Департамент образования города Москвы

Государственное образовательное учреждение высшего профессионального образования

“Московский Городской Педагогический Университет”

Математический факультет

Кафедра информатики и прикладной математики

Дипломная работа

По теме:

Создание справочного пособия с помощью системы управления контентом “Joomla” для обучения информатике в школе

Студента 5 курса

очной формы обучения

Цыганова Виктора Ивановича

Научный руководитель: заведующий

кафедрой информатики и прикладной математики доктор технических наук, профессор

Григорьев Сергей Георгиевич

Москва, 2010

**Ведение**

Современный уровень развития электронных устройств и в частности компьютерной техники позволяет создавать новые технологии в различных сферах научной и практической деятельности. Одной из таких сфер стало образование – процесс и результат усвоения систематизированных знаний, умений и навыков.

Образование является мощной информационной сферой, в которой используется опыт различных классических (не компьютерных) информационных систем. Это позволило образованию быстро откликнуться на возможности современной техники. На наших глазах возникают образовательные ресурсы и нетрадиционные информационные системы, связанные с обучением. Эти системы называют информационно-обучающими.

Такие автоматизированные обучающие системы помогают учащимся осваивать новый материал, а преподавателям производить контроль знаний обучаемых, а также подготовить, наглядно и визуально красиво представить учебный материал.

Работа учителя информатики в школе требует применения компьютерных технологий обучения, и мне очень часто приходится использовать компьютерные информационные технологии: мультимедиа, обучающие и контролирующие программы, всемирную компьютерную сеть и многое другое. Подобные ресурсы являются одним из наиболее оптимальных и эффективных средств обучения в современном образовании.

Говоря об объекте исследования, можно сказать, что данная дипломная работа поможет понять и изучить процесс создания такой неотъемлемой части любой автоматизированной системы обучения как справочное пособие. К сожалению, такие системы обучения применяются на практике не так часто, как хотелось бы. Это связано не столько с не эффективностью подобных обучающих систем, сколько с трудностями, возникающими при их создании. В этой дипломной работе будет показан один из вариантов создания и практической реализации одного из основных элементов автоматизированной системы обучения. Данное справочное пособие будет создано с помощью системы управления контентом CMS «Joomla». Дипломная работа покажет, что при умеренной теоретической подготовке и небольшом практическом навыке и опыте работы в CMS «Joomla», подобное справочное пособие сможет создать не только любой преподаватель информатики, но и даже обучающиеся у него ученики.

Предметом исследования моей дипломной работы будет определение основных принципов и методов работы с CMS «Joomla», с помощью которых можно создать образовательный ресурс. В практической части дипломной работы рассмотрены основные этапы создания обучающей системы, с которыми сталкивается разработчик при реализации аналогичного образовательного ресурса. Будет показана эффективность результатов использования справочного пособия по HTML на уроках информатики, при изучении темы «Основные понятия языка HTML».

Актуальность темы дипломной работы заключается в том, что благодаря подобным образовательным ресурсам процесс обучения для учащихся становится более интересным, наглядным, быстрым и эффективным. Также автоматизированные системы обучения помогают преподавателю производить контроль знаний обучаемых, поднимать успеваемость в классе и повышать уровень мотивации изучения предмета у учащихся и развивать у них универсальные учебные действия.

Проблема состоит в необходимости изучения теоретических основ и основных принципов построения образовательных ресурсов на CMS «Joomla».

Цель исследования – разработать справочное пособие по HTML с помощью системы управления контенком CMS «Joomla». Перечислить основные возможности системы управления контентом CMS «Joomla» и показать способы их практической реализации на примере создания справочного пособия по HTML. Продемонстрировать эффективность использования данного справочного пособия на уроках информатики.

Гипотеза исследования – если педагог хочет создать справочное пособие, помогающее учащимся лучше и быстрее осваивать материал, то его можно реализовать с помощью системы управления контентом CMS «Joomla».

Задачи исследования:

* Провести анализ компьютерных обучающих систем, уточнить **основные принципы новых информационных технологий обучения**, выявить типы обучающих систем;
* Рассказать о цели создания и развития глобальной сети интернет, описать современные средства разработки web - сайта, выявить основные достоинства системы управления контентом CMS «Joomla»;
* Подготовить операционную систему к работе с CMS «Joomla», создать справочное пособие по HTML с помощью системы управления контентом CMS «Joomla»; произвести апробацию результатов исследования, показать эффективность использования справочного пособия по HTML на уроках информатики, при изучении темы « Основные понятия языка HTML».

Методы исследования – теоретический анализ литературы по проблемам исследования основных характеристик дипломной работы. Нахождение приемлемых способов и методов апробации. Обобщение педагогического опыта.

Значимость исследования заключается в выявлении основных методов создания справочного пособия по HTML с помощью CMS «Joomla» и дальнейшей модернизации и улучшении данного автоматизированного обучающего ресурса для повышения уровня универсальных учебных действий учащихся.

Структура работы - введение, теоретические и практические главы, заключение и библиография. Во введении описываются основные характеристики дипломной работы: проблема и актуальность, объект и предмет, цель и задачи, гипотеза и методы исследования; определяется значимость исследования, приводится структура работы и краткое изложение содержания каждой главы дипломной работы.

Теоретическая часть работы содержит теоретический материал и основные понятия, затронутые и используемые в практической части дипломной работы. Их знание необходимо для выполнения задач исследования.

Практическая часть работы вытекает из теоретического исследования и содержит основные практические принципы разработки справочного пособия, применяемые при создании ресурса с помощью CMS «Joomla»; основные выводы по эффективности применения разработанного справочного пособия на уроках информатики; описание условий апробации, а также полученные результаты исследования.

Результаты работы – мною было разработано справочное пособие по HTML с помощью системы управления контенком CMS «Joomla». Апробация показала, что благодаря данному образовательному ресурсу, учащиеся стали лучше и быстрее усваивать новый учебный материал. У них выросла общая мотивация к учебе. Улучшилось качество обучения, повысился уровень информационной культуры. Также у учащихся стала проявляться большая степень самостоятельности, в их деятельности появились элементы творчества.

В дипломной работе были перечислены основные возможности CMS «Joomla», определены ее достоинства и недостатки, оговорены особенности, основные принципы и способы работы с данной системой управления контентом.

**Глава I. Теоретическая часть**

**§1. Компьютерные обучающие системы. Основные принципы новых информационных технологий обучения**

С изобретением компьютера, перед человеком отрылось много новых и до этого не существующих возможностей. Наше общество уже нельзя представить без использования компьютерной техники. Современные компьютеры применяются практически во всех областях знаний. Одной из таких наиболее важных и значимых областей, является сфера образования. В педагогике такое направление в образовании называется компьютерные технологии обучения.

Первая обучающая система носила название Plato. Она была разработана в США в конце 50-х годов на основе мощной ЭВМ фирмы « Control Data Corporation ». Эта система развивалась в течение 20 лет. По-настоящему массовым создание и использование обучающих программ стало с начала 80-х годов, когда появились и получили широкое распространение персональные компьютеры. С тех пор образовательные применения ЭВМ выдвинулись в число их основных применений наряду с обработкой текстов и графики, оттеснив на второй план математические расчеты.

Внедрение компьютерного обучения позволило начать массовую разработку обучающих программ. Программы создавались представителями конкретных дисциплин, опираясь в основном на интуицию и практический опыт. В программах разработчики воплощали свои представления о преподавании с помощью компьютера. Педагоги-теоретики долгое время оставались в стороне от этого нового направления в обучении. В результате до сих пор отсутствует общепризнанная психолого-педагогическая теория компьютерного обучения, компьютерные обучающие программы продолжают создаваться и применяться без необходимого учета принципов и закономерностей обучения.

Благодаря своим мультимедийным особенностям современный ПК является уникальным по своим возможностям обучающим средством. Персональный компьютер применяется в обучении самым разнообразным дисциплинам и служит базой для создания большого числа новых информационных технологий обучения. Особенности персонального компьютера выгодно отличают его от других технических средств обучения. Эти особенности заключаются в возможности компьютера одновременно сочетать:

* интерактивный (диалоговый) режим работы (действие человека – реакции компьютера - ... - действие человека - реакция компьютера и т.д.);
* «персональность» компьютера (небольшие размеры и относительно невысокая стоимость, позволяют обеспечить компьютерами целый класс);
* хорошие графические, иллюстративные возможности (среднестатистический компьютер имеет разрешающую способность монитора 1024x768 точек при 32 млн. цветовых оттенков - это качество несопоставимо выше чем у обычного цветного телевизора или иллюстрации в журнале);
* простота управления и интуитивно понятный интерфейс;
* легкость регистрации учащихся в обучаемой программе и хранения информации о процессе обучения и работе учащегося.

Если компьютер используется как обучающее средство, то обычно его технические возможности позволяют:

* + активизировать учебный процесс;
  + индивидуализировать обучение;
  + работать каждому ученику в своем индивидуальном темпе;
  + повысить наглядность в предъявлении материала;
  + сместить акценты от теоретических знаний к практическим (и наоборот);
  + повысить интерес учеников к изучению нового материала и обучению в целом;
  + реализовать автоматизированную систему проверки и контроля учащихся.

Активизация обучения связана с диалоговым характером работы компьютера и с тем, что каждый ученик работает за своим компьютером. При традиционном классном обучении основное - это восприятие учащимися информации в устной форме, при этом ученику не часто приходится проявлять активность на уроке и учитель не в состоянии организовать и контролировать активную работу каждого ученика на его рабочем месте. Поэтому традиционное обучение, в основном, является пассивным и на уроке активно работают 20 – 30% учащихся. Если же обучение ведется в компьютерном классе, компьютер диалоговым характером своей работы стимулирует ученика к деятельности и позволяет контролировать ее результаты.

Индивидуализация обучения при использовании компьютера также связана с интерактивным характером работы с компьютером и наличием компьютеров на рабочих местах: каждый ученик теперь может сам выбирать темп обучения, делать в работе паузы. Более глубокий и тонкий учет индивидуальных особенностей учащихся может осуществлять компьютерная программа, с помощью которой ведется обучение (педагогическое программное средство). Определив уровень обученности ученика с помощью начального теста, программа может, в соответствии с этим уровнем, предъявлять теоретический материал, вопросы и задачи, а также подсказки и помощь. Обучение слабых учеников программа ведет на самом легком (базовом) уровне, изложение теоретических сведений максимально упрощено, вопросы и задачи облегчены, помощь имеет характер прямой подсказки. Обучение сильных учеников ведется на наиболее сложном уровне, теория излагается углубленно, предлагаются творческие задачи, требующие изобретательности и интуиции, а помощь имеет косвенный характер - намека или наводящего на правильный путь вопроса. Между этими крайними случаями обучающая программа может учитывать более тонкую градацию подготовленности учащихся.

Каждый ученик в процессе обучения сталкивается с трудностями связанными с наличием пробелов в знаниях или особенностями мышления. При обучении с помощью компьютера обучающая программа может диагностировать пробелы в знаниях ученика, его индивидуальные особенности и строить обучение в соответствии с ними.

Графические возможности дисплеев персональных компьютеров позволяют сделать компьютерное обучение очень наглядным. На экране дисплея можно показывать геометрические фигуры и построения, стилизованные изображения реальных объектов и т.п. Все это можно реализовать как статически (т.е. неподвижно), так и динамически, в движении. С помощью компьютерной графики можно сделать зримыми такие явления и процессы, которые не могут быть увидены в действительности, можно создать наглядный образ того, что на самом деле никакой наглядности не имеет (например, эффектов теории относительности, закономерностей числовых рядов и т.п.). На этой возможности компьютеров основывается, так называемая, когнитивная компьютерная графика - особое направление применения компьютеров в научных исследованиях, когда иллюстративные возможности компьютера используются для изучения различных закономерностей.

Всегда остро стоит вопрос о соотношении теории и практики применительно к научному знанию. Традиционное обучение является преимущественно теоретическим. Классно-урочная форма обучения подталкивает каждого педагога в отдельности и всю систему образования в целом к усилению теоретической стороны обучения в ущерб практической. В самом деле, любому педагогу излагать теоретические знания у доски и требовать от учеников воспроизведения этого изложения значительно легче, чем организовывать ориентированную на практику работу учащихся. Если же вести обучение с помощью компьютера, оно приобретает практический уклон: диалоговый характер работы с компьютером, его вычислительные моделирующие возможности предрасполагают к обучению в форме решения задач практической направленности.

Важным условием успешного обучения является интерес учеников к изучаемому предмету, ходу обучения и его результату. Этот интерес связан с множеством факторов: содержанием изучаемого предмета, уровнем его сложности, организацией процесса обучения, системой поощрений и наказаний, применяемой учителем, мастерством и интересом к предмету самого учителя, системой ценностей ученика, его ближайшего окружения, родителей, взаимоотношениями в классном коллективе.

Компьютерная технология повышает интерес к обучению предметам, не связанным с информатикой. Новое в организации учебного процесса с участием компьютера, само изменение характера работы ученика на уроке способствуют повышению интереса к учебе. В то же время, более тонкое использование возможностей компьютера позволяет управлять мотивацией учеников во время компьютерного обучения. Здесь имеются в виду, в первую очередь, мотивирующие реплики обучающих программ, т.е. фразы, в которых обучающая программа оценивает работу ученика и стимулирует дальнейшее обучение. Эти фразы могут иметь неформальный характер и создавать положительную эмоциональную атмосферу при работе с компьютером. Важное значение имеют элементы игры, состязательности в компьютерном обучении (например, подсчет очков и сравнение достижений различных учеников) или звуковые и зрительные эффекты (звучание музыкальных мелодий, мигание и цвета на экране дисплея). Вот далеко неполный арсенал возможностей компьютера, делающих его очень перспективным для использования в учебном процессе обучающим средством.

Несмотря на все преимущества использования компьютерных программ в обучении, очень часто у учеников возникает сложности в работе за компьютером. Именно поэтому одной из главных задач учителя работающего с компьютерной техникой правильно подобрать обучающие программы и разработать верную методику использования возможностей компьютера в обучении.

**§2. Типы обучающих систем**

Основанием для классификации служат обычно особенности учебной деятельности обучаемых при работе с программами. Многие авторы выделяют четыре типа обучающих программ:

* тренировочные и контролирующие;
* наставнические;
* имитационные и моделирующие;
* развивающие игры.

Программы 1-го типа (тренировочные) предназначены для закрепления умений и навыков. Предполагается, что теоретический материал уже изучен. Эти программы в случайной последовательности предлагают учащемуся вопросы и задачи и подсчитывают количество правильно и неправильно решенных задач (в случае неправильного ответа может выдаваться поощряющая ученика реплика). При неправильном ответе ученик может получить помощь в виде подсказки.

Программы 2-го типа (наставнические) предлагают ученикам теоретический материал для изучения. Задачи и вопросы служат в программах для организации человеко-машинного диалога, для управления ходом обучения. Так если ответы, даваемые учеником, неверны, программа может «откатиться назад» для повторного изучения теоретического материала.

В таком типа программ основным теоретическим источником современного компьютерного обучения следует считать программированное обучение. Его главным элементом является программа, понимаемая как упорядоченная последовательность рекомендаций (задач), которые передаются с помощью дидактической машины или программируемого учебника и выполняются обучаемыми. Существует несколько известных разновидностей программируемого обучения:

1. Линейное программированное обучение. Разработано Скиннером в 1954 году. При создании автор опирался на бихевеористскую психологию, в соответствии с которой обучение основано на принципе S – R, т.е. на появлении некоторых факторов (S-stimulus) и реакции на них (R-reaction). По этой концепции для любой реакции, соответственно усиленной, характерна склонность к повторению и закреплению. Поощрением для обучаемого является подтверждение программой каждого удачного шага, причем, учитывая простоту реакции, возможность совершения ошибки сводится к минимуму.

Линейная программа в понимании Скиннера характеризуется следующими особенностями:

* Дидактический материал делится на незначительные дозы, называемые шагами, которые обучаемые преодолевают значительно легко, шаг за шагом;
* Вопросы, содержащиеся в отдельных рамках программы, не должны быть очень трудными, чтобы обучаемые не потеряли интереса к работе;
* Обучаемые сами дают ответы на вопросы, привлекая для этого необходимую информацию;
* В ходе обучения учащихся сразу же информируют о том, правильны или ошибочны их ответы;
* Все обучаемые проходят по очереди все объем программы, но каждый делает это в удобном для него темпе;
* Во избежание механического запоминания одна и та же мысль повторяется в различных вариантах и разных местах программы.

2. Разветвленная программа. Автор концепции разветвленного программирования – А. Кроудер. Разветвленная программа основана на выборе одного правильного ответа из нескольких данных, она ориентирует текст многократного выбора. По мнению автора, выбор правильных ответов требует от обучаемых больших умственных способностей, нежели припоминание какой-то информации. Непосредственное подтверждение правильности ответа он считает своеобразным типом обратной связи.

Вопросы, в понимании Кроудера, нужны для достижения с следующих целей:

* Проверить, знает ли ученик материал;
* В случае отрицательного ответа отсылать обучаемого к координирующим и обосновывающим ответ порциям информации;
* Увеличение усилий обучаемого и одновременную ликвидацию механического обучения через многократное повторение информации;
* Формирование требуемой мотивации обучаемого.

Если основой линейной программы является стремление избежать ошибок, то разветвленная программа не направлена на ликвидацию ошибок в процессе обучения. Ошибки Кроудер трактует как возможность обнаружить недостатки в знаниях обучаемых, а также выяснить, какие проблемы обучаемые уяснили недостаточно. Благодаря такому методу о его программе можно сказать, что она сводится к «управлению процессом мышления», в то время как линейная программа основана на «управлении ответами».

Постепенно оба классических типа - линейное и разветвленное программированное обучение - уступили место смешанным формам.

Известно несколько видов организации программ наставнического типа, называемых также алгоритмами программированного обучения.

1. Последовательно-подготовительный алгоритм. Начальный элемент задания относительно прост, он подготавливает выполнение второго, более сложного, а тот, в свою очередь, третьего и т.д. Заключительные элементы имеют достаточно высокий уровень сложности.

1. Параллельно-подготовительный алгоритм. Начальные элементы заданий независимо один от другого подготавливают выполнение следующего за ним комплексного элемента высокого уровня.
2. Последовательно-корректирующий алгоритм. Начальные элементы задания имеют высокий уровень сложности, а каждый последующий элемент корректирует выполнение предыдущего, указывая, например, на противоречия, к которым приводят неправильные ответы.
3. Параллельно-корректирующий алгоритм. Обучаемому предлагается комплексный элемент высокого уровня, последующие элементы играют роль наводящих (подсказывающих), причем с разных позиций, независимо один от другого.
4. Алгоритм переноса. Приводятся два массива элементов А(N) и В(N). Ими могут быть понятия, отношения, действия, характеристики и т.д. Требуется установить логическое соответствие между ними.

6. Аналитический алгоритм. Предлагаются элементы А(N). Необходимо установить принадлежность каждого из них к одному из классов В(N).

7. Синтезирующий алгоритм. Элементы массива А(N) уже разбиты на подгруппы. Задача обучаемого - установить критерий, по которому осуществлялась классификация.

8. Алгоритм упорядочения. Элементы массива А(N) необходимо упорядочить по некоторому указанному критерию В(N). Этот алгоритм требует для своего выполнения комплексной умственной деятельности.

Большинство инструментальных систем предоставляют преподавателю возможность составлять обучающие и контролирующие задания с различными типами ответов.

3. С выборочным ответом. Обучаемому предлагается задание (вопрос) и набор (меню) готовых ответов, из которых он может сделать выбор правильного, по его мнению, ответа (утверждения).

Такой вариант задания наиболее удобен для машинной реализации, так как ЭВМ анализирует лишь номер, по которому легко определяет правильность ответа. На первый взгляд задания с выборочным ответом имеют ряд недостатков, а именно: обязательное предъявление верного ответа, возможность его угадывания, а значит, ограничение мыслительной деятельности обучаемого. Эти недостатки существенно снижаются путем правильного, творческого и остроумного применения различных принципов составления таких заданий.

Вероятность угадывания правильного ответа сводится к минимуму следующими простыми приемами:

* Повторением аналогичного по смыслу вопроса в нескольких различных формах;
* Увеличением числа элементов для выбора (при выборе из пяти ответов вероятность угадывания равна 0,2);
* Увеличение числа верных ответов до двух или до нескольких пар. Подбирать ответы в заданиях необходимо таким образом, чтобы они были правдоподобными и равнопривлекательными.

Формы заданий с выборочным ответом:

1. С частично-конструируемьш ответом. Задания этого типа являются промежуточным и связующим звеном между заданиями с выборочным ответом и свободно-конструируемым. Частично-конструируемый ответ составляется из частей, предложенных преподавателем.

Эта форма используется для заданий по составлению определений законов, теорем, стандартных формулировок и т.д. В верный ответ входят, как правило, не все элементы задания, и порядок их выбора не является жестким.

2. Со свободно-конструируемым ответом. Задания такого типа являются наиболее предпочтительными для автоматизированного обучения и контроля. Они позволяют слушателю общаться с компьютером на естественном языке, имитируя диалог обучаемого и преподавателя.

Задания со свободно-конструируемым ответом наиболее сложны для обучаемого, так как полностью исключают возможность угадывания и требуют значительной умственной работы перед вводом в компьютер ответа, набираемого на клавиатуре в свободной форме. В то же время резко возрастает сложность деятельности преподавателя - автора курса по формированию автономных ответов для анализатора инструментальной системы. Применяемый при оценке полученного ответа обучаемого на заданный вопрос эталон сравнивается с текстом и вырабатывается соответствующий признак ответа: «верный», «неверный», «предполагаемый» и т.д.

Таким образом, автор курса формирует шаблоны, предъявляемые обучаемому в зависимости от признака ответа, что создает иллюзию «понимания» системой смысла введенной фразы, так как при разных ответах на один и тот же вопрос обучаемый получает и различную реакцию компьютера.

В современных инструментальных системах реализованы следующие методы сравнения эталонного ответа с ответом обучаемого.

1. Анализ по ключевым словам. Этот метод анализа достаточно прост и универсален. Эталонный ответ, заранее введенный преподавателем, используется в качестве ключа, который сравнивается с ответом обучаемого на протяжении всей строки. Ключом может быть один символ, слово или группа слов.

При использовании ключевых слов можно достичь достаточно хороших результатов. Но применять метод надо достаточно осторожно, так как возможности распознавания смысла с его помощью ограничены и ответ не распознается при перестановках внутри ключа.

2. Синтаксический анализ с использованием символов частичной обработки ответа обучаемого. Этот метод анализа целесообразно использовать в том случае, когда требуется выполнить сравнение не по ключу, а по жесткому эталону. Выполняется посимвольное сравнение ответа с эталоном. При совпадении всех символов ответа с символами эталона вырабатывается признак «верно».

Однако при работе могут возникнуть ситуации, когда необходимо, с целью более корректного толкования смысла ответа, сделать некоторые отступления от правил прямого сравнения. В подобных ситуациях метод синтаксического анализа предусматривает средства частичной обработки ответов обучаемого.

Символы частичной обработки ответа (спецсимволы), включенные в эталон ответа, позволяют игнорировать один или несколько символов при сравнении с эталоном. Все остальные символы, отличные от символов частичной обработки, в тексте обучаемого должны следовать в том же порядке, что и в эталоне ответа.

3. Логический анализ. Логический метод анализа дает возможность формирования ответа в свободно-конструируемой форме. В данном случае ответ может представлять собой фразу или предложение, в котором порядок слов строго не определен. В словах могут игнорироваться окончание или другие части.

Основным отличием данного метода анализа от анализа по ключевым словам является то, что исключается необходимость перечисления всех возможных последовательностей ключевых слов при рассмотрении многословных ответов, так как логический метод позволяет с помощью одного эталона проанализировать насколько вариантов ответов. Недостатками такого рода программ являются:

* Снижение мотивации в ходе работы с программой;
* Возникновение пробелов в знаниях, связанных с непроизвольным рассеянием внимания в процессе работы с программой, а также ослаблением системного связывания знаний при отсутствии их интонационного выделения;
* Сложность и высокая трудоемкость организации учебного диалога, а также диагностирующей и управляющей обучением части программы.

Ввиду чрезвычайно высокой трудоемкости написания программ такого рода на языках программирования и высоких требований к программистской квалификации разработчиков, они часто разрабатываются с использованием программных оболочек автоматизированных учебных курсов, имеющих свой язык программирования, интерфейс, рассчитанный на разработчика-непрограммиста.

Недостатком таких программ является высокая трудоемкость разработки, затруднения организационного и методического характера при использовании в реальном учебном процессе школы. Организационные трудности связаны с тем, что такие программы невозможно использовать в структуре урока из-за больших различий в темпе обучения разных учащихся. Методические трудности заключаются в том, что многие педагоги нередко склонны не соглашаться с методическими решениями при изложении теоретического материала, предложенными разработчиками программы.

Программы 3-го типа (моделирующие) основаны на графически-иллюстративных возможностях компьютера, с одной стороны, и вычислительных, с другой, и позволяют осуществлять компьютерный эксперимент. Такие программы предоставляют ученику возможность наблюдать на экране дисплея некоторый процесс, влияя на его ход подачей команды с клавиатуры, меняющей значения параметров.

Программы 4-го типа (игры) предоставляют в распоряжение ученика некоторую воображаемую среду, набор каких-то возможностей и средств их реализации. Использование предоставляемых программой средств для реализации возможностей, связанных с изучением мира игры и деятельностью в этом мире, приводит к развитию обучаемого, формированию у него познавательных навыков, самостоятельному открытию им закономерностей, отношений объектов действительности, имеющих всеобщее значение.

Наибольшее распространение получили обучающие программы первых двух типов в связи с их относительно невысокой сложностью, возможностью унификации при разработке многих блоков программ.

Основные действия, выполняемые программами первых двух типов:

* Предъявление кадра с текстом и графическим изображением;
* Предъявление вопроса и меню вариантов ответа (или ожидание ввода открытого ответа);
* Анализ и оценка ответа;
* Предоставление кадра помощи при нажатии специальной клавиши.

Они могут быть запрограммированы, так что разработчику обучающей программы остается ввести в компьютер только соответствующий текст, варианты ответов, нарисовать на экране с помощью манипулятора «мышь» картинки. Создание обучающей программы в этом случае выполняется совершенно без программирования, не требует серьезных компьютерных познаний и по силам любому педагогу-предметнику средней школы. Наиболее известны отечественные автоматизированные системы обучения: «Урок», «Адонис», «Магистр», «Stratum». Используются в России и зарубежные системы: «Linkway», «Techcad» и др. Многие из этих систем имеют хорошие графические подсистемы и позволяют создавать не только статические картинки, но и динамические графические фрагменты в духе «мультимедиа».

Создание обучающей системы обычно проходит четыре стадии:

1. Разработка сценария обучающей программы: на этой стадии педагог должен принять решение о том, какой раздел какого учебного курса он будет переводить в обучающую программу, продумать материал информационных кадров, такие вопросы и варианты ответов к ним, чтобы они диагностировали трудности, с которыми будут сталкиваться ученики при освоении материала. Разработать схему прохождения программы, систему взаимосвязей между ее отдельными фрагментами.

2. Ввод в компьютер текстов отдельных кадров будущей программы, рисование картинок, формирование контролирующих фрагментов: вопросов, вариантов ответов к ним и способов анализа правильности ответов. На этой стадии педагогу потребуется минимальное владение функциями компьютера и возможностями ввода и редактирования, встроенными в инструментальную программу.

1. Связывание отдельных элементов обучающей программы в целостную диалоговую систему, установление взаимосвязей между фрагментами, вопросами и помощью, окончательная доводка программы.
2. Сопровождение программы во время ее эксплуатации, внесение в нее исправлений и дополнении, необходимость которых обнаруживается при ее использовании в реальном процессе обучения.

Очевидно, что создание обучающих программ средствами инструментальных систем поможет снять остроту главной проблемы компьютерного обучения - отсутствия в достаточном количестве и разнообразии качественных обучающих программ, так чтобы компьютерное обучение могло превратиться из жанра «показательных выступлений» на открытых уроках в действительно систематическое обучение учебным дисциплинам или их большим разделам.

В качестве первого шага к компьютерным технологиям обучения нужно рассматривать тренирующие и контролирующие программы. Использовать такие контролирующие программы можно систематически. Это не потребует кардинальных изменений в существующем учебном процессе и избавит учителя от непроизводительных, рутинных операций по проверке письменных работ, контролю знаний учащихся, решит проблему накопляемости оценок. Из-за тотальности контроля учащиеся получат мощный стимул к обучению.

Следующая проблема компьютерного обучения связана с тем, что использование компьютера не вписывается в стандартную классно-урочную систему. Важно, чтобы ученик при компьютерном обучении не был ограничен жесткими временными рамками, чтобы педагогу не надо было работать «на класс» в целом, а чтобы он мог пообщаться с каждым учеником, дать индивидуальную консультацию по работе с обучающей программой и по материалу, в ней содержащемуся, помочь преодолеть индивидуальные затруднения.

При проведении урока с использованием компьютеров работа педагога проходит фазы:

* Планирования урока (определяется место урока в системе занятий по данной дисциплине, время проведения в кабинете электронно-вычислительной техники, тип урока и его примерная структура, необходимые для его проведения программные средства);
* Подготовки программных средств (наполнение оболочек контролирующих программ и обучающих систем соответствующими дидактическими материалами, подбор моделирующих программ, размещение программных средств на носителях информации, проверка работоспособности программ);
* Проведения самого урока;
* Подведения итогов (внесение исправлений в обучающие программы, архивирование их для будущего использования, обработка результатов компьютерного тестирования, удаление лишних временных файлов).

Отдельное направление использования компьютера в обучении - интегрирование предметных учебных курсов и информатики. При этом компьютер используется уже не как средство обучения, а как средство обработки информации, получаемой при изучении традиционных дисциплин - математики, физики, химии, экологии, биологии, географии. Компьютер выступает при этом в качестве средства предметной деятельности.

Особые надежды при таком использовании компьютера возлагаются на компьютерные телекоммуникации, возможностями локальных и глобальных компьютерных сетей.

Такая технология включает следующие моменты:

* Первоначальную мотивацию исследования, постановку цели и задачи исследования;
* Поиск решения задачи, построение гипотез;
* Проведение исследований, экспериментов, наблюдений и измерений, с целью доказать или опровергнуть выдвинутые гипотезы;
* Групповое обсуждение результатов, составление отчета, проведение научной конференции;
* Решение вопроса о практическом применении результатов исследований, разработку и защиту итогового проекта по теме.

Использование компьютера очень хорошо вписывается в эту технологию обучения, особенно если имеется возможность реализовать компьютерные телекоммуникации. Телекоммуникационная составляющая проекта позволяет резко повысить интерес учащихся к выполнению проекта, делает естественным использование компьютера для представления результатов наблюдений и измерений, способствует формированию информационной культуры учащихся. Содержание обучения по методу проектов является межпредметным, интегрированным, привлекающим знания из различных областей. На практике позволяет достигать следующие педагогические цели:

* Развитие письменной речи;
* Овладение компьютерной грамотностью, освоение текстового редактора, компьютерных телекоммуникационных программ;
* Развитие общих навыков решения проблем;
* Развитие навыков работы в группе;
* Развитие навыков творческой работы.

**§3. Развитие глобальной сети Интернет**

Интернет – всемирная компьютерная сеть, объединяющая миллионы компьютеров в единую информационную систему. Интернет предоставляет широчайшие возможности свободного получения и распространения научной, деловой, познавательной и развлекательной информации. Интернет возник как воплощение двух идей – глобального хранилища информации и универсального средства ее распространения.

В 1960-х компьютерные сети стали бурно развиваться, но крупным недостатком больших сетей была их низкая устойчивость.

Поль Барен, Ларри Робертс и Винтсент Серф разработали и применили методы, ставшие основой дальнейшего развития сетевых технологий: пакетная коммутация, динамическая маршрутизация сообщений в распределенной сети, использование универсального сетевого протокола.

В 1969 была создана сеть ARPANET. Именно она стала основой современного интернета, который постепенно разросся до масштабов всей Земли.

В 1976 Серф разработал универсальный протокол передачи данных TCP/IP (Transmission control protocol/ Internet protocol). Он стал стандартом для межсетевых коммуникаций.

В 1990 Тим Бернерс - Ли создал систему, реализующую идею единого гипертекстового пространства. Для описания гипертекстовых страниц служил специальный язык HTML (HyperText Markup Language), а для их пересылке по сети – протокол передачи HTTP (HyperText Transfer Protocol). Новый способ указания адресов с помощью URL (Uniform Resource Locator – универсальный указатель ресурсов) позволял легче запоминать их и лучше ориентироваться в информационном пространстве Интернета. Была написана также специальная программа отображения гипертекстовых страниц – первый браузер (browser – обозреватель). Бернерс - Ли назвал свой проект WWW – World Wide Web, то есть «Всемирная паутина».

В 1992 был разработан графический браузер «Мозаика» и с учетом возросшей пропускной способности сетей появилась возможность быстро передавать цветные изображения, фотографии, рисунки.

В настоящее время Интернет развивается экспоненциально: каждые полтора - два года его основные количественные показатели удваиваются. Это относится к числу пользователей, числу подключенных компьютеров, объему информации и трафика, количеству информационных ресурсов.

Растут требования к оперативности и надежности информационных услуг, появляются новые их виды. Уже сейчас ученые разрабатывают принципиально новые формы глобальных информационных сетей. В недалеком будущем многие процессы сетевого проектирования, администрирования и обслуживания будут полностью автоматизированы.

**§4. Понятие web-сайта**

Web-сайт – это информация, представленная в определенном виде, которая располагается на Web-сервере и имеет свое имя (адрес). Для просмотра Web-сайтов на компьютере пользователя используются специальные программы, которые называются браузерами. Web-сайт состоит из связанных между собой Web-страниц. Web-страница представляет собой текстовый файл с расширением \*.html, который содержит текстовую информацию и специальные команды – HTML-коды, определяющие в каком виде эта информация будет отображаться в окне браузера. Вся графическая, аудио - и видео - информация непосредственно в Web-страницу не входит и представляет собой отдельные файлы с расширениями \*.gif, \*.jpg (графика), \*.mid, \*.mp3 (звук), \*.avi (видео). В HTML-коде страницы содержатся только указания на такие файлы.

Каждая страница Web-сайта также имеет свой Internet адрес, который состоит из адреса сайта и имени файла, соответствующего данной странице. Таким образом, Web-сайт – это информационный ресурс, состоящий из связанных между собой гипертекстовых документов (Web-страниц), размещенный на Web-сервере и имеющий индивидуальный адрес.

**§5. Средства разработки** **web-сайта**. **HTML**

Обмен информацией в Интернет осуществляется с помощью протоколов прикладного уровня и реализующих тот или иной прикладной сервис. Одним из популярных сервисов Интернет стала World Wide Web (WWW), основанная на протоколе HTTP (Hyper Text Transfer Protocol - протокол передачи гипертекстовой информации). В гипертекстовых документах, представленных в WWW пользователи, не ограничены одним документом, и более того, не ограничены одним компьютером.

Для подготовки гипертекстовых документов используется язык HTML, предоставляющий широкие возможности по форматированию и структурной разметке документов, организации связей между различными документами, средства включения графической и мультимедийной информации. Как уже было сказано выше, HTML-документы просматриваются с помощью специальной программы - браузера. Наибольшее распространение в настоящее время получили браузеры Opera, FireFox и Internet Explorer. Реализации этих браузеров доступны практически для всех современных программных и аппаратных платформ (Windows, Linux, Macintosh).

HTML-документ состоит из текста, представляющего собой содержание документа, и тегов, определяющих его структуру и внешний вид при отображении браузером. Простейший html-документ выглядит следующим образом:

<html>

<head>

<title>Название</title>

</head>

<body>

<p>Тело документа

</body>

</html>

Как видно из примера, тег представляет собой ключевое слово, заключенное в угловые скобки. Различают одинарные теги, как, например, <p>, и парные, как <body> </body>, в последнем случае действие тега распространяется только на текст между его открывающей и закрывающей скобкой. Теги также могут иметь параметры - например, при описании страницы можно задать цвет фона, цвет шрифта и т.д.:

<body bgcolor="white" text="black">.

Текст всего документа заключается в теги <html>, сам документ разбивается на две части - заголовок и тело. Заголовок описывается тегами <head>, в которые могут быть включены название документа (с помощью тегов <title>) и другие параметры, использующиеся браузером при отображении документа. Тело документа заключено в теги <body> и содержит собственно информацию, которую видит пользователь. При отсутствии тегов форматирования весь текст выводится в окно браузера сплошным потоком, переводы строк, пробелы и табуляции рассматриваются как пробельные символы, несколько пробельных символов, идущих подряд, заменяются на один. Для форматирования используются следующие основные теги:

<p> - начало нового абзаца, может иметь параметр, определяющий выравнивание:

<p align=right>;

<br> - перевод строки в пределах текущего абзаца;

<u></u> - выделение текста подчеркиванием

Ссылка на другой документ устанавливается с помощью тега

<a href="URL">...</a>,

где URL - полный или относительный адрес документа. При этом текст, заключенный в тег <a>, обычно выделяется подчеркиванием и цветом, и после щелчка мышью по этой ссылке браузер открывает документ, адрес которого указан в параметре href. Графические изображения вставляются в документ с помощью тега <img src="URL">.

**§6. PHP: Процессор гипертекста**

PHP – это язык программирования, созданный для генерации HTML‑страниц на web-сервере и работы с базами данных. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров. Входит в LAMP – «стандартный» набор для создания web-сайтов (Linux, Apache, MySQL, PHP (Python или Perl)).

В области программирования для Сети, PHP – один из популярнейших скриптовых языков (наряду с JSP, Perl и языками, используемыми в ASP.NET) благодаря своей простоте, скорости выполнения, богатой функциональности и распространению исходных кодов на основе лицензии PHP. PHP отличается наличием ядра и подключаемых модулей, «расширений»: для работы с базами данных, динамической графикой, криптографическими библиотеками, документами формата PDF и т.п. Интерпретатор PHP подключается к web-серверу либо через модуль, созданный специально для этого сервера (например, для Apache или IIS), либо в качестве CGI‑приложения.

В 1994 Расмус Лердорф написал набор скриптов на Perl/CGI обрабатывающий шаблоны HTML‑документов. Лердорф назвал набор Personal Home Page (Личная Домашняя Страница) а затем написал на языке C новый интерпретатор шаблонов PHP/FI (англ. Personal Home Page / Forms Interpreter – «Личная Домашняя Страница / Интерпретатор форм». PHP/FI включал базовую функциональность сегодняшнего PHP: оформление переменных в стиле Perl ($имя\_переменной для вывода значения), автоматическую обработку форм и встраиваемость в HTML‑текст и многое другое.

В 1997 году Энди Гутманс и Зив Сураски сочли PHP/FI 2.0 непригодным и объявили PHP 3.0 официальным преемником PHP/FI.

Одной из сильнейших сторон PHP 3.0 была возможность расширения ядра, что дало PHP возможность работать с огромным количеством баз данных, протоколов, поддерживать большое число API .Был разработан более мощный и полный синтаксис.

Абсолютно новый язык программирования получил новое имя. Разработчики отказались от дополнения о персональном использовании, которое имелось в аббревиатуре PHP/FI. Язык был назван просто PHP (Препроцессор Гипертекста).

К зиме 1998 года, практически сразу после официального выхода PHP 3.0, Энди Гутманс и Зив Сураски начали переработку ядра PHP. В задачи входило увеличение производительности сложных приложений и улучшение модульности базиса кода PHP. Расширения дали PHP 3.0 возможность успешно работать с набором баз данных и поддерживать большое количество различных протоколов, но PHP 3.0 не имел качественной поддержки модулей и приложения работали неэффективно.

В1999 был разработан PHP 4.0, основанный на движке Zend Engine и принёсший с собой набор дополнительных функций. В дополнение к улучшению производительности, PHP 4.0 имел ещё несколько ключевых нововведений, таких как поддержка сессий, буферизация вывода, более безопасные способы обработки вводимой пользователем информации и несколько новых языковых конструкций.

Пятая версия PHP была выпущена через 5 лет, в 2004 году. Изменения включали обновление ядра Zend (Zend Engine 2), что существенно увеличило эффективность интерпретатора. Была введена поддержка языка разметки XML, полностью переработаны различные функции, которые стали во многом схожи с моделью, используемой в Java.

Шестая версия PHP находится в стадии разработки с октября 2006 года. В ней уже сделано множество нововведений, как, например, исключение из ядра регулярных выражений POSIX и «длинных» суперглобальных массивов, удаление директив safe\_mode, php\_magic\_quotes и register\_globals из конфигурационного файла php.ini. Также много внимания уделено поддержке Юникода.

**§7. СУБД MySQL**

Программное обеспечение MySQL представляет собой очень быстрый многопоточный, многопользовательский надежный SQL‑сервер баз данных (SQL – язык структурированных запросов). Сервер MySQL предназначен как для критических по задачам производственных систем с большой нагрузкой, так и для встраивания в программное обеспечение массового распространения.

MySQL является решением для малых и средних приложений. Входит в LAMP. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Благодаря открытой архитектуре и GPL‑лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

В 2008 был выработан новый SQL‑интерфейс, но API‑интерфейс остался в наследство от mSQL.

MySQL портирована на большое количество платформ: AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, OpenBSD, OS/2 Warp, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Server 2003 и Windows Vista. Существует также порт MySQL к OpenVMS.

MySQL имеет API для языков C, C++, Java, Perl, PHP, Python, Ruby, Smalltalk и Tcl, библиотеки для языков платформы.NET, а также обеспечивает поддержку для ODBC посредством ODBC‑драйвера MyODBC.

MySQL 5.0 содержит следующие нововведения: хранимые процедуры и функции; обработчики ошибок; курсоры; триггеры; представления; информационная схема (так называемый системный словарь, содержащий метаданные).

**§8. Системы управления контентом**

В последние годы стали популярны системы управления контентом. С помощью подобных систем легко создать базовый web-проект, а web-интегратор получает возможность развернуть масштабный проект за счет модульности.

Применение системы управления контентом при разработке web-проекта позволило не работать над созданием кода разметки каждой страницы, программированием и интегрированием их графического оформления. Достаточно выбрать готовый модуль, из ранее созданных и протестированных. Интеграция в систему делается по единому стандарту.

Начало развитие web-среды происходит в 90‑ых гг. Этот период характеризовался относительно невысоким уровнем развития web технологий и web-проекты способные на интерактивное взаимодействие могли создаваться только программистами.

Вся динамика реализовывалась через CGI и другие сложные технологии и сайт обычно представлял собой набор статичных html‑страниц, подготовленных в html редакторах. После набора, страницы, для осуществления возможности межстраничных переходов, объединялись ссылками и размещались на сервере. Вся работа по обновлению информации и проверке работоспособности проекта перекладывалась на его автора. Если требовалось не просто изменить существующую страницу, а добавить новую, то приходилось решать вопросы, связанные с логическим и физическим внедрением последней в весь проект. И если объем страниц возрастал, то «справляться» с ними становилось еще сложнее. Изменять и расширять динамические проекты было еще более проблематично.

Систем автоматизации всех этих процессов на тот момент практически не было. Однако такое положение длилось относительно недолго, и в дополнение к статичным html‑страницами, CGI программированию появились более «дружественные» технологии – Asp (конец 1996 г.), ColdFusion (июнь 1995 г.), а позже и PHP. Новые технологии позволили совместить разметку html‑страниц и несложный программный код, сделав тем самым пассивные html‑страницы активными. Активность страниц позволила организовать интерактивное взаимодействие с пользователями и при обращении к одной и той же активной странице пользователь мог получать новые данные. В то же время стали создаваться активные страницы для автоматизации определенных действий, например, процесса загрузки файлов на сервер или же процесса создания новых страниц. Такие полезные страницы постепенно собирались во вспомогательные пакеты. Подобные пакеты применялись в типовых задачах. Вскоре разработчики пришли к решению о необходимости создания универсальных систем. Таким образом и появились первые универсальные коммерческие системы управления. Утвердилась и аббревиатура CMS. Позже появились и открытые системы управления. Дальнейшая эволюция этих систем привела к расширению функциональности и теперь они способны работать с темами, модулями, а также управлять другими элементами. Современные системы уже можно смело именовать «системами управления web-проектом».

Сейчас системы управления это не только удобная оболочка-менеджер для пользователя, но и мощный инструмент для web-разработчика (последнее справедливо далеко не для всех систем). Благодаря таким системам, все реже возникает необходимость в разработке web-проектов «с нуля» – подготовленному пользователю достаточно выбрать, установить и настроить существующую систему, чтобы получить приемлемый результат профессионального уровня.

К сожалению, некоторая часть существующих web-сайтов создана без систем управления. Это так называемые «пустышки».

Создателю web-проекта при необходимости внесения корректировок или изменений в проект приходится самостоятельно выполнять сложные манипуляции с гипертекстом и программным кодом, либо потребуется обратиться к разработчикам проекта и за дополнительную оплату решать вопросы с которыми он и сам бы мог справиться, имея систему управления.

По объективным оценкам на данный момент существует пока немалая часть web-проектов без систем управления.

Школьнику безразлично, на базе какой системы управления функционирует разрабатываемый проект, ему важен результат – информация и способ ее получения. Система управления для школьника – это страницы с материалами, навигационные элементы, формы обратной связи, подсказки и т.д., но не более того. Поэтому недостаточно создать удобный и хорошо настроенный проект, необходимо также позаботиться и о его содержании – оно должно быть для школьника интересным и актуальным.

Преподаватели, модераторы, администраторы – это лица, которым делегирован набор дополнительных прав. Пользователям такого уровня система управления представляется в расширенном виде – к интерфейсу системы добавляются дополнительные элементы управления или целые административные разделы. В таких разделах администратор может легко и быстро добавлять например новых пользователей, а модератор может вносить правки в оставленные посетителями сообщения и делать другие разрешенные администратором действия.

Для лучшего понимания предметной области, а также установления порядка, была создана такая методика как классификация. Классификация – это разделение общего множества на частные по выбранным критериям, т.е. попросту деление на группы по общим признакам. Классификация также требуется и в области систем управления, где она должна сделать четкие разграничения между системами.

На данном этапе будут рассмотрены два наиболее очевидных вида классификации:

* классификация по «степени открытости»;
* классификация по «разработчику системы».

а) классификация по «степени открытости»

Многие пользователи уже успели попробовать такие «открытые» продукты как OpenOffice, Firefox, Linux, а также другие известные программы и системы, которые распространяются не просто бесплатно, но и с открытыми исходными кодами. Последнее означает, что каждый желающий может вносить изменения в исходный код таких программ и даже распространять их в модифицированном виде.

Главным преимуществом открытых продуктов является их доступность – открытый продукт можно не только свободно получать, но и быстро обновлять, ведь новые версии таких продуктов выходят с завидной регулярностью, а исправление найденных ошибок не выливается в очередное ожидание заплаток или сервиспаков, как это обычно происходит с коммерческими решениями.

Открытый продукт можно «пересобрать» с учетом своих собственных требований и тем самым получить максимальный результат.

б) классификация по «разработчику системы»

Разработчик системы, как правило, является автором, и сопроводителем системы. Современная CMS система должна позволять, не прибегая к дополнительному программированию, выполнять: редактировать содержимое страниц, включая добавление / удаление графики; добавление новых страниц; изменение структуры сайта и различных мета данных; настройку регистрационных форм; управление опросами, голосованиями и форумами; вывод статистики посещений; распределение прав по управлению сайтом среди пользователей т.д.

**§9. CMS «Joomla»**

«Joomla» – это система управления контентом, написанная на языке PHP и использующая в качестве хранилища содержания базу данных MySQL. «Joomla» является свободным программным обеспечением, защищённым лицензией GPL. Одной из главных особенностей «Joomla» является относительная простота управления при практически безграничных возможностях и гибкости при изготовлении сайтов. Название «Joomla» фонетически идентично слову «Jumla», которое в переводе означает «все вместе» или «единое целое», что отражает подход разработчиков и сообщества к развитию системы. Это название было выбрано сообществом среди многих, прошло проверку профессионалами маркетинга, и было решено, что «Joomla» – это самый удачный выбор. Административная панель «Joomla» представлена на рисунке.

Система управления содержанием «Joomla» является ответвлением широко известной CMS Mambo. Команда независимых разработчиков отделилась от проекта Mambo по причине несогласия в экономической политике. 16 сентября 2005 года в свет вышла первая версия «Joomla», являющаяся по сути переименованной Mambo и включающая в себя исправления найденных на тот момент ошибок и уязвимостей.

CMS «Joomla» включает в себя различные инструменты для изготовления web-сайта. Важной особенностью системы является минимальный набор инструментов при начальной установке, который обогащается по мере необходимости. Это снижает загромождение административной панели ненужными элементами, а также снижает нагрузку на сервер и экономит пространство на хостинге.

Достоинства «Joomla»:

* «Joomla» бесплатна.
* «Joomla» развивается профессионалами уже больше 5 лет. Т.е. можно быть уверенным что система будет развиваться и дальше, а сайты написанные на ней будут и дальше актуальны.
* Система работает с различными расширениями и модулями. Изначально в нее заложена только базовые компоненты, все остальное можно установить исходя из направленности сайта.
* Более 4000 бесплатных и платных расширений.
* Система реализована по блочному принципу. У каждого созданного сайта на «Joomla» существует своя шапка, левый и правый блоки, меню и т.д. Такое построение упрощает понимание системы.
* «Joomla» позволяет создавать сайты любой сложности.
* «Joomla» имеет много платных и бесплатных шаблонов которые легко модифицируются. Недостатки «Joomla»:
* «Joomla» не очень быстра на локальном сервере.
* В 2008 году система перешла на новую ступень развития (версия 1.5.х), поэтому часть дополнений, сделанных под версию 1.0.х стали недоступными.

**§10. Web-сервер**

Web-сервер – это сервер, принимающий HTTP‑запросы от клиентов. Обычно в роли клиентов выступают web-браузеры, выдающие им HTTP‑ответы, вместе с которыми передаются HTML‑страницы, изображения, файлы, медиа-потоки или другие данные. Web -серверы являются основой Всемирной паутины.

Фактически web-сервером называют как программное обеспечение обеспечивающее все вышеперечисленные действия, так и компьютер, на котором это программное обеспечение работает. Клиенты получают доступ к web-серверу посредством единого указателя ресурсов – URL‑адресу нужной им web-страницы или другого ресурса располагаемого на сервере.

Изначально URL предназначался для обозначения мест расположения ресурсов (чаще всего файлов) во Всемирной паутине.

Сейчас URL позиционируется как часть более общей системы идентификации ресурсов URI, сам термин URL постепенно принимает более широкое значение.

Среди web-серверов самыми популярными являются: Apache HTTP-Server от компании Apache и IIS от компании Microsoft.

Появление Apache HTTP Server явилось одним из важнейших средств, стимулировавших развитие всемирной сети Интернет. Apache был первой бесплатной альтернативой web-серверу от Netscape Communications (в настоящее время известен как Sun Java System Web Server). Приложение доступно для установки на многих платформах, таких как: Unix, FreeBSD, Linux, Solaris, Novell NetWare, Mac OS X, Microsoft Windows и т.д. Apache распространяется как бесплатное программное обеспечение с открытым исходным кодом, что позволяет множеству разработчиков со всего мира участвовать в процессе доработки и выпускать дополнительные надстройки, выполняющие специфические, нужные для определенных целей, функции.

Первая версия была выпущена в 1994 году. Вторая версия заметно отличалась от предшествующей. Она была выпущена только в 2002 году. Имея модульную структуру, новый API для разработчиков, поддержку IPv6 и улучшенную поддержку не-Unix платформ (например Windows) она стала самым используемым http‑сервером в сети Интернет. Web-сервер Apache имеет модульную структуру ядра, что помогает разработчику, определившись с целями проекта, использовать именно те средства разработки, которые он планирует, отключив все остальные возможности. Данная архитектура ядра делает программный продукт более безопасным и быстрым. Web-сервер поддерживает различные модули интерпретаторов, например: perl, python, php, tcl. В Apache встроена поддержка аутентификации, реализован протокол шифрования SSL, имеется прокси-модуль и модуль создающий отчеты. Кроме этого в возможности web-сервера входит подключение внешнего модуля сжатия gzip.

Удобная конфигурация позволяет администратору использовать один web-сервер для обслуживания нескольких узлов одновременно. В апреле 1996 Apache стал самым популярным HTTP‑сервером во Всемирной сети, но к ноябрю 2006 г. доля web-серверов работающих на ПО от Apache Group значительно снизилась, почти половина всех web-ресурсов перешла на сервер Microsoft Internet Information Services или сторонние разработки. Именно Microsoft Internet Information Services является вторым по полярности сервером в глобальной сети интернет.

В основе выпуска IIS 7.0 лежит полностью модульный web-сервер, включающий более 40 компонентов, которые можно объединять в компактные web-серверы, оптимизированные для необходимой роли в топологии приложения. Эти компоненты создаются на основе нового слоя расширяемости, что позволяет разработчикам расширять или замещать практически любую функцию сервера в машинном коде.

IIS 7.0 предлагает расширяемость компонентов выполнения, управления и рабочих компонентов, облегчая создание комплексных решений в соответствии с конкретными потребностями. Диспетчер служб IIS 7 представлен на рисунке .

В платформе IIS 7.0 по сравнению с IIS 6 решаются многие проблемы, связанные с управляемостью и эксплуатацией сервера. Она обладает принципиально новой системой настройки, обеспечивающей полностью делегированное управление узлами. Новые интерфейсы API для управления и диагностические компоненты делают процедуры развертывания, администрирования и устранения неполадок сервера значительно проще и удобнее, чем когда-либо прежде.

Web-сервер IIS поддерживает несколько различных технологий создания web-приложений:

* ASP.NET – для систем Windows это основное, на сегодняшний день, средство создания web-приложений и web-служб. Поддержка ASP.NET встроена в IIS 6.0. Для более ранних версий необходимо отдельно загрузить и установить NET Framework.
* ASP – устаревшая технология создания динамических web-страниц на основе сценариев. Входит в поставку IIS начиная с версии 3.0.
* CGI – стандартная технология создания динамических web -страниц.
* ISAPI – для Windows систем это наиболее мощная технология, предоставляющая полный доступ ко всем возможностям IIS.

При помощи CGI и ISAPI к web-серверу IIS могут подключаться сторонние средства поддержки web-приложений, например, PHP и Perl.

Одним из интересных и перспективных нововведений в IIS 7.0 является пакет IIS Media Pack. Два дополнительных бесплатных модуля позволят превратить web-сервер в современный инструмент медиа-вещания. Новые технологии Microsoft позволяют оптимизировать и грамотно управлять цифровым потоком медиа-данных. Сервер позволяет производить вещание данных в форматах: ASF, AVI, FLV, M4V, MOV, MP3, MP4, RM, RMVB, WMA, WMV.

Еще одним нововведением является встроенная поддержка новой технологии Silverlight, Silverlight 2 и Silverlight 3. Это новая технология представления данных в Интернете. Предназначена для запуска на различных платформах. Она позволяет создавать насыщенные, визуально привлекательные web-страницы, работающие в различных обозревателях, устройствах и настольных операционных системах (например Apple Macintosh).

**§11. Принципы построения клиент / серверных систем**

компьютерный обучающий интернет joomla

Для разработки клиент / серверных систем имеется два подхода:

* построение систем на основе двухзвенной архитектуры;
* построение систем на основе трехзвенной архитектуры.

Двухзвенная архитектура состоит из клиентской и серверной части. Как правило, серверная часть представляет собой сервер БД, на котором расположены общие данные. А клиентская часть представляет приложение, которое связывается с сервером БД, осуществляет к нему запросы и получает ответы. Такие системы используются в локальных сетях, т. к. нет затруднений с установкой клиентской части. Также системы с такой архитектурой более безопасны, т. к. могут использовать собственные протоколы передачи данных, не известные злоумышленникам.

При разработке информационных систем, рассчитанных на широкую аудиторию, возникают проблемы с использованием двухзвенной архитектуры. Во-первых, пользователю необходимо иметь в наличии клиентскую часть, а, во-вторых, у неопытного пользователя, могут возникнуть проблемы с конфигурированием такой системы. Поэтому в последнее время, более часто разрабатывают приложения на базе трехзвенной архитектуры.

Трезвенная архитектура также состоит из двух частей: клиента и сервера. Но серверная часть в этой архитектуре представляет собой сервер приложений и сервер БД. А в качестве клиента выступает web‑браузер. Такая система очень проста для пользователя. Клиенту необходимо знать только адрес сервера и иметь web‑браузер на компьютере. Все данные представляются в виде html‑разметки с использование графики (jpeg, gif, flash), каскадных слоев CSS и JavaScript. Недостатками такой архитектуры является использование общеизвестных протоколов передачи данных. Злоумышленник может осуществить взлом системы, если она будет не достаточно хорошо проверять поступившие запросы от клиента.

**§12. Вывод**

Для создания эффективного справочного пособия необходимо иметь высокий уровень знаний и большой опыт работы с автоматизированными обучающими ресурсами. Разработчик такой обучающей системы должен не только знать педагогические приемы применения данного пособия в школе и уметь понятно, наглядно представлять новый материал учащимся, но и хорошо разбираться в основных особенностях построения образовательного ресурса, знать основные принципы новых информационных технологий обучения, типы обучающих систем и т.д. При создании любой обучающий автоматизированной системы разработчик должен провести предварительную работу по изучению уже существующих автоматизированных систем, понять основные принципы их построения, учесть все положительные и отрицательные стороны, присутствующие в подобных обучающих системах.

При создании справочного пособия по HTML автор непременно должен пытаться создать ресурс, направленный на помощь учащимся в освоении темы «Основные понятия языка HTML», и на повышение эффективности обучения учащихся при изучении этой темы. Разработчик должен знать, какие цели он преследует, используя в школе данное справочное пособие, и исходя из них верно выбрать тип обучающей системы и правильно ее организовать.

**Глава II. Практическая часть**

**§1. Загрузка последней версии CMS «Joomla»**

При работе с любым программным обеспечением, очень важно устанавливать и использовать только последнюю версию. Зачастую, разработчики вносят большое количество изменений уже после выпуска финальной версии продукта. Для загрузки последней версии необходимо обратиться к любому русскоязычному сайту о «Joomla», например http://www.joom.ru/. Используя раздел «Скачать», выбрать последний дистрибутив и загрузить его на свой компьютер. На данный момент последней русскоязычной версией «Joomla», является версия CMS Joomla 1.5.15 RE (Russian Edition).

Данная версия включает локализованные демо-данные, языковые файлы «Joomla» и визуального редактора. Установка CMS происходит на русском языке, являющемся языком по умолчанию сразу после установки «Joomla» RE.

### 

**§2. Установка локального сервера**

### **Локальный сервер** - специальный набор скриптов, который устанавливается на компьютер **и** позволяет web-мастерам разрабатывать, создавать динамические, CMS, PHP и другие сложные сайты, на локальном (домашнем) компьютере, без использования Интернет. Т.е. это своего рода имитатор интернет сервера.

Необходимость в локальном сервере возникает именно при разработке сложных, динамических, т.е. сайтов, использующие php (или perl) – скрипты, базы данных (mysql) или другие сценарии. Для обычных html+css сайтов локальный сервер в принципе не нужен, но для удобства и порядка, в своих разработках, их тоже лучше хранить в такой программе.

Форумы, блоги, гостевые книги, большинство форм обратной связи, CMS да и просто php-скрипты, – все нужно тестировать, проверять, корректировать на правильную работоспособность, а этого можно добиться только с помощью локального сервера.

Локальный сервер позволяет обработать сценарии и возвратить результат в HTML формат, что дает браузерам считывать код и показывать визуальную картинку.

Такой обработчик имеется на каждом **интернет**-сервере, предоставляющем услуги хостинга. К сожалению, при установке операционной системы Windows на домашний компьютер, установка такого обработчика не предусмотрена, поэтому для работы с локальным сервером приходится пользовать сторонние дистрибутивы.

Существуют несколько локальных серверов, но на мой взгляд самый простой и практичный – это «Джентльменский набор Web-разработчика» («Д.н.w.р», читается «Денвер»).

Денвер - набор дистрибутивов (Apache, PHP, MySQL, Perl и т.д.) и программная оболочка, используемая Web-разработчиками для разработки сайтов на «домашней» (локальной) Windows - машине без необходимости выхода в Интернет. Главная особенность Денвера - удобство при удаленной работе сразу над несколькими независимыми проектами и возможность размещения на Flash-накопителе. При обычной установке по умолчанию, Денвер имеет нечто вроде «ядра» - так называемый «базовый пакет», занимающий около 5.5 MB. Все остальное поставляется в виде пакетов расширений. Базовый пакет содержит большинство необходимых программ и утилит:

* Инсталлятор (поддерживается также инсталляция на flash-накопитель).
* Apache, SSL, SSI, mod\_rewrite, mod\_php.
* PHP5 с поддержкой GD, MySQL, sqLite.
* MySQL5 с поддержкой транзакций.
* Система управления виртуальными хостами, основанная на шаблонах.

Чтобы создать новый хост, вам нужно лишь добавить директорию в каталог /home, править конфигурационные файлы не требуется. По умолчанию уже поддерживаются схемы именования директорий многих популярных хостеров, схемы также можно добавлять в ручную.

Денвер также в себя включает:

* Систему управления запуском и завершением всех компонентов Денвера.
* phpMyAdmin — система управления MySQL через Web-интерфейс.
* Эмулятор sendmail и SMTP-сервера (на localhost могут приходить письма в /tmp в формате .eml).

Скачать последнюю версию Денвер можно с его официального сайта http://www.denwer.ru/.

После скачивания извлеките и запустите установочный файл.

Нажмите «Да».

После запуска установщика появляется чёрное окно и запускается браузер, где идёт описание Денвера.

Закройте браузер.

В появившемся чёрном окне нажмите кнопку «Enter», для продолжения установки.

У вас появится следующее диалоговое окно:

Вам предлагается создать директорию, и указать диск на которой Вы намерены установить Денвер. Я предлагаю указать диск «**D»**, в этом случае при возникновении каких либо проблем с Windows, Денвер не пострадает. Название директории выбираем произвольно.

Таким образом, пишем D:\Joomla и нажимаем «**Enter»**.

Вас спрашивают – действительно ли Вы желаете установить Денвер на локальный диск «**D»**? Вы должны согласиться, нажать клавиши «**Y»** и «**Enter»**.

Нажмите «Enter**».**

Далее требуется указать букву, под которой будет создан виртуальный диск для сервера Денвер. Нажмите «Enter**».**

Снова нажимаем клавишу «Enter**».**

Выберите вариант **1** и нажмите на «Enter**».**

Согласитесь с созданием ярлыка на рабочем столе, нажатием клавиш «Y**»** и «Enter**».**

Денвер установлен. На рабочем столе должны появиться 3 ярлыка. Запустите ярлык «Start Denver**»** для того что бы Денвер загрузился и начал корректно работать.

Теперь можно открыть браузер и в адресной строке набрать«**localhost».**

Если у Вас появилась эта страничка, это означается, что Денвер успешно установился и локальный сервер готов к работе. Все, установка локального сервера завершена!

### 

**§3. Создание базы данных MySQL для работы с CMS «Joomla»**

Денвер установлен и готов к работе, об этом свидетельствуют три новых значка на рабочем столе вашего компьютера. А два значка появившиеся на панели задач, говорят о том что локальный сервер запущен. Следующим шагом будет создание базы данных для сайта на «Joomla». Добавление пользователя и создание его конфигурации.



Что бы создать базу данный для нашего сайта, нужно в адресной строке браузера набрать адрес localhost/Tools. У Вас появится следующее окно:

В открывшемся меню выбираем пункт **pnpmyadmin/. Это программа, которая позволит создать базу данных для нашего сайта и в дальнейшем управлять ею.**

**От Вас требуется создать базу данных. Назовем ее site и нажмем на кнопку** «**Создать».**

**База данных создана. Теперь нужно нажать на кнопку и зайти в пункт меню** «П**ривилегии».**



**Вам нужно нажать на кнопку** «**Добавить нового пользователя».**

**Заполнить пустые поля следующим образом:**

Пароль придумывается Вами самостоятельно. Далее н**ажмите на кнопку «Пошел».**

**Должна появиться надпись «Был добавлен новый пользователь». В результате этих операций мы создали базу данный «site» и пользователя «site» который имеет права администратора. Следующий этап создания нашего справочного пособия, это установка самой «Joomla» на локальный компьютер.**

**§4. Установки CMS «Joomla»**

После того как на компьютере был установлен локальный сервер и создана база данных, можно приступать непосредственно к установке сомой «Joomla». C сайта http://www.joom.ru/ нужно скачать последнюю русскоязычную версию Джумлы. На данный момент это версия CMS Joomla! 1.5.15 RE (Russian Edition).

Запускаем сервер, заходим на вновь появившейся диск Z, здесь нам необходима директория «**www»** для этого заходим в: **\home\localhost\www\** . В открывшемся каталоге создаём новую паку с названием «site». В неё мы и должны распаковать файлы из архива CMS Joomla! 1.5.15 RE (Russian Edition).

«Joomla» распакована, теперь перезапускаем Денвер, для того чтобы север увидел новый каталог «site». Открываем браузер и в адресной строке набираем **http://localhost/site**

Запустился установщик Джумлы.

Важно: В первую очередь в браузере должны быть включены cookies.

* Выберем язык и нажимаем кнопку – «Далее»;
* Окно yачальная проверка пропускаем, нажимаем кнопку – «Далее»;
* Лицензия – читаем, нажимаем кнопку – «Далее».

Окно: «Конфигурация базы данных», здесь нам необходимо заполнить пустые поля, уже имеющимися у Вас данными.

Окно «**Конфигурация FTP»** пропускаем, т.к. при отправке файлов на сервер мы будем использовать сторонний FTP клиент. Нажимаем «**Далее»**.

На странице **«Главная конфигурация»**, необходимо заполнить все пустые поля и нажать на кнопку «Установить демо-данные». Пароль администратора придумывается Вами самостоятельно. Он нужен для входа в «административный раздел» «Joomla».

Нажимаем «Далее» и видим окно завершения.

Здесь нам предлагается удалить папку **installation.** Это делается в целях безопасности, для избегания повторной установки «Joomla» на локальный сервер. Путь к папке **Z:\home\localhost\www\site\installation**. **Установка** «Joomla» **закончена**. Теперь набираем в адресной строке браузера адрес нашего будущего справочного пособия: **localhost/site** если всё прошло нормально должен загрузится стандартный шаблон «Joomla». Установка «Joomla» прошла успешно!

## 

**§5. Настройка системы управления контентом «Joomla».** **Конфигурация системы**

Конфигурация представляет набор условий и свойств, которые выполняет система. Для ее настройки необходимо выполнить процедуру авторизации и зайти в административный интерфейс управления по адресу http://localhost/site/administrator/.

В качестве имени пользователя и пароля вводим учетные данные, указанные при установке. В данном случае, именем пользователя является «admin», а паролем – «123456».

После ввода правильной комбинации имени пользователя и пароля, произойдет перенаправление на главную страницу административного раздела.

Система управления контентом обладает широким набором функций:

* создание, редактирование и публикация информационного материала;
* конфигурирование системы управления контентом;
* установка тем оформления, компонентов и модулей;
* статистика посещений, просмотров, пользователей;
* информация о системе, обновлениях и исправлениях.

Рассмотрим панель управления более подробно. В самом верху слева размещен логотип системы, а справа номер текущей версии.

Ниже размещена главная строка меню, которая состоит из различных элементов. Справа в этой строке размещены четыре элемента:

* ссылка для перехода на Ваш web-сайт (Front End);
* информация о количестве сообщений, присланных Вам;
* информация о количестве пользователей он-лайн на сайте;
* кнопка выхода из панели администрирования.

Слева размещены семь пунктов меню, которые содержат в свою очередь определенное количество подпунктов.

Рассмотрим по порядку данные пункты слева на право:

Вкладка «Сайт» - меню управления, с помощью которого можно управлять общими параметрами конфигурации сайта, пользователями и медиа -файлами.

Вкладка «Все меню» - менеджер, с помощью которого осуществляется управление всеми видами меню на сайте.

Вкладка «Материалы» - меню управления информационным наполнением сайта. Здесь мы можем создавать разделы, категории, статьи, а также управлять ними.

Вкладка «Компоненты» - меню управления основными составляющими системы, которые расширяют функциональность «Joomla». Среди них, такие как менеджер баннеров, контактов, лент новостей (RSS каналов), компонент организации голосований, поиска и компонент организации ссылок. Здесь упомянуты те компоненты, которые устанавливаются вместе с установкой системы. Но «Joomla» дает возможность устанавливать и другие компоненты. А их очень много в зависимости от исполняемых функций, что в свою очередь дает возможность настраивать данную систему под различные требования для того или иного сайта.

Вкладка «Расширения» - данное меню управляет дополнительными функциями CSM, а именно здесь есть возможность перейти в менеджер модулей, шаблонов, плагинов, языков. Здесь также есть возможность перейти в менеджер установки/удаления всех этих компонентов.

Вкладка «Инструменты» - через данное меню мы можем читать служебные сообщения системы, обмениваться сообщениями с другими участниками, зарегистрировавшимися на сайте, рассылать электронные письма всем пользователям системы, а также снимать блокировку с заблокированных элементов системы и чистить кеш (временные файлы).

Вкладка «Помощь» - через данное меню администратор может перейти к сайту помощи «Joomla», а также с помощью подменю «информация о системе» перейти в управление всей служебной информацией сайта (параметры базы данных, параметрах версии PHP, установленных правах на папки системы, конфигурационный файл и другое).

Ниже этих пунктов меню слева размещена секция, в которой находятся наиболее распространенные ссылки на различные области управления «Joomla».

Справа расположена наиболее востребованная информация системы (популярные статьи, новые статьи на сайте, список зарегистрированных пользователей в он-лайн, информация о пунктах меню). Данными блоками информации администратор может управлять и настраивать под свои нужды.

**§6. Шаблоны дизайна**

В «Joomla» четко разделено содержимое и представление содержимого в виде шаблона сайта, включающего в себя цвета, шрифты, выравнивание, расположение и т.д. Сайту на «Joomla» можно назначать различные шаблоны.

Создание шаблона технически несложно, но требует знаний языков HTML и PHP. Можно воспользоваться доступными, свободно распространяемыми шаблонами или услугами профессиональных дизайнеров.

Одним из первых действий по настройке «Joomla» является установка шаблона сайта. Стандартная тема оформления «Joomla» подходит для демонстрации возможностей системы, но никак не для создания полноценного web-сайта.

Следует помнить, что для «Joomla» 1.5.x подходят шаблоны разработанные конкретно под версию 1.5.x. Шаблоны, созданные для версии 1.0.x, на 1.5.x будут отображаться некорректно или не будут работать вообще. Сам шаблон сайта для «Joomla» обычно распространяется (так же как и расширения) в виде.zip файлов (или также.tar.gz файла). Для использования таких файлов необходимо сохранить файл шаблона на компьютер и далее перейти в «Расширения» – «Установить /удалить»

На странице «Установить /удалить» нужно выбрать .zip файл для установки. Нажмите кнопку «Обзор» для выбора файла шаблона сайта и нажмите кнопку «Загрузить файл и установить».

Установка шаблона не приведет к мгновенной смене шаблона сайта. Для смены шаблона сайта, необходимо выбрать нужный шаблон и установить его по умолчанию.

Чтобы начать использовать шаблон сайта, необходимо перейти на страницу «Расширения» – «Менеджер шаблонов».

Выбрать в панели «Менеджер шаблонов» интересующий нас шаблон и нажать на копку «По умолчанию», что бы напротив выбранного шаблона появилась звездочка. Это означает, что шаблон выбран по умолчанию и будет использоваться для отображения сайта автоматичеки.

**§7. Модули и компоненты**

Система управления контентом «Joomla» имеет модульную структуру и в своем базовом варианте не обеспечивает всех необходимых возможностей, требуемых для реализации готового проекта.

В «Joomla» присутствует деление на компоненты и модули. «Компонентами» называются расширения, позволяющие добавлять дополнительный функционал к работе системы управления контентом. «Модулями» называются элементы вывода данных, отображение которых настраивается отдельно для каждой страницы web-сайта. Обычно, в комплекте с компонентами поставляются и модули, обеспечивающие вывод интерактивных элементов и любой информации хранящейся в них в любом предусмотренном шаблоном месте.

Модуль – это определенный участок сайта, позиция которого определяется через интерфейс администратора. В качестве модуля может быть представлены: меню сайта, список последних статей, счетчики посещений, поиск, авторизация пользователей, формы опросов и контактов, любая статическая и динамическая информация, заданная пользователем.

Стоит отметить отдельно, что позиции вывода модулей определяются в шаблоне оформления, действующем на странице. Кроме того, в «Joomla» существует специальная таблица позиций модулей. Для того, чтобы обозначить в системе активность модуля, необходимо зайти в следующий пункт меню: «Расширения» – «Менеджер модулей».

При использовании стандартных шаблонов оформления, либо шаблонов на основании стандартных макетов дизайна – никаких изменений производить не нужно. Необходимость в изменениях возникает лишь в случаях, когда в шаблон добавляются дополнительные позиции для вывода модулей.

Компонентная структура позволяет обеспечить гибкость и высокие функциональные возможности системы. Используя только необходимые, для работы сайта, модули и компоненты, администраторы web-ресурса защищают себя от узких мест и уязвимостей, имеющихся в незадействованных компонентах.

В качестве дополнительных компонентов к CMS «Joomla» возможно устанавливать: форумы, гостевые книги, блоги, галереи, интернет-магазины, видео-конференции, базы документов и многое другое.

При необходимости, существует много компаний, оказывающих платные услуги по созданию компонентов по техническим заданиям заказчиков. Фактически – любую задачу, необходимую администратору, можно решить путем интеграции сторонних компонентов.

Загрузка компонентов производится аналогично загрузке шаблонов через пункт меню в «Расширения» – «Установить /удалить». Управление модулями производится через «Менеджер модулей».

## 

**§8. Создание информационной структуры в системе управления контентом «Joomla» на примере web-сайта «Справочное пособие по HTML».** **Создание материалов сайта**

Наполним наш сайт необходимым содержимым. Для добавления нового материала, на главной странице палении администрирования существует кнопка «Добавить материал». Кроме того, возможно добавление материала через пункты меню: «Материалы» – «Менеджер материалов».

Таблица «Менеджер материалов» состоит из нескольких колонок:

* Заголовок — название материала:
* Опубликовано — если стоит значок , значит материал опубликован на сайте. Если стоит значок , то нет;



* На главной — если стоит значок , значит материал опубликован на главной странице. Если ,то нет (изменить состояние можно, кликнув по значку);



* Порядок — Порядок расположения материалов (можно изменить стрелочками  . Чтобы зафиксировать изменения, необходимо нажать );



* Доступ — регулирует доступ к данному разделу;
* Раздел — к какому разделу относится данная статья;
* Категория — к какой категории относится данная статья;
* Автор — автор статьи;
* Дата — дата создания статьи;
* Хиты — количество просмотров.

На странице «Менеджер материалов», в панели инструментов существуют следующие кнопки:

* Кнопка . Вызвать в новом окне страницу описания по окну «Менеджер разделов»;



* Кнопка . Открыть в новом окне настройки глобальных параметров материалов сайта;



* Кнопка . Создать новый материал;



* Кнопка . Изменить существующий материал. Для этого необходимо отметить нужный материал и нажать кнопку «Изменить»;



* Кнопка . Поместить выбранный материал в корзину;



* Кнопка . Копировать существующий материал в другой раздел/категорию. Отметьте нужный материал и нажмите «Копировать». Выберите нужный раздел из списка и нажмите «Сохранить»;



* Кнопка . Переместить существующий материал в другой раздел/категорию, а на прежнем месте удалить. Отметьте нужный материал и нажмите «Перенести». Выберите нужный раздел из списка и нажмите «Сохранить»;



* Кнопка . Скрыть (Сделать неопубликованным) существующий материал. Отметьте нужный материал и нажмите «Снять с публикации»;



* Кнопка . Показать (Опубликовать) существующий материал. Отметьте нужный материал и нажмите «Опубликовать»;



* Кнопка . Позволяет поместить ненужные или уже не актуальные материалы в архив. При этом из менеджера материалов они не удаляются, а просто появляется надпись «В архиве». Отметьте нужный материал и нажмите «В архив»;



* Кнопка . Позволяет достать материал из архива. Отметьте нужный материал и нажмите «Из архива».



Перед тем как создавать новый материал для сайта, нужно настроить глобальные параметры, которые будут относиться ко всем статьям. Это можно сделать, нажав на кнопку Параметры , которая находится в левом верхнем углу. Глобальные параметры для материалов выглядят следующим образом.



Если мы не задаем конкретные настройки для статьи, то она будет использовать по умолчанию эти глобальные параметры.

После того как глобальные параметры настроены можно переходить к созданию материалов сайта.

Для создания нового материала необходимо нажать на кнопку  «Создать».



Открывшуюся страницу можно условно разделить на 4 вкладки:

* Заголовок;
* Параметры – Статьи;
* Параметры – Расширенные;
* Мета – данные;

На вкладке «Заголовок» нужно заполнить следующие поля:

* Заголовок — ввести название статьи;
* Псевдоним — текст, который станет основой url для этого пункта;
* Раздел — указать раздел, к которому будет принадлежать статья;
* Опубликовано — отметить, будет опубликована статья или нет;
* На главной — отметить, будет статья опубликована на главной странице или нет;
* Категория — указать категорию, к которой будет принадлежать статья.

Можно не указывать раздел и категорию. Тогда материал можно вывести на сайт в виде отдельной страницы, создав под него отдельный пункт меню.

В поле для статьи вводится сама статья. С помощью установленного визуального редактора можно производить форматирование статьи, вставлять картинки и т.п.

В правой части страницы отражается статистика, а именно:

* ID Материала — идентификационный номер;
* Состояние — опубликована статья или нет;
* Хиты — количество посещений. Можно обнулить с помощью кнопки «Сбросить» количество исправлений;
* Создан — дата создания статьи;
* Изменён — дата последнего изменения;
* Во вкладке «Параметры - Статьи» заполняются следующие поля;
* Автор — имя авторизированного пользователя, создавшего статью. Это имя будет отображаться в административной панели;
* Псевдоним автора — если необходимо, вводится псевдоним автора;
* Доступ — определяется уровень доступа к статье;
* Дата создания — указывается дата создания статьи;
* Опубликовано — указывается дата опубликования статьи;
* Истекает — устанавливается дата, когда истекает срок опубликования.

Во вкладке «Параметры - Расширенные» можно использовать глобальные настройки для всех материалов или установить новые для отдельно взятой статьи:

* Заголовок — Показать/скрыть заголовок;
* Заголовок как ссылка — Использовать заголовок как ссылку на сам материал;
* Вступительный текст — Показать/скрыть вступительный текст;
* Название раздела — Показать/скрыть раздел, к которому принадлежит материал;
* Название раздела как ссылка — выводить текст раздела как ссылку на этот раздел;
* Название категории — Показать/скрыть категорию, к которой принадлежит материал;
* Название категории как ссылка — Выводить текст категории как ссылку на эту категорию;
* Рейтинг — Показать/скрыть рейтинг;
* Имя автора — Показать/скрыть автора;
* Дата и время создания — Показать/скрыть дату создания;
* Дата и время последнего изменения — Показать/скрыть дату изменения;
* Иконка PDF — Показать/скрыть кнопку «PDF». Позволяет пользователю сайта скачать версию материала в PDF-формате;
* Иконка печати — Показать/скрыть кнопку «Печать». Позволяет сразу отправить на печать просматриваемый пользователем документ;
* Иконка E-mail — Показать/скрыть кнопку «E-mail». Пересылает посетителя к форме письма для связи;
* Язык материала — Указать язык, на котором написана статья;
* Ключевая сноска — Текстовый ключ, по которому можно сослаться на эту статью;
* Альтернативный текст подробнее… — Ввести текст, который будет отображаться вместо слова «подробнее…», т.е. текст, который будет являться ссылкой на полную версию материала при создании анонса.

Мета-данные используются для раскрутки и продвижения сайта в Интернете.

Во вкладке «Мета-данные» заполняются следующие поля:

* Описание — краткое описание материала;
* Ключевые слова — ввести список слов через запятую на языке материала;
* Параметр Robots — Параметр позволяет задавать специфические для данной страницы правила индексации поисковыми роботами.

Допустимые значения:

* all - индексировать и переходить по ссылкам;
* index,nofollow - индексировать, но не переходить по ссылкам;
* noindex,follow - не индексировать, но переходить по ссылкам;
* noidex,nofollow - не индексировать и не переходить по ссылкам;
* Автор — автор материала.

С помощью кнопок на верхней панели можно:

* - Кнопка . Вызвать в новом окне страницу описания по окну «Редактирование материала»;



* - Кнопка . Закрыть редактирование материала без сохранения изменений. Нажмите «Отменить»;



* - Кнопка . Применить (Сохранить) изменения, но остаться на странице редактирования материала. Нажмите «Применить»;



* - Кнопка . Сохранить изменения и закрыть страницу редактирования материала. Нажмите «Сохранить»;



* - Кнопка . Просмотреть полученный результат. Нажмите «Просмотр».



### Рассмотрим пример добавления материала из справочного пособия по HTML. Материал называется «Плавающие фреймы»:

При добавлении материалов на сайт, меню сайта автоматически не формируется, его нужно создавать вручную.

**§9. Создание элементов меню**

После успешного добавления всего материала появляется возможность перейти к заключительной стадии создания информационной структуры сайта. В системе управления контентом «Joomla» нет прямой зависимости категорий, разделов и элементов меню. Администратор ресурса сам может определять, какие ссылки ему необходимо отображать в меню сайта.

Устройство системы управления контентом вносит свои ограничения в процесс создания и публикации пунктов меню. Для начала, разберем схему построения меню на сайте.

Меню в «Joomla» – это модуль, отображаемый в определенном, заранее обозначенном месте, состоящий из ссылок на документы, содержимое категорий и различные компоненты системы управления контентом.

Вывод меню возможен лишь в том месте графического шаблона, где обозначен контейнер для вывода модуля. В готовых, коммерческих и бесплатных, шаблонах существуют контейнеры, специально созданные для отображения меню. Для удобства администратора, они расположены в верхней, боковой и нижней частях страницы.

Открыв окно «Материалы» - «Менеджер меню», можно увидеть страницу с уже созданными типами меню.

С помощью кнопок на верхней панели можно:

* Кнопка . Вызвать в новом окне страницу описания по окну «Менеджер меню»;



* Кнопка . Создать новое меню;



* Кнопка . Изменить существующее меню. Для этого необходимо отметить нужное меню и нажать кнопку «Изменить»;



* Кнопка . Поместить существующее меню в корзину. Отметьте нужное меню и нажмите «Удалить»;



* Кнопка . Копировать существующее меню. Отметьте нужное меню и нажмите «Копировать»;



Для создания нового меню нужно нажать на кнопку «Создать».



На открывшейся странице «Детали меню» необходимо заполнить поля:

системное имя (латинским шрифтом), заголовок – имя пункта меню при выводе на сайт, описание меню и заголовок модуля – который будет использоваться при выборе способа вывода меню на сайт.

После этого нажать на кнопку  «Сохранить».



Если необходимо закрыть страницу без изменения необходимо нажать кнопку «Отменить».



При нажатии на кнопку  «Помощь» можно вызвать в новом окне страницу описания по окну «Детали меню».



Для того, чтобы попасть на страницу «Пункты меню» необходимо в «Менеджере меню» нажать на кнопку «Пункты меню» в одноименном столбце.



На странице доступен фильтр по названию пункта меню и по состоянию.

Таблица «Пункты меню» состоит из колонок:

* Пункт меню — Название пункта меню;
* По умолчанию — Какой пункт меню будет по умолчанию выводиться на главной странице;
* Опубликовано — опубликован пункт меню или нет;
* Порядок — порядок отображения (изменить можно с помощью стрелочек и далее сохранить новый порядок с помощью значка );



* Доступ — показывает, кто имеет доступ к данному пункту меню (изменить можно¸ кликнув на данный столбец);
* Системное имя — системное имя материала.

С помощью кнопок на верхней панели можно:

* Кнопка . Вызвать в новом окне страницу описания по окну «Пункты меню»;



* Кнопка . Создать новый пункт меню;



* Кнопка . Изменить существующий пункт меню. Для этого необходимо отметить нужный пункт меню и нажать кнопку «Изменить»;



* Кнопка . Поместить существующий пункт меню в корзину. Отметьте нужный пункт и нажмите «В корзину»;



* Кнопка . Копировать существующий пункт меню. Отметьте нужный пункт меню и нажмите «Копировать»;



* Кнопка . Переместить существующий пункт меню. Отметьте ей пункт меню и нажмите «Перенести»;



* Кнопка . Скрыть (Сделать неопубликованным) существующий пункт меню. Отметьте нужный пункт меню и нажмите «Скрыть»;



* Кнопка . Показать (Опубликовать) существующий пункт меню. Отметьте -нужный пункт меню и нажмите «Опубликовать»;



* Кнопка . Вывести пункт меню на главную страницу. Отметьте нужный пункт меню и нажмите «По умолчанию»;



* Кнопка . Показывает все меню в менеджере меню.



Для создания пункта меню необходимо: Нажать на **кнопку  «Создать»** и перейти на страницу **«Пункт меню»**. Выбрать раздел «Материалы» и в нем выбрать «Стандартный шаблон материала»:



Во вкладке «Подробности пункта меню» существуют следующие поля:

* Заголовок — название пункта меню
* Псевдоним — псевдоним пункта меню латинским шрифтом
* Показать в — привязать пункт к определенному меню
* Родительский элемент — указать родительский элемент пункта меню
* Опубликовано — показать/скрыть пункт меню
* Доступ — указать, кто имеет доступ к данному пункту меню
* Открывать в — в каком окне будет открываться пункт меню

Для того, что бы прикрепить пункт меню к уже добавленному нами выше материалу «Плавающие фреймы», поля необходимо заполнить следующим образом.

В пункте меню «Параметры – Основные», нужно нажать на кнопку «Выбрать», и указать путь к материалу к которому нам нужно привязать создаваемый пункт меню. В данном случае это материал «Плавающие фреймы».

После проделанных выше операций нажимаем на пункт меню Сохранить и смотрим полученных результат. Теперь наш пункт меню «Плавающие фреймы» связан с материалом «Плавающие фреймы» и при нажатии на пункте в меню «Плавающие фреймы» мы переходим на нужный нам материал.



Аналогичным образом добавляются и связываются с нужными материалами все оставшиеся пункты меню.

## 

**§10. Выводы по работе с системой управления контентом CMS «Joomla»**

Система управления контентом CMS «Joomla» , рассмотреная в данной дипломной работе является бесплатной и свободно распространяемой по лицензии GPL. В данной ситуации имеются свои плюсы и минусы. С одной стороны – открытый исходный код позволяет дорабатывать, изменять, добавлять элементы систем управления контентом без необходимости связи с разработчиками. С другой – дает возможность злоумышленникам подробно изучить все «узкие места» и воспользоваться ошибками разработчиков.

Система управления контентом «Joomla» имеет большое количество выявленных уязвимостей, постоянно закрываемых заплатками разработчиков. В данный момент система выпускается в двух версиях 1.5.х и 1.0.х. Версия 1.0.х, с недавнего времени более не поддерживается. В данном случае, есть необходимость переводить все имеющиеся проекты на версию 1.5.15 и выше.

В плане удобства и функциональности – эта система управления контентом показала себя хорошо. «Joomla» отлично подходит для стандартных проектов, выполняющих функцию представления определенной информации заказчика, каталогизации по разделам, добавления форм связи.

Со стороны пользователя – система абсолютно пригодна для комфортного просмотра страниц и получения информации.

«Joomla» обладает административным разделом, предназначенным для создания, редактирования и удаления записей и метриалов. При помощи Административного раздела, манипуляций с темами оформления и внутренними параметрами системы становятся легкими и интуитивно понятными. Очень легко происходит добавление расширений, операции с пользователями и материлами.

Расход системных ресурсов в «Joomla» весьма невелик, нагрузка на ОЗУ сервера доходит до 20–40 Мб. Работа с шаблонами оформления в «Joomla» организована на высоком уровне. Эта система управления контентом позволяет изменять код шаблона из интерфейса администратора.

При разработке и вводе в эксплуатацию система показала себя хорошо. После детального изучения функциональных особенностей, можно сделать вывод о том, что «Joomla» предназначена для сайтов разного уровня и класса. Эта система пригодна для создания типовых web-сайтов, организации интернет-сообществ или небольших по размерам медиа-проектов.

**§11. Апробация результатов исследования**

Эффективность результатов использования справочного пособия по HTML на уроках информатики при изучении темы «Основные понятия языка HTML»

Использование данного справочного пособия по HTML на уроках информатики при изучении темы «Основные понятия языка HTML» помогает:

* Лучше изучить и усвоить новый материал;
* Улучшить качество обучения и воспитания, что повысит информационную культуру учащихся;
* Повысить мотивацию к учебе;
* Совершенствовать систему управления обучением на различных этапах урока;
* Повысить уровень подготовки учащихся в области современных информационных технологий;
* Использовать дифференцированный подход к обучению;
* Узнать о новых возможностях компьютера;
* Повысить уровень самостоятельности и внести элементы творчества в процесс образования.

**Справочное пособие помогает решить следующие дидактические задачи:**

* Усвоить базовые знания по теме;
* Систематизировать усвоенные знания;
* Сформировать навыки самоконтроля;
* Сформировать мотивацию к учению в целом и к информатике в частности;
* Оказать учебно-методическую помощь учащимся в самостоятельной работе над учебным материалом.

Данную технологию можно рассматривать как объяснительно-иллюстративный метод обучения, основным назначением которого является организация усвоения учащимися информации путем сообщения учебного материала и обеспечения его успешного восприятия, которое усиливается при подключении зрительной, моторной и эмоциональной памяти. Давно известно, что для большинства людей, зрение является основным средством получения информации из окружающего мира. 90% всей информации человек получает именно с помощью зрения, а оставшиеся 10% делятся между другими органами чувств (слухом, вкусом, осязанием, обонянием). Поэтому применяемый объяснительно-иллюстративный метод обучения является одним из наиболее эффективных в образовании.

Экономия времени, необходимого для изучения конкретного материала, в среднем составляет 30%, а приобретенные знания сохраняются в памяти значительно дольше.

При использовании на уроке справочного пособия по HTML структура урока принципиально не изменяется. В нем по-прежнему сохраняются все основные этапы, изменятся, только их временные характеристики.

Необходимо отметить, что по времени этап активизации учащихся и мотивации их к обучению в данном случае увеличивается и несет познавательную нагрузку, а этап введения нового материала, наоборот, уменьшается. Он компенсируется за счет активной, самостоятельной деятельности учащихся. Мотивация учащихся в начале урока - необходимое условие успешности обучения, так как без интереса к пополнению недостающих знаний, без воображения и эмоций немыслима самостоятельная, творческая деятельность ученика.

Благодаря наличию в учебном пособии по HTML гипертекстовых ссылок у учащихся развивается системное, аналитическое мышление. Кроме того, с помощью данного учебного пособия можно использовать разнообразные формы организации познавательной деятельности: фронтальную, групповую, индивидуальную.

Таким образом, данное справочное пособие наиболее оптимально и эффективно соответствует триединой дидактической цели урока:

* **Образовательный аспект** - восприятие учащимися учебного материала, осмысливание связей и отношений в объектах изучения.
* **Развивающий аспект** - развитие познавательного интереса у учащихся, умения обобщать, анализировать, сравнивать, активизация творческой деятельности учащихся.
* **Воспитательный аспект** - воспитание научного мировоззрения, умения четко организовать самостоятельную и групповую работу, воспитание чувства товарищества, взаимопомощи.

Справочное пособие по HTML может быть использовано:

**1. Как сопровождение объяснения учителя.**

При использовании учебного пособия в процессе объяснения новой темы достаточно выбрать и открыть необходимый раздел учебного пособия, в котором будут показаны основные моменты, необходимые для изучения данной темы. На экране могут также появляться определения, схемы, примеры выполнения заданий и т.д. Ребята не должны тратить время на переписывание данной информации себе в тетрадь. Акцент учителя должен быть направлен на доступное и понятное объяснение материала, на повышение заинтересованности в нем учащихся. Учащиеся должны понимать, что их основная задача не переписать показанный учителем материал себе в тетрадь, а разобраться в нем, понять основные принципы, законы и правила, используемые в данной теме.

**2. Как информационно-обучающее пособие.**

В обучении особенный акцент ставится сегодня на собственную деятельность ребенка по поиску, осознанию и переработке новых знаний. Учитель в этом случае выступает, как организатор процесса учения, руководитель самостоятельной деятельности учащихся, оказывающий им нужную помощь и поддержку.

Такие пособия удобно использовать в тех случаях, когда ученик по какой-то причине не успел выполнить задание во время урока, или если он пропустил тему по причине болезни. В этом случае учащиеся могут прийти в кабинет информатики после уроков и доработать материал. И, наоборот, учащиеся которые успевают за урок выполнить все предложенные по теме задания, могут, не дожидаясь остальных, переходить к следующему разделу темы или выполнять творческое задание по изученной теме. Таким образом, благодаря индивидуальному режиму работы каждого учащегося, все достигают положительного результата.

Справочное пособие, позволяющее организовать такую работу, должно быть более полным и включать в себя материалы по нескольким сопутствующим темам. В этом случае обеспечивается возможность для самостоятельного изучения разделов темы, а также для опережающего обучения. Структура пособия в этом случае должна быть достаточно сложной, нелинейной, с большим количеством разветвлений и основываться на «ручной» навигации. Также желательно, присутствие ссылок на другие ресурсы, срабатывающие, когда пользователь выполняет щелчок мышью на соответствующем объекте. При наличии такой сложной структуры важно предусмотреть понятный интерфейс, приятное визуальное оформленные, а главное постоянно доступное «главного меню» для выбора желаемой темы и подтемы.

При изучении темы «Основные понятия языка HTML» кроме описания работы различных тегов, должны приводиться конкретные примеры реализации этих тегов на практике.

Использование компьютера на этом этапе имеет много плюсов: индивидуальный темп работы с программой, большой объем информации по теме, возможность посмотреть пример и сразу увидеть, для чего конкретно используется тот или иной тег.

Преимущества использования справочного пособия по HTML на уроках информатики:

Учеников привлекает новизна проведения таких самостоятельных уроков. В классе во время таких занятий создаётся обстановка реального общения, при которой ученики стремятся выразить мысли «своими словами», они с желанием выполняют задания, проявляют интерес к изучаемому материалу, у учеников пропадает страх перед компьютером. Учащиеся учатся самостоятельно работать с учебной, справочной и другой литературой по предмету. У учеников появляется заинтересованность в получении более высокого результата, готовность и желание выполнять дополнительные задания. При выполнении практических действий проявляется самоконтроль.

Можно выделить следующие особенности данной технологии:

* Качество изображения, выполняемого мелом на доске, не выдерживает никакого сравнения с аккуратным, ярким, чётким и цветным изображением на экране;
* С помощью доски и мела затруднительно объяснять работу с различными приложениями;
* В случаях выявления в пособии недостатков или ошибок, можно сравнительно легко устранить дефекты;
* В зависимости от подготовленности учащихся один и тот же материал можно объяснять или очень быстро или очень подробно. Темп и объём излагаемого материала определяется по ходу урока.
* Повышение уровня использования наглядности на уроке.
* Появляется возможность организации проектной деятельности учащихся по созданию учебных программ под руководством преподавателей информатики и учителями-предметниками.
* Преподаватель, создающий или использующий информационные технологии, вынужден обращать огромное внимание на логику подачи учебного материала, что положительным образом сказывается на уровне знаний учащихся.
* Изменяется отношение к ПК. Ребята начинают воспринимать его в качестве универсального инструмента для работы в любой области человеческой деятельности.

Апробация справочного пособия по HTML проводилась учениками 9а, 9б и 9в классов ГОУ СОШ № 390 города Москвы на уроках информатики. Глава «Основные понятия языка HTML» изучалась детьми в рамках большого раздела курса информатики «Компьютерные коммуникации».

Тематическое планирование курса рассчитано на преподавание информатики и информационных технологий для обучающихся 9 классов на 70 учебных часов (2 часа в неделю). Рабочая программа составлена на основе примерной программы «Информатика и информационные коммуникационные технологии», разработанной на основе проекта федерального компонента государственного стандарта общего образования и базисного учебного плана. Данное тематическое планирование соответствует примерной программе базового курса «Информатика и ИКТ», рекомендованной Министерством образования.

При изучении информатики каждый из трех 9-х классов делится на 2 подгруппы. В апробации справочного пособия по HTML учувствовала только подгруппа № 1 из каждого класса. Подгруппа № 2 в апробации участия не принимала. На основании этого и делается результат о эффективности данного справочного пособия. Результаты апробации можно посмотреть на следующих диаграммах.

Данное справочное пособие я использовал при изучении темы «Основные понятия языка HTML» в разделе школьного курса информатики «Компьютерные коммуникации». Ниже представлены диаграммы качества обучения учеников 9-х классов по другим разделам школьного курса информатики.

Как видно из диаграмм, раздел «Компьютерные коммуникации» на котором применялось справочное пособие по HTML был хорошо усвоен детьми. По этому разделу самое высокое качество обучения.

Опытная проверка показала хорошую эффективность результатов использования справочного пособия по HTML на уроках информатики при изучении темы «Основные понятия языка HTML». Ребята стали активнее вести себя на уроке, у них появился больший интерес к предмету, повысился уровень мотивации к обучению в целом. Увеличилась скорость усвоения материала, объем выполняемых заданий на уроке также вырос. Появилась заинтересованность узнавать больше, повысился уровень самостоятельной и творческой деятельности. Учащиеся с удовольствием работали, как индивидуально, так и в коллективе. Вырос общий уровень универсальных учебных действий.

**§12. Вывод**

Создание справочного пособия с помощью системы управления контентом «Joomla» включает в себя множество этапов:

* загрузка последней версии **CMS** «Joomla»;
* установка локального сервера Денвер;
* создание базы данных и пользователей в MySQL;
* установка CMS «Joomla» на локальный компьютер;
* конфигурация системы управления контентом;
* изменение и настройка шаблона дизайна сайта;
* подключение к «Joomla» необходимых модулей и компонентов;
* создание материалов сайта;
* создание элементов меню сайта.

Каждый из этих этапов имеет свои нюансы и особенности. У пользователя, впервые столкнувшегося с созданием такой автоматизированной образовательной системы, может возникнуть много вопросов и затруднений по работе с данной системой управления контентом. В практической части диплома рассмотрены принципы работы и основные этапы создания образовательного ресурса с помощью CMS «Joomla». Оговорены особенности и достоинства, благодаря которым работа с данной системой управления над созданием справочного пособия не должна вызывать затруднений.

Используя основные приемы, описанные в дипломе, пользователь, захотевший создать собственную обучающую систему на основе CMS «Joomla», сможет с легкостью решить поставленную перед собой задачу.

Проводимая учащимися 9-х классов апробация данного справочного пособия показала высокую эффективность изучения нового материала и подтвердила ожидания, возложенные на данный образовательный ресурс.

**Заключение**

Современный уровень развития информационных технологий может стать инструментом решения одной из главных проблем педагогики - создания у учащихся тяги к знаниям и желания покорять новые вершины в сфере образования. Компьютерная техника должная увлечь и заинтересовать учащихся, возродить интерес и увеличить уровень мотивации к получению новых знаний.

Цели, поставленные в дипломной работе, были достигнуты. Было разработано справочное пособие по HTML с помощью системы управления контенком «Joomla». Показано, что при минимальной теоретической и практической подготовке и небольшом навыке работы с CMS «Joomla» подобный образовательный ресурс сможет создать не только преподаватель, но и учащийся.

В дипломе перечислены основные возможности данной системы управления контентом, показаны способы их практической реализации на примере создания справочного пособия по HTML.

Также в дипломной работе были определены основные достоинства и недостатки CMS «Joomla», оговорены ее особенности, продемонстрированы основные принципы и способы работы с данной системой управления контентом.

Созданное справочное пособие применялось на уроках информатики при изучении темы «Основные понятия языка HTML».

Апробация справочного пособия проводилась учащимися 9-х классов. Опытная проверка показала, что благодаря данному образовательному ресурсу, учащиеся стали лучше и быстрее усваивать новый учебный материал. У школьников выросла общая мотивация к учебе. Улучшилось качество обучения, повысился уровень воспитания информационной культуры учащихся. Стала проявляться большая степень самостоятельности в обучении. Появились элементы творчества.

В разработанном справочном пособии с успехом реализованы все перечисленные во введении цели и задачи. Это позволяет говорить о высокой эффективности использования данного справочного пособия по HTML при изучении темы «Основные понятия языка HTML».

Материал в справочном пособии, благодаря простоте и доступности изложения, рассчитан на широкий круг читателей. Текст снабжен рисунками, таблицами и примерами, что также упрощает его понимание и усвоение. Благодаря интегрированной системе поиска в учебном пособии можно без труда отыскать интересующее понятие.

Данное обучающее пособие является одним из наиболее эффективных средств повышения качества образования на уроках информатики.

**Библиография**

# Гончаров, А.Н. Самоучитель HTML [Текст] / А.Н. Гончаров.- СПб.: Питер, 2002.

# Горнаков, С.Г. Осваиваем популярные системы управления сайтом [Текст] / С.Г. Горнаков.- М.: Наука, 2009.

# Ежевский, Д.О. О критериях создания электронных учебных пособий [Текст] / Д.О. Ежевский.- М.: Эксмо, 2004.

# Колисниченко, Д.Н. Движок для вашего сайта. CMS Joomla!, Slaed, PHP-Nuke [Текст] / Д.Н. Колисниченко.- М.: НТ Пресс, 2008.

# Колисниченко, Д.Н. Joomla! [Текст]: Руководство пользователя / Д.Н. Колисниченко.- М.: Диалектика, 2009.

# Норт, Б. Joomla! [Текст]: Практическое руководство / Б. Норт.- М.: Символ-плюс, 2008.

1. Рамел, Д. Самоучитель Joomla! [Текст] / Д. Рамел.- СПб.: Питер, 2008.

# Хаген, Г. Создание веб-сайтов с помощью Joomla! 1.5 [Текст] / Г. Хаген.- М.: Вильямс, 2008.

# Чиртик, А.В. Популярный самоучитель HTML [Текст] / А.В. Чиртик.- СПб.: Питер, 2006

# Шишкин, В.В. Многоуровневая концепция создания электронных учебных пособий [Текст] / В.В. Шишкин.- М.: Махаон, 2001.