователей в CMS. Тем не менее, если обеспечить большее количество пользователей, например, ведение несколько блогов, используйте возможность WordPress для добавления пользователей в систему управления контентом.

WordPress платформы, есть пять уровней-ролей пользователей, начиная от пользователей и заканчивая администратором [3].

- Администратор (Administrator). В этом типе пользователь имеет доступ ко всем функциям сайта. Это учетная запись с максимально возможной функцией контроля.

- Редактор (редактор). Редактор имеет доступ ко всему, что связано с содержанием блога, в дополнение к настройкам. Редактор может читать, публиковать, удалять записи, страницы, ссылки, заголовки и метки. Многие эксперты по безопасности рекомендуют владельцам блогов и сайтов, основанных на WordPress создать для себя дополнительный пользовательский логин с возможностями редактора контента, и использовать его для повседневной работы. Если эта учетная запись будет взломана, сама сайт не будет страдать, если же взломана администраторская учетная запись, то последствия могут быть разрушительные для сайта.

- Автор (Автор). Авторы могут создавать и публиковать свои собственные материалы, но они не могут изменить содержание от других пользователей, или добавлять новые разделы и ссылки.

- Участник (Автор). Помощники могут создавать свои собственные записи, но они не имеют права публиковать блог или Web-сайт.

- Абонент (Абонент). Абоненты могут только читать записи в блоге и оставлять комментарии. Эта роль может быть полезна при использовании плагина почту.

Что касается кэша, есть разные мнения. Но автор этой CMS считает, что WordPress в состоянии справиться с возросшим трафиком без посторонней помощи. Другие люди считают, что использование кэширования для этой платформы является оправданным. Это позволяет повысить производительность и уменьшить нагрузку на сервер.

Несмотря на то, что есть встроенная функция кэширования, также можно использовать плагины, такие как WP-SuperCache, WP-Cache и W3 Total Cache. Эти плагины могут не только кэшировать страницу, но делать также легче конфигурацию и использовать дополнительные функции, такие как сжатие файлов CSS и скриптов PHP. Большинство сайтов на основе WordPress не требует серьезных кэширований.

2 ИСПОЛЬЗУЕМЫЕ ЯЗЫКИ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Для разработки современных сайтов обычно используются: HTML 5.0, CSS 3.0, JavaScript, PHP и реляционная база данных MySQL.

Существуют другие платформы для создания сайтов, такие как ASP.NET, Ruby или Python, но стоит рассматривать их в качестве платформы для разработки ресурсоемких и специализированных Web-приложений.

Большинство современных сайтов разрабатываются с помощью PHP (статистика говорит о 40%), так как он прост в освоении, имеет наибольшую распространённость у хостинг-провайдеров.

Стоит заметить, что все веб-технологии легко взаимодействуют друг с другом. CSS и JavaScript код легко внедряется в HTML и PHP код, в свою очередь, HTML интегрируется в PHP и наоборот, все это происходит на уровне вставок кода или подключения отдельных файлов. Но, как и в жизни, если что-то и позволительно, то совсем не значит что правильно. Существуют правила хорошего тона и в программировании.

Итак, создав «каркас» сайта написанный на HTML + CSS + JavaScript, начинаем организовывать «движок» будущего динамического PHP сайта. Но перед этим еще немного о JavaScript и плавно переходим к самому PHP.

Как уже было замечено, JavaScript (он же JS, но не путать с JAVA) выполняется на стороне клиента, прямо в браузере и служит в основном для расширения скромного базового всегда в соответствии со стандартами World Wide Web Consortium (W3C) – организации которая вырабатывает стандарты верстки. Особенно это касается IE (браузера от компании Microsoft). Мало того, браузеры еще и «самовольничают» с выводом страницы: подтягивают блоки, изменяют отображение шрифтов и т.п. В своей попытке исправить так называемый ляп (незакрытый, не там открытый/закрытый тег) браузеры делают вывод страницы – зачастую очень непредсказуемым. Они не злодеи, просто хотят чтобы все сайты выглядели хорошо, и при этом каждый браузер «помогает» по-своему. Вот, в решении всех этих проблем и приходится обращаться к JavaScript.

Непосредственно PHP – это язык программирования для более серьезных задач. На нем можно создавать полноценные приложения, не только для нужд персонального или корпоративного Web-сайта. Код PHP выполняется на стороне сервера, что положительно сказывается на скорости работы вашего сайта. В случае с PHP сам сервер обрабатывает скрипт и выдает результат браузеру, то есть ресурсы клиентского компьютера используются только для вывода результата работы скрипта. Дело в том, что если браузер – интерпретатор программного кода, то на обработку программы (скрипта) расходуются ресурсы компьютера пользователя, в операционной системе которого установлен данный браузер. Интерпретатор PHP установлен на web-сервере и все ресурсозатраты ложатся на этот удаленный компьютер. Есть еще много языков программирования, которые могут выполняться (интерпретироваться) на стороне сервера, например PERL. [4].

2.1 HTML – язык гипертекстовой разметки

HTML (от англ. HyperText Markup Language — «язык разметки (маркировки) гипертекста») — стандартный язык разметки документов в Интернете. В основном Web-страницы создаются с помощью языка HTML (или XHTML). Язык HTML интерпретируется браузерами и отображается в виде документа в удобной форме для человека.

Гипертекст своему существованию обязан интернету, хоть и создавался он совсем для другого. HTML позволяет производить переход от одной части текста к другой, при этом эти части могут храниться у разных пользователей на компьютерах.

HTML создан специально для разметки веб-страниц. Именно язык разметки дает браузеру необходимые инструкции о том, как отображать тексты и другие элементы страницы на мониторе. Не только различные браузеры, но и их версии могут по-разному воспринимать и отображать на экране код. Следовательно, некоторые элементы корректно выглядящие в браузере Mozilla Firefox могут выглядеть иначе в Internet Explorer или Chrome. При верстке (создании страниц) HTML-страниц необходимо просматривать их в самых популярных браузерах, например Google Chrome, Mozilla Firefox, Opera, Internet Explorer, Safari.

В дополнение к официальной спецификации языка которая развивается, производители браузеров от W3C (W3 Консорциум) их собственные элементы (расширения). Затем некоторые из этих пунктов после всемирного признания в описании следующих версиях языка получили участие. Инновационное решение «фреймы» не были включены в спецификации. Но браузеры поддерживают фреймы и многие книги по HTML, содержали описание без упоминания о том, что это нестандартное изделие. Впоследствии, стандарт стал де-факто. В версии 4 они уже были включены в базу.

HTML теги - это основа языка HTML. Каждый тег несет в себе определенную информацию, он описывает документ или способ форматирования текста. С помощью тегов задается все содержимое веб-страницы [5].

Тег помещается в угловые скобки <тег>. В большинстве случаев для тега задается парный закрывающий тег, но в некоторых случаях он отсутствует. Информация, заключенная между открывающим и закрывающим тегом, называется его контейнером (содержимым тега).

Общепринятым современным стандартом считается версия HTML 5.0

Отличия XHTML от HTML.

XHTML (англ. Extensible HyperText Markup Language) - расширяемый язык разметки гипертекста. Правильно не расширенный, а расширяемый. Это означает, что XHTML дополняется (расширяется) до сих пор. Так что такое XHTML? XHTML - это не новый язык, это набор дополнительных правил, которые необходимо соблюдать при написании HTML кода. Главным различием XHTML от HTML является способ обработки документа (интернет-

странички). Необходимо дать еще одно определение. Парсер — программа или часть программы, которая выполняет синтаксический анализ. Другое название - синтаксический анализатор. Программа выполнения анализа всего строения страницы т.е. кода страницы.

В HTML (во время анализа) при нахождении ошибки, она исправлялась, что затрачивало дополнительное время - браузеру нужно было понять, что разработчик хотел написать. В XHTML не исправлялись эти ошибки.

Например, при ошибке в теге, он просто с остальным текстом выводился.

Еще одно отличие заключается в том, что все элементы должны быть закрыты, а одиночные теги должны после символов иметь знак /, например: `
`.

2.2 CSS (каскадные таблицы стилей)

CSS (англ. Cascading Style Sheets — каскадные таблицы стилей) — формальный язык описания внешнего вида документа, написанного с использованием языка разметки. В основном используется как средство описания, оформления внешнего вида веб -страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к разным XML-документам, например, к SVG или XUL [6].

CSS они служат для того чтобы отделить друг от друга структуру страницы и ее содержание от ее внешнего вида.

Если страница полностью написана на HTML, то каждый элемент кода определяет не только элемент содержимого страницы, но и его способ отображения. При использовании CSS кода все происходит по-другому. С помощью HTML описываются только очередность элементов содержания страницы и их классы. В CSS файле прописаны соответствующие классы. Каждому присвоен набор свойств. Когда в HTML элементу приписываем какой-то класс, то к нему применяются все свойства этого класса. Не надо писать все свойства каждый раз. Без CSS невозможно представить современную веб-разработку. Общепринятым современным стандартом считается версия CSS 3.0 [7].

Ко всему прочему, следует заметить, что таблица стилей не является частью HTML. Это связано с тем, что всемирная сеть (www) была создана учеными как средство для обмена текстовыми документами и HTML был языком, посредством которого писали эти документы. Для ученых, основное содержание документа было превыше внешнего вида.

2.3 DHTML – динамический HTML

DHTML (динамический HTML) – это набор средств, которые позволяют создавать более интерактивные веб-страницы без увеличения загрузки сервера. DHTML – это не какой-то специальный язык разметки и не модифицированный HTML. Понятие DHTML основывается на трех элементах: собственно на языке

HTML, на каскадных таблицах стилей CSS и на встроенном в браузер языке – преимущественно это JavaScript т.к. является многосторонним.

DHTML построен на объектной модели документа (Document Object Model, DOM), которая расширяет традиционный статический HTML-документ. DOM обеспечивает динамический доступ к содержимому документа, его структуре и стилям. В DOM каждый элемент веб-страницы является объектом, который можно изменять. DOM не определяет новых тэгов и атрибутов, а просто обеспечивает возможность программного управления всеми тэгами, атрибутами и каскадными листами стилей (CSS).

События динамического HTML.

Динамический HTML непосредственно связан с основной структурой документа и предоставляет собой мощную модель событий. Модель событий динамического HTML основана на двух элементах, служащих для управления поведением документа: «всплывание» событий (event bubbling) и действия по умолчанию (default actions) [7].

Всплывание событий - это элемент модели событий, при помощи которого обеспечивается просмотр структурной иерархии документа. На любое событие может последовать реакция от всех родительских элементов в иерархии контейнеров, а также от элемента, в котором возникло событие. Другими словами, каждое действие, происходящее в элементе, его родительском элементе и так далее вплоть до тела документа, и самого документа, получает извещение события. Событие может быть обработано на любом уровне, что позволяет создавать компактные настраиваемые программы.

Действия по умолчанию обеспечивают встроенную в браузеры обработку события. Большинство событий позволяют заменять действия по умолчанию на индивидуальную обработку или добавлять дополнительные процедуры обработки.

2.4 JavaScript (язык программирования)

Если обратиться к истории и посмотреть немного назад, то выяснится, что раньше Web-сайты были статичными. Другими словами, они просто показывали страницу с заранее созданным контентом из текста и изображений, которая никак не изменялась от действий пользователя. Таким образом, пользователи могли щелкать по элементам, нажимать на картинки и ничего не происходило, страница никак не изменялась. Только в случае нажатия на ссылку, вы переходили на другую страницу.

Сейчас ситуация кардинально другая, практически все сайты в той или иной степени динамические и реагируют на ваши действия на странице, могут подстраиваться под ваш компьютер или вообще быть полноценным веб-приложением. Одним из самых распространенных способов достижения данного результата является использование JavaScript на страницах сайта.

JavaScript — это сценарный язык программирования, выполняемый непосредственно в браузере. Программы, написанные на нем, называют

скриптами, и они интегрируются непосредственно в HTML страницу различными способами. Результат выполнения сценариев отображается на странице, вызывая какие-то изменения на ней, а может и не отображаться, если это не предусмотрено.

Во избежание недоразумений нужно запомнить, что JavaScript и Java это совершенно разные языки программирования и кроме схожего названия у них не так много между собой общего. Чаще всего код на языке JavaScript обрабатывают обычные браузеры. Практически все современные браузеры поддерживают JavaScript, например Internet Explorer, Opera, Google Chrome, FireFox и др. В браузерах есть специальный модуль, который может обрабатывать команды, написанные на этом языке и приводить их в понятный вид [8].

В браузере пользователя исполняется код JavaScript, а браузеры еще называют «клиентами», то этот язык программирования еще называют клиентским. Этим он отличается от так называемых серверных языков веб-программирования, которые выполняются на сервере.

Так же браузеры разных разработчиков или даже разные версии одного и того же браузера порой по-разному интерпретируют одну и ту же страницу и тогда можно с помощью JavaScript определить используемый обозреватель и на лету изменить код страницы, чтобы она отображалась корректно. Добавить новый комментарий или обновить их без перезагрузки всей страницы, так же делается с его помощью [9].

В мире есть большое количество самых разнообразных устройств, с которых можно просматривать сайты и у них всех разные размеры экранов. Скрипт присутствующий на страницах этого сайта, проверяет разрешение монитора и подстраивает ширину страницы под его разрешение. На некоторых страницах в тексте есть миниатюры изображений, если щелкнуть по ним мышкой, то всплывет полноразмерное изображение. Это делается с помощью скриптов, без них изображение будет загружаться в отдельном окне или вместо текущей страницы, что довольно неудобно.

JavaScript еще, например, нужен для проверки форм на корректность заполнения без отправки данных на сервер, то есть сразу выводится сообщение, что поле заполнено неверно или например, что логин уже занят. Различные слайдеры, спойлеры тоже работают с помощью скриптов. Как и каждый язык программирования, он умеет выполнять математические операции, это в том числе используется в различных калькуляторах и конвертерах. Например, в онлайн-калькуляторе для пересчета единиц измерения информации.

Вообще, JavaScript довольно мощный язык, на котором можно создавать целые веб-приложения.

2.5 PHP (язык программирования)

Преимуществами этого языка являются его простота, а следовательно лёгкое освоение, возможность создания HTML документов с внедренными

командами RHP, создание качественных веб-приложений за очень короткие сроки, получение продуктов, легко модифицируемых и поддерживаемых в будущем.

На RHP написаны движки для сайтов, темы, плагины, различные формы связи (комментарии, почтовые программы) и многое другое.

RHP истоки лежат в старом продукте, имени RHP / FI. Последний был создан в 1995 году Расмусом Лердорфом, содержал в себе набор сценариев Perl и была предназначена для ведения статистики посещений его резюме. Именовался этот набор скриптов «Инструменты для домашних страниц». Вскоре приняли целый ряд функций, и Расмус создает гораздо большую версию на C, в работе с базами данных и позволяет пользователям разрабатывать простые Web-приложения. Расмус решил поместить исходный код RHP / FI на всеобщее обозрение, то есть для того чтобы предусмотреть будущие исправления ошибок и их последующего дополнения.

RHP / FI (Personal Home Page / Forms Interpreter – индивидуальная домашняя страница / Интерпретатор форм) содержит основную функциональность сегодняшнего RHP. Он имел тип переменной Perl, автоматическую интерпретацию формы и способность быть встроен в HTML код. Собственно, синтаксис языка, имел достаточно много общего с Perl, несмотря на то, что был намного легче и более ограниченным.

В 1997 состоялся релиз RHP / FI 2.0, второй версии C-имплементациями назначенной группы пользователей: десятки тысяч людей по всему миру, и если округлить, то примерно речь идет о 50,000 доменах, что составляло около 1% всего числа доменов в Интернете. Не глядя на то, что развитие нескольких человек сделало, RHP / FI 2.0 еще более крупным используемым продуктом для всех веб-разработчиков.

Официально RHP / FI 2.0 был выпущен в ноябре 1997 года. Большую часть своей жизни провела в бета-версиях. Вскоре после чего так и была заменена на альфа-версию RHP 3.0.

RHP 3.0 был первой версией, которая очень похожа на RHP, как все его знают сегодня. В 1997 году Энди Гутманс и Зив Сураски они переписали код с начала: разработчики пришли к выводу и сочли что RHP / FI 2.0 не подходит для дальнейшего развития, они работали над проектом для университета.

Один из самых сильных особенностей RHP 3.0 была возможность расширения ядра. Позже процесс написания интерфейса был обречен. На RHP обратили внимание многие разработчики сторонних производителей, работающих над своими модулями, RHP имеет возможности работать с большим количеством баз данных, протоколов к большому количеству поддержке API. На самом деле, это было самым важным ключом к успеху, но стоит добавить, что важным шагом стало развитие новой, гораздо более мощный и полный синтаксис при поддержке ООП.

Совершенно новый, язык программирования имеет новое имя. Разработчики отказались, от дополнения личного использования, и оставили аббревиатуру RHP / FI. Он был назван просто «RHP» аббревиатура которая

содержит рекурсивный акроним: «PHP: Hypertext Preprocessor» (PHP: препроцессор гипертекста).

В 1998 году PHP был использован десятком тысяч пользователей. Сотни тысяч Web-сайтов, которые работали с языком. В то время как PHP 3.0 установлен на около 10% Интернет-серверов.

PHP 3.0 официально выпущен после 9 месяцев публичного тестирования в июне 1998 года.

К зиме 1998 года, вскоре после официального релиза PHP 3.0, Энди Гутманс и Зеев Сураски начали переписывать ядро процессора PHP. Цели были для повышения производительности сложных приложений и улучшение модульности PHP кода. PHP 3.0 расширение предоставила возможность успешно работать с несколькими базами данных возможность поддерживать намного больше количество различных API и протоколов, но PHP 3.0 не был предназначен для обработки таких сложных приложений и работал не стабильно.

Новый «движок», под названием «Zend Engine» (по именам: Zeev и Andi, Энди Гутманс и Зеев Сураски), успешно справлялись с поставленными задачами и впервые был представлен в середине 1999 года. PHP 4.0, основанный на этом движке, а в сочетании с рядом нововведений и дополнительных функций, официально был представлен и вышел в мае 2000 года, почти через два года после своего предшественника, PHP 3.0. В дополнение к улучшению производительности, PHP 4.0 имеет несколько важных функций, таких как поддержка сессий, буферизация вывода, более безопасные способы обработки вводимой пользователем информации несколько новых языковых конструкций.

Сегодня PHP используется сотнями тысяч разработчиков. Несколько миллионов Web-сайтов работают с PHP, что составляет более 20% интернет - доменов. Команда разработчиков PHP состоит из множества людей. Для расширения ядра и PHP и смежных проектов, таких как PEAR и документации языка.

Пятая версия PHP разработчиков была выпущена 13 июля 2004 года. Изменения включают в себя новое ядро Zend (Zend Engine 2), новый «движок» повышает эффективность интерпретатора. Введена поддержка языка разметки XML.

Полностью переработаны особенности ООП, были очень схожи модели используемые в Java. В частности, добавлен деструктор, открытые, закрытые и защищенные члены и методы, окончательные члены и методы, интерфейсы и клонирование объектов. Тем не менее, инновации являются, с ожиданием, что поддерживать высокую совместимость с более ранними версиями кода на языке. Самая стабильная и наиболее часто используемая версия 5.xx.

Шестой вариант PHP находился в стадии разработки с октября 2006 года, он имеет ряд инноваций, таких как исключение из регулярных выражений POSIX. Удаление руководств `safe_mode` `PHP_magic_quotes` и `register_globals`

файл конфигурации `php.ini`. Кроме того, внимание устремлено на поддержку Unicode.

В 2014 году было проведено голосование, по результатам которого следующая версия получила название PHP 7. Выход новой версии планировался в середине октября 2015 года. В марте 2015 года Zend представили инфографику в которой описаны основные нововведения PHP 7.

3 декабря 2015 года было объявлено о выходе PHP версии 7.0.0 [10].

Новая версия основывается на экспериментальной ветке PHP, которая изначально называлась `phpng` (PHP Next Generation - следующее поколение), и разрабатывалась с упором на увеличение производительности и уменьшение потребления памяти. В новой версии добавлена возможность указывать тип возвращаемых из функции данных, а также добавлен контроль передаваемых типов для скалярных данных.

Таким образом, PHP способен осуществлять все что делает любая другая программа CGI. Например, чтобы редактировать данные формы, генерировать динамические страницы, получать и отправлять cookies.

2.6 MySQL

База данных (БД) вещь полезная для ведения сайта. Работать со статическим проектом, состоящим из нескольких `html` страничек легко и без использования БД. Однако сайты разрастаются. С динамичным проектом такой метод работы уже себя не оправдывает. Хранить массивы различной информации в сотнях файлов, а потом при работе веб-сервера требовать от них нужные строки – дело медленное. Базы данных позволяют структурировать и систематизировать информацию. Код для использования БД намного легче, чем аналогичный для работы с файлами, да и времени на запрос уходит меньше.

База данных (database) – это совокупность связанных между собой таблиц. Например, в одной таблице храниться информация о пользователе, зарегистрированном на сайте, а в другой – информация о данных, которые оставил пользователь на сайте [11].

Наилучшее решение – хранить информацию в БД, например списков, комментариев и т.д. Однако база данных далеко не всегда статичное образование, следовательно она регулярно пополняется и корректируется. Для легкости управления БД, изменения и добавления данных существуют системы управления базами данных (СУБД).

Одна из самых популярных СУБД в современных интернет-технологиях, бесспорно, MySQL.

К основным плюсам MySQL можно отнести высокую скорость работы, быстроту обработки данных и оптимальную надежность. Немаловажно и то, что данная СУБД распространяется бесплатно и представляет собой программное обеспечение с открытым кодом. За счет этого можно вносить свои изменения и модифицировать код, что очень полезно для веб-мастеров.

Однако, MySQL – это один из множества ПО для работы с SQL базами данных.

SQL – это структурированный язык запросов, созданный для управления БД. Он обладает широким перечнем возможностей, например, создать таблицу, редактировать и удалять данные, производить запросы из таблиц и многое другое.

Начиная с пятой версии MySQL поддерживает хранимые процедуры и функции, обработчики ошибок, курсоры, триггеры, представления. В MySQL используется немного «подрезанный» стандартный SQL. MySQL поддерживает очень большой список платформ. Надо сказать, что почти все хостеры предоставляют хостинг вместе с MySQL – поэтому она и является такой популярной. Но существует множество СУБД, способных работать с сайтами, к примеру, ORACLE поддерживает все вышеописанные функции, но в ней есть еще ряд особенных функций: объектно-ориентированные свойства, автономные транзакции, последовательности, пакеты, аналитические функции.

Для работы можно использовать как стандартный SQL, так и язык программирования PL/SQL – в него добавлено множество дополнительных функций, упрощающих работу программиста. Кроме того, последняя на данный момент версия СУБД ORACLE – это первая в мире база данных, специально разработанная для работы в сетях распределенных вычислений Grid. ORACLE, также, как и MySQL – поддерживает очень много различных платформ.

3 АНАЛИЗ, ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ САЙТА

3.1. Обзор и анализ методов проектирования сайта

Технологический прогресс и всемирная сеть способствовали значительному увеличению объема информации, углублению и расширению наших представлений об окружающем мире. До сих пор основным способом предъявления полученных новых научных знаний была публикация научных статей и разработок ученых. Гипермедиа ресурсы позволили осуществить быстрый доступ ко всей научной информации, размещенной в Интернете. Научные дискуссии, обмен мнениями и критическими замечаниями могут быть реализованы в режиме реального времени, что, несомненно, способствует более быстрому продвижению научных открытий в реальную жизнь. Сфера интернет-технологий является одной из наиболее динамично развивающихся направлений обработки информации. Технологии не успевают получить своего полного распространения, как их сменяют другие, более совершенные.

В связи с этим появляется большое количество информационных сайтов, на которых можно найти практически любую информацию. Интернет предоставляет новые возможности в области образования и науки.

Перейдем к проектированию разработки. Итак, существуют различные методы проектирования сайта. Проектирование сайта предполагает совмещение некоей предметной концепции сайта (область применения, цели и задачи, аудитория) с актуальным функциональным наполнением и контентом, сроками и стоимостью разработки. Рассмотрим основные методы, используемые в современном «сайтеостроении». Надо отметить, что методы проектирования сайтов мало чем отличаются от обычных методов проектирования, принятых в инженерной или творческой среде, и, чаще всего, наилучшего результата можно достичь используя сочетание различных подходов. Рассмотрим три метода – Метод анализа существующих решений, Метод морфологической карты и Модель Диснея.

Метод анализа существующих решений, используется на этапе анализа проектной ситуации. Исследуются существующие аналоги с целью выявления недостатков - визуальных, функциональных и конструкционных.

Важнейшим является изучение образцов или изображений аналогов. Здесь есть два способа: первый - визуальное изучение объекта, второй – его изучение в процессе эксплуатации. Второй способ значительно предпочтительнее. По сути, таким образом проводится нечто вроде юзабилити-тестирования, и данное исследование в нашем случае будет более объективным.

На этом этапе важно определить визуальные несоответствия между элементами объекта, а также между объектом и средой, которые в свою очередь появляются по причине несовершенства конструкции, то есть могут служить индикатором конструкционных проблем. Выявить функциональные недостатки

и противоречия (тем не менее более достоверные результаты возможно получить только при условии эксплуатации объекта).

Определить причины несоответствий и противоречий. Это поможет вам выяснить, сможете ли вы разрешить возникшие противоречия (какие-то причины могут оказаться неустраняемыми).

Наметить пути разрешения противоречий, что является переходным этапом к поиску решений, поэтому важно, намечая пути, оставить большое пространство для маневра, предоставить больше гибкости. Результат проведенного анализа позволяет сформулировать задачи, которые будут решаться в проекте.

Метод морфологической карты используется на начальном этапе непосредственного проектирования, после предпроектного анализа, когда уже определена проектная ситуация и выделены проблемы, которые следует решать. Метод позволяет наглядно представить поле поиска возможных решений и служит гарантией того, что ни одно из новых возможных решений не будет упущено. С целью расширить область поиска решений проектной проблемы и найти варианты наиболее удачных.

Порядок действий здесь отличается. Для начала необходимо определить важнейшие функции или функциональные признаки объекта. Разместить найденные функции и признаки на одной из осей карты. То, чем будет заполняться карта (функциями или признаками) зависит от типа объекта и целей проектирования. Для этого нужно очень хорошо представлять свою проектную ситуацию и иметь четкую формулировку проблемы. Чем больше мы знаем о проектируемом объекте, тем проще составить карту [12].

Для каждой функции или признака перечислить максимум возможных вариантов частичных решений. Разместить частичные решения по другой оси в соответствующих колонках (столбцах). Частичное решение - это решение какого-либо одного аспекта всей проблемы. В идеале нужно записать такие варианты частичных решений. В каждом ряду решений можно добавить вариант "прочее", оставив тем самым себе свободу для маневра.

Выбрать по одному приемлемому частичному решению для каждого признака и функции. Для удобной ориентировки разные комбинации выделяются разным цветом. Можно так же отмечать выбранные решения точками и соединять их ломаными линиями - разных цветов, толщины, конфигурации. Комбинаций вариантов может быть достаточно много. Чем больше функций и частичных решений, тем быстрее растут возможности комбинаторики, достигая огромных чисел.

Как результирующее, получается несколько вариантов решения проектной проблемы. Но необходимо использовать ту комбинацию, которая максимально соответствует требованиям проектной ситуации.

Модель Диснея как метод проектирования сайта

Для разработки и создания сайта существует несколько моделей, которые позволяют поэтапно, шаг за шагом, реализовать свой проект от идеи до ее воплощения. Какую модель предпочесть зависит, в общем, только от

разработчиков и тех целей, которые они преследуют. Для коммерческого проекта, например, нужно рассчитать все затраты на разработку и выполнить проект качественно и в срок. В этом случае требуется подход, позволяющий максимально быстро, при этом на достаточно высоком уровне, сделать сайт и сдать его заказчику.

Для большинства разработчиков вполне подойдут простые модели проектирования, которые четко определяют цели сайта и описывают его внутреннюю структуру. Одной из примечательных с этой позиции методик можно считать «Модель Уолта Диснея». Разумеется, сам Дисней не занимался никаким созданием сайтов, даже не знал, что это такое, но он сочетал в себе умение превращать образы воображения в реальный результат. Кроме того, Дисней был не только талантливым аниматором, но и весьма удачливым бизнесменом, успешно сочетая новаторское творчество с деловой активностью.

Надо признаться, выбранное название не является устоявшимся, но непосредственно связано со стратегией мышления Диснея, которая вполне действенно может применяться при проектировании сайтов. Вдобавок, упоминание имени Диснея в названии модели несет позитивный смысл, поскольку связано у людей, как правило, с положительными эмоциями.

Модель Уолта Диснея используется для проектирования сайта и состоит из трех этапов. В терминах модели эти этапы называются фазами мечтателя, реалиста и критика. Однако, нужно придерживаться других названий, поскольку они более точно отражают суть процесса разработки сайта. Итак, процесс проектирования разбивается на следующие стадии:

1. концептуальное проектирование;
2. логическое проектирование;
3. физическое проектирование.

Этапы следуют последовательно один за другим, но в некоторых случаях возможен переход к следующей стадии без окончания предыдущей. Это может происходить, например, когда разработчиков несколько и каждый работает со своей частью сайта. В любом случае, после окончания этапа физического проектирования следует вернуться к началу и внести соответствующие коррективы.

Для лучшего понимания каждая часть проектирования описана более подробно.

Концептуальное проектирование

Порой бывает сложно оценить эффективность сайта. Действительно, какими критериями воспользоваться, чтобы понять, что сайт, над которым столько корпели и тратили на него свое время, действительно хорош. Разумеется, можно использовать количество посетителей как некоторое мерило. Но этот показатель относителен и зависит от множества внешних параметров. То же самое обстоит и с другими способами оценки сайта.

Есть однако универсальный критерий, который довольно точно характеризует эффективность сайта. Это достижение разработчиков сайта

поставленных перед ними целей. В этом случае сайт превращается в действенный инструмент, который выполняет возложенные на него функции.

Концептуальное проектирование служит для указания целей, задач сайта и определения аудитории, на которую он рассчитан.

На этом этапе учитывается следующее:

1. основные и второстепенные цели;
2. действия, которые необходимо предпринять для достижения поставленных целей;
3. состав пользователей;
4. интересы групп пользователей;
5. разделы сайта;
6. критерии достижения цели.

С учетом интересов пользователей и поставленных целей в итоге получаем список сервисов и разделов, которые будут располагаться на сайте.

Логическое проектирование

Разделы сайта, продуманные на предыдущем этапе, пока не упорядочены и не структурированы, поэтому их нужно привести к удобному и понятному виду. Логическое проектирование включает организацию информации на сайте, построение его структуры и навигации по разделам.

На данном этапе следует задаться вопросом, каким образом будет упорядочена информация. Варианты могут быть самыми разными и зависеть от типа данных и предпочтений создателей сайта: по времени, разделам, в алфавитном порядке, определенным группам или другим критериям. Одновременное использование различных способов охватывает большую аудиторию и позволяет быстрее найти нужную информацию на сайте.

В этом этапе необходимо описать следующее:

1. тип структуры сайта (линейная, иерархическая, контекстная, другая);
2. названия разделов;
3. что будет содержать в себе каждый раздел;
4. организация и связь разделов между собой;
5. какая информация будет размещена на определенных страницах сайта.

Конечный результат логического проектирования оформляется в виде блок-схем, структурных диаграмм или другими способами, показывающими взаимосвязь различных частей сайта.

Физическое проектирование – это этап поиска проблем, а не их решений, связанных, по большей части, с технической реализацией сайта.

Данный этап предполагает следующее:

1. технологии, которые будут применяться на сайте;
2. используемое программное обеспечение;
3. возможные проблемы и способы их устранения;
4. как будет обновляться информация.

После завершения данного этапа следует вернуться к концептуальному проектированию и проверить, не нужно ли внести изменения в связи с переосмыслением проекта на других стадиях.

Так, если на сайте планируется использовать базы данных и доступ к данным с помощью CGI, следует подумать о хостинге, который поддерживает выбранные технологии. Возвращаясь к концептуальному проектированию, ставим цель разместить сайт на платном хостинге, затратив на это определенную сумму денег, при этом получая взамен дополнительные возможности. Либо зажав наши кровные окунуться в мир бесплатного хостинга. При этом часть возможностей просто теряется.

Эффективность данной модели обусловлена четко поставленными целями сайта, удовлетворением интересов его посетителей и направлена на разработку понятной и четкой внутренней организации содержимого сайта для того, чтобы его посетители могли легко найти нужную им информацию.

Предложенная методика может применяться не только для разработки сайта, но и с успехом использовалась для проектирования других сетевых работ.

3.2 Анализ ведущих Интернет-сайтов России

Слово «Web-сайт» происходит от английского website, где web – паутина, а site — «место», то есть «место в паутине» так сказать.

Википедия говорит, что сайт – это совокупность электронных файлов организации или частного лица во всемирной компьютерной сети, которые объединены одним доменным именем, то есть адресом.

Первое «совокупность электронных файлов». Каждый сайт состоит из множества файлов, которые взаимодействуют друг с другом. Например, в какой-либо html-странице указывается, где находится css-файл со стилями для этой страницы, то есть где «написано, как она должна выглядеть» – все файлы связаны друг с другом.

Второе – «организации или частного лица». Это значит, что каждый сайт принадлежит определенной организации или физическому лицу. И последнее – «объединены одним доменным именем, то есть адресом». Как все знают, у каждого сайта есть свой адрес, например, site.ru. Этот адрес и называется доменным именем. Доменное имя есть у каждого сайта без исключения. Оно состоит из имени (в нашем случае «site»), а также доменной зоны («.ru»). С определением разобрались.

Сайты бывают статистические и динамические.

Статистические сайты состоят из статистических страниц. Каждая из них храниться на сервере в своем первоначальном виде. Посетитель видит ее именно такой, какой она есть на сервере. Например, загружаем себе на компьютер картинку. После этого загрузили ее в какую-то социальную сеть. Открыв картинку в этой социальной сети, видим ее такой, какой она была на

компьютере. То есть в её первозданном виде. Вот так и страница, как хранилась на сервере, так и вывелась на экран пользователя.

Динамический сайт состоит из частей страниц, формирующих конечную страницу, которую видит пользователь [13].

Представим, что есть несколько файлов `shapka.php`, `podval.php`, `content.php` и `sidebar.php`. Пользователь заходит на сайт и видит готовую страницу, которая состоит из этих 4 файлов:

Получается, что страница, которую увидел пользователь, автоматически сгенерировалась. Изначально такой страницы на сервере не существует, существуют только ее части.

Также динамическая страница может быть сгенерирована путем введения каких-то данных пользователем. Например, при переписке в социальных сетях видим страницу с сообщениями. Естественно, что изначально такой страницы на сервере не было, но в результате работы скрипта она появилась.

В современном Интернете распространены именно динамические сайты, так как они более функциональны и удобны в пользовании. Статистические практически не используются, но именно с них все и начиналось.

Представим, что на сайт в 100 страниц нужно установить рекламный баннер. Если сайт статистический, то нужно будет открыть 100 файлов и в каждый в нужное место вставить код. Если же сайт динамический, то достаточно будет открыть один файл, который отвечает за то место, где должен быть расположен баннер, и вставить туда код. Преимущество динамических сайтов очевидно.

Окончательно убедившись в бесперспективности сайта со статическим содержанием, как более-менее серьезного проекта теперь будем рассматривать только динамический PHP сайт. Язык программирования PHP широко используется для создания веб-проектов разной сложности, это очень демократичный язык и позволяет создать рабочий динамический PHP сайт как любительского, так и профессионального уровня. Другими словами, работать с PHP можно легко научиться, но гораздо сложнее делать это профессионально. Возможно трудно понять суть этого подхода, но привыкать к нормам качественного написания программного кода нужно в самом начале. PHP как таковой позволяет даже слишком фривольно писать код и такие программы тоже будут работать, отсюда и популярность языка у начинающих и низкоквалифицированных программистов.

Виды сайтов

Сайты разделяют на:

1. сайты-визитки;
2. промо-сайты;
3. информационные сайты;
4. корпоративные сайты;
5. порталы.

Сайт-визитка

Сайт-визитка – это не большой сайт, который состоит из нескольких страниц. Сайт-визитка представляет какую-то кампанию, как бы являясь его визиткой (откуда и название). Это полноценный сайт с собственным уникальным дизайном и программной частью.

Почему фирма должна иметь свой сайт-визитку.

Сайт-визитка помогает компании находить новых клиентов в сети Интернет.

Экономит бюджет компании на обычных методах рекламы (на рекламных материалах размещается ссылка на сайт, куда посетитель может зайти и узнать какую-то дополнительную информацию).

У компании появляется возможность работать с клиентами более качественно. Ведь к сайту, в отличие от офиса, можно обратиться в любом месте в любое время.

Сайт-визитка представляет собой элемент имиджевой рекламы. Хорошо разработанный и привлекательный Web-сайт может вызвать у посетителя положительные эмоции и сформировать позитивный взгляд и доверие к компании и ее деятельности.

Получение прибыли. При большой посещаемости владелец ресурса сможет продавать рекламные места на сайте под баннеры или контекстную рекламу.

Не высокая цена. Как правило, сайт визитка ценится намного дешевле, чем любой другой сайт.

Промо-сайт

Промо-сайт – не большой (до 5-ти страниц) сайт, главной задачей которого является эффективная реклама продукта или услуги, имиджа или бренда компании. Сегодня ни одна крупная компания не обходится без создания такого сайта, ведь его разработка – важная веха на пути к успеху. В основном, такие сайты воздействуют на определенную часть целевой аудитории. Главная задача ресурса – стимулировать человека к какому-то действию, например, к покупке товара.

Промо-сайт обычно маленький по объему, но всегда имеет эксклюзивный, броский и запоминающийся дизайн. Используется огромное количество графики.

И наконец промо-сайт – незаменимый по своей результативности маркетинговый инструмент.

Информационный сайт

Название говорит само за себя. Информационный сайт – это сайт, который предоставляет пользователю какую-либо информацию. Именно информационные сайты являются наиболее популярными в современном Интернете. Информационный сайт это что-то среднее между сайтом-визиткой и корпоративным сайтом (о нем далее). В сравнении с первым имеет более сложный и эксклюзивный дизайн, более широкие возможности, например, систему покупок, большее число страниц. Основная задача информационного

сайта – предоставление полной и, что очень важно, постоянно обновляющейся информации.

Корпоративный сайт

Основная задача, которую должен выполнять корпоративный сайт – донесение информации о компании, ее деятельности, предлагаемых услугах и производимых товарах «к ушам» как можно большей аудитории. На сегодня корпоративный сайт – это обычный ресурс, который помогает решать маркетинговые задачи, которые компания ставит перед собой, а также сложные системы, постоянно взаимодействующие с внутренними системами компании: системами управления клиентами, базами данных и пр.

Портал

Портал – сайт с большим объемом информации и многоуровневой структурой. Порталы имеют огромную постоянную посещаемость.

Порталы могут решать сложные коммерческие задачи. Он включает в себя свойства таких сайтов: сайт-визитка, промо-сайт, интернет-магазин и корпоративный онлайн ресурс [14].

Разобрав виды сайтов, и теперь, в качестве первого шага проанализируем существующие сайты в предметной области (флористика), определим основные тренды, попробуем понять что в них хорошо, функционально, а что не очень. Для анализа будем использовать поисковую выдачу Яндекса по запросу «студия флористики» по Москве, которая покажет нам самые популярные в Интернете сайты студий флористики. Среди них встречается много интернет-магазинов с доставкой букетов, функционал которых выходит за рамки работы, поэтому отбираем только сайты-визитки.

1) <http://kristina-ageeva.com/> - типичный сайт-визитка с акцентом на фотографии выполненных работы флориста (портфолио), текстовая информация представлена минимально, дизайн сайта минималистичен. На сайте есть раздел контакты с контактными данными, картой и формой обратной связи. Предусмотрены ссылки на социальные сети, оформленные в виде иконок.

2) <http://florellino.ru/> - выразительный сайт-визитка с акцентом на фотографии выполненных работы флориста (портфолио), текстовая информация представлена минимально, дизайн сайта так же минималистичен. На сайте есть раздел контакты с контактными данными, картой и формой обратной связи. Предусмотрены ссылки на социальные сети, оформленные в виде иконок и виджетов. Удачно оформлена главная страница, которая позволяет оценить студию и связаться с ней без перехода на другие страницы. Простая и понятная навигация.

3) <http://www.med-flowers.ru/> - сайт-визитка выглядит архаично, много текстовой информации на главной странице, которая в этой сфере практически не нужна, все впечатление об уровне студии можно и должно получить визуально. Запутанная навигация со множеством ссылок, пользователь не захочет оставаться на таком сайте.

4) <http://coralcharm.ru/> - крайне невыразительный сайт-визитка, который скорее отталкивает отсутствием дизайна, темно-серыми тонами, главная страница не информативна.

5) <http://forget-me-not.ru/> - яркий интересный сайт, с простым и понятным, лаконичным наполнением. Навигация не вызывает вопросов, представлены все основные функциональные элементы современного сайта-визитки: ссылки на социальные сети, форма обратной связи, карта местонахождения, контактные данные, портфолио и описание работ флористов и декораторов.

На данном этапе хотелось бы выделить сайт <http://florellino.ru/> как наиболее отвечающий понятию сайта-визитки, с современным набором функционала и соответствующими теме дизайном и представлением контента.

3.3 Проектирование сайта и классификация пользователей проекта

Целью сайта <http://kseniya-flower.ru/> является интернет представление студии флористики и праздничного декора «Ксения», чтобы потенциальные клиенты могли оценить качество услуг и принять решение об обращении в студию.

На основании целей сайта и анализа ведущих интернет-сайтов в области флористики, формируем требования к сайту. Анализ ведущих интернет-сайтов выявил тенденцию к визуальному, а не текстовому представлению информации в данной области применения, назовем данное требование «информативностью».

Тенденция к максимальному облегчению навигации по сайту – в свете последних изменений в клиентских устройствах, с которых осуществляется просмотр интернет-сайтов, происходит смещение в сторону мобильных устройств, следовательно, навигация по сайту должна быть удобна на низких разрешениях и нестандартных пропорциях экранов мобильных устройств. Данное требование к сайту называется «адаптивность».

На сайте должна быть представлена полная контактная информация с возможностью связаться со студией прямо с сайта (форма обратной связи).

Меню навигации должно быть простым и понятным.

Работы студии должны быть представлены в виде фотографий с описаниями, так же работы должны быть разделены на категории, для облегчения поиска нужной информации. Реализуем данный функционал простым фильтром отбора работ по категориям.

На сайте должен быть представлен поисковый механизм, позволяющий показать релевантную информацию с сайта по поисковому запросу.

Исходя из вышеперечисленных требований к сайту, планируем создать 5 основных разделов сайта:

1) Главная – страница, дающая первое представление о студии, показывающая примеры работ, возможности студии, команду флористов,

контактные данные и наделенная функциями навигации (меню, поиск, прямые ссылки).

- 2) О нас – страница с развернутой информацией о студии.
- 3) Портфолио – примеры работ с фильтром отбора по категориям.
- 4) Новости – освещение деятельности студии, участие в проектах, интересные новые работы, которые захочется выделить, события, которыми захочется поделиться с клиентами.
- 5) Контакты – данные о местоположении студии, средствах связи (телефон, электронная почта, форма обратной связи)

В целях облегчения навигации по разделам на сайте должно быть меню первого уровня, состоящее из следующих пунктов:

- 1) Главная – ссылка на главную страницу сайта
- 2) О нас – ссылка на страницу «О нас»
- 3) Букеты – категория «Букеты» из раздела «Портфолио»
- 4) Декор – категория «Декор» из раздела «Портфолио»
- 5) Новости – ссылка на страницу новостей студии
- 6) Контакты – ссылка на страницу «Контакты»

Данный сайт предназначено для следующих категорий пользователей:

1) Администратор сайта – пользователь, имеющий полные права на сайт, доступ к изменению контента сайта, стилей, скриптов. Для осуществления администрирования сайта этот пользователь должен пройти идентификацию (ввести имя и пароль) на сайте по специальному URL. Делается это с целью защиты данных сайта от несанкционированного доступа.

2) Гость - этот пользователь может только просматривать информацию на сайте, заполнять и отправлять форму обратной связи.

3.4 Принципы реализации элементов дизайна сайта

В целях обеспечения простоты навигации основное меню сайта располагаем вверху и «прибиваем» его к верху экрана, чтобы оно постоянно было доступно пользователю. Меню располагаем на полупрозрачной подложке, чтобы не «резать» контент.

Поле поиска по сайту располагаем в традиционном и привычном для пользователей месте – справа в верхнем углу экрана, в одной строке с меню. Кнопку запуска поиска оформляем в виде иконки.

Логотип сайта располагаем слева в «прибитой» шапке на одном уровне с основным навигационным меню.

Основной принцип контента – блочное построение, чтобы в различных разрешениях браузера эти блоки, их размер и порядок, могли подстраиваться под текущее разрешение, вплоть до минимального.

Контент должен быть представлен максимально наглядно с использованием фотографий или рисунков.

Для создания первого впечатления пользователей об уровне и качестве работ студии, на первом экране главной страницы должен быть большой слайдер баннеров, с мотивирующей надписью крупным шрифтом.

На главной странице необходимо разместить максимально большое количество информации, так как современные пользователи уже привыкли иметь дело с прокруткой экрана, особенно на мобильных устройствах, гораздо проще прокрутить страницу, чем попасть в ссылку меню или кнопку.

Использовать в оформлении формы обратной связи на главной странице «паралакс» эффект, для привлечения внимания к ней.

3.5 Описание структуры хранения данных

Основными информационными элементами сайта являются:

- 1) Портфолио – работа студии с названием и миниатюрой анонса, галереей изображений, текстовым описанием и видеоматериалами, категорией работы, поисковыми тэгами.
- 2) Команда – имя, фотография и описание сотрудника студии.
- 3) Новости – название, фотография, текст новости.
- 4) Пользователь – имя, логин, пароль (пользователь используется в данном сайте только для доступа в панель администратора с целью управления контентом)

Хранение данных реализовано в виде записей в таблицах CMS Wordpress в БД MySQL. Прежде чем разобраться, как данные о контенте хранятся в базе WordPress, рассмотрим стандартные типы контента Wordpress:

- Записи (posts)
- Страницы (pages)
- Пользовательские типы записей (custom post types)
- Вложения (attachments)
- Ссылки (links)
- Элементы меню (navigation menu items)

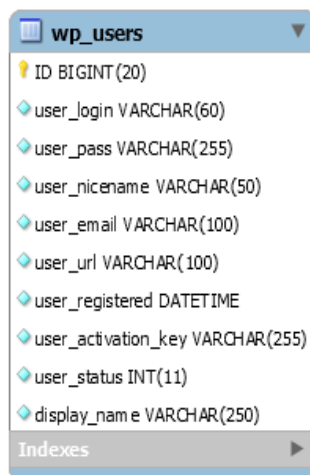
Эти типы контента имеют свойства:

- Категории (categories)
- Метки (tags)
- Пользовательские определения (custom taxonomies and terms)
- Метаданные (post metadata)

Кроме того, существуют дополнительные типы контента, хранящиеся в ином виде:

- Виджеты (widgets)
- Опции (options)
- Пользователи (users)
- Сайты для MU WordPress
- Нестандартный контент (hardcoded content), который добавляют некоторые темы/плагины.
- Сторонний контент (third party content) (например RSS)

Весь этот контент хранятся в таблицах базы данных или в файлах настроек тем и плагинов CMS Wordpress. Каждый тип контента может быть представлен как отдельной записью в таблице, так и отдельной таблицей (например слайдеры для слайдеров и сами слайдеры). Кроме, того они могут быть связаны с данными в других таблицах. Например, данные о записях связаны с данными о пользователях, так что WordPress знает, кто является автором записи.

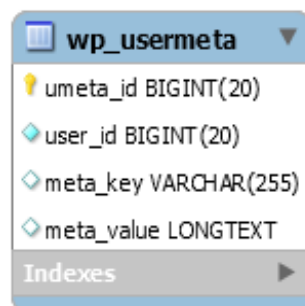


The screenshot shows the structure of the `wp_users` table. It lists the following fields:

Field Name	Field Type
ID	BIGINT(20)
user_login	VARCHAR(60)
user_pass	VARCHAR(255)
user_nicename	VARCHAR(50)
user_email	VARCHAR(100)
user_url	VARCHAR(100)
user_registered	DATETIME
user_activation_key	VARCHAR(255)
user_status	INT(11)
display_name	VARCHAR(250)

Below the fields, there is a section for "Indexes" with a right-pointing arrow.

Рисунок 3.1-Таблица данных о пользователях

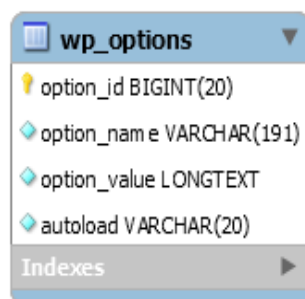


The screenshot shows the structure of the `wp_usermeta` table. It lists the following fields:

Field Name	Field Type
umeta_id	BIGINT(20)
user_id	BIGINT(20)
meta_key	VARCHAR(255)
meta_value	LONGTEXT

Below the fields, there is a section for "Indexes" with a right-pointing arrow.

Рисунок 3.2-Таблица метаданных о пользователе



The screenshot shows the structure of the `wp_options` table. It lists the following fields:

Field Name	Field Type
option_id	BIGINT(20)
option_name	VARCHAR(191)
option_value	LONGTEXT
autoload	VARCHAR(20)

Below the fields, there is a section for "Indexes" with a right-pointing arrow.

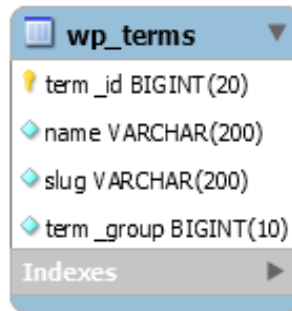
Рисунок 3.3-Таблица настроек и опция CMS WordPress

wp_posts	
ID	BIGINT(20)
post_author	BIGINT(20)
post_date	DATETIME
post_date_gmt	DATETIME
post_content	LONGTEXT
post_title	TEXT
post_excerpt	TEXT
post_status	VARCHAR(20)
comment_status	VARCHAR(20)
ping_status	VARCHAR(20)
post_password	VARCHAR(20)
post_name	VARCHAR(200)
to_ping	TEXT
pinged	TEXT
post_modified	DATETIME
post_modified_gmt	DATETIME
post_content_filtered	LONGTEXT
post_parent	BIGINT(20)
guid	VARCHAR(255)
menu_order	INT(11)
post_type	VARCHAR(20)
post_mime_type	VARCHAR(100)
comment_count	BIGINT(20)
Indexes	

Рисунок 3.4-Таблица для хранения записей новостей, команды, работ из портфолио

wp_postmeta	
meta_id	BIGINT(20)
post_id	BIGINT(20)
meta_key	VARCHAR(255)
meta_value	LONGTEXT
Indexes	

Рисунок 3.5-Таблица для хранения метаданных к записям




wp_terms

term_id	BIGINT(20)
name	VARCHAR(200)
slug	VARCHAR(200)
term_group	BIGINT(10)

Indexes

Рисунок 3.6-Таблица объектов

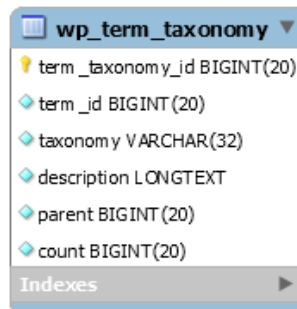


wp_term_relationships

object_id	BIGINT(20)
term_taxonomy_id	BIGINT(20)
term_order	INT(11)

Indexes

Рисунок 3.7-Таблица взаимосвязей объектов

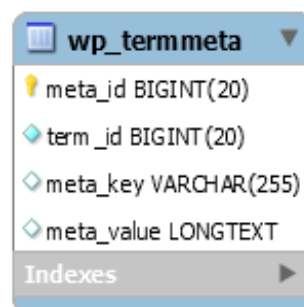


wp_term_taxonomy

term_taxonomy_id	BIGINT(20)
term_id	BIGINT(20)
taxonomy	VARCHAR(32)
description	LONGTEXT
parent	BIGINT(20)
count	BIGINT(20)

Indexes

Рисунок 3.8-Таблица классификации объектов



wp_termmeta

meta_id	BIGINT(20)
term_id	BIGINT(20)
meta_key	VARCHAR(255)
meta_value	LONGTEXT

Indexes

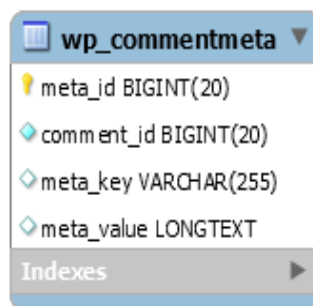
Рисунок 3.9-Таблица метаданных объектов

Column	Field Type
link_id	BIGINT(20)
link_url	VARCHAR(255)
link_name	VARCHAR(255)
link_image	VARCHAR(255)
link_target	VARCHAR(25)
link_description	VARCHAR(255)
link_visible	VARCHAR(20)
link_owner	BIGINT(20)
link_rating	INT(11)
link_updated	DATETIME
link_rel	VARCHAR(255)
link_notes	MEDIUMTEXT
link_rss	VARCHAR(255)

Рисунок 3.10-Таблица внешних ссылок

Column	Field Type
comment_ID	BIGINT(20)
comment_post_ID	BIGINT(20)
comment_author	TINYTEXT
comment_author_email	VARCHAR(100)
comment_author_url	VARCHAR(200)
comment_author_IP	VARCHAR(100)
comment_date	DATETIME
comment_date_gmt	DATETIME
comment_content	TEXT
comment_karma	INT(11)
comment_approved	VARCHAR(20)
comment_agent	VARCHAR(255)
comment_type	VARCHAR(20)
comment_parent	BIGINT(20)
user_id	BIGINT(20)

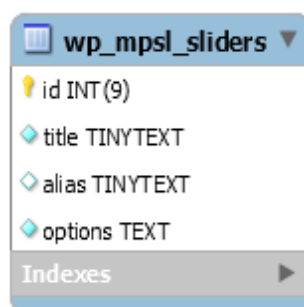
Рисунок 3.11-Таблица комментариев



The screenshot shows the structure of the `wp_commentmeta` table. It has four columns: `meta_id` (BIGINT(20) with a primary key icon), `comment_id` (BIGINT(20) with a foreign key icon), `meta_key` (VARCHAR(255) with a foreign key icon), and `meta_value` (LONGTEXT with a foreign key icon). Below the columns is a tab labeled "Indexes" with a right-pointing arrow.

Column Name	Data Type	Constraints
<code>meta_id</code>	BIGINT(20)	Primary Key
<code>comment_id</code>	BIGINT(20)	Foreign Key
<code>meta_key</code>	VARCHAR(255)	Foreign Key
<code>meta_value</code>	LONGTEXT	Foreign Key

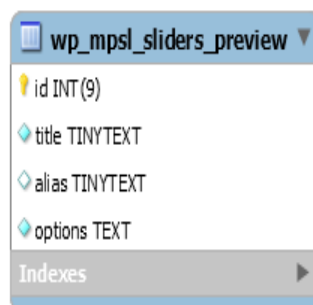
Рисунок 3.12-Таблица метаданных к комментариям



The screenshot shows the structure of the `wp_mpsl_sliders` table. It has four columns: `id` (INT(9) with a primary key icon), `title` (TINYTEXT with a foreign key icon), `alias` (TINYTEXT with a foreign key icon), and `options` (TEXT with a foreign key icon). Below the columns is a tab labeled "Indexes" with a right-pointing arrow.

Column Name	Data Type	Constraints
<code>id</code>	INT(9)	Primary Key
<code>title</code>	TINYTEXT	Foreign Key
<code>alias</code>	TINYTEXT	Foreign Key
<code>options</code>	TEXT	Foreign Key

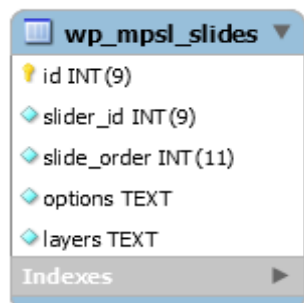
Рисунок 3.13-Таблица слайдеров



The screenshot shows the structure of the `wp_mpsl_sliders_preview` table. It has four columns: `id` (INT(9) with a primary key icon), `title` (TINYTEXT with a foreign key icon), `alias` (TINYTEXT with a foreign key icon), and `options` (TEXT with a foreign key icon). Below the columns is a tab labeled "Indexes" with a right-pointing arrow.

Column Name	Data Type	Constraints
<code>id</code>	INT(9)	Primary Key
<code>title</code>	TINYTEXT	Foreign Key
<code>alias</code>	TINYTEXT	Foreign Key
<code>options</code>	TEXT	Foreign Key

Рисунок 3.14-Таблица превью слайдеров



The screenshot shows the structure of the `wp_mpsl_slides` table. It has five columns: `id` (INT(9) with a primary key icon), `slider_id` (INT(9) with a foreign key icon), `slide_order` (INT(11) with a foreign key icon), `options` (TEXT with a foreign key icon), and `layers` (TEXT with a foreign key icon). Below the columns is a tab labeled "Indexes" with a right-pointing arrow.

Column Name	Data Type	Constraints
<code>id</code>	INT(9)	Primary Key
<code>slider_id</code>	INT(9)	Foreign Key
<code>slide_order</code>	INT(11)	Foreign Key
<code>options</code>	TEXT	Foreign Key
<code>layers</code>	TEXT	Foreign Key

Рисунок 3.15-Таблица слайдов

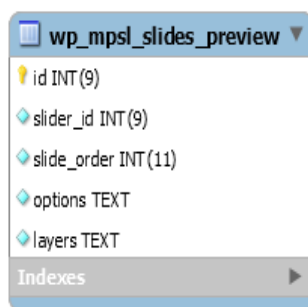


Рисунок 3.16-Таблица превью слайдов

Представленная модель базы данных CMS WordPress на MySQL обеспечивает хранение и обработку разнородных данных в некоем подобии объектно-ориентированного вида, где описания объектов данных хранятся отдельно от значений этих объектов.

Ниже приведена наглядная схема данных со связями (индексами):

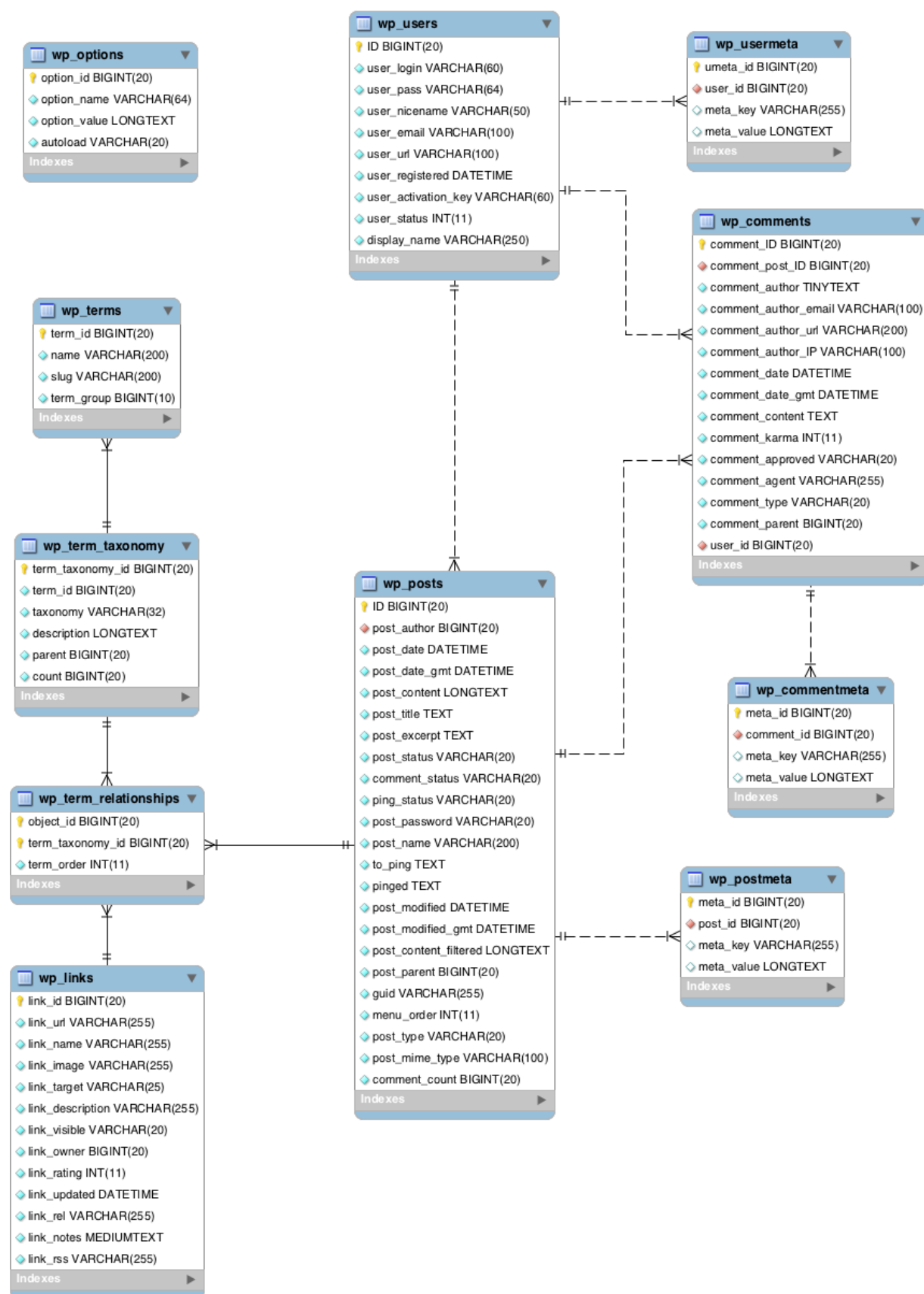


Рисунок 3.17- Схема данных со связями (индексами)

Большинство таблиц связаны с одной или несколькими другими таблицами с помощью одного ключа. Этот ключ является уникальным идентификатором для каждой записи (например `post_id`). Более подробно для каждой таблицы приведено в таблице 3.1

Таблица 3.1-Связь с другими таблицами

Таблица	Данные	Связи с другими таблицами
<code>wp_posts</code>	Записи, страницы, вложения, редакции, пользовательские записи	<code>wp_postmeta</code> через <code>post_id</code> <code>wp_term_relationships</code> через <code>post_id</code>
<code>wp_postmeta</code>	Метаданные записей, страниц и т.д.	<code>wp_posts</code> через <code>post_id</code>
<code>wp_comments</code>	Комментарии	<code>wp_posts</code> через <code>post_id</code>
<code>wp_commentmeta</code>	Метаданные комментариев	<code>wp_comments</code> через <code>comment_id</code>
<code>wp_term_relationships</code>	Связи между таксономиями и записями, страницами и т.д.	<code>wp_posts</code> через <code>post_id</code> <code>wp_term_taxonomy</code> через <code>term_taxonomy_id</code>
<code>wp_term_taxonomy</code>	Таксономии (включая категории и метки)	<code>wp_term_relationships</code> через <code>term_taxonomy_id</code>
<code>wp_terms</code>	Ваши категории, метки и термины пользовательских таксономий	<code>wp_term_taxonomy</code> через <code>term_id</code>
<code>wp_links</code>	Ссылки в вашем блоке (как правило, сейчас не используется)	<code>wp_term_relationships</code> через <code>link_id</code>
<code>wp_users</code>	Пользователи	<code>wp_posts</code> через <code>post_author</code>
<code>wp_user_meta</code>	Метаданные для каждого пользователя	<code>wp_users</code> через <code>user_id</code>
<code>wp_options</code>	Опции и настройки сайта (устанавливаются в админке на странице настроек и в темах/плагинах)	Отсутствуют

4 ОПИСАНИЕ РАЗРАБОТАННОГО ПРИЛОЖЕНИЯ

Сайт-визитка студии флористики и праздничного декора «Ксения» разработан для просмотра в интернете информации о студии любым пользователем без регистрации. Специально для удобства просмотра сайта на различных устройствах, интерфейс сайта трансформируется внешне, подстраиваясь под разрешение браузера пользователя, но всегда обеспечивает пользователей соответствующим функционалом.

Для администратора сайта предусмотрен специальный интерфейс CMS WordPress, в который можно попасть только посредством логина в администраторский раздел сайта (панель администратора) по адресу <http://kseniya-flower.ru/wp-admin/>, чтобы залогиниться, администратор должен ввести имя и пароль в соответствующих полях.

4.1 Описание интерфейса пользователя

Незарегистрированный пользователь сайта называется «гость». Обычно большинство общедоступных сайтов предоставляют гостевой доступ к основному объему информации сайта без регистрации. Именно потому, что на сайте-визитке студии флористики и праздничного декора «Ксения» не предусматривается функционал для зарегистрированных пользователей, нигде в пользовательском интерфейсе не предусмотрена форма логина или регистрации.

Основным навигационным элементом сайта является «главное меню». «Главное меню» состоит из текстовых ссылок на заданные страницы сайта, с целью облегчить пользователю доступ к основной информации сайта.

Логотип студии является дополнительной навигационной ссылкой, нажатие на него на внутренних страницах сайты приводит к возврату на главную страницу.

Поисковый интерфейс выполнен в виде дополнения к Главному меню и состоит из поля поискового запроса, и кнопки активации поиска в виде общепринятой иконки «лупы».

Все три вышеназванных элемента составляют шапку сайта и прикреплены к верху окна браузера.

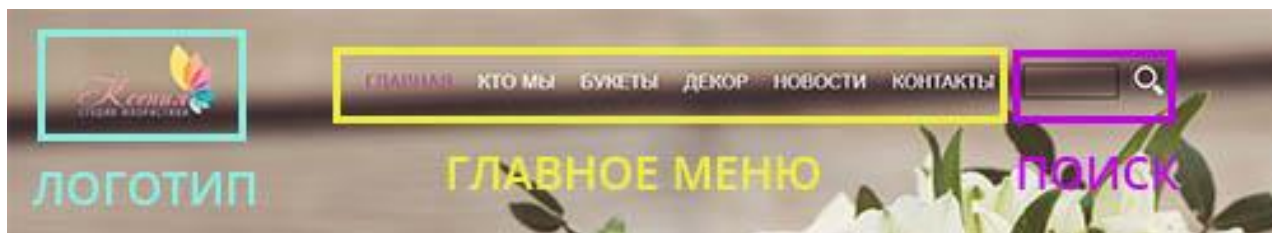


Рисунок 4.1-Шапка сайта

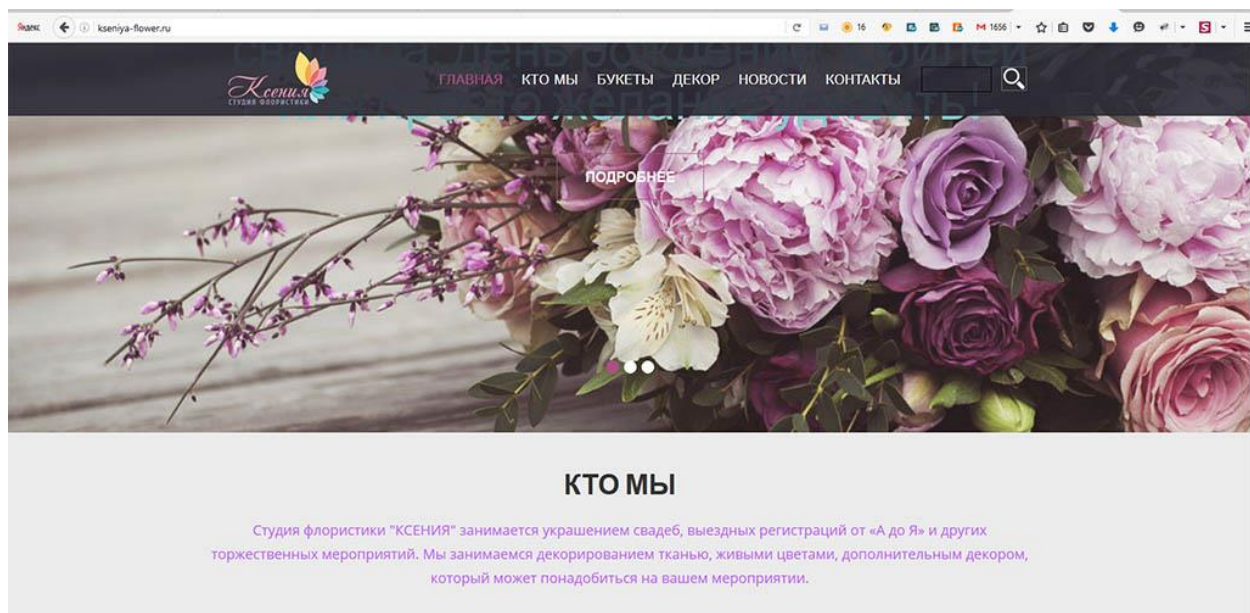


Рисунок 4.2-Прикрепленная шапка сайта

Так же на сайте предусмотрен слайдер рекламных баннеров, демонстрирующих преимущества студии и мотивирующих пользователя изучить глубже услуги студии «Ксения». С помощью сменяющихся баннеров (слайдов) можно ярко акцентировать внимание пользователя на определенной информации сайта, например, какие-либо акции или новинки. Переход на страницу сайта с информацией осуществляется при помощи ссылки с кнопки «Подробнее».

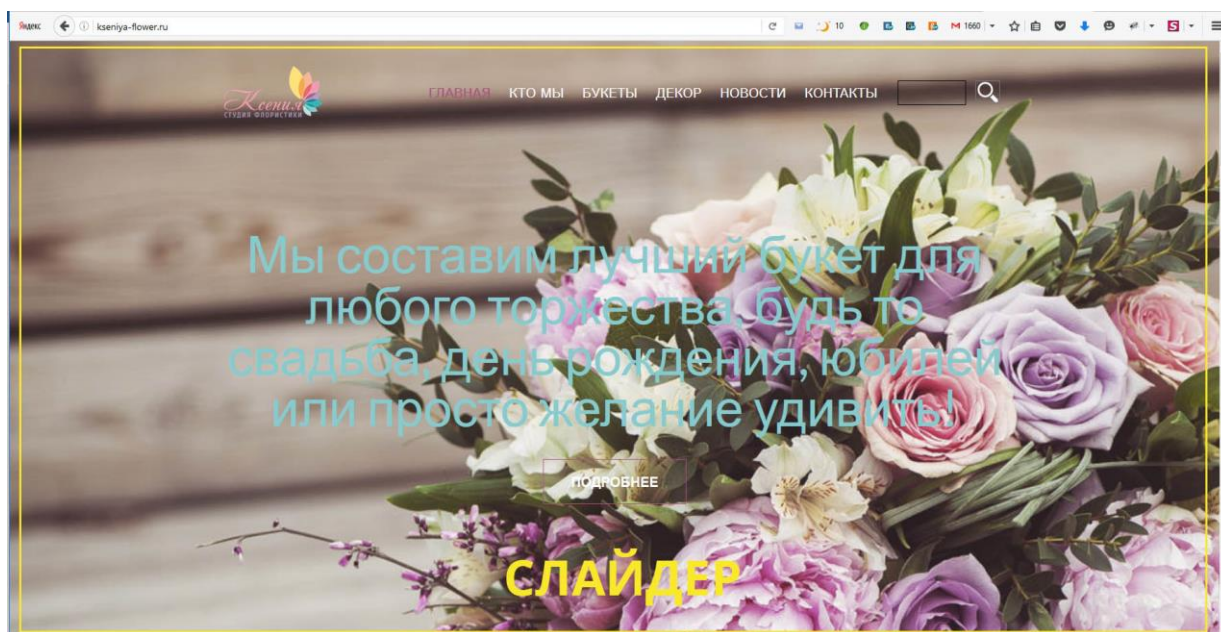


Рисунок 4.3 - Слайдер

Сразу же после слайдера идет информационный блок о студии, ее целях и преимуществах, задача блока продемонстрировать особенности студии «Ксения», ее подход к задачам и клиентам. В данном блоке информация

статическая, без переходов. Это основное послание пользователю от студии, декларация качественного подхода к пользователю и своему делу.



Рисунок 4.4-Информационный блок «О нас»

Следующий информационный блок показывает портфолио возможностей студии флористики и праздничного декора «Ксения». Пользователь на экране видит миниатюры изображений цветов, букетов и декора, с возможностью перехода на просмотр детальной информации о той или иной работе из портфолио студии. Переход осуществляется при клике на название работы. При клике на саму картинку (при наведении на картинку появляется иконка «лупа масштабирования»), можно рассмотреть картинку в большем масштабе.

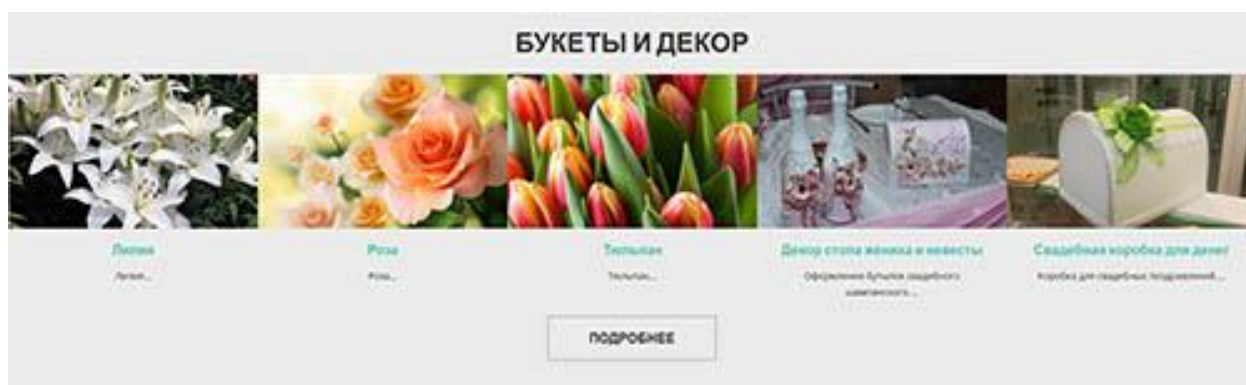


Рисунок 4.5-«Портфолио» на главной странице

В блоке присутствует кнопка «Подробнее», по нажатию на которую пользователь переходит на страницу со всеми работами студии. На странице

присутствует функционал отбора работ по заданной категории путем нажатия на соответствующую категорию, а так же возможность перехода на страницу детальной информации о работе, повторяющая функционал в блоке «Букеты и декор» с главной страницы (нажатие на название работы или товара).

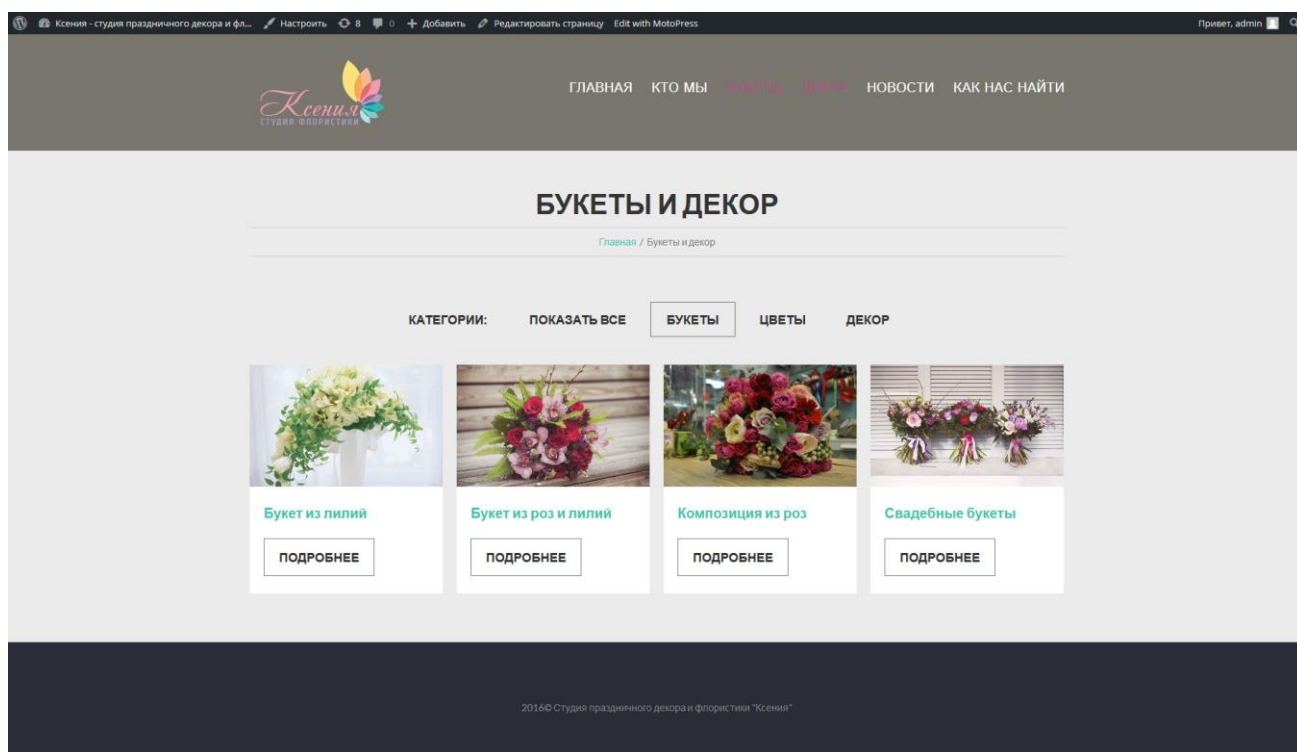


Рисунок 4.6-Портфолио студии с фильтром отбора по категории (на данном скриншоте отобраны работы в категории «Букеты»)

При переходе на страницу детальной информации пользователь видит детальную картинку о работе студии, а так же текстовое описание, галерею с любым количеством изображений, видео контент. При нажатии на изображение в галерее открывается увеличенное изображение в модальном окне. При нажатии на видео будет проигрываться видео-контент с youtube.com в текущем окошке. Проигрыватель видео-контента обладает функционалом пауза, перемотка вперед-назад, повторный запуск, переход на оригинальное видео на youtube.com, увеличение окна проигрывателя на весь экран. В дополнительном блоке с правой стороны выводится информация о доступных цветовых вариантах, при нажатии на название цвета будут показаны все работы из портфолио, имеющие варианты в таком же цвете.

В нижнем блоке содержится информация о схожих по тем или иным критериям работах из портфолио с возможностью перехода по названию работы в детальное описание. Там же находится форма для того, чтобы пользователь мог оставить комментарий о данной работе.

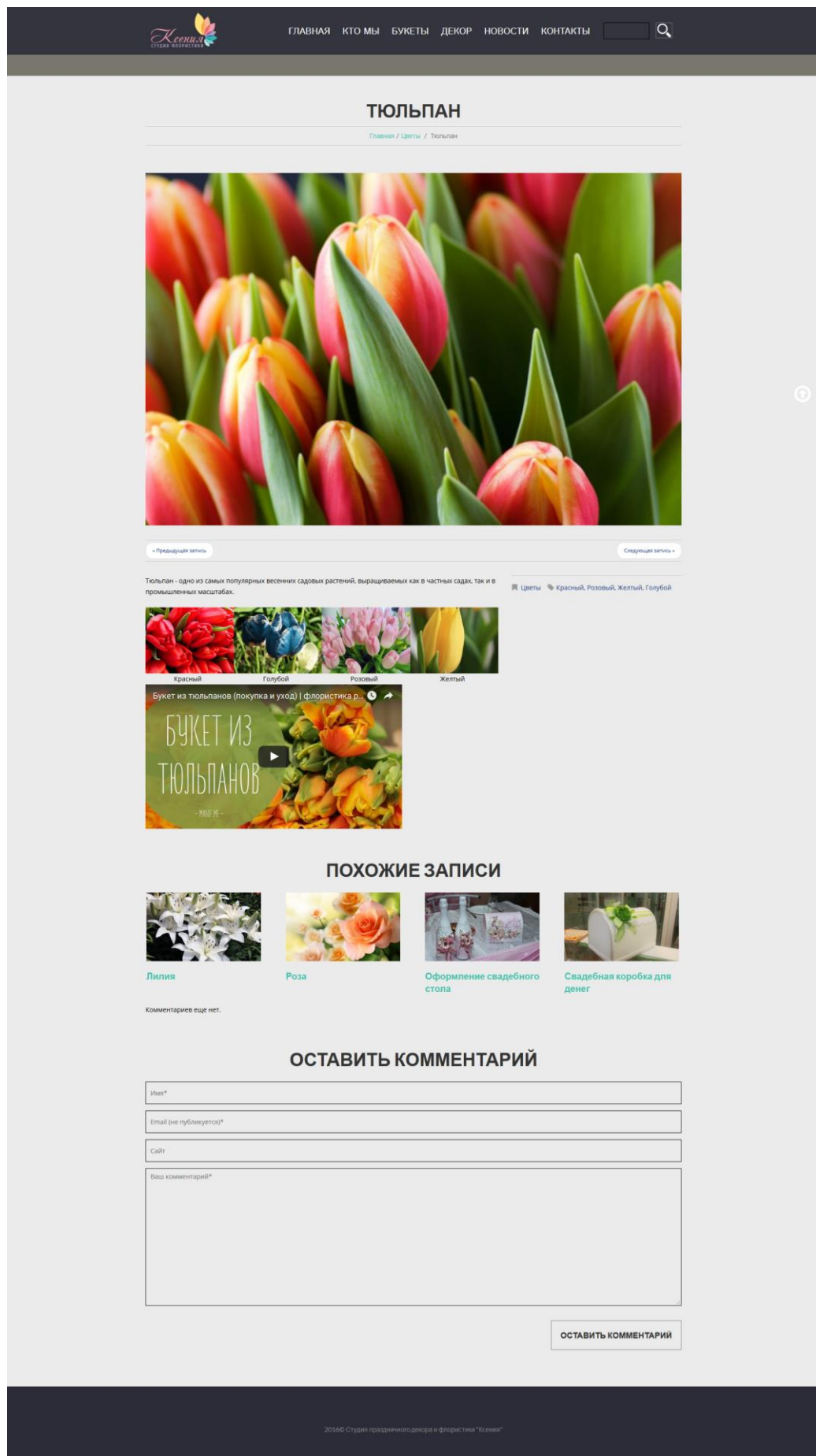


Рисунок 4.7-Пример детальной страницы о работе из «Портфолио»

Следующий за «Портфолио» («Букеты и декор») блок на главной странице называется «Наша команда» и представляет информацию о флористах и сотрудниках студии. Функционал данного блока такой же, как и в блоке «Портфолио» на главной, с той лишь разницей, переход на детальную страницу ведет на информацию о соответствующем сотруднике.



Рисунок 4.8-Блок «Наша команда» на главной странице сайта

Следующий блок на главной странице сайта называется «Форма обратной связи» и представляет из себя поля, в которых пользователь может заполнить Имя, адрес электронной почты, телефон и написать Сообщение менеджеру. Сообщения с данной формы, при нажатии на кнопку «Отправить» придет на электронную почту администратора. В сообщении будут содержаться поля формы, заполненные пользователем.



Рисунок 4.9-Форма обратной связи

Завершает главную страницу подвал, в котором содержится интерактивная карта местонахождения студии, с возможностью масштабировать и смещать видимую область карты. Следом за картой в подвале идет блок статической информации о правах на сайт. Данный блок статической информации о правах на сайт присутствует на всех страницах сайта.



Рисунок 4.10-Подвал с интерактивной картой на главной странице

При нажатии на пункт главного меню «Кто мы», пользователь переходит на страницу со статической информацией о студии «Ксения».

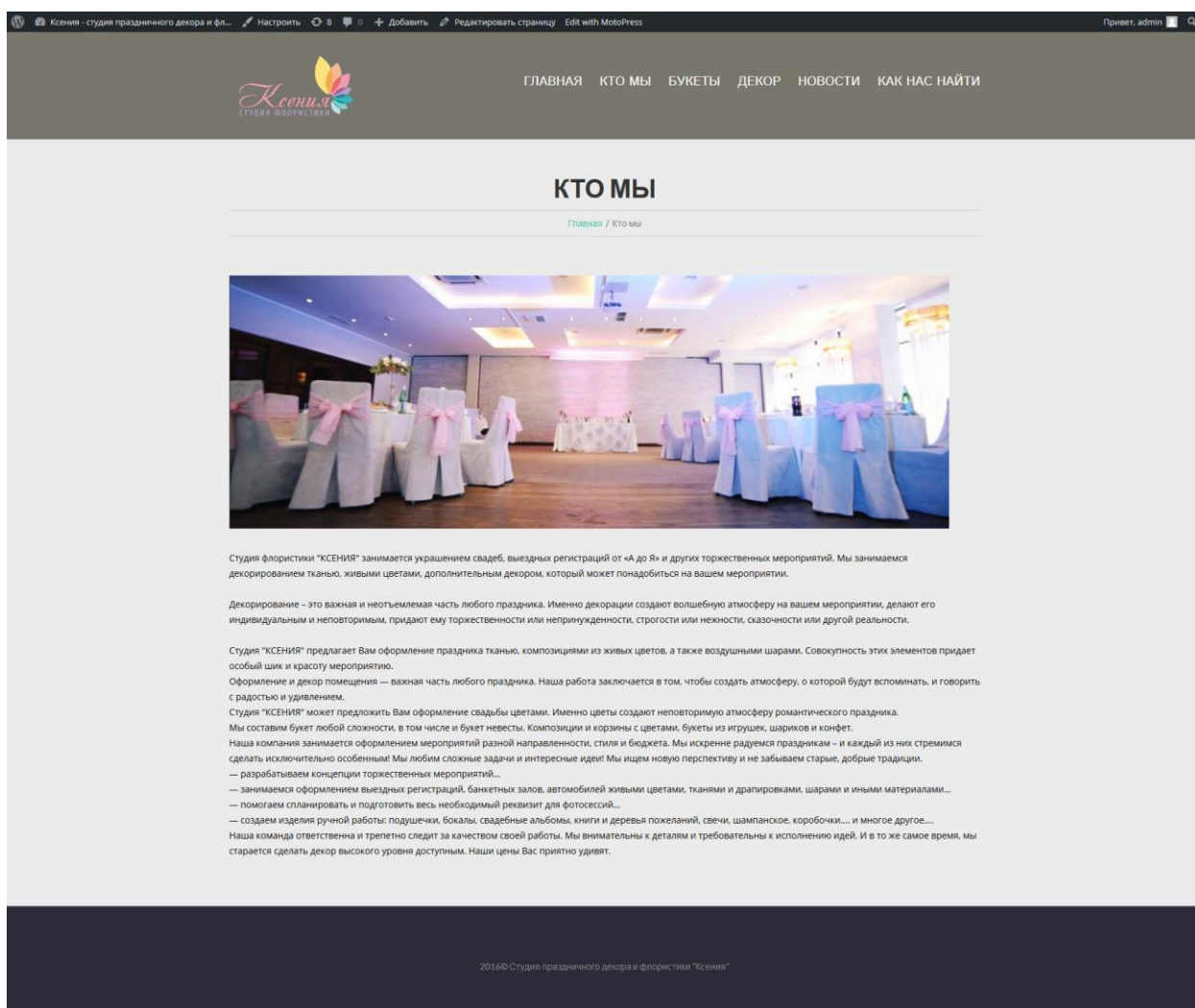


Рисунок 4.11-Страница «Кто мы»

При нажатии на пункт главного меню «Новости», пользователь переходит на страницу со списком новостей. В списке новостей показаны

заголовок новости, изображение анонса (если оно задано у новости), анонс и кнопка «Подробнее». Для просмотра новости детально, пользователю необходимо нажать на соответствующий заголовок, картинку анонса или кнопку «Подробнее». Новости в списке представлены в обратном хронологическом порядке (с начала самые свежие).

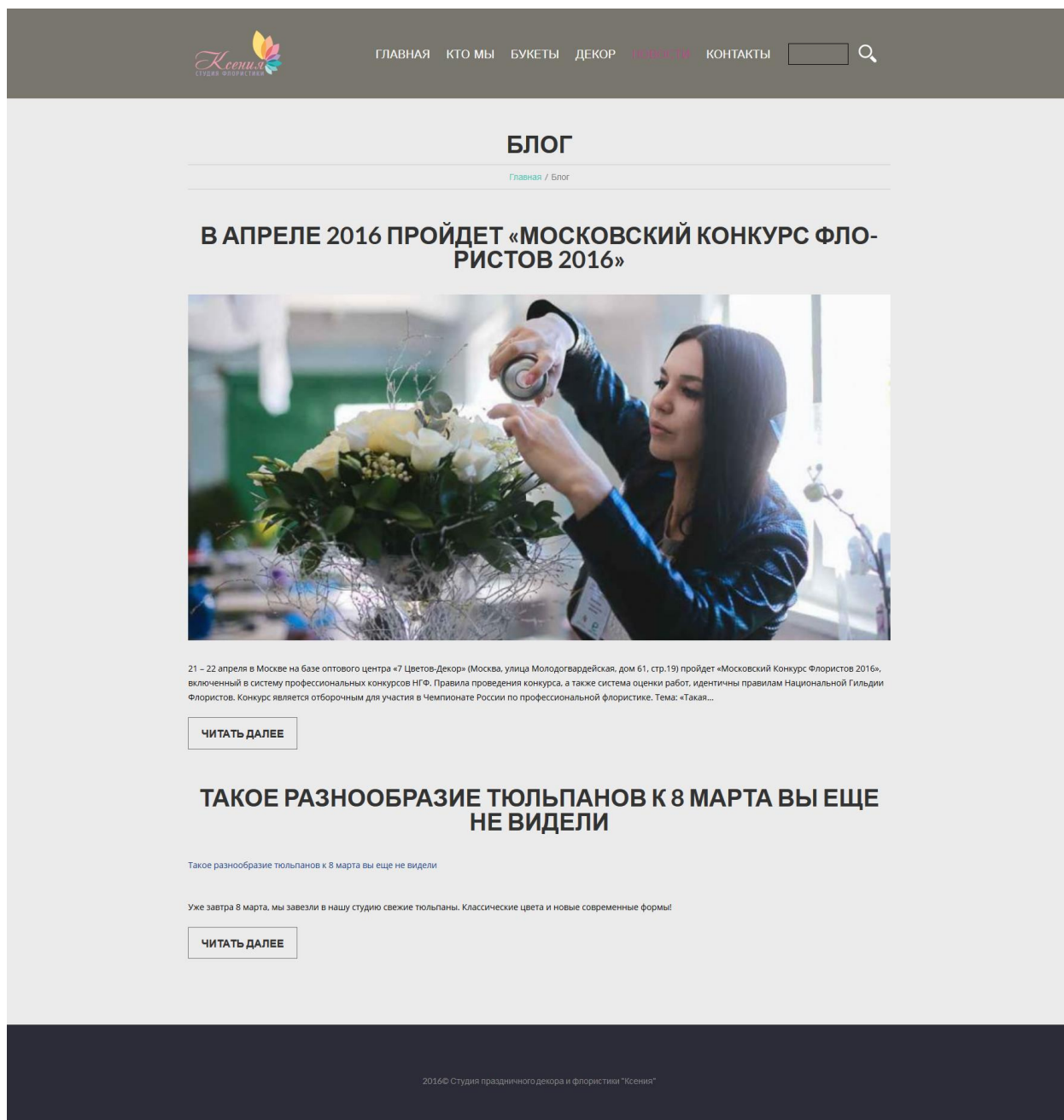


Рисунок 4.12-Страница со списком новостей

Функционал страницы детальной новости повторяет функционал детальной работы из «Портфолио», но дополнен навигацией с правой стороны по архивам новостей.

[Главный / Все рубрики / В апреле 2016 пройдет «Московский Конкурс Флористов 2016»](#)



Вся информация о Конкурсе размещена на сайте www.IAMFLORIST.com

ОСТАВИТЬ КОММЕНТАРИЙ

Главная

КТО МЫ

Букеты

Декор

Новости

Контакты

КОНТАКТЫ

Главная / Контакты

КартаСпутник

Знаменск
Знаменск, Гостиница

Сбербанк

Сбербанк

Метафон

Администрация ЗАТО Знаменский

Дополнительный офис "Знаменский"

Кит В Гоститель

Салон Красоты "Ника"

Лукойл

Воскресенская ул.

Перовская ул.

КОНТАКТНЫЕ ДАННЫЕ

«Ксения» - студия праздничного декора и флористики

416550 г.Знаменск, ул. Ленина, д.31

Телефон: +7 (937) 128 21 11
+7 (960) 856 31 50
E-mail: salon-kseniy@mail.ru

ОБРАТНАЯ СВЯЗЬ

Имя:

E-mail:

Телефон:

Сообщение:

ОЧИСТИТЬ

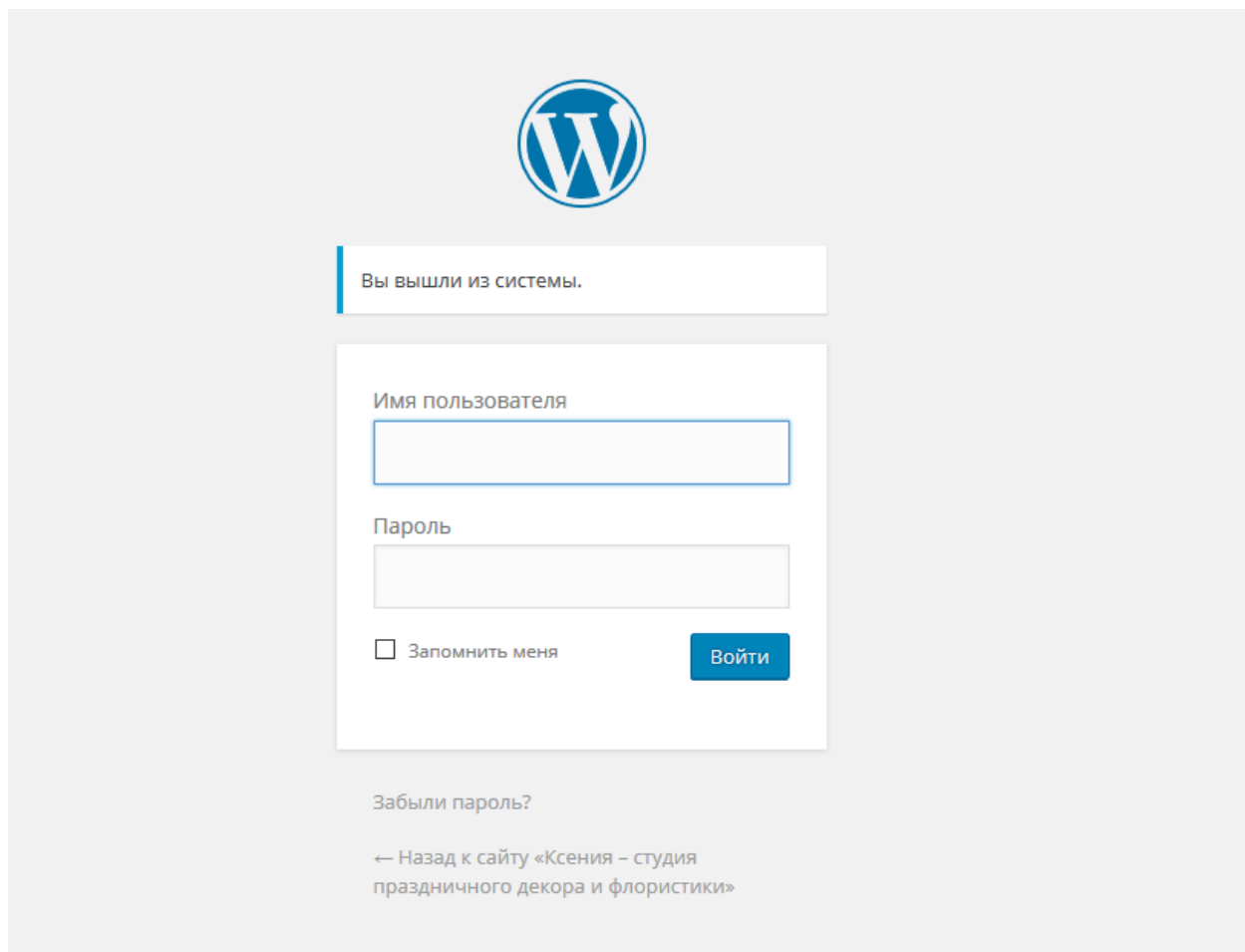
ОТПРАВИТЬ

© 2016 Студия праздничного декора и флористики «Ксения»

4.2 Описание интерфейса администратора

49

административной панелью, либо останется на текущей странице ожидая ввода верных данных администратора.



The image shows the WordPress administrator login page. At the top center is the WordPress logo. Below it is a white box with a blue border containing the message "Вы вышли из системы." (You have logged out of the system). Below this is another white box with a blue border containing the login form. The form has two input fields: "Имя пользователя" (Username) and "Пароль" (Password). Below the password field is a checkbox labeled "Запомнить меня" (Remember me). To the right of the checkbox is a blue button labeled "Войти" (Log in). Below the login form is a link "Забыли пароль?" (Forgot password?). At the bottom is a link "← Назад к сайту «Ксения – студия праздничного декора и флористики»" (← Back to the site «Ksenia – studio of holiday decoration and floristry»).

Рисунок 4.15-Страница авторизации администратора

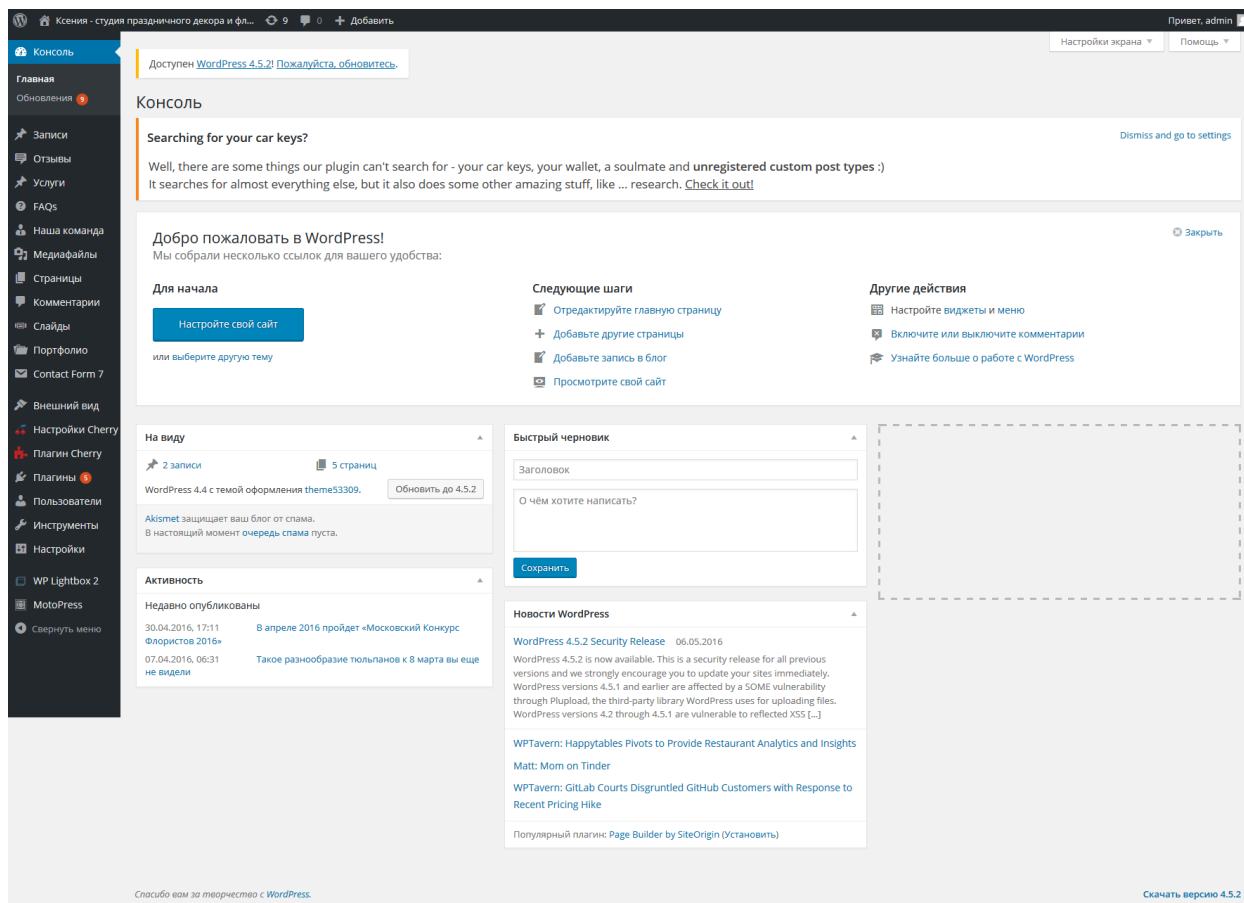


Рисунок 4.16-Административная панель сайта

Административная панель CMS Wordpress предоставляет пользователю широкие возможности по редактированию (добавлению, удалению) контента, управлению настройками сайта, метаданными для поисковых систем, управление плагинами, виджетами, темами, пользователями, созданием дополнительного функционала. Более подробную информацию о возможностях управления сайтом можно получить в документации по CMS Wordpress https://codex.wordpress.org/ru:Main_Page.

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы был разработан полнофункциональный web-сайт «Студия флористики», отвечающий всем современным нормам и полностью готовый к применению. Он сочетает в себе удобство, функциональность и яркий, запоминающийся внешний вид. Сайт содержит страницы с фотографиями, видеозаписями, соответствующими выбранной тематике. Данный сайт ориентирован на потенциальных покупателей - пользователей интернета, которых интересует работа флористов т.е. покупка букетов или цветов, украшение свадеб, выездные регистрации от «А до Я» и другие торжественные мероприятия, также декорирование тканей, живыми цветами, дополнительным декором, который может понадобиться на мероприятии.

С его помощью пользователи смогут получить представление о студии флористики «Ксения» и написать вопросы на электронный адрес или позвонить по указанным телефонам и уточнить интересующие вопросы.

Web-сайт размещен в глобальной сети, следовательно, география распространения возрастает до масштабов всего мира.

При разработке web-сайта были проанализированы современные web-технологии, позволяющие создавать интерактивные web-страницы. Наиболее подходящими для выполнения поставленной задачи оказались Macromedia Dreamweaver MX 2004 и CMS WordPress.

Разработанный сайт удовлетворяет всем требованиям, поставленным на этапе постановки задачи. Рассмотрены основные теоретические понятия разработки интернет-сайтов, рассмотрена структура разработанного сайта, приобретены навыки проектирования и разработки web-сайтов, разработан сайт на основе WordPress «Студия флористики».

При разработке web-сайта были использованы и доработаны готовые модули аутентификации, навигации (меню, слайдер, хлебные крошки), новостей, портфолио работ, команды и поиска по сайту. Данные модули были доработаны с учетом специфики web-сайта и успешно внедрены в его структуру.

В качестве дальнейшего совершенствования web-сайта представляется возможным дополнить сайт функционалом интернет-магазина, с возможностью приобрести и оплатить товар дистанционно. Так же возможна доработка интерфейса сайта с целью дальнейшего повышения его информативности, привлекательности и удобства.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Дроздев Н. Создай свой собственный WordPress сайт – 2011, PDF.- 39с.
2. <http://ru.wikipedia.org/>
3. Крис Койер, Джефф Стэрри, Погружение в WordPress - 2014, PDF.- 167с.
4. Мариков Ф.Ф. Самоучитель PHP5.-М.:Вильямс, 2013.-420с.
5. Прохоренок Н.А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера, 2013.-912с.
6. Эрик А. Мейер CSS-каскадные таблицы стилей. Подробное руководство - Москва: Символ, 2006.-122с.
7. Кожемякин А.А. HTML и CSS в примерах. Создание Web-страниц – М.: Альтекс-А, 2014.-416с.
8. Эрл Каслдайн, Крейг Шарки Изучаем JQUERY. Новый стиль программирования на JavaScript Санкт-Петербург: Питер, 2011-368с.
9. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript. –СПб.:Питер-2011.-496с.
10. <http://www.tadviser.ru/index.php/>
11. Кузнецов С.Д. Базы данных: учебник для студентов вузов, обучающихся по направлению подготовки "Прикладная математика и информатика" / С. Д. Кузнецов. - М.: Академия, 2012-496с.
12. Советов Б.Я. Информационные технологии: учебник для бакалавров, студентов вузов, обучающихся по направлениям подготовки дипломированных специалистов "Информатика и вычислительная техника" и "Информационные системы" / Б. Я. Советов, В. В. Цехановский; С.-Петерб. гос. электротехн. ун-т. - 6-е изд. - М.: Юрайт, 2012.-271с.
13. Грачев А. – Создаем свой сайт на WordPress - СПб.: Питер, 2011.- 288с.
14. Алексеев А.П. Введение в Web-дизайн: учебное пособие - М.:СОЛОН-ПРЕСС, 2008.-192с.
15. Материалы с сайта: <http://edu.evnts.pw/materials/131/>