

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(Н И У « Б е л Г У »)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК

Кафедра общей математики

**Система дистанционного обучения
по дисциплине «Дифференциальные уравнения»**

Выпускная квалификационная работа

**студента очной формы обучения
направления подготовки 01.03.02
Прикладная математика и информатика
4 курса группы 07001206
Гринева Вадима Алексеевича**

Научный руководитель:
д.т.н., проф.
Аверин Г.В.

БЕЛГОРОД 2016

ОГЛАВЛЕНИЕ

Введение.....	3
1 Современное состояние вопроса	6
1.2 Понятие дистанционного обучения	10
1.2.1 Характерные особенности дистанционного обучения	11
1.2.2 Принципы дистанционного обучения	13
1.3 Развитие мирового рынка дистанционного обучения.....	14
1.4 Развитие дистанционного образования в России	17
1.4.1 Целесообразность создания системы ДО в России.....	18
1.5 Выводы по разделу	19
2 Методика преподавания курса дифференциальные уравнения	21
2.1 Содержание курса дифференциальные уравнения.....	21
2.2 Техническое содержание курса	23
2.2.1 Теоретические материалы.....	23
2.2.2 Видео уроки.....	24
2.2.3 Тестовые задания	25
2.3 Структура сайта.....	25
2.4 Выводы по разделу	28
3 Разработка требований к системе дистанционного обучения.....	29
3.1 Анализ структурно-функциональных особенностей системы дистанционного обучения.....	29
3.1.1 Инструктивный блок	30
3.1.2 Информационный блок	30
3.1.3 Контрольный блок	31
3.1.4 Управляющая система.....	32
3.2 Анализ аппаратного и программного обеспечения системы	33
3.2.1 Операционная система	33
3.2.2 Язык разметки гипертекстовых страниц HTML.....	34
3.2.3 Программное обеспечение для создания мультимедийных компонентов системы.....	34

3.2.4 Программное обеспечение для реализации интерфейса и базы данных	35
3.2.5 Аппаратное обеспечение системы	37
3.3 Выводы по разделу	38
4 Проектирование базы данных и интерфейса пользователя системы дистанционного обучения	39
4.1 Выбор инструментальных средств для разработки базы данных системы	39
4.2 Определение содержательной структуры базы данных системы	41
4.3 Разработка структуры базы данных	42
4.4 Формирование запросов	46
4.5 Выбор инструментальных средств для разработки системы	49
4.5.1 Среда Adobe Flash	49
4.5.2 Язык разметки гипертекстовых страниц HTML	50
4.5.3 Язык программирования PHP	52
4.6 Архитектура клиент-сервер	54
4.7 Структура системы дистанционного обучения	56
4.7.1 Модуль регистрации	57
4.7.2 Модуль аутентификации и авторизации	60
4.7.3 Модуль пользователя	62
4.7.4 Модуль администратора	64
4.8 Выводы по разделу	67
Заключение	69
Список использованной литературы	71
Приложение 1	75

ВВЕДЕНИЕ

Огромное влияние на все сферы жизни человека, в том числе и образование, имеет одна из основных проблем современности - большой поток информации, которую необходимо не только собрать, но и обработать, проанализировать и сделать соответствующие выводы.

Современное общество ставит задачу предоставления каждому человеку свободного открытого доступа к образованию на протяжении всей его жизни, с учетом его интересов, способностей и потребностей. При этом нужно обеспечить его подготовку к жизни в условиях быстрого изменения информационных потоков, возможной смене профессии, активной, самостоятельной деятельности и т.д.

Необходима новая форма получения образования, адекватная информационному обществу. Она должна в полной мере обеспечивать право на получение образования и удовлетворять принципу гуманистичности в том, что никто не должен быть лишен возможности учиться вследствие бедности, географической изолированности, социальной незащищенности и невозможности посещать образовательные учреждения в силу физических недостатков или занятости производственными и личными делами [2].

В сложившейся ситуации специалисты считают дистанционное обучение эффективной системой подготовки и непрерывного поддержания высокого квалификационного уровня для любого человека.

Дистанционное обучение (ДО) - совокупность образовательных услуг, предоставляемых многим слоям населения в стране и за рубежом с помощью специализированной информационной образовательной среды, базирующейся на средствах обмена учебной информацией на расстоянии. Информационно-образовательная среда ДО - системно организованная совокупность средств передачи данных, информационных ресурсов, протоколов взаимодействия, аппаратно-программного и организационно-методического обеспечения,

ориентированного на удовлетворение образовательных потребностей пользователей. Дистанционное образование - одна из форм непрерывного образования, которая призвана реализовать права человека на образование и получение информации [2].

Традиционная форма обучения, как существовала, так и продолжает существовать, и пока уступает свои позиции альтернативным подходам. С другой стороны, обучение онлайн уже перешло из статуса экспериментального в привычный режим, и активно используется как учебными заведениями, так и предприятиями, заинтересованными в постоянном повышении уровня знаний и навыков своих сотрудников [2].

Мировой рынок ДО развивается достаточно активно, чему способствует с одной стороны повышение спроса на образовательные услуги, а с другой - развитие собственно информационных технологий, рост интернет-аудитории пользователей [19].

Цель работы – изучить методы и возможности дистанционных образовательных технологий по программам математики и разработать систему дистанционного обучения по дисциплине: “Дифференциальные уравнения”.

Для достижения поставленной цели следует решить следующие задачи:

- проанализировать, существующие технологии дистанционного образования и тенденции их развития;
- изучить методику преподавания курса дифференциальные уравнения и особенности реализации курса с использованием дистанционных технологий образования;
- определить требования к системе дистанционного обучения “Дифференциальные уравнения”;
- спроектировать базу данных и интерфейс пользователя системы дистанционного обучения по дисциплине “Дифференциальные уравнения”.

Объектом исследования являются существующие технологии дистанционного образования в области преподавания высшей математики.

Предметом исследования является методы, средства и инструменты представления информации и разработки систем дистанционного обучения применительно к курсу дифференциальные уравнения.

Сегодня, количество студентов и слушателей, способных и желающих учиться по дистанционным технологиям образования, непрерывно и растет, так что создание системы дистанционного обучения на сегодняшний момент актуальной задачей.

1 Современное состояние вопроса

1.1 Возникновение и развитие дистанционного обучения

Дистанционное обучение по определению это обучение на расстоянии, когда преподаватель и обучаемый удалены друг от друга. Актуальность использования дистанционного обучения в практике очевидна. Достаточно привести аргументы американского педагога Кэрола Твига о востребованности дистанционного университетского образования в США. По его данным и данным федерального департамента образования в США только 43% студентов вузов моложе 25 лет, лишь четверть - молодежь 18-22 лет. Остальная часть студентов - люди взрослые, обремененные семейными и деловыми заботами. Для них достаточно проблематичны очные формы университетского образования. Дистанционное обучение отвечает требованиям современной жизни, особенно, если учесть не только транспортные расходы, но и расходы на организацию всей системы очного обучения. Отсюда все повышающийся интерес к дистанционному обучению не только университетскому, но к самым различным его формам, необходимым на протяжении всей жизни человека.

В последние годы университеты разных стран мира обратили внимание на возможности использования компьютерных телекоммуникационных технологий для организации дистанционного обучения. Компьютерные телекоммуникации обеспечивают эффективную обратную связь, которая предусматривается как в организации учебного материала, так и общении с преподавателем, ведущим данный курс. Такое обучение на расстоянии и получило в последние годы название "дистанционного обучения" в отличие от знакомого всем заочного обучения. Эта проблема особенно актуальна для России с ее огромными территориями и сосредоточием научных центров в крупных городах. В настоящее время Министерство образования РФ обеспечивает создание единой образовательной телекоммуникационной сети.

Необходимо, чтобы при разработке такой сети учитывались возможности и потребности разных видов образовательных систем России - как высшего, так и общего среднего образования, системы повышения квалификации. Проблема непрерывного образования, профессиональной переориентации актуальна сегодня, как никогда раньше, и ее значимость будет с годами возрастать по мере развития рыночной экономики в нашей стране, усиления миграции населения. Отсюда становится очевидной значимость научно обоснованной концепции наполнения и использования единого телекоммуникационного образовательного пространства для разных образовательных систем [10, 11].

Исследование рынка труда показало, что контингент лиц, остро нуждающихся в образовательных услуг, которые традиционная система образования предоставить не может, достаточно обширен. Это, например:

- лица всех возрастов, проживающие в малоосвоенных регионах страны, удаленных от центров высшего образования;
- лица, которые желают приобрести новые знания, получить второе образование;
- лица, которые не имеют возможности получить образование в традиционной системе из-за ограниченной пропускной способности этой системы, невозможности совмещения учебы с работой и других специфических условий (сельские жители, спортсмены, кочевники и т.п.);
- граждане, проходящие действительную срочную службу в рядах вооруженных сил, а также офицеры запаса и члены их семей;
- лица, имеющие медицинские ограничения для получения регулярного образования в стационарных условиях;
- группа менеджеров различного уровня, преподавателей и других специалистов, нуждающихся в переподготовке и повышения квалификации;
- субъекты и объекты пенитенциарной системы (осуждённые и обслуживающий персонал исправительных учреждений);

- граждане, желающие получить образование в зарубежных образовательных учреждениях;
- лица, зарегистрированные в соответствующих службах занятости [3, 28].

Приведенные факты обусловили необходимость внедрения новой формы получения образования, адекватной информационному обществу. Она должна в полной мере обеспечивать право на получение образования и удовлетворять принципу гуманности и доступности заключающихся в том, что никто не должен быть лишен возможности учиться по причине бедности, географической изолированности, социальной незащищенности и невозможности посещать образовательные учреждения через физические недостатки или занятость производственными и личными делами. Эта форма получения образования должна:

- сделать образование доступным для широких слоев населения независимо от места проживания, возрастного ценза, условий жизни и работы, на основе всеобщего равенства и в зависимости от способностей каждого;
- реализовать потребности населения в образовательных услугах, а страны в качественно подготовленных специалистах;
- реализовать важные и конструктивные идеи опережающего и непрерывного образования, быть способной реагировать на запросы рынка труда, которые постоянно меняются;
- компенсировать сокращение государственного финансирования, снять социальную напряженность, повысить социальную и профессиональную мобильность населения;
- сохранять и приумножать знания, кадровый и материальный потенциал, накопленный отечественной системой образования, полнее использовать педагогический и научный потенциал вузов, эффективно использовать существующие и перспективные средства новых

информационных технологий и решить ряд других социально-экономических задач [3].

В сложившейся ситуации специалисты по стратегическим проблемам образования считают дистанционное обучение самой эффективной системой подготовки и непрерывного поддержания высокого квалификационного уровня [29, 33].

Историко-педагогический анализ проблем становления и развития ДО показал, что в мире накоплен опыт реализации систем дистанционного обучения. Вообще мировая тенденция перехода к нетрадиционным формам образования прослеживается в росте числа вузов, ведущих подготовку по новым информационным технологиям [33].

Традиционная форма обучения, как существовала, так и продолжает существовать, и пока что не очень уступила свои позиции альтернативному – дистанционному его варианту. С другой стороны, онлайн обучение уже перешло из статуса экспериментального в обычный режим, и активно используется как учебными заведениями, так и предприятиями, заинтересованными в постоянном повышении уровня знаний и навыков своих сотрудников [3, 20].

Дистанционное образование, хотя отличается от двух традиционных форм обучения – очной и заочной, и при этом, по сути, является их гибридом. От заочной формы оно «наследует» факт удаленности ученика и преподавателя. Так же, как при очных занятиях, предполагает личный учебный план и постоянное взаимодействие с преподавателем. Задача последнего – координация обучения, консультировать, проверять контрольные работы и тесты, помогать готовиться к экзаменам [9, 33].

От привычных очной и заочной форм дистанционное обучение отличается возможностью самостоятельного выбора последовательности изучения предметов и темпа работы. Например, за один семестр можно изучить

курс, которому в рамках дневной формы обучения посвящают целый год, или, наоборот, можно «растянуть» его на два года [9, 33].

Еще одной особенностью ДО является фактическое стирание формальных границ, существующих между средней школой, колледжем и последиplomного образования. Если раньше определенная дисциплина относящаяся, например, к курсу программы делового администрирования, могла выкладываться только на уровне последиplomного образования, то теперь независимая тренинг-компания может преподавать ее также студентам, которые не имеют диплома [9].

Развитие ДО не стоит расценивать исключительно как дополнительный сервис для клиента-ученика. Учебные заведения и государство ищут пути увеличения числа студентов при одновременном стремлении к общему снижению затрат на образование. Например, для США развитие дистанционных программ - вопрос принципиальный, обусловлено совокупным дефицитом бюджета. Поскольку государственные вузы, и большинство частных, лишаются достаточных средств для дальнейшего развития своих кампусов, им все чаще приходится задумываться о переносе части учебных программ в Интернет [1, 23].

1.2 Понятие дистанционного обучения

Многолетний отечественный и мировой опыт теории и практики ДО подтверждает актуальность и новизну направлений исследований, приближает к пониманию сути дистанционного обучения, которое является одним из средств получения образования, основанной на использовании специфических образовательных технологий, на современных методиках обучения, технических средствах и передачи информации, информационных и телекоммуникационных технологиях [2].

Дистанционное обучение - совокупность технологий, обеспечивающих доставку обучаемым основного объема изучаемого материала, интерактивное взаимодействие обучаемых и преподавателей в процессе обучения, предоставление обучаемым возможности самостоятельной работы по освоению изучаемого материала, а также в процессе обучения [4].

Информационно-образовательная среда ДО - системно-организованная совокупность средств передачи данных, информационных ресурсов, протоколов взаимодействия, аппаратно-программного и организационно-методического обеспечения, ориентированного на удовлетворение образовательных потребностей пользователей. Дистанционное обучение – одна из форм непрерывного образования, которая призвана реализовать права человека на образование и получение информации [2].

В последние годы все шире в образовании используются виртуальные образовательные программы. Виртуальная образовательная программа – это программа, моделирующая ситуацию учебного процесса: лекцию, семинар, экзамен и т.д., построенная по принципу компьютерной игры с виртуальным участием студента и преподавателя. Дополнительные технические средства (очки для трёхмерного восприятия виртуального пространства и т.д.) могут создавать эффект «присутствия» («нахождения внутри») и сделают виртуальную лекцию, семинар или экзамен максимально приближенными к реальности [36].

1.2.1 Характерные особенности дистанционного обучения

Дистанционное обучение имеет следующие основные особенности.

1. Гибкость. Клиенты системы дистанционного обучения (СДО) не посещают регулярных занятий в виде лекций и семинаров, а работают в удобное для себя время в удобном месте и в удобном темпе, СДО представляющей большое преимущество для тех, кто не может или не хочет прекратить свой привычный уклад жизни. Это наиболее актуально для крупных

предприятий и корпораций, где остро стоит проблема массового обучения персонала без отрыва от производства.

2. Модульность. В основу программ ДО положен принцип модульности. Каждый отдельный курс создает целостное представление об определенной предметной области. Это позволяет из набора независимых курсов-модулей формировать учебную программу, отвечающую индивидуальным или групповым потребностям.

3. Экономическая эффективность. Относительно низкая себестоимость обучения обеспечивается за счет использования более концентрированной визуализации и унификации содержания, ориентированности технологий ДО на большое количество учеников, а также за счет более эффективного использования существующих учебных площадей и технических средств.

4. Новая роль преподавателя. На него возлагаются такие функции, как координирование познавательного процесса, корректировка преподаваемого курса, консультирование при составлении индивидуального учебного плана, руководство учебными проектами и так далее. Он руководит учебными группами взаимоподдержки, помогает ученикам в профессиональном самоопределении.

5. Специализированный контроль качества образования. Формой контроля в ДН является дистанционно организованные экзамены, собеседования, практические, курсовые и проектные работы, экстернат, компьютерные интеллектуальные тестирующие системы. Решение проблемы контроля качества ДО, его соответствия образовательным стандартам имеет принципиальное значение для успеха всей системы ДО.

6. Использование специализированных технологий и средств обучения. Прогрессивная технология строится на фундаменте определенного содержания и должна соответствовать требованиям его визуализации [2].

1.2.2 Принципы дистанционного обучения

Сегодня выделяют следующие принципы дистанционного образования.

1. Принцип гуманистичности обучения является главным в системе непрерывного интенсивного обучения. Его суть заключается в направленности обучения и образовательного процесса целиком на человека; в создании максимально благоприятных условий для овладения учащимися социально накопленного опыта, заключенного в содержание обучения; освоении выбранной профессии для развития и проявления творческой индивидуальности, высоких гражданских, нравственных, интеллектуальных качеств, которые обеспечивали бы ему социальную защищенность, безопасное и комфортное существование.

2. Принцип приоритетности педагогического подхода при проектировании образовательного процесса в СДО. Суть принципа: проектирование СДО необходимо начинать с разработки теоретических концепций, создания дидактических моделей явлений, которые предполагается реализовать. Опыт компьютеризации позволяет утверждать, что когда приоритетной является педагогическая сторона, система получается эффективнее.

3. Принцип педагогической целесообразности применения новых информационных технологий. Он требует педагогической оценки эффективности каждого шага проектирования и создания СДО. Поэтому на первый план необходимо ставить внедрение техники, а соответствующее содержательное наполнение учебных курсов и образовательных услуг.

4. Принцип выбора содержания образования. Содержание образования СДО должен соответствовать нормативным требованиям Государственного образовательного стандарта и требованиям рынка.

5. Принцип обеспечения безопасности информации, циркулирующей в СДО. Необходимо предусматривать организационные и технические средства

безопасного и конфиденциального хранения, передачи и использования нужных сведений, обеспечения безопасности при хранении, передаче и использовании.

6. Принцип стартового уровня образования. Эффективное обучение в СДО требует определенных знаний, умений, навыков.

7. Принцип мобильности обучения. Он заключается в создании информационных сетей, баз и банков знаний и данных для ДН, позволяющих студенту корректировать, дополнять свою образовательную программу в необходимом направлении при отсутствии соответствующих услуг в вузе, где он учится.

8. Принцип неантогонистичности ДО существующим формам образования. Проектируемая СДО сможет предоставить необходимый социальный и экономический эффект при условии, если создаваемые и внедряемые информационные технологии не станут посторонним элементом в традиционной системе высшего образования, а будут естественным образом интегрированы в него [3, 6].

1.3 Развитие мирового рынка дистанционного обучения

Глобализация дистанционного образования предоставляет широкие возможности развивающимся странам для реализации систематических задач, из которых решающими являются следующие:

1. Расширенный доступ к международным образовательным программам, предлагаемые институтами в Америке, Канаде, Австралии, Великобритании, теперь доступны в большинстве развивающихся стран. Действует множество других программ, предложенных развитыми странами, которые доступны в развивающихся странах. Учащиеся в развивающихся странах имеют доступ к образовательным и учебным программам, предложенным многими развитыми странами. Теперь можно получить диплом

об образовании за рубежом, не выезжая за пределы своей страны. Многие развивающиеся страны с ограниченными образовательными ресурсами получают широкий допуск к мировым образовательным и учебным средствам, чтобы дополнить и усовершенствовать свои усилия по предоставлению образования большому количеству людей. Спутниковые технологии широко используются развитыми странами для передачи своих образовательных программ в отдаленные части стран, таких как Индия. Большая потребность существует в получении образования в технической и профессиональных сферах, таких как менеджмент, пользование компьютером, образование, мультимедиа и др. Список иностранных учреждений в Индии, Китае и Арабских странах, предлагающих дистанционные программы образования, очень впечатляющий и большой. Это повлияло на то, что у людей появилось больше возможностей для получения образования за рубежом [34].

2. Обучение с использованием чужого опыта и знаний. Благодаря современным технологиям коммуникации происходит легкий доступ к источникам информации, материалам и осуществляется быстрая связь между людьми, что способствует получению образования, используя чужой опыт и накопленные знания. Различные методы, используемые учреждениями дистанционного образования в развитии, создании и подачи материалов, становятся легко доступными посредством баз данных. Информационные технологии позволили разработать учебные материалы для дистанционного образования, опираясь на существующие знания и практику, и сделали процесс обучения более доступным. Глобализация, в этом смысле, способствует процессу получения знаний, используя существующий опыт и практику [34].

3. Взаимовыгодное партнерство.

Пословица «Два лучше, чем один» оправдывает себя как в дистанционном образовании, так и во многих других областях. Партнерские договоренности между участниками, находящимися в разных странах, что опять-таки возможно благодаря технологиям, могут внести вклад в

предоставление услуг, которые выходят за рамки какого-то одного учреждения или института. Партнерство в развитии равноправного дистанционного образования определенным образом дополнит опыт и ресурсы, и будет способствовать созданию улучшенных учебных материалов. Одним из таких примеров международного сотрудничества, возможного благодаря глобализации, является партнерство между Commonwealth of Learning, Канада, и национальным Открытым Университетом Индиры Ганди. Они предложили Masters Programme в дистанционном образовании (MADE) под партнерской программой Раджива Ганди. Сотрудничество между Commonwealth of Learning, Канада и Национальным Открытым Университетом Индиры Ганди, Индия, является одним из таких примеров международной деятельности, возможной благодаря процессу глобализации. Эти процессы повлияли также на деятельность Европейского Сообщества в развитии дистанционного образования. Партнерские соглашения могут эффективно удовлетворять потребности в образовании небольших развивающихся стран, в особенности способствовать развитию учебных материалов, которые являются основополагающей составляющей в дистанционном образовании [34].

4. Конкурентная среда

Расширенный доступ может повлиять на усиление конкурентно-способности. Например, как упоминалось ранее, некоторые из лучших мировых административных и компьютерных программ, развитых в определенных передовых странах доступны многим другим странам, включая развивающиеся. Это, предполагается, должно инициировать дух конкурентной борьбы среди местных учреждений/институтов. Технологическое преодоление границ может поставить под угрозу существование посредственных учреждений/программ. Мировые программы дистанционного образования вносят конкурентоспособность в работу местных учреждений/институтов дистанционного образования. Этот процесс может повлечь за собой улучшение организации дистанционного образования [34].

1.4 Развитие дистанционного образования в России

За последние 15 лет в России многократно возросло количество вузов, существенным образом изменились образовательные программы, улучшилось техническое оснащение образовательного процесса, появились новые методики и формы обучения. Безусловно, система высшего образования России не лишена недостатков, которые вызваны многочисленными проблемами вузов, связанными с недостатком финансирования, нехваткой помещений, лабораторного оборудования, учебной и учебно-методической литературы, оттоком квалифицированных кадров в коммерческие структуры. По оценкам специалистов, в настоящее время Россия по количеству населения с высшим образованием, приходящихся на 10 тыс. человек, находится только в четвёртой десятке стран мира. Вместе с тем, несмотря на проблемы экономического характера в нашей стране, интерес к высшему образованию остаётся на высоком уровне. Ежегодно в вузы России принимается около 1 млн. студентов. Но конкурс составляет в среднем около трёх человек на место. Следовательно, 2 млн. человек в год – это неудовлетворённый спрос на получение высшего образования [27].

На законодательном уровне в России было закреплено понятие «электронное обучение» в 2012 году [38, 42].

Под электронным обучением понимается организация образовательной деятельности с применением содержащейся в базах данных и используемой при реализации образовательных программ информации и обеспечивающих ее обработку информационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу по линиям связи указанной информации, взаимодействие обучающихся и педагогических работников. Под дистанционными образовательными технологиями понимаются образовательные технологии, реализуемые в основном с применением информационно-телекоммуникационных сетей при

опосредованном (на расстоянии) взаимодействии обучающихся и педагогических работников [42].

1.4.1 Целесообразность создания системы ДО в России

Для России необходимость и актуальность дистанционного обучения связана со многими факторами:

- огромные территории и скопление научно-технических центров в больших городах,
- формирование новых потребностей населения по отношению к содержанию и технологиям образования,
- развитие рыночной экономики,
- усиление миграции населения и др.

Развитие ДО в России позволит обеспечить населению нашей страны доступ к качественному образованию и даст России возможность завоевать определенное место на мировом рынке образовательных услуг [8].

В нашей стране есть ВУЗы, которые предлагают получить высшее или второе высшее образование дистанционно по гуманитарным, экономическим и юридическим специальностям.

На сегодня интерес к дистанционному обучению в России присутствует. Существуют успешные разработки, на базе которых могут разворачиваться системы дистанционного обучения.

Университеты, которые прилагают дистанционное обучение:

- Белгородский государственный национальный университет имеет систему <http://pegas.bsu.edu.ru>.
- Центр дистанционного обучения Московского государственного технического университета им. Н.Э. Баумана www.cdl.bmstu.ru.
- Московский государственный университет экономики, статистики и информатики (МЭСИ) www.mesi.ru.

– Российский государственный институт открытого образования (РГИОО) www.openet.ru.

1.5 Выводы по разделу

Являясь следствием объективного процесса информатизации и вбирая в себя лучшие черты других форм, дистанционное обучение вошло в XXI века как самая перспективная, гуманистическая, интегральная форма образования [24].

Сейчас ДО - предоставление каждому человеку свободного открытого доступа к образованию на протяжении всей его жизни, с учетом способностей, потребностей и интересов. Также это возможность подготовить человека к жизни в условиях быстрого изменения информационных потоков, возможной смене профессии, активной, самостоятельной деятельности и т.д.

Дистанционная форма обучения полностью обеспечивает право на получение образования и удовлетворяет принципу гуманности в том, что никто не должен быть лишен возможности учиться вследствие каких-то причин, которые не позволяют получить образование в традиционной форме:

- бедности
- географической изолированности
- социальной незащищенности
- невозможности посещать образовательные учреждения в силу физических недостатках
- занятость производственными и личными делами [2].

В мире имеется опыт для реализации систем дистанционного обучения (СДО). Дистанционное обучение уже перешло из статуса экспериментального в привычный режим, и активно используется как учебными заведениями, так и предприятиями, заинтересованными в постоянном повышении квалификации своих сотрудников [18].

Что касается нашей страны, то наличие полноценной современной системы образования позволит обеспечить населению нашей страны доступ к качественному образованию, которое признается мировым сообществом, является жизненной необходимостью.

2 Методика преподавания курса дифференциальные уравнения

2.1 Содержание курса дифференциальные уравнения

Основной задачей при изучении дисциплины “Дифференциальные уравнения” является ознакомление с методами теории дифференциальных уравнений, освоение и применение на практике следующих тем:

- Дифференциальные уравнения первого порядка
- Дифференциальные уравнения высших порядков
- Системы дифференциальных уравнений
- Операционное исчисление

Для повышения знаний по дисциплине “Дифференциальные уравнения” в курсе есть дополнительные темы:

- Устойчивость движения.
- Колебания.
- Параметрический резонанс

Содержание основных тем:

1) *Дифференциальные уравнения первого порядка* – данный раздел включает в себя основные понятия дифференциальных уравнений 1-го порядка, методы их решения и примеры практических задач. Задача Коши для дифференциального уравнения 1-го порядка. Понятие линейного дифференциального уравнения первого порядка и методы решения. Задачи Коши для линейного дифференциального уравнения 1-го порядка и ее решение [40]. Типовые примеры задач.

2) *Дифференциальные уравнения высших порядков* – данный раздел включает в себя основные понятия линейного дифференциального уравнения 2-го порядка с постоянными коэффициентами и методы его решения. Задачи Коши для линейного дифференциального уравнения 2-го порядка с постоянными коэффициентами и методы решения задачи. Понятие линейного

дифференциального уравнения n -го порядка с постоянными коэффициентами. Задача Коши для линейного дифференциального уравнения n -го порядка с постоянными коэффициентами [40]. Типовые примеры задач.

3) *Системы дифференциальных уравнений* - данный раздел включает в себя понятие системы дифференциальных уравнений. Нормальные системы дифференциальных уравнений. Задача Коши для нормальной системы дифференциальных уравнений. Решение системы линейных однородных дифференциальных уравнений первого порядка с постоянными коэффициентами. Решение задачи Коши для системы линейных неоднородных дифференциальных уравнений с постоянными коэффициентами методом вариации постоянных. Определитель Вронского и его свойства [40]. Типовые примеры задач.

4) *Операционное исчисление* - данный раздел включает в себя понятие оригинала и его изображения. Прямое и обратное преобразования Лапласа. Основные свойства преобразования Лапласа. Теоремы о линейности преобразования Лапласа, сдвиге оригиналов и изображений, дифференцировании и интегрировании оригиналов и изображений. Теорема о свёртке оригиналов и изображений. Применение преобразования Лапласа для редукции дифференциальных задач к алгебраическим задачам. Применения преобразования Лапласа для решения дифференциальных уравнений [40]. Типовые примеры задач.

5) *Устойчивость движения* - данный раздел включает в себя определение устойчивости. Дифференциальные уравнения возмущенного движения. Устойчивость по Ляпунову и некоторые другие определения устойчивости. Основные методы решения задачи устойчивости. Типовые примеры задач.

6) *Параметрический резонанс* - данный раздел включает в себя изучение параметрического резонанса и его свойства. Общее дифференциальное уравнение. Примеры механических систем с параметрическими колебаниями.

Параметрическое возбуждение по закону синуса. Области неустойчивости. Типовые примеры задач.

Все указанные выше темы должны отображаться в системе дистанционного обучения в виде уроков, а для закрепления материала должны быть сделаны тесты на веб-ресурсе.

2.2 Техническое содержание курса

Рабочая программа дистанционного обучения по дисциплине “Дифференциальные уравнения” представляет собой обязательный структурный элемент основной образовательной программы, которая включает в себя содержание, теоретические материалы, видео уроки и практические задания в виде тестов [35, 37]. Рабочая программа должна быть отображена на веб-ресурса.

2.2.1 Теоретические материалы

Теоретические материалы содержат полное изложение всех разделов и тем в соответствии с требованиями образовательного стандарта дисциплины и действующих методических указаний [35, 37].

Вначале необходимо определить глубину тематического деления всего комплекса материалов, т.е. количество иерархических уровней учебного материала. Рекомендуется вводить не более трех уровней. Центральным элементом конспекта лекций является тема. Допускается разбиение большого по объему текста темы на компактные составляющие. Необходимыми элементами каждой темы являются:

- цель изучения темы;
- текст темы (при большом объеме должен быть разбит на отдельные вопросы);

- вопросы для повторения и закрепления материала.
- вопросы и задания для самостоятельной работы [35].

Теоретические материалы должны быть размещены на веб-ресурсе в виде уроков и быть доступны для пользователей.

2.2.2 Видео уроки

Учебные видеоматериалы – это современная, эффективная форма представления учебного контента, незаменимая в условиях дистанционного образования [35, 41]. Учебные видеоматериалы представляют собой подборку учебных видеозаписей, соответствующих лекционному и практическому курсу и позволяющих организовать различные формы обучающей работы в интерактивном формате. Видеоматериалы способствуют лучшему пониманию учебного материала, за счет повышения информационной плотности, степени восприятия, эмоциональной насыщенности. Видеоматериалы используются для:

- объяснения сложных тем или вопросов курса (когда другие форматы менее информативны и понятны);
- демонстрации того, что требует дополнительной визуализации для понимания материала (внутреннее устройство, объекты микромира и т.п.);
- придания большей реалистичности и усиления эмоционального воздействия;
- создания эффекта виртуального присутствия преподавателя.

Видеоматериалы могут создаваться для дополнения конспекта лекций и как элементы практических заданий [35].

Видео уроки являются дополнительным учебным пособием к теоретическим материалам, и требуют отображения на веб-ресурсе после теоретического материала.

2.2.3 Тестовые задания

Тестовые задания являются компонентом оценочных средств, и используется для самоконтроля и контроля знаний обучающихся. Система тестирования даёт пользователю возможность не только проверить знания, но и исправить ошибки и отработать слабые места всего процесса восприятия материалов. Электронный вариант тестовых заданий разрабатывается и оформляется в соответствии с требованиями к электронным вариантам тестовых заданий [35].

Тестовые задания на веб-ресурсе должны быть отображены после уроков и доступны для пользователей.

2.3 Структура сайта

При создании веб-ресурса дистанционного обучения по дисциплине «Дифференциальные уравнения» структура сайта должна формироваться с требованиями рабочей программы.

Содержание материалов веб-ресурса:

- Теоретические материалы и видео уроки как дополнительное пособие.
- Тесты для самоконтроля изучения теоретических материалов

Программное обеспечение веб-ресурса обеспечивают следующие программные компоненты:

- Операционная система Microsoft Windows 7;
- СУБД MySQL;
- Сервер Apache;
- Язык программирования PHP;
- Программный код системы дистанционного обучения по дисциплине «Дифференциальные уравнения»

СУБД MySQL обеспечивает хранение и представление информации по запросам языка SQL. Сервер Apache служит для обеспечения связи с между ЭВМ по протоколу HTTP и взаимодействия с СУБД MySQL и языком программирования PHP. Общая структура взаимодействия элементов ПО на рисунке 2.1.

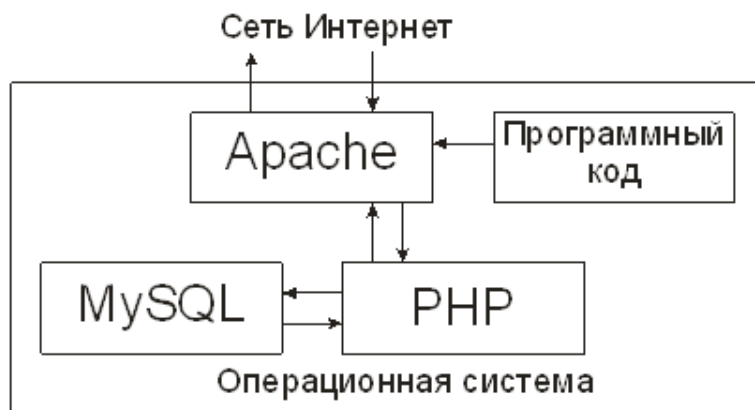


Рис. 2.1 – Общая структура взаимодействия элементов системы

Программный код должен быть реализован на гипертекстовом языке разметки HTML и языке программирования PHP.

Данная структура системы позволит обеспечить прием, хранение, обработку и представление информации в автоматическом режиме и организовать доступ пользователю к информации через Интернет.

Структура сайта:

Главная страница дистанционного обучения по дисциплине «Дифференциальные уравнения» представлена на рисунке 2.2.



Рис. 2.2 – Главная страница

Страница пользователя представлена на рисунке 2.3.

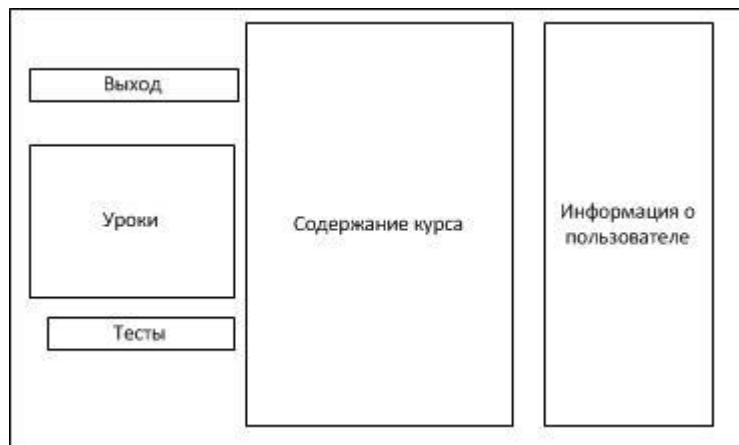


Рис. 2.3 – Страница пользователя

Страница администратора представлена на рисунке 2.4.

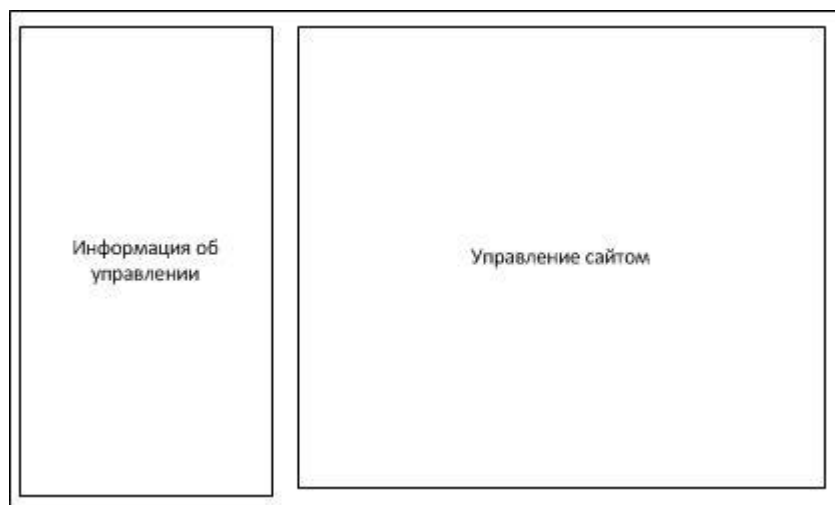


Рис. 2.4 – Страница администратора

2.4 Выводы по разделу

Составлено содержание курса "Дифференциальные уравнения" с учетом действующих методических материалов, сформулирована структура основных тем курса и форма представления теоретических и практических материалов. Разработаны требования к системе дистанционного обучения по дисциплине "Дифференциальные уравнения" и выбрана и определена структура сайта для дистанционного представления материалов по курсу.

3 Разработка требований к системе дистанционного обучения

3.1 Анализ структурно-функциональных особенностей системы дистанционного обучения

Основная цель выполнения данной работы заключается в создании системы дистанционного обучения дисциплине "Дифференциальные уравнения".

Идея работы заключается в использовании современных аппаратных и программных средств для предоставления возможности самостоятельного обучения по дисциплине "Дифференциальные уравнения".

Подобная система должна обеспечить пользователю возможность получить теоретическую информацию по дисциплине, приобрести практические навыки решения задач, проверить свои знания по дисциплине с помощью тестов в удобное время в удобном месте и в удобном темпе.

Система дистанционного обучения состоит из:

- инструктивного блока;
- информационного блока (информационное наполнение ресурса)
- контрольного блока (механизма тестирования и оценки);
- управляющей системы, объединяющей все это воедино.

Таблица 3.1– Схема системы онлайн-обучения

Инструктивный блок	Информационный блок	Контрольный блок	Коммуникативный блок
УПРАВЛЯЮЩАЯ СИСТЕМА			

3.1.1 Инструктивный блок

В инструктивном блоке дается краткая характеристика курса: для кого он предназначен, что необходимо знать и уметь для успешного усвоения материала, цели и задачи курса, аннотация курса, организация курса, необходимая литература, порядок обучения, приводятся методические указания, как работать с данным курсом, место и взаимосвязь с другими дисциплинами [13].

3.1.2 Информационный блок

При создании курса ДО важно учесть особенности целевой группы, для которой создается этот курс, поскольку успешность ДО во многом зависит от организации учебного материала. Если курс предназначен для обучения при взаимодействии преподавателя и студента, то соответственно и требования к организации такого курса, принципы отбора и организации, структурирования материала будут определяться особенностями этого взаимодействия. Если курс предназначен для самообразования, как в нашем случае, то отбор материала, его структурирование и организация будут существенно другие [13].

Одно из условий разработки курса ДО является модульность. Модули – это относительно небольшие, логически замкнутые части (разделы), включающие четко обозначены объемы знаний и умений, или зачетная единица, качество работы которой фиксируется контрольными работами, тестовыми, зачетными и экзаменационными средствами [13, 39].

Главным требованием к построению такой структуры: логичность выделения структурной единицы, наличие для учащегося возможности прямой навигации из любой структурной единицы в другую, логически с ней связанной. Это позволяет пользователю самому формировать учебную программу, соответствует одному из главных принципов

ДО – ориентированности на потребности пользователя (индивидуализации процесса обучения).

С учетом этого требования курс дистанционного обучения дисциплине "Дифференциальные уравнения" был разбит на модули, содержащие информацию по небольшим, логически завершенным темам. Все уроки занесены в соответствующую таблицу базы данных для исключения возможности изменения учебного материала не уполномоченными лицами.

Кроме того, для наилучшего представления материала следует использовать средства мультимедиа. Это комплекс аппаратных и программных средств, позволяет человеку общаться с компьютером, используя самые разные, естественные для себя среды: звук, видео, графику, тексты, анимацию и др. Мультимедиа открывает значительные возможности иллюстрации изучаемого явления. Это повышает качество образования и позволяет удерживать внимание учеников [13].

3.1.3 Контрольный блок

Система дистанционного обучения по дисциплине "Дифференциальные уравнения" рассчитана на самостоятельное изучение представленного материала. Поэтому блок контроля усвоения материала имеет вид тестов, с помощью которых пользователь системы может проверить свои знания теоретических основ курса. При этом результат тестирования (оценка в баллах) выводится сразу и позволяет сделать выводы об уровне усвоения материала. В случае неудовлетворенности оценкой пользователь может вернуться к соответствующим урокам, пополнить недостающие знания или повторить материал, а затем снова пройти тест.

По результатам всех тестов выставляется итоговая оценка.

Тесты, так же как и уроки, занесены в базу данных, что, кроме всего прочего, исключает возможность пользователя "подсмотреть" правильные ответы [13].

3.1.4 Управляющая система

Управляющая система, которая включает пользовательский интерфейс, должна обеспечить: взаимодействие всех вышеупомянутых блоков; наполнения, хранения и представления информации с помощью системы управления реляционными базами данных; защиту информации, имеющихся в системе; обработку результатов тестов и выставления ученикам оценок. Для продуктивной работы пользователей система должна иметь понятный, простой в освоении и удобный интерфейс. Неудачность интерфейса приводит, как показывает практика, к малой эффективности процесса обучения. Именно эта причина во многих случаях служит основанием для отказа от автоматизированных технологий в образовании [5]. Поэтому разработка пользовательского интерфейса является очень важным этапом создания системы дистанционного обучения.

Для наполнения базы данных необходимо реализовать страницу администратора, на которой он сможет редактировать, удалять и создавать новые уроки и тесты. Доступ к этой странице должен иметь только администратор системы, следовательно, необходимо реализовать разграничение доступа.

Для пользователя необходимо предусмотреть страницу, где он сможет просмотреть свои оценки по тестам.

Таким образом, перечень задач, которые необходимо решить в процессе работы над системой дистанционного обучения по дисциплине "Дифференциальные уравнения", следующие:

- сбор информации для учебных материалов по всем разделам;

- систематизация, редактирование и подготовка материалов уроков;
- формирование тестов;
- разработка структуры интерфейса и базы данных;
- разработка алгоритмов взаимодействия модулей системы;
- создание и наполнение базы данных системы;
- разработка алгоритмов администрирования базы данных;
- разработка системы информационной безопасности системы;

Задачи интерфейса разрабатываемой системы:

- обеспечение пользователю доступа к информации с помощью стандартных типов браузеров;
- обеспечение доступа к странице администрирования только пользователю с правами администратора;
- хранение информации с помощью системы управления реляционными базами данных MySQL;
- сортировка и представление информации в виде HTML по запросам пользователя;
- простота эксплуатации, интуитивность.

3.2 Анализ аппаратного и программного обеспечения системы

3.2.1 Операционная система

Операционной системой была выбрана ОС Microsoft Windows 7. На сегодняшний день Microsoft Windows 7 одна из популярных операционных систем в мире [43].

Операционная система Microsoft Windows 7 простая в установке и настройке, не требует специальных знаний и навыков по установке и настройке и поддерживает все необходимые программные продукты для реализации

системы дистанционного обучения по дисциплине: "Дифференциальные уравнения" [43].

3.2.2 Язык разметки гипертекстовых страниц HTML

Для обработки текстов уроков был выбран язык разметки гипертекстовых страниц HTML. Hypertext Markup Language является языком, разработанным специально для создания Web-документов. Язык HTML определяет синтаксис и размещение специальных инструкций (тегов), которые не выводятся на экран, но указывают браузеру, как отображать содержимое документа. Он также используется для создания ссылок на другие документы, локальные или сетевые [15].

Документы HTML являются обычными текстовыми ASCII-файлами. Это означает, что для их создания можно использовать любого текстового редактора, даже с минимальными возможностями. Существуют средства редактирования, разработанные специально для написания HTML. Они позволяют экономить время, так как содержат клавиши быстрого доступа для выполнения повторяющихся операций, например, установление начальных установок документов, таблиц или просто применения стилей к тексту [15].

3.2.3 Программное обеспечение для создания мультимедийных компонентов системы

Средой для разработки мультимедийных компонентов был выбран программный пакет Macromedia Flash MX. Flash служит отличным инструментальным средством для выполнения векторной анимации, если при этом используется собственный формат файлов векторной графики. Цветовые и прозрачные эффекты создаются с помощью встроенной в Flash свойств заполнения промежутков, ряда видоизмененных ключевых кадров или

символов, управляемых средствами Action Script. Анимация, что синхронизируется, также может быть потоковой, благодаря чему воспроизведение Flash-фильма выполняется еще до момента его полной загрузки в Flash Player [25].

Кроме этого в Flash MX применяется надежный и стандартизованный язык объектно-ориентированного программирования Action Script. Эта среда разработки, благодаря которой возможности использования Flash MX как основного инструментального средства очень широкие [25].

3.2.4 Программное обеспечение для реализации интерфейса и базы данных

Локальным сервером был выбран Denwer, который включает в себя:

- 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, необходимо лишь добавить директорию в каталог /home, при этом править конфигурационные файлы не требуется. По умолчанию уже поддерживаются схемы именования директорий многих популярных хостеров, новые можно без труда добавить;

- система управления запуском и завершением всех компонентов Денвера;

- phpMyAdmin — система управления MySQL через Web-интерфейс;
- Эмулятор sendmail и SMTP-сервера (отладочная «заглушка» на localhost:25, складывающая приходящие письма в /tmp в формате.eml); поддерживается работа совместно с PHP, Perl, Parser и т.д.

Сервер обладает рядом преимуществ перед конкурентами:

- Denwer простой в установке и настройке

- Denwer имеет в своем пакете веб-сервер, Apache, который может использоваться как программа-координатор в работе с PHP и СУБД MySQL. Так же Apache является самым надежным и самым распространенным HTTP-сервером;

- Denwer можно установить на flash-накопитель;

- Denwer позволяет обращаться к локальным сайтам по именам, совпадающим с именем папки, расположенной в каталоге home Денвера;

- Denwer может использоваться на выбранной операционной системе Microsoft Windows 7;

- Denwer является самым распространенным локальным сервером WAMP и доступен бесплатно.

Как серверный язык создания сценариев из списка проанализированных (PHP, Microsoft Active Server Pages, Allaire Cold Fusion и т.д.), был выбран язык PHP, поскольку имеет ряд преимуществ:

- PHP входит в пакет Denwer, а следовательно прост в установке;

- как показывают сравнительные тесты, проведенные независимыми компаниями, PHP обладает более высокой производительностью по сравнению с конкурирующими продуктами;

- имеет интерфейсы ко многим базам данных, в том числе СУБД MySQL;

- встроенные библиотеки (для выполнения множества общих задач, связанных с Web)

- пакет PHP распространяется бесплатно;

- простота изучения и использования;

- переносимость (PHP можно использовать под управлением множества различных операционных систем);

- доступность исходного кода (открытый код).

Системой управления реляционными базами данных из списка всевозможных СУБД (MySQL, Postgre SQL, Microsoft SQL Server, mSQL и т.д.),

была выбрана СУБД MySQL. Система рассчитана на большое количество пользователей, это многопоточный сервер, который обеспечивает быстрый доступ к данным и гарантирует предоставление только санкционированного доступа. Так же как и PHP, СУБД MySQL входит в пакет Denwer. Превосходства над конкурентами:

- СУБД MySQL обеспечивает высокий уровень надежности хранения и защиты информации;
- как показывают сравнительные тесты, проведенные независимыми компаниями СКРБД MySQL обладает высокой производительностью и работает значительно быстрее конкурирующих продуктов;
- пакет MySQL распространяется бесплатно;
- MySQL простой в установке;
- использует стандартный язык запросов SQL;
- MySQL может использоваться на выбранной нами операционной системе;
- исходный код MySQL можно корректировать (согласно лицензии на программное обеспечение Open Source).

Выбранное программное обеспечение должно выполнять следующие основные функции:

- представление и хранения данных в базе данных;
- выполнение скриптов на языке PHP;
- координация работы программных модулей;
- передачу данных пользователю с помощью сети Интернет.

3.2.5 Аппаратное обеспечение системы

После анализа программного обеспечения и задач, выполняемых системой в качестве технического обеспечения, было решено использовать

ПЭВМ Samsung R540 с тактовой частотой работы процессора не ниже 2.40 ГГц, включающей:

- сетевую карту;
- жесткий диск объемом не менее 500 Гб;
- Объем оперативной памяти 3Гб;
- Размер экрана 15,6.

На компьютере было установлено лицензионное программное обеспечение:

- Операционная система Microsoft Windows 7;
- Flash player;
- Denwer, который включает в себя сервер Apache, PHP, MySQL и др.

3.3 Выводы по разделу

Выполнен анализ структурно функциональных особенностей технологий дистанционного обучения, сформулированы требования к программному и техническому обеспечению системы, исходя из решения задач, связанных с дистанционным обучением студента. Разработаны требования к аппаратному и программному обеспечению системы дистанционного обучения по дисциплине "Дифференциальные уравнения". Выбран локальный сервер, который включил в себя веб-сервер Apache, серверный язык PHP, реляционную базу данных СУБД MySQL.

4 Проектирование базы данных и интерфейса пользователя системы дистанционного обучения

4.1 Выбор инструментальных средств для разработки базы данных системы

Основные идеи современной информационной технологии базируются на концепции, согласно которой данные должны быть организованы в базы данных для удовлетворения информационных потребностей пользователей. Эти базы данных создаются и функционируют под управлением специальных программных комплексов – систем управления базами данных [14].

Увеличение объема и структурной сложности хранимых данных, расширение круга пользователей информационных систем привели к широкому распространению удобных и сравнительно простых для понимания реляционных (табличных) СУБД. Для обеспечения одновременного доступа к данным множества пользователей, нередко расположенных достаточно далеко друг от друга и от места хранения баз данных, созданные сетевые версии СУБД. В них тем или иным путем решаются специфические проблемы параллельных процессов, целостности и безопасности данных, а также санкционирования доступа [14].

Понятно, что совместная работа пользователей в сетях с помощью унифицированных средств общения с базами данных возможна только с помощью стандартного языка манипулирования данными, владеющей средствами, необходимыми для реализации вышеперечисленных возможностей. Таким языком стал SQL, разработанный в 1974 году фирмой IBM для экспериментальной реляционной СУБД System R [14].

SQL (Structured Query Language) – в переводе на русский структурированный язык запросов, наиболее распространенный язык для

работы с базами данных. Он дает возможность пользователям создавать и взаимодействовать с реляционными базами данных.

В качестве СУБД был выбран MySQL. На сегодняшний день она является самой известной, надежной и быстрой из всего семейства существующих СУБД. Одной из причин выбора MySQL это её доступность и распространенность, другая причина высокая производительность.

Кроме того MySQL написаны для многих операционных систем. MySQL особенно распространена на платформах Linux и Windows.

Принцип работы СУБД MySQL аналогичен принципу работы любой СУБД, использующей SQL как командный язык для создания, удаления баз данных, таблиц, для пополнения таблиц данными, для осуществления выборки данных [17].

MySQL, как и любая другая СУБД представляет собой программу-сервер, находящийся в памяти компьютера и обслуживает TCP порт. В случае с MySQL, номером порта будет число 3306. А клиентская программа, будь то CGI-приложение на Perl или программный продукт на C, соединяется с СУБД через этот порт и посылает ей запрос на SQL. И, в свою очередь, их интерпретирует, выполняя необходимые действия, и отправляет результаты запроса клиенту. Таким образом происходит общение сервера баз данных с клиентскими программами [12].

Еще MySQL имеет развитую систему доступа к базам данных.

Пользователю может быть предоставлен доступ ко всей базе данных, отдельным таблицам или отдельным столбцам таблиц. Есть разграничения прав на действия, которые может делать пользователь с записями [26].

Администратор сервера MySQL имеет все привилегии, а обычные пользователи – ни одной привилегии. То есть обычные пользователи не могут изменять конфигурацию сервера MySQL. Пароли хранятся в зашифрованном виде и не имеют функции обратной дешифровки [26].

Основные преимущества этой СУБД, выбранной для разработки базы данных системы дистанционного обучения, заключаются в следующем:

- Поддержка нескольких одновременных запросов;
- Оптимизация связей с присоединением большого количества данных за один проход;
- Записи фиксированной и переменной длины;
- Гибкая система привилегий и паролей;
- До 16 ключей в таблице, каждый ключ может иметь до 15 полей;
- Поддержка чисел длины от 1 до 4 байт (ints, float, double, fixed), строк переменной длины и меток времени;
- Интерфейс с языком PHP;
- Основанная на потоках, быстрая система памяти;
- Утилита проверки и ремонта таблицы (isamchk)
- Все данные хранятся в формате ISO8859_1;
- Все операции работы со строками не обращают внимания на регистр символов в обрабатываемых строках;
- Все поля имеют значение по умолчанию;
- Легкость управления таблицами [20].

4.2 Определение содержательной структуры базы данных системы

После проведения анализа информации, необходимой для работы системы, была определена содержательная структура базы данных.

Во-первых, надо хранить информацию о пользователях, а именно:

- логин;
- пароль;
- электронный адрес;
- идентификатор уровня доступа (администратор или обычный пользователь)

- результаты тестирования.

Во-вторых, необходимо занести в базу данных учебные материалы уроков. А также информацию, которой оперирует блок тестирования, а именно:

- вопросы тестов;
- варианты ответов;
- отметку правильного варианта ответа.

4.3 Разработка структуры базы данных

Определение и анализ содержательной структуры БД разрабатываемой системы показали, что для хранения необходимых данных нужно составить пять таблиц:

- для информации пользователей;
- для учебных материалов;
- для номеров или названий тестов;
- для вопросов тестов и отметок правильных вариантов ответа;
- для всех вариантов ответов на вопросы тестов.

Последние три таблицы должны быть связаны между собой.

Учитывая все эти требования, была разработана база данных "diplom.db", состоящая из следующих таблиц:

- таблица "users" - содержит в себе информацию пользователей;
- таблица "lessons" - учебные материалы;
- таблица "Tests" - номера или названия тестов;
- таблица "questions" - вопросы тестов и отметки правильных вариантов ответа;
- таблица "answers" - все варианты ответов на вопросы тестов.

Структура базы данных "diplom.db" приведена на рисунке 4.1.

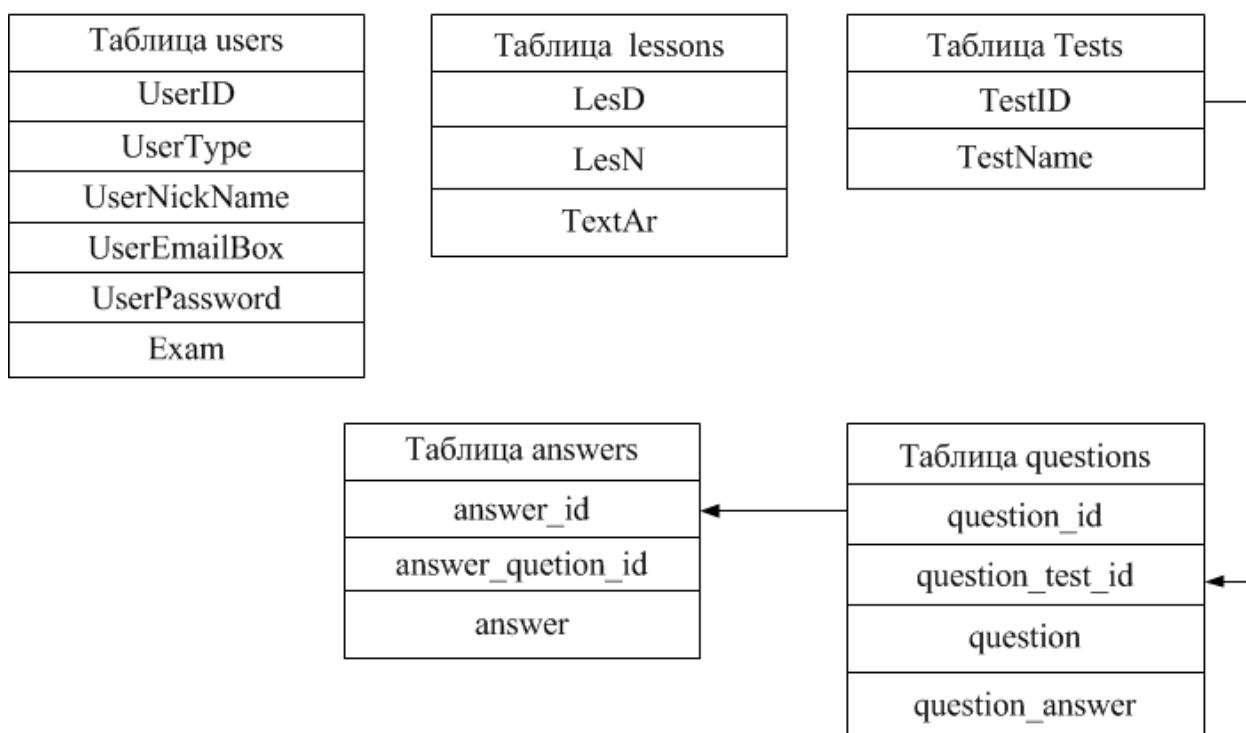


Рис. 4.1 - Структура базы данных "diplom.db"

Рассмотрим подробнее каждую из таблиц базы данных системы дистанционного обучения по дисциплине "Дифференциальные уравнения".

Таблица "users" имеет шесть полей, характеристики которых представлены в таблице 4.1.

Таблица 4.1 – таблица пользователей "users"

Название поля	Тип поля	Назначение поля
UserID	int (255)	Уникальный идентификатор записи
User Type	tinyint (1)	Идентификатор права доступа. Имеет два значения: "1" - администратор, "0" - ученик.
UserNickName	char (20)	Имя пользователя
UserEmailBox	char (40)	Электронная почта пользователя
UserPassword	char (20)	Пароль пользователя
Exam	Float (2,1)	Результаты тестирования

Эта таблица не имеет связей с другими таблицами базы данных.

Теперь рассмотрим таблицу "lessons" (таблица 4.2). Она состоит из трех полей и тоже не имеет связей с другими таблицами.

Таблица 4.2 - таблица уроков "lessons"

Название поля	Тип поля	Назначение поля
LesID	int (255)	Уникальный идентификатор записи
LesN	int (255)	Номер урока
TextAr	Text	Текст урока

Таблица "tests" имеет два поля, характеристики которых приведены в таблице 4.3. Она связана с таблицей "questions" с помощью уникальных идентификаторов тестов.

Таблица 4.3 - таблица тестов "tests"

Название поля	Тип поля	Назначение поля
TestID	int (11)	Уникальный идентификатор теста, который используется для связи с таблицей "questions"
TestName	varchar (255)	Название теста

Теперь рассмотрим таблицу "questions" (таблица 4.4), предназначенную для хранения вопросов тестов и отметок правильных вариантов ответа. Эта таблица имеет связь с таблицей "tests", чтобы сохранить информацию о том, к какому тесту относятся вопросы.

Таблица 4.4 - Таблица тестовых вопросов "questions"

Название поля	Тип поля	Назначение поля
question_id	int (8)	Уникальный идентификатор записи

Продолжение таблицы 4.4 - Таблица тестовых вопросов "questions"

question_test_id	int (8)	Это поле служит для связи с таблицей "tests" и принимает то или иное значение поля TestID, в зависимости от того, к какому тесту относится вопрос.
question	varchar (255)	Текст вопроса
question_answer	int (8)	Номер правильного ответа. Это поле служит для связи с таблицей "answers".

Наконец рассмотрим последнюю таблицу - "answers" (таблица 4.5), предназначенную для хранения всех вариантов ответа на вопросы тестов. Она имеет в своем составе три поля и связана с таблицей "questions" с помощью поля "answer_question_id", которое, в зависимости от того, к какому вопросу теста принадлежит вариант ответа, принимает то или иное значение поля "question_answer" таблицы "questions".

Таблица 4.5 - Таблица вариантов ответа на вопросы тестов "answers"

Название поля	Тип поля	Назначение поля
answer_id	int (8)	Уникальный идентификатор записи
answer_question_id	int (8)	Это поле служит для связи с таблицей "questions" и принимает то или иное значение поля "question_answer", в зависимости от того, к какому вопросу относится вариант ответа.
Answer	varchar (255)	Текст варианта ответа

4.4 Формирование запросов

С помощью языка PHP были сформированы запросы к базе данных diplom.db, необходимые для работы системы дистанционного обучения дисциплине "Дифференциальные уравнения".

При регистрации нового пользователя надо проверять, не совпадает его логин с логином уже зарегистрированного пользователя. Для этого в системе ДО был сформирован следующий запрос к базе данных:

```
$db->query("SELECT* FROM `users` where UserNickName
='".$_POST['UserNickName'];
```

В этом запросе "UserNickName" - поле в таблице "users" базы данных системы, а переменная \$_POST ['UserNickName'] содержит логин, введенный новым пользователем. Если совпадений логинов нет, информация о пользователе заносится в базу данных. Для этого используется следующий запрос:

```
$db->query("INSERT into users values ('NULL',' ".$UserType. "',"
".$_POST['UserNickName']."',".$_POST['UserEmailBox']."',".$_POST['UserPasswor
d']."'").
```

Чтобы вывести все список уроков, занесенных в базу данных, используется следующая последовательность запросов:

```
$db->query("SELECT * FROM `lessons`").Запрос к таблице "lessons".
После получения ответа формируется ссылка на каждый из найденных уроков:
if(!$db->result['is_error']) {
$db->get_query_result();
for($i = 0; $i < $db->result['count']; $i++) {
echo "<A href=\"?act=viewles&lesid=".$db->result['data'][$i]['LesID'].\">
Урок №".$db->result['data'][$i]['LesN'].\"</A><BR>".
```

По аналогии построены запросы для формирования ссылок на тесты.

Запрос для вывода учебных материалов избранное урока выглядит так:

```
$db->query("SELECT * FROM `lessons` WHERE `LesID`='".
$_GET['lesid'].");
```

```
if(!$db->result['is_error']) {
$db->get_query_result();
for($i=0; $i<$db->result['count']; $i++) {
echo $db->result['data'][0]['TextAr'];
```

Запрос для вывода результатов тестирования пользователя (оценки) выглядит следующим образом:

```
$db->query("SELECT * FROM `users` where UserNickName =
".$_SESSION['student'].");
```

Если ошибки при запросе не было, то

```
if(!$db->result['is_error']) {
$db->get_query_result();
echo $db->result['data'][0]['Test1'];
echo $db->result['data'][0]['Test2'];
echo $db->result['data'][0]['Exam'];}
```

Для проведения идентификации и аутентификации был сформирован такой запрос к базе данных:

```
$db->query("SELECT * FROM `users` where UserNickName = '".
$_POST['UserNickName']."' and UserPassword = '".$_POST['UserPassword'].");
```

Запрос для удаления уроков из базы данных выглядит следующим образом:

```
$db->query("DELETE FROM `lessons` where LesID = '$DelLesList'", );
```

Подобный вид имеет запрос для удаления пользователя.

Удалить тест немного сложнее, ведь надо удалить информацию из трех таблиц. Для реализации такого удаления была сформирована такая последовательность запросов:

```
$db->query("SELECT * FROM `questions` WHERE question_test_id =
".$_POST['DelTestList'].");
```



```

if(!$db->result['is_error']) {
    $db->get_query_result();
    $search_i = $db->result['count'];
    for($i = 0; $i < $search_i; $i++) {
        $db->query("SELECT * FROM `questions` WHERE question_test_id =
''.$_POST['DelTestList'].''");
        if(!$db->result['is_error']) {
            $db->get_query_result();
            $search = $db->result['data'][$i]['question_id'];
            $db->query("SELECT * FROM `answers` WHERE answer_question_id =
'$search'");
            if(!$db->result['is_error']) {
                $db->get_query_result();
                $search_j = $db->result['count'];
                for($j = 0; $j < $search_j; $j++)
                    $db->query("DELETE FROM `answers` WHERE answer_question_id =
'$search' LIMIT 1");
            } } }
            $db->query("DELETE FROM `questions` WHERE question_test_id =
''.$_POST['DelTestList'].''", );}
            $db->query("DELETE FROM `Tests` WHERE testid =
''.$_POST['DelTestList'].''");}

```

Сначала в таблице "tests" ищется уникальный идентификатор выбранного для удаления теста. Затем в поле "question_id" таблицы "questions" ищется запись, значение которого совпадает с уникальным идентификатором выбранного теста, а также запоминаются соответствующие записи поля "question_test_id". Следующим шагом будет поиск в поле "answer_question_id" таблицы "answers" записей, совпадающих с выбранными на предыдущем этапе записями поля "question_test_id". Так что сначала удаляются варианты ответа на

вопрос выбранного теста. Следующими удаляются вопросы теста, а затем непосредственно название теста и его номер. То есть удаление теста идет с конца.

4.5 Выбор инструментальных средств для разработки системы

Система дистанционного обучения по дисциплине "Дифференциальные уравнения" создана с помощью среды Adobe Flash, как HTML и PHP. Для разработки базы данных и обработки запросов к ней была использована система управления базами данных (СУБД) MySQL. Связь между приложением и базой данных осуществлялся с помощью языка программирования PHP.

4.5.1 Среда Adobe Flash

Технология Adobe Flash предоставляет возможность создавать мультимедийный контент, то есть информацию, которая содержит видео, аудио, текст и анимацию [32]. Первоначальное применение программы Flash было предназначена для создания быстро загружаемой веб-анимации. Flash развивается и становится все более мощным и эффективным. И он широко используется для анимирования изображения, а так же применяют для решения более сложных интерактивных задач.

Подход Flash к разработке также облегчает создание сложных мультимедийных презентаций, при этом размеры файлов остаются небольшими, так как такие элементы, как векторы, растровые изображения и звук обычно используются в одном проекте несколько раз. Flash, благодаря своей внутренней функции Symbol Conversation позволяет создавать единственный экземпляр объекта, который можно повторно использовать вместо того, чтобы каждый раз создавать новый. Такой подход существенно уменьшает размер файлов. Анимация, что синхронизируется, также может быть

потоковой, благодаря чему воспроизведения Flash-фильма выполняется еще до момента его полной загрузки в Flash Player [31].

Кроме этого, во Flash MX применяется надежный и стандартизованный язык объектно-ориентированного программирования Action Script. Это зрелая и сложная среда разработки, благодаря которому возможности использования Flash MX как основное инструментальное средство практически безграничны.

Весьма важным свойством программного пакета Flash MX является то, что производимые с его помощью фильмы могут быть доступны на самых разных платформах: от портативных устройств к настольным компьютерам телевизионной аппаратуры. Как правило, Flash-фильмы не претерпят существенных изменений при обеспечении доступа широкой и разнообразной аудитории к их содержанию в разном контексте [25].

Поэтому для создания мультимедийных компонентов системы дистанционного обучения по дисциплине "Дифференциальные уравнения" был выбран именно эта среда - Adobe Flash.

4.5.2 Язык разметки гипертекстовых страниц HTML

Для систематизации, редактирования и подготовки модулей учебного материала была использована язык HTML.

Язык разметки гипертекстовых страниц (HTML - Hypertext Markup Language) является языком, разработанной специально для создания Web-документов. Она определяет синтаксис и размещение специальных инструкций (тегов), которые не выводятся на экран, но указывают браузеру, как отображать содержимое документа. Этот язык также используется для создания ссылок на другие документы, локальные или сетевые [15].

Документы HTML являются обычными текстовыми ASCII-файлами. Это означает, что для их создания можно использовать любой текстовый редактор, даже с минимальными возможностями. Существуют средства редактирования,

разработанные специально для написания HTML. Они позволяют экономить время, так как содержат клавиши быстрого доступа для выполнения повторяющихся операций, например, задания начальных установок документов, таблиц или просто применения стилей к тексту [15].

Документ HTML содержит текст (содержимое страницы) и встроенные теги - инструкции о структуре, внешний вид и функции содержания. Документ HTML разделяется на две основные части:

- заголовок (head);
- тело (body) [16].

Заголовок содержит такие сведения о документе, как его название и методическая информация, описывающая содержимое. В теле находится непосредственно сам содержание документа (то, что выводится в окне браузера) [16].

Каждый тег состоит из имени, за которым может идти список необязательных атрибутов. Теги и атрибуты находятся внутри угловых скобок < >. Содержание скобок никогда не выводится в окне браузера. Имя тега, как правило, является аббревиатурой его функции, что облегчает его запоминание. Атрибуты являются свойствами, которые расширяют или уточняют функцию тега. Как правило, имя и атрибуты внутри тега не чувствительны к регистру [16].

Большинство тегов являются контейнерами. Это означает, что у них есть начальный и конечный теги. Текст, находящийся в контейнере, выполняет инструкции, содержащиеся в тегах контейнера. Конечный тег имеет то же имя, что и начальный, но перед ним стоит слэш (/), который можно рассматривать как "выключатель" тега. Конечный тег никогда не содержит атрибутов [16].

Атрибуты добавляются в тег для расширения или модификации его действий. К одному тега можно добавить несколько атрибутов. В тег HTML можно помещать другие HTML-теги для осуществления действия нескольких тегов на один элемент. Это называется вложением, и, чтобы правильно его

осуществить, начальный и конечный теги вложенного тега должны обязательно находиться между начальным и конечным тегами внешнего [16].

В HTML 5 имеются следующие новые возможности:

- Геолокация — определение местоположения пользователя на карте и использование этой;
- Воспроизведение медиафайлов;
- Локальное хранилище – позволяет сайтам сохранять информацию на локальном компьютере и обращаться к ней позже;
- Web Workers — стандартный способ запуска JavaScript в браузере в фоновом режиме;
- Автономные Web-приложения — Web-сайты, которые могут работать при отключении Интернета;
- Элемент Canvas — Внутри тега <canvas> с помощью JavaScript можно рисовать фигуры, линии, создавать градиенты и трансформировать объекты на лету;
- Новые элементы форм: для даты, времени, поиска, чисел, выбора цвета и др;
- Включены новые теги для разметки документа, исключены устаревшие теги и модифицированы некоторые другие [22].

4.5.3 Язык программирования PHP

При создании пользовательского интерфейса необходимо было реализовать следующие задачи:

- обеспечить взаимодействие всех блоков разрабатываемой системы;
- организовать связь с базами данных;
- обеспечить защиту информации, что есть в системе;
- организовать обработку результатов тестов и выставление оценок.
- разработать страницу администратора.

Для реализации этих задач была использована язык программирования PHP.

Первоначально аббревиатура PHP означала Preprocessor of Home Pages – препроцессор домашних страниц. Это язык внедряемых в HTML-страницы сценариев, выполняемых на сервере. По большей части его синтаксис заимствован из таких языков, как C, Perl, Java, и при этом добавлена масса возможностей, которых этим языкам не хватает [7].

PHP наделен практически полным набором функциональности, о которой (до появления PHP) мог только мечтать web-программист. Его цель – позволить максимально быстро создавать динамично генерированные web-страницы. С полным основанием можно заявить, что изучение и использование PHP будет выгодно как начинающим, так и профессиональным программистам [7].

Основными конкурентами PHP есть технологии JSP (Java Server Pages и Java Scriptlets), ASP (Active Server Pages), Perl, SSI (Server Side Includes), Cold Fusion Server Pages [7].

Рассмотрим недостатки, присущие указанным технологиям. JSP – довольно сложная для изучения и использования вещь. ASP, основанная на синтаксисе VBScript (Visual Basic), имеет всего несколько десятков собственных функций и поэтому заставляет использовать Com-объекты; кроме того, она ориентирована исключительно на работу под Windows. Perl – язык головоломный, и ее возможности затрудняют ее понимание. SSI позволяет всего лишь компоновать HTML-страницу с нескольких файлов [7].

По сравнению со своими конкурентами PHP бесспорно имеет множество преимуществ, среди которых:

- бесплатность;
- постоянное совершенствование;
- работает на UNIX и Windows платформах;
- допускает работу с большинством СУБД;
- имеет широкий набор функций (более 3 тыс.);

- допускает объектно-ориентированное программирование;
- способен использовать протоколы HTTP, FTP, SNMP, NNTP, POP3, net sockets и другие;
- позволяет выполнять все операции, и перечислены его конкуренты, а также работать с файлами графики [7].

4.6 Архитектура клиент-сервер

При проектировании структуры системы дистанционного обучения дисциплине "Дифференциальные уравнения" была использована архитектура клиент-сервер. База данных составляет одну часть архитектуры клиент-сервер. Она является «сервером», а использующий ее приложение является «клиентом». Часто клиент и сервер расположены на разных машинах и в большинстве случаев приложения клиента является дружественным интерфейсом к базе данных [30].

При создании приложения для работы с базой данных прежде всего необходимо иметь возможность связать клиента с базой данных. Поставщики баз данных предпочитают скрывать от разработчиков основополагающие механизмы связи с помощью API, ориентированных на конкретный язык. Когда создается приложение для работы с базой данных, то используются специальные библиотеки, которые транслируют запросы в пакеты TCP / IP, передаваемых по сети к серверу базы данных. Внешний вид этих API для доступа к базам данных разный и зависит от языка программирования, или и от самой базы данных [30].

Связь с языком программирования осуществляется через Common Gateway Interface (CGI – общий шлюзовой интерфейс). При разработке системы дистанционного обучения в качестве серверного языка программирования был использован PHP [30].

CGI – это набор правил, согласно которым программы на сервере могут через веб посылать данные клиентам. Спецификация CGI сопровождалась изменениями в HTML и HTTP, что вводили новую характеристику, известную как формы. Если CGI позволяет программам посылать данные клиенту, то формы расширяют эту возможность, позволяя клиенту посылать данные для этой CGI-программы [30].

В упрощенном виде архитектура клиент-сервер представлена на рисунке 4.2.

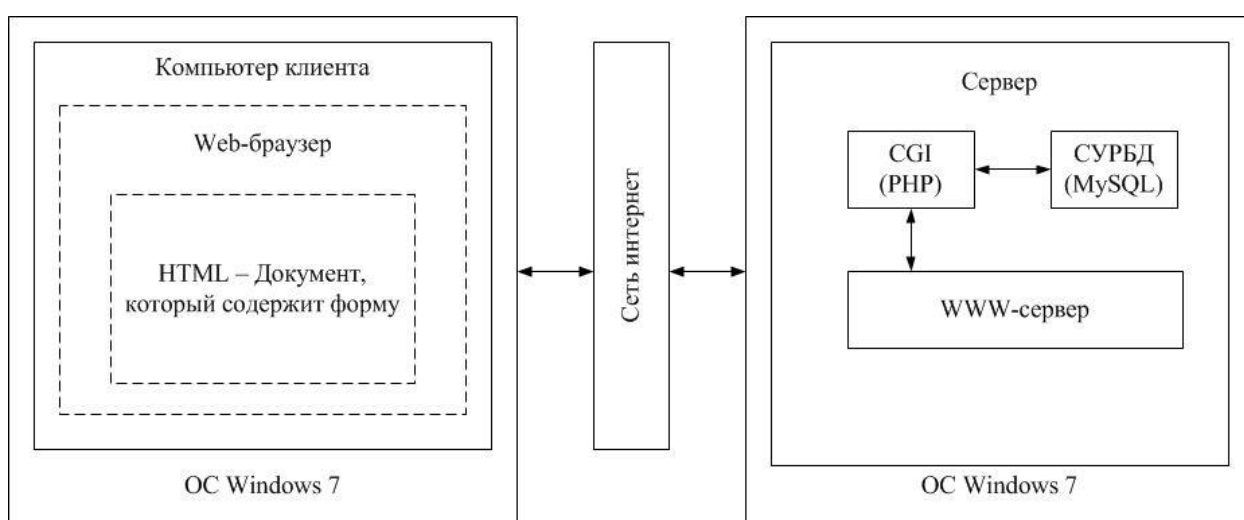


Рис. 4.2 - Архитектура клиент-сервер

Итак, система дистанционного обучения дисциплине "Дифференциальные уравнения" состоит из совокупности аппаратных и программных средств. К аппаратным средствам относятся web-сервер с комплектующими.

Рабочие места не входят в состав системы дистанционного обучения. Рабочее место – это персональный компьютер, подключенный к сети. Доступ к серверу осуществляется по IP-адресу.

Программное обеспечение системы включает следующие пакеты программ:

- СУРБД MySQL;

- Язык программирования PHP;
- Сервер Apache;
- Adobe Flash;
- Операционная система Microsoft Windows 7.

4.7 Структура системы дистанционного обучения

При разработке любых систем большое внимание следует уделить интерфейса программного продукта. Хорошо разработанный интерфейс приносит пользователям и разработчикам экономию времени и средств. Пользователи системы получают больше пользы, если они потратят меньше времени, чтобы научиться использовать приложение, а затем при функционировании им будет нужно меньше времени для выполнения работы. Дополнительные услуги для пользователя будут отражены в их отношении к приложениям.

Модули структуры системы дистанционного обучения дисциплине "Дифференциальные уравнения" и связи между ними представлены на рисунке 4.3.

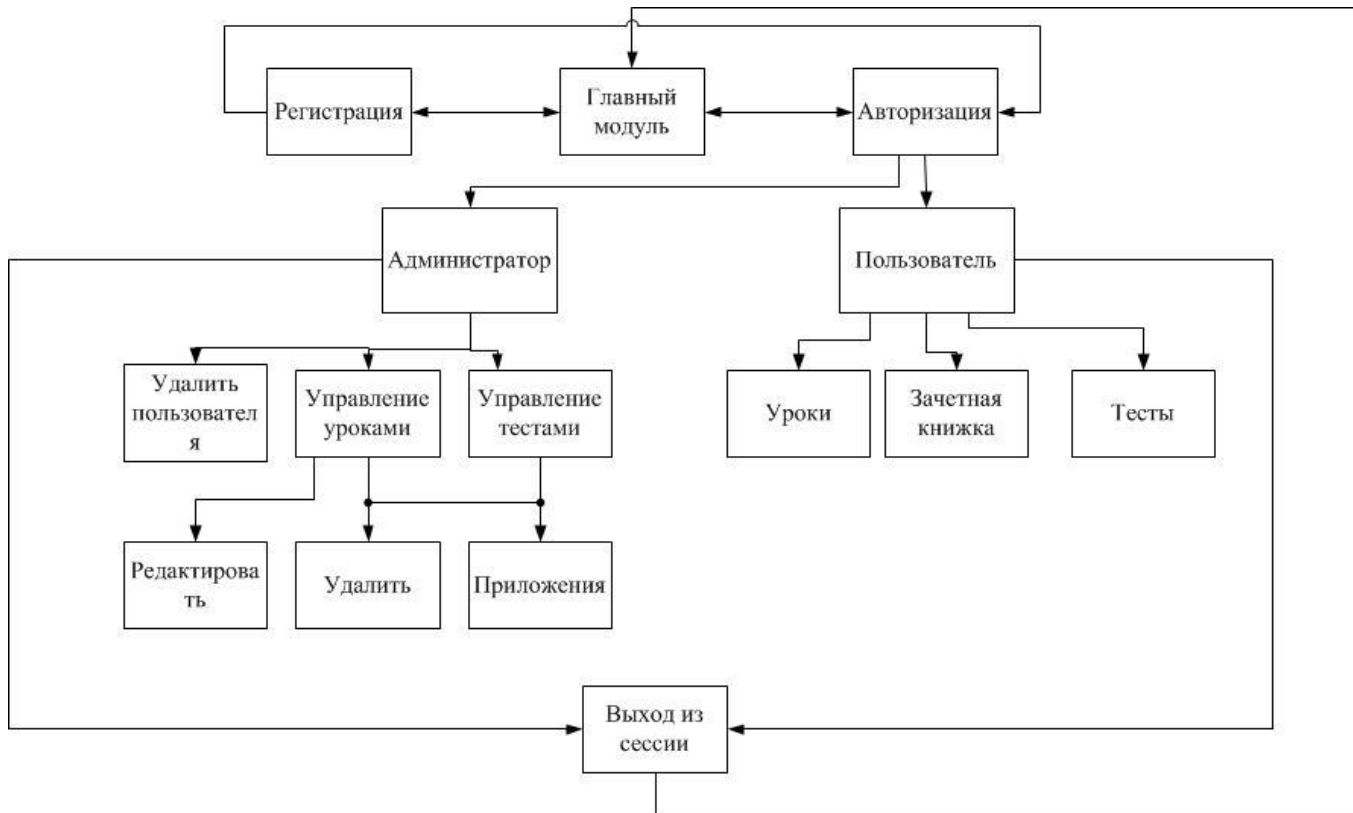


Рис. 4.3 - Структура системы дистанционного обучения

4.7.1 Модуль регистрации

Чтобы войти в систему, пользователь должен сначала зарегистрироваться. Блок-схема модуля регистрации представлена на рисунке 4.4.

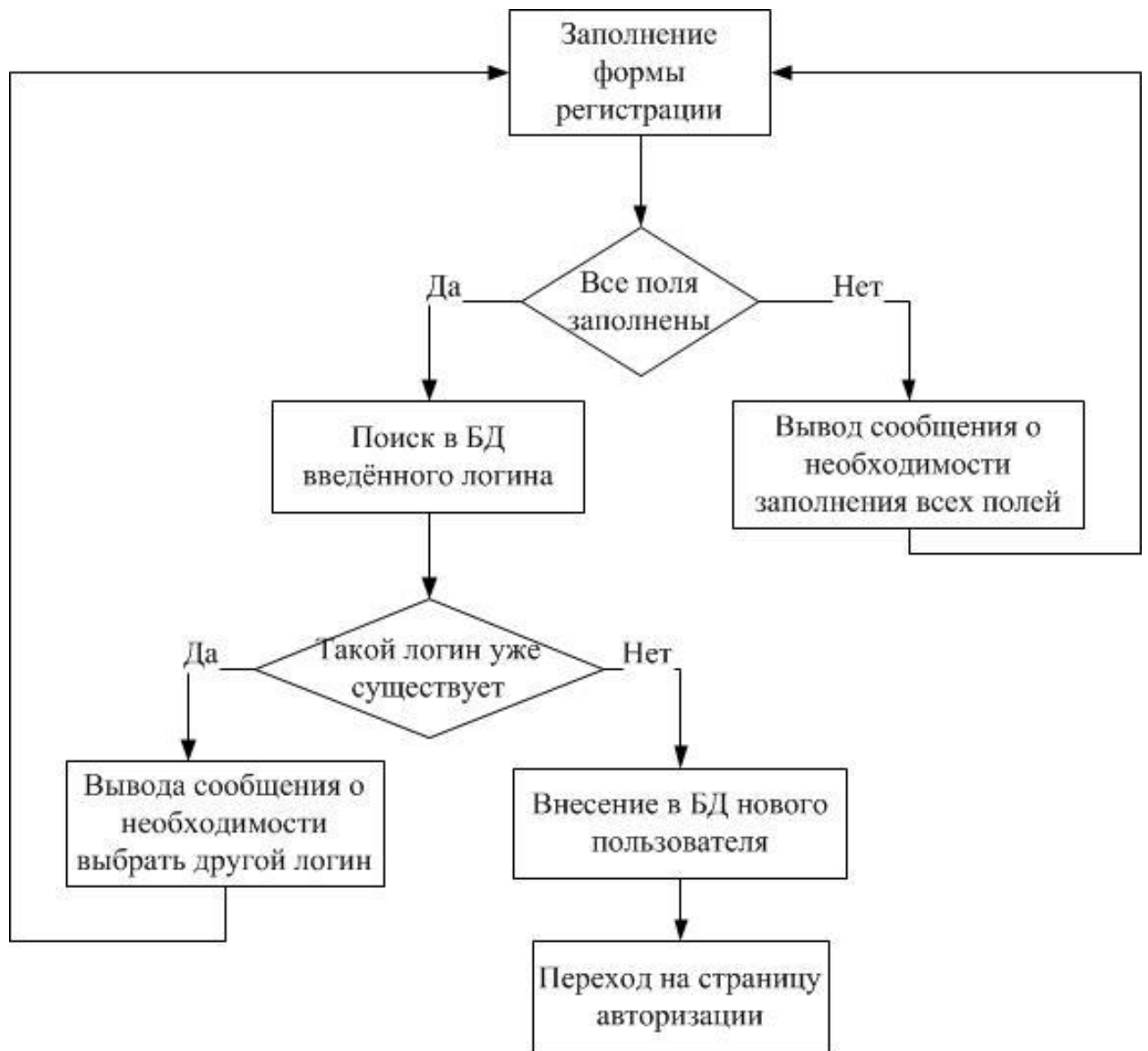


Рис. 4.4 - Блок-схема модуля регистрации

В процессе регистрации ему присваивается уникальный идентификатор (логин), для подтверждения которого при аутентификации надо ввести удостоверение (пароль). Для этого с главной страницы надо перейти на страницу регистрации и заполнить специальную форму (рисунок 4.5).

Заполните, пожалуйста регистрационную форму.
Поля, отмеченные *, обязательны для заполнения.

Логин: *	<input type="text" value="test"/>
Адрес e-mail:	<input type="text" value="dip@dip.ru"/>
Новый пароль: *	<input type="password" value="...."/>
Подтвердите пароль: *	<input type="password" value="...."/>
<input type="button" value="Назад"/>	<input type="button" value="Зарегистрироваться"/>

Рис. 4.5 - Регистрационная форма

Когда пользователь нажимает кнопку "Зарегистрироваться", посылается запрос в базу данных в случае существования логина (имени), который хочет присвоить себе пользователь. Если пользователь с таким логином уже существует, выводится сообщение с просьбой выбрать себе другой (рисунок 4.6).

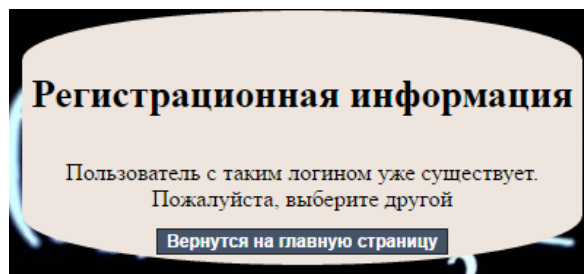


Рис. 4.6 - Сообщение о необходимости выбрать другой логин

Если поля регистрационной формы вообще не заполнены, или поле, отмеченное специальной пометкой, пустое, выводится сообщение с просьбой заполнить обязательные поля (рисунок 4.7).

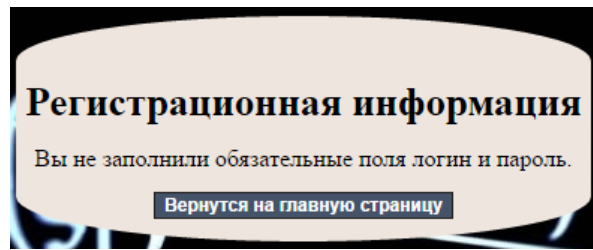


Рис. 4.7 - Сообщение о необходимости заполнить поля формы

Если все заполнено верно и совпадений логинов нет, информация, введенная пользователем, заносится в таблицу users базы данных системы.

4.7.2 Модуль аутентификации и авторизации

После регистрации пользователь возвращается на главную страницу, где должен нажать кнопку войти в систему, затем ввести в соответствующие поля свои логин и пароль, а затем нажать кнопку "Войти" (рисунок 4.8).

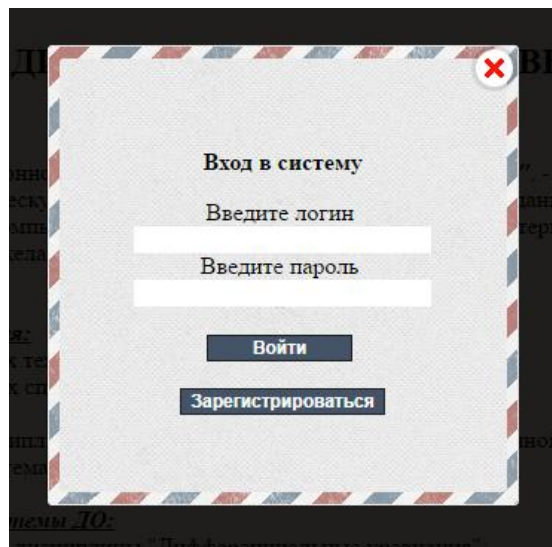


Рис. 4.8 - Форма авторизации

Начинает действовать модуль авторизации, который также включает в себя идентификацию и аутентификацию. Блок-схема этого модуля представлена на рисунке 4.9.

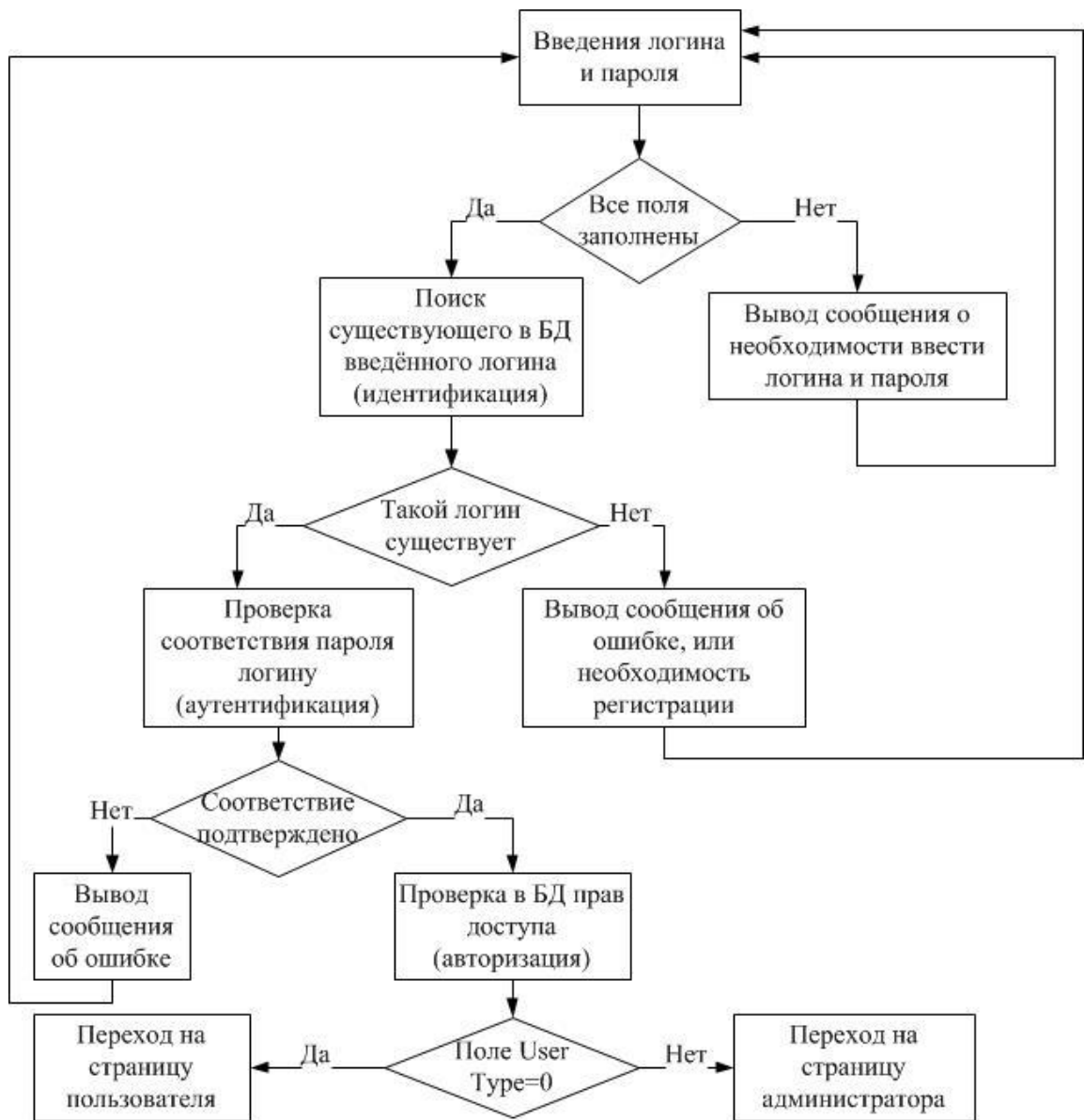


Рис. 4.9 - Блок-схема модуля авторизации

Если хоть одно поле не заполнено, выводится сообщение, представленное на рисунке 4.10.

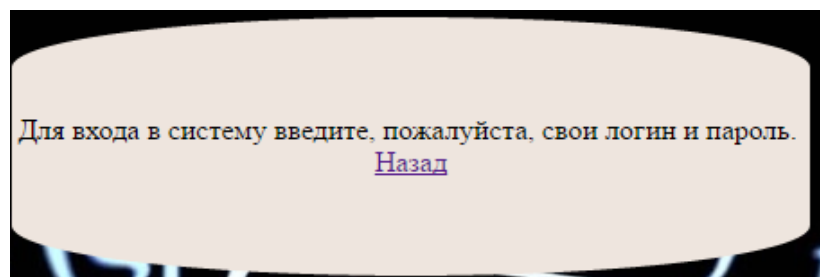


Рис. 4.10 - Сообщение модулю авторизации

Если поля заполнены, проверяется наличие введенного логина в базе данных системы. В случае, когда пользователя с таким логином не существует, выводится сообщение "Пользователь не найден".

После проверки логина (процесс идентификации) начинается процесс аутентификации пользователя, то есть проверяется соответствие введенного логина и пароля. Если соответствие не подтвержден, выводится соответствующее сообщение. Иначе – начинается процесс непосредственно авторизации.

Авторизация – это разграничение доступа к ресурсам на основе аутентификации и полученной информации об объекте.

Для системы дистанционного обучения предусмотрены два вида пользователей: студент и администратор. И так, если соответствие логина и пароля подтверждена, проверяется уровень доступа пользователя.

Для отметки об уровне доступа в таблице "users" было предусмотрено поле "User Type", которое может принимать только два значения:

- 1 – пользователь с правами администратора;
- 0 – обычный пользователь.

В зависимости от уровня доступа, пользователь переходит или к обычной странице пользователя системы, или страницу администратора.

4.7.3 Модуль пользователя

Войдя в систему, обычный пользователь-ученик попадает на начальную страницу, разделенную на три части. В левой части страницы находятся ссылки на уроки и тесты (рисунок 4.11). Эта часть не изменяется, если пользователь переходит к урокам или тестов.

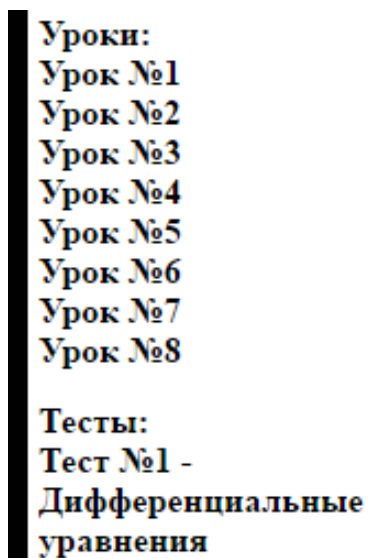


Рис. 4.11 - Левая часть страницы пользователя выбор урока

В правой части страницы представлена информация об успехах пользователя в обучении, то есть отображаются оценки по тестам и итоговая оценка - экзамен. В средней части предоставляется информация об организации курса, порядок обучения, рассказывается о том, как работать с данным курсом и так далее. В этой же части отображаются учебные материалы уроков и тесты, если пользователь переходит к ним.

Вернуться со страницы урока, или теста можно с помощью ссылки "Ваша зачетка" в левой части окна выше уроков (рисунок 4.12). Там же находится ссылка "Выход", кликнув на котором пользователь выйдет из системы обучения и попадет на главную страницу с формой авторизации.

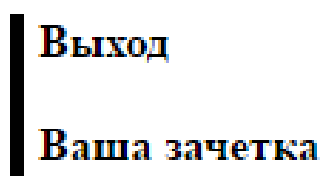


Рис. 4.12 - Левая часть страницы пользователя

4.7.4 Модуль администратора

Страница администратора отличается от страницы обычного пользователя наличием ссылки "админка" (рисунок 4.13). Это ссылка появляется только тогда, когда авторизацией подтверждено администраторские права доступа пользователя.

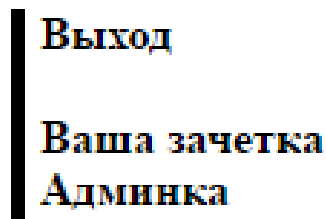


Рис. 4.13 - Ссылки "админку" на странице администратора

Это ссылка дает возможность перейти к странице, с помощью которой можно:

- удалить пользователя;
- добавить новый урок;
- удалить урок по курсу;
- редактировать существующие уроки;
- добавить или удалить тест.

Удалить пользователя можно, воспользовавшись формой, представленной на рисунке 4.14. В выпадающем списке нужно выбрать пользователя, а затем нажать кнопку "Удалить".

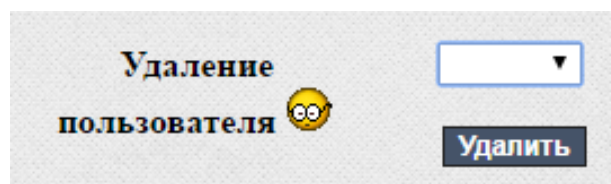


Рис. 4.14 - Форма для удаления пользователя

Формы для удаления уроков и тестов подобные формы удаления пользователя.

Для добавления нового урока используется форма с текстовыми полями, представленная на рисунке 4.15.

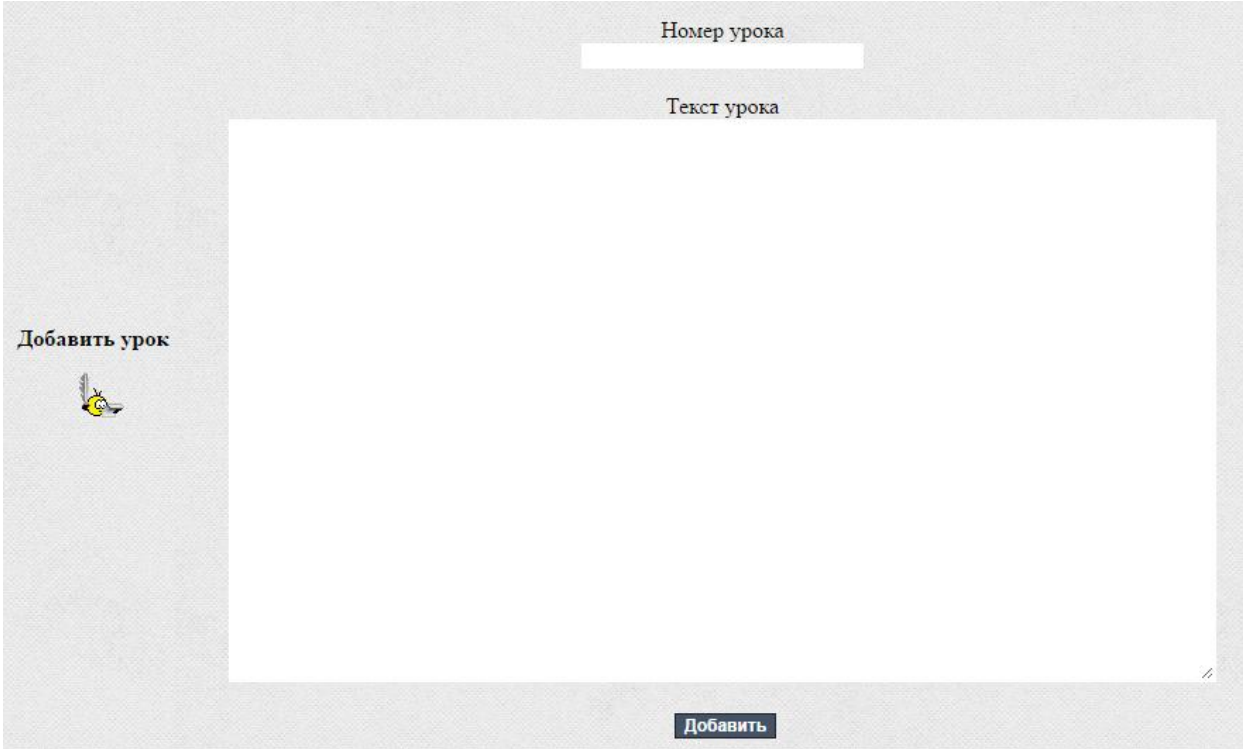


Рис. 4.15 - Форма добавления уроков

Надо ввести номер урока и текст в соответствующие поля, а затем нажать кнопку.

Для редактирования урока, надо выбрать его из выпадающего списка и нажать кнопку (рис.4.16)

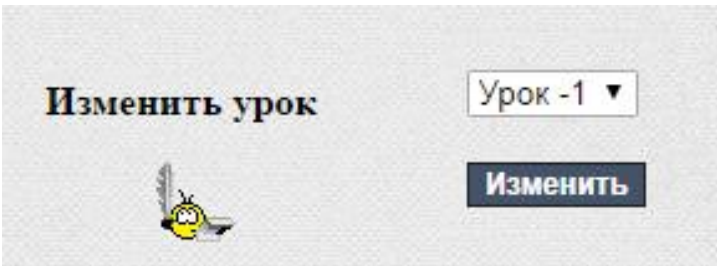


Рис. 4.16 - Форма для выбора урока, что требует изменения

После этого появится форма для редактирования урока, представленная на рисунке 4.17.

Текст урока

Изменить урок

```
<p align=center><font size=5><u><b>определение устойчивости</b></font></u><br>
<p align=justify>
<DD><DD>Понятие об устойчивости движения является непосредственным обобщением понятия устойчивости равновесия,
которое, как известно, заключается в следующем.
Рассмотрим произвольную динамическую систему с
<big><i>k</i></big>
степенями свободы, определяемую обобщенными координатами
<big><i>q<sub>1</sub>, q<sub>2</sub>...q<sub>k</sub></i></big>.
Допустим, что рассматриваемая система имеет положение равновесия, определяемое значениями <big>
<i>α<sub>i</sub></i></big>
обобщенных координат, так что уравнения движения допускают частное решение
<big><i>q<sub>i</sub>=α<sub>i</sub></i></big>.
<DD><DD>Выведем систему из положения равновесия, отклонив ее координаты на величины <big><i>ε<sub>i</sub></i></big>
</big>,
и сообщим ей начальные скорости
<big><i>ε'<sub>i</sub></i></big>,
т. е. рассмотрим движение системы, определяемое начальными условиями:
<p align=center><big>q<sub>i</sub>(t<sub>0</sub>)=α<sub>i</sub>+ε<sub>i</sub>, <br> q<sub>i</sub>
(t<sub>0</sub>)=ε'</big> <br>
<p align=justify>где (i = 1, 2, ..., k)
<DD><DD>Если для всех такого рода движений отклонения координат
<big><i>q<sub>i</sub> - α<sub>i</sub></i>,
и скорости будут все время оставаться численно меньшими сколь угодно малого положительного числа
<big><i>ε</i>,
при условии, что начальные отклонения
<big><i>ε<sub>i</sub>,
и начальные скорости
<big><i>ε'<sub>i</sub>,
численно меньше достаточно малого положительного числа
<big><i>η</i>,
```

Изменить

Рис. 4.17 - Форма для редактирования уроков

Тесты можно только добавлять или удалять.

Чтобы добавить тест, надо ввести его название, указать количество вопросов и нажать кнопку "Добавить" (рис.4.18).

Название теста

Добавить тест

Введите количество вопросов в тесте

Добавить

Рис. 4.18 - Форма для ввода названия теста и количества вопросов

После этого появится форма для добавления вопросов и вариантов ответов (рис. 4.19).

Добавить вопросы теста

Вопросы:

Вопрос 1

Ответ 1 Правильный ответ

Ответ 2 Правильный ответ

Ответ 3 Правильный ответ

Ответ 4 Правильный ответ

Ответ 5 Правильный ответ

Добавить

Рис. 4.19 - Форма для добавления вопросов и вариантов ответов

Правильный вариант ответа надо обозначить с помощью радио кнопки.

Для удаления теста надо выбрать его из выпадающего списка и нажать кнопку "Удалить". Форма для удаления теста подобная представленной на рисунке 4.20.

Удалить тест 🤖

Удалить

Рис. 4.20 - Форма для удаления теста

4.8 Выводы по разделу

После анализа информации, необходимой для работы системы дистанционного обучения по дисциплине "Дифференциальные уравнения" была определена содержательная структура базы данных.

Для разработки базы данных системы ДО использовалась СУБД MySQL, которая на сегодняшний день является самой известной, надежной и быстрой из всего семейства существующих СУБД.

Предложенная структура базы данных состоит из пяти таблиц, и имеет гибкий профиль, позволяющий расширять и модифицировать базу данных.

Запросы к базе данных "diplom.db", необходимые для работы системы дистанционного обучения по дисциплине "Дифференциальные уравнения" были сформированы с помощью языка PHP.

Обращение к БД может осуществляться с любого компьютера, имеющего доступ к сети Интернет по протоколу HTTP. Пользователи системы имеют право на осуществление каких-либо изменений базы данных. Незарегистрированные пользователи, то есть любые пользователи сети Интернет, не имеют непосредственного доступа к базе данных.

Система дистанционного обучения по дисциплине "Дифференциальные уравнения" была создана с помощью среды Adobe Flash, языков программирования HTML и PHP, а так же СУБД MySQL. Связь между приложением и базой данных осуществлялся с помощью языка программирования PHP.

Функционирующая система дает возможность учиться дисциплине "Дифференциальные уравнения" в удобное время, месте и в удобном темпе.

ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы была решена поставленная цель, связанная с изучением методов и возможностей дистанционных образовательных технологий по программам математики и разработана система дистанционного обучения по дисциплине: “Дифференциальные уравнения”

Основные выводы по работе можно изложить в следующем виде

1) Проанализированы, существующие технологии дистанционного образования и тенденции их развития. Установлено что технологии ДО стали более обширны, доступнее, появляются новые способы преподавания, такие как видеоуроки. Количество курсов и вузов предоставляющие дистанционное образование за последние 5 лет значительно увеличилось.

2) Изучена методика преподавания курса “Дифференциальные уравнения” и особенности реализации курса составлено содержание курса с учетом действующих методических материалов, сформулирована структура основных тем курса и форма представления теоретических и практических материалов. Разработаны требования к системе дистанционного обучения по дисциплине "Дифференциальные уравнения" и выбрана и определена структура сайта для дистанционного представления материалов по курсу.

3) Выполнен анализ структурно функциональных особенностей дистанционного обучения, сформулированы требования к программному и техническому обеспечению системы, исходя из решения задач связанных с дистанционным обучением студента. В качестве технического обеспечения был выбран локальный сервер, который включил в себя веб-сервер Apache, серверный язык PHP, реляционную базу данных СУБД MySQL.

4) После анализа информации, необходимой для работы системы дистанционного обучения по дисциплине "Дифференциальные уравнения" была определена содержательная структура базы данных. Разработаны

программные модули и определены их основные функции, а также основные алгоритмы системы. Система дистанционного обучения была создана с помощью среды Adobe Flash, языков программирования HTML и PHP, а так же СУБД MySQL. Связь между приложением и базой данных осуществлялся с помощью языка программирования PHP.

Функционирующая система дает возможность вести дистанционное обучение по дисциплине "Дифференциальные уравнения" студентам, которые хотят изучить данный раздел математики или углубить дополнительные знания по курсу в удобное время, месте и темпе.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Агапонов С.В., Джалиашвили З.О., Кречман Д.Л., Никифоров И.С., Ченосова Е.С., Юрков А.В. Средства дистанционного обучения. Методика, технология, инструментарий. Под ред. З.О.Джалиашвили. – СПб.: БХВ-Петербург, 2007. – 336 с.
2. Андреев А.А. "Введение в дистанционное обучение". – М.: МЭСИ, 2005г. – 420с
3. Андреев А.А., Солдаткин В.И. Дистанционное обучение: сущность, технология, организация.- М.: МЭСИ, 1999. - 196 с.
4. Ахаян А.А. Виртуальный педагогический вуз. Теория становления. - СПб.: Изд-во "Корифей", 2001. - 170 с
5. Беляев М.И., Гриншкун В.В., Краснова Г.А. Технология создания электронных средств обучения – 2007г-130с.
6. Волженина Н. В. Организация самостоятельной работы студентов в процессе дистанционного обучения: учебное пособие / Н. В. Волженина. – Барнаул : Изд-во Алт. ун-та, 2008.
7. Григин И. "РНР 4. Специальный справочник" -СПб., Питер.2002г-672с.
8. Громова Т.В. Подготовка преподавателя к деятельности в системе дистанционного обучения как ресурс повышения качества образования Вестник Северного (Арктического) федерального университета. Серия: Гуманитарные и социальные науки - Выпуск№ 3 / 2008
9. Губский Е. Г. Таракин Д. С. Особенности эффективности и применение дистанционного образования - Энергобезопасность и энергосбережение - 2008г.
10. Гусев Д.А. Заметки о пользе дистанционного обучения.
11. Зайченко Т.П. Основы дистанционного обучения: Теоретико-практический базис: Учебное пособие. - СПб.: Изд-во РГПУ им. А.И. Герцена, 2004. - 167 с.

12. М. Каба. MySQL и Perl – СПб.: Питер, 2001.
13. В. Канаво. «Методические рекомендации по созданию курса дистанционного обучения через Интернет» [Электронный ресурс]. – 2016. – Режим доступа: <http://www.curator.ru/method.html>
14. Кириллов В.В., Громов Г.Ю. Структуризированный язык запросов (SQL). Учебное пособие. - СПб.: ИТМО, 1995.
15. Крамер Н.А. “HTML: наглядный курс Web-дизайна.” – М.: Мир, 2001г. – 312с.
16. Леонтьев “Web-дизайн. Руководство пользователя”. – М.: Гостехиздат, 2003г. – 530с.
17. Литвинюк А. MySQL. Установка. Настройка. Программирование.
18. Максимов В.П., Вашакидзе Н.С., Филиппова Г.В. Дистанционные технологии обучения студентов. Интернет-журнал СахГУ: наука, образование, общество Издательство: Сахалинский государственный университет (Южно-Сахалинск) – 2010г.
19. Морозов И. О., E-learning в бизнесе: потребности, возможности, перспективы, Изд. “Академии АйТи”, 2006г.
20. Овсянников В.И. Дистанционное образование: теоретические проблемы и противоречия / В.И. Овсянников // Педагогическое образование без отрыва от основной деятельности. Ежегодник. – 1999. – № 6.
21. Паутов А.. Документация по MySQL.
22. Пилгорим М. "Погружение в HTML5" -БХВ-Петербург, 2011 – 304с.
23. Прокофьева Т.Ю. Становление и развитие дистанционного обучения в мире [Электронный ресурс] - <http://www.samoupravlenie.ru/29-09.php>.
24. Рачевский Е. Л. Информационные технологии в образовании: Школа будущего /Е. Л. Рачевский // Директор школы.- 2010.-№ 1.-С.55–58.
25. Рейнхардт Р, Лотт Дж. "Macromedia Flash MX Action Script. Библия пользователя". Пер. с англ. – М.: “Вильямс”, 2003г. – 1280с.

26. Томсон Л., Велинг Л. "Разработка Web-приложений на PHP и MySQL". Пер. с англ. – 2-е изд., испр. – СПб:ООО «ДиаСофтЮП», 2003г. – 672 с.
27. Трухачёв В.И., Банникова Н.В., Гаврилова О.Г. Активные методы в профессиональной подготовке экономистов-менеджеров: [учебное пособие] – Ставрополь: ЗАО «Диамант», 2003. – 188 с.
28. Хузиахметов А.Н., Насибуллов Р.Р. Учебная деятельность студентов вуза на этапе перехода к дистанционному образованию - журнал образование и саморазвитие Издательство: Казанский (Приволжский) федеральный университет (Казань).
29. Шахмаев Н.М. Технические средства дистанционного обучения. М. – «Знание», 2000. – 276 с.
30. Ягер Р.Д. Риз Д. Кинг Т. "MySQL и mSQL. Базы данных для небольших предприятий и Интернета" - Символ-Плюс-2000г-560с.
31. Macromedia Flash [Электронный ресурс] - <http://gitak.ru/spravochnik-vebmastera/razrabotka-sajta/550-macromedia-flash.html>.
32. Videоблог А.Суханова Простым языком о компьютерах 09.10.2014 - [Электронный ресурс]- <http://pcsecrets.ru/internet/rabota-v-internet/chto-takoe-flesh-adobe-flash.html>.
33. Дистанционное образование в мире и в России [Электронный ресурс] - <http://www.curator.ru/e-learning/publications/publication8.html>.
34. Достоинства и недостатки дистанционного обучения // "Образование: путь к успеху". - Уфа., 2010.
35. Методические рекомендации по формированию электронного учебного методического комплекса дисциплины (ЭУМКД) [Электронный Ресурс] – http://Pegas.Bsu.Edu.Ru/File.Php/1/Prilozhenie_1._Metodicheskie_Ukazaniya.Doc.
36. Перспективы развития дистанционных образовательных технологий России [Электронный ресурс] - http://www.confcontact.com/20102911/4_kisluh.htm.

37. Положение об учебно-методическом комплексе учебной дисциплины [Электронный Ресурс] - http://www.uim.ru/index.php?gid=23&Itemid=375&option=com_docman&task=doc_view.
38. Положение ФОС-НИУ БелГУ.
39. Правила разработки учебно-методического комплекса дистанционного курса [Электронный Ресурс] - <http://www.kazsu.kz/content/files/pages/folder15266/§35.pdf>
40. Рабочая программа дисциплины Дифференциальные Уравнения [Электронный Ресурс] - http://Dekanat.Bsu.Edu.Ru/F.Php/1/Programmi/Disc/2015/3086/224305_24_B3.B.2_Differencialnye_Uravneniya.Pdf
41. Учебное видео и качество обучения в вузе [Электронный Ресурс] - <http://eto.kai.ru/files/2015/08/Video.pdf>
42. Федеральный закон Российской Федерации от 29 декабря 2012 г. N 273-ФЗ. "Об образовании в Российской Федерации"
43. Чем отличается Windows 8 от Windows 7? [Электронный Ресурс] - <http://thedifference.ru/chem-otlichaetsya-windows-8-ot-windows-7/>.

ПРИЛОЖЕНИЕ 1

Листинг файла index.php

```
<?php
ob_start();
session_start();
error_reporting(E_ALL);
// Библиотеки
include("functions.php");
include("mysql_class.php");
$db = new mysql($config['db']); // Объект класса mysql
if($db->result['is_error']) {
html_header("SQL Error", "bgcolor=#FF3330");
exit($db->result['error_message']."<BR>".$db->result['verbose_error_info']);
}
if(!empty($_SESSION['student'])) {
if(empty($_GET['act'])) $_GET['act'] = "main";
if(!empty($_GET['act']) && $_GET['act'] == "exit") {
unset($_SESSION['student']);
header("Location: index.php");
exit();
}
html_header("Главная", "BACKGROUND=img/background.bmp");
?>
<table border = "0" width = "100%" height = "100%" cellspacing = "5" cellpadding
= "5">
<p align="center"><FONT color=FFFFFF><B><FONT style="FONT-SIZE: 22pt"
face="Comic Sans MS" >Система дистанционного обучения по дисциплине
"Дифференциальные уравнения"</p>
```

```

<tr>
<td class="formpoll" bgcolor="#FFFFFF" valign=top width="150">
<br>
<br><FONT color=000000><B><FONT style="FONT-SIZE: 12pt"
face=TimesNewRoman >
<a href="index.php?act=exit">Выход</a>
<br>
<br>
<a href="">Ваша зачетка</a>
<br>
<?php
$db->query("SELECT * FROM `users` where UserNickName =
'".$_SESSION['student']."' LIMIT 1 ", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
html_header("Страница администратора", "bgcolor=#80AFED");
if($db->result['data'][0]['UserType'] == 1) echo "<A
href=\"admin.php\">Админка</A>";
}
echo "<BR><BR>Уроки:<BR>";
mysql_query("SET NAMES 'cp1251'");
$db->query("SELECT * FROM `lessons`", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($i = 0; $i < $db->result['count']; $i++) {
echo "<A href=\"?act=viewles&lesid=".$db->result['data'][$i]['LesID']."\">Урок
№". $db->result['data'][$i]['LesN']."</A><BR>";
}
}
}

```

```

echo "<BR><br>Тесты:<BR>";
$db->query("SELECT * FROM `Tests`", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($i=0; $i<$db->result['count']; $i++) {
echo "<A href=\"?act=viewtest&testid=\".$db->result['data'][$i]['TestId'].\">Тест
№".($i+1)." - ".$db->result['data'][$i]['TestName'].\"</A><BR>";
}
}
echo "</TD><TD class=\"formpol\" bgcolor=\"#FFFFFF\" valign = \"top\">";
if($_GET['act'] == "viewles") {
if(!empty($_GET['lesid'])) {
$db->query("SELECT * FROM `lessons` WHERE `LesID`='".$_GET['lesid']."'
LIMIT 1", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($i=0; $i<$db->result['count']; $i++) {
echo "<B>Урок № ".$db->result['data'][0]['LesN'].\"</B><BR>";
echo "<p align = justify>";
echo $db->result['data'][0]['TextAr'];
echo "</p>";
}
} else echo "Вы не выбрали номер урока для просмотра.";
}
}
elseif($_GET['act'] == "viewtest") {
if(!empty($_POST['Submit'])) {
$db->query("SELECT * FROM `Tests` WHERE `TestID`='".$_POST['testid']."'
LIMIT 1", __FILE__, __LINE__);

```

```

if(!$db->result['is_error']) {
$db->get_query_result();
echo "<B>".$db->result['data'][0]['TestName']."</B><BR><BR>";
$questions = array();
$db->query("SELECT `question_answer` FROM `questions` WHERE
`question_test_id`='".$$_POST['testid']."' , __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$questions = $db->result['data'];
}
$good_answers = 0;
$result = array();
for($i=0; $i<$_POST['count']; $i++) {
$index = "ques".$i;
if(empty($_POST[$index]) || ($_POST[$index] !=
$questions[$i]['question_answer'])) $result[$i] = false;
else {
$result[$i] = true;
$good_answers +=1;
}
}
$user = array();
$db->query("SELECT * FROM `users` WHERE
`UserNickName`='".$$_SESSION['student']."' LIMIT 1", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$user = $db->result['data'][0];
$cost = 5 / $_POST['count'] * $good_answers;
if($_POST['testid'] == 1) {

```

```

$exam = ($cost + $user['Test2']) / 2;
$db->query("UPDATE `users` SET `Test1`='".$cost.'" WHERE
`UserID`='".$user['UserID']."' , __FILE__, __LINE__);
}
if($_POST['testid'] == 2) {
$exam = ($user['Test1'] + $cost) / 2;
$db->query("UPDATE `users` SET `Test2`='".$cost.'" WHERE
`UserID`='".$user['UserID']."' , __FILE__, __LINE__);
}
$db->query("UPDATE `users` SET `Exam`='".$exam.'" WHERE
`UserID`='".$user['UserID']."' , __FILE__, __LINE__);
echo "За тест Вам было начислено ".$cost." бала из 5<BR>";
echo "Ваш итоговый бал (экзамен) ".$exam." бала из 5<BR>";
echo "Вы ответили правильно на ".$good_answers." из ".$_POST['count']."
вопросов<BR>";
echo "<br><a href='','?'>Главная</a><br>";
}
}
} else {
$db->query("SELECT * FROM `Tests` WHERE `TestID`='".$_GET['testid']."'
LIMIT 1", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
echo "<B>".$db->result['data'][0]['TestName']."</B><BR><BR>";
?>
<br><br>
<DD><DD><p align=center><big><i>Внимательно прочитайте вопросы и,
выбрав вариант ответа, нажмите соответствующую ему радиокнопку (слева от
варианта ответа).</big></i></p>

```



```

<br><br>
<form name = "lesson" action = "?act=viewtest" method = "post" enctype =
"application/x-www-form-urlencoded">
<?php
$db->query("SELECT * FROM `questions` WHERE
`question_test_id`='".$_GET['testid']."', __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$data = $db->result['data'];
$count = 0;
for($i=0; $i<count($data); $i++) {
echo ($i+1)." ".$data[$i]['question']."<BR>";
$db->query("SELECT * FROM `answers` WHERE
`answer_question_id`='". $data[$i]['question_id']."', __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($j=0; $j<$db->result['count']; $j++) {
echo "<label><input type=\"radio\" name=\"ques\".$i.\"\" value=\"\".$db-
>result['data'][$j]['answer_id'].\"\">".$db->result['data'][$j]['answer']."</label><BR>";
}
}
$count += 1;
echo "<BR>";
}
}
?>
<input type = "hidden" name = "count" value = "<?=$count?>">
<input type = "hidden" name = "testid" value = "<?=$_GET['testid']?>">
<input type = "submit" name = "Submit" value = "Проверить">

```

```

</form>
<?php
}
}
} else {
?>
<FONT color=000000><FONT style="FONT-SIZE: 14pt"
face=TimesNewRoman>
<p align = "justify">
<font size=3>
<DD><DD>В системе дистанционного обучения по дисциплине
"Дифференциальные уравнения". Доступны следующие разделы:<br>
<i>Дифференциальные уравнения первого порядка
<i>Дифференциальные уравнения высших порядков
<i>Системы дифференциальных уравнений
<i>Операционное исчисление</i>
Для повышения знаний по дисциплины "Дифференциальные уравнения" в
курсе есть дополнительные темы:
<i>Устойчивость движения.
<i>Колебания.
<i>Параметрический резонанс.<br><br><br></i>
<DD><DD><DD><b><i><u>Содержание представленных
разделов:</b></i></u><br><br>
<DD><b><i>Дифференциальные уравнения первого порядка</b></i><br><br>
<DD><DD> - Понятия дифференциальных уравнений 1-го порядка и примеры
практических задач.
<DD><DD> - Задача Коши для дифференциального уравнения 1-го порядка.
<DD><DD> - Понятие линейного дифференциальные уравнения первого
порядка и методы решения.

```

<DD><DD> - Задачи Коши для линейного дифференциального уравнения 1-го порядка и ее решение.

<DD><i>Дифференциальные уравнения высших порядков
</i>

<DD><DD> - Понятия линейного дифференциального уравнения 2-го порядка с постоянными коэффициентами и методы их решения.

<DD><DD> - Задачи Коши для линейного дифференциального уравнения 2-го порядка с постоянными коэффициентами и методы решения задачи.

<DD><DD> - Понятие линейного дифференциального уравнения n-го порядка с постоянными коэффициентами.

<DD><DD> - Задача Коши для линейного дифференциального уравнения n-го порядка с постоянными коэффициентами.

<DD><i>Системы дифференциальных уравнений</i>

<DD><DD> - Понятие системы дифференциальных уравнений.

<DD><DD> - Нормальные системы дифференциальных уравнений.

<DD><DD> - Задача Коши для нормальной системы дифференциальных уравнений.

<DD><DD> - Решение системы линейных однородных дифференциальных уравнений первого порядка с постоянными коэффициентами.

<DD><DD> - Решение задачи Коши для системы линейных неоднородных дифференциальных уравнений с постоянными коэффициентами методом вариации постоянных.

<DD><DD> - Определитель Вронского и его свойства.

<DD><i>Операционное исчисление </i>

<DD><DD> - Понятие оригинала и изображения.

<DD><DD> - Прямое и обратное преобразования Лапласа.

<DD><DD> - Основные свойства преобразования Лапласа.

<DD><DD> - Применения преобразования Лапласа для решения дифференциальных уравнений.

<DD><i>Устойчивость движения:</i>

<DD><DD> - Определение устойчивости.

<DD><DD> - Дифференциальные уравнения возмущенного движения.

<DD><DD> - Устойчивость по Ляпунову и некоторые другие определения устойчивости.

<DD><DD> - О методах решения задачи устойчивости.

<DD><i>Колебания</i>

<DD><DD> - Колебания.

<DD><i>Параметрический резонанс:</i>

<DD><DD> - Параметрический резонанс. Общее дифференциальное уравнение.

<DD><DD> - Примеры механических систем с параметрическими колебаниями.

<DD><DD> - Параметрическое возбуждение по закону синуса.

<DD><DD> - Области неустойчивости.

<DD><DD><DD><i><u>Пожелания и напоминания:</i></u>

Время обучения в данной системе ДО не ограничено, Вы сами выбираете удобный для Вас темп работы. Однако мы советуем Вам работать регулярно, равномерно, не перегружаясь и не откладывая обучение в "долгий ящик".

Вы можете сами определять в какой последовательности изучать уроки, однако если Вы совершенно не знакомы с дисциплиной "Дифференциальные уравнения", советуем Вам проходить уроки последовательно.

Постарайтесь пожалуйста разобраться в приведенных примерах. Это поможет Вам лучше понять суть изучаемых положений.

Старайтесь "привязывать" материал курса к своей деятельности, находите примеры реализации тех или иных теоретических положений в практике Вашей работы.

При ответах на вопросы тестов не подглядывайте, пожалуйста, в учебник. Помните, что эти тесты предназначены прежде всего для самоконтроля. И не всегда у вас будет возможность подсмотреть правильный ответ, поэтому лучше выучить предлагаемый материал уроков.

`<td class="formpoll1" bgcolor="#FFFFFF" valign=top height = "100%" width="150">`

`

`

Добро пожаловать

`<?=</p>`

`$_SESSION['student']`

`?>`

`
`

`

`

Ваши результаты по:

`

`

`<?php`

`echo "<i>Первому тесту: ";`

`$db->query("SELECT * FROM `users` where UserNickName =
" .$_SESSION['student']. " LIMIT 1 ", __FILE__, __LINE__);`

`if(!$db->result['is_error']) {`

`$db->get_query_result();`

`echo $db->result['data'][0]['Test1'];`

`echo "
";`

`echo "<i>Второму тесту: ";`

`echo $db->result['data'][0]['Test2'];`

```

echo "<br>";
echo "<li>Экзамену: ";
echo $db->result['data'][0]['Exam'];
echo "<br>";
}
?>
<br><br><li><a href="kamke.djvu">Справочник по дифференциальным
уравнениям Камке</a>
<br><br><li><a href="задачи по диф уравнениям.djvu">Задачник по
дифференциальным уравнениям</a>
<br><br><li><a href="54.pdf">Учебное пособие по ДУ</a>
<br><br><li><a href=" ">Программа для решения ДУ</a>
</td>
</tr>
<?php
}
echo "</TD><TR></TABLE>";
} else {
if(empty($_POST['Auto'])) {
html_header("Авторизация", "BACKGROUND=img/background.bmp");
?>
<FONT face="Impact" color="#FFFFFF"> <p align="center">
<a1 onmouseover='this.style.cursor="pointer" ' onfocus='this.blur();'
onclick="document.getElementById('parent_popup').style.display='block'">
<span style="text-decoration: none;">Вход в систему</span></a1></p>
</FONT>
<body>
<div id="parent_popup" >
<div id="popup" >

```

```

<a class="closen" style="cursor: pointer;"
onclick="document.getElementById('parent_popup').style.display='none';">Закреть
</a>
<td>
<form name = "check" action = "index.php" method = "post" enctype =
"application/x-www-form-urlencoded">
<table align=center bordercolor = "blue" border = "0" width = "150" height = "100%"
cellspacing = "0" cellpadding = "0">
<tr>
<td valign = "top" height = "100">
<div align = "center">
<br><br><br>
<p><b>
    Вход в систему
</b> </p>
    Введите логин
<br>
<input type = "text" name = "UserNickName" maxlength = "15" class =
"enter_input_text">
<br>
    Введите пароль
<br>
<input type = "password" name = "UserPassword" maxlength = "15" class =
"enter_input_text">
<br><br>
<input type = "submit" name = "Auto" class = "button" value = "    Войти    ">
<br><br>
<input type = "button" class = "button" value = "Зарегистрироваться" onclick =
"location.href = 'registration.php'" >

```

```

</div>
</td>
</tr>
</table>
</form>
</td>
<!--<p style="cursor: pointer;"
onclick="document.getElementById('parent_popup').style.display='none';">Закреть
</p> -->
</div>
</div>
</body>
<table class="form" width="600" bgcolor="#EEE5DE"
align="center">
<tr>
<td>
<br><br><p align=center><font size=5><b>ДИФФЕРЕНЦИАЛЬНЫЕ
УРАВНЕНИЯ
</b></font><br><br>
<br>
<DD>Система дистанционного обучения <b><i>"Дифференциальные
уравнения"</b></i>, - это возможность изучить данную математическую
дисциплину в удобное для вас время и в заданном вами темпе. Все что вам
необходимо - это компьютер с возможностью подключения к сети Интернет,
немного свободного времени и безусловно ваше желание получить новые
знания.
<br><br><br>
<b><i><u>Целевая аудитория:</b></i></u><br>
<i>студенты высших технических учебных заведений;<br>

```


студенты средних специальных технических учебных заведений.

 <i><u>Цели и задачи системы ДО:</i></u>

 обучить основам дисциплины "Дифференциальные уравнения";

 объяснить с помощью примеров главные аспекты дисциплины;

 выработать у обучающегося навыки самостоятельной работы и самоконтроля.

Конечный результат, безусловно, во многом зависит от вас: от вашего усердия, организованности, осознания необходимости учиться, стремления преуспеть в учебной и профессиональной деятельности. А данная система дистанционного обучения создаст для вас максимально комфортные условия с учетом Ваших требований и возможностей, поможет Вам достичь поставленных целей.

<P align=center><i><u>Содержание курса:</i></u><ol start=1>
</P>

<DT>1 Дифференциальные уравнения первого порядка .

<DD><DD> 1.1 Понятия дифференциальных уравнений 1-го порядка и примеры практических задач.

<DD><DD> 1.2 Задача Коши для дифференциального уравнения 1-го порядка.

<DD><DD> 1.3 Понятие линейного дифференциальные уравнения первого порядка и методы решения.

<DD><DD> 1.4 Задачи Коши для линейного дифференциального уравнения 1-го порядка и ее решение.

<DT>2 Дифференциальные уравнения высших порядков

<DD><DD> 2.1 Понятия линейного дифференциального уравнения 2-го порядка с постоянными коэффициентами и методы их решения.

<DD><DD> 2.2 Задачи Коши для линейного дифференциального уравнения 2-го порядка с постоянными коэффициентами и методы решения задачи.

<DD><DD> 2.3 Понятие линейного дифференциального уравнения n-го порядка с постоянными коэффициентами.

<DD><DD> 2.4 Задача Коши для линейного дифференциального уравнения n -го порядка с постоянными коэффициентами.

<DT>3 Системы дифференциальных уравнений

<DD><DD> 3.1 Понятие системы дифференциальных уравнений.

<DD><DD> 3.2 Нормальные системы дифференциальных уравнений.

<DD><DD> 3.3 Задача Коши для нормальной системы дифференциальных уравнений.

<DD><DD> 3.4 Решение системы линейных однородных дифференциальных уравнений первого порядка с постоянными коэффициентами.

<DD><DD> 3.5 Решение задачи Коши для системы линейных неоднородных дифференциальных уравнений с постоянными коэффициентами методом вариации постоянных.

<DD><DD> 3.6 Определитель Вронского и его свойства.

<DT>4 Системы дифференциальных уравнений

<DD><DD> 4.1 Понятие оригинала и изображения.

<DD><DD> 4.2 Прямое и обратное преобразования Лапласа.

<DD><DD> 4.3 Основные свойства преобразования Лапласа.

<DD><DD> 4.4 Применения преобразования Лапласа для решения дифференциальных уравнений.

<DT>5 Устойчивость движения.

<DD><DD> 5.1 Определение устойчивости.

<DD><DD> 5.2 Дифференциальные уравнения возмущенного движения.

<DD><DD> 5.3 Устойчивость по Ляпунову и некоторые другие определения устойчивости.

<DD><DD> 5.4 О методах решения задачи устойчивости.

<DT>6 Колебания.

<DT>7 Параметрический резонанс

<DD><DD> 7.1 Параметрический резонанс. Общее дифференциальное уравнение.

<DD><DD> 7.2 Примеры механических систем с параметрическими колебаниями.

<DD><DD> 7.3 Параметрическое возбуждение по закону синуса.

<DD><DD> 7.4 Области неустойчивости.

<P align=center><i>НИУ БелГУ
Администратор ресурса

Grinev

</i></P>

</td>

</tr>

</table>

<?php

} else {

if(empty(\$_POST['UserNickName']) && empty(\$_POST['UserPassword'])) {

html_header("Ошибка авторизации", "BACKGROUND=img/background.bmp");

?>

<div align="center"><table class="freg" bgcolor="#EEE5DE"> <tr> <td valign = "middle" nowrap height = "10">

<P align="center">Для входа в систему введите, пожалуйста, свои логин и пароль.
Назад</P></tr> </td></table></div>

<? } else {

\$db->query("SELECT * FROM `users` where UserNickName =

"".\$_POST['UserNickName']."' and UserPassword = "".\$_POST['UserPassword']."'

LIMIT 1", __FILE__, __LINE__);

if(!\$db->result['is_error']) {

\$db->get_query_result();

if(\$db->result['data']) {

\$_SESSION['student'] = \$_POST['UserNickName'];

header("Location: index.php");

```

} else {
html_header("User not found", "BACKGROUND=img/background.bmp");?>
<div align="center"><table class="freg" bgcolor="#EEE5DE"> <tr> <td valign =
"middle" nowrap height = "10">
<P align="center">Пользователь не найден <BR><A
href="index.php">Назад</A></P></tr> </td></table></div>
<?
}
} else {
html_header("SQL Error", "BACKGROUND=img/background.bmp");
echo $db->result['error_message']."<BR>".$db->result['verbose_error_info'];
}
}
}
}
html_footer();
?>

```

Листинг программы admin.php

```

<?php
ob_start();
// Активизируем сессию
session_start();
error_reporting(E_ALL);
// Библиотеки
include("functions.php");
include("mysql_class.php");
$db = new mysql($config['db']);

```

```

mysql_query("SET NAMES 'cp1251'");
if($db->result['is_error']) {
    html_header("SQL Error", "BACKGROUND=img/background4.bmp");
    exit($db->result['error_message']."<BR>".$db->result['verbose_error_info']);
}
if(!empty($_SESSION['student'])) {
    $db->query("SELECT * FROM `users` where UserNickName =
    '".$_SESSION['student']."' LIMIT 1 ", __FILE__, __LINE__);
    if(!$db->result['is_error']) {
        $db->get_query_result();
        html_header("Страница администратора",
        "BACKGROUND=img/background4.bmp");
        if($db->result['data'][0]['UserType'] == 0) {
            echo "Вы не являетесь администратором";
            echo "<BR><A href='\"index.php\"'>Назад</A>";
        }
        else {
            echo "Что будем делать...";
            echo "<BR><A href='\"index.php\"'>Назад</A>";
            if(empty($_GET['act'])) $_GET['act'] = "default";
            // Отсюда ИДУТ ВСЕ ФОРМЫ И ВСЕ ОБРАБОТЧИКИ АДМИНСКИХ TOOLS
            if($_GET['act'] == "change") {
                if(!empty($_POST['ChangeLesID'])) {
                    $db->query("SELECT * FROM `lessons` where
                    `LesID`='".$_POST['ChangeLesID']."' ", __FILE__, __LINE__);
                    if(!$db->result['is_error']) {
                        f(!empty($_POST['ChangeLessonText2'])) { // Переменная отвечающая за
                        изменение и присутствие текста на экране

```

```

$db->query("UPDATE `lessons` SET `TextAr` = ".$_POST['ChLessonText']."'
WHERE `LesID`='".$_POST['ChangeLesID']."' LIMIT 1 ", __FILE__, __LINE__);
if(!$db->result['is_error']) echo "Урок обновлен";
else echo $db->result['verbose_error_info'];
} else {
$db->query("SELECT * FROM `lessons` where LesID =
".$_POST['ChangeLesID'].\"", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$text = ($db->result['data'][0]['TextAr']) ? $db->result['data'][0]['TextAr'] : "";
$ID = ($db->result['data'][0]['LesID']) ? $db->result['data'][0]['LesID'] : "";
}
?>
<form name = "change" action = "?act=change" method = "post" enctype =
"application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p> <b>
Изменить урок <img src = img/letter.gif>
</b> </p> </div>
</td>
<td valign = "center">
<div align = "center">
<p>
Текст урока
<br>

```

```

<textarea name = "ChLessonText" maxlength = "200" class =
"enter_input_textarea"><?=$text?></textarea>
<input name="ChangeLesID" type="hidden" value="<?=$ID?>">
<br> <br>
<input type = "submit" name = "ChangeLessonText2" class = "button" value =
"ИЗМЕНИТЬ">
</p></div></td></tr>
</table>
</form>
<?php
}
}
} else echo "<BR>Вы не выбрали номер урока для изменения";
}
elseif($_GET['act'] == "addtestform") { // При нажатии на кнопку добавления Теста
if(!empty($_POST['AddTestName'])) {
if(!empty($_POST['NewTestName']) && !empty($_POST['NewTestQCount'])) {
$temp = false;
$db->query("SELECT * FROM `Tests`", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
if($db->result['count'] == 0) $temp = 1;
if($db->result['count'] == 1) {
$db->query("SELECT * FROM `Tests` LIMIT 1", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
if(($db->result['data'][0]['TestId']) == 1) $temp = 2;
if(($db->result['data'][0]['TestId']) == 2) $temp = 1;
}
}
}
}
}
}
}

```



```

}
?>
<input type = "hidden" name = "TestID" class = "button" value = "<?=$temp?>">
<input type = "submit" name = "AddTest" class = "button" value = "Добавить">
</td></tr>
</table>
</form>
<?php
} else echo "Новый тест не создан";
}
} else {
echo "Введите название теста";
}
}
} elseif($_GET['act'] == "addtest") {
if(!empty($_POST['AddTest'])) {
for($i=0; $i<count($_POST['ques']); $i++) {
if($_POST['ques'][$i] != "") {
$db->query("INSERT into `questions` values ('NULL','" . $_POST['TestID'] . "',
'" . $_POST['ques'][$i] . "', 'NULL')", __FILE__, __LINE__);
$next_id = $db->get_next_id();
$last_id = false;
for($j=0; $j<count($_POST['answ'][$i]); $j++){
if($_POST['answ'][$i][$j] != ""){
$db->query("INSERT into `answers` values ('NULL','" . $next_id . "',
'" . $_POST['answ'][$i][$j] . "')", __FILE__, __LINE__);
if($_POST['ransw'][$i] == $j) $last_id = $db->get_next_id();
}
}
}
}
}
}
}

```

```

if($last_id) $db->query("UPDATE `questions` SET `question_answer` = ".$last_id."
WHERE `question_id` = ".$next_id." LIMIT 1", __FILE__, __LINE__);
}
}
}
}
else {
if(!empty($_POST['DelUser'])) { // Удаление пользователя
$db->query("DELETE FROM `users` where UserID = {$_POST['DelUserList']}",
__FILE__, __LINE__);
if(!$db->result['is_error']) echo "Пользователь удален";
else echo "Пользователь не удален";}
if(!empty($_POST['AddLessonText'])) { // Добавление урока
$db->query("INSERT into lessons values
(NULL, '{$_POST['LesNumber']}', '{$_POST['LessonText']}')", __FILE__,
__LINE__);
if(!$db->result['is_error']) echo "Урок добавлен";
else echo "Урок не добавлен";
}
if(!empty($_POST['DelLesson'])) { // Удаление урока
$db->query("DELETE FROM `lessons` where LesID = {$_POST['DelLesList']}",
__FILE__, __LINE__);
if(!$db->result['is_error']) echo "Урок удален";
else echo "Урок не удален";
}
if(!empty($_POST['AddTestName'])) { // Добавление названия теста
$db->query("INSERT into Tests values (NULL, '{$_POST['NewTestName']}')",
__FILE__, __LINE__);
if(!$db->result['is_error']) echo "Создан новый тест";
}
}
}
}

```

```

else echo "Новый тест не создан";}
if(!empty($_POST['DelOldTests'])) { // Удаление теста
$db->query("SELECT * FROM `questions` WHERE question_test_id =
".$_POST['DelTestList'].\"", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$search_i = $db->result['count'];
for($i = 0; $i < $search_i; $i++) {
$db->query("SELECT * FROM `questions` WHERE question_test_id =
".$_POST['DelTestList'].\"", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$search = $db->result['data'][$i]['question_id'];
$db->query("SELECT * FROM `answers` WHERE answer_question_id = '$search'",
__FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
$search_j = $db->result['count'];
for($j = 0; $j < $search_j; $j++)
$db->query("DELETE FROM `answers` WHERE answer_question_id = '$search'
LIMIT 1", __FILE__, __LINE__);
}
}
} $db->query("DELETE FROM `questions` WHERE question_test_id =
".$_POST['DelTestList'].\"", __FILE__, __LINE__);
} $db->query("DELETE FROM `Tests` WHERE testid =
".$_POST['DelTestList'].\"", __FILE__, __LINE__);
}
// НИЖЕ ИДУТ ФОРМЫ ДЛЯ ВЫБОРА ДЕЙСТВИЙ

```

```

?>
<form name = "delU" action = "admin.php" method = "post" enctype =
"application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p><b>
Удаление пользователя <img src = img/umnik2.gif>
</b></p></div></td>
<td valign = "center">
<div align = "center">
<p>
<select name = "DelUserList">
<option value = "0"></option>
<?php
$db->query("SELECT * FROM `users` where UserType = 0", __FILE__,
__LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($i = 0; $i < $db->result['count']; $i++)
echo "<option value = \"\".$db->result['data'][$i]['UserID'].\"\">".$db-
>result['data'][$i]['UserNickName'].\"</option>\";
}
?>
</select>
<br><br>
<input type = "submit" name = "DelUser" class = "button" value = "Удалить">

```

```

</p></div></td></tr>
</table>
</form>
<form name = "delL" action = "admin.php" method = "post" enctype =
"application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p><b>
Удаление урока <img src = img/umnik2.gif>
</b></p></div></td>
<td valign = "center">
<div align = "center">
<p>
<select name = "DelLesList">
<option value = "0"></option>
<?php
$db->query("SELECT * FROM `lessons`", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($i = 0; $i < $db->result['count']; $i++)
echo "<option value = \"\".$db->result['data'][$i]['LesID'].\"\"> Урок - ".$db-
>result['data'][$i]['LesN'].\"</option>";}
?>
</select>
<br><br>
<input type = "submit" name = "DelLesson" class = "button" value = "Удалить">

```

```

</p></div></td></tr>
</table>
</form>
<form name = "addlesson" action = "admin.php" method = "post" enctype =
"application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p><b>
Добавить урок <img src = img/letter.gif>
</b></p></div></td>
<td valign = "center">
<div align = "center">
<p>
Номер урока
<br>
<input type = "text" name = "LesNumber" maxlength = "20" class =
"enter_input_text">
<br><br>
Текст урока
<br>
<textarea name = "LessonText" class = "enter_input_textarea">
</textarea>
<br><br>
<input type = "submit" name = "AddLessonText" class = "button" value =
"Добавить">
</p></div></td></tr>

```

```

</table>
</form>
<form name = "changeles" action = "?act=change" method = "post" enctype =
"application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p><b>
Изменить урок <img src = img/letter.gif>
</b></p></div></td>
<td valign = "center">
<div align = "center">
<select name = "ChangeLesID">
<option value = "0"></option>
<?php
$db->query("SELECT * FROM `lessons`", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
for($i = 0; $i < $db->result['count']; $i++)
echo "<option value = \"\".$db->result['data'][$i]['LesID'].\"\"> Урок -\".$db-
>result['data'][$i]['LesN'].\"</option>";}
?>
</select>
<br><br>
<input type = "submit" name = "AddLessonText" class = "button" value =
"Изменить">
</p></div></td></tr>

```

```

</table>
</form>
<?php
    $temp = 0;
    $db->query("SELECT * FROM `Tests`", __FILE__, __LINE__);
    $db->get_query_result();
    if($db->result['count'] == 0) $temp = 1;
    if($db->result['count'] == 1) $temp = 2;
    if($temp != 0)
    {
?>
<form name = "Tests" action = "admin.php?act=addtestform" method = "post"
enctype = "application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p><b>
Добавить тест <img src = img/letter.gif>
</b></p></div></td>
<td valign = "center">
<div align = "center">
Название теста
<br><br>
<input type = "text" name = "NewTestName" maxlength = "40" class =
"enter_input_text">
<br><br>
Введите количество вопросов в тесте

```



```

<br><br>
<input type = "text" name = "NewTestQCount" maxlength = "2" size="3" class =
"enter_input_text">
<br><br>
<input type = "submit" name = "AddTestName" class = "button" value =
"Добавить">
</p></div></td></tr></table></form>
<?php
}
?>
<form name = "TestsDel" action = "admin.php" method = "post" enctype =
"application/x-www-form-urlencoded">
<table bordercolor = "blue" border = "0" width = "100%" height = "100" cellspacing
= "1" cellpadding = "5">
<tr>
<td valign = "center" width = "150">
<div align = "center">
<p><b>
Удалить тест <img src = img/umnik2.gif>
</b></p></div></td>
<td valign = "center">
<div align = "center">
<p>
<select name = "DelTestList">
<option value = "0"></option>
<?php
$db->query("SELECT * FROM `Tests`", __FILE__, __LINE__);
    if(!$db->result['is_error']) {
        $db->get_query_result();

```

```

        for($i = 0; $i < $db->result['count']; $i++)
echo "<option value = \"\".$db->result['data'][$i]['TestId'].\"\">".$db-
>result['data'][$i]['TestName'].\"</option>\";
}
?>
</select>
<br><br>
<input type = "submit" name = "DelOldTests" class = "button" value = "Удалить">
</p></div></td></tr>
</table>
</form>
<?php
}
}
}
}
html_footer();
?>

```

Листинг программы functions.php

```

<?php
// Функция формирует заголовок HTML документа
function html_header($title, $body_param = "") {
echo "<HTML>\r\n<HEAD>\r\n<TITLE>\r\n".$title."\r\n</TITLE>\r\n<LINK
href=\"style.css\" rel=\"stylesheet\" type=\"text/css\">\r\n<META http-
equiv=\"Content-Type\" content=\"text/html; charset=windows-
1251\">\r\n</HEAD>\r\n<BODY ".$body_param.">\r\n\";
}

```

```
// Функция формирует конец HTML документа
function html_footer() {
    echo "</BODY>\r\n</HTML>";
}
?>
```

Листинг программы `mysql_class.php`

```
<?php
// Настройки базы данных
$config['db']['login']          = "root";           // Логин пользователя для
подключения к серверу баз данных MySQL.
$config['db']['password']      = "";              // Пароль пользователя для
подключения к серверу баз данных MySQL.
$config['db']['host']          = "127.0.0.1";     // Имя или IP адрес сервера баз
данных MySQL.
$config['db']['name']          = "diplom";        // Название базы данных.
$config['db']['persistency']   = false;          // Постоянные соединения с
сервером баз данных MySQL.
$config['db']['transaction']   = false;          // Поддержка транзакций в базе
данных (для баз данных с таблицами типа InnoDB).
$config['db']['verbose_error_info'] = true;      // Вывод подробной информации
об ошибке (отладочная информация).
$config['db']['prefix']        = "";              // Префикс к названиям таблиц.
//=====
class mysql {
// Инициализация переменных для работы класса
// Параметры подключения к серверу баз данных MySQL.
```

```

var $db_login    = false;           // Логин пользователя для
подключения к серверу баз данных MySQL.
var $db_pass     = false;           // Пароль пользователя для
подключения к серверу баз данных MySQL.
var $db_host     = false;           // Имя или IP адресс сервера баз
данных MySQL.
var $db_name     = false;           // Название базы данных.
var $persistency = false;           // Постоянные соединения с сервером
баз данных MySQL.
var $transaction = false;           // Поддержка транзакций в базе
данных (для баз данных с таблицами типа InnoDB).
var $verbose     = false;           // Вывод подробной информации об
ошибке (отладочная информация).
var $result_type = MYSQL_ASSOC;     // Результат как ассоциативный
массив (MYSQL_ASSOC), неассоциативный (MYSQL_NUM), оба варианта
(MYSQL_BOTH).
// Ссылки на ресурсы.
var $connection_id = false;         // Ресурс на подключение к MySQL-серверу.
var $query_link    = false;         // Ресурс на SQL-запрос.
// Статусные индикаторы.
var $is_connected  = false;         // Индикатор статуса подключения к
базе данных.
var $in_transaction = false;        // Индикатор статуса транзакций в
запросах.
var $transaction_error = false;     // Индикатор статуса ошибок в SQL-
запросах, используемые транзакции.
// Структурированный результат SQL-запроса.
var $result = array(      'is_error' => false); // Индикатор статуса ошибок
(если ошибка то равно true).

```

```

//'data'          => false    // Результат SQL-запроса.
//'count'        => 0        // Количество выбранных записей в SQL-запросе.
//'next_id'      => 0        // При добавлении информации в базу данных
//                // содержит ID последней добавленной записи.
//'error_message' => false    // Пользовательский текст сообщения об ошибке.
//'verbose_error_info' => false // Отладочная информация об ошибке.
/* Конструктор класса mysql. Инициализирует переменные и создает
подключение к MySQL-серверу. */
function __construct($db_config) {
if(empty($db_config['login']) || empty($db_config['host']) ||
empty($db_config['name'])) {
$this->get_error_message("empty_db_config");
} else {
$this->db_host   = $db_config['host'];
$this->db_login  = $db_config['login'];
$this->db_pass   = @$db_config['password'];
$this->db_name   = $db_config['name'];
$this->persistency = @$db_config['persistency'];
$this->transaction = @$db_config['transaction'];
$this->verbose   = @$db_config['verbose_error_info'];
$this->connection_id = ($this->persistency) ? @mysql_pconnect($this->db_host,
$this->db_login, $this->db_pass) : @mysql_connect($this->db_host, $this-
>db_login, $this->db_pass);
if($this->connection_id) {
@mysql_select_db($this->db_name) ? $this->is_connected = true : $this-
>get_error_message("bad_db_name");
} else {
$this->get_error_message("bad_db_connection_config");
}
}
}

```

```

}
}
// Функция для инициализации конструктора.
function mysql($db_config) {
$this->__construct($db_config);
}
// Деструктор класса mysql. Функция для закрытия подключения к MySQL-
серверу.
function __destruct() {
$this->disconnect();
}
// Функция для закрытия подключения к MySQL-серверу.
function disconnect() {
if($this->connection_id) {
$this->query("END_TRANSACTION");
@mysql_close($this->connection_id);
$this->is_connected = false;
}
}
// Функция для выполнения SQL-запросов.
function query($query_string, $file = false, $line = false) {
if($this->is_connected) {
if(!empty($query_string)) {
$this->query_link = false;
unset($this->result);
$this->result['is_error'] = false;
switch($query_string) {
case "BEGIN_TRANSACTION" :   if($this->transaction && !$this-
>in_transaction) {

```

```

if(@mysql_query("BEGIN", $this->connection_id)) {

    $this->in_transaction = true;
}
}
break;
case "END_TRANSACTION" :    if($this->transaction && $this->in_transaction)
{
if($this->transaction_error) {
    $this->get_error_message("bad_transaction", $file, $line);
    @mysql_query("ROLLBACK", $this->connection_id);
} else {
if(!@mysql_query("COMMIT", $this->connection_id)) {
    $this->get_error_message("bad_transaction", $file, $line);
    @mysql_query("ROLLBACK", $this->connection_id);
}
}
    $this->transaction_error = false;
    $this->in_transaction = false;
}
break;
default :    if(!$this->query_link = @mysql_query($query_string, $this-
>connection_id)) {
if($this->in_transaction)
    $this->transaction_error = true;
    $this->get_error_message("bad_query", $file, $line, $query_string);
}
break;
}

```

```

}
}
}
// Функция для нахождения результата SQL-запроса.
function get_query_result() {
if($this->query_link) {
if($count = @mysql_num_rows($this->query_link)) { // Находит кол-во записей
после выполнения запроса
for($i = 0; $i < $count; $i++) {
$this->result['data'][] = @mysql_fetch_array($this->query_link, $this->result_type);
// Возвращает найденные записи
}
$this->result['count'] = $count;
} else {
$this->result['data'] = false;
$this->result['count'] = 0;
}
}
return $this->result;
}
// Функция для нахождения ID последней добавленной записи.
function get_next_id() {
$this->result['next_id'] = ($this->connection_id) ? @mysql_insert_id($this-
>connection_id) : 0;
return $this->result['next_id'];
}
// Функция которая формирует отчет об ошибке.
function get_error_message($error_code, $file = false, $line = false, $query = false) {

```



```

$error_messages['empty_db_config']      = "Вы не настроили подключение к
базе данных. Проверьте настройки подключения к базе данных.";
$error_messages['bad_db_connection_config'] = "Не возможно подключится к
базе данных. Проверьте настройки подключения к базе данных.";
$error_messages['bad_db_name']          = "Не возможно выбрать базу данных.
Проверьте имя базы данных в настройках.";
$error_messages['bad_query']            = "Не верный SQL запрос. Проверьте
правильность запроса.";
$error_messages['bad_transaction']      = "Транзакция завершилась неудачей";
$this->result['is_error']    = true;
$this->result['error_message'] = (@isset($error_messages[$error_code])) ?
$error_messages[$error_code] : false;
if($this->verbose) {
    $this->result['verbose_error_info'] = "";
    if($file)    $this->result['verbose_error_info'] .= "<B>Файл :</B> ".$file.". ";
    if($line)    $this->result['verbose_error_info'] .= "<B>Номер строки :</B>
".$line.". ";
    if($query)   $this->result['verbose_error_info'] .= "<B>SQL-запрос :</B>
".$query.". ";
    if($mysql_error = @mysql_error()) $this->result['verbose_error_info'] .=
"<B>mysql_error :</B> ".$mysql_error.". ";
    if($mysql_errno = @mysql_errno()) $this->result['verbose_error_info'] .=
"<B>mysql_errno :</B> ".$mysql_errno.". ";
}
else $this->result['verbose_error_info'] = false;
}
}
?>

```

Листинг программы registration.php

```

<?php
ob_start();
session_start();
error_reporting(E_ALL);
// Библиотеки
include("functions.php");
include("mysql_class.php");
$db = new mysql($config['db']);
if($db->result['is_error']) {
html_header("SQL Error", "BACKGROUND=img/background.bmp");
exit($db->result['error_message']."<BR>".$db->result['verbose_error_info']);
}
if(empty($_POST['Regist'])) {
html_header("Регистрация", "BACKGROUND=img/background.bmp");
?>
<table class="form" bgcolor="#EEE5DE" align=center border = "0" width = "40%"
height = "20%" cellspacing = "0" cellpadding = "0">
<tr>
<td valign="top">
<br><br>
<p align=center><FONT color=000000><B><FONT style="FONT-SIZE: 20pt"
face=TimesNewRoman>Регистрация нового пользователя</p></font>
<br><br><br><br>
<p align=center><FONT color=000000><B><FONT style="FONT-SIZE: 12pt"
face=TimesNewRoman>Заполните, пожалуйста регистрационную форму.<br>
Поля, отмеченные *, обязательны для заполнения.<br></p>

```

```

<form action = "registration.php" method = "post" enctype = "application/x-www-
form-urlencoded">
<div align = "center">
<table bordercolor = "blue" border = "1" width = "80%" cellspacing = "0"
cellpadding = "5">
  <tr>
  <br><br>
  </tr>
  <tr>
  <td valign = "middle" width = "20%" height = "60">
  Логин: *
  </td>
  <td valign = "middle">
  <div align = "center">
  <input type = "text" name = "UserNickName" maxlength = "20" class =
  "enter_input_text">
  </div></td></tr><tr>
  <td valign = "middle" width = "20%" height = "40">
  Адрес e-mail:
  </td>
  <td valign = "middle">
  <div align = "center">
  <input type = "text" name = "UserEmailBox" maxlength = "40" class =
  "enter_input_text">
  </div></td></tr><tr>
  <td valign = "middle" width = "20%" height = "40">
  Новый пароль: *
  </td>
  <td valign = "middle">

```

```

<div align = "center">
  <input type = "password" name = "UserPassword" maxlength = "20" class =
  "enter_input_text">
</div></td></tr><tr>
<td valign = "middle" width = "20%" height = "40">
Подтвердите пароль: *
</td>
<td valign = "middle">
<div align = "center">
<input type = "password" name = "UserPasswordAgain" maxlength = "20" class =
  "enter_input_text">
</div></td></tr><tr>
<td valign = "middle" width = "20%" height = "10"><div align = "center">
<br> <input type = "button" name = "Back" value = "Назад" onclick = "location.href
= 'index.php'" class = "button">
</div></td>
<td valign = "middle">
<div align = "center">
<br> <input type = "submit" name = "Regist" value = "Зарегистрироваться" class =
  "button">
</div></td></tr></table></td></tr>
</table>
</div>
</form>
<?php
}
// Если форма отправлена, то обработать ее и зарегистрировать нового
пользователя.
else {

```

```

html_header("Отчет регистрации", "BACKGROUND=img/background.bmp");
?>
<div align="center">
<table class="freg" bgcolor="#EEE5DE" border = "0" width = "50%" cellspacing =
"1" cellpadding = "5">
<tr>
<td valign = "middle" nowrap height = "10">
<div align = "center">
<br><br><FONT color=000000><B><FONT style="FONT-SIZE: 20pt"
face=TimesNewRoman> Регистрационная информация
</div></td></tr><tr>
<td valign = "middle">
<div align = "center">
<?
// Проверка ввода имени и пароля
if((!$_POST['UserNickName']) || (!$_POST['UserPassword'])) {
echo "Вы не заполнили обязательные поля логин и пароль.";
} else {
// Поиск существующих имен (Повторений)
$db->query("SELECT * FROM `users` where UserNickName =
"$_POST['UserNickName'].\"", __FILE__, __LINE__);
if(!$db->result['is_error']) {
// выбираем данные (находим дубликат логина) и ..
$db->get_query_result();
// если пользователь существует, то выводим ошибку.
if($db->result['data']) {
echo "<br>Пользователь с таким логином уже существует. Пожалуйста,
выберите другой";
}
}

```

```

// иначе регистрируем нового пользователя.
else {
$db->query("SELECT * FROM `users`", __FILE__, __LINE__);
if(!$db->result['is_error']) {
$db->get_query_result();
if($db->result['count'] == 0) $UserType = 1;
else $UserType = 0;
$Test1 = 0;
$Test2 = 0;
$Exam = 0;
}
$db->query("INSERT into users values
(NULL,'" . $UserType . "','" . $_POST['UserNickName'] . "','" . $_POST['UserEmailBox'] .
,'" . $_POST['UserPassword'] . "','" . $Test1 . "','" . $Test2 . "','" . $Exam . "')", __FILE__,
__LINE__);
if(!$db->result['is_error']) {
echo "<br>Вы зарегистрированы. Нажмите кнопку, чтобы вернуться на
главную страницу и войти в систему.";
} else {
echo "<br> Произошла ошибка при регистрации<BR>";
echo $db->result['error_message'] . "<BR>" . $db->result['verbose_error_info'];
}
}
}
// Выводим сообщение об ошибке SQL запроса.
else {
html_header("SQL Error", "bgcolor=#80AFED");
echo $db->result['error_message'] . "<BR>" . $db->result['verbose_error_info'];
}
}

```

```

}
?>
</div></td>
<tr>
<td valign = "top" nowrap>
<div align = "center">
<input type = "button" name = "Back" value = "Вернутся на главную страницу"
onclick = "location.href = 'index.php'" class = "button">
</div></td></tr></table></div>
<?php
}
html_footer();
?>

```

Листинг файла style.css

```

/* СТИЛЬ КНОПОК */
.button
{
FONT-SIZE: 12px; FONT-WEIGHT : bold;
BORDER-RIGHT: #222222 1px solid;
BORDER-TOP: #222222 1px solid;
BORDER-LEFT: #222222 1px solid;
COLOR: #ffffff;
BORDER-BOTTOM: #222222 1px solid;
BACKGROUND-COLOR: #455368;
}
/* СТИЛЬ ТЕКСТОВЫХ ПОЛЕЙ РЕГИСТРАЦИОННОЙ ФОРМЫ */
.register_input_text

```

```
{
BORDER-RIGHT: blue 2px solid;
BORDER-TOP: blue 2px solid;
FONT-SIZE: 14px;
FONT-WEIGHT : bold;
BORDER-LEFT: blue 2px solid;
WIDTH: 200px;
COLOR: #FFFFFF;
BORDER-BOTTOM: blue 2px solid;
HEIGHT: 18px;
BACKGROUND-COLOR: blue;
}
/* Стиль текстовых полей формы для авторизации */
.enter_input_text
{
BORDER-RIGHT: white 2px solid;
BORDER-TOP: white 2px solid;
FONT-SIZE: 14px;
FONT-WEIGHT : bold;
BORDER-LEFT: white 2px solid;
WIDTH: 200px;
COLOR: #000000;
BORDER-BOTTOM: white 2px solid;
HEIGHT: 18px;
BACKGROUND-COLOR: white;
}
/* Стиль текстового поля для добавления урока*/
.enter_input_textarea
{
```



```
BORDER-RIGHT: white 2px solid;
BORDER-TOP: white 2px solid;
FONT-SIZE: 11px;
FONT-WEIGHT : bold;
BORDER-LEFT: white 2px solid;
WIDTH: 700px;
COLOR: #000000;
BORDER-BOTTOM: white 2px solid;
HEIGHT: 400px;
BACKGROUND-COLOR: white;
}
```

```
#parent_popup {
    background: url(img/vsokno.png) ;
    height: 100%;
    position: fixed;
    display: none;
    width: 100%;
    z-index: 100;
    top: 0;
    left: 0;
}
```

```
#popup {
    background: url(img/vxod2.png);
    height: 310px;
    position: fixed;
    top: 100px;
    left: 35%;
    color: #F0F0F0;
    width: 318px;
```

```
}  
.closen {  
    width: 31px;  
    height: 31px;  
    left: 20%;  
    display: block;  
    cursor: pointer;  
    background: url(img/close.png);  
    position: absolute;  
    left: 90%;  
    text-indent: -4999px;  
}  
.form {  
    width: 800px; height: 800px;  
    border: 7px;  
    border-radius: 100%/10%;  
}  
.freg{  
    width: 150px; height: 150px;  
    border: 7px;  
    border-radius: 100%/40%;  
}  
.formpol {  
border-radius: 20px;  
}  
.formpoll {  
border-radius: 35px;  
}  
a {text-decoration:none !important;}
```

```
a {color: black !important}
a:hover{color:purple !important}
a1 {text-decoration:none !important;}
a1 {color: White !important}
a1:hover{color:White !important}
```