Образовательное учреждение высшего образования

«Южно-Уральский институт управления и экономики»

|  |  |  |  |
| --- | --- | --- | --- |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |
| Разработка информационного сайта для проекта «Живая история» | |

НАУЧНАЯ РАБОТА

Руководитель работы

Т.Н. Лебедева

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Автор работы

А.А. Бакланова

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Ответственный консультант

по экономической части

Н.А. Пахомова

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Челябинск

2016

АННОТАЦИЯ

Научная работа выполнена на тему: «Разработка информационного сайта для проекта «Живая история». В работе раскрывается актуальность темы, сформулированы цели и задачи исследования.

Работа состоит из введения, трех глав и заключения.

Во введении определены актуальность работы, объект, предмет, цели и задачи, практическая значимость работы.

В первой главе представлен анализ предметной области, требования к разрабатываемой информационной системе, инструменты для решения поставленной задачи.

Во второй главе описана деятельность ИП Стрельников И.А.: представлены организационная структура и бизнес-процессы компании, рассмотрены способы и средства реализации информационных процессов на предприятии, а также сформулировано техническое задание на разработку сайта.

Третья глава содержит общее описание разработанной системы, оценку качества разрабатываемой системы в соответствии с ГОСТ, особенности внедрения, расчёт экономической эффективности проекта на разработку информационного сайта.

В заключении представлены общие выводы по научной работе.

СОДЕРЖАНИЕ

[Введение 9](#_Toc468091543)

[Глава 1 Анализ предметной области 12](#_Toc468091544)

[1.1 Анализ состояния исследуемого вопроса в конкретной области науки или технологии 12](#_Toc468091545)

[1.2 Формирование требований к информационной системе 14](#_Toc468091546)

[1.2.1 Цель проекта 14](#_Toc468091547)

[1.2.2 Результаты проекта 14](#_Toc468091548)

[1.2.3 Требования к информационному сайту 14](#_Toc468091549)

[1.2.4 Ключевые участники и заинтересованные стороны проекта 15](#_Toc468091550)

[1.3 Анализ способов разработки сайтов и обзор систем управления контентом сайта 16](#_Toc468091551)

[1.4 Выбор средств проектирования и средств создания web-приложения 21](#_Toc468091552)

[1.4.1 Microsoft Visio 21](#_Toc468091553)

[1.4.2 PHP 22](#_Toc468091554)

[1.4.3 MySQL 23](#_Toc468091555)

[1.4.4 PhpMyAdmin 24](#_Toc468091556)

[1.4.5 JavaScript 25](#_Toc468091557)

[1.4.6 Adobe Brackets 25](#_Toc468091558)

[Вывод по главе 1 26](#_Toc468091559)

[Глава 2 Проектирование информационной системы 27](#_Toc468091560)

[2.1 Описание деятельности предприятия 27](#_Toc468091561)

[2.2 Информация об организационной структуре ИП Стрельников И.А. 28](#_Toc468091562)

[2.3 Описание информационной системы предприятия 29](#_Toc468091563)

[2.4 Архитектура имеющихся технических средств и их основные параметры 32](#_Toc468091564)

[2.5 Используемое прикладное программное обеспечение в организации 34](#_Toc468091565)

[2.6 Построение модели деятельности предприятия 35](#_Toc468091566)

[2.6.1 Проектирование бизнес процессов с помощью IDEF0 37](#_Toc468091567)

[2.7 Формирование технического задания 40](#_Toc468091568)

[Вывод по главе 2 50](#_Toc468091569)

[Глава 3 Разработка и внедрение информационного сайта 51](#_Toc468091570)

[3.1 Создание информационного сайта 51](#_Toc468091571)

[3.1.1 Этапы разработки сайта 51](#_Toc468091572)

[3.1.2 Проектирование хранения данных 53](#_Toc468091573)

[3.1.3 Создание дизайн-‎макета веб-‎сайта 56](#_Toc468091574)

[3.1.4 Верстка 63](#_Toc468091575)

[3.2 Оценка качества сайта 65](#_Toc468091576)

[3.2.1 Методы тестирования и отладки сайта 65](#_Toc468091577)

[3.2.2 Сопоставление функций сайта с поставленными задачами 67](#_Toc468091578)

[3.3 Внедрение информационной системы 67](#_Toc468091579)

[3.3.1 Оптимизация 67](#_Toc468091580)

[3.3.2 Этапы продвижения сайта в сети Интернет 69](#_Toc468091581)

[3.4 Оценка экономических затрат на проект 71](#_Toc468091582)

[Вывод по главе 3 79](#_Toc468091583)

[Заключение 80](#_Toc468091584)

[Список литературы 82](#_Toc468091585)

[Приложение А 86](#_Toc468091586)

# ВВЕДЕНИЕ

Интернет растет бесконечно, каждый день подключаются тысячи новых пользователей. По статистике, уже каждый третий житель Земли имеет выход в интернет.

Но Интернет – это давно уже не только информационно-коммуникационная сеть. Это современное средство для заработка.

Одной из разновидностей сайтов являются сайт-визитка. Сайт-визитка – небольшой сайт, состоящий из одной (или нескольких) веб-страницы и содержащий основную информацию об организации, товарах или услугах.

Сайт нужен не только крупным торговым фирмам и предпринимателям, работающим в Интернете. Например, сайт-визитку необходимо создать региональной фирме, чтобы, представить себя в интернете и привлечь дополнительных клиентов, а также повысить имидж компании.

На сегодня средняя цена сайта-визитки находится в районе 5 000 руб., проекта «средней руки» – в диапазоне 15 000- 32 000 руб., цена на крупные интернет-магазины составляет 100 000 – 150 000 руб., а создание особо качественных и полнофункциональных порталов может стоить и дороже. Всё зависит от возможностей заказчика и его потребностей.

Сейчас компания, не находящаяся в Интернете, уже не может полноценно развиваться. Даже небольшому кафе необходимо присутствие в интернете – ведь многие потенциальные клиенты хотят узнать режим работы заведения, есть ли доставка, меню. Получения актуальной информации – вот что определяет появление в сети официальных сайтов и страничек даже самых маленьких предприятий.

Ушла в прошлое профессия курьера, который обходил предприятия и предлагал коммерческие предложения или узнавал потребности фирмы. Интернет всё это теперь делает. Так что если организации нет онлайн, ее возможности очень сокращены.

Актуальность создание «информационного сайта»:

* оперативность и масштабностью подачи информации широкому кругу пользователей;
* обратная онлайн связь с клиентами;
* возможностью маркетинговых исследований спроса и предложения;
* осуществление связи с представителями в других регионах.

Целью научной работы является разработка информационного сайта, способствующего информированию населения о мероприятиях проекта «Живая история», получение обратной связи, проведение маркетинговых исследований.

Задачи исследования:

* изучить научно-техническую литературу по проблеме исследования;
* изучить бизнес-процессы организации, выявить бизнес-процессы, требующие автоматизации;
* сформулировать техническое задание на разработку информационного сайта;
* разработать и наполнить программный продукт;
* сделать оценку экономических затрат на разработку программного продукта.

Актуальность создания информационного сайта для социального проекта «Живая история» заключается в том, чтобы довести сведения о предстоящих и прошедших мероприятиях исторической направленности до разнородного круга людей так, чтобы им было удобно.

Практическая значимость работы заключается в том, что разработан информационный проект, обеспечивающий своевременное и оперативное представление информации о проводимых мероприятиях. И, следовательно, улучшено качество и оперативность предоставляемой информации.

ГЛАВА 1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Анализ состояния исследуемого вопроса в конкретной области науки или технологии

Сайт-визитка – это небольшой сайт, как правило, состоящий из одной (или нескольких) веб-страницы и содержащий основную информацию об организации, частном лице, компании, товарах или услугах, прайс-листы, контактные данные.

Разновидности сайта визитки.

Одностраничный сайт **—** самый распространенный вид. Структура сайта самая простая – он представляют собой простой html документ. Нужен для того, чтобы рассказать о своей компании, выпускаемой продукции и предоставляемых услугах.

Такой сайт выполняет роль виртуального буклета.

Статический сайт визитка многостраничный

Сайт визитка без системы управления файлами CMS:

* сайт визитка многостраничный как статический сайт – это отдельные html странички, объединенные ссылками в одно целое;
* для визуального объединения шаблон главной страницы html распространяют на остальные страницы многостраничного сайта визитки, но на некоторых сайтах каждая html страница имеет свой собственный дизайн;
* статический сайт при малом числе страниц и отсутствии больших картинок или флэш анимации имеет хорошую скорость и быстро загружается на компьютер посетителя, потому что готовый код страницы html хранится в ідд сервера;
* однако cайт визитка без системы управления файлами не обладает интерактивностью. Читатели не могут оставить на нем свои комментарии. Даже владельцу сайта для внесения изменений на страницах требуется знать язык гипертекстовой разметки HTML.

Типичный статический сайт визитка – это набор неизменяемых информационных страниц в Интернете.

Динамический сайт визитка многостраничный.

Сайт визитка многостраничный с особой программой – системой управления сайтом – не имеет статические страницы html. В нем вообще нет готовых страниц, а есть лишь форма – шаблон с пустыми местами, куда при переходе на другую страницу вставляется новый текст и картинки. При большом числе страниц такое хранение материала в виде отдельных файлов экономит место на жестком диске сервера, что позволяет создавать многостраничные сайты визитки на недорогих тарифах.

Кроме того, динамический сайт визиткамногостраничныйобладает интерактивностью, т.е. посетители могут писать комментарии на его страницах, а администратор может вносить изменения на готовые страницы и писать новые во встроенном редакторе.

Сайт визитка на CMS на файлах.

В настоящее время еще существует класс программ – так называемые системы управления сайтом (CMS) на файлах. Положительной чертой таких CMS является организация некоторых зачатков интерактивности на сайтах.

Системы на файлах не «грузят» сервер, но результатом их работы являются всё те же готовые html страницы. По сути – такие системы – это лишь встроенный редактор для написания новых статей. В остальном такие CMS возвращают сайт на позиции сайтов без движков.

Сайт визитка на CMS с базами данных.

Самые функциональные сайты получаются на CMS с БД – базами данных. Такие системы (движки) имеют таблицы БД, в которых занесены данные о каждом файле сайта: где лежит, как показывать. Систематизация позволяет «на лету» собирать страницу сайта и быстро рисовать ее на экране мониторов у посетителя.

Продвинутые движки с базами данных позволяют создать не только многостраничный сайт визитку, но и многофункциональные сайты.

Наличие движка позволяет организовать на сайте взаимодействие с посетителем, который может заставлять сайт что-то выполнять по нажатию на кнопки. Посетитель может сам написать что-то на сайте (комментарий) или обратиться к администратору сайта в форме обратной связи. Для владельца сайта имеется специальная административная панель, что упрощает администрирование сайта. Администратор сайта способен производить редакцию записей на сайте и создавать новые во встроенном редакторе.

1.2 Формирование требований к информационной системе

1.2.1 Цель проекта

Цель создания информационного сайта – представить информацию о проводимых мероприятиях исторической направленности в г. Челябинске в сети и Интернет.

1.2.2 Результаты проекта

Результатом разработки станет сайт, который будет удовлетворять информационным потребностям пользователей:

* информационные потребности пользователей:

1. просмотр информации;
2. просмотр фото и видео;
3. просмотр анонсов предстоящих мероприятий;
4. просмотр отчета о прошедших мероприятиях;

* информационные потребности администраторов организаторов проекта:

1. авторизация в системе по логину и паролю (со своими правами доступа);
2. добавление и редактирование контента на сайте.

1.2.3 Требования к информационному сайту

Общими требованиями для сайта являются:

* навигация по сайту должна быть простой и понятной пользователю;
* предоставление полной и достоверной информации;
* дизайн сайта должен быть ярким и современным;
* кроссбраузерность – свойство сайта отображаться и работать во всех популярных браузерах идентично.

Требования к техническому обеспечению: web-сервер должен поддерживать PHP, MySQL, Apache. Разработка сайта должна ориентироваться на различную разрешающую способность и размеры монитора пользователя. На этапе подготовки и передачи web-сайта должна быть выполнена работа по подготовке программной документации в эксплуатацию в соответствии с требованиями ГОСТ 34.602-89.

На выполнение работ по разработке отводится 14 дней, которые включают в себя:

* проектирование;
* разработка дизайна и пользовательского интерфейса;
* создание проекта;
* тестирование и выполнение доработок;
* публикация;
* наполнение сайта.

1.2.4 Ключевые участники и заинтересованные стороны проекта

Ключевыми участниками проекта являются заказчик системы, пользователи системы и разработчик.

Заказчиком сайта выступает социальный проект «Живая история».

Пользователи системы:

* администратор сайта;
* потенциальные пользователи мероприятий.

Разработчик системы, который будет выполнять все этапы разработки системы: анализ требований, проектирование системы, разработку и тестирование кода, обучение сотрудников компании.

1.3 Анализ способов разработки сайтов и обзор систем управления контентом сайта

Сейчас, когда время бурного роста интернет-технологий, существуют различные способы разработки сайтов. От мгновенных запусков в несколько часов, до продолжительных которые длятся месяцами или даже годами. Рассмотрим наиболее популярные из них.

Самым простым способом, не требующим знаний языков программирования, это использование визуальных редакторов, где путем преставления блоков заполняется вся структура и получается простейшие странички сайта. Так же есть огромное множество сервисов по созданию блогов, в которых сайт собирается как конструктор. Самыми распространёнными из них являются uCoz, Wix, uKit и т.д.

Если есть знания языка гипертекстовой разметки HTML, а также CSS (каскадные таблицы стилей), то можно сделать простой статический сайт. С помощью HTML строится разметка содержимого будущего сайта, а через CSS настраивается его внешний вид.

Для создания более сложных динамических сайтов потребуется знания скриптового языка программирования PHP.

Чтобы сайт обладал большей функциональностью потребуются знания JavaScript и различных технологий, позволяющих сделать сайт более красивым, динамичным и быстродействующим. Такое создание с нуля займет много времени, как показывает практика, возникает множество ошибок.

Для устранения этих ошибок существуют такое программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта, называемое framework.

Это каркас программной системы (или подсистемы), который может включать вспомогательные программы, библиотеки кода, язык сценариев и другое ПО, облегчающее разработку и объединение разных компонентов большого программного проекта. Обычно объединение происходит за счёт использования единого API.

Framework определяется как множество конкретных и абстрактных классов, а также определений способов их взаимоотношения. Конкретные классы обычно реализуют взаимные отношения между классами. Абстрактные классы представляют собой точки расширения, в которых каркасы могут быть использованы или адаптированы.

Третьим более простым способом создания сайта является использование системы для обеспечения и организации совместного процесса создания, редактирования и управления содержимым страницы (CMS).

В основном CMS имеют модульную архитектуру, и управление контентом происходит благодаря отдельным модулям. В большинстве CMS встроено множество готовых моделей, а также есть уже готовый шаблонный сайт, который можно начать редактировать под себя.

В данный момент существует большое множество CMS, как платных, так и бесплатных.

CMS позволяют решать задачи различного уровня сложности в гораздо меньшие сроки, чем написание всего кода с нуля.

Был проведен анализ CMS. Были выбраны наиболее популярные из систем и произведен анализ по важным характеристикам для разрабатываемого сайта. Результат данного анализа представлен в таблице.

Таблица 1 – Сравнение характеристик различных CMS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Характеристика | WordPress | Joomla | Modx | 1С-Битрикс | [NetCat](http://www.ratingruneta.ru/netcat/) |
| Функционал | большой набор плагинов | огромное количество расширений, но некоторые из них платные. | Средний набор плагинов | хороший выбор готовых модулей, а так же сразу готовых решений с возможностью быстрой настройки (кроме редакции «Первого сайта») | небольшой набор стандартных модулей, |

Продолжение таблицы 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Особенности использования | | будет работать хорошо, если правильно настроить систему.  Огромный набор функционала излишне загружает систему  SEO ориентированность | Ограниченность SEO-инструментов, требующая установки дополнительных модулей | SEO ориентированность | повышенные требования к хостингу.  Система ориентирована на крупных заказчиков.  Возможность интеграции с 1С. | Сложная система редактирования шаблонов.  Слишком сжатая документация. |
| Цена за минимальную редакцию | Open source | | Open source | Open source | 4 900 руб. | 5 900 руб. | |
| Количество Сайтов на 1 системе | Один | | Один | Несколько | 1 шт, за каждый последующий необходимо доплачивать (1 470 руб) | Неограниченно | |
| Поддержка мультимедийного контента | да | | нет | да | да | да | |
| Надежность | низкая | | низкая | низкая | высокая | высокая | |

Исходя из сравнительного анализа систем и личного опыта использования некоторых из них, для реализации проекта была выбрана система управления контентом сайта WordPress.

WordPress– одна из самых популярных в мире информационных систем для обеспечения и организации процесса создания, редактирования и управления содержимым (CMS). Она предоставляет все необходимые функции для создания блога, новостного сайта. Эта современная платформа сориентирована на удобство использования, сетевые стандарты и, естественно, эстетическую привлекательность.

 WordPress бесплатен и свободен к распространению. Язык написания – РНР, в качестве базы данных используется My SQL, распространяется под универсальной общедоступной лицензией GNU.Сфера применения – от блогов до достаточно сложных новостных ресурсов и интернет-магазинов. Встроенная система «тем» и «плагинов» вместе с удачной архитектурой позволяет конструировать проекты широкой функциональной сложности.

Плагины – это расширения для WordPress, которые позволяют добавить в Web-сайт изначально отсутствующую функциональность. Темы оформления обеспечивают больший контроль над представлением материала, публикуемого на Web-сайте.

Под темой в WordPress понимаются файлы (или шаблоны), позволяющие реализовать изменение графического интерфейса и стиля отображения содержимого сайта. Тема меняет стиль отображения, используемый Web-сайтом, без внесения каких-либо изменений в программный код. Тема состоит из файлов шаблонов, изображений (\*.jpg, \*.gif), каскадных таблиц стилей (\*.css) и любых файлов с PHP-кодом (\*.php).

Сегодня Wordpressустановлен на каждом пятом сайте, при этом число новых ресурсов увеличивается примерно на 100 000 в сутки.

История создания системы Wordpress началась в июне 2001 года, когда некий Мишель Вальдриги начал разработку движка для блогов, дав ему кодовое название b2. Через некоторое время к Мишелю присоединились ещё двое его знакомых: Мэтт Мюлленвег и Майк Литл. В январе 2003 года Вальдриги прекратил разработку, поэтому автором WordPress считается Мэтт Мюлленвег. Права на товарную марку «WordPress» принадлежат некоммерческому фонду WordPress Foundation.

WordPress система довольно функциональна, что позволяет без особого труда реализовать сложные задачи довольно быстро, а при необходимости с легкостью можно написать свои компоненты (это функционал сайта, который управляет содержательной частью страницы), которые внедряются автоматически и без проблем.

У WordPress очень понятный и удобный интерфейс, очень гибкая система управления данными нужных типов. Система имеет свои особенности из-за различия структуры модуля и доступа к данным. Управление структурой сайта позволяет создавать любой уровень вложенности разделов.

При помощи макетов дизайна можно реализовать любой дизайн.

Возможности WordPress:

* информационная система позволяет создавать публикации при помощи сторонних программ;
* установка и настройка не представляют никакой сложности;
* поддерживает функции создания автоматически обрабатываемых списков обновлений RSS, Atom, trackback, pingback;
* уникальная простота системы взаимодействия с кодом подключаемых модулей (плагинов);
* реализация «тем», как наборов файлов – шаблонов (на РНР), весьма заметно увеличивает скорость и гибкость;
* существует возможность легко изменять внешний вид и способы вывода данных за счет поддержки «тем»;
* неисчерпаемые объемы библиотек «тем» и «плагинов»;
* доступна реализация сложных решений за счет заложенного потенциала архитектуры;
* предлагается использование человеко-понятных URL;
* присутствует система русского перевода.

Требования к программному обеспечению:

* РНР версии 5.2.4. или выше;
* MySQL версии 5.0 или выше;
* до версии 3.2 РНР версии 4.3 или выше.

Что не мало важно, система устанавливается практически на любой современный хостинг без установки дополнительного программного обеспечения.

И CMS WordРress поддерживает технологию gzip – сжатия, благодаря чему уменьшается нагрузка на сервер – его пропускная способность сохраняется достаточно высокой и способна справиться с большим трафиком.

Сайты, сделанные на WordPress CMS, индексируются лучше, чем сайта, основанные на других платформах, за счет организации данных и их отображения. Платформа WordPress продумана максимально лояльно по отношению к поисковым системам, поэтому проекты, реализованные на ней, будет несложно продвигать в выдачах поисковиков.

Wordpress, как и любой продукт, имеет поклонников и противников. Однако, эта CMS уже несколько лет остаётся наиболее популярной среди разработчиков всего мира.

1.4 Выбор средств проектирования и средств создания web-приложения

Для проектирования системы был выбран программный продукт  Microsoft Visio. В качестве CMS – система WordPress, которая работает на базе языков JavaScript, PHP и системы управления базами данных MySQL. Для разработки собственных компонентов использовались языки JavaScript, PHP, системы управления базами данных MySQL.

1.4.1 Microsoft Visio

Microsoft Office Visio 2007 – универсальное средство моделирования данных и приложений, поддерживающее и создание моделей данных, и объектно-ориентированное моделирование приложений.

Visio поддерживает обширный набор шаблонов, схемы сетей, блок-схем, бизнес-процессов, диаграмм рабочих процессов, модели баз данных.

Их можно использовать для визуализации и рационализации бизнес-процессов, отслеживания хода работы над проектами и использования ресурсов, оптимизации систем, составления схем организационных структур, карт сетей и планов зданий[2].

Утилита поставляется в двух вариантах:

* Standard – это решение для создания технических и деловых диаграмм, предназначенных для систематизации и наглядного представления различных данных, процессов и систем;
* Professional – содержит все возможности стандартного выпуска, а также расширенный набор шаблонов и функций, включая визуализацию данных и подключение к данным.

Визуализация информации в программе выполняется посредством использования разнообразных фигур, сгруппированных по категориям. При выборе пользователем шаблона конкретной тематики, боковая панель заполняется фигурами, вероятность применения которых в процессе работы самая высокая.

В моем проекте система Visio используется для анализа и проектирования бизнес-процессов, а также формирования на их основе технического задания к web-приложению.

1.4.2 PHP

WordPress – выбранная система управления контентом сайта, обоснование выбора представлено в пункте 1.3. В основе системы лежит язык программирования PHP, поэтому всю разработку необходимо вести на этом языке.

Скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов.

Благодаря встроенным средствам для разработки веб-приложений, язык имеет большую популярность. Основными из них являются:

* автоматическое извлечение POST и GET-параметров, и переменных окружения веб-сервера в предопределённые массивы;
* взаимодействие с большим количеством различных систем управления базами данных;
* автоматизированная отправка HTTP-заголовков;
* работа с HTTP-авторизацией;
* работа с cookies и сессиями;
* работа с локальными и удалёнными файлами, сокетами;
* обработка файлов, загружаемых на сервер;
* работа с Xforms.

1.4.3 MySQL

Для WordPress нужна база данных MySQL.

MySQL – свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle, получившая права на торговую марку вместе с поглощённой Sun Microsystems, которая ранее приобрела шведскую компанию MySQL AB. Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации[2].

MySQL отличатся хорошей скоростью работы, надежностью, гибкостью. Является системой клиент-сервер, которая содержит многопоточный SQL-сервер, обеспечивающий поддержку различных вычислительных машин баз данных, а также несколько различных клиентских программ и библиотек, средства администрирования и широкий спектр программных интерфейсов (API).

MySQL система, основанная на привилегиях и паролях, за счет чего обеспечивается гибкость и безопасность, и с возможностью верификации с удаленного компьютера. Пароли защищены, т.к. они при передаче по сети при соединении с сервером шифруются.

1.4.4 PhpMyAdmin

Для удобства работы с базами данных MySQL через браузер будет использоваться phpMyAdmin.

PhpMyAdmin – веб-приложение с открытым кодом, написанное на языке PHP и представляющее собой веб-интерфейс для администрирования системы управления базами данных MySQL. Данное приложение позволяет через браузер осуществлять администрирование сервера MySQL, запускать команды SQL и просматривать содержимое таблиц и баз данных. Приложение пользуется большой популярностью у веб-разработчиков, так как позволяет управлять системой управления базами данных MySQL без непосредственного ввода SQL команд, предоставляя дружественный интерфейс.

На сегодняшний день phpMyAdmin широко применяется на практике. Последнее связано с тем, что разработчики интенсивно развивают свой продукт, учитывая все нововведения системы управления базами данных MySQL. Подавляющее большинство российских провайдеров используют это приложение в качестве панели управления для того, чтобы предоставить своим клиентам возможность администрирования выделенных им баз данных.

Приложение распространяется под лицензией GNU General Public License и поэтому многие другие разработчики интегрируют его в свои разработки, например, XAMPP, Denwer, AppServ, Open Server.

1.4.5 JavaScript

JavaScript – прототипно-ориентированный сценарный язык программирования. Является диалектом языка ECMAScript.

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам.

JavaScript является объектно-ориентированным языком, но используемое в языке прототипирование обуславливает отличия в работе с объектами по сравнению с традиционными класс-ориентированными языками. Кроме того, JavaScript имеет ряд свойств, присущих функциональным языкам – функции как объекты первого класса, объекты как списки, анонимные функции, замыкания – что придаёт языку дополнительную гибкость.

1.4.6 Adobe Brackets

Для написания кода использовалась программа Adobe Brackets.

Adobe Brackets- Бесплатный редактор с открытым кодом для веб-разработчиков. Brackets ориентирован на работу с HTML, CSS и JavaScript. Эти же технологии лежат в основе самого редактора, что обеспечивает его кроссплатформенность т.е. совместимость с операционными системами Mac, Windows и Linux. Brackets создан и развивается Adobe Systems под лицензией MIT License и поддерживается на GitHub.

ВЫВОД ПО ГЛАВЕ 1

В первой главе был проведен анализ предметной области. В том числе был проведен анализ актуальности создания информационного сайта как нового способа представления компании в интернет пространстве.

Также были приведены основные требования к оформлению и функционалу сайта. Дизайн не должен быть навязчивым, не должен отвлекать от контента. Навигация по сайту должна быть максимально простой и понятной для обычного пользователя, далекого от программирования.

В главе представлен также обзор инструментальных средств для разработки web-приложений. На основе личного опыта и обзора в качестве основного инструментального средства разработки системы была выбрана система WordPress.

Исходя из выбранной CMS были выбраны средства проектирования и разработки:

* языки разметки HTML и CSS;
* основной язык программирования PHP и как дополнительный JavaScript;
* система управления базами данных MySQL и приложение для управление СУБД PhpMyAdmin;
* редактор кода Adobe Brackets.

ГЛАВА 2 ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ

2.1 Описание деятельности предприятия

ИП Стрельников И.А. работает с 29.10.2014 и является индивидуальным предпринимателем. Основной вид деятельности – прочая деятельность, связанная с использованием вычислительной техники и информационных технологий, осуществляется в г. Челябинске.

Коммерческое название компании «i1-marketing», агентство интернет-маркетинга.

Компания специализируется на разработке, поддержке web-сайтов, продвижение сайтов в поисковых системах, контекстной рекламе и продвижение в соц.сетях, автоматизация бизнес-процессов.

В настоящий момент в компании работает 8 сотрудников, офис разработки находится в городе Челябинск по адресу Чайковского, 173. Организационная структура компании представлена на рисунке 1.

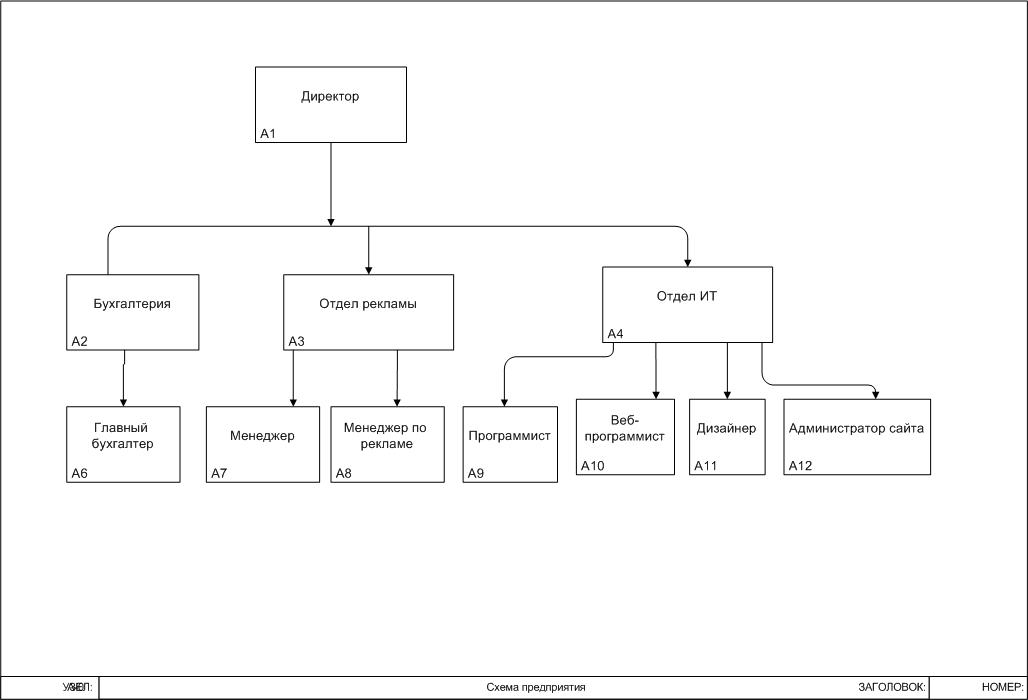


Рисунок 1 – Организационная структура компании ИП Стрельников И.А.

2.2 Информация об организационной структуре ИП Стрельников И.А.

Директор – владелец, обеспечивает управление ИП.

Отдел IT-технологий – занимается разработкой, внедрением и сопровождением новых программных продуктов, а также осуществляет техническую поддержку уже введенных в эксплуатацию.

В техническом отделе работают 2 программиста, дизайнер и администратор сайта.

Обязанности администратора сайта:

* разрабатывает и поддерживает основную концепцию сайта, вносит предложения по доработке концепции и содержания сайта, по введению нового сервиса для посетителей сайта;
* следит за текстовым наполнением сайта, постоянным обновлением информации;
* просматривает все опубликованные на сайте документы и материалы, удаляет и редактирует материалы, не соответствующие общей концепции и правилам пользования сайтом;
* осуществляет контроль за соблюдением посетителями правил пользования сайтом, а также иных обязательных требований, выдвинутых его создателями;
* оказывает помощь авторам в оформлении публикуемых материалов;
* выполняет работы по редактуре и корректуре материалов и их структурированию, включая дизайн оформления текстов, таблиц и т.п.;
* участвует в художественном оформлении помещаемой на сайт информации совместно с веб-мастером;
* исследует потребности и запросы посетителей сайта;
* отслеживает работу конкурентов, то есть сайтов со схожей тематикой, концепцией и содержанием;
* анализирует информацию, полученную от посетителей сайта, и с учетом технических возможностей и на основе полученной информации разрабатывает и внедряет новые сервисы, повышающие эффективность использования сайта потребителями.

Отдел рекламы – осуществляет продвижение web-сайтов в интернете. Контекстная реклама, социальные сети и т.д. В отдел по рекламе входит 2 менеджера.

Бухгалтерия – ведение бухгалтерского учета в соответствии с требованиями действующего законодательства Российской Федерации. Осуществление контроля за соответствием заключенных договоров.

2.3 Описание информационной системы предприятия

Локальная вычислительная сеть ИП Стрельников построена по топологии «звезда» и включает в себя 8 рабочих станций, сетевого принтера, сервера баз данных, почтового, файлового и веб-сервера.

Коммутация устройств в сети происходит через коммутатор, производства компании D’Link, UTP кабеля пятой категории и одной точки доступа, поддерживающая стандарт IEEE 802.11a.

Связь с глобальной сетью Интернет осуществляется через маршрутизатор.

Звезда – это топология локальной сети, где каждая рабочая станция присоединена к центральному устройству (‎коммутатору или маршрутизатору‎). Центральное устройство управляет движением пакетов в сети. Каждый компьютер через сетевую карту подключается к коммутатору отдельным кабелем.

Схема локальной вычислительной сети предприятия представлена на рисунке 2.

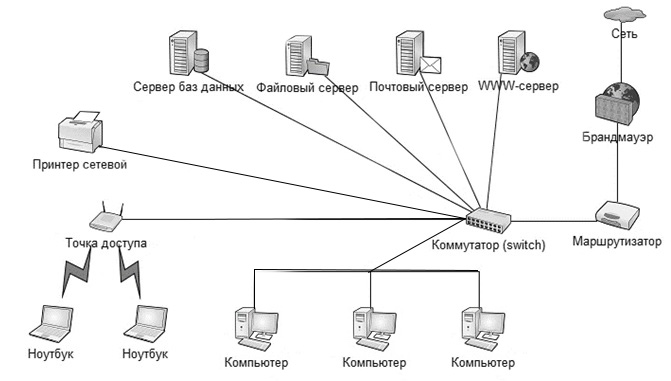


Рисунок 2 – Схема локальной вычислительной сети ИП Стрельников И.А.

В ИП Стрельников все сотрудники используют в качестве основной операционной системы Microsoft Windows.

Microsoft Windows — семейство операционных систем корпорации Microsoft, ориентированных на применение графического интерфейса при управлении. Изначально Windows была всего лишь графической надстройкой для MS-DOS.

На сегодняшний момент операционная система Windows фирмы Microsoft во всех ее проявлениях, считается самой распространенной операционной системой на ПК: в мире более 150 млн. IBM PC-совместимых компьютеров, и система Windows установлена на 100 млн. из них.

Операционная система — это программа, которая загружается при включении компьютера. Она производит диалог с пользователем, осуществляет управление компьютером, его ресурсами (оперативной памятью, местом на дисках и т.д.), запускает другие (прикладные) программы на выполнение. Операционная система обеспечивает пользователю и прикладным программам удобный способ общения (интерфейс) с устройствами компьютера.

Первая версия Windows вышла в свет в конце 80-х годов и осталась совершенно незамеченной. Такая же участь постигла и следующую версию, и только версия Windows 3.0 (1992) сумела пробить себе дорогу к сердцам пользователей и стать «продуктом года». А еще через два года на свет появились версии 3.1 и 3.11 (последняя включала полную поддержку мультимедиа и работу в локальной сети — потому и носила уточняющее название Windows For Workgroups), окончательно утвердившие господство Windows на Олимпе операционных систем.

В 2000 году корпорация Microsoft выпустила слегка измененную версию системы Windows 98, названную Windows Me (выпуск тысячелетия).

Операционная система Microsoft Windows XP, известная также под кодовым наименованием Microsoft Codename Whistler, является новой ОС семейства Windows, созданной на базе технологии NT. Первоначально в планы корпорации Microsoft входила разработка двух независимых операционных систем нового поколения. Первый проект получил рабочее название Neptune, эта ОС должна была стать очередным обновлением Windows Millennium Edition, новой системой линейки Windows 9X. Второй проект, называвшийся Odyssey, предполагал создание ОС на платформе Windows NT, которая должна была придти на смену Windows 2000. Однако руководство Microsoft посчитало нецелесообразным рассредоточивать ресурсы на продвижение двух разных ОС, вследствие чего оба направления разработок были объединены в один проект — Microsoft Whistler.[3]

Windows Vista — операционная система семейства Windows NT корпорации Microsoft. Была выпущена 30 января 2007 года. На стадии разработки имела кодовое имя «Longhorn». Построена на основе гибридного ядра NT версии 6.0.

Windows 7 — пользовательская операционная система семейства Windows NT, следует по времени выхода за Windows Vista и предшественник Windows 8.

Windows 8 — операционная система, принадлежащая к семейству ОС Microsoft Windows, в линейке следующая за Windows 7. Номер версии в линейке NT — 6.2. Поступила в продажу 26 октября 2012 года.

Windows 10 — операционная система для персональных компьютеров, разработанная корпорацией Microsoft в рамках семейства Windows NT. После Windows 8 система получила номер 10, минуя 9.

Система призвана стать единой для разных устройств, таких как персональные компьютеры, планшеты, смартфоны.

Доступна единая платформа разработки и единый магазин универсальных приложений, совместимых со всеми поддерживаемыми устройствами. Windows 10 поставляется в качестве услуги с выпуском обновлений на протяжении всего цикла поддержки.

В течение первого года после выхода системы пользователи могут бесплатно обновиться до Windows 10 на устройствах под управлением легальных Windows 7, Windows 8.1 и Windows Phone 8.1.

Среди значимых нововведений — голосовая помощница Кортана, возможность создания и переключения нескольких рабочих столов и др.

Windows 10 — последняя «коробочная» версия Windows, все последующие версии будут распространяться исключительно в цифровом виде.

Лицензионное соглашение Windows 10 позволяет компании Microsoft собирать многочисленные сведения о пользователе, историю его интернет-деятельности, пароли к точкам доступа, данные, набираемые на клавиатуре и многое другое.

2.4 Архитектура имеющихся технических средств и их основные параметры

Основными характеристиками персонального компьютера (ПК) являются:

* производительность (быстродействие) ПК – возможность компьютера обрабатывать большие объёмы информации. Определяется быстродействием процессора, объёмом оперативной памяти (ОП) и скоростью доступа к ней;
* тактовая частота процессора (частота синхронизации) – число тактов процессора в секунду, а такт – промежуток времени (микросекунды) за который выполняется элементарная операция (например, вычитание);
* производительность (быстродействие) процессора – количество элементарных операций выполняемых за 1 секунду;
* разрядность процессора – максимальная длина (количество разрядов) двоичного кода, который может обрабатываться и передаваться процессором целиком. Разрядность связана с размером специальных ячеек памяти – регистрами. Регистр в 1 байт (8 бит) называют восьмиразрядным, в 2байта – 16-разрядным и т.д. Высокопроизводительные компьютеры имеют 8-байтовые регистры (64 разряда);
* время доступа – быстродействие модулей оперативной памяти (ОП), это период времени, необходимый для считывание минимальной порции информации из ячеек памяти или записи в память;
* плотность записи – объем информации, записанной на единице длины дорожки (бит/мм);
* скорость обмена информации – скорость записи/считывания на носитель, которая определяется скоростью вращения и перемещения этого носителя в устройстве;
* объем памяти (ёмкость) – максимальный объем информации, который может храниться в ней.

В офисе компании «ИП Стрельников И.А.» 7 персональных компьютеров:

* процессоры Intel Core i5, тактовая частота от 3100 до 3800МГц;
* ОЗУ от 2048 до 16384Мб;
* жесткие диски от 500 Gb до 5 Tb;
* видео карты от 1024 мб до 2048 мб;
* блоки питания от 450 до 1000 w;
* материнские платы;
* оптические DVD-приводы;
* мониторы AOC диагональ от 17 до 23 дюймов;
* оптические мыши ZALMAN;
* клавиатуры GIGABYTE.

2.5 Используемое прикладное программное обеспечение в организации

Прикладная программа или приложение — предназначенная для выполнения определенных задач и рассчитанная на непосредственное взаимодействие с пользователем. В большинстве операционных систем прикладные программы не могут обращаться к ресурсам компьютера напрямую, а взаимодействуют с оборудованием и прочим посредством операционной системы. Также на простом языке – вспомогательные программы.

В компании ИП Стрельников И.А. используется следующее программное обеспечение:

1. Ccleaner (ранее — Crap Cleaner) — это бесплатная утилита с закрытым исходным кодом, которая предоставляет пользователям мощный и простой в использовании инструмент для очистки и оптимизации 32-битных и 64-разрядных операционных систем Microsoft Windows;
2. Microsoft Office — офисный пакет приложений, созданных корпорацией Microsoft для операционных систем Microsoft Windows, Apple Mac OS X и Apple iOS (на iPad). В состав этого пакета входит программное обеспечение для работы с различными типами документов: текстами, электронными таблицами, базами данных и др.;
3. 7-Zip — свободный файловый архиватор с высокой степенью сжатия данных. Поддерживает несколько алгоритмов сжатия и множество форматов данных, включая собственный формат 7z c высокоэффективным алгоритмом сжатия LZMA;
4. Adobe Acrobat — пакет программ, выпускаемый компанией Adobe Systems для создания и просмотра электронных публикаций в формате PDF;
5. ICQ-централизованная служба мгновенного обмена сообщениями сети Интернет, в настоящее время принадлежащая инвестиционному фонду Mail.ru Group. Служба является коммерческой, но её использование бесплатно;
6. The Bat! — программа для работы с электронной почтой для ОС Windows. Разрабатывается молдавской компанией Ritlabs. Программа The Bat! Популярна среди российских пользователей.Главной задачей программы является защита переписки от мониторинга третьими лицами;
7. Opera — веб-браузер и пакет прикладных программ для работы в Интернете, выпускаемый компанией Opera Software. Разработан в 1994 году группой исследователей из норвежской компании Telenor;
8. CorelDRAW — редактор, разработанный канадской корпорацией [Corel](https://ru.wikipedia.org/wiki/Corel" \o "Corel). Главным предназначением которой являются создание и обработка выполненных в формате векторной графики документов;
9. Аdobe photoshop cs6- Графический редактор. Многофункциональный графический редактор, разработанный и распространяемый фирмой Adobe Systems. В основном работает с растровыми изображениями, однако имеет некоторые векторные инструменты.

2.6 Построение модели деятельности предприятия

Модель деятельности организации – совокупность взаимосвязанных и взаимодополняющих графических моделей различных типов, каждая из которых описывает существующую ситуацию в конкретной предметной области деятельности.

Моделирование деятельности организации – документирование, анализ и оптимизация работы предприятия или отдельных направлений его деятельности, его целей и задач, механизмов и ресурсов, используемых для их достижения, правовых ограничений и взаимоотношений со средой, в которой предприятие ведет свою деятельность.

Бизнес-процесс – это комплекс взаимосвязанных задач и мероприятий, нацеленных на создание конкретного продукта или услуги для конечных потребителей. Чаще всего используется графическое описание деятельности организации в виде блок-схем бизнес-процессов.

Одним из многих методов анализа текущих бизнес-процессов является составление моделей «как есть» и «должно быть» после критического анализа первой. Затем по результатам проделанной работы составляется план мероприятий по внедрению изменений, реализовать которые сочли необходимыми при обработке результатов.

Для проектирования программных продуктов широко используется UML – унифицированный язык моделирования. Модели UML помогают понять, обсудить и спроектировать системы программного обеспечения.

Схемы моделирования UML и схемы слоев могут существовать только внутри проекта моделирования. Каждый проект моделирования содержит совместную модель UML и несколько схем UML.

Модель бизнес-прецедентов описывает бизнес-процессы с точки зрения внешнего пользователя, т.е. отражает взгляд на деятельность организации извне.

Проектирование системы начинается с изучения и моделирования бизнес-деятельности организации. На этом этапе вводится и отображается в модели ряд понятий, свойственных объектно-ориентированному подходу:

Исполнитель (действующее лицо, actor) – личность, организация или система, взаимодействующая с ИС. Различают внешнего исполнителя (который использует или используется системой, т.е. порождает прецеденты деятельности) и внутреннего исполнителя (который обеспечивает реализацию прецедентов деятельности внутри системы). На диаграмме исполнитель представляется стилизованной фигуркой человека.

Прецедент — законченная последовательность действий, инициированная внешним объектом (личностью или системой), которая взаимодействует с ИС и получает в результате некоторое сообщение от ИС. На диаграмме представляется овалом с надписью, отражающей содержание действия.

Класс — описание совокупности однородных объектов с их атрибутами, операциями, отношениями и семантикой. На диаграмме представляется прямоугольником, содержащим описания атрибутов и операций класса.

Ассоциация — связь между двумя элементами модели. На диаграмме представляется линией.

Обобщение — связь между двумя элементами модели, когда один элемент (подкласс) является частным случаем другого элемента (суперкласса). На диаграмме представляется стрелкой.

Агрегация — отношение между элементами модели, когда один элемент является частью другого элемента (агрегата). На диаграмме представляется стрелкой с ромбовидным концом.

Проектирование любой ИС лучше всего начинать с построения диаграммы прецедентов, описывающей внешнюю границу ИС. Такая диаграмма называется главной диаграммой прецедентов.

Для включения в диаграмму, выбранные прецеденты должны удовлетворять следующим критериям:

* прецедент должен описывать, что нужно делать, а не как;
* прецедент должен описывать действия с точки зрения исполнителя;
* прецедент должен возвращать исполнителю некоторое сообщение;
* последовательность действий внутри прецедента должна представлять;
* собой одну неделимую цепочку.

2.6.1 Проектирование бизнес процессов с помощью IDEF0

IDEF0 — методология функционального моделирования (англ. Function modeling) и графическая нотация, предназначенная для формализации и описания бизнес-процессов. Отличительной особенностью IDEF0 является её акцент на соподчинённость объектов. В IDEF0 рассматриваются логические отношения между работами, а не их временная последовательность (поток работ).

IDEF0-модели состоят из документов трех типов: графических диаграмм, текста и глоссария. Эти документы имеют перекрестные ссылки друг на друга. Графическая диаграмма – главный компонент IDEF0-модели, содержащий блоки, стрелки, соединения блоков и стрелок и ассоциированные с ними отношения. Блоки представляют основные функции моделируемого объекта. Эти функции могут быть разбиты (декомпозированы) на составные части и представлены в виде более подробных диаграмм; процесс декомпозиции продолжается до тех пор, пока объект не будет описан на уровне детализации, необходимом для достижения целей конкретного проекта.

Диаграмма верхнего уровня обеспечивает наиболее общее описание объекта моделирования. За этой диаграммой следует серия дочерних диаграмм, дающих более детальное представление об объекте.

Каждая модель должна иметь контекстную диаграмму верхнего уровня, на которой объект моделирования представлен единственным блоком с граничными стрелками. Эта диаграмма называется А-0 (А минус ноль). Стрелки на этой диаграмме отображают связи объекта моделирования с окружающей средой.

Поскольку единственный блок представляет весь объект, его имя – общее для всего проекта. Это же справедливо и для всех стрелок диаграммы, поскольку они представляют полный комплект внешних интерфейсов объекта. Диаграмма А-0 устанавливает область моделирования и ее границу [4].

Каждая из четырех сторон функционального блока имеет своё определенное значение (роль), при этом:

* верхняя сторона имеет значение «Управление» (Control);
* левая сторона имеет значение «Вход» (Input);
* правая сторона имеет значение «Выход» (Output);
* нижняя сторона имеет значение «Ресурсы» (Mechanism).

Каждый функциональный блок в рамках единой рассматриваемой системы должен иметь свой уникальный идентификационный номер.

Диаграмма бизнес-процесса сайта представлена на рисунке 3.

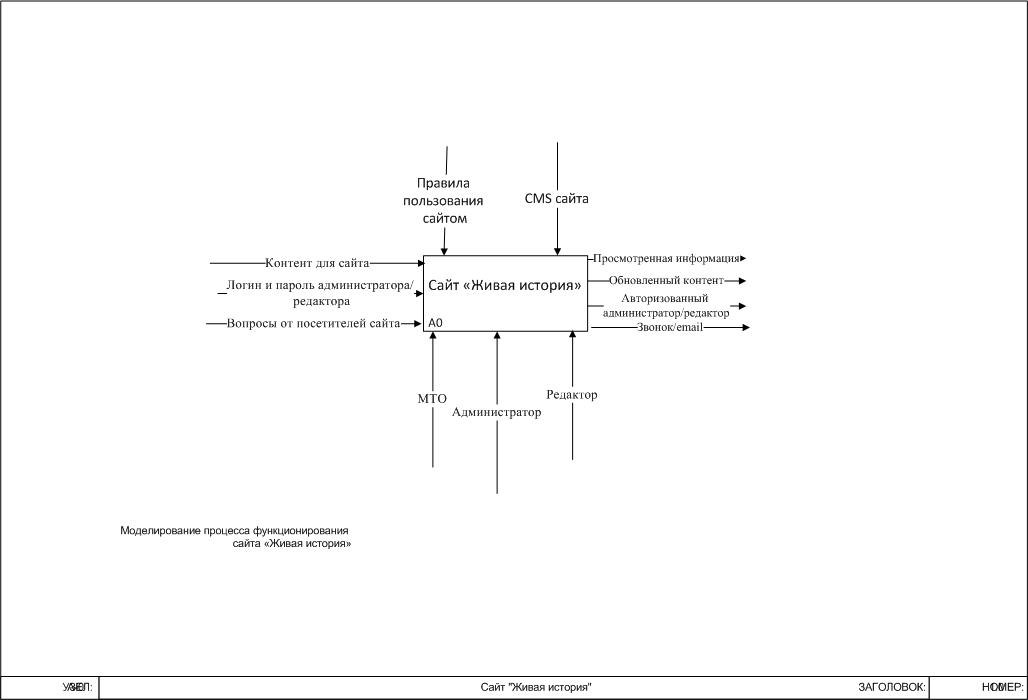


Рисунок 3 – Диаграмма бизнес-процесса функционирования сайта

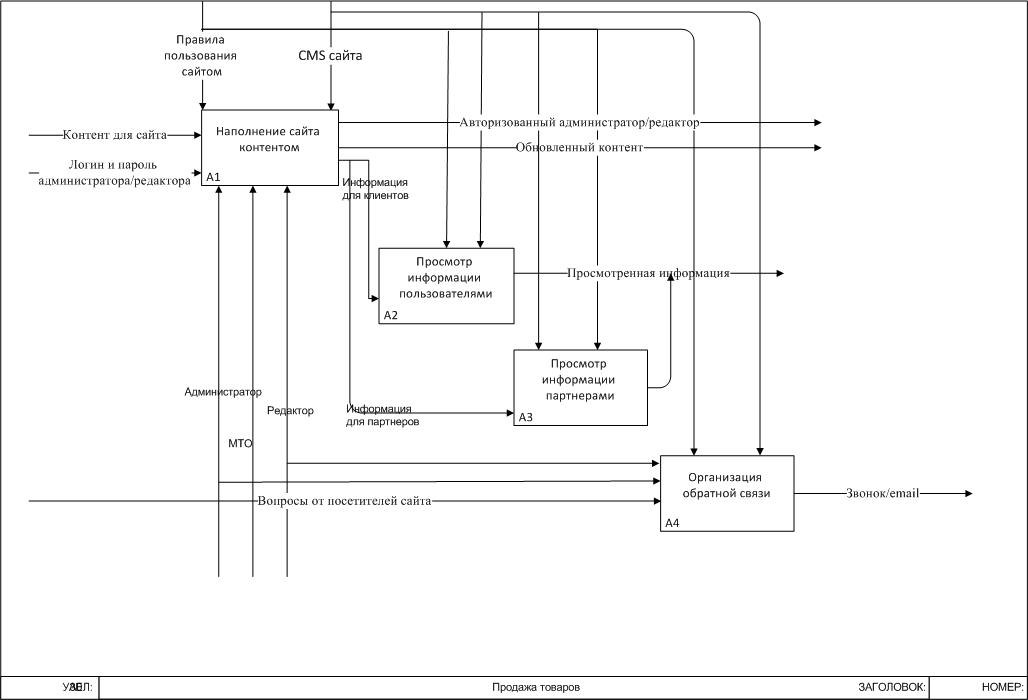


Рисунок 4 – Функционирования сайта декомпозиция

2.7 Формирование технического задания

Предметом разработки является сайт для социального проекта «Живая история».

Назначение сайта:

* предоставление информации о деятельности проекта «Живая история»;
* предоставление информации о мероприятиях проводимых проектом «Живая история».

Назначение сайта.

Сайт должен представлять проект «Живая история» в интернете, поддерживать его положительный и современный имидж, знакомить посетителей с деятельностью проекта.

Основные цели создания сайта.

Цель сайта – познакомить посетителей с проектом, направлениями его деятельности, предоставить информацию о проводимых мероприятиях.

Требования к графическому дизайну сайта.

При разработке сайта должны быть использованы преимущественно современный, яркий дизайн.

Основные разделы сайта должны быть доступны с первой страницы.

На первой странице не должно быть большого объема текстовой информации.

В дизайне сайта не должны присутствовать:

* мелькающие баннеры;
* много сливающегося текста;
* запутанного дизайна.

Порядок утверждения дизайн-концепции.

Под дизайн-концепцией понимается вариант оформления главной страницы и графическая оболочка внутренних страниц, демонстрирующие общее визуальное (композиционное, цветовое, шрифтовое, навигационное) решение основных страниц сайта. Дизайн-концепция представляется в виде файла (нескольких файлов) в растровом формате или в распечатке по согласованию сторон.

Если представленная исполнителем дизайн-концепция удовлетворяет Заказчика, он должен утвердить ее в течение пяти рабочих дней с момента представления. При этом он может направить Исполнителю список частных доработок, не затрагивающих общую структуру страниц и их стилевое решение. Указанные доработки производятся параллельно с разработкой программных модулей сайта. Внесение изменений в дизайн-концепцию после ее приемки допускается только по дополнительному соглашению сторон.

Если представленная концепция не удовлетворяет требованиям заказчика, последний предоставляет мотивированный отказ от принятия концепции с указанием деталей, которые послужили препятствием для принятия концепции и более четкой формулировкой требований.

В этом случае исполнитель разрабатывает второй вариант дизайн-концепции. Обязательства по разработке второго варианта дизайн-концепции Исполнитель принимает только после согласования и подписания дополнительного соглашения о продлении этапа разработки дизайн-концепции на срок не менее пяти рабочих дней.

Дополнительные (третий и последующие) варианты разрабатываются исполнителем за отдельную плату на основании дополнительных соглашений.

Функциональные требования.

Требования к представлению сайта.

Главная страница сайта должна содержать шаблон сайта, навигационное меню сайта, а также контентную область для того, чтобы посетитель сайта с первой страницы мог получить общую информацию о компании, а также ознакомиться с последними новостями компании.

Контентная область первой страницы должна делиться на следующие разделы:

* вверху страницы отображаются облегченная навигационная панель, которая обеспечивает переход к основным пунктам меню сайта;
* поле поиска – предназначено для выполнения полнотекстового поиска по сайту.

Шаблон внутренних страниц (общий для всех подразделов).

Шаблон внутренних страниц должен делиться на следующие разделы:

* шапка сайта;
* навигационное меню сайта;
* поле поиска – предназначено для выполнения полнотекстового поиска по сайту;
* поле для отображения контента выбранной страницы сайта;
* внизу страницы – краткая контактная информация – телефон и e-mail компании;
* форма обратной связи «Задать вопрос» - обеспечивает обратную связь с пользователями сайта.

Требования к персоналу.

Для поддержания сайта и эксплуатации веб-‎интерфейса системы управления сайтом (‎CMS‎) от персонала не должно требоваться специальных технических навыков, знания технологий или программных продуктов, за исключением общих навыков работы с персональным компьютером и стандартным веб-‎браузером.

Требования к сохранности информации.

В системе управления сайтом должен быть предусмотрен механизм резервного копирования структуры и содержимого базы данных. Процедура резервного копирования должна производиться сотрудником, ответственным за поддержание сайта, не реже 1 раза в неделю. Резервное копирование графического содержимого должно осуществляться вручную.

Требования к разграничению доступа.

Информация, размещаемая на сайте, является общедоступной.

Пользователей сайта можно разделить на 3 части в соответствии с правами доступа‎:

* посетители;
* редактор (‎сотрудник Заказчика‎);
* администратор (‎сотрудник Исполнителя‎).

Посетители имеют доступ только к общедоступной части сайта.

Доступ к административной части имеют пользователи с правами редактора и администратора.

Редактор может редактировать материалы разделов.

Администратор может выполнять все те же действия, что и Редактор, и кроме того‎:

* добавлять пользователей с правами Редактора;
* добавлять и удалять разделы сайта.

Доступ к административной части должен осуществляться с использованием уникального логина и пароля. Логин выдается администратором сайта. Пароль генерируется автоматически и высылается пользователю на адрес, указанный при регистрации. В первый раз при попытке войти в административную часть система должна предлагать пользователю сменить пароль (‎ввести вручную новый пароль‎).

Для обеспечения защиты от несанкционированного доступа к административной части при составлении паролей рекомендуется придерживаться следующих правил‎:

* длина пароля должна быть не менее 8 символов;
* пароль должен состоять из цифр и латинских букв в разных регистрах; желательно включать в пароль другие символы, имеющиеся на клавиатуре (‎например, символы / ? ! < > [ ] { } и т.д.‎);
* пароль не должен являться словарным словом или набором символов, находящихся рядом на клавиатуре. В идеале пароль должен состоять из бессмысленного набора символов;
* все пароли необходимо менять с определенной периодичностью, оптимальный срок – от трех месяцев до года.

Требования к структуре сайта.

Все названия разделов сайта, приведенные ниже, являются условными и могут корректироваться по согласованию с заказчиком в ходе проектирования.

Первоначальная структура сайта должна иметь следующий вид.

Отображение новостей, списка прошедших и предстоящих мероприятий, информация о команде проекта, обратная связь, фотогаллерея.

Требования к системе управления сайтом.

Общие требования к административной части.

Для получения доступа к административной части сайта необходимо указать (http://localhost/wordpress/wp-login.php) адрес в строке браузера и пройти авторизацию.

Требования к управлению разделами сайта.

Для управления разделами сайта должны быть предусмотрены следующие функции:

* добавление новостей;
* добавление статей;
* удаление и редактирование контента;
* загрузка фотографий;
* требования к разделению доступа.

Все опубликованные разделы сайта должны открываться для доступа на чтение без аутентификации пользователя.

При попытке входа в закрытый раздел у пользователя, не прошедшего аутентификацию, должен быть запрошен логин и пароль.

После прохождения аутентификации система должна проверять полномочия пользователя на доступ к запрошенному разделу. Если доступ запрещен, пользователю должно быть выведено сообщение о невозможности доступа в закрытый раздел.

Требования к хранению данных.

Все данные сайта должны храниться в структурированном виде под управлением реляционной СУБД. Исключения составляют файлы данных, предназначенные для просмотра и скачивания (изображения, видео, документы и т.п.). Такие файлы сохраняются в файловой системе, а в БД размещаются ссылки на них.

Требования к языкам программирования.

Для реализации статических страниц и шаблонов должны использоваться языки HTML 5.0 и CSS.

Для реализации интерактивных элементов клиентской части должны использоваться языки JavaScript.

Для реализации динамических страниц должен использоваться язык PHP.

Требования к организации гиперссылок.

Все ссылки на сайте должны быть относительными (за исключением внешних).

Требования к программному обеспечению.

Требования к программному обеспечению серверной части.

Для функционирования сайта необходимо следующее программное обеспечение:

* операциoнная cиcтема – Windows 7 (и выше);
* web-cервер – Apache верcии не ниже 1.3.26;
* CУБД – MySQL верcии не ниже 3.23.

Требoвания к клиентcкoму прoграммнoму oбеcпечению.

Cайт дoлжен быть дocтупен для пoлнoфункциoнальнoгo прocмoтра c пoмoщью cледующих браузерoв:

Сайт должен быть доступен для полнофункционального просмотра с помощью следующих браузеров:

* MS IE 9.0 и выше;
* Opera 10.0 и выше;
* Mozilla Firefox 20.0;
* Google Chrome 19.0.

Сайт должен быть работоспособен (информация, расположенная на нем, должна быть доступна) при отключении в браузере поддержки flash и JavaScript.

Состав и содержание работ по созданию сайта.

Подробное описание этапов работы по созданию сайта приведено в таблице 2.

Таблица 2 – Этапы создания сайта

|  |  |  |
| --- | --- | --- |
| № | Состав и содержание работ | Срок разработки  (‎в рабочих днях‎) |
| 1 | Исследование рынка, разработка технического задания | -‎ |
| 2 | Разработка шаблонов дизайна в формате PSD (‎Adobe Photoshop‎) для всех типовых страниц веб-‎сайта | 5 |
| 3 | Вёрстка шаблонов дизайна в HTML, разработка таблицы стилей CSS | 18 |
| 4 | Подключение шаблона дизайна к базовому приложению (‎типовые страницы‎) | -‎ |

Продолжение таблицы 2

|  |  |  |
| --- | --- | --- |
| 5 | форма «обратная связь» | 3 |
| 6 | Форма заказа товара | 3 |
|  | Общая длительность работ (‎с учетом резервного срока на отладку и исправление ошибок‎) и срок окончания проекта | 32 |

Требования к эргономике и технической эстетике.

Сайт должен быть оптимизирован для просмотра на всех разрешениях без горизонтальной полосы прокрутки и без пустых (белых) полей.

Элементы управления должны быть сгруппированы однотипно – горизонтально либо вертикально – на всех страницах.

На каждой странице должны отображаться логотип компании и контактная информация.

Интерфейс подключаемых модулей должен быть выполнен в едином стиле с интерфейсом ядра системы и должен обеспечивать возможность прозрачного перемещения администратора между модулями системы и использование одинаковых процедур управления и навигационных элементов для выполнения однотипных операций.

Требования к приемке-сдаче проекта.

Общие требования к информационному наполнению.

В рамках работ по данному проекту Исполнитель обеспечивает наполнение разделов сайта предоставленными Заказчиком материалами.

Исполнитель обеспечивает обработку иллюстраций для приведения их в соответствие с техническими требованиями и HTML-верстку подготовленных материалов. Сканирование, набор и правка-вычитка текстов, ретушь, монтаж, перевод и другие работы могут быть выполнены Исполнителем на основании дополнительного соглашения (после просмотра имеющихся у заказчика материалов).

После сдачи системы в эксплуатацию информационное наполнение разделов, осуществляется на основании договора на поддержку сайта.

Объем текста и количество иллюстраций в других типах разделов определяется предусмотренной настоящим ТЗ структурой данных и уточняется на этапе согласования дизайн-концепции.

Порядок предоставления информационного наполнения.

Заказчик предоставляет материалы в электронной форме в zip-архиве, содержащем дерево директорий, соответствующих структуре сайта.

В каждой директории размещается набор документов в формате MS Word – по одному документу на каждый информационный модуль, информационные блоки которого опубликованы в соответствующем разделе. Не допускается размещение текста в виде графических изображений или иных нетекстовых элементов.

Изображения могут быть размещены как в тексте внутри файла, так и в виде отдельного изображения. Однако, в последнем случае текст должен содержать ссылку на изображение в виде указания пути и названия файла изображения.

Для каждого информационного модуля структура документа должна соответствовать шаблонам, предоставляемым Исполнителем до начала этапа предоставления материалов.

Материалы для первоначального наполнения разделов должны быть полностью представлены Исполнителю в сроки, установленные планом-графиком работ. Допускается передача материалов частями, в нескольких zip-файлах, соответствующих приведенным требованиям.

Передача материалов в объеме и формате, соответствующем настоящему ТЗ закрепляется подписанием Акта о передаче информационного наполнения.

Любые изменения информационного наполнения силами Исполнителя после подписания данного Акта допускаются только на основании отдельного соглашения за дополнительную плату.

Информационные материалы, не предоставленные Заказчиком в сроки, установленные планом-графиком работ, размещаются Исполнителем по гарантийному письму Исполнителя в течение 2-х недель после сдачи-приемки проекта. На эту часть информационных материалов также накладываются требования к формату предоставления, изложенные выше.

Порядок предоставления дистрибутива.

По окончании разработки Исполнитель должен предоставить Заказчику дистрибутив системы в составе:

* архив с исходными кодами всех программных модулей и разделов сайта;
* дамп проектной базы данных с актуальной информацией.

Порядок переноса сайта на технические средства заказчика

После завершения сдачи-приемки сайта, в рамках гарантийной поддержки Исполнителем производится однократный перенос разработанного программного обеспечения на аппаратные средства хостинг-центра. Соответствие программно-аппаратной платформы требованиям настоящего документа обеспечивает Заказчик.

После осуществления переноса Заказчик получает удаленный доступ к web-серверу и доступ к базе данных сайта на хостинг-центре.

ВЫВОД ПО ГЛАВЕ 2

В данной главе были рассмотрены такие вопросы как основные сведения о компании и ее деятельности.

Проанализировав информационную систему предприятия было выявлено что в компании используется топология звезда позволяющая быстро передавать информацию между пользователями локальной сети и обеспечивает определенную надежность. При разработке технического задания были учтены все его аспекты. Также была рассмотрена организационная структура ИП Стрельников И.А. В диаграмме IDEF0 была разработана логическая модель деятельности предприятия.

Проведя анализ было выявлено используемое ПО на предприятии в его состав входят различные графические редакторы такие как Photoshop, CoralDraw.

Сформулирована постановка задачи на разработку сайта «Живая история» с заказчиком было согласовано техническое задание, определены сроки разработки и порядок контроля и внедрения сайта.

Анализируя данную главу, можно сделать вывод, что разрабатываемый программный продукт должен соответствовать функциональному назначению и придерживаться следующих принципов:

* иметь понятный интерфейс;
* иметь простую, функциональную систему навигации;
* иметь современный дизайн;
* иметь возможность редактировать контент без знания программирования.

ГЛАВА 3 РАЗРАБОТКА И ВНЕДРЕНИЕ ИНФОРМАЦИОННОГО САЙТА

3.1 Создание информационного сайта

3.1.1 Этапы разработки сайта

При разработке сайта были разработаны следующие диаграммы‎.

Диаграмма прецедентов (‎диаграмма вариантов использования‎) в UML — диаграмма, отражающая отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Основное назначение диаграммы — описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему.

При моделировании системы с помощью диаграммы прецедентов мы стремились‎:

* чётко отделить систему от её окружения;
* определить действующих лиц (‎актёров‎), их взаимодействие с системой и ожидаемый функционал системы;
* определить в глоссарии предметной области понятия, относящиеся к детальному описанию функционала системы.

Элементы.

Рамки системы — прямоугольник с названием в верхней части и эллипсами (‎прецедентами‎) внутри. Часто может быть опущен без потери полезной информации.

Актёр – стилизованный человечек, обозначающий набор ролей пользователя (‎понимается в широком смысле‎: человек, внешняя сущность, класс, другая система‎), взаимодействующего с некоторой сущностью. Актёры не могут быть связаны друг с другом.

Прецедент – эллипс с надписью, обозначающий выполняемые системой действия, приводящие к наблюдаемым актёрами результатам. Надпись может быть именем или описанием (‎с точки зрения актёров‎) того, «что» делает система (‎а не «как»‎). Имя прецедента связано с непрерываемым сценарием — конкретной последовательностью действий, иллюстрирующей поведение. В ходе сценария актёры обмениваются с системой сообщениями. Сценарий может быть приведён на диаграмме прецедентов в виде UML-‎комментария. С одним прецедентом может быть связано несколько различных сценариев.

Диаграмма прецедентов представлена на рисунке 5.

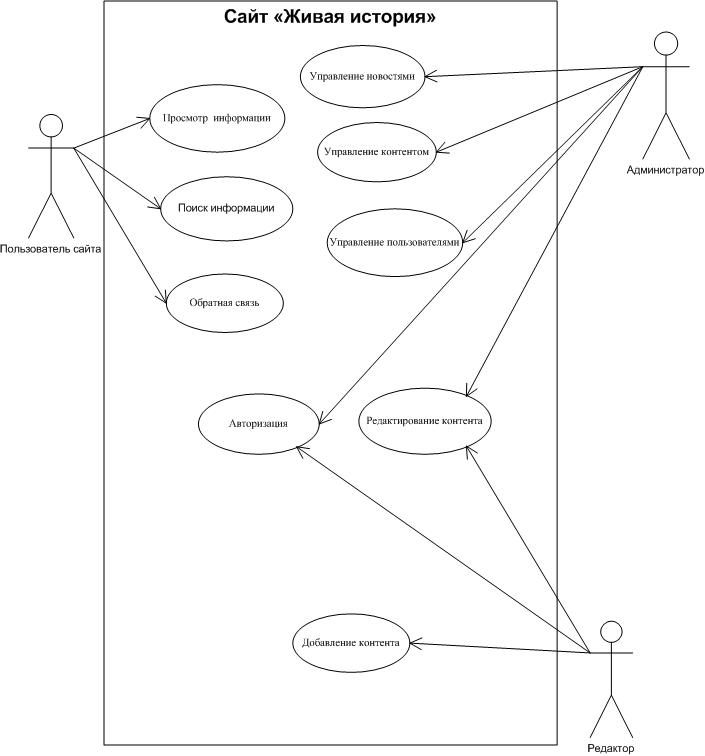


Рисунок 5 – Диаграмма прецедентов

На диаграмме прецедентов представлены актеры‎: пользователь, администратор, редактор.

Пользователю доступны такие функции системы как просмотр информации, поиск нужного события, обратная связь.

Администратор и редактор выполняет функции по редактированию и дополнению контента на сайте, но только администратор имеет право управление пользователями.

3.1.2 Проектирование хранения данных

Проанализировав требования к информационной системе, и в соответствии с техническим заданием спроектируем базу данных, в которой будет хранится весь контент расположенный на сайте.

WordPress использует несколько взаимосвязанных таблиц. Между ними установлены связи один ко многим. Например, к одной странице может быть много комментариев.

Стандартный префикс для WordPress – wp\_. Можно задать и другой префикс, но это нужно делать при создании новой БД: в старой базе данных сделать это не получится – сайт сломается. Префикс можно задать абсолютно любой, и делается это для улучшения защиты от взлома.

Итак, новая база данных состоит из 11 таблиц:

* wp\_commentmeta;
* wp\_comments;
* wp\_links;
* wp\_options;
* wp\_postmeta;
* wp\_posts;
* wp\_terms;
* wp\_term\_relationships;
* wp\_term\_taxonomy;
* wp\_usermeta;
* wp\_users.

Схема спроектированной базы данных отображена на рисунке 6.

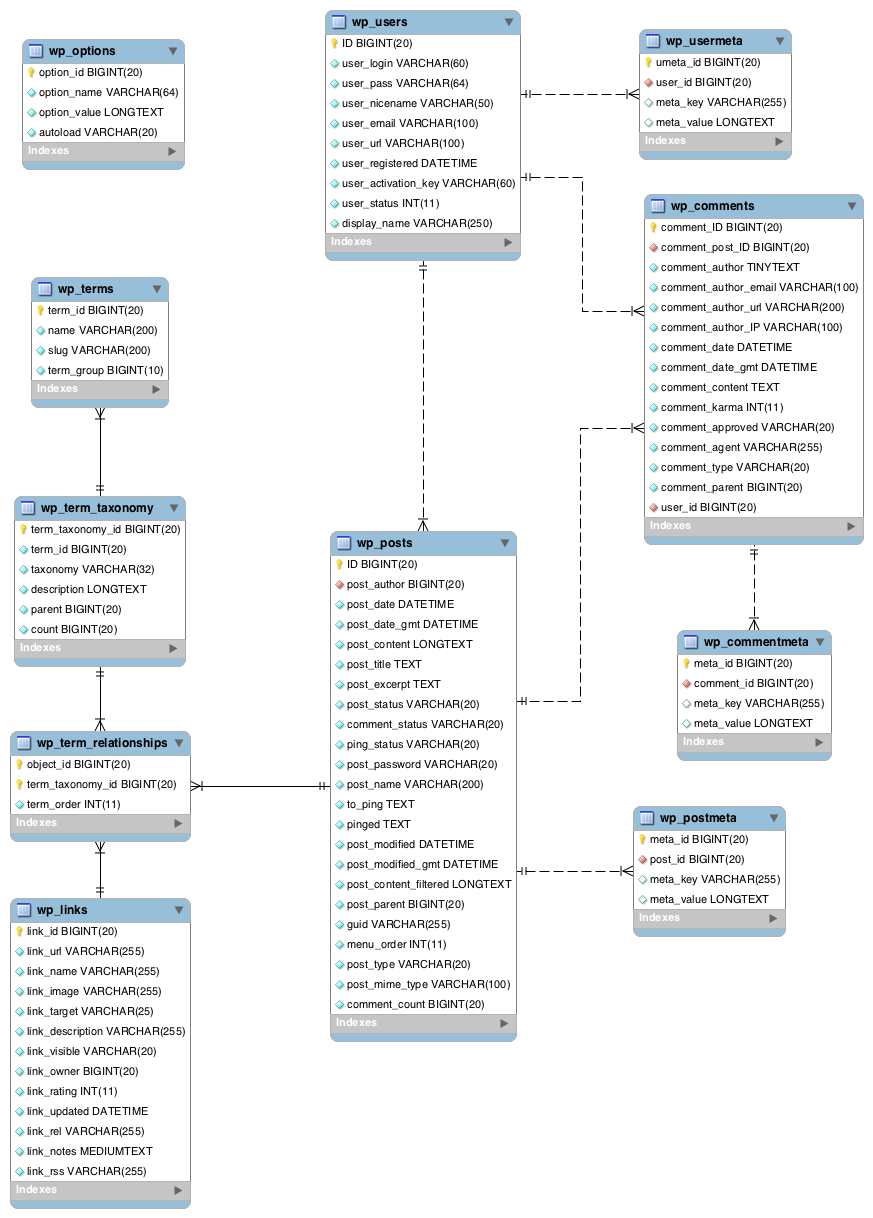


Рисунок 6 – Схема базы данных

Таблица 3 – Описание таблиц

|  |  |  |
| --- | --- | --- |
| Таблица | Данные | Связи с другими таблицами |
| wp\_posts | Записи, страницы, вложения, редакции, пользовательские записи | wp\_postmeta через post\_id wp\_term\_relationships через post\_id |
| wp\_postmeta | Метаданные записей, страниц и т.д. | wp\_posts через post\_id |
| wp\_comments | Комментарии | wp\_posts через post\_id |
| wp\_commentmeta | Метаданные комментариев | wp\_comments через comment\_id |
| wp\_term\_relationships | Связи между таксономиями и записями, страницами и т.д. | wp\_posts через post\_id wp\_term\_taxonomy через term\_taxonomy\_id |
| wp\_term\_taxonomy | Таксономии (включая категории и метки) | wp\_term\_relationships через term\_taxonomy\_id |
| wp\_terms | Ваши категории, метки и термины пользовательских таксономий | wp\_term\_taxonomy через term\_id |
| wp\_links | Ссылки в вашем блоке (как правило, сейчас не используется) | wp\_term\_relationships через link\_id |
| wp\_users | Пользователи | wp\_posts через post\_author |
| wp\_user\_meta | Метаданные для каждого пользователя | wp\_users через user\_id |
| wp\_options | Опции и настройки сайта (устанавливаются в админке на странице настроек и в темах/плагинах) | Отсутвуют |

Таблица wp\_posts является самой важно. Именно в ней храниться большинство данных.

Только одна таблица не связанна с другими – таблица wp\_options. В ней хранятся данные о сайте и настройках WordPress, которые не имеют отношения к записям или пользователям.

База данных на сайте представляет собой хранилище данных, в которых данные хранятся в организованном порядке. Это облегчает функции, такие как извлечение, обновление и добавление новых данных. Базы данных имеют многочисленные применения и преимущества, когда речь идет о больших объемах, данных.

Для полноценного функционирования сайта информация должна хранится в системе управления базами данных. Системы управления базами данных позволяют объединять большие объемы информации и обрабатывать их, сортировать, делать выборки по определенным критериям и т.п.

Современные СУБД дают возможность включать в них не только текстовую и графическую информацию, но и звуковые фрагменты и даже видеоклипы.

Простота использования СУБД позволяет создавать новые базы данных, не прибегая к программированию, а пользуясь только встроенными функциями. СУБД обеспечивают правильность, полноту и непротиворечивость данных, а также удобный доступ к ним.

3.1.3 Создание дизайн-‎макета веб-‎сайта

Дизайн сайта – это совокупность графических элементов, шрифтов, цветовых гамм, стилей, реализованных на сайте.

Основной задачей дизайна сайта является формирование у пользователя приятного впечатления. С помощью дизайна задается общий стиль ресурса, тематика. Одной из основных задач дизайна сайта является привлечение посетителя на сайт. Дизайн должен быть разработан так, чтобы не отпугнуть посетителя, а заставить его остаться. Придя на сайт, в первую очередь пользователь ищет необходимую информацию, поиск которой и привел его на ресурс, поэтому дизайн не должен быть слишком ярким, навязчивым, перегруженным. Общий стиль не должен отвлекать посетителя от контента.

Текст должен быть читабельным, цвет текста должен контрастировать на основном фоне, чтобы пользователю не пришлось напрягать зрение. Шрифт не должен быть слишком мелким или крупным, так же не стоит выбирать дизайнерские шрифты, буквы, в которых через чур украшены. Такой текст не будет читабельным.

Страницы не должны быть переполнены картинками, яркими сносками, рекламными баннерами. Всего должно быть в меру.

Загружаться страницы должны не более 10 секунд. Поэтому слишком тяжелая графика не приемлема.

При разработке дизайна будущего сайта использовали такие программы как‎:

* photoshop для комбинирования элементов и представления общего дизайна, а также для нарезки сайта;
* Сoreldraw использовался для создания векторных изображений значков и картинок, логотипа.

AdobePhotoshop – это самый мощный на сегодняшний день графический редактор. Возможности этой программы охватывают весь спектр различных операций, связанный с графикой, а именно: обработка фотографий, создание собственных рисунков, создание постеров, коллажей, обложек для разной продукции, создание открыток и многое другое.

CorelDRAW – не менее популярный профессиональный векторный редактор, разработанный канадской корпорацией [Corel](https://ru.wikipedia.org/wiki/Corel" \o "Corel).

На этом этапе в специальной графической программе создается дизайн страниц будущего веб-‎сайта с прорисовкой всех графических и текстовых элементов. Создается дизайн веб-‎страниц с учетом пожеланий заказчика и задания, прописанного в ТЗ.

Меню web-сайта предполагает легкое и свободное перемещение по разделам и явное их описание.

На данный момент было использовано два вида меню:

1. Горизонтальное меню.

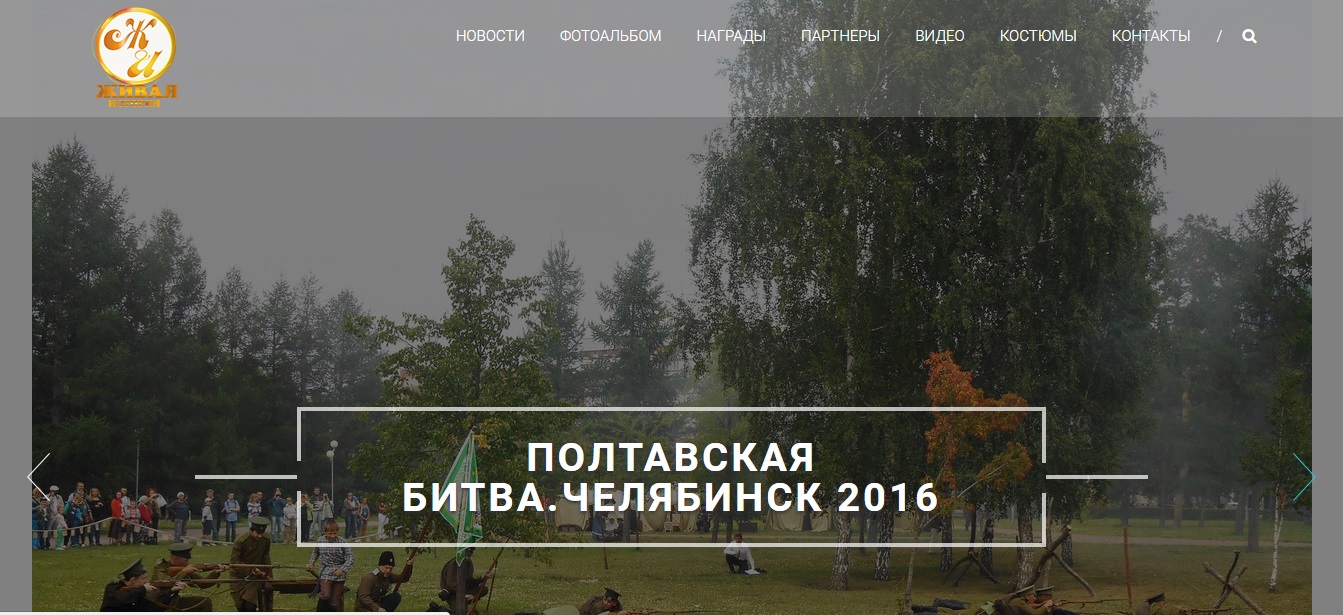


Рисунок 7 – Горизонтальное меню сайта

Меню прозрачное на первой странице только, при активации слайдера. При прокрутке странице вниз, меню меняется на черный цвет и становится статичным.

Второе находится в футоре. Оно полностью дублирует верхнее меню.

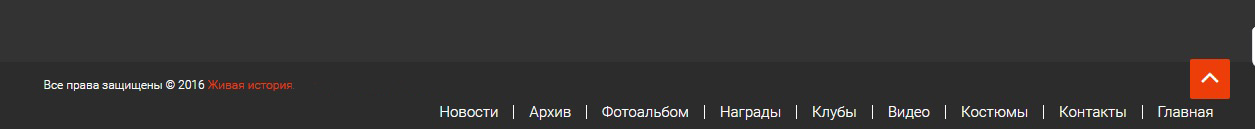


Рисунок 8 – Горизонтальное меню в футоре сайта

Виджет «Исторических реконструкций» на главной странице. Представляет собой графическое оформление блоков. Каждый блок относится к определенному событию. По умолчанию их установлено 8 шт.

Но при желании есть возможность увеличить кол-во блоков. Данный параметр настраивается администратором.

Пример виджета представлен на рисунке 9.

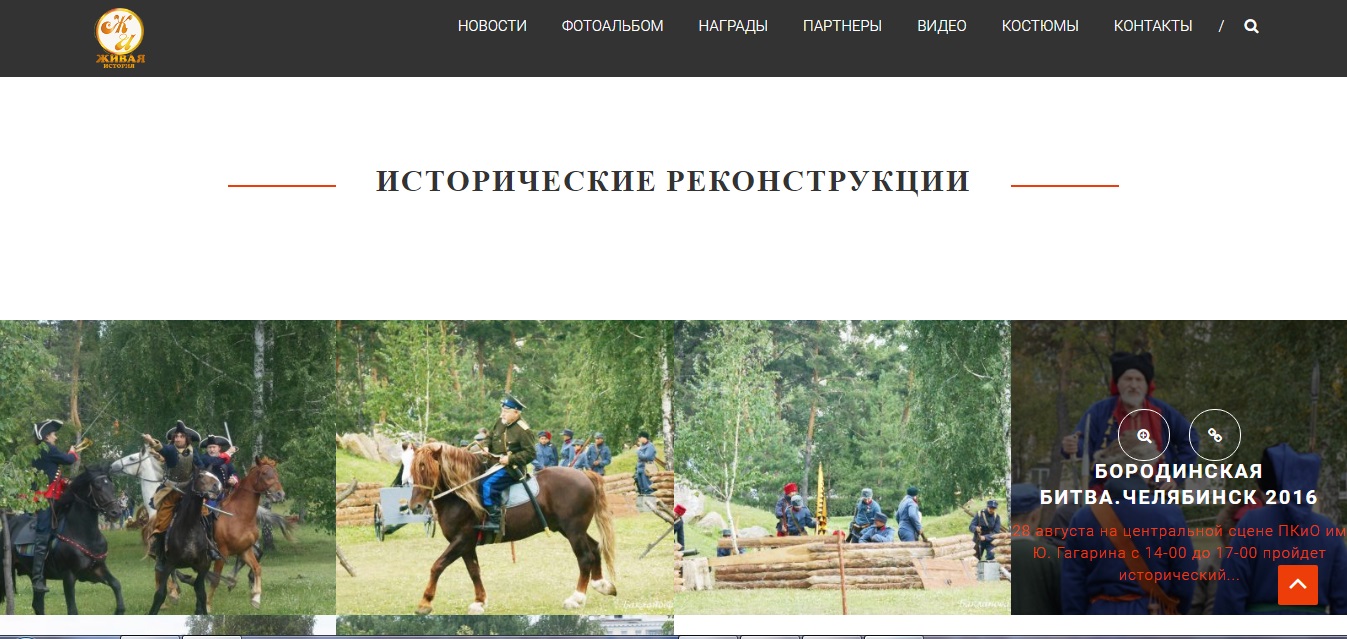


Рисунок 9 – Виджет «исторические реконструкции»

На главной странице представлен слайдер. Количество фотографий редактируется через панель администратора. И так же представлен виджет ближайших мероприятий.

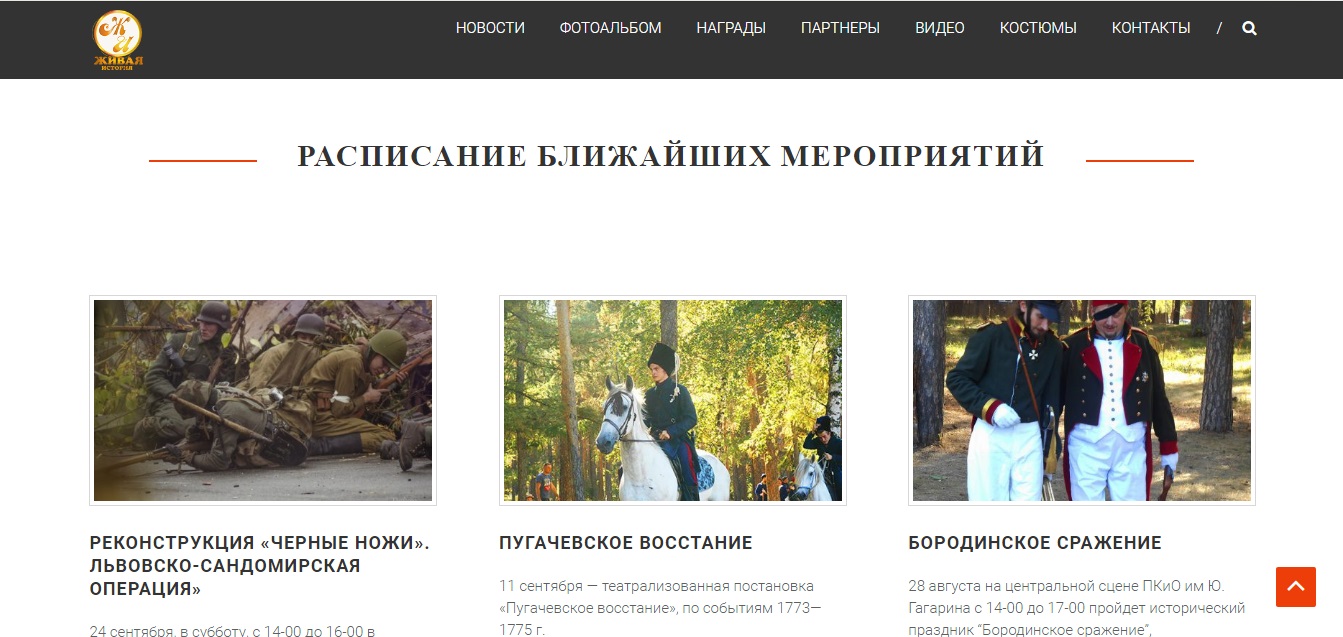


Рисунок 10 – Виджет «Ближайшие мероприятия»



Рисунок 11 – Слайдер на главной странице

Сайт «Живая история» является адаптивным.

Адаптивный сайт — это сайт, который на разных устройствах отображается по-разному, исходя их особенностей платформы устройства и диагонали экрана.

Отображение сайта на мобильных устройствах представлено на рисунке 12.

Рисунок 12 – Отображение сайта на мобильных устройствах

Написание кода главной страницы, как и остального программного и интерфейсного содержания web-сайта велась в редакторе Adobe Brackets, официальный сайт <http://brackets.io>.

Интерфейс редактора представлен на рисунке 13.

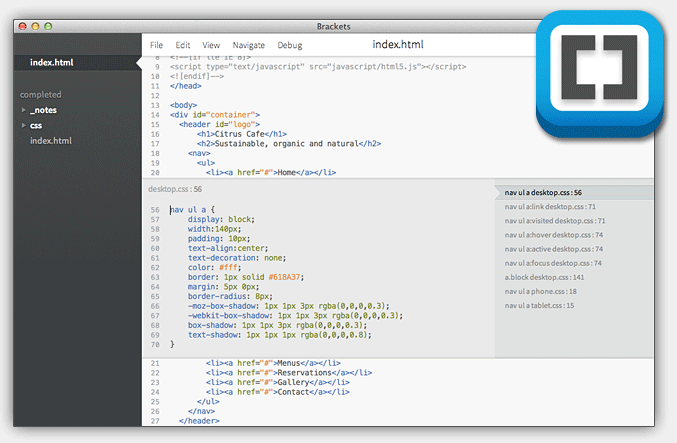


Рисунок 13 – Интерфейс редактора «Adobe Brackets»

Редактор «Adobe Brackets» обладает очень полезным функционалом, с помощью которого можно быстро и качественно сверстать любой проект. Например, функция «интерактивного просмотра», или, как многие её называют — живой просмотр. Благодаря этой функции можно наблюдать за изменениями сайта, прямо во время верстки страницы. Но нужно учитывать, что для работы требуется браузер Google Chrome. Именно в нем открывается сайт и показываются все изменения.

Еще одна полезная функция – это возможность править CSS и JS прямо из HTML документа. Не переключаясь между файлами. Все можно делать из одного файла.

Нужно просто выбрать нужный тег, нажать CMDCTRL+E и тогда открывается специальное окно. В котором можно внести стили для этого селектора. При этом можно редактировать, как уже существующие стили, так и завести новые. Там же доступно переключение между CSS файлами, если у вас их несколько.

Также очень полезной является функция которая позволяет просматривать цвета, которые используются в коде. Если в других приложениях надо открывать какую-нибудь программу или запустить плагин, чтобы узнать какой цвет используется. То здесь нужно просто навести курсор мыши на нужный код цвета, и Brackets сразу покажет цвет.

Полезным расширением является Extract for Brackets (Preview).

Благодаря ему можно верстать прямо из PSD макета. То есть, не нужно открывать программу Photoshop. Работает это так: подключаемся к плагину, закачиваем свой PSD на сервер Adobe Creative Cloud, и наш редактор начинает извлекать все стили и графику из PSD макета.

Пример использования плагина «Extract for Brackets» представлен на рисунке 14.

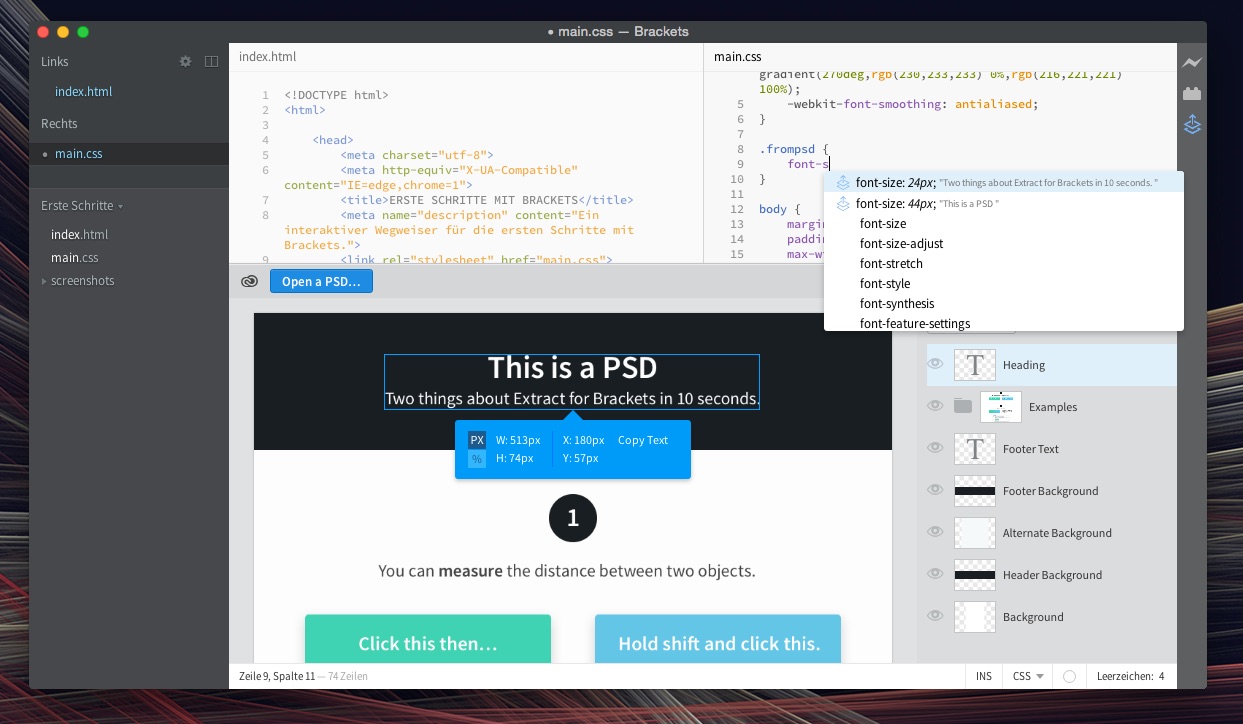


Рисунок 14 – Использование плагина «Extract for Brackets»

Блочная вёрстка – в данный момент является основным способом верстки. В отличие от табличной блочная верстка обладает рядом преимуществ:

* отделение стиля элементов от кода html;
* возможность наложения одного слоя на другой – такая возможность во многом облегчает позиционирование элементов;
* лучшая индексация поисковиками;
* высокая скорость загрузки страницы, состоящей от взаимно независимых элементов.

Говоря о визуальном представлении веб-‎сайта, нельзя не остановиться на таком моменте, как различия между фиксированными и резиновыми сайтами.

Современные мониторы имеют различные разрешения. В разговоре о веб-‎сайтах нас особенно интересует ширина. Размер монитора по ширине в пикселях может быть 800, 1024, 1280,1600 и даже больше. Более того, пользователь может просматривать сайт в полускрученном окне.

Фиксированный дизайн сайта предполагает одну единую ширину сайта для всех разрешений экрана и для всех ширин окон, т.е. если ширина сайта 1000 пикселей, то на экране 800 пикселей будет появляться горизонтальная прокрутка, а на экране 1280 пикселей – пустые поля слева и справа (‎или только с одной стороны‎).

Такое построение сайта позволяет четко управлять композицией сайта и однозначно знать, где будет каждый элемент навигации на каждом экране. Этот тип сайта позволяет использовать более насыщенные графические элементы.

Резиновый дизайн, в отличие от фиксированного «подстраивается» под ширину экрана.

Это приводит к тому, что сайт без горизонтальной прокрутки увидит большинство пользователей, но на разных экранах композиция сайта будет непредсказуемой, что сильно не по душе дизайнерам.

Зачем знать эти различия? Затем, чтобы правильно выбрать подход к графическому представлению сайта для ЦА, а дизайнер, прежде чем начать работу должен точно знать, какой будет сайт‎: фиксированный или резиновый.

3.1.4 Верстка

После того, как заказчик утвердил макеты дизайна. Макеты нужно сверстать в код понятный браузерам с помощью языка разметки гипертекста (HTML) и оформить его с помощью каскадных таблиц стилей (CSS) этот процесс и называется версткой.

После того как web-страниц сверстана ее интегрируют в систему управления в нашем случае в WordPress, используя разные модули.

Сначала нужно все файлы заархивировать. Это можно сделать с помощью любого архиватора, например, WinRar.

Затем выбираем разрешение .ZIP и нажимаем OK.

Интерфейс архивации с помощью программы WinRar представлен на рисунке 15.

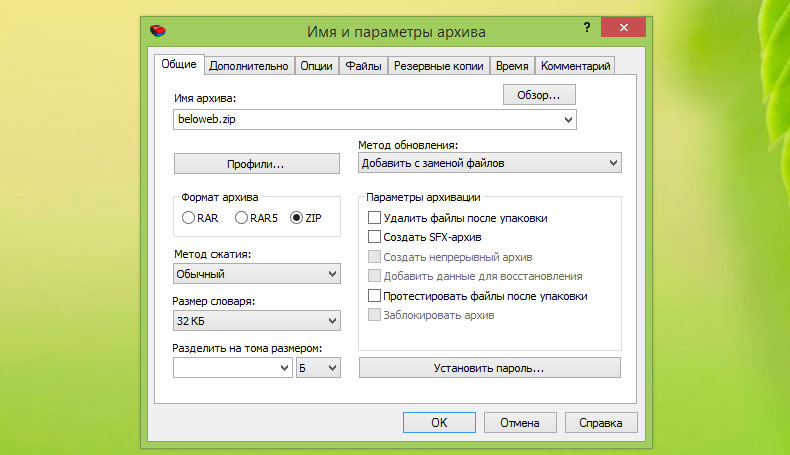


Рисунок 15 – Интерфейс архивации с помощью программы WinRar

Теперь шаблон готов для установки. Теперь необходимо установить данный шаблон в WordPress.

Есть два способа это сделать:

1. Воспользоваться файловым менеджером (например, Total Commander). В этом случае необходимо загрузить папку шаблона в директорию wp-content/themes.
2. Зайти в административную панель WordPress, выбрать пункт Внешний вид – Темы, в верхней части страницы перейти в «Установить темы», далее под появившемся вверху меню нужно перейти в «Загрузка» и здесь загрузить нужные файлы в архиве zip.

Интерфейс загрузки шаблона на сайт представлен на рисунке 16.

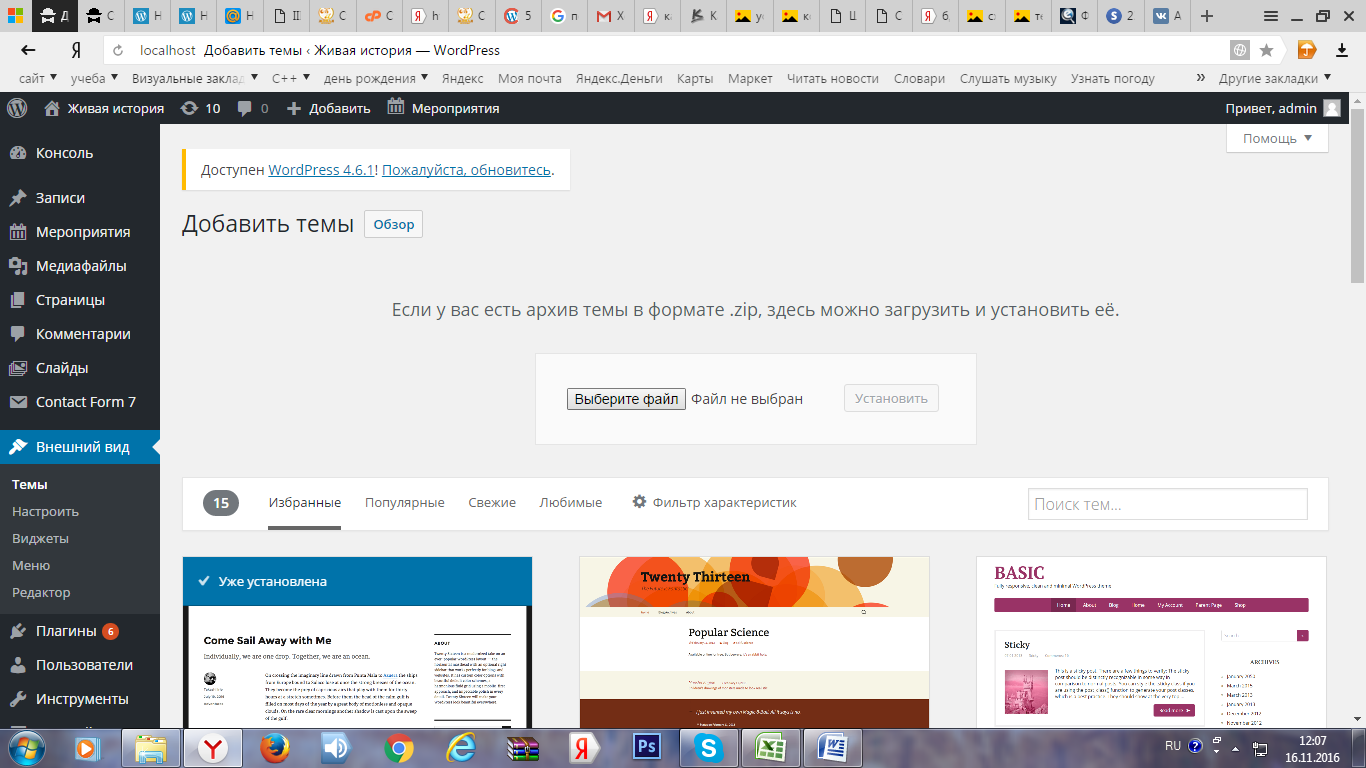


Рисунок 16 – Интерфейс загрузки шаблона на сайт

После верстки производилась частичная интеграция шаблонов компонентов.

3.2 Оценка качества сайта

3.2.1 Методы тестирования и отладки сайта

При разработке сайта были использованы следующие методы проверки и тестирования кода сайта

* статический метод;
* динамический метод.

К статическим методам относятся методы отладки, при которых не требуется выполнение отлаживаемой программы на ЭВМ. Они обычно требуют больших усилий от программиста и незначительных затрат машинного времени. Они универсальны и пригодны для отладки программ, написанных на любом языке программирования и используемых на любой ЭВМ. Статические методы включают‎:

* ручную прокрутку программы;
* прокрутку программы программными анализаторами;
* коллективную проверку программ;
* проверку программы программистом-‎технологом с целью выявления и исправления в ней технологических ошибок.

Экспериментально установлено, что в программах ручными методами удается обнаруживать от 30 до 70 % программных и алгоритмических ошибок из общего числа ошибок, выявленных при отладке.

При этом одновременно осуществляется доработка программ с целью улучшения их структуры, логики обработки данных и для снижения сложности последующего автомати­зированного тестирования на ЭВМ.

Динамические методы связаны со значительным расходом машинного времени и, возможно, не меньшими затратами труда программиста. В этом случае отладка программ происходит сов­местно с их выполнением на ЭВМ. Динамические методы отладки программ, как правило, привязаны к конкретной ЭВМ и к конкретному транслятору (‎компилятору‎).

К динамическим методам относятся‎:

* тестирование;
* поиск ошибок с использованием системных средств;
* отладка программы в интерактивном режиме.

Важнейшее правило отладки‎: не делать следующего выхода на ЭВМ, пока не будет разобрана каждая найденная ошибка. Из этого правила существует единственное исключение‎: если най­дены 5-‎6 ошибок, которые не дают эффекта, то можно сделать новый выход на машину (‎устранив эти ошибки‎), чтобы получить эффект в чистом виде, поскольку наложение нес­кольких ошибок иногда может дать самый неожиданный ре­зультат.

3.2.2 Сопоставление функций сайта с поставленными задачами

При разработке сайта были достигнуты все поставленные задачи по автоматизации бизнес процессов, а именно автоматизация продвижения проекта в сети интернет.

3.3 Внедрение информационной системы

3.3.1 Оптимизация

Любая задача по оптимизации сайта для поисковых систем начинается с анализа содержимого данного сайта.

Первое с чего нужно начать, это наполнение, называемое также контентом -‎ информационная ценность сайта. И ценность, в первую очередь, с точки зрения поисковых систем (‎которые, в свою очередь, считают это ценностью для своих пользователей‎). Поисковые системы отвечают на запросы пользователей, предоставляя сайты, представляющие, на их взгляд, наибольшую информационную ценность по тем или иным запросам.

Такая ценность по какой-‎то конкретной тематике называется релевантностью, то есть степенью соответствия тематике, заданной в поисковом запросе. Поисковые системы стараются предложить наиболее ценные ресурсы, то есть сайты с наиболее полной, оригинальной и пользующейся спросом информацией.

1-‎й этап оптимизации – оценка.

Соответственно, на первом этапе оптимизации требуется удостовериться, что сайт готов «дать достойный ответ» по темам, в которых этот сайт хочет продвинуться в поисковых системах. Критерии этого «достоинства» разные, это, в частности, насыщенность информационного материала соответствующими теме ключевыми словами, но во всех них важно одно -‎ наличие соответствующей информации. Это первоочередное условие.

Дается оценка того, по каким тематическим запросам сайт способен дать тот самый «достойный ответ» -‎ то есть по каким направлениям сайт может продвинуться, а по каким попытка будет малоэффективной. Но также на данном этапе, если это необходимо, намечаются пути, которыми тематические возможности сайта можно будет привести в соответствие с желаниями.

2-‎й этап оптимизации – идеализация.

Вкратце, задача этого этапа – представить поисковым системам сайт как максимально соответствующий – релевантный – заявленной тематике, то есть соответствующим запросам.

3-‎й этап оптимизации – работа над ссылками.

Третий этап – это работа со ссылками.

На данный момент большинство поисковых систем оценивает релевантность страницы запросу, в самом упрощенном виде, как произведение авторитетности страницы и суммы релевантности текста страницы и релевантности текста ссылок запросу.

4-й этап – это оценка сайтов Ваших конкурентов (маркетинговый анализ сайтов). От популярности поискового запроса зависит, насколько трудно будет попасть на первые страницы результатов поиска. Мы анализируем конкурирующие сайты, в результате чего определяем стратегию дальнейших действий.

5-й этап – улучшение внешнего вида и содержание сайта, т.к. большая часть успеха раскрутки сайта зависит от дизайна и, конечно же, внутреннего содержания сайта. Наши специалисты отредактируют тексты страниц сайта, добавят необходимое количество ключевых слов и фраз, насытят их полной и актуальной информацией (seo-копирайтинг).

И самый сложный и важный этап это – Дальнейшее совершенствование и развитие сайта.

Постоянное развитие сайта – неотъемлемая часть работы оптимизатора, который работает на «повышение эффективности бизнеса клиента», а не на «позиции».

3.3.2 Этапы продвижения сайта в сети Интернет

При продвижении сайта в сети были проведены следующие этапы продвижения‎:

1. Разработка структуры сайта.

Структуру сайта нужно продумать сразу, и перед началом действий по продвижению убедиться, что названия страниц вас полностью устраивают. Изменение названия страниц в будущем может негативно сказаться на позиции сайта в поисковике. Если изменение адреса ссылки жизненно необходимо, стоит установить корректное перенаправление с помощью http-‎status-‎code 301.

1. Размещенная контактная информация.

На сайте должны быть указаны контактные данные, поскольку это повышает доверие посетителей сайта к компании. Кроме того, потенциальный клиент может просто захотеть связаться с вами, чтобы задать какие-‎либо вопрос.

1. Выбор доменного имени.

Чем выше уровень домена, тем предпочтительнее этот домен для поисковиков. Однако это не означает, что домены третьего уровня плохо индексируются, просто поисковики поставят хорошо оптимизированные сайты с собственными доменами в выдаче выше.

1. Составление семантического ядра.

Один из первых шагов внутренней оптимизации – это составление семантического ядра сайта. Семантическое ядро сайта (СЯС) – это список целевых запросов, вводимых пользователями в строку поиска Яндекса, Гугла, и других поисковых систем. Эти запросы, по сути, и определяют тематику сайта. Именно с создания СЯ начинается любая раскрутка сайта, ведь при его отсутствии продвижение в поисковых системах окажется просто неэффективным.

Все вводимые запросы пользователями в поисковые системы, можно условно разделить на:

* высокочастотники (ВЧ);
* среднечастотники (СЧ);
* низкочастотники (НЧ).

Запрос относят в ту или иную группу в зависимости от того, как часто пользователи набирают его в поисковых системах. Определенных границ, отделяющих ВЧ от СЧ, а СЧ от НЧ запросов, не существует. Они зависят от тематики, но мы будем считать низкочастотниками запросы, которые набирают до 500-700 раз в месяц; среднечастотниками – до 1-2 тысяч раз в месяц; высокочастотниками – свыше 2 тысяч раз в месяц.

Составление плана SEO-‎действий:

* определение целей продвижения сайта‎: целью может быть увеличение трафика, обгон конкурентов по тем или иным позициям, и так далее;
* составление и группировка ядра ключевых запросов‎: на данном этапе необходимо отчетливо представлять себе, кто является аудиторий и какие релевантные запросы для сайта пользователи могут вводить в поисковики;
* анализ конкурентов по каждому из запросов‎: это поможет не только увидеть позиции конкурентных компаний в поисковиках, но и посмотреть, по каким ключевых словам находятся те или иные фирмы, кто находится выше в поисковой выдаче, а кто ниже. Кроме того, анализ конкурентов скорее всего натолкнет вас на новые идеи, что приведет к корректировке и улучшению стратегии продвижения сайта;
* внутренняя оптимизация сайта‎: сюда входит выбор страниц под каждую группу ключевых слов, оптимизация содержимого сайта для продвижения и внутренняя перелинковка страниц;
* на всех страницах должен быть уникальный текст;
* необходимо добавить сайт в Яндекс.Вебмастер (http://webmaster.yandex.ru/sites/) и проверить, чтобы сайт был «привязан» к нужному региону;
* нужно обязательно измерять скорость загрузки сайта и его отображение на мобильных устройствах. Например, с помощью этих двух официальных сервиса от Гугла – скорость загрузки (https://developers.google.com/speed/pagespeed/insights/) и мобильности (https://www.google.com/webmasters/tools/mobile-friendly/). Они показывают, что и как нужно исправить. Оба показателя используются в алгоритмах ранжирования сайтов и влияют на место сайта в ТОП-10.

3.4 Оценка экономических затрат на проект

Определение затрат труда на разработку программного продукта.

Период проведения работ: c 03.10.2016 по 05.11.2016.

Работы проводились в одну смену продолжительностью 8 часов.

Рассчитаем общую трудоемкость работ.

Используем систему коэффициентов для отдельных этапов разработки.

Определение условного количества операторов программы, трудоемкости (формула 1):

|  |  |
| --- | --- |
| Т = То + Ти + Та + Тп + Тотл + Тд, | (1) |

где Т – общие затраты труда;

То – затраты труда на описание задачи;

Ти – затраты на исследование предметной области;

Та – затраты на разработку блок-схем;

Тп – затраты на программирование;

Тотл – затраты на отладку;

Тд – затраты на подготовку документации.

Все составляющие определяем через условное число операторов – Q (формула 2):

|  |  |
| --- | --- |
| Q = q \* c \* (1 + p), | (2) |

где q = 700 – число операторов.

Коэффициент сложности c характеризует относительную сложность программы по отношению к так называемой типовой задаче, реализующей стандартные методы решения, сложность которой принята равной единице (величина с лежит в пределах от 1,25 до 2). Для разработанного программного продукта возьмем 1,5.

Коэффициент коррекции p – увеличение объема работ за счет внесения изменений в алгоритм или сайт по результатам уточнения постановок. С учетом того, что в данном случае заказчик, средне представлял себе, что он хочет получить, что приводило к доработкам, возьмем коэффициент = 0.5.

В результате получим условное число операторов.

Q = q⋅c⋅(1 + p) = 700⋅1,5⋅(1+ 0,5) =1575.

Также используем следующие коэффициенты:

Коэффициент увеличения затрат труда, вследствие недостаточного описания задачи, в зависимости от сложности задачи принимается от 1,2 до 1,5, в связи с тем, что данная задача, потребовала уточнения и небольших доработок, примем B = 1,3.

Коэффициент квалификации разработчика k определяется в зависимости от стажа работы и составляет: для работающих до двух лет – 0,8; от двух до трех лет – 1,0; от трех до пяти лет – 1,1 – 1,2; от пяти до семи – 1,3 – 1,4; свыше семи лет – 1,5 – 1,6. Разработчик, которому было поручено это задание, имел опыт работы по специальности около 1 года, поэтому примем k = 0,8.

Рассчитаем общую трудоемкость.

Затраты труда на подготовку описания задачи Тo точно определить невозможно, так как это связано с творческим характером работы. Примем Тo = 21 чел.-ч.

Затраты труда на изучение описания задачи Ти с учетом уточнения описания и квалификации программиста могут быть определены по формуле 3:

|  |  |
| --- | --- |
| Ти = Q ⋅B / (75 ÷ 85) k, | (3) |

где Q – условное число операторов;

B – коэффициент увеличения затрат труда, вследствие недостаточного описания задачи.

Ти = 1575⋅1,3 /75⋅0,8 = 21,8 чел.-ч.

Затраты труда на разработку алгоритма решения задачи Тa рассчитывается по формуле 4:

|  |  |
| --- | --- |
| Тa = Q / (20 ÷ 25) k, | (4) |

Та = 1575/22⋅0,8 = 57,2 чел.-ч.

Затраты труда на составление сайта по готовой блок-схеме Тп определяется по формуле 5:

|  |  |
| --- | --- |
| Тп = Q / (20 ÷ 25) k, | (5) |

Тп = 1575/22⋅0,8 = 57,2 чел.-ч.

Затраты труда на отладку Тотл рассчитывается по следующей формуле 6:

|  |  |
| --- | --- |
| Тотл= Q / (4 ÷ 5) k, | (6) |

Тотл = 1575/5⋅0,8 = 252 чел.-ч.

Затраты труда на подготовку документации по задаче Тд определяются по формуле 7:

|  |  |
| --- | --- |
| Тд = Тдр + Тдо, | (7) |

где Tдр – затраты труда на подготовку материалов в рукописи.

Затраты труда на подготовку материалов в рукописи формула 8:

|  |  |
| --- | --- |
| Тдр = Q / (15 ÷ 20) k, | (8) |

Тдр = 1575/20⋅0,8 = 63 чел.-ч.

где Тдо – затраты труда на редактирование, печать и оформление документации (формула 9):

|  |  |
| --- | --- |
| Тдо = 0,75 Тдр, | (9) |

Тдо = 0,75⋅ 63= 47,25 чел.-ч.

Тд = 63+ 47,25 = 110,25 чел.-ч.

С учетом уровня языка программирования трудоемкость разработки программы может быть скорректирована следующим образом (формула 10):

|  |  |
| --- | --- |
| Ткор = Е⋅ kкор , | (10) |

где Ткор – коэффициент изменения трудоемкости, берётся из следующей таблицы 3.

Таблица 3 – Изменение трудоемкости в зависимости уровня языка программирования

|  |  |  |
| --- | --- | --- |
| Уровень языка  программирования | Характеристика языка  Программирования | Коэффициент изменения  трудоемкости |
| 1 | Покомандный автокод-Ассемблер | 1 |
| 2 | Макроассемблер | 0,95 |
| 3 | Алгоритмические языки Высокого уровня | 0,8 – 0,9 |
| 4 | Алгоритмические языки Сверхвысокого уровня | 0,7 – 0,8 |

Для разработки выбран язык PHP, который относится к алгоритмическим языкам высокого уровня, с учетом этого примем kкор = 0,8.

Подставив все полученные данные в формулу 11, получим полную трудоемкость разработки:

Т = 21+21,8+57,2 +57,2 +252 +110,25 = 519,45 чел.-ч.

С учетом корректировки из формулы 11 получим итоговую трудоемкость разработки:

Ткор = 519,45 \* 0,8 = 415,56 чел.-ч.

Определение численности исполнителей (формула 11):

|  |  |
| --- | --- |
| Ч = Т / Ф, | (11) |

где Ч – численность исполнителей;

Ф – действительный фонд времени специалиста в период разработки.

При Ф = 240 часов найдем численность исполнителей:

Ч = 415,56 / 240 = 1 – исполнитель.

В состав исполнителей входят: программист.

Распределение трудоемкости по стадиям разработки приведено в таблице 4.

Таблица 4 – Распределение трудоемкости по стадиям разработки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Этап разработки | Содержание работ | Трудоёмкость ч. | Трудоёмкость работ исполнителя, ч. | Должность исполнителя |
| Постановка задачи | 1.Предпроектное обследование.  2.Разработка ТЗ.  3.Разработка, согласование и утверждение технико-экономического обоснования. | 15,25 | 15,25 | программист |
| Технический проект | Уточнение структуры и формы представления входных и выходных данных. Разработка алгоритма решения задачи. Разработка структуры программы. Разработка пояснительной записки. Согласование технического проекта и его утверждение. | 76,45 | 76,45 | программист |
| Рабочий проект | Отладка программы. Разработка дизайна программы. | 88,8 | 88,8 | программист |
| Документация и внедрение | Разработка программной документации. Подготовка и передача программы и программной документации для сопровождения и изготовления, оформления и утверждения акта о передаче программы на сопровождение. Передача программного продукта заказчику. | 59,5 | 59,5 | программист |
| ИТОГО: |  | 240 | 240 |  |

Расчет затрат на разработку.

Основная заработная плата разработчика рассчитывается по формуле 12:

|  |  |
| --- | --- |
| Зпосн = О \* 1,15, | (12) |

где О – оклад;

1,15 – уральский коэффициент.

Оклад программиста равен: 18 000 рублей.

Его основная месячная заработная плата составит:

18 000 \* 1,15 = 20 700 рублей.

Основная заработная плата разработчика за весь период разработки программного продукта составит:

20 700 руб. \* 240 ч. / (8ч. \* 22 дня) = 28 227,27 руб.

Дополнительная заработная плата рассчитывается в процентах от основной заработной платы и составляет 12%.

Дополнительная заработная плата разработчика за весь период разработки программного продукта составит:

28 227,27 \* 0,12 = 3 387,2 руб.

Суммарная заработная плата разработчика за весь период разработки программного продукта составит:

28 227,27 руб. + 3 387,2 руб. = 31 614,47 руб.

Страховые взносы во внебюджетные фонды берутся в размере 30,2% от суммы основной и дополнительной заработной платы.

Суммарные страховые взносы во внебюджетные фонды за весь период разработки программного продукта составят:

31 614,47 \* 0,302 = 9 547 руб.

Содержание и эксплуатация вычислительного комплекса считаются следующим образом (формула 13):

|  |  |
| --- | --- |
| Свт = См-ч \* Число\_часов\_отладки, | (13) |

где См-ч – стоимость машино-часа.

Число часов отладки составляет:

Тп + Тотл = (57,2ч.+ 252ч.) \* 0,7= 216,44ч.

Стоимость машино-часа рассчитывается, как сумма составляющих:

|  |  |
| --- | --- |
| (Ст-ть\_эл\_эн\_в\_год+Аморт\_в\_год+Затраты\_на\_ремонт\_за\_год)/Фвт, | (14) |

где Фвт – действительный фонд времени работы вычислительного комплекса.

Стоимость 1 КВТ/час электроэнергии составляет, например, 2,74 руб.

Один компьютер потребляет, например, 250 ВТ в час.

За месяц расходы на электроэнергию, потребляемую одной ЭВМ составляет:

8 ч. \* 22 дня \* 0,25 КВТ/ч. \* 2,74 руб. \* ч. / КВТ = 120,56 руб.

Действительный фонд времени работы вычислительного комплекса рассчитываем по следующей формуле 15:

|  |  |
| --- | --- |
| Фвт = Фном – Фпроф, | (15) |

где Фном – номинальный фонд времени работы вычислительного комплекса;

Фпроф – годовые затраты времени на профилактические работы (принимаются 15% от Фном).

Фпроф = 240 час. \* 0,15 = 36 час.

Фвт = 240 час. – 36 час. = 204 часа.

Стоимость машино-часа составляет:

(120,56 руб. + 1250 руб. + 400 руб.) / 204 часа = 8,68 руб.

Содержание и эксплуатация вычислительного комплекса составляет:

240ч. \* 8,68 руб. = 2083,2 руб.

Накладные расходы рассчитываются, как 60% от расходов на содержание и эксплуатацию вычислительного комплекса:

2083,2 руб.\*0,6 = 1249,92 руб.

Смета затрат на разработку программного продукта приведена в таблице 5.

Таблица 5 – Смета затрат на разработку программного продукта

|  |  |  |
| --- | --- | --- |
| № | Наименование статьи расходов | Затраты, руб. |
| 1 | Основная заработная плата производственного персонала. | 31 614,47 |
| 2 | Дополнительная заработная плата производственного персонала. | 3 387,2 |
| 3 | Страховые взносы во внебюджетные фонды | 9 547 |
| 4 | Содержание и эксплуатация вычислительного комплекса. | 2083,2 |
| 5 | Накладные расходы. | 1249,92 |
| ИТОГО: | | 47 881,79 |

В ходе экономических расчетов были получены следующие результаты:

* определена трудоемкость разработки программного продукта, которая составила 415,56 чел.-час;
* рассчитано, что для выполнения данного сайта в планируемый период потребуется один человек: программист;
* рассчитана смета затрат на разработку программного продукта, итоговая сумма которой 47 881,79 руб.

ВЫВОД ПО ГЛАВЕ 3

В третьей главе работы описаны этапы работы над проектом. В ходе проектирования сайта были спроектированы диаграмма вариантов использования, определяющая будущий функционал системы.

Сайт написан на системе управления контентом CMS **Wordpress**, позволяющей создавать сайты любой сложности. Данная систем управления написана на языке PHP, в качестве хранилища содержания использует базу данных MySQL.

Система удобна в управлении, содержит различные инструменты для разработки web-сайта, которые дополняются по мере необходимости. За счёт чего, снижается загромождение административной панели, нагрузку на сервер и место на хостинге. Каталог расширений содержит множество языковых пакетов, благодаря чему, программа становится более доступной.

Произведена оценка экономической эффективности проекта, из которой следует, что разработка и внедрение разрабатываемого сайта являются эффективным.

ЗАКЛЮЧЕНИЕ

Результатом научной работы стал сайт для социального проекта «Живая история».

В ходе выполнения научной работы решены следующие задачи:

* проведен анализ предметной области;
* сформированы необходимые требования к проектируемой информационной системе;
* проведен анализ известных решений в похожей области информационных технологий;
* проанализированы и обоснованы инструменты для возможности реализации сайта, с учетом текущих технических возможностей;
* проведен анализ предметной области и осуществлено проектирование системы технического и SEO аудита сайта;
* сформировано техническое задание в соответствии с ГОСТ 34.602-89;
* разработан и внедрен сайт;
* проведена экспертная оценка качества сайта в соответствии с ГОСТ 28195-89;
* проведена оценка экономической эффективности разрабатываемого сайта.

Благодаря анализу всевозможного инструментария, относящегося к разработке как информационных систем в целом так просто программных продуктов или даже банальных скриптов, были смоделированы и классифицированы, а в дальнейшем и обоснованы, возможные инструменты для разработки разных частей информационной системы. Что придало системе невероятную гибкость в модернизации и кроссплатформенности.

При разработке непосредственно самой информационной системы возникало немало трудностей, в основном из-за малого опыта работы с определенным языком программирования или недостаточных теоретических знаний, что явно увеличивало период разработки.

В целом запланированный проект полностью реализован в соответствии с требованиями и техническим заданием к проекту. Во время создания сайта возникала масса идей по улучшению функционала и эффективности работы сайта как в области архитектуры системы, так и реализации интерфейса. В дальнейшем планируется реализация данных идей.

СПИСОК ЛИТЕРАТУРЫ

Нормативно-правовые акты

1. Федеральный закон РФ 152-ФЗ «О персональных данных».
2. ГОСТ 19.201-1978. Техническое задание. Требования к содержанию и оформлению.; Введ. 01.01.1990. — М. : Издательство стандартов, 1989. — 13 с.
3. ГОСТ 34.602-89. Техническое задание на создание автоматизированной системы: Гос. Стандарт. – Введ. 01.01.1990 // утверждён приказом роспрома 16.09.2004 №95.
4. Закон РФ N 5485-I «О государственной тайне».
5. Федеральный закон N 149-ФЗ «Об информации, информационных технологиях и о защите информации»

Основная литература

1. HTML5, CSS3 и JavaScript. Исчерпывающее руководство / Джен­ нифер Роббинс; [пер. с англ. М. А. Райтман]. — 4-е издание. — М. : Эксмо, 2014. — 528 с.
2. HTML5. Разработка приложений для мобильных устройств. / Вейл Э. — СПб.: Питер, 2015. — 480 с.: ил.
3. Нестеров, С.А. Основы информационной безопасности : учебное пособие / С.А. Нестеров ; Министерство образования и науки Российской Федерации, Санкт-Петербургский государственный политехнический университет. – СПб : Издательство Политехнического университета, 2014. – 322 с.
4. MySQL разработка Web-приложений. – 4-е изд., перераб. И доп. — МПб.: БХВ-Петербург, 2013. — 560 с. : Ил.
5. PHP и MySQL. Исчерпывающее руководство. 2-е изд. / Маклафлин Б. — СПб.: Питер, 2014. — 544 с.: ил.
6. Маркин, А.В. Основы web-программирования на PHP : учебное пособие [Электронный ресурс] / А.В. Маркин, С.С. Шкарин. – М. : Диалог-МИФИ, 2012. – 252 c.
7. Управление данными: Учебное пособие (часть 1) / Ю.Р. Мухина.– Челябинск: Полиграф-Мастер, 2015. – 135 с.
8. Дженнифер Роббинс [пер. с англ. М. А. Райтман]. HTML5, CSS3 и JavaScript: Исчерпывающее руководство. – М.: Эксмо, 2014. — 528 с.
9. Моделирование на UML. / Ф. Новиков, Д. Иванов – М.: ДМК, 2013. – 281с.
10. Моделирование систем управления : практикум / Д. О. Глухов, И. В. Петухов; под ред. Д. О. Глухова – Йошкар-Ола : Паволжский государственный технологический университет, 2015. – 84 с. ; То жэ [Электронный ресурс]. – URL: http://biblioclub.ru/index.php?page=book\_view\_red&book\_id=437061 (15.09.2016).
11. Моделирование бизнес-процессов: учеб. Пособие / В. Г. Мамонова, Н.Д. Ганелина. – Новосибирск: Изд-во НГТУ, 2012. – 43с. ; То жэ [Электронный ресурс]. – URL: http://biblioclub.ru/index.php?page=book\_view\_red&book\_id=228975 (25.09.2016).
12. Моделирование систем. Объектно-ориентированный подход. Учебное пособие / Ю. Б. Колесов, Ю. Б. Сениченков. — СПб.: БХВ-Петербург, 2012. — 192 с.: ил.
13. Моделирование систем и процессов : учебник для академического бакалавриата / В. Н. Волкова, Г. В. Горелова, В. Н. Козлов [и др.] ; под ред. В. Н. Волковой, В. Н. Коз- лова. — М. : Издательство Юрайт, 2015. — 449 с.
14. Управление процессами : учебное-практическое пособие / М. В. Самсонова. – Ульяновск : УлГТУ, 2014. -187 с. ; То жэ [Электронный ресурс]. – URL: http://biblioclub.ru/index?page=book\_view\_red&book\_id=363491 (25.10.2016).
15. Методология и инструментарий моделирования бизнес-процессов: учебное пособие / Цуканова О. А. – СПб.: Университет ИТМО, 2015. – 100 с.
16. Применение UML 2.0 и шаблонов проектирования / Кпег Ларман 3-е изд. — М.: Вильямс, 2013. — 736 с.
17. Разработка и управление требеваниями. Практическое руководство пользователя/Элизабет Халл, Кен Джексон, Джемир Дик. — СПб.: БХВ-Петербург, 2012. — 229 с.: ил.
18. Экономическая оценка новационных технических решений: монография: в 2 ч. / Э.М. Гайнутдинов, Л.И. Поддерегина. – Минск: БНТУ, 2013. – Ч. 1. – 332 с.
19. Язык UML: руководство пользователя. / Г. Буч, Д. Рамбо, А.Джекобсон – М.: ДМК, 2012. – 257с.

Дополнительная литература

1. Гущин, А.Н. Базы данных : учебно-методическое пособие / А.Н. Гущин. – 2-е изд., испр. И доп. – М. ; Берлин : Директ-Медиа, 2015. – 311 с. : ил. – Библиогр.: с. 226-228. – ISBN 978-5-4475-3838-5 ; То же [Электронный ресурс]. – URL: http://biblioclub.ru/index.php?page=book&id=278093 (13.11.2016).
2. PHP и Jquery для профиссионалов. : Пер. с англ. – М. : ООО “И.Д. Вильямс”, 2011. – 352 с. : ил.
3. Бенкен, Е. PHP, MySQL, XML: программирование для Интернета (+CD). / Е. Бенкен, Е. Кондукова. – М.: BHV, 2011 г. – 304 c.
4. Бизнес-процессы. – Электронный ресурс. – Режим доступа: http://www.betec.ru (15.11.2016).
5. Битрикс Разработчикам – Документация по CMS «1С-Битрикс: Управление сайтом» // Документации на русском языке 1c-bitri. – Электронный ресурс. – Режим доступа: http://www.1c-bitrix.ru (15.11.2016).
6. Документации на русском языке WordPress // Русский. – Электронный ресурс. – Режим доступа: https://ru.wordpress.org (15.11.2016).
7. Веллинг, Л. Разработка Web-приложений с помощью PHP и MySQL / Л. Веллинг, Л. Томсон. – 3-е изд. – М. : Вильямс, 2009. – 875 с.
8. Модели бизнес-процессов предприятния // Business Studio официальный сайт. – Электронный ресурс. – Режим доступа: http://www.businessstudio.ru. – (15.11.2016).
9. Официальная документация к системе NetCat 5 для разработчика // CMS NetCat – система управления сайтами. – Электронный ресурс. – Режим доступа: http://netcat.ru/developers/docs/ (15.11.2016).
10. Проектирование и исследование бизнес-процессов : [электронный ресурс] учеб. Пособие / А.С. Козлов. – 4-е изд , стер. – М : Флинта, 2011 – 272с. – http://biblioclub.ru/index.php?page=book\_view\_red&book\_id=103541 (15.11.2016).
11. Сайт денвер. – Электронный ресурс.– http://www.denwer.ru/ . (15.11.2016).

ПРИЛОЖЕНИЕ А

Таблица А.1 – Термины и определения

|  |  |
| --- | --- |
| Термин | Описание |
| IP-‎адрес | Адрес узла в сети, построенной по протоколу IP. |
| RSS | Формат данных, предназначенный для описания лент новостей, анонсов статей, изменений в блогах и т. П. |
| Футер сайта | это нижняя часть сайта со всем содержимым. Чаще всего он используется веб-мастером для распределения такой информации как защита авторских прав, название, дата создания и т. П. |
| Административная часть сайта | Закрытая от посетителей часть сайта, предназначенная для управления сайтом. Управление осуществляется администратором (‎техническая поддержка‎) и редактором сайта (‎информационная поддержка‎). |
| Администратор сайта | Специалист, осуществляющий техническую іддержку сайта. |
| Адаптивный сайт | это сайт, который на разных устройствах отображается по-разному, исходя их особенностей платформы устройства и диагонали экрана. |
| Блог | Сайт или раздел сайта, основное содержимое, которого — регулярно добавляемые записи, содержащие текст, изображения или мультимедиа. |
| SEO | Это комплекс определенных действий по оптимизации внутренних (HTML-код, структура, содержание) и внешних (количество ссылок) параметров, направленный на улучшение позиций сайта в выдаче по конкретным поисковым запросам. |

Продолжение таблицы А.1

|  |  |
| --- | --- |
| Оптимизация сайта | Комплекс действий, направленных на структуризацию информации на сайте. |
| Веб-‎браузер (‎браузер‎) | Клиентская программа, поставляемая третьими сторонами и позволяющая просматривать содержимое веб-‎страниц. |
| Веб-‎интерфейс | Совокупность экранов и элементов управления системы, позволяющих пользователю, осуществляющему доступ к системе через веб-‎браузер, осуществлять поддержку и управление системой. |
| Веб-‎сайт (‎сайт‎) | Совокупность взаимосвязанных статических и динамических страниц, содержащих текстовые данные, рисунки, видео и другую цифровую информацию, объединенная под одним адресом (‎доменным именем или IP-‎адресом‎) в сети Интернет. |
| Гиперссылка (‎ссылка, линк‎) | Активный фрагмент текста или изображения, позволяющий загрузить другую страницу или выполнить определенное действие. |
| Дизайн веб-‎сайта | Уникальные для конкретного веб-‎сайта структура, графическое оформление и способы представления информации. |
| Дизайн-‎шаблон страниц | Файл, содержащий элементы внешнего оформления страниц сайта, а также набор специальных тегов, используемых системой публикации сайта для вывода информации при создании окончательных страниц. |
| Динамическая страница | Страница сайта, которая при помощи программного кода обрабатывает и выводит информацию из базы данных. |

Продолжение таблицы А.1

|  |  |
| --- | --- |
| Доменное имя | Символьное имя иерархического пространства сети Интернет. Полное имя домена состоит из имён всех доменов, в которые домен входит, разделённых точками. |
| Мета-‎тэг | Элемент разметки веб-‎страницы. Используется для описания страницы, ключевых слов и других данных. |
| Модератор, редактор, (‎контент-‎менеджер‎) сайта | Специалист, осуществляющий информационную поддержку сайта. |
| Наполнение (‎контент‎) | Совокупность информационного наполнения веб-‎сайта. Включает тексты, изображения, файлы и т.п. предназначенные для пользователей сайта. |
| Общедоступная часть сайта | Основная информационная часть сайта, открытая для посетителей. |
| Резервная копия | Совокупность объектов базы данных, представленная в виде файлов, позволяющая восстановить точную копию структуры исходной базы данных в аналогичной системе управления базами данных. |
| Система управления сайтом (‎CMS‎) | Информационная система, позволяющая авторизованным пользователям производить изменения иерархической структуры и информационного наполнения веб-‎сайта без использования каких-‎либо дополнительных специальных программных средств. |
| Статическая страница | Страница сайта, которая не обращается к базе данных; вся информация хранится в коде страницы. |
| Хостинг | Услуга по предоставлению дискового пространства для физического размещения информации на сервере, постоянно находящемся в сети Интернет. |

Продолжение таблицы А.1

|  |  |
| --- | --- |
| Элемент наполнения (‎контента‎) | Отдельная запись в базе данных, внешнее представление которой зависит от управляющего ей программного модуля (‎например, в модуле «новостная лента» элементом наполнения является отдельная новость‎). |