

Содержание:

Введение

В настоящее время достаточно популярной и перспективной стала профессия web-разработчика или web-программиста. Это произошло не только за счет появления большого количества сайтов, но и за счет возможности работать удаленно из любой точки планеты.

Основой всего web-программирования является HTML - язык разметки гипертекстовых документов. Актуальность использования данного языка HTML состоит в широком использовании при написании современных web-сайтов. Этот язык составляет основу всех web-страниц, которые пользователи видят, заходя на любой сайт.

С помощью языка HTML легко и быстро можно сделать форму для отправки какого-либо запроса. Такая форма будет иметь все необходимые атрибуты: и поле для ввода и кнопку отправки и остальные элементы web-страницы.

Ч. Муссиано и Б. Кеннеди в работе «HTML и XHTML. Детальное руководство»[\[1\]](#) предоставили полную информацию по этим двум языкам разметки гипертекста. Начиная от самого простого базового синтаксиса и семантики до практических советов, которые можно использовать на любом уровне владения профессией.

Книга Б. Лоусона и Р. Шарпа «Изучаем HTML5»[\[2\]](#), изданная в 2011, знакомит читателя с новыми стандартами web-программирования в среде HTML 5. В ней есть примеры новых функциональных возможностей, которые активно используются ведущими разработчиками.

Развитие web-технологий и дизайна нарастает быстрыми темпами, появляются новые инструменты и стандарты для разработчиков, ориентация на которые является обязательной для создания современных и качественных web-проектов.

Среди современных средств и технологий web-индустрии существует достаточно большое количество средств для создания web-сайтов, но лишь немногие из них способны предоставить разработчикам инструменты для решения подавляющего большинства задач, стоящих перед ними. В этих условиях важным является

отслеживание трендов в области web-индустрии.

Предметом курсовой работы является совокупность характеристик гипертекста и тегов, используемые в HTML5.

Объектом курсовой работы является язык гипертекстовой разметки HTML5.

Целью данной работы является аналитический обзор основных понятий языка гипертекстовой разметки HTML5.

Согласно поставленной цели определены следующие задачи для ее достижения:

- проработать теоретические сведения об истории возникновения HTML5 и Web-технологий;
- провести аналитический обзор характеристик гипертекста и тегов HTML 5;

1. История возникновения HTML 5 и Web-технологий

Вопросам развития HTML 5 и Web-технологий и их применения в настоящее время посвящены работы отечественных исследователей В.Ю. Быкова, М.И. Жалдака, Н.В. Морзе, С.А. Семерикова, А.В. Спиваковского, А.Н. Спирина и др., зарубежных ученых Т. Бернерс-Ли (Tim Бернерс-Lee), А. Кемписато (Oswald Campesato), К. Нильсона (Kevin Nilson), Т. О'Рейлли (O'Reilly, Tim), Д. Хариса (Daniel Harris) и др.

Web-технология - это понятие, которым пользуются в случае обозначения ряда технологий и сервисов Интернета [1].

Т. О'Рейлли отмечает, что Интернет-технология - это методика проектирования систем, которые путем учета сетевого взаимодействия становятся лучше в зависимости от увеличения их пользователей [2].

Всемирная глобальная сеть (World Wide Web дальше Web) имеет интересную историю своего возникновения и развития. Годом рождения технологии Web считается 1992 год, а отцом-основателем Tim Berners-Lee, которому удалось, используя новые сетевые технологии и опыт своих предшественников, сделать технологии Web понятным и удобным средством распространения информации во всемирной сети Internet.

Одновременно с развитием Web-технологий, насыщением Web новыми сервисами и возможностями, развились и язык разметки гипертекста HTML5. С момента своего появления стандарт HTML получил множество изменений, последнее из которых это спецификация HTML 5.0.

HTML (от англ. Hyper Text Markup Language - «язык гипертекстовой разметки») - стандартный язык разметки документов в сети Интернет [3]. Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами и отображается в виде документа в удобной для человека форме.

Язык HTML является приложением SGML (стандартного обобщённого языка разметки) и соответствует международному стандарту ISO 8879.

Во всемирной паутине HTML-страницы, как правило, передаются браузерам от сервера по протоколам HTTP или HTTPS, в виде простого текста или с использованием шифрования.

Язык HTML был разработанной британским ученым Тимом Бернерс-Ли примерно в 1986-1991 годах в стенах Европейского Центра ядерных исследований в Женеве (Швейцария) [4]. Элементами являются структурные и семантические элементы - дескрипторы. Дескрипторы также часто называют «тегами». Также, в HTML внесена поддержка гипертекста.

Таким образом, web-страницы создаются с помощью языка разметки гипертекста HTML. Описание web-страниц содержится в HTML-программе, которая хранится в обычном текстовом файле с расширением htm или html. Программы на языке HTML содержат инструкции (коды), называемые тегами [5]. Все теги языка HTML выделяются символами-ограничителями <i>, между которыми записывается идентификатор (имя) тега и, возможно, его параметры. Большинство тегов HTML используется попарно, то есть для отдельного тега (открывающего) в документе есть соответствующий закрывающий тег, который записывается так же, как и открывающий, но символом / перед именем тега (закрывающие теги не используют параметры) [6].

Со временем Интернет стал востребован не только учеными: к нему пришли и рядовые пользователи, надо отметить, что их количество неуклонно росло.

Для многих было очевидно, что HTML, не предусматривал динамики в созданных с его помощью документов. Это дало толчок развитию технологий CSS, внедрение

поддержки апплетов Java, а после и сценариев (первым языком была JavaScript).

Нельзя не отметить, что с образованием и ростом Сети создателей HTML не на шутку взволновала «чистота» своего детища. Существовали небезосновательные опасения, что производители браузеров, которые вскоре должны были прийти на рынок, будут «баловать» своих пользователей фирменными нововведениями, которые, в свою очередь, будут поддерживаться только фирменными браузерами.

В 1994 году была создана организация, которая взяла на себя разработку единых стандартов развития WWW - World Wide Web Consortium (W3C). Эта организация и занялась подготовкой стандартов HTML (начиная с HTML 2.0). но, несмотря на наличие W3C, нововведения в HTML начинали поддерживаться производителями браузеров гораздо раньше, чем эта организация их стандартизировала (так, например, было с фреймами, с использованием сценариев в HTML-документах, с объектной моделью документов и т.п.).

Организация W3C существует и сейчас. Она занимается теми же вопросами стандартизации (последняя спецификация HTML5 была принята 14 февраля 2014 года).

Язык HTML в том виде, в котором он существует сейчас, обладает большим потенциалом представления информации. Причем он рассчитан не только на пользователей персональных компьютеров. Документы HTML можно просматривать на очень большом количестве различных по своим возможностям устройств: от черно-белого экрана мобильного телефона до телетайпа или терминала. Кроме того, в последние версии HTML включен ряд возможностей, облегчающих представление документов неграфическими средствами (например, речевой синтезатор), которые позволяют пользоваться услугами WWW людям с ограниченными возможностями.

Текущее положение HTML5 не так запутано, как было прежде, но все равно не до конца понятно. Над HTML5 работают две группы. WHATWG создает спецификацию HTML5 в рамках процесса «утвердить, затем пересмотреть». Рабочая группа по W3C HTML берет эту спецификацию и проводит ее через процесс «пересмотреть, затем утвердить».

Ян Хиксон в интервью сказал, что ожидает, что HTML5 получит статус предложенной рекомендации в 2022 году. Поэтому последовала волна общественного возмущения от ряда web-разработчиков. Они не понимали, что значит «Предложенная рекомендация». В данном случае для того, чтобы получить

статус «предложенной рекомендации», нужно иметь две полных реализации HTML5. Учитывая объем спецификации, эта дата невероятно амбициозная.

Вывод: в данной главе рассмотрена история возникновения языка гипертекстовой разметки HTML 5 и Web-технологий, проведен критический анализ литературы по теме.

2. Обзор характеристик гипертекста и тегов HTML 5

Гипертекст обеспечивает функционирование больших объемов текстовой, графической и других видов информации в виртуальном пространстве [7]. Коммуникационные процессы осуществляются в условиях множественного, фрагментарного дискурса, который постоянно разрастается и приобретает новые форматы. Формируются новые типы сообщений, смешанные с лексическими и грамматическими характеристиками в результате взаимодействия языков и видов информации. Все это связано с процессами влияния новейших информационно-коммуникационных технологий на процессы глобализации [8].

Важной спецификой "гипертекста" является его имманентность - рост сознания личности со средствами коммуникации и способность человека усваивать знания мгновенно и приобретать навыки работы с новейшими коммуникационными средствами [9].

Информативность гипертекста связана с использованием системы ссылок. В частности гипертекст характеризуется специфическими параметрами, которые зависят от особенностей построения системы ссылок в гипертексте: интрагипертекстовость (когда ссылки из одного источника на другой связываются в пределах одного текста), интергипертекстовость (когда ссылка из одного источника на другой связываются в пределах отдельных текстов) и экстрагипертекстовость (когда из ссылки одного источника на другой образовывается связь между отдельными гипертекстами в пределах одного текста).

К общим характеристикам гипертекста целесообразно отнести [10]: интерактивность (включение читателя к созданию собственного текста с помощью устройств ввода-вывода информации в персональном компьютере - читатель, "кликаая" мышью на определенные ссылки, активизирует ту информацию, которая

его интересует), мультимедийность (гипертекст кроме собственно текстового представления информации может содержать аудио-, фото- или видеоинформацию в on-line или off-line режиме); динамику (гипертекст можно в любом виде и в любой момент вывести на распечатку, в середине гипертекста могут содержаться ссылки на графики или диаграммы, которые постоянно меняются - динамические данные), дисперсность структуры (разбросанность элементов гипертекста как совокупности внешних элементов и связей между ними).

Большинство документов имеют стандартные элементы, такие, как заголовок, параграфы или списки. Используя теги HTML5 можно обозначать данные элементы, обеспечивая WEB-браузеры минимальной информацией для отображения данных элементов, сохраняя в целом общую структуру и информационную полноту документов [11]. Все что необходимо, чтобы прочитать HTML-документ — это WEB-браузер, который интерпретирует тэги HTML и воспроизводит на экране документ в виде, который ему придаст автор.

В большинстве случаев создатель документа строго определяет внешний вид документа. В случае HTML разработчик (основываясь на возможностях WEB-браузера) может в определенной степени управлять внешним видом документа (но не его содержимым). HTML5 позволяет отметить, где в документе должен быть заголовок или абзац с помощью тега HTML5, а затем предоставляет WEB-браузеру интерпретировать эти тэги [12]. Например, один web-браузер может распознавать тэг начала абзаца и представлять документ в нужном виде, а другой не имеет такой возможности и предоставляет документ в одну строку. Пользователи некоторых WEB-браузеров имеют, также, возможность настраивать размер и вид шрифта, цвет и другие параметры, влияющие на отображение документа.

HTML-теги стандартно подразделяются на два подвида [13]:

- Теги, которые определяют, как будет визуализировано тело документа WEB-браузером;
- Теги, которые описывают общие свойства документа, например, такие как заголовок документа.

Главным достоинством HTML5 является то, что документ может быть отображен на большинстве WEB-браузерах и на многочисленных платформах [14]. HTML-документы могут быть разработаны и внедрены благодаря любому текстовому редактору или специальным HTML-редакторам.

Например, HTML редакторы, такие, как "Netscape Navigator Gold" компании Netscape позволяют создавать документы графически с использованием технологии WYSIWYG (What You See is What You Get). С другой стороны, существует большинство традиционных средств для создания гипертекстовых документов - конвертеры, позволяющие превращать документы к формату HTML.

В приложении 1 приведен список базовых тегов языка гипертекстовой разметки HTML [15].

HTML5 - новая открытая платформа, предназначенная для создания веб-приложений использующих аудио, видео, графику, анимацию и многое другое [16].

HTML5 существенным образом упростил процесс создания веб-приложений. Благодаря HTML5 веб-страницы могут хранить данные локально в браузерах пользователей, что позволяет отказаться от HTTP-cookie. Контент доставляется быстрее и безопаснее. HTML 5 также упростил процесс кроссбраузерности. Обычно, браузеры использовали различные плагины для проигрывания мультимедиа файлов. Встроенная поддержка аудио и видео устраняет проблемы с совместимостью.

К основным преимуществам использования HTML 5 относится:

- В основном HTML5 совместим с существующими стандартами для HTML 4. Новые средства разметки работают согласно существующих, новые API основываются на существующих JavaScript / DOM, которые разработчики используют в течение многих лет.
- Добавляет новые мощные средства в HTML, которые были ранее доступны в Интернет только с помощью технологии плагинов, таких как Flash, или с помощью сложного кода JavaScript или иных специальных приемов.
- Более эффективен для написания динамических приложений, чем предыдущие версии HTML.
- Имеет четкий алгоритм синтаксического анализа, поэтому все браузеры, которые реализуют HTML 5, будут создавать одинаковое DOM дерево с одной разметкой, что является огромным плюсом для совместимости.

Главной задумкой разработчиков стандарта HTML5 является создание рабочего языка гипертекстовой разметки, который будет успешно и качественно функционировать на практике.

HTML5 должен стать более гибким и устойчивым языком гипертекстовой разметки, как HTML 4.0, поэтому в стандарте большое внимание отводится обратной совместимости. С одной стороны, новый язык не полностью совместим с предыдущими версиями, но, с другой стороны, имеет весомые отличия, например, как XHTML 2.0.

На сегодняшний день HTML5 известен в двух форматах - как разновидность языка XHTML и как оптимизированная модификация HTML. Разработчики web-страниц могут использовать любой из этих форматов в своей деятельности. Согласно первому формату новый язык согласуется с XML 1.0, таким образом разработчикам предоставлены все преимущества языка XML и сопутствующих технологий. Во втором формате HTML5 ликвидирована формальная совместимость с SGML. Все предшествующие стандарты HTML5 претендовали на данную совместимость, однако никакой браузер не мог обработать код HTML со сложными правилами SGML. Вместо этого применялись простые алгоритмы, распознающие набор определенных атрибутов и элементов.

Данное действие браузеров было окончательно согласовано в языке HTML5. Новый стандарт не поддерживает технологию SGML, но при этом описывает формальные правила для перевода определенных тегов и других синтаксических конструкций. В итоге, HTML5 формализует принципы, по которым браузеры могут обработать код.

На современном этапе своего развития HTML5 структурировано и детально описывает поведение браузеров в отличие от устаревших стандартов, предназначенных в большинстве своем только для разработчиков кода. Он выдвигает требование для браузеров производить обработку не только рационального кода, но и по возможности исправлять ошибки, вплоть до использования отсутствующих элементов в стандарте.

Например, язык гипертекстовой разметки HTML5 не предоставляет возможности разработчикам кода применять такие элементы как `nobr` или `marquee`, однако при этом описывает, как браузеру необходимо действовать, если в документе они присутствуют.

Еще одной ключевой особенностью нового стандарта HTML5 является то, что производит описание документа HTML5 в определении объектной модели (DOM). Перед тем, как показывать web-страницы браузер создает древовидную структуру объектов и соответствующих элементов.

Данный этап преднамеренно скрывался в устаревших стандартах, однако HTML5 данный этап демонстрирует в виде соответствующего представления документа.

Это позволяет включить в спецификацию объекты, свойства и методы DOM, которые широко используются с незапамятных времен, однако никогда не были стандартизованы: `window.open ()`, `alert ()`, `prompt ()`, `document.links`, `document`, `images`, `innerHTML`, `contentEditable` и так далее. Наконец, в HTML 5 официально включен элемент `embed`.

Однако HTML 5 содержит в своем составе не все, что поддерживается в браузерах. Некоторые элементы, наоборот, удалены из стандарта: `acronym`, `applet`, `basefont`, `big`, `center`, `dir`, `font`, `isindex`, `s`, `strike`, `tt`, `u`. Отсутствуют фреймы (`frame`, `frameset`, `noframes`), хотя остался элемент `iframe`. Нет больше атрибутов, которыми почти никто не пользовался (`accesskey`, `longdesc` и др.) или которыми нет смысла пользоваться (`align`, `border`, `valign`, `cellpadding`, `cellspacing` и др.). Некоторые старые элементы иначе переосмыслены: например, `i` и `b` теперь означают разные формы смыслового выделения, а не курсив и полужирный шрифт; `hr` отвечает за разрыв в структуре документа.

Введена замена следующих элементов:

- `<applet>`, `<object>` заменено тегом `<embed>` (используется для внедрения объектов);
- `<acronym>` заменено тегом `<abbr>` (используется для аббревиатур);
- `<bgsound>` заменено тегом `<audio>` (для внедрения проигрывателя на страницу);
- `<frame>` заменено тегом `<iframe>`;
- `<listing>`, `<xmp>` заменено тегами `<pre>` и `<code>` (для вставки листингов программ и кодов);
- `<plaintext>` - заменено тегом `<pre>`.

Элемент `menu` получил второе значение и позволяет создавать настоящее меню, а `address` ограничен в правах и предназначен строго для контактной информации в узком контексте.

Также в HTML5 введены множество новшеств. Для статей, блогов, документации и подобных материалов будут полезны элементы `article` и `section`, а также новая система заголовков и рубрикации. "Шапка" и "подвал", панель навигации и боковая панель описываются элементами `header`, `footer`, `nav` и `aside`.

Для мультимедийного содержания появились элементы audio и video. Они внедряют в страницу соответствующий ресурс и обеспечивают стандартные элементы управления - кнопки запуска, паузы и остановки, регулятор громкости.

Для управления отображением аудио доступны атрибуты, перечисленные в таблице 1.

Таблица 1

Атрибуты тега <audio>

Атрибут Описание

controls Показывает или прячет кнопки управления проигрыванием

autoplay Начинает воспроизведение сразу после отображения браузером

loop Повторяет воспроизведение аудио-файла

src URL аудио-файла

Варианты значений для загрузки файла перед воспроизведением: none: preload не буферизировать файл заранее metadata: загружать метаданные из файла auto: загрузить весь файл до воспроизведения

Разные браузеры отображают управляющие элементы по-разному, так как это не определено в спецификации W3C (рис. 1). К моменту утверждения стандарта разработчики браузеров обещают добавить возможность стилизации элементов управления с помощью CSS-стилей.

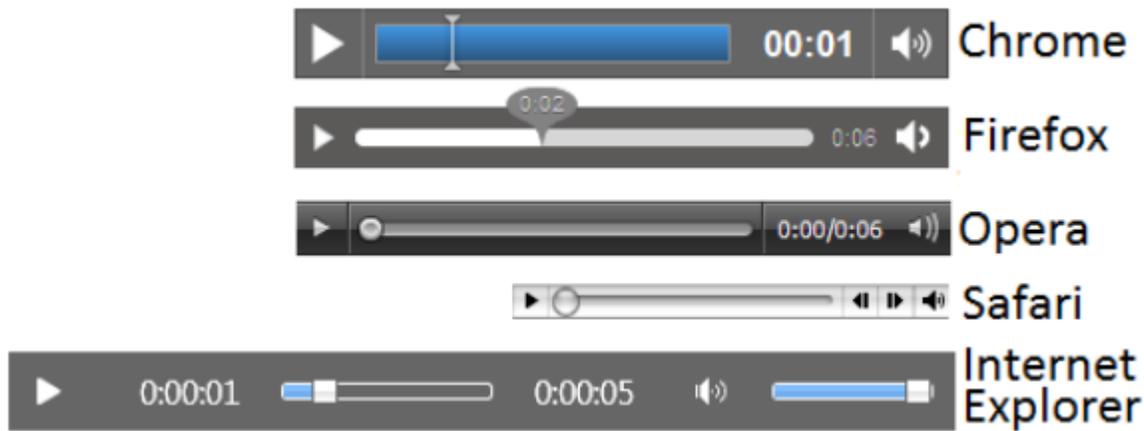


Рис.1 Представления элементов управления тега <audio> в разных веб-обозревателях

Тег <video> имеет дополнительные атрибуты, позволяя настроить то, как следует отображать видео, однако атрибуты controls, autoplay, loop, src и preload семантически равносильны таким же атрибутам тега <audio>. Дополнительные возможные атрибуты тега перечислены в таблице 2.

Таблица 2

Атрибуты тега < video>

Атрибут Описание

width Ширина видео-фрагмента

height Высота видео-фрагмента

poster Адрес картинки, которая будет показываться до начала воспроизведения видео файла

Для тех браузеров, где эти элементы не поддерживаются, HTML 5 позволяет дополнить их альтернативным содержанием - например, роликом Flash, с помощью которого сейчас внедряется большинство аудио- и видеозаписей. В продвинутых браузерах полноценно работает элемент audio или video, а во всех других - альтернативный ролик, так что все пользователи останутся довольны.

Здесь проявляется еще один принцип - новые возможности HTML 5 должны создавать как можно меньше проблем в тех браузерах, где они не поддерживаются. Почти все нововведения языка устроены так, что для старых браузеров можно обеспечить их реализацию с помощью альтернативного содержания, CSS, JavaScript и других технологий. Отдельного внимания заслуживает элемент `canvas`, предоставляющий область для рисования изображений "на лету" средствами JavaScript.

Элемент `<canvas>` и соответствующий API позволяют определить на странице область для рисования, и использовать команды JavaScript для рисования линий, фигур и текста, импорта и манипуляций с изображениями и видео, экспорта в различные форматы изображений и многих других вещей.

Элемент `<canvas>` и соответствующий API для рисования и анимации. Canvas использует команды JavaScript для создания растровой графики, то есть линий, фигур и текста, импорта и манипуляций с изображениями и видео, экспорта в различные форматы изображений и многих других вещей на странице. Canvas задает прямоугольную область, которая контролируется разработчиком и предназначена для сценариев визуализации двумерных 2D-форм и растровых изображений.

Элемент `<canvas>` идеально подходит для создания визуальных материалов, которые улучшают пользовательские интерфейсы, диаграммы, фотоальбомы, схемы, графики, анимированные материалы и встроенные графические приложения.

Собственно, массовый интерес к HTML 5 начался после того, как Firefox и Opera решили поддерживать этот элемент.

Формы в HTML5 также претерпели существенные изменения. Стандартные процедуры заполнения и отправки формы, описанные в прежних версиях HTML, уже сейчас не используются, поэтому разработчики сайтов создают собственные расширения интерфейса, средства проверки, новые элементы управления. В HTML 5 эти разработки урегулированы со стандартными элементами, атрибутами и интерфейсами DOM.

Например, `input type = "number"` - это поле для ввода числа, а `input type = "range"` - ползунок для регулировки громкости или чего-то похожего. В качестве второстепенных атрибутов данный подвид тега `<input>` может принимать параметры, определяющие стиль обрисовки и другие характеристики элемента

(таблица 3).

Таблица 3

Атрибуты тега `<input type = "number">`

Атрибут Описание

`value` Значение по умолчанию

`min` Минимальное значение, которое может ввести пользователь

`max` Максимальное значение, которое может ввести пользователь

`step` Шаг увеличения значения

Атрибуты `min`, `max` и `step` определяют диапазон и точность допустимых значений в поле. Элемент `datalist` обеспечивает поле со списком, а элемент `output` - область для вывода динамически генерируемых результатов. Атрибут `required` для элемента управления означает, что этот элемент обязателен для заполнения, без него форма не будет отправлена. Атрибут `autofocus` позволяет установить фокус ввода в этот элемент.

Метод `checkValidity ()` проверяет правильность заполнения полей. До сих пор эти возможности имитировались на сайтах с помощью недоработанных сценариев JavaScript. В стандарте предлагается элемент `datagrid` для организации динамических структур данных, элемент `command` предназначен для вызова пользовательских команд, элемент `progress` - индикатор хода выполнения операции, а также реализованы средства DOM для редактирования содержания прямо на странице, перетаскивания, выделения, работы с буфером обмена.

Атрибут тега `<input>` - `datalist`. Выдает список возможных значений в поле, при нажатии клавиши "наверх" или "вниз". Первый вводимый в поле символ осуществляет фильтрацию значений по этой букве. Данный атрибут может быть полезен в случае как усложненный вариант самостоятельного вписывания решения задачи (текстом), для осуществления проверки на правописание. Отличие данного

подхода от использования готовых решений на базе AJAX, когда данные при вводе пользователем подгружаются с сервера в том, что весь список уже заранее подготовлен (скачан) и поиск ведется без использования сетевого подключения.

Атрибуты тега `<input>` – `email`, `website`, `phone` являются помощниками для мобильных устройств, где, в зависимости от данной пометки, телефоны могут автоматически выбрать специальную раскладку клавиатуры, предназначенную для облегчения ввода данных запрашиваемого формата.

Общая тенденция развития Всемирной Паутины такова, что сайты все больше превращаются в полноценные приложения, требующие функционального и удобного пользовательского интерфейса. Соответственно, новый HTML также должен стать не столько языком публикаций, сколько языком описания приложений.

Вывод: в данной главе проведен обзор базовых характеристик гипертекста и особенности тегов, используемых в языке гипертекстовой разметки HTML 5.

3. HTML5 как эффективное средство создания Web-приложений: отличия от предыдущих версий

В последнее время все большее признание и развитие в сети Интернет приобретают так называемые браузерные web-приложения. Для решения многих современных задач сегодня нет необходимости обращаться к домашним и офисным стационарным системам, достаточно лишь иметь web-обозреватель и выход в Интернет. Серьезным шагом в этом направлении стало появление языка программирования HTML5.

Каждый программист знает о существовании языка гипертекстовой разметки HTML. Пятая версия этого языка представляет собой качественно новый этап в развитии интернет - технологий. HTML5 разработан для повышения качества интеграции интернет - страниц с мультимедийными технологиями, при этом обладая хорошей обратной совместимостью. HTML5 содержит в себе огромное количество инструментов для разработки сайтов.

Данный язык программирования предоставляет обширные возможности по созданию игровых и веб-приложений, по работе с мультимедийным контентом, а

также в сфере технологий веб-дизайна. Спецификой HTML5 является простая структура и чистота кода. Разметка страниц предыдущих версий HTML осуществлялась в основном при помощи элементов, которым присваивали классы `class` или идентификаторы `id` для наглядности разметки (например, `<div id="header">`). С их помощью в HTML-документе размещали верхние и нижние колонтитулы, боковые панели, навигацию и многое другое.

Стандарт HTML5 предоставил новые элементы для структурирования, группировки контента и разметки текстового содержимого. Новые семантические элементы позволили улучшить структуру веб-страницы, добавив смысловое значение заключенному в них содержимому (было `<div id="header">`, стало `<header>`). Для отображения внешнего вида элементов не задано никаких правил, поэтому элементы можно стилизовать по своему усмотрению.

Таким образом, оформление сайтов, а также размещение на них каких-либо медиа-элементов претерпело значительные изменения по сравнению со старыми версиями HTML. В старых версиях этого программного языка для внедрения какого-либо медиа-контента в интернет-страницу необходимо было использовать сторонние платформы (например, adobe flash).

HTML5 дает возможность размещать медиа-элементы на веб-страницах без каких-либо сторонних программ и, таким образом, существенно облегчает работу с файлами мультимедиа. Изменения также затронули процесс ввода данных, который в пятом поколении языка программирования стал намного проще и удобнее. Что касается внешнего вида, то новый язык программирования существенно расширяет возможности оформления сайтов и приложений, за счёт чего HTML5 игры и сайты станут ещё более оригинальными и интересными. Функции Canvas, добавленные в пятой версии HTML, позволяют размещать на сайтах небольшие HTML5 игры, анимацию и динамическую графику без использования дополнительных программ. К тому же данная функция позволяет создавать качественно новые HTML5 игры и пользовательские интерфейсы. Несмотря на длительный срок существования и появление все новых технологий в веб-дизайне, HTML до сих пор остается одним из самых популярных инструментов разметки. Это обусловлено многочисленными достоинствами этого языка, которых со временем становится только больше.

Перечислим основные достоинства HTML5:

1. Возможность применения более простого кода по сравнению с предыдущими версиями, где `div` заменены усовершенствованными элементами `nav`, `article`, `header`, `footer`, `aside` и `section`.

2. `Header` собирает в группы все вводные и навигационные элементы, что для оформления кода страниц не является обязательным. Включает в себя заголовки, содержание раздела страниц, форму поиска и логотип. Интернет-страница может содержать несколько таких элементов и располагать их в любой своей визуальной части. Применяется для смыслового выделения головной части информационного блока.

3. Элемент `nav`, предназначен для выделения главного навигационного меню сайта. Элемент `nav` может располагаться в HTML-документе в нескольких экземплярах, что ошибкой не является но не рекомендуется, так как тогда теряется его смысловое значение.

4. `Article` применяется для группировки записей, к которым относятся статьи, разного рода публикации, комментарии, записи блога. Представляет собой обособленный независимый блок, начинающийся, как правило, с заголовка и предназначенным для многократного применения. В `article` оборачивается семантически не зависимый элемент, который является самостоятельным и может быть показан на любой другой странице сайта не зависимо от контекста. Так же элемент может быть вынесен на другой сайт. Если на странице присутствует только одна статья с заголовком и текстовым содержимым, она не нуждается в обертке элементом. Элемент целесообразно использовать только в том случае, когда его содержимое непосредственно указано в схеме документа. Типичный пример использования такого подхода, это новостные сайты. На сайте источника размещен блок `article` с описанием статьи, датой публикации, заголовком, автором и др. На сайте агрегаторе новостей работает бот собирающий такие статьи на разных ресурсах и размещает их на свое сайте со ссылкой на первоисточник.

5. Элемент `section` представляет собой универсальный раздел документа. Группирует тематическое содержимое и обычно содержит заголовок. Не является блоком-оберткой, для этих целей уместнее использовать элемент `div`. В качестве содержимого может выступать абсолютно любой контент, который по смыслу может быть определен в один блок. Домашняя страница сайта также может быть поделена на секции – блок вводной информации, новости и контакты. Элемент рекомендуется использовать только в том случае, если его содержимое будет явно указано в схеме документа.

6. Элемент `<aside>` группирует содержимое, связанное с окружающим его контентом напрямую, но в то же время является независимым (т.е. удаление этого блока не повлияет на понимание основного содержимого). Чаще всего элемент позиционируется как боковая колонка (как в книгах) и включает в себя группу элементов: цифровые данные, цитаты, рекламные блоки, архивные записи.
7. Утонченные дизайнерские формы, позволяющие создавать эксклюзивный облик веб-сайта. В HTML5 удалось существенно улучшить пользовательский интерфейс, сделать его более функциональным, простым и интуитивно понятным. Валидация форм теперь доступна встроенными в HTML 5 средствами.
8. Возможность использования новых полей ввода для различных целей, включая поиск.
9. Усовершенствованная семантика HTML5 позволяет быстрее и проще различать футер, заголовки, панель навигации.
10. Использование элементов HTML5 позволяет сделать более удобным для восприятия семантическое ядро каждой страницы. Для реализации такой задумки теперь используются только стандартные коды.
11. Усовершенствованное юзабилити. Теперь возможно создание сайтов с HTML5, используя новые элементы, включая `<canvas>`, `<audio>`, `<video>`. Усовершенствованы возможности интеграции с содержимым SVG. Таким образом, при использовании новых элементов отпала необходимость в установке таких плагинов, как Flash. Музыка, чарты, видео и рисунки можно встраивать без привлечения сторонних программ, поскольку такая функция имеется в самом браузере. При желании можно использовать старую систему интеграции аудио- и видеофайлов.
12. Оптимизация структурирования документов. Для этого используются новые теги, например, `<header>`, `<section>` и `<article>`.
13. Улучшенное восприятие разметки страницы: теперь исходный код интуитивно понятен пользователю.
14. При помощи HTML5 удалось значительно улучшить клиентскую часть приложений. Для этого разработаны доски общений, вики и drag-n-drop.
15. Благодаря уменьшению размера куки разработчики смогли существенно увеличить скорость отклика страницы. В HTML5 этого удалось добиться за счет

использования хранилищ данных localStorage и sessionStorage, которые стали качественной частичной заменой куки.

16. Geolocation API для определения местоположения, полученная информация может использоваться в веб-приложениях. HTML5 считается одним из самых лучших современных инструментов разметки, теперь только этот язык позволяет комфортно и быстро создавать веб-порталы.

С каждым днем все больше браузеров поддерживают основные функции HTML5. Основными нововведениями HTML5 являются:

- Figure – рисунок.
- Header – заголовок страницы.
- Section – крупный блок страницы.
- Footer – «Подвал» страницы.
- meta charset = «UTF-8» – обновление кодировки страницы.
- Nav – навигация по интернет-сайту.
- Aside – дополнительный контент в виде боковой колонки.
- Article – статья, часть основного контента.

Специалисты утверждают, что в ближайшем времени HTML5 займет лидирующую позицию среди всех существующих инструментов разметки. Разработчики уверены, что главным преимуществом HTML5 является возможность с легкостью выставлять на страницы аудио, видео и картинки. Для этого используются новые теги. К примеру, при помощи всего лишь одного тега audio можно вставить аудиозапись.

Тег video позволит вставить в выбранное место страницы видеофайл. При помощи тега canvas появилась возможность размещения на страничках динамических изображений и карт. Для введения метрических данных используется тег meter, а для выделения даты и времени теперь применяется time. Подробная информация об объекте обозначается тегом details, процесс выполнения в динамике обозначен тегом progress.

Разрабатывая сайты с использованием HTML5, можно ощутить преимущества использования таких тегов, как menu, который служит для системы меню, и

datagrid, который представляет собой таблицу, построенную через динамический источник данных. Тег `command` позволяет определить действие, которое осуществляется при активации объекта. Кроме того, в новую версию HTML добавлены формы полей, включая `email`, `datetime`, `url`, `time`, `number`, `datetime-local` и множество других. Разумеется, введение новых тегов во все браузеры, как и освоение новых данных специалистами, потребует определенного времени. Однако уже совсем скоро использование новых тегов HTML5 станет де-факто стандартом на рынке веб-разработки.

Помимо неоспоримых достоинств HTML5 следует сказать несколько слов и об имеющихся недостатках этого языка. К их числу можно отнести, прежде всего, недостаточную защиту данных. При использовании HTML5 на винчестере пользователя долгое время хранится большой объём данных, что позволяет злоумышленникам осуществлять несанкционированный доступ к ресурсам компьютера с целью сбора конфиденциальной информации. В отношении мультимедийного контента также имеются некоторые недостатки. С одной стороны, поддержка видео HTML5 позволяет воспроизводить мультимедийный контент на компьютерах и устройствах, которые не поддерживают Flash Player. Вместе с тем, в полной мере HTML5 всем этим потребностям пока не отвечает. Например, в HTML5 пока нет полноэкранный режим. Кроме того, имеются недостатки в использовании HTML5 для создания игр. Наконец, новый стандарт поддерживается пока не всеми браузерами.

Вместе с тем, недостатки новой версии языка программирования не снижают его востребованности даже среди таких гигантов, как YouTube, Google, Apple, Facebook и др., которые активно занимаются его популяризацией.

Выводы: в данной главе рассмотрены и систематизированы основные отличия HTML5 от предыдущих версий. HTML5 рассмотрен как эффективное средство для создания web-приложений.

4. Анализ проблемы обеспечения кроссбраузерности в HTML5 и пути ее решения

Сегодня существует большое количество веб-браузеров, которые отличаются как визуально, так и функционально. Для разработчика сайта функциональность браузера и его поддержка кроссбраузерности стоит на первом месте при создании

сайта. Больше всего возникает проблем с использованием трансформации, анимации, реагирования на клики мышкой или любой другой клавишей, которые прописаны с помощью html и css. Во избежание проблем с такими эффектами программисты от них или отказываются, или прописывают для каждого браузера соответствующую функцию. Так для прописи трансформации (transform) добавляют префикс -o- (Opera), -moz- (Mozilla FireFox), -ms- (Internet Explorer), -webkit- (Chrome) и т.д.

С появлением HTML5 проблема отображения сайтов временно обострилась. На конец 2018 года стандарт HTML 5 поддерживают практически все браузеры.

Другим фактором, который влияет на функцию кроссбраузерности, является применение различных CMS, фреймворков, которые без соблюдения соответствующих правил не смогут сохранять возможность сайта адаптироваться под различные расширения мониторов компьютеров или других устройств. Хорошим примером является разработка CSS-фреймворка Twitter Bootstrap с помощью HTML5, CSS, LESS и JavaScript, в котором предусмотрена адаптивная верстка и есть возможность сразу прописывать ссылки на фиксированные элементы Bootstrap с соответствующими свойствами. Совершенно противоположная ситуация с CMS: для обеспечения кроссбраузерности есть два варианта разработки сайта:

- 1) верстка под различные расширения при помощи медиа-запросов;
- 2) основываясь на расстояниях в %, а размеры элементов задавая через `em`:
`position: relative; left: 10%.`

В ходе исследования проблемы нарушения кроссбраузерности HTML5 было установлено, что в определенной степени по вертикали можно избежать этой проблемы, применяя якоря и свойство `overflow-y: hidden`, что позволит разбить страницу на несколько страниц, по горизонтали - полностью отказавшись от прописывания пикселей и сцентрировав элементы в блоках вместе с самими блоками с помощью `margin: 0 auto`. Но сначала, перед разработкой сайта, необходимо определиться, из скольких блоков будет состоять сайт. Очень важно разбить правильно блоки. Чем больше блоков, тем больше вероятность, что они будут размещаться правильно, не нарушая заданных расстояний. Также было обнаружено, что элементы значительно проще размещать в блоках, собственноручно созданными с помощью php или javascript. Блокам можно задать `padding`, и тогда элементы не будут выходить за рамки блока.

Следовательно, для достижения кроссбраузерности сайтов необходимо учитывать изменения в функционале тех браузеров, с помощью которых пользователи могут просматривать содержание сайта, пытаться использовать такие теги, которые понятны для разных браузеров в сети Интернет.

Выводы: в данной главе рассмотрены основные проблемы обеспечения кроссбраузерности в HTML5, сформированы пути решения данной проблемы.

5. Сравнительный анализ преимуществ и недостатков HTML5

В данном разделе рассмотрим основные преимущества и недостатки языка гипертекстовой разметки HTML5:

1) Улучшенные методы безопасности. На сегодняшний день применяемый большинством сайтов тег `iframe` несет в себе явную угрозу безопасности для пользовательской системы. Проблема состоит в том, что главной функцией этого тега является отображение контента с другого сайта, например, в плане рекламы или сведений о состоянии каких-либо данных. Но если сайт, передающий эту информацию, подвергнется взлому, появляется большая вероятность того, что злоумышленники через него направят вредоносный код на сайт-приемник, который и подвергнет его пользователей опасности заражения вирусом.

В HTML 5, во избежание аналогичной ситуации для тегов `iframe` была спроектирована технология `Sandbox`. Согласно данной технологии, все скрипты, запускаемые через `iframe` функционируют в специально ограниченной виртуальной среде, которая не выпустит результаты их работы за пределы браузера, а значит и не сможет поразить пользовательскую операционную систему.

Даже если предположить, что запуск вируса с помощью этого тега все же произойдет, какие-либо серьезные последствия он понести уже не сможет [17].

2) Введение данных в HTML5 стало намного безопаснее и совершеннее. Правильность ввода данных контролируется в режиме реального времени непосредственно во время ввода. Например, в HTML 4.0 проверка ошибок и сообщения о них происходит уже после отправки документа, что может создавать дополнительные проблемы пользователям [18].

3) Поддержка полноценной графики. До недавнего времени бесспорным лидером анимированной графики для браузеров была технология Flash. С появлением в новом стандарте HTML5 такого инструмента как canvas позиции Flash- графики будут, вероятно, сильно потеснены.

Теперь браузер будет способен воспроизводить не только красивые двумерные интерактивные приложения, но и полноценные трехмерные игры.

Если для HTML 5 и дальше продолжат развиваться направление трехмерных игр, в скором будущем будут все шансы получить игровые проекты с современной шейдерной графикой и соответствующими спецэффектами для нее. Как только графику браузеров полностью переведут на вычислительную поддержку от видеокарт, в сети будут размещены ресурсы с онлайн-играми практически ничем не отличающимися от сегодняшних графически красивых, но отдельно устанавливаемых клиентских игровых приложений.

4) Поддержка Видео и аудио. Для того, чтобы воспроизвести видео или аудио поток в браузере пользователю придется прибегать к помощи все той же Flash технологии, необходимой в HTML 4.0.

Браузерам для этого приходится постоянно пользоваться Flash проигрывателями, а также скачивать постоянные обновления от Adobe для этой технологии, без которых пользователи рискуют заразить свои компьютеры вирусами через проломы в старых версиях Flash.

HTML 5 призван наконец избавить браузеры от этих дополнительных программ и дать им возможность полноценного воспроизведения аудио и видео контента своими силами.

5) Многократное увеличение скорости работы. Сложно представить современные web-сайты без большого числа интерактивных форм, видео и аудио вставок, анимированных графических элементов и разного рода оживляющих сайт программ-скриптов. Конечно, мощности современных вычислительных машин вполне хватает, чтобы обрабатывать такое огромное количество элементов веб-страницы, но Интернет технологии не стоят на месте, и чем больше будет развиваться визуализация, интерактивность и качество мдиа-информации, тем выше будут требования к аппаратному обеспечению пользователя.

Конечно же, проектировщики HTML 5 не могли обойти эту проблему стороной. Новый стандарт обеспечивает поддержку многопоточности, которую именовали

как Web Works. Таким образом, современные многоядерные системы получают возможность параллельно выполнять обработку скриптов и тем самым увеличить скорость работы сайта на несколько порядков в зависимости от его конструкции и сложности.

6) Технология Web Storage. Пользователи комфортно работают с теми сайтами, которые способны сохранять изменения в настройках, например для внешнего вида рабочего пространства. Для того, чтобы сохранить какую-либо информацию о действиях пользователя сайт обычно пользуется технологией cookies, которая позволяет хранить на компьютере посетителя текстовый файл со своими данными.

Технология Web Storage для HTML 5 затмит cookies своими возможностями только потому, что представляет из себя не форматно ограниченный текстовый файл, а специально ориентированную базу данных для различных видов информации. Теперь в такой базе сайт может хранить не только отдельные настройки сайта, а целые фрагменты его кода, что позволит работать с веб-приложениями даже при отсутствии подключения к сети Интернет.

7) Структурные возможности HTML5 создают структуру веб-документа более простую, а код более логичный.

Новые семантические элементы header, nav, section, article, aside и footer могут использоваться вместо контейнеров div, применяемых сейчас в HTML 4.0.

Новые элементы позволяют конкретно описывать заголовок, нижний колонтитул, колонки текста и другие. Эти нововведения могут использоваться для автоматической генерации содержания и организации эффективной и простой навигации по веб-странице. Код при этом становится рациональнее и менее "засоренным" второстепенными деталями.

На сегодня в HTML5 присутствуют также и недостатки, из-за которых он проигрывает прошлым технологиям. Имеет место некоторая уязвимость HTML 5 в области защиты данных. Такую уязвимость создает большой, по сравнению с cookie стандартом HTML4, объем данных, который хранится на жестком диске пользователя. Опасность заключается в том, что в больших объемах данных может содержаться подробная информация о пользователе.

Эксперты высказывают мнение, что если хранилище HTML 5 остается на компьютере пользователя в течение многих месяцев, то интернет-сайты, обращаясь к нему через JavaScript после загрузки страницы, могут незаметно для

пользователя собирать и передавать данные владельцу сайта.

Относительно мультимедийного контента тоже есть недостатки, не позволяющие сделать окончательный выбор в пользу HTML5. Поддержка видео HTML5 позволяет воспроизводить мультимедийный контент на компьютерах и устройствах, которые не поддерживают Flash Player, однако в полной степени HTML5 всем этим потребностям все еще не соответствует. Сегодня Adobe Flash обеспечивает лучшую платформу для воспроизведения видео. В HTML5 пока нет, например, полноэкранный режим.

Есть недостатки, касающиеся использования HTML5 для создания игр. И, кроме того, новый стандарт пока еще не поддерживается всеми браузерами.

Отличие в форматах HTML5 видео / аудио в различных браузерах может привести:

- 1) К увеличению времени тестирования и разработки;
- 2) К увеличению расходов дискового пространства и нагрузки на сервер.

Недостатками HTML 5 при разработке игр являются:

- Медленная работа игр, созданных с помощью HTML 5;
- Не каждая платформа поддерживает игры, разработанные на HTML5;
- При разработке игры на HTML 5 требуется значительно больше ресурсов и времени;
- Открытость кода HTML5 неприемлема для бесплатных игр;
- Отсутствие мощных и удобных инструментов для создания игр, особенно для анимации;
- невозможность адекватно упаковывать игру в один файл для дистрибуции на разных порталах;

Платформа Flash используется также для анимации и разработки интерфейсов, ее применяют для создания сложной рекламы и в трехмерной графике. Она постоянно эволюционирует и совершенствуется.

Как отметил Филипп Ле Эгаре (Philippe Le Hegaret), руководитель направления компании W3C, которая разрабатывает стандарт HTML 5, в ближайшие годы

невозможно будет отказаться от технологии Flash. Причиной этому является длительное время, которое потребуется для перехода всех веб-клиентов к поддержке HTML 5.

Выводы: в данной главе систематизированы главные преимущества использования языка гипертекстовой разметки HTML5, выявлены недостатки при разработке игр.

Заключение

В ходе написания курсовой работы были решены следующие задачи:

- проработаны теоретические сведения об истории возникновения HTML 5 и Web-технологий;
- проведен аналитический обзор характеристик гипертекста и тегов HTML 5.

В итоге отметим, что существует большое количество средств для создания веб-сайтов, в частности язык программирования HTML 5, системы управления базами данных для поддержки веб-сайтов, но лишь немногие из них способны предоставить разработчикам инструменты для решения подавляющего большинства задач, стоящих перед ними.

Разработчики базовых технологий современного веб-пространства обновляют свои продукты, все больше внимания уделяя вопросам скорости, доступности, технологиям динамического обращения к серверу, осуществлению прямого обмена разнообразной информацией между пользователями.

Таким образом, анализ современных тенденций в области веб- программирования позволяет сделать вывод, что веб-индустрия продвигается быстрыми темпами.

Появляются новые стандарты для веб-разработчиков, ориентация на которые является обязательным для привлечения пользователей при создании качественного проекта.

Дальнейшего анализа требуют особенности современных тенденций HTML 5 как составляющей отрасли веб-программирования.

Список использованной литературы

1. Андронатий П.И., Котик В.В. Компьютерные технологии в образовательных измерениях: учебно-методическое пособие. - Кировоград: Лысенко Ф., 2011. - 144 с.
2. O'Reilly, Tim (2007): What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software. Published in: International Journal of Digital Economics, No. 65 (March 2007) — P. 17–37.
3. Идиятуллина А. Т. ОСОБЕННОСТИ КРОССПЛАТФОРМЕННОЙ РАЗРАБОТКИ ИГР НА HTML 5 //ББК 24.3 К 65. - 2015. - С. 15.
4. Робин Н. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 5-е изд. - Издательский дом: "Питер", 2018.
5. Reimers S., Stewart N. Presentation and response timing accuracy in Adobe Flash and HTML5/JavaScript Web experiments //Behavior research methods. - 2015. - Т. 47. - №. 2. - С. 309-327.
6. Белозубов А. В., Николаев Д. Г. ПРИЕМЫ РАБОТЫ С HTML-РЕДАКТОРОМ ADOBE DREAMWEAVER Учебное пособие. - 2009.
7. Лутовинова О. В. Гипертекст: понятие, основные характеристики, возможные подходы к лингвистическому анализу //Известия Волгоградского государственного педагогического университета. - 2009. - №. 5.
8. Асхатова Л. И., Галимов Э. Р., Габдуллин И. М. Развитие языков разметки //APRIORI. Серия: Естественные и технические науки. - 2015. - №. 2.
9. Соболева О. В. Понятие и основные характеристики гипертекста //Альманах современной науки и образования. - 2011. - №. 12. - С. 163-164.
10. Богданова С. В., Ермакова А. Н. Технология создания гипертекстовых документов на базе языка HTML //учеб. пособие/СВ Богданова, АН Ермакова.-Ставрополь: Литера. - 2013.
11. Keith J., Zeldman J. HTML5 for web designers. - New York, NY : A Book Apart, 2010. - Т. 1.
12. Дакетт Д. HTML и CSS. Разработка и дизайн веб-сайтов. - М.: - Litres, 2018. - 408 с.
13. Дунаев В.В. HTML, скрипты и стили. - 2 изд. - СПб.: БХВ-Петербург, 2008. - 1024 с.
14. Дубовский, А. А. Создание и развитие сайта HTML / А. А. Дубовский, И. В. Прусова // Новые направления развития приборостроения : материалы 9-й международной научно-технической конференции молодых ученых и студентов, Минск, 20–22 апреля 2016 г. : в 2 т. / Белорусский национальный технический университет ; редкол.: О. К. Гусев [и др.]. - Минск, 2016. - Т. 2. - 105 с.

15. Ковальчук В. М., Выговская Н. В. Основы WEB-программирования. - М.: Litres, 2018. - 225 с.
16. Брайан Хоган «HTML5 и CSS3. Веб-разработка по стандартам нового поколения» 2 издание, 2014 г.
17. Лэнс Лавдэй, Сандра Нихаус «Web-design for ROI: проектирование прибыльных веб-сайтов», 2016 г.
18. Digital-agency «Ingate» Landing Page: идеальная страница для больших продаж, 2015 г., 8-29 ст.

Приложение 1

Список базовых тегов HTML

`<HTML>`

Этот тег указывает на начало HTML-документа

`<HEAD>`

Этот тег указывает на начало области заголовка Web-страницы. Служит для формирования общей структуры документа.

`<TITLE>Название Web-страницы</title>`

Элемент для размещения заголовка Web-страницы. Строка отображается в заголовке окна браузера.

`<META http-equiv="Content-Type" content="text/html; charset=windows-1251">`

Этот тег несет служебную информацию и не отображается на экране браузера. В данном случае идет речь о кодировке Web-страницы. Вам достаточно лишь каждый раз вставлять этот тег в таком виде на свою страничку. Тогда ваша страничка будет использовать кодировку windows-1251, наиболее распространенную на сегодняшний день.

`<META name="Author" content="Ivanov Ivan">`

Имя автора Web-страницы.

