СОДЕРЖАНИЕ

ВВЕДЕНИЕ 3

1 ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА СОЗДАНИЯ ВЕБ-САЙТОВ 5

1.1 Процесс разработки сайтов 5

1.2 Визуальные редакторы создания веб-сайтов и веб-сервисов 6

1.3 Конструкторы сайтов 8

1.4 Языки разметки гипертекста HTML и XML 10

1.5 Скриптовые языки программирования 14

ЗАКЛЮЧЕНИЕ 16

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 17

ВВЕДЕНИЕ

Создание веб-сайта состоит из нескольких этапов и взаимосвязанных процессов — таких, как проектирование сайта, создание макетов его страниц, написания наполнения и помещения его на сайт, обслуживание сайта и его программной основы.

Одним из этапов разработки веб-сайта является веб-дизайн, который, в узком смысле термина, означает визуальное оформление веб-страниц. В какой-то степени — это аналогия верстки газеты или журнала, или создание изображения в полиграфическом дизайне. В то же время веб-дизайн может включать в себя и проектирование структуры сайта, его навигации и в некоторых случаях даже движков ресурса. Другими словами, продукт веб-дизайна должен включать в себя не только визуальные аспекты сайта, но и его юзабилити — то есть удобство при использовании.

Пять областей охватывают основные аспекты Web.

* Содержимое. Сюда входят форма и организация содержимого сайта. Возможный диапазон – от того, как написан текст до того, как он организован, представлен и структурирован с помощью технологии разметки, такой как HTML;
* Зрительные образы. Это относится к компоновке экранного пространства на сайте. Эта компоновка обычно создается с помощью HTML, CSS или даже Flash и может включать графические элементы, выполняющие функции украшения или навигации. Визуальная сторона сайта — это наиболее очевидный аспект Web-дизайна, но не единственная, и не самая важная, сторона дисциплины;
* Технология. Хотя применение разнообразных базовых Web-технологий вроде HTML или CSS попадает в эту категорию, под технологией в этом контексте чаще подразумеваются различные интерактивные элементы сайта, в особенности, созданные с использованием программных методов. Это могут быть элементы в диапазоне от языков сценариев, работающих на стороне клиента, наподобие JavaScript, до серверных приложений, таких как Java-сервлеты, PHP-сценарии;
* Доставка. Скорость и безотказность доставки сайта по сети Internet или внутренней корпоративной сети связаны с применяемым аппаратным программным обеспечением и задействованной сетевой архитектурой;
* Назначение. Причина, по которой сайт существует, часто связанная с экономическими вопросами, вероятно, является наиболее важной частью Web-дизайна. Этот элемент следует учитывать при принятии любых решений, затрагивающих другие области. Конечно, степень, в которой каждая сторона Web-дизайна оказывает воздействие на сайт, может изменяться в зависимости от типа создаваемого сайта. Личная домашняя страничка обычно не связана с экономическими соображениями, характерными для Internet-магазина. Внутренняя сеть производственной компании может не попадать под влияние соображений, связанных с визуальным представлением, важных для общедоступного Web-сайта, рекламирующего остросюжетный фильм.

1. ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА СОЗДАНИЯ ВЕБ-САЙТОВ

В настоящее время большинству сайтов свойственна динамичность и интерактивность. Для реализации этих параметров используются веб-приложения. Это готовые программные комплексы для решения задач сайта. Веб-приложение входит в состав сайта, но само по себе без "содержимого" сайтом является только технически - это оболочка или шаблон, который необходимо наполнить и активизировать. Как раз этим занимаются специалисты по продвижению и раскрутке сайтов.

Как правило, одному сайту соответствует одно доменное имя. Именно по нему любой сайт идентифицируются в глобальной сети. Впрочем, это не единственный возможный вариант. Один сайт может размещаться на нескольких доменах, а также несколько сайтов могут существовать под одним доменом.

За последние десятилетия всемирная паутина стала отличной информационно-рекламной платформой и потому компании различного масштаба (от крупных транснациональных и мировых корпораций до частных предпринимателей), в том числе не связанные непосредственно с деятельностью в сети, создают собственные сайты следующих типов:

* Сайт-визитка. На таких сайтах размещаются самые общие данные о владельце сайта. Предоставляется информация о виде деятельности, истории бизнеса, информация о сотрудниках, прайс-лист, контактные данные, реквизиты, схема проезда;
* Представительский сайт. От описанной выше "визитке" отличается расширенной функциональностью: подробное описание услуг, портфолио, отзывы, форма обратной связи и так далее;
* Корпоративный сайт. Содержит полную информацию о компании-владельце, услугах/продукции, событиях в жизни компании. Отличается от предыдущих двух типов сайтов полнотой представленной информации, зачастую содержит различные функциональные инструменты для работы с контентом (поиск и фильтры, календари событий, фотогалереи, корпоративные блоги, форумы). Может содержать закрытые разделы для тех или иных групп пользователей – сотрудников, дилеров, контрагентов и прочих.
  1. Процесс разработки сайтов

Процесс разработки сайтов проходит в несколько этапов:

*Первый этап*. Разработка макетов шаблонов веб-страниц. Этим занимаются веб-дизайнеры, в задачи которых входит: определить, каким образом конечный потребитель будет получать доступ к информации и услугам сайта. Другими словами – разработать пользовательский интерфейс.

Готовые шаблоны предоставляются заказчику на одобрение. Чтобы макеты выглядели более наглядно, в них помещается произвольное содержимое. Если заказчик удовлетворён внешним видом шаблонов, то наступает следующая фаза разработки – вёрстка страниц сайта.

*Этап номер 2*. Верстальщик получает макеты шаблонов в виде простых изображений (например, в формате JPEG или PNG), либо разбитых по слоям (например, в PSD или AI). Его основная задача – получить из этих графических макетов гипертекстовые веб-страницы с подготовленными для интернета изображениями.

Самое сложное на этом этапе – обеспечить совместимость со множеством браузеров, так как в некоторых из них одни и те же элементы разметки могут отображаться не так, как задумано. Когда достигнуто правильное отображение в большинстве браузеров, переходят к завершающему этапу.

*Этап номер 3*. Веб-программирование. Программисту передаются готовые шаблоны страниц, а также указания дизайнеров по работе и организации элементов сайта. С нуля создаётся программная основа сайта. Выбор языка программирования в данном случае – вопрос непринципиальный. После того, как сайт готов к эксплуатации, остаётся наполнить его задуманной информацией.

Существует множество инструментов, с помощью которых веб-дизайнер осуществляет верстку страниц. Такие программы называются HTML-редакторами. В веб-дизайне используется два типа редакторов – визуальные и невизуальные (текстовые).

Первые работают по принципу WYSIWYG (от англ. – What You See Is What You Get – что видишь, то и получаешь. Другими словами, это способ подразумевает, что при редактировании материал выглядит так же, как он и будет выглядеть в конечном результате.

Текстовыми же редакторами, в основном, пользуются профессиональные веб-дизайнеры – так как такие инструменты подразумевают написание кода самостоятельно. С помощью текстовых редакторов создается чистый программный код, который позволяет веб-дизайнерам полностью воспроизводить задуманное, без вмешательства инструмента “по умолчанию”. Существуют также редакторы, которые поддерживают оба принципа работы.

Наиболее известными визуальными редакторами являются следующие программы, некоторые из которых будут рассмотрены ниже.

* 1. Визуальные редакторы создания веб-сайтов и веб-сервисов

*Macromedia Dreamweaver MX*– профессиональный инструмент для создания Web-сайтов и приложений. Пожалуй, это самая лучшая программа из визуальных редакторов на данный момент, по крайней мере она весьма любима многими пользователями, а раз любят, значит есть за что.

Разработчики утверждают, что Macromedia Dreamweaver MX предназначена для проектирования, разработки и администрирования профессиональных Web-сайтов и приложений. Кроме того, Dreamweaver легко интегрируется с другими программами от Macromedia, например, такими, как Flash. То есть Dreamweaver гораздо больше, чем просто визуальный редактор, это достаточно мощный и сложный инструмент, а всякий сложный инструмент требует, чтобы на его освоение было потрачено какое-то время, прежде, чем пользователь сможет работать в нем. Однако, я считаю, что вас не должно смущать, что придется изучать, как пользоваться программой, Dreamweaver стоит того.

*Adobe GoLive и LiveMotion.* Возможно, Adobe GoLive понравится тем, кто любит программы от Adobe, и много с ними работал: знакомая среда, достаточно легко разобраться, что к чему. Кроме того, еще один плюс для любителей Adobe - все программы от Adobe прекрасно взаимодействуют друг с другом и дополняют друг друга, GoLive не исключение, он прекрасно дружит с собратьями. Однако, GoLive не более, чем визуальный редактор для верстки веб-страниц, больше чем поддержки таких технологий, как Html, Dhtml, CSS, XML и нескольких готовых Javascript`ов не стоит ждать от этой программы. Однако, следует отметить, что встроенный редактор кода (текстовой) в этой программе очень хороший. То есть возможно, GoLive оптимальное сочетание визуального и текстового редактора (два в одном).

*Microsoft FrontPage –* считается простейшим визуальным редактором и популярным среди тех, кто отдает предпочтение программным инструментам Microsoft Office и создает сайт специально для сервера на основе технологий Microsoft. **FrontPage**поддерживает HTML, JavaScript (скриптовый язык), CSS (каскадные таблицы стилей), DHTML (сочетание HTML, JavaScript, CSS и DOM – объектной модели документа), совместим с технологиями ASP (создание веб-приложений), XML (язык разметки, текстовый формат, применяемый для хранения структурированных данных), VBScript (скриптовый язык программирования), XSL (расширяемый язык таблиц стилей – смена CSS). FrontPage также имеет неплохие возможности по управлению изображениями, flash-роликами.

По сути, **FrontPage является чуть ли не типичным приложением Microsoft Office,** и интеграция редактора с этой серией является его очевидным преимуществом. Любой пользователь, имеющий опыт работы с такими приложениями как Word, Excel, Access, не будет испытывать особых сложностей в работе с редактором FrontPage – знакомые панели инструментов, наборы мастеров и редакторов, шаблонов, автоматических функций и так далее. Другими словами, FrontPage – прекрасное решение для начинающих пользователей и веб-мастеров.

***Microsoft Expression Web и SharePoint Designer*** отличается тем, что предоставляет своим пользователям больше свободы, производит чистый, совместимый со стандартами, код, работает с CSS и CSS-макетами. По сути,**это не приемник FrontPage**, он, скорее, представляет собой конструктор и имеет большее сходство с SharePoint Designer, делая акцент на разработке и адаптации к сайтам, выполненных с помощью последнего. Приемником является SharePoint - хотя и не предлагает ничего революционного и также вызывает недовольство среди веб-мастеров. Среди возможностей – поддержка интерактивных страниц ASP. NET, создание вида данных из RSS, XML, Office XML, сотрудничество со встроенным Workflow Designer, инструменты CSS, отслеживание статистики страниц, проверка орфографии,**интеграция с SharePoint Server 2007**.

*Hotdog*– программа имеет простой и понятный пользователю интерфейс. Кроме того, что Hotdog дружит с пользователем, эта программа дружит также с Flash, SQL, PHP, ASP, имеет работать с GIF-изображениями (оптимизация, анимация), включает в себя Html-компрессор, может создавать файлы справки (CHM).

*Homesite* – этот редактор, пожалуй, самый популярный и мощный среди текстовых. Кроме того, что в нем достаточно легко работать не только с Html-кодом (есть все, от списка всевозможных атрибутов ко всем тегам вплоть до проверки кода (правильность проверяется с точки зрения W3C.org)), но также есть поддержка XHTML, CSS-редактор, и так далее.

*HTML Pad –* эта программа тоже пользуется большой любовью пользователей. Помимо всего стандартного HTMLPad поддерживает JavaScript, VBScript, SSI, ASP и Perl, умеет создавать макросы (наподобие Word и Excel ‘я), включает в себя много различных справочных материалов по CSS и Html, и многое другое.

*Notepad –*это Блокнот. В этой программе нет никаких функций, которые облегчат создание сайта: ни подсветки кода, ни вставки готовых конструкция кода, ничего, но зато эта программа есть на любом компьютере у каждого пользователя в папке Стандартные. С нее вы можете начать свои первые шаги в написании кода, а затем уже сменить на более понравившийся редактор.

* 1. Конструкторы сайтов

Конструктор сайтов – это система из набора инструментов, которая позволяет создавать сайты онлайн и администрировать их без каких-либо специализированных знаний. С её помощью можно выбрать тип будущего сайта (визитка, магазин и т. д.), готовый шаблон дизайна, цветовое оформление и модули, которые будут на нём отображаться.

Зачастую они предоставляют достаточно возможностей для создания сайтов, которые по качеству могут превзойти продукты небольших веб-студий, выполняющих заказы для малого и среднего бизнеса.

В сети существует множество сервисов, помогающих создать веб – приложение как новичку, так и человеку далекому от программирования, вёрстки сайтов и графических редакторов.

*uCoz –* это бесплатная система управления сайтом и хостинг для сайтов, созданных с её использованием.

Модули uCoz могут использоваться как в единой связке для создания полнофункционального сайта, так и по отдельности, например, в качестве блог-платформы, веб-форума и др. На данный момент в системе создано более миллиона сайтов.

Модули – веб-приложения, скомбинировав которые можно получить сайт любой сложности: от простой домашней странички с гостевой книгой до большого портала.

Taba.ru – онлайн-конструктор сайтов позволяет быстро создать многофункциональный сайт, предоставляя доменное имя 2-го или 3-го уровня и хостинг от 1 Гб.

Конструктор сайтов обладает удобным и понятным интерфейсом. В нем легко сможет разобраться даже неопытный пользователь. Готовые функциональные модули добавляются, перемещаются и настраиваются одним кликом мыши. Он позволяет посмотреть данные о новых пользователях, принять или отклонить их регистрацию.

*Nethouse* – конструктор сайтов, который позволяет любому человеку создать качественный и функциональный сайт для бизнеса буквально за полчаса. Чтобы получить готовый сайт, достаточно зарегистрироваться и заполнить всю необходимую информацию – Контакты, Услуги и цены, О себе. Nethouse предлагает пользователю выбрать один из множества красивых шаблонов и дизайнов, а также по желанию изменить фон.

*APS* – cтандарт пакетов автоматической установки для модели SaaS.

APS-формат разработан, чтобы упростить реализацию бизнес-модели Software-as-a-Service (SaaS) для облачных сервис-провайдеров и разработчиков ПО. APS-пакет – упакованное в соответствии с APS-стандартом веб-приложение.

На данный момент, APS-стандарт поддерживается хостинг-панелями Parallels: Plesk, Small Business Panel, POA и SysCP.

Всего в APS-каталоге опубликованы более 200 наименований CMS и различных веб-приложений (как коммерческих, так и open source).

*Wix* – имеет огромное количество качественных и разнообразных шаблонов (более 1000). Наряду с HTML5 есть возможность создания сайтов на flash и поддерживается функции Drag & Drop. Сайты на Wix хорошо индексируются поисковыми системами.

Большинство конструкторов сайтов бесплатны (расширение функционала потребует перехода на платный тариф), имеют более-менее удобный интерфейс, неплохой набор шаблонов, акции и т. д. Конкуренция велика, поэтому каждый ресурс стремится привлечь как можно большее количество клиентов всеми доступными средствами. Лучшие конструкторы сайтов дают возможность на бесплатном пакете сделать свой сайт и продвигать его в Интернете.

* 1. Языки разметки гипертекста HTML и XML

Любой Internet-сайт состоит из одного или нескольких связанных между собой HTML-документов и сопутствующих файлов (например, файлы картинок). HTML (Hypertext Markup Language) – язык разметки гипертекста.

XML используется для разметки стандартных документов во многом так же, как HTML. Однако XML ориентирован на работу со структурированными данными, такими, как результаты запроса, метаинформация об узле Web или элементы и типы схем документов.

Все файлы, используемые сайтом, должны иметь имена только латинскими буквами, цифрами и знаком подчеркивания (\_).

Первым файлом сайта, который автоматически загружается, является файл с именем index или default (лучше всего использовать index).

Каждый HTML-документ составляется в соответствии с определенными правилами и должен иметь расширение .htm или .html

Программа, которая в частности расшифровывает HTML-документ, называется браузером.

Любой браузер реализует следующие функции: осуществляет алгоритм обмена файлами с сайта, расшифровывает и отображает содержимое HTML-документа, обеспечивает выполнение программ на языках скриптов JavaScript или VBScript. Для реализации и исполнения всех этих функций не нужны никакие другие программные продукты.

Таким образом, алгоритм работы браузера при обслуживании сайта заключается в следующем. Вы запускаете сайт по сети или головной файл сайта из локальной папки компьютера. Браузер при этом получает полный URL-адрес сайта, т.е. попросту папку, где искать файлы сайта, и отображает содержимое головного файла. Если в процессе работы с файлом требуется новый файл картинки или HTML-документа, то браузер делает запрос на сервер, откуда и получает требуемый файл. С этим файлом он делает все, что предусмотрено сценарием.

Таким образом, сеть реально задействуется только в момент получения браузером нужного файла. Если по сценарию происходит частое переключение между HTML-файлами, то это замедляет работу сайта и задействует ресурсы сети (так называемый трафик).

Для устранения этого недостатка используется динамический HTML (DHTML), реализуемый программами на языках скриптов. Программы скриптов могут изменять сценарий, модифицировать окно браузера и все это делается в пределах того HTML-документа, с которым в данный момент работает браузер.

Единственное, чего не может браузер даже с использование скриптов – это каким-либо образом сохранять введенную в процессе сценария информацию.

Структура HTML-документа, основные элементы

HTML-документ оформляется по определенным правилам и всегда содержит в себе элементы языка HTML. Элементы могут быть парными (контейнеры) и непарными (дескрипторы).

Контейнер имеет следующую структуру:

<имя-элемента> любой текст</имя-элемента>

Как видно, контейнер открывается по <имя-элемента> и закрывается по </имя-элемента>. Действие контейнера распространяется на весь текст, который этот контейнер окаймляет (в данном примере это любой текст).

Существуют парные элементы, которые допускается не закрывать, так как они автоматически закрываются по контексту HTML-документа.

Дескриптор имеет следующую структуру:

<имя-элемента>

Как видно, дескриптор не закрывается по </имя-элемента>, а смысл его использования не связан с сопутствующим текстом документа, а заложен в названии самого элемента.

При открытии любого элемента по <имя-элемента> можно указывать допустимые атрибуты этого элемента. Атрибуты описывают свойства элемента, записываются в любой последовательности и отделяются друг от друга пробелом. Допустимые значения атрибутов указываются через знак равно (=) за именем атрибута и должны заключаться в кавычки («значение») за исключением зарезервированных слов этого атрибута (не будет ошибки, если и зарезервированное слово будет заключено в кавычки). Значения атрибута необходимо набирать с соблюдением больших и малых букв. Перечень допустимых атрибутов всегда приводится вместе с описанием конкретного элемента. Примером использования парного элемента может служить, например, задание для фрагмента текста цвета и размера символов:

<Font color=red size=4>фрагмент текста</Font>

Текст HTML-документа набирается в любом текстовом редакторе, например, Блокнот (но ни в коем случае не в редакторе Word). Имена элементов, их атрибуты и значения можно набирать большими или малыми буквами. Принято элементы набирать большими буквами.

Для правильного отображения в окне браузера HTML-документ набирается по определенным правилам.

Структура HTML-документа имеет следующий вид:

<HTML>

<HEAD>

<TITLE> название документа </TITLE>

<МЕТА...>

</HEAD>

<BODY>

Тело документа

</BODY>

</HTML>

Элементы <BODY> и <HTML> можно не закрывать.

Как видно HTML-документ состоит из двух частей: заголовок (HEAD) и тело (BODY). В каждой части документа используются свои элементы.

Браузером отображается только часть документа, заключенная между <Body> и </Body>

Элемент <TITLE> предназначен для указания названия документа и предназначен для лиц, изучающих ваш HTML-документ. Отображается элемент <TITLE>, как правило, в заголовке окна браузера.

Элементы <МЕТА> используются для указания автора файла и для указания ключевых слов, на которые можно ориентироваться при поиске информации в поисковых сайтах:

<МЕТА NAME = "Author" Content = "Глеб Окунев”> – указание автора документа;

<МЕТА NAME = "Keywords" Content = "информация, технология, система"> – указание ключевых слов для поиска поисковыми сайтами.

Лучше всего с использованием элемента META указывать также вид кодировки русских букв:

<META Charset=“ windows-1251”>.

В теле документа записывается отображаемая часть сайта. В принципе здесь можно записывать любой полезный текст, который будет отображаться в окне браузера. В отличие от редакторов оформительская часть текста (например, курсив или жирный текст) должна быть указана определенным элементом, поэтому нет смысла набирать в редакторе текст отличный от обычного его написания.

При наборе текста переносы в окне редактора не будут соответствовать переносам в окне браузера. Переносы в окне браузера выполняются после заполнения соответствующей строки окна браузера. Это же относится и к пробелам между словами – всегда между словами будет один пробел независимо от их количества в редакторе. Принудительный перенос на новую строку в браузере осуществляется элементом <BR>.

Текст, помещенный между <H1> и </H1>, будет являться заголовком.

Вместо H1 можно указывать H2, H3, H4, H5, H6. Элемент H1 отображает заголовок самым большим шрифтом, элемент Н6 – соответственно самым малым шрифтом. Атрибут Align используется для выравнивания заголовка относительно окна браузера. Значение Right выравнивает заголовок по правой границе окна браузера, Center – по центру, Left – по левой границе окна браузера.

Ненумерованный список определяется элементом <UL>, внутри которого может находиться элемент заголовка списка <LH> и обязательно один или несколько элементов <LI> непосредственно самих элементов списка. Фрагмент документа для формирования ненумерованного списка имеет следующий вид:

<UL>

<LH> заголовок списка</LH>

<LI> первый элемент списка

<LI> второй элемент списка

<LI> третий элемент списка

</UL>

В элементе <UL> можно указывать тип маркера списка следующим образом:

<UL Type = Disk> – заполненный кружок

Square – заполненный квадрат

Circle – незаполненный кружок

Если тип маркера указан в явном виде, то именно он и отображается. Если тип маркера не указан, то браузером автоматически происходит его изменение для вложенных списков.

В качестве маркера может быть использована любая картинка, подключение которой выполняется с использованием листов стилей, что будет рассмотрено в соответствующем разделе.

Нумерованный список формируется с использованием элемента <OL>, внутри которого может находиться элемент формирования заголовка списка <LH> и обязательно один или несколько элементов <LI> формирования непосредственно самих элементов списка. Фрагмент документа для формирования нумерованного списка имеет следующий вид:

<OL>

<LH> заголовок списка </LH>

<LI> первый элемент списка

<LI> второй элемент списка

<LI> третий элемент списка

</OL>

По умолчанию нумерация осуществляется арабскими цифрами, начиная с единицы.

Элемент <DL> списка определений в отличие от нумерованных и ненумерованных списков состоит из двух частей: первая служит для задания термина, вторая – для вывода определения этого термина. В одном <DL> может находиться определение нескольких терминов:

<DL>

<LH> заголовок списка </LН>

<DT> термин 1

<DD> определение термина 1

<DT> термин 2

<DD> определение термина 2

</DL>

Текст после элемента <DD> будет отображаться с отступом от левой границы окна браузера.

Внутри <DD> могут находиться нумерованные и ненумерованные списки.

Таблицы являются одними из самых используемых при формировании HTML-документа элементов. Часто таблицы используются при формировании меню и еще более часто для рационального и компактного размещения информации на странице. Это можно было заметить в созданных по предыдущим разделам документам: в них содержалось много пустого неиспользуемого места.

* 1. Скриптовые языки программирования

Скриптовый язык (англ. scripting language, также называют язык сценариев) – язык программирования, разработанный для записи «сценариев», последовательностей операций, которые пользователь может выполнять на компьютере. Простые скриптовые языки раньше часто называли языками пакетной обработки (batch languages или job control languages). Сценарии всегда интерпретируются, а не компилируются.

Сценарий (скрипт) – это программа, которая автоматизирует некоторую задачу, которую без сценария пользователь делал бы вручную, используя интерфейс программы.

Поскольку сценарии интерпретируются из исходного кода динамически при каждом исполнении, они выполняются обычно значительно медленнее готовых программ, оттранслированных в машинный код на этапе компиляции. Поэтому сценарные языки не применяются для написания программ, требующих оптимальности и быстроты исполнения. Но из-за простоты они часто применяются для написания небольших, одноразовых («проблемных») программ. Также, в плане быстродействия скриптовые языки можно разделить на языки динамического разбора (sh, command.com) и предварительно компилируемые (Perl).

Языки динамического разбора считывают инструкции из файла программы минимально требующимися блоками, и исполняют эти блоки, не читая дальнейший код.

Предкомпилируемые языки вначале считывают всю программу, компилируют её всю либо в машинный код, либо в какой-то внутренний формат, и лишь затем – исполняют получившийся код.

Для создания пользовательских расширений язык сценариев удобен в ряде случаев:

* безопасность. Скриптовый язык обеспечивает программируемость без риска дестабилизации системы. Скрипты не компилируются, а интерпретируются. Поэтому неправильно написанная программа выведет диагностическое сообщение, не вызывая падение системы;
* наглядность. Язык сценариев используется, если необходим выразительный код. Концепция программирования в скриптовом языке может кардинально отличаться от основной программы;
* простота. Код имеет собственный набор программ, поэтому одна строка может выполнять те же операции, что и десятки строк на обычном языке. Поэтому для написания кодов не требуется программист высокой квалификации;
* кроссбраузерность. Скриптовые языки ориентированы на кроссбраузерность. Например, JavaScript может исполняться браузерами практически под всеми современными операционными системами.

Выделяют следующие типы скриптовых языков:

* универсальные: Forth, AngelScript, Perl, PHP, Python, Tcl (Tool command language), Squirrel, REBOL, Ruby, AutoIt, Lua;
* встроенные в прикладные программы: VBA, UnrealScript, AutoLISP, Emacs Lisp, Game Maker Language, MQL4 script, ERM;
* командные оболочки: sh, AppleScript, bash, csh, ksh, JCL, cmd.exe, command.com, REXX, Visual Basic Script;
* встраиваемые: Guile, Script.NET, ActionScript, Lingo (используется в редакторе Director), Sleep, браузерные Jscript и JavaScript.

Некоторые приложения имеют встроенную возможность расширения сценариями, написанными на любом универсальном скриптовом языке, например, автоматический планировщик задач или библиотека SWIG.

К скриптам также относят многие консольные утилиты, которые поддерживают выполнение записанной в файл последовательности команд.

ЗАКЛЮЧЕНИЕ

Создание веб-сайта состоит из нескольких этапов и взаимосвязанных процессов – таких, как проектирование сайта, создание макетов его страниц, написания наполнения и помещения его на сайт, обслуживание сайта и его программной основы.

В какой-то степени процесс разработки веб-сайта – это аналогия верстки газеты или журнала, или создание изображения в полиграфическом дизайне. В то же время веб-дизайн может включать в себя и проектирование структуры сайта, его навигации и в некоторых случаях даже движков ресурса. Другими словами, продукт веб-дизайна должен включать в себя не только визуальные аспекты сайта, но и его юзабилити – то есть удобство при использовании.

За последние десятилетия всемирная паутина стала отличной информационно-рекламной платформой и потому компании различного масштаба (от крупных транснациональных и мировых корпораций до частных предпринимателей), в том числе не связанные непосредственно с деятельностью в сети. Как правило, одному сайту соответствует одно доменное имя. Именно по нему любой сайт идентифицируются в глобальной сети.

Существует множество инструментов, с помощью которых веб-дизайнер осуществляет верстку страниц. Такие программы называются HTML-редакторами. В веб-дизайне используется два типа редакторов – визуальные и не визуальные – текстовые. Каждый из инструментов имеет свои плюсы и минусы и выбирается под нужны конкретного проекта сайта, либо на усмотрение разработчика.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Батин, Н. В. Основы информационных технологий: учебно-метод. пособие / Н.В.Батин [и др.], под общ. ред. В.В.Шкурко. – Мн.: Институт подготовки научных кадров Национальной академии наук Беларуси, 2008. – 253 с.

[2] Быков, В. Л. Информатика: пособие / В. Л. Быков, Н. Г. Серебрякова – Минск: БГАТУ, 2013. – 656 с.

[3] Лукьянова Н. В. Основы современных информационных технологий: учебно-методическое пособие / Н.В. Лукьянова – М.: МГИУ, 2006. – 285 с.

[4] 5. Сапун, О.Л. Компьютерные информационные технологии: учебно-методический комплекс / О.Л. Сапун [и др.] – Минск: БГАТУ, 2012. – 160 c.

[5] Mozilla Developer Network – Font sizes [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://developer.mozilla.org/ru/> docs/Web/CSS/font-size#Ems

[6] Баканов, А.С. Проектирование пользовательского интерфейса: эргономический подход. / Баканов А.С., Обознов А.А.; под ред. Баканова А.С. – М.: Изд-во «Институт психологии РАН», 2009. – 184с.