

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
Кафедра «Прикладная математика и информатика»

09.04.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

ПРИКЛАДНАЯ ИНФОРМАТИКА В ОБРАЗОВАНИИ И
ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЯХ

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему: Технологии уровневой дифференциации обучения в построении
информационно-образовательной среды вуза

Студентка _____ А.А. Селезнева _____

Научный
руководитель _____ Е.В. Панюкова _____

Руководитель программы _____ д.пед.н., профессор, А.Н. Ярыгин _____

« _____ » _____ 20 _____ г.

Допустить к защите

Заведующий кафедрой _____ к.тех.н., доцент, А.В. Очеповский _____

« _____ » _____ 20 _____ г.

Тольятти 2016

Содержание

Введение.....	4
Глава 1 Анализ проблемных областей в обеспечении дистанционного обучения в ВУЗе.....	12
1.1 Выявление особенностей дистанционного обучения.....	12
1.2 Анализ систем дистанционного образования	18
1.3 Анализ внедрений систем дистанционного обучения	21
1.4 Нормативное правовое обеспечение сферы дистанционного обучения	25
Глава 2 Проектирование технологии уровневой дифференциации обучения для электронно-образовательной среды вуза.....	29
2.1 Технологии уровневой дифференциации	29
2.2 Технология уровневой дифференциации для системы дистанционного обучения	42
Глава 3 Проектирование структуры системы дистанционного обучения	45
3.1 Основные подходы к разработке электронно-образовательной среды с функцией выбора уровня сложности обучения	45
3.2 Обоснование выбора технологии проектирования	48
3.3 Структурная и функциональная модели информационного обеспечения системы организации дополнительного образования	53
3.4 Проектирование информационного обеспечения системы дистанционного образования.....	56
3.5 Функциональное моделирование организации процесса обучения в ВУЗе .	60
3.6 Инфологическая модель данных электронно-образовательной среды обучения уровневой дифференциации	65
3.7 Даталогическая модель данных электронно-образовательной среды обучения уровневой дифференциации	66
3.8 Разработка физической модели базы данных электронно-образовательной среды обучения уровневой дифференциации	69
3.9 Выбор архитектуры электронно-образовательной среды обучения уровневой дифференциации	70

3.9 Описание технической и программной структуры	71
Глава 4 Экспериментальная оценка качества обучения с использованием информационно-образовательной среды с технологией уровневой дифференциации обучения	74
Заключение	82
Список литературы	85
Приложение А	91
Приложение Б	108

Введение

Основным направлением стратегии развития информационного общества в области повышения качества образования является оптимизация информационных и телекоммуникационных технологий под новые формы и методы обучения.

Перед современным образованием поставлены задачи формирования конкурентоспособной и успешной личности в информационно-электронной среде.

В условиях информационного общества от преподавателей требуется не только совершенное владение информационно-коммуникационными технологиями, но и умение организовать учебный процесс на инновационном уровне. Одна из основных тенденций развития современного образования состоит в пересмотре концепций организации учебной деятельности.

Актуальность исследования определяется отсутствием технологий дистанционного обучения, которые позволяли бы формировать обучение в зависимости от исходного уровня знаний обучающихся.

Постоянное развитие в сфере информационных технологий в качестве действенного средства обучения дает возможность использовать персональные компьютеры. Автоматизация процесса обучения осуществляется с внедрением компьютерных обучающих программ и электронных учебников, использующие не только лазерные диски, но и локальные и глобальные компьютерные сети. В последнем случае происходит формирование специализированной информационно-образовательной среды, позволяющей реализовать современные технологии обучения. Для наполнения информационно-образовательной среды, также для действенного использования локальных и глобальных компьютерных сетей нужна оперативная разработка систем дистанционного обучения достойного уровня, соответствующих прогрессивному состоянию науки в этой предметной области.

Сначала 80-х годов прошедшего века количество различных автоматизированных обучающих систем измерялось тысячами [8]. Но

стремительная смена поколений вычислительной техники, переход аппаратного и программного обеспечения на принципиально новейший уровень привели к тому, что в том числе более успешные варианты автоматизированных обучающих систем оказались «не у дел».

Все же, автоматизированные обучающие системы первого поколения сыграли довольно существенную роль. Во-первых, они проявили, что компьютерное обучение — дело вполне реальное. Во-вторых, накопленный опыт разрешил обнаружить трудности в разработке обучающих программ, которые приведены ниже:

- до начала внедрения обучающей программы в предметной области маловероятно прогнозировать её эффективность;
- учебный материал по некоторым дисциплинам трудно представить в достаточно формализованном виде;
- при исследовании неких дисциплин необходим личный контакт («глаза в глаза») между преподавателем и студентом;
- далеко не всегда возможно выбрать адекватные критерии оценивания знаний и навыков студентов.

К проблемам традиционного обучения следует отнести:

1. Не берется за основу начальный уровень знаний, умений обучаемых.
2. В случае если дистанционное обучение уже началось, существует сложность внесения оперативных изменений в обучающий курс.
3. По сравнению с иными формами обучения, отсутствует возможность формирования дополнительной мотивации у слушателей дистанционного обучения.
4. Необходимость больших вложений на предварительном этапе проведения дистанционного обучения (вложения в разработку дистанционных курсов).
5. Высокая зависимость от технической инфраструктуры. Сбой в инфраструктуре сможет привести к снижению эффективности или к срыву дистанционного обучения.

6. Отсутствие необходимого количества квалифицированных специалистов в области технологий дистанционного обучения.
7. Высокие вложения на внесение изменений в дистанционное обучение.

Второе рождение компьютерных обучающих систем связано с распространением персональных компьютеров, оснащенных графическим пользовательским интерфейсом и способных воспринимать и воспроизводить мультимедийные данные в разных форматах [10].

Возможность работы с мультимедийными данными (графикой, видео и звуком) обеспечила создание обучающих программ и тренажеров нового поколения, полностью правильно воспроизводящих реальные устройства и объекты. Известно, к примеру, что сейчас подготовка летчиков не обходится без применения соответствующих имитаторов, воспроизводящих как органы управления самолетом, так и воздушную обстановку.

Именно благодаря «выходу» обучающих программ в Интернет возник термин eLearning – «электронное обучение» [59].

Другими словами, в последнее время под электронным обучением понимается такая его форма, при которой обучаемые и преподаватель отдалены друг от друга в пространстве и при работе над учебным материалом используют сервисы Интернета. Другое, наиболее традиционное название такой схемы образования — дистанционное (либо удаленное) обучение, DistanceLearning.

Под дистанционным обучением надлежит понимать такую организацию обучения, при которой обучаемый получает доступ к учебно-методическим материалам и консультациям преподавателя в любое время суток, семь дней в неделю и в том месте, где он присутствует. Актуальность применения дистанционного обучения в настоящее время уже не вызывает сомнений.

Технологии уровневой дифференциации дал определение Гузик Н. П. [23]. По мнению Г.К. Селевко, «дифференцированное обучение – это форма организации учебного процесса, при котором учитель работает с группой учащихся, составленной с учетом наличия у них каких-либо значимых для учебного процесса общих качеств; часть общей дидактической системы, которая

обеспечивает специализацию учебного процесса для различных групп обучающихся» [54].

Сложность и важность отмеченных проблем для теории и практики применения технологии уровневой дифференциации в дистанционном обучении требуют проведения серьезных методологических и методических исследований в области разработки модели применения технологии уровневой дифференциации в дистанционном обучении.

В связи с этим целью диссертационной работы является теоретическое обоснование и практическая реализация информационно-образовательной среды вуза с технологией уровневой дифференциации обучения.

Объект исследования: процесс дистанционного обучения.

Предмет исследования: система дистанционного обучения.

Гипотеза диссертации:

Является предположение о зависимости эффективности дистанционного обучения от применения технологии уровневой дифференциации.

Для достижения поставленной цели исследования и проверки сформулированной гипотезы необходимо решить следующие **задачи**:

1. На основе анализа научной литературы определить особенности построения информационно-образовательной среды системы дистанционного обучения (СДО).
2. Провести анализ существующих электронных обучающих сред.
3. Спроектировать технологию уровневой дифференциации для СДО.
4. Построить модель СДО в учебном заведении.
5. Спроектировать и реализовать СДО в учебном заведении.
6. Экспериментально подтвердить эффективность использования технологии уровневой дифференциации.

Теоретической и методологической основой исследований послужили педагогические, технические и психологические аспекты проблем инновационных изменений в системе образования А.В. Аракелова, В.П. Беспалько, И.И. Валеева, Н.Ф. Ильиной, Л.В. Криволаповой, Ю.А. Лобейко, А.И.

Иванова, В.А. Сластенина, Н.В. Чекалевой, А.Д. Цедринского; направления развития виртуального пространства и компонентов виртуальной образовательной среды А.А. Андреева, В.П. Тихомирова, Д. Тиффин, Л. Раджасингама, А.В. Хуторского, И.В. Холодковой; особенности педагогической деятельности и технологий обучения в системе дистанционного образования В.П. Демкина, С.Д. Каракозова, М.П. Лавровой, И.В. Масловой, И.Н. Розиной, Г.Д. Хорошавиной; методические аспекты проектирования и использования информационных технологий в образовании А.А. Андреева, А.В. Соловова, В.И. Солдаткина; инновационный потенциал сетевого обучения, проектного метода обучения Е.С. Полат, О.Н. Ионовой, А.А. Андреева, А.В. Дьяченко, В.К. Мязотс, В.П. Сергеевой, В.В. Гузеева, Г.Л. Ильина, Э.М. Никитина, Т.Г. Новиковой, Е. Паркхерст; тьюторское сопровождение и открытые образовательные технологии в сетевом обучении Э. Гордон, Л.В. Бендовой; использование сетевых учебных сообществ в рамках сетевой педагогики В. Буш, Д. Энгельберт, Д. Уоттс, П. Сейж, С. Пейперт, Ч. Ледбетт, Е.Д. Патаракина; оценивание качества сетевых учебных ресурсов В.С. Тоискина, В.В. Красильникова, А.А. Андреева, К.Ю. Лупанова, В.И. Солдаткина; проблемы самообразования в контексте сетевого обучения П.И. Пидкасистого, И.Н. Розиной, А.В. Хуторского).

В качестве **методов** исследования использовались: анализ учебно-методической, научно-технической литературы, нормативных документов по проблеме исследования; педагогические наблюдения за самостоятельной работой студентов с СДО; интервьюирование и анкетный опрос студентов, преподавателей и экспертов; анализ результатов тестирования студентов; анализ экспертных оценок методических материалов и разработанных при подготовке эксперимента; наблюдение.

Основные этапы исследования: исследование велось с 2014 по 2016 гг. в три этапа.

На **первом** – *констатирующем этапе* исследования (2014 г.) – определялась актуальность темы исследования, прорабатывалась литература по избранной теме, уточнялся аппарат и программа исследования, формулировалась гипотеза,

определялись цели, задачи, предмет, объект и методы исследования, изучалось состояние проблемы дистанционного обучения, анализировались существующие системы дистанционного образования.

Второй этап (2014 - 2015 гг.) – *поисковый*. Здесь формировались концептуальные основы дистанционного обучения, определялась технология уровневой дифференциации, уточнялись цели, задачи и гипотеза исследования, осуществлялась апробация теоретических подходов в выступлениях и публикациях.

На **третьем** этапе (2015 - 2016 гг.) – осуществлялся *эксперимент*. Проводилась апробация разработанной обучающей образовательной среды, проверялась обоснованность и достоверность сформулированной гипотезы, обрабатывались и анализировались результаты эксперимента, обобщались итоги экспериментальной работы.

Апробация результатов исследования осуществлялась в «Поволжском православном институте имени святителя Алексия» г. Тольятти. Результаты исследования были опубликованы на международных конференциях Международного центра инновационных исследований «Инновации, технологии, наука», так же в научно-издательском центре Аэретна «В мире науки и инноваций».

Научная новизна. В результате проведенных исследований и анализа была разработана технология уровневой дифференциации для ДО, информационное обеспечение системы дистанционного обучения с технологией уровневой дифференциации, которая учитывает начальный уровень знаний студента, дает возможность выбора самим студентом уровня сложности обучения, таким образом, ориентирует студента на конкретные объемы работы, мотивирует его к самостоятельной работе.

Теоретическая значимость исследования заключается в определении проблем дистанционного обучения, применяемых в вузах; анализе программно-методических средств при дистанционном обучении.

Практическая значимость диссертационного исследования заключается в разработке информационного обеспечения системы дистанционного обучения с технологией уровневой дифференциации.

На защиту выносятся:

1. Технология уровневой дифференциации для СДО.
2. Реализованное информационное обеспечение СДО, включая анализ аналогов, основные алгоритмы.
3. Модель информационного обеспечения СДО.
4. Показатели уровня успеваемости студентов, полученные с использованием разработанного информационного обеспечения СДО.

Работа представляет собой результат теоретической и практической значимости в области дистанционного образования.

Объем выпускной квалификационной работы – 108 страниц, которые включают текст, модели, схемы, таблицы, изображения и программный код. В структуру работы входят: введение, 4 главы, выводы по главам, заключение, список литературы и приложение.

В первой главе рассматриваются научно-методологические основы создания и использования информационного обеспечения в учебном процессе дистанционного образования.

Во второй главе описана и разработана технология уровневой дифференциации обучения.

В третьей главе проводится процесс проектирования основных процессов информационного обеспечения, определяются основные типы пользователей системы, их права и функции, проводится анализ и выбор методики определения профессиональных способностей, проектируется даталогическая и физическая модели данных, выбираются средства реализации информационного обеспечения и непосредственно осуществляется реализация информационного обеспечения системы дистанционного обучения, спроектирована диаграмма потоков данных DFD, которая показывает все сущности и потоки данных спроектированной системы. А также спроектированы диаграмма деятельности и

диаграмма вариантов использования в нотации UML, которые отражаются всю сущность системы организации дистанционного обучения. Для отражений функции и структур системы построены функциональная и структурная схемы проектируемой системы.

В четвертой главе проводится апробация разработанной информационной системы дистанционного образования с целью выявления преимуществ и недостатков системы.

В заключении подводятся итоги выполненной работы.

Глава 1 Анализ проблемных областей в обеспечении дистанционного обучения в ВУЗе

1.1 Выявление особенностей дистанционного обучения

Использование педагогами дистанционных образовательных технологий - ответ на требования времени. Эти требования носят как формальный характер нормативных документов, так и неформальный – потребность самого педагога сделать процесс обучения более эффективным.

Во-первых, самообучение и помощь дистанционного образования дает возможность приобрести ценный опыт. Недостаточно просто познакомиться с курсом, просмотреть лекции и попытаться выполнить некоторые задания. Для того чтобы оценить содержание курса и то, в какой методической форме это содержание представлено, необходимо пройти курс полностью. Прослушать все (почти все) лекции, выполнить задания, ответить на вопросы и получить сертификат. Отличия между курсами существенны. Методические и технологические приемы разнообразны. Но независимо от тематики курса, опыт прохождения будет очень полезным.

Все больше курсов предлагает работы, связанные с взаимной оценкой слушателей. И здесь на первый план выдвигаются критерии, по которым слушатели будут оценивать работы коллег. Неоднозначные формулировки могут погубить идею и вызвать недовольство слушателей, которое они выплеснут в форумах.

В последнее время резко возрос интерес к дистанционному обучению. Образовательные учреждения создают собственные системы, функциональные возможности которых определяются платформой, на которой они разворачиваются, и, вообще говоря, различаются незначительно.

При этом на первый план выходит проблема разработки курсов, изначально ориентированных на дистанционное обучение, т.к. попытка использовать стандартные курсы может рассматриваться, в лучшем случае, как паллиатив.

Учащийся при дистанционном обучении в значительной степени лишен постоянного контакта с преподавателем, работает только с материалами курса. Именно поэтому проблема восприятия и понимания научного (учебного) текста становится решающим фактором, который определяет эффективность и результаты обучения в целом.

Т. Ван Дейк дает следующее определение дискурса в широком смысле (как комплексного коммуникативного события) - дискурс есть коммуникативное событие, происходящее между говорящим и слушающим (наблюдателем и др.) в процессе коммуникативного действия в определенном временном, пространственном и прочих контекстах [47].

Это коммуникативное действие может быть речевым, письменным, иметь вербальные и невербальные составляющие. Явно, что мультимедийный курс полностью удовлетворяет данному определению. При всем этом научный текст представляет собой законченный продукт коммуникативного действия, его письменный результат, который интерпретируется студентом.

Таким образом, предполагается подходить к решению проблем порождения и понимания письменного научного текста и мультимедийной составляющей курса системы дистанционного обучения в рамках когнитивно-дискурсивной парадигмы, разрабатываемой в настоящее время школой Е.С. Кубряковой.

К задачам наиболее общего характера причисляют следующие:

- определение специфичных свойств учебного курса системы дистанционного обучения как совокупности текста и мультимедийного контента с особой когнитивно-дискурсивной функцией;
- анализ типов знаний, передаваемых в курсе дистанционного обучения, а также средств их вербализации;
- анализ информативности единиц номинации в курсе дистанционного обучения;
- уточнение роли предзнания для адекватного понимания текста.

Как любой научный текст, курс для системы дистанционного обучения должен обладать такими чертами, как информативность, связность, целостность, членимость.

Однако дистанционный характер обучения предполагает, прежде всего, что вводимые в нем знания должны быть особенно четко структурированы и специально предназначены для более простого овладения ими.

В этом смысле особо значимыми специфическими чертами являются наличие в курсе продуманного набора ключевых понятий и способы упорядоченной подачи понятийного аппарата.

Понятие членимости является одной из основных категорий, проявление которой в мультимедийном контенте отличает его от других типов текста. Членимость в дистанционном курсе реализуется через особую систему построения курса.

В последнее время популярным становится подход, при котором весь материал курса разбивается на модули небольшого размера (20–30 минут видеокурса или озвученной презентации, текст, задание для самостоятельной работы, тест). Общее время для изучения модуля – не более 6 часов. При этом модули и система в целом должны быть построены таким образом, чтобы максимально автоматизировать процесс общения с обучаемым. Однако необходимо учитывать, что на уровне макроструктуры курса соотношение дистанционного курса в целом с единицами членения (модулями) строится на основе реализации принципа аддитивности, когда между такими единицами устанавливаются отношения дополнения, развития и взаимодействия. Преподаватель, разрабатывающий модули, должен, естественно, строить схему курса с указанием зависимости модулей друг от друга. Помимо обучающих модулей, необходимо создание разделов FAQ, что сейчас крайне редко можно увидеть в дистанционных курсах, но является общепринятым в компьютерных системах.

Понимание научного текста и его последующая интерпретация связаны также с восстановлением скрытой информации, которая раскрывается в ходе

последовательности умозаключений. Очевидно, что умозаключение возможно только при наличии у адресата определенной базы предзнания, причем предзнания экспертного (специального).

Информативность представляет собой способность текста (мультимедийного контента) отображать авторское мировосприятие, выраженное в конкретной форме, которая важна для декодирования вербализованной информации.

Каждому контенту присуща своя информативность, которая меняется от нулевой, когда содержание текста не дает ничего нового, а лишь повторяет уже известное, до максимальной, когда для ее выявления необходимо подвергнуть текст анализу, используя предзнание, и опять падает до нуля в случае, когда предзнания недостаточно для понимания научного текста (контента). С точки зрения И.Р. Гальперина, информация приобретает концептуальность, если несет результат исследований, наблюдений, опыта, размышлений и т.п.

Предзнания, полученные человеком в результате познания мира через свой опыт либо в процессе абстрактного мышления, могут быть извлечены в результате восприятия и понимания в дискурсе.

Модульный принцип построения системы дистанционного обучения, наличие гиперссылок, позволяет отойти от линейности в процессе изучения материала, обеспечивает возможность использования системы дистанционного обучения не только для получения систематического образования, но и как информационно-справочной системы.

С другой стороны, это накладывает дополнительные обязательства на разработчика, т.к. необходимо учитывать различный уровень предзнания при обращении учащегося к тому или иному элементу курса. Именно здесь схема курса с указанием взаимозависимости модулей играет решающую роль.

Рассматривая соотношение текстовой и мультимедийной составляющей в курсе, вернемся к понятию дискурса. Некоторые исследователи указывают, что дискурс и текст присутствуют в оппозиции «устный/письменный», когда

дискурс ассоциируется исключительно со звучащей устной речью, а текст соотносится с письменной формой.

Сопоставляя текст и дискурс, обычно указывают на то, что последний анализируется по мере своего поступления к адресату, то есть онлайн, а для его понимания употребляется не вся информация в голове человека, а специфическим образом уже «расклассифицированная», конкретно относящаяся к определенной области знания.

Восприятие дискурса в режиме онлайн требует от студента умения мгновенно ориентироваться в ситуации, адекватно обращать внимание на поступающую информацию, восстанавливать на основе полученной языковой формы, стоящий за ней ментальный концепт, либо ментальную структуру. Часто студент должен «домысливать» то, что имел в виду говорящий, используя догадку, интуицию, делая определенные выводы. В случае же восприятия письменного текста у студента всегда есть возможность неоднократного возвращения к конкретному фрагменту, обращения к справочной литературе, обдумывания собственной реакции на полученную информацию и так далее [23].

И здесь встает вопрос индивидуальных предпочтений студентов относительно стиля обучения. Обычно выделяют визуальный (V: графики, таблицы, блок-схемы), слуховой (A: обучение через речь), чтение-запись (R: обучение через чтение материалов и запись), кинестетический (K: обучение через обмен реальным опытом) стили обучения.

Студенты с V-предпочтением учатся лучше, если видят или рассматривают (рисунки, диаграммы, презентации и т.п.). Студенты, предпочитающие A, лучше учатся, если слушают записанные лекции, обсуждения и разговаривая о предмете сами с собой или с другими. R-тип студентов обучается через взаимодействие с текстовым материалом. K-стиль предполагает использование физического опыта: трогать, действовать, двигаться, лекции с акцентом на действие и манипуляцию с объектами. Студенты способны использовать все эти сенсорные режимы при обучении, но в

то же время каждый индивидуум обладает уникальными предпочтениями, или набором предпочтений, в котором один из них является доминирующим [29].

В процессе разработки курса дистанционного обучения необходимо учитывать гендерные особенности восприятия и возможные особенности, зависящие от социального уровня. Одним из путей увеличения мотивированности студентов является адаптация учебных материалов дистанционных курсов для обеспечения соответствия выбираемых методов предпочтениям учащихся. Единственный известный на сегодня способ отнесения к той или иной категории – тестирование.

Таким образом, разработчик курса должен предусмотреть возможность изучения курса учащимися с различными предпочтениями и, по возможности, по результатам тестирования предлагать тот или иной вариант обучения.

Разработчик курса производит выбор лексико-грамматических единиц и языковых структур, применяемых для вербализации научного сообщения, определяет последовательность представления информации и формирует принципы построения и подачи текста и мультимедийного контента зависимо от того, какие характеристики он приписывает предполагаемому студенту.

При всем этом автору необходимо учитывать не только лишь предполагаемую базу знаний гипотетического студента, но и (в отдельных случаях) возраст, социальную принадлежность, прагматические установки, мотивацию и прочие экстралингвистические характеристики возможного адресата. От данных параметров также зависит и выбор конкретных средств авторского воздействия на адресата, в том числе аргументация, иллюстративные примеры (графические и/или вербальные), ссылки на другие тексты, либо авторов (интертекстуальность) [53].

В этой ситуации представляется разумным использование принципов поисковой оптимизации (в частности – анализ частоты использования слов) для создания мультимедийного контента, в максимальной степени понятного учащемуся.

Очевидно, что для понимания научного текста особое значение имеет знание терминологии как общенаучных, так и специальных терминов, причем главным образом – терминов базового уровня. Поэтому так важно наличие глоссария и, главное, возможности обращения к нему из любого модуля системы дистанционного обучения.

Из вышеизложенного можно сделать вывод, что особенности мультимедийного контента для системы дистанционного обучения подтверждают «взаимопроникновение» когнитивного и дискурсивного аспектов.

Наиболее важным результатом понимания изложенного в курсе материала можно считать возможность применить приобретенные знания на практике, либо применять их в ситуациях, подобных тем, которые представлены в тексте. Таким образом, реализуется переход от знания «что» к знанию «как», что важно при прогнозировании определенных действий, принятии решений, выработке способов разрешения проблемы и так далее.

Связь полученных знаний с успешной практической деятельностью специалиста подтверждает адекватность понимания научного текста и также может рассматриваться как косвенный результат такого понимания.

Так же работа с языковым произведением требует от разработчика знания самого языка, т.е. обладания языковой компетенцией. Эти знания включают знание лексики, знание идиоматических выражений, правил грамматики (морфологии и синтаксиса), правил орфографии, знание лексических и синтаксических средств связности текста и т.д. Помимо этого, разработчику необходимо иметь общее представление о том, как употребляются средства языка, чтобы сделать вывод о корректности и правильности вербализующих научное знание языковых структур и об их уместности в научном тексте.

1.2 Анализ систем дистанционного образования

Moodle – система управления курсами (электронное обучение), помимо прочего известная как система управления обучением, либо виртуальная обучающая среда (англ.) [5]. Представляет из себя свободное

(распространяющееся по лицензии GNU GPL) веб-приложение, предоставляющее возможность создавать веб-сайты для онлайн-обучения.

Система реализует философию «педагогики социального конструкционизма» и нацелена сначала на организацию взаимодействия между преподавателем и студентом, хотя подходит и для организации обычных дистанционных курсов, также поддержки очного обучения [55]. Moodle написана на PHP с внедрением SQL-базы данных (MySQL, PostgreSQL, Microsoft SQL Server и другие БД — используется ADOdb XML). Moodle имеет возможность работать с объектами SCO и отвечает стандарту SCORM.

Плюсы:

- наиболее популярная система LMS в мире;
- бесплатная система, с открытым исходным кодом;
- большое онлайн-сообщество на официальном сайте moodle.org;
- подробная документация и множество учебных пособий;
- имеет встроенную систему разработки курсов, способную реализовать большое количество педагогических технологий, в том числе проектное обучение.

Минусы:

- система сложна для освоения;
- система требовательна к серверу;
- для расширения функционала или исправления возникающих в системе программных ошибок может понадобиться помощь профессионалов;
- отсутствует возможность создания курсов уровневой дифференциации.

Национальный Открытый Университет «ИНТУИТ» – организация, предоставляющая при помощи собственного веб-сайта сервисы дистанционного обучения по нескольким образовательным программам, многие из которых касаются информационных технологий [37]. Веб-сайт имеет несколько сотен открытых образовательных курсов, по прохождении которых возможно бесплатно получить электронный сертификат. Кроме того, возможно коммерческое получение сертификатов о повышении квалификации. Помимо

всего этого, организация функционирует как издательство, выпуская учебную литературу по курсам.

В Интуите возможно освоить более 500 курсов по разным областям информатики — так же, изучить различные языки программирования и разметки, численные методы, параллельные вычисления и так далее, есть несколько курсов по физике, математике, экономике, философии.

Claroline LMS – данная платформа для электронного обучения (eLearning) и электронной деятельности (eWorking), позволяющая учителям создавать эффективные онлайн-курсы и управлять процессом обучения и совместными действиями на основе веб-технологий [29]. Claroline LMS имеет обширное сообщество пользователей и разработчиков по всему миру, переведена на 35 языков.

Claroline LMS выпущена на базе лицензии с открытым кодом. Она используется во многих организаций 90 стран мира. Платформа позволяет создавать и администрировать курсы в режиме онлайн.

Предвидеть, что станет в дальнейшем трудно, хотя на данный момент можно выбрать необходимое программное обеспечение для автоматизации обучения, и не только дистанционного, но также очного и заочного.

Главные недостатки множества систем:

- отсутствие во многих системах русской «ВУЗовской» специфики организации учебного процесса, то есть документооборота между руководством университета, деканатами факультетов, кафедрами и так далее;
- отсутствие поддержки русского языка;
- отсутствие российских сертификатов по защите информации.

Возможно, что в будущем придется перейти на другую систему, хотя особо стоит обратить внимание на то, что останутся навыки работы с подобными системами, то есть будет создана инфраструктура поддержки подобных систем [38].

Сравнение систем приведено в таблице 1.1.

Таблица 1.1 – Сравнение систем

Критерии/Системы	MOODLE	Интуит	Claroline
Языки приложения	PHP	PHP	PHP, Java
СУБД	MySQL	MySQL	MySQL
Русский язык	+	+	+
Другие языки	>54	-	36
Система проверки знаний	тесты, задания, семинары, активность на форумах	тесты	тесты, упражнения
Демонстрационный сервер	+	-	+
Возможность участия в вебинарах	+	-	-
Легкость в освоении	-	+	-

На основе приведенного выше сравнения можно сделать вывод, что большинство систем сложные в освоении, что является не удобным и времязатратным для преподавателей.

1.3 Анализ внедрений систем дистанционного обучения

Некоторые крупные организации, в настоящее время, реализуют проекты по внедрению СДО персонала, то есть создают отраслевые, либо корпоративные системы дистанционного обучения. Системы дистанционного обучения создаются в отраслевых учебных заведениях Федеральной таможенной службы, РАО «Российские железные дороги» и так далее.

Кроме того, создают данные системы и образовательные учреждения, специализирующиеся на организации обучения, – Российская академия государственной службы при Президенте РФ, Академия народного хозяйства при Правительстве РФ, Государственный университет «Высшая школа экономики» и другие.

В числе положительных сторон внедрения систем дистанционного обучения их разработчики видят снижение издержек на обучение одного

обучаемого, исключение проблем, связанных с отрывом от производства, быстрая передача корпоративной культуры и современных профессиональных компетенций, легкость организации непрерывного обучения, а также возможность соблюдения единого образовательного стандарта вне зависимости от удаленности территориальных отделений организации. Опыт внедрения СДО в разных организациях имеет не только положительные результаты, но и отражает проблемы собственно процесса внедрения СДО.

Использование дистанционных образовательных технологий позволяет интенсифицировать учебный процесс методом ужесточения контроля знаний, широкого внедрения иллюстративного материала и виртуального эксперимента. Однако исключение «живого» общения преподавателя со студентами не позволяет полностью развивать их доказательную способность, способность вести активный диалог с оппонентом, способность к логическому мышлению. Центры дистанционного образования российских институтов уже используют сочетание традиционных и дистанционных форм. Как показывает практика, использование одного-единственного приема, либо метода обучения неэффективно.

Проект по внедрению СДО гораздо труднее, чем установка программного продукта в учебных классах. Комплексом мероприятий является внедрение, предполагающее адаптацию системы под задачи и оргпроцессы, наполнение системы контентом (курсами, материалами, тестами, справочниками), интеграцию в ИТ-инфраструктуру организации.

СДО, для успешного внедрения и использования которого необходимы согласованные действия ИТ-подразделений, представляет собой продукт менеджмента компании, специалистов HR-службы (от кадровиков, ответственных за учет персонала, до специалистов по оценке). «Система дистанционного обучения – это система автоматизации кадровой деятельности, интранет-портал, доступ к которому должен быть обеспечен каждому сотруднику организации независимо от местоположения и качества каналов

связи, библиотека учебного контента по самым различным темам: от продуктов и технологий компании до технологий менеджмента.

Внедрение СДО в организации предполагает появление целого ряда новых технологий и позиций (ролей), в том числе разработку электронных курсов, экспертную поддержку процесса дистанционного обучения и т.п. СДО постоянно наполняется и изменяется, адаптируется под потребности организации [14].

Методология организации и управления всем жизненным циклом проекта разработки, внедрения и сопровождения СДО должна базироваться на стандартах системы качества и стандартах управления проектами.

Управление системой взаимосвязанных процессов, организованных таким образом, чтобы способствовать достижению единых целей бизнеса более эффективным с точки зрения затрачиваемых ресурсов способом, это предполагает системный подход. Данный стандарт характеризуется системным подходом к внедрению информационных систем масштаба отрасли, поддерживающих процессы управления внедрением информационных технологий, и управлению.

Полный цикл работы над проектом внедрения СДО включает в себя следующие этапы:

- разработку стратегии развития СДО,
- подготовку проекта,
- концептуальное проектирование,
- реализацию проекта,
- заключительную подготовку,
- ввод в эксплуатацию и поддержка.

Так же можно выделить основные факторы, влияющие на СДО:

1. Преподавателям не стоит тратить много времени и сил на изучение сложных мультимедийных сред или сложных LMS. А сложные опции часто бывают не нужны.

2. Дистанционный курс, который разрабатывает преподаватель в поддержку своего очного курса, может быть свободен от недостатка, присущего чисто-дистанционным системам: провала в целевой аудитории. Преподаватель знает учащихся, с которыми работает. Можно встретиться в аудитории и обсудить материал. Поэтому дистанционная поддержка имеет много преимуществ перед чисто дистанционным курсом. Подготовив несколько дистанционных уроков и проанализировав результаты, преподаватель будет готов к построению дистанционного курса.
 3. Надо быть готовым к тому, что дистанционный курс - самостоятельный уникальный проект, а не простое перенесение очного курса в электронный вид.
 4. Во многих дистанционных курсах страдает этап закрепления. А часто слушатели не желают тратить на него время. Можно относиться к этому как к выбору слушателей. Но в условиях учебного заведения нам важно добиться понимания, усвоения, формирования навыка. Поэтому нужно организовать закрепление таким образом, чтобы слушатели хотели задержаться на этом этапе. Эту задачу может помочь решить, например, использование игр.
 5. Чем короче курс, тем больше слушателей пройдут его полностью. Можно построить курс из модулей: каждый модуль завершается контрольным мероприятием, слушатели получают отметку, сертификат и т.п. А дальше имеют возможность записаться на следующий модуль. Важна мотивация. И награда не должна быть слишком далёкой.
 6. Видео-лекции должны обязательно дублироваться презентациями и текстом. Во-первых, так удобнее читать и делать пометки. Во-вторых, не все хорошо усваивают информацию на слух.
- Новый материал.

Видеоматериалы преподаватель готовит в программе с помощью сервисов интернета. Затем видео загружается на сайт видеохостинга, а ссылка размещается в курсе (LMS).

- Материалы занятия.

Текст, презентации, графические файлы, музыкальные фрагменты. Готовится оффлайн с применением офисных технологий.

- Тест.

Тесты разрабатываются и анализируются в LMS.

- Обсуждение в дискуссионном форуме.

Основные темы предлагает преподаватель. Дополнительные - слушатели.

Итоговый контроль по всему курсу, как правило, планируется в очной форме.

Преподаватели имеют возможность опубликовать свой курс в каталоге и открыть доступ всем желающим. А могут допускать на свой курс только по приглашениям.

Образовательные онлайн-технологии развиваются высокими темпами.

1.4 Нормативное правовое обеспечение сферы дистанционного обучения

Система образования как институт общества является объектом управления со стороны государства. Оно определяет функции и цели, осуществляет финансирование, устанавливает рамки ее правовой деятельности, проводит образовательную политику. В рамках образовательной политики разрабатываются и принимаются соответствующие федеральные программы на государственном уровне, разрабатывается концепция развития системы образования. Ведущим направлением для развития образования в России является его информатизация. Под информатизацией образования понимается комплекс преобразований, связанных с введением в образовательную систему информационных средств и технологий, базирующихся на этих средствах.

В нормативных документах Российской Федерации и Министерства образования в качестве стратегии информатизации образования выступает рационализация интеллектуальной деятельности в связи с введением новых

информационных технологий, повышение качества и эффективности подготовки специалистов с современным типом мышления.

Для достижения выдвинутой цели необходимо привить обществу компьютерную грамотность и сформировать информационную культуру. Достигнуть эту цель можно путем информатизации образования. Данная цель будет сохранять свою актуальность еще долгое время.

Для достижения поставленной цели были сформированы задачи:

- необходимо подготовить учащихся к полному их участию в жизнедеятельности информационного общества;
- необходимо увеличить доступность образования путем внедрения систем дистанционного образования и средств массовой телекоммуникации;
- необходимо интегрировать системы образования во все сферы жизнедеятельности общества.

В результате ставится задача изменить систему начального, высшего, а также дополнительного образования, с целью ускорения развития интеллекта каждого человека.

На данный момент процесс информатизации Российской Федерации развивается очень медленно. Этому способствуют ряд причин:

1. Отсутствие единой методологии проектирования и внедрения в педагогическую практику информационных средств обучения в общеобразовательных учреждениях.
2. Оснащение образовательных учреждений компьютерной техникой происходит слишком медленно наряду с быстрым развитием компьютерной техники и программно-аппаратных средств.
3. Информационные средства применяются не повсеместно, автоматизируются лишь отдельные модули образовательного процесса, нет комплексного подхода к автоматизации.
4. Нет слаженного подхода различных министерств к разработке информационно-методического обеспечения.
5. Недостаточная обученность персонала.

Все это мешает выполнить поставленную цель в отношении информатизации образовательного процесса. Складывается противоречие между необходимостью и возможностью реализации современных технологий обучения.

В настоящее время сфера дистанционного обучения регламентируется рядом документов федерального законодательства:

- законом Российской Федерации от 13.01.1996 г. № 12-ФЗ «Об образовании» (в ред. от 05.03.2004 г.);
- федеральным законом от 22.08.1996 г. № 125-ФЗ «О высшем и послевузовском профессиональном образовании» (в ред. от 31.12.2005 г.);
- указом Президента Российской Федерации от 28.12.2006 г. № 1474 «О дополнительном профессиональном образовании государственных гражданских служащих Российской Федерации»;
- приказом Министерства образования и науки Российской Федерации от 06.05.2005 г. № 137 «Об использовании дистанционных образовательных технологий».

Указанные документы помогают раскрыть функцию дистанционного обучения и дистанционных образовательных технологий (ДОТ) в системе общего и профессионального образования. Кроме того, действуют нормативные документы, определяющие ряд существенных процедур функционирования учреждений образования, использующих ДОТ в образовательном процессе, в том числе Письмо Рособрнадзора от 11.06.2004 г. № 01-17/05-01 «О применении дистанционных образовательных технологий в образовательных учреждениях высшего, среднего и дополнительного профессионального образования».

«Созданные нормативные рамки констатируют возможность использования ДОТ в образовательном процессе. Но в связи с тем, что образовательная среда развивается высокими темпами, нормативное закрепление сложившихся форм ДОТ, утверждение перечня рекомендуемых образовательных ресурсов (порталов, баз знаний и т.п.) не состоялось в настоящий момент. [62]».

Вместе с тем, сфера нормативного правового регулирования соблюдения авторских прав принципиально изменилась с принятием Части IV Гражданского кодекса Российской Федерации (Глава 70; вступает в действие с 01.01.2008 г.). В этой связи использование мультимедийных учебных материалов и учебно-методических комплексов, созданных ранее, как и программных комплексов ДОТ, будет осуществляться в новом правовом пространстве. Также действуют такие нормативные правовые документы, как:

- закон Российской Федерации от 05.07.1993 г. № 5351-1 «Об авторском праве и смежных правах» (в ред. от 20.07.2004 г.);
- закон Российской Федерации от 23.09.1992 г. № 3523-1 «О правовой охране программ для электронных вычислительных машин и баз данных» (в ред. от 24.12.2002 г.).

Ответственность за нарушение авторских и смежных прав предусмотрена Кодексом об административных правонарушениях Российской Федерации; (статья 7.12).

В первой главе была определена цель, задачи, гипотеза, предмет и объект исследования, теоретическая и практическая значимость. Также проведен анализ предметной области, нормативно правовой сферы.

Глава 2 Проектирование технологии уровневой дифференциации обучения для электронно-образовательной среды ВУЗа

2.1 Технологии уровневой дифференциации

Организация учебного процесса, при которой учитываются индивидуально-типологические особенности личности (способности общие и специальные, интересы, уровень развития, психофизиологические свойства нервной системы и т.д.), характеризуется созданием групп учащихся, в которых содержание образования, методы обучения, организационные формы различаются – это дифференциация обучения.

Дифференциации обучения разделяется на два вида: дифференциация внутренняя (внутриклассная) и внешняя.

Внешняя дифференциация – это разделение студентов по определенным признакам (способностям, интересам и так далее) на стабильные группы, в которых и содержание образования, и способы обучения, и организационные формы различаются.

Внутренняя дифференциация принимает во внимание индивидуально-типологические специфики обучаемых в ходе обучения их в стабильной группе, созданной по случайным признакам. Разделение на группы может быть явным или неявным, состав групп изменяется зависимо от поставленной учебной задачи.

Виды дифференциации определяются, исходя из тех признаков (причин), который лежат в основе разделения студентов на группы. Традиционные виды дифференциации – это дифференциация по общим и специальным способностям, по интересам, проектируемой специальности.

По типу в дифференциации внутриклассной выделяются следующие виды:

- уровневая дифференциация, по способностям (формы: задания различного уровня сложности, дозирование помощи преподавателя);
- по интересам, проектируемой профессии.

По индивидуально-физиологическим отличительным чертам студентов внутренняя дифференциация существует традиционно в форме

индивидуального подхода к ним, когда учитываются их психофизиологические отличительные черты (темперамент, особенности мыслительных операций, доминирующий тип памяти и так далее).

Заметим, что внешняя дифференциация не опровергает, а напротив, подразумевает одновременное существование и внутренней в организации учебного процесса, потому что создаваемые при внешней дифференциации классы являются более или менее гомогенными по одному признаку, но гетерогенными по другим, что оставляет необходимый простор для внутренней дифференциации.

Дифференциация обучения подразумевает обязательный учет индивидуально-типологических особенностей студентов, различное построение учебного процесса в выделенных группах и форму их группирования. Не предполагает негативных последствий данное понимание дифференциации обучения, потому что учет индивидуально-типологических особенностей личности является обязательным, что приспособливает учебный процесс к обучающемуся. Но вместе с содержанием дифференциация обучения имеет и форму, в которой реализуется на практике. Например, могут быть группы углубленного изучения предметов, факультативные занятия, профильные, компенсирующего обучения, включенные в учебный процесс задания различного уровня сложности. При анализе форм дифференцированного обучения выделяются не только положительные, но и отрицательные проявления дифференциации. Рассмотрим их более подробно.

Внутриклассная дифференциация обучения, при которой внутри разнородного класса создаются группы учащихся по каким-либо признакам, чаще – по обучаемости, то есть по лёгкости и скорости усвоения учебного материала, распространена более широко в практике. Внутриклассная дифференциация выражается в заданиях разного уровня сложности, дозировании помощи преподавателя студентам. Данная гибкая, мягкая форма дифференцированного обучения, дает возможность переходить из группы в группу студентам, что является более комфортным для них. Внимание учащихся

на группы, которые выделяются неявно, не акцентируется. Однако в условиях внутриклассной дифференциации различное построение учебного процесса возможно в основном на этапе закрепления и обобщения знаний. Объяснение же нового учебного материала происходит одинаково для всех, преподаватель ориентируется на “среднего” студента, что тормозит развитие “сильных” и создаёт дополнительные трудности для “слабых”.

Группы компенсирующего обучения, создающиеся на первом курсе и сохраняющиеся до последнего курса включительно, стали ещё одной широко распространённой в практике формой дифференцированного обучения. Некоторые педагоги выступают против таких групп, считая их негуманными по отношению к студентам. Группы действительно будут негуманными относительно к учащимся, в случае если в них снижается уровень изучения материала и не предпринимаются попытки сделать лучше малоразвитые познавательные функции обучаемых. Основной идеей их создания является – развитие студентов группы риска, что сделать в условиях работы с 9-12 студентами легче, чем в обычной группе.

На занятиях в группе компенсирующего обучения преподаватель имеет возможность индивидуально поработать с каждым студентом, соотнести темп пары с возможностями учебной деятельности обучаемых, несколько раз объяснить материал. С моей точки зрения, главная проблема коррекционного обучения заключается в том, что студенты, перейдя в группу компенсирующего обучения и привыкнув к неспешному темпу с пошаговым разъяснением учебного материала, поощрением за мельчайший положительный результат, не захотят возвращаться в обычную группу. Группы компенсирующего обучения, таким образом, планировавшиеся как временные на этапе помощи студентам “риска”, приходится сохранять в последующие годы. Правда, в этом случае студенты действительно учатся, а не просто отсиживают на парах положенное время, чтобы потом находить вне института не всегда социально приемлемые сферы самореализации.

В организации групп повышенного уровня существуют острые проблемы. Набор такой группы “обедняет” другие группы, из которых уходят способные студенты. Студенты лишаются возможности ориентироваться на хорошо успевающих учащихся, слушать их ответы. Пары для них становятся “серыми” и скучными. В группах повышенного уровня появляются иные проблемы: у большинства студентов очень развита мотивация достижений и они стремятся реализоваться за счёт других. В такой группе складываются отношения соперничества и ревности, и отсутствует взаимопомощь и поддержка. В учебной сфере преподавателям очень сложно найти оптимальное соотношение между повышенной информативностью и развивающими элементами обучения.

Группы гибкого состава характеризуются тем, что учащиеся при изучении базовых дисциплин делятся на гибкие группы: из всего учебного потока организуется несколько групп: одна — желающих и способных углублённо изучать математику, другая — изучающих на общеобразовательном уровне, третья в темпе и объёме, специфичном для гуманитариев. Аналогично учебный процесс строится и на парах русского языка, литературы и других базовых дисциплинах. Основная проблема здесь — организационная. Возникают трудности с составлением расписания, так как трём преподавателям, например, математики, нужно одновременно провести уроки в трех аудиториях одного потока.

Предоставление студентам возможности выбрать предметы в дополнение к базисным — это элективная дифференциация. Такая форма, дифференцированного обучения позволяет студентам определиться в собственных предрасположенностях и способностях, чтобы в дальнейшем осознанно выбрать профиль обучения. С одной стороны, элективные курсы, должны быть небольшими по времени (полгода или год), чтобы студент мог выбрать другой курс, если понял, что ошибся, а с другой стороны — преимущественными.

Специальные дифференцированные задания — еще одна форма дифференциации. К таким заданиям можно отнести многовариантные

самостоятельные работы, а также задания с использованием специальных карточек.

Любая из рассмотренных форм дифференциации обучения имеет свои проблемы, снижающие эффективность образовательного процесса. Значит ли это, что университету необходимо отказаться от дифференциации процесса обучения? Нет, необходимо попытаться сгладить отрицательные проявления дифференциации и подготовиться к тому, что полностью устранить их не удастся. Необходимо признать, что в условиях традиционной системы наиболее комфортно чувствуют себя “средние” студенты, а в условиях дифференцированного обучения – “сильные” и “слабые”, а также студенты, имеющие ярко выраженные интересы. Дифференциация обучения ведёт к тому, что “средних” студентов, ничем не проявляющих себя в университете, остаётся всё меньше. В условиях дифференциации университет к каждому студенту относится как к уникальной, неповторимой личности. Оставаясь в рамках системы и используя при этом дифференциацию обучения, мы сможем приблизиться к личностной ориентации образовательного процесса.

«Разновидность внутриклассной дифференциации – дифференциация уровневая, при которой студент получает право и возможность выбирать уровень усвоения учебного материала (но не ниже минимального). Уровни усвоения предъявляются студентам в форме перечня знаний, умений и навыков, которые они должны приобрести, образцов задач, которые должны научиться решать. При данной форме дифференциации объяснения для всех студентов даются опять же на одном, чаще среднем или повышенном уровне. Для совершенствования этой формы дифференцированного обучения предлагалось повторять объяснение нового материала три раза (сначала на уровне минимальных требований, затем — обогатив материал, и, наконец, на уровне его углублённого изучения). [20]».

Дифференциация по уровню интеллектуального развития не получает в современной педагогике конкретной оценки; в ней есть вместе с

положительными и некоторые негативные аспекты, которые отображены в таблице 2.1.

Таблица 2.1 – Аспекты дифференциации

Положительные:	Отрицательные:
Отсутствуют нецелесообразные для общества усреднение студентов.	Деление по уровню развития студентов негуманно.
Появляется возможность у преподавателя помогать слабому, уделять внимание сильному.	Акцентируется социально-экономическое неравенство.
В группе отсутствуют отстающие, снимается необходимость в снижении общего уровня преподавания.	Слабые студенты не имеют возможности тянуться за более сильными, получать от них помощь, соревноваться с ними.
Появляется возможность более эффективно работать с трудными учащимися, плохо адаптирующимися к общественным нормам.	Перевод в слабые группы воспринимается студентами как унижение их достоинства.
Реализуется желание сильных учащихся быстрее и глубже продвигаться в образовании.	Несовершенство диагностики приводит к тому, что в разряд слабых переводятся неординарные студенты.
Повышается уровень Я-концепции: сильные утверждают в своих способностях, слабые получают возможность испытывать учебный успех, избавиться от комплекса неполноценности.	Понижается уровень Я-концепции: в элитарных группах возникает иллюзия исключительности, эгоистический комплекс; в слабых снижается уровень самооценки, появляется установка на фатальность своей слабости.
Повышается уровень мотивации ученья в сильных группах.	Понижается уровень мотивации ученья в слабых группах.

Рассмотрим подробнее несколько видов уровневой дифференциации.

Уровневая дифференциация обучения на основе обязательных результатов (В.В. Фирсов).

В этой технологии предполагается введение двух стандартов: для обучения и стандарта обязательной общеобразовательной подготовки. «Пространство между уровнями обязательной и повышенной подготовки заполнено своеобразной “лестницей” деятельности, добровольное восхождение по которой от обязательного к повышенным уровням способно реально обеспечить школьнику постоянное пребывание в зоне ближайшего развития, обучение на индивидуальном максимально посильном уровне [16]».

Нельзя представлять базовый уровень в виде “суммы знаний”, предназначенных для изучения в школе. Так как значительно не столько то, что изучалось, сколько то, что реально усвоено студентом. Потому его надлежит описывать в терминах планируемых результатов обучения, доступных проверке и контролю над их достижением.

Обязательность базового уровня для всех студентов в условиях гуманного обучения значит, что совокупность планируемых обязательных результатов обучения должна быть реально выполнима, то есть посильна и доступна основной массе обучающихся.

При демократической организации учебного процесса обязательность базового уровня, помимо всего этого, означает, что вся система планируемых обязательных результатов должна быть заблаговременно известна и понятна студенту (принцип открытости обязательных требований).

В форме, не допускающей разночтений, двусмысленностей и так далее, должен быть задан по возможности однозначно базовый уровень.

Базовый уровень должен обеспечивать гибкость и адаптивность, возможности для эволюционного развития. Являясь основным рабочим механизмом новой технологии обучения, не следует тесно увязывать с какой-либо одной методической системой и жестко фиксировать.

«Новая психологическая установка для учащегося: “возьми столько, сколько можешь, но не меньше обязательного”. Студент должен испытывать учебный успех. [27]».

Наличие стандартов базовых образовательных областей, состоит из двух уровней требований:

- содержание образования, которое институт обязан предоставить учащемуся;
- содержание образования, которое институт должен потребовать от учащегося, и усвоение которого является минимально обязательным для учащегося.

Уровневая дифференциация обучения предусматривает:

- присутствие обязательного базового уровня общей подготовки, которого обязан достичь учащийся;
- основой для дифференциации и индивидуализации требований к учащимся является базовый уровень;
- базовый уровень должен быть реально выполним для всех учащихся;
- должна быть открытой система результатов, которых должен достичь по базовому уровню учащийся (студент знает, что с него требуют);
- вместе с базовым уровнем студенту предоставляется возможность повышенной подготовки, определяющаяся глубиной овладения содержанием учебного предмета.

Отличительными чертами методики преподавания считаются:

- блочная подача материала;
- работа с небольшими группами на нескольких уровнях усвоения;
- наличие учебно-методического комплекса: система специальных дидактических материалов, банк заданий обязательного уровня, выделение обязательного материала в учебниках, заданий обязательного уровня в задачниках.

Систематическая повседневная работа по предупреждению и ликвидации пробелов путем организации пересдачи зачетов – это основное условие уровневой дифференциации по Фирсову.

Значимой особенностью технологии уровневой дифференциации обучения считается ее органическая взаимосвязь с системой контроля результатов учебного процесса и системой оценивания достижений студентов. Традиционному способу оценки “вычитанием” альтернативой является “оценка методом сложения”, в основу которой кладется минимальный уровень общей подготовки, достижение которого требуется в обязательном порядке от каждого учащегося. Критерии более высоких уровней строятся на системе зачетов и базе учета того, что достигнуто сверх базового уровня.

Учитывается:

- направленный на определенную тематику контроль;
- полнота проверки обязательного уровня подготовленности;
- открытость образцов проверочных заданий обязательного уровня;
- оценка методом сложения (общий зачет = сумма частных зачетов);
- двоичность в системе обязательного уровня (зачет-незачет);
- повышенные оценки за достижение сверх базового уровня;
- “закрытие” пробелов (досдача, но не пересдача);
- вероятность “дробных” зачетов;
- суммарность итоговой оценки (годовая оценка вытекает из всех полученных).

Зачеты проводятся в учебное время, при этом:

- для доработки учитывается резерв времени;
- возможна помощь преподавателя в период зачета;
- студентам даются “ключи” к проверочным заданиям;
- на каждого ведется лист учета и контроля;
- тогда, когда студент претендует на оценки 4 и 5, итоговый контроль предусматривает экзамен “на подтверждение” по всему материалу.

Внутриклассная (внутрипредметная) дифференциация (Н.П. Гузик).

«Автор назвал свою систему “Комбинированной системой обучения”, имеющей две отличительные стороны: внутриклассную дифференциацию обучения по уровню и развивающий цикл курсов по теме. [23]».

Пары по каждой учебной теме составляют пять типов, которые следуют друг за другом:

1. Пары общего разбора темы (их называют лекциями).
2. Комбинированные семинарские занятия с углубляющейся проработкой учебного материала в процессе самостоятельной работы учащихся (таких пар по каждой теме несколько, как правило, от трех до пяти).
3. Пары обобщения и систематизации знаний (так называемые тематические зачеты).
4. Пары межпредметного обобщения материала (защиты тематических заданий).
5. Практикумы.

Различия личностных качеств и других причин в группе, в силу неравномерности развития, появляются отличники, хорошисты, отстающие. Преподаватель поэтому организует уровневую работу этих студентов на паре, на всех этапах: при предъявлении нового материала, закреплении и повторении, при контроле ЗУН.

Выделяют три типа дифференцированных программ, разной степени сложности: “А”, “В”, “С”.

Дифференцированные программы (именно программы, а не задания) предусматривают два важнейших аспекта:

- обеспечение конкретного уровня овладения знаниями, умениями и навыками (от репродуктивного до креативного);
- обеспечение конкретной степени самостоятельности студентов в учении (от постоянной помощи со стороны преподавателя – работа по образцу, инструктаж и так далее до полной самостоятельности).

Существует строгая преемственность между программами “А”, “В”, “С”, каждой теме предоставлено обязательное минимальное значение, которое позволяет обеспечить непрерывную логику изложения и создать пусть неполную, но обязательно целостную картину главных представлений.

Задания программы “С” отмечены как базовый стандарт. Студент овладевает конкретным материалом по предмету на уровне его воспроизведения, выполняя программы. Работа по первичному усвоению материала на данном уровне имеет собственные отличительные черты. Требуется многократное его повторение, умения вычленять главное, выделять основные группы, знание приемов запоминания и так далее. Потому в содержание программы “С” вводится инструктаж про то, как обучать, на что направить свой взгляд, какой из этого следует вывод и так далее.

Каждый студент должен уметь выполнять задания программы “С”, прежде чем приступить к работе по более сложной программе (а лучше сказать, по следующей за ней программе).

Программа “В” обеспечивает овладение студентами теми общими и специфическими приемами учебной и интеллектуальной работы, которые нужны для решения задач на применение. Поэтому кроме определенных задач в данную программу вводятся дополнительные сведения, расширяющие материал первого уровня, иллюстрируют, доказывают и конкретизируют основное знание, демонстрируют функционирование и применение понятий. Данный уровень немного увеличивает объем сведений, делает общую картину более цельной, может помочь глубже понять основной материал.

Выполнение студентами программы “А”, поднимает их на уровень творческого, осознанного применения знаний. Программа предусматривает свободное владение фактическим материалом, умственных действий и приемами учебной работы. Она вводит студента в суть проблем, решаемых на основе полученных в ВУЗе знаний, дает развивающие сведения, углубляющие материал, его логическое обоснование, открывающие перспективы творческого

применения. Данный уровень позволяет студенту проявить себя в дополнительной самостоятельной работе.

Применяется методика свободного выбора разноуровневых заданий при повторении материала. Выделяют три варианта-уровня дидактического материала для решения задач, самостоятельных работ, практических заданий и лабораторных. Точно соответствует обязательным результатам обучения - первый вариант (С). Второй вариант (В) предполагает включение дополнительных задач и упражнений из учебника, третий (А) – заданий из вспомогательной учебно-методической литературы.

Выбор программы изучения каждого из предметов предоставляется лично студенту. Так обеспечивается общий для всех базовое (системное) минимальное значение знаний и одновременно открывается простор для развития творческой индивидуальности каждой личности.

При контроле знаний дифференциация переходит в индивидуализацию (индивидуальный учет достижений каждого учащегося) и углубляется. Внутрипредметная уровневая методика, по принципам и содержанию, сходна с методом “полного усвоения”. Переход к новому материалу осуществляется только после овладения учащимися общим для всех уровнем образовательного стандарта. Сочетание групповой и индивидуальной работы позволяет на фоне уровня базового стандарта выявить различия в знаниях учащихся. Для этого используются формы занятий: работа по группам (командам и тому подобному), работе в режиме диалога (постоянные пары, динамические пары), семинарско-зачетная система, модульное обучение, дополнительные индивидуальные занятия, индивидуализированное консультирование и помощь на паре, учет знаний по системе “зачет-незачет”.

Смешанная дифференциация (модель сводных групп).

«Объединенной формой двух видов дифференциации обучения – по интересам и по уровню развития является модель сводных групп по параллелям (модель гибкого состава класса). Образуются три сводные группы. [23]».

Студенты, имеющие в этом направлении достаточно высокие показатели и интересующиеся данным учебным предметом, объединяются в одну сводную группу (временную гомогенную группу) продвинутого уровня. Из остальных учащихся потока по принципу уровневой дифференциации (группа базового стандарта и группа усиленной педагогической поддержки) формируются еще две сводные группы.

Для потока разрабатываются три варианта учебной программы. Первый работает в рамках группы по интересу и имеет продвинутый уровень; второй и третий варианты реализуются в тех группах, для которых этот предмет не выбран “интересным” и главная цель которых – достижение обязательных результатов обучения.

Состав основных групповых коллективов при таком разделении остается неизменным, а состав меняется в зависимости от предметов дифференциации. По остальным предметам занятия ведутся по единым программам базового стандарта.

Таким образом, учебный процесс в течение дня организуется так: первые три-четыре пары идут по аудиториям, на четвертой-пятой паре группы распадаются – учащиеся в соответствии с выбранными ими направлениями переходят в сводные группы, где занятия ведут одновременно разные преподаватели по разноуровневым программам. Реализуется идея о форме дифференциации, которая не ущемляет достоинства учащихся и не нарушает сложившихся межличностных отношений в коллективах.

Модель сводных групп, объединяющая два вида дифференциации – по уровню достижений и по интересам.

Переформирование групп проводится по результатам итогового годового контроля. Возможны переходы учащихся из группы в группу и в середине учебного года.

2.2 Технология уровневой дифференциации для системы дистанционного обучения

На основе анализа технологий уровневой дифференциации для системы дистанционного обучения за основу была выбрана технология смешанной дифференциации (модель сводных групп).

Данная технология основывается на определении начального уровня знаний студентов и распределения их по группам в зависимости от результата тестирования, что наиболее подходит для дистанционного обучения.

Для данной технологии уровневой дифференциации было определено три уровня обучения: начальный, средний и высокий. Каждому уровню соответствуют определенные баллы:

- от 0 до 59 баллов «начальный» уровень;
- от 60 до 79 баллов «средний» уровень;
- от 80 до 100 баллов «высокий» уровень.

В начале курса студент проходит входное тестирование по дисциплине, где определяется его уровень знаний по данному курсу. Затем система оценивает результаты тестирования и предлагает начать обучение на определённом уровне (начальный, средний и высокий). Например, если после прохождения тестирования студенту был определен начальный уровень, то он обучается на данном уровне.

В процессе обучения студент осваивает учебный материал, который соответствует выбранному уровню обучения. После окончания обучения, учащиеся проходят итоговое тестирование, на котором определяется результат обучения. Важно отметить, что если студент проходит обучение на «начальном» уровне, то он может получить за прохождение курса только оценку «удовлетворительно», для «среднего» уровня – «хорошо».

Если учащегося устраивает оценка «удовлетворительно» для «начального» уровня или «хорошо» для «среднего» уровня, то он может закончить обучение. Если же студент хочет получить оценку выше, то он переходит на следующий уровень обучения.

На рисунке 2.1 отображена последовательность обучения студента с помощью технологии уровневой дифференциации.

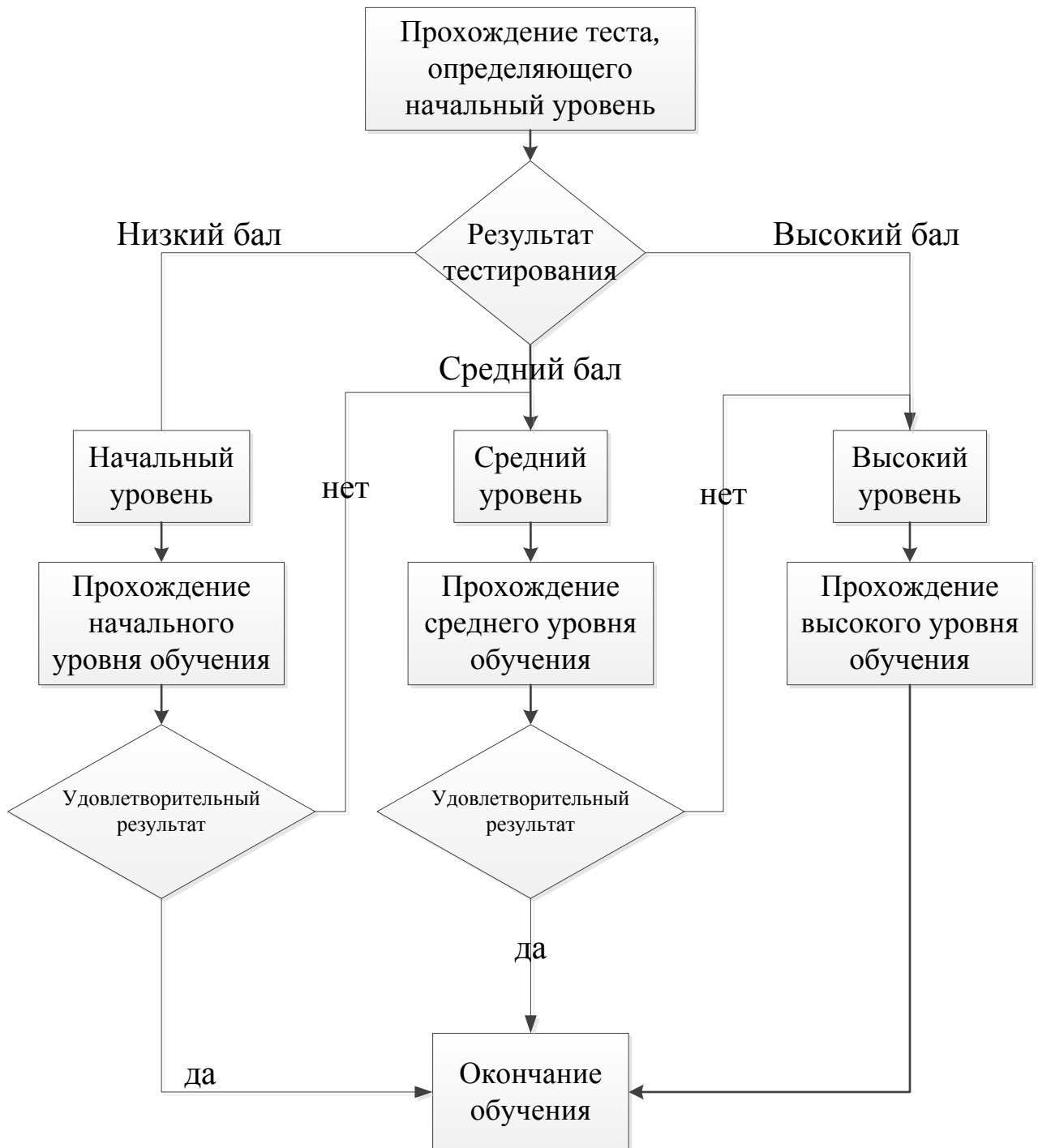


Рисунок 2.1 – Технология уровневой дифференциации для СДО

Для определения уровня знаний студентов, они проходят тест, определяющий начальный уровень, после прохождения, которого студент может приступить к началу обучения, это наглядно отображено на рисунке 1.1.

Если же студент прошел входное тестирование, набрав высокие баллы, то ему необходимо пройти обучение на «высоком» уровне.

В данной технологии уровневой дифференциации обучение на последующих уровнях происходит с постепенным наращиванием плотности учебной информации и с постепенным усложнением решаемых задач, в зависимости от уровня обучения. Повторяемость понятий на каждом последующем уровне приводит к лучшему восприятию и усвоению учебного материала. Последующий уровень накладывается на предыдущий, делая знания фундаментальными, прочными, востребованными, а студентов все более свободно владеющих учебным материалом курса.

Предлагаемая технология обучения направлена на развитие у студентов потребности в саморазвитии, самостоятельности и активности.

Во второй главе было дано определение уровневой дифференциации, разработана технология уровневой дифференциации для СДО.

Глава 3 Проектирование структуры системы дистанционного обучения

3.1 Основные подходы к разработке электронно-образовательной среды с функцией выбора уровня сложности обучения

В настоящее время существуют три подхода к проектированию информационных систем: структурный, модульный и объектно-ориентированный.

Структурный подход

Совокупность рекомендуемых технологических приемов, охватывающих выполнение всех этапов разработки программного обеспечения, представляют собой структурный подход к программированию. Внедрение его обеспечивает:

1. Увеличение производительности труда разработчиков программного обеспечения при написании и контроле программ.
2. Разработка программ коллективом разработчиков.
3. Получение программ, которые наиболее применимы для сопровождения, потому что состоят из отдельных модулей.
4. Окончание разработки программ в заданный срок.

В структурированные программы удобнее в отладке и менее чувствительны к ошибкам программирования. Данные характеристики считаются следствием важной отличительной черты подпрограмм, каждая из которых представляет собой во многом самостоятельный фрагмент программы, связанный с главной программой только при помощи нескольких характеристик. Такая самостоятельность подпрограмм разрешает локализовать в них все составные части программной реализации какого-нибудь алгоритмического действия, и поэтому изменение этих частей, например, в процессе отладки, обычно не приводит к изменениям основной программы.

Концепция структурного программирования предполагает разбиение программы на отдельные компоненты согласно принципу утаивания информации.

«Принцип утаивания информации заключается в том, что идентификаторы локальных объектов (имена констант, типов, переменных, процедур, функций, меток, и полей в записях переменных), то есть тех, которые используются только внутри заданной последовательности инструкций, не должны иметь смысла за пределами этих инструкций [17]».

Модульный подход

Концепцию модульного подхода сформулируем в виде нескольких положений и понятий:

1. Огромные задачи разбиваются на ряд наиболее маленьких, функционально самостоятельных подзадач - модулей, которые соединены друг с другом только по входным и выходным данным.
2. Модуль представляет собой «черный ящик» с одним входом и одним выходом, что позволяет безболезненно производить модернизацию программы в процессе ее эксплуатации, облегчает ее сопровождение, а также позволяет разрабатывать части программного проекта на разных языках программирования.
3. В каждом модуле должны исполняться понятные задачи. Если назначение модуля непонятно, это означает, что декомпозиция на модули была проведена недостаточно качественно. Процесс декомпозиции продолжается до тех пор, пока не будет четкого понимания назначения всех модулей и их оптимального сочетания.
4. Начальный текст модуля обязан иметь заголовок и интерфейсную часть, где отражаются назначение модуля и все его внешние связи.
5. В процессе разработки модулей программы следует предусматривать специальные блоки операций, учитывающие реакцию на возможные ошибки в данных или в действиях пользователя.

Огромное значение в концепции модульного подхода придается информационным связям между модулями программы, совместно решающими одну или несколько больших задач и организации управляющих.

При работе с модулями необходимо помнить основное их отличие от функций и процедур. Классические правила сферы действия локальных и глобальных переменных не работают для модулей. Данная языковая конструкция разработана так, чтобы исключить влияние глобальных переменных, объявленных в главной программе, на внутренние описания модуля. В связи с этим, если возникает необходимость ввести доступные для всех блоков программы глобальные описания, то следует создать модуль глобальных объявлений и включить его в список импорта всех модулей, где нужны его описания.

Модульное проектирование предполагает выделение групп подпрограмм, использующих одни и те же глобальные данные, в отдельно компилируемые модули (библиотеки подпрограмм), например, модуль графических ресурсов. Связи между модулями при использовании данной технологии осуществляются через специальный интерфейс, в то время как доступ к реализации модуля (телам подпрограмм и некоторым «внутренним» переменным) запрещен. Эту технологию поддерживают современные версии языков Pascal и C (C++), языки Ада и Modula.

Объектно-ориентированный подход

Методология проектирования, соединяющая в себе процесс объектной декомпозиции и приемы представления моделей, отражающих логическую (классы и объекты) и физическую структуру системы (процессы и деление на компоненты, файлы или модули), а также ее статические и динамические аспекты - это объектно-ориентированное проектирование (ООП).

Объектно-ориентированная парадигма становится парадигмой высокого уровня, и главенствует над принципами парадигмы структурного программирования, так как занимается моделированием реальности, строит модели предметных областей на языке специалистов этих областей. В случае если пренебречь этим ради написания хорошей программы, которую станет легко расширять, модифицировать, в которой будут независимые модули и четкие интерфейсы, то это будет означать возврат на уровень парадигмы

структурного программирования. Программа будет является хорошей, но её нельзя будет понять, потому что она не будет соответствовать реальности, будет объяснена в терминах только известных программисту, а специалист знающий предметную область не сможет разобраться в программе без помощи. В итоге, будет понижаться сложность в очень узком диапазоне, хотя и организована хорошая программа, но не модель. Лишь поверхностное представление модели, или её отсутствие, «взорвет» хорошую программу изнутри, и не даст сопровождать её в дальнейшем и дальше развивать.

Проектирование программных и технических систем при объектно-ориентированном подходе основывается на том, что ни одна подсистема этого уровня не должна зависеть от устройства любой другой подсистемы данного уровня. Инкапсуляцией называется независимость внутреннего устройства одного объекта от внутреннего устройства другого.

Рассмотрев основные подходы разработки ИС, можно сделать вывод, что структурный подход является наиболее удобным при построении функциональных моделей. Они явно отображают многофункциональную структуру объекта: производимые действия, связи между этими действиями. Таким образом, четко прослеживается взаимодействие процессов организации и логика.

Главным достоинством, благодаря её жестко регламентированной структуре, является возможность получить полную информацию о каждой работе. С её помощью можно выявить все недостатки, касающиеся как самого процесса, так и то, с помощью чего он реализуется: отсутствие механизмов, регламентирующих данный процесс, дублирование функций, отсутствие контрольных переходов и прочее.

3.2 Обоснование выбора технологии проектирования

Для решения комплекса задач необходимо программное обеспечение, которое можно настроить на особенности конкретной предметной области, а также использование, которого будет максимально эффективным для компании.

Для того чтобы непосредственно приступить к разработке приложения необходимо выбрать программные средства для реализации СДО.

Требования к операционной системе.

К операционной системе можно выдвинуть следующие требования:

- надежность (коды драйверов устройств должны быть доступны только для чтения, критически важные структуры ядра системы, операционная система должна позволять пользователям и системным администраторам восстанавливать раннее состояние компьютерной системы без потери данных);
- понятность пользователю (изменение параметров настроек системы должно производиться в диалоговом режиме, интерфейс операционной системы обязан быть «дружественным»);
- защита информации (операционная система должна иметь средства идентификации и аутентификации, иметь встроенные средства аудита доступа к информации);
- минимизация затрат на поддержку и сопровождение;
- эргономичность.

Используемая операционная система Windows 8 Professional удовлетворяет всем требованиям надежности, понятности пользователю, защиты информации, мобильности, масштабируемости, минимизации затрат на поддержку и сопровождение, эргономичности, описанным выше.

Требования к системе управления базами данных

К системе управления базами данных (СУБД), используемой для использования и разработки базы данных, можно выдвинуть следующие требования:

- надежность (СУБД должна позволять системным администраторам и пользователям восстанавливать предшествующее состояние СУБД без потери данных);
- защита информации (СУБД обязана иметь средства идентификации и аутентификации, иметь возможность разграничения доступа к объектам

базы данных – различные права на доступ к объектам базы данных должны принадлежать различным группам пользователей);

- модифицируемость (при помощи добавления новых объектов база данных должна легко расширяться);
- минимизация затрат на поддержку и сопровождение;
- эргономичность.

Некоторые характеристики основных СУБД представлены в таблице 3.1 в виде относительного сравнения.

Таблица 3.1– Некоторые характеристики основных СУБД

Характеристика	MS Access	Interbase	MySQL	MS SQL Server	Oracle
Размер базы данных	<100 МБ	<100 ГБ	<100 ГБ	>100 ГБ	>100 ГБ
Количество одновременных пользователей	<10	<1000	<100	>100	>100
Платформа	Windows	Windows/ Linux	Windows/ Linux	Windows/ Linux	Windows/ Linux
Защита данных	слабая	сильная	сильная	сильная	сильная
Требования к техническому оснащению	низкие	средние	низкие	высокие	высокие
Сложность настройки и администрирования	низкая	высокая	средняя	высокая	высокая

В качестве СУБД выбрана MySQL. MySQL – это свободная реляционная система управления базами данных.

MySQL является решением для небольших и средних приложений. Входит в состав серверов AppServ, WAMP, LAMP и в портативные сборки серверов XAMPP, Denwer. Чаще всего MySQL используется в качестве сервера, к которому обращаются локальные, либо удалённые клиенты, но в дистрибутив

входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД поддерживает огромное количество типов таблиц: пользователи могут выбрать как таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей, так и таблицы типа MyISAM, поддерживающие полнотекстовый поиск. Кроме того, MySQL распространяется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря GPL-лицензированию и открытой архитектуре, в СУБД появляются постоянно новые типы таблиц.

К программной среде, осуществляющее доступ к базе данных, где будет разрабатываться приложение, можно выдвинуть следующие требования:

- надежность (программная среда в течение заданного периода времени с достаточно большой вероятностью должна безотказно выполнять определенные функции при заданных условиях);
- отдельные задачи должны быть реализованы соответствующими программными модулями, каждый из которых будет иметь возможность выполнения ряда операций, таких как корректировка информации, поиск данных, формирование отчетов;
- процессы поиска данных, корректировки и заполнения должны быть реализованы однотипным способом во всех модулях. При добавлении и редактировании данных применяется контроль правильности ввода пользователем информации, путем ее сравнения с диапазоном изменения возможных значений, маской ввода, необходимым типом и т.д.;
- при добавлении, редактирования данных необходимо использовать экранные формы со всеми необходимыми полями ввода, управляющими элементами (например, кнопками), предназначенными для выработки управляющих воздействий (сохранение, отмена изменений) и навигации (перемещению) по БД и пояснениями;
- для обеспечения поиска данных следует использовать экранные формы, позволяющие задавать различные значения (диапазоны значений)

интересующей информации, с контролем допустимости значений условий поиска;

- в каждом из программных модулей системы предполагается наличие всех необходимых отчетных форм для формирования и печати документов установленной формы. Все отчеты должны генерироваться автоматически, используя выборки информации из БД;
- понятность пользователю (интерфейс программной среды должен быть «дружественным»);
- масштабируемость (программная среда должна корректно работать на больших и на малых системах с производительностью, которая увеличивается пропорционально вычислительной мощности системы);
- минимизация затрат на поддержку и сопровождение;
- эргономичность.

Наибольшей известностью пользуются системы PHP, Borland Delphi, Microsoft Visual Basic, Microsoft Visual C++. Каждый из указанных классов имеет свои достоинства и недостатки, поэтому в общем случае трудно отдать предпочтение одному из них. В предлагаемой работе в качестве средства разработки выбран язык программирования высокого уровня PHP, пользующийся большой популярностью.

Сравнительная характеристика некоторых существующих средств разработки представлена в таблице 3.2. Данная оценка является субъективным мнением автора работы, сформированным на основе собственного опыта и данных из источников, рассматривающих те или иные аспекты указанных средств разработки.

Критерии оценивания средств разработки:

- 1 – очень сложный в освоении материал;
- 2 – сложный в освоении материал;
- 3 – средней сложности в освоении материал;
- 4 – легкий в освоении материал;
- 5 – очень легкий в освоении материал.

Таблица 3.2 – Сравнительная характеристика некоторых средств разработки

Аспект	Delphi	Visual Basic	Visual C++	PHP
1. Развитые средства визуального проектирования	3	2	5	3
2. Обширный набор визуальных компонентов	5	2	5	5
3. Развитость встроенного генератора кода	5	3	2	4
4. Удобство встроенного отладчика	5	2	2	5
5. Наличие простого в освоении механизма доступа к базам данных	2	2	3	5
Итоговое количество баллов	20	11	17	22

В результате оценки по выбранным критериям по 5-ти балльной шкале (отражающей субъективное мнение автора) средств разработки, указанных в таблице 3.2, максимальное количество баллов получил PHP.

Таким образом, лучшим вариантом для разработки системы является язык PHP в связке с БД MySQL, так как PHP является наиболее простым и удобным языком для разработки динамических сайтов. MySQL была выбрана как СУБД для разработки сайта, т.к. она является стандартной, а также удобной для разработки сайтов на PHP. Также при разработке использовался язык JavaScript для того, чтобы «оживить» страницу и сделать сайт удобным для пользователя.

3.3 Структурная и функциональная модели информационного обеспечения системы организации дополнительного образования

Структура образования – это состав, порядок и принципы взаимодействия элементов системы, определяющие основные свойства системы.

В состав СДО входят все данные, необходимые пользователю для работы. Взаимодействие пользователя с системой осуществляется в пользовательском режиме. Вначале пользователь обязательно проходит авторизацию. На рисунке 3.1 представлена структурная модель СДО.



Рисунок 3.1 – Структурная схема СДО

Из представленной структуры видно, что от статуса учётной записи зависит, какими модулями пользуются пользователи системы. На рисунке 3.2 представлена функциональная схема системы.

Функциональная схема — служит для разъяснения определенных процессов, происходящих в отдельных частях изделий или в целом изделии; используется для изучения принципов работы изделий (установок, устройств), а также при наладке, регулировании, контроле и ремонте.

В СДО в зависимости от типа учетной записи пользователи выполняют различные по своему роду функции.

Например, администратор системы имеет возможность зарегистрироваться, проектировать расписание, организовывать курсы.



Рисунок 3.2 – Функциональная модель СДО

В процессе функционирования системы СДО предполагается накопление структурированной информации, осуществление непрерывного анализа результатов и формирование единой базы данных.

Таким образом, из рассмотренных бизнес-процессов можно сделать выводы о том, что в разрабатываемой системе должны быть реализованы такие функции как:

- добавление и редактирование списков групп и студентов;
- добавления преподавателей;
- прохождение тестирования;
- выбор курсов;
- добавление курсов;
- формирование расписания;
- редактирование расписания;
- добавление и редактирование данных преподавателем теоретического и практического материала;
- оформление данных о результатах выполнения заданий учащимися;

- наглядное представление информации о результатах работы учащихся.

Таким образом, были построены функциональная и структурная модели информационного обеспечения систем дистанционного образования. Из приведенных моделей видно, что данная система состоит из студента, администратора и преподавателя. СДО должна проектироваться на основании выделенных выше функций системы и ее состава. Для демонстрации работы некоторых функций необходимо рассмотреть алгоритм работы модулей системы.

3.4 Проектирование информационного обеспечения системы дистанционного образования

Для того чтобы информационная система в дальнейшем смогла реализовываться без выявления ошибок и недоработок в самом ее проектировании необходимо провести работу по изучению и анализу предметной области. Анализ предметной области проводился с использованием методологии построения диаграмм UML: вариантов использования и деятельности. Общецелевой язык визуального моделирования, который разработан для визуализации, спецификации, документирования и проектирования компонентов программного обеспечения, бизнес-процессов и других систем является язык UML.

Унифицированный язык моделирования (UML) является стандартным инструментом для создания "чертежей" программного обеспечения. С помощью UML можно визуализировать, специфицировать, конструировать и документировать артефакты программных систем.

UML пригоден для моделирования любых систем: от информационных систем масштаба предприятия до распределенных Web-приложений и даже встроенных систем реального времени. Несмотря на свои достоинства, UML - это всего лишь язык; он является одной из составляющих процесса разработки программного обеспечения, и не более того. Хотя UML не зависит от моделируемой реальности, лучше всего применять его, когда процесс моделирования основан на рассмотрении прецедентов использования, является

итеративным и пошаговым, а сама система имеет четко выраженную архитектуру.

При бизнес-анализе моделирования видов работ, выполняемых организацией, и для моделирования функциональных требований к проектируемой системе при ее проектировании и разработке применяются диаграммы вариантов использования. Построение модели требований при необходимости дополняется их текстовым описанием.

Построение диаграммы вариантов использования, с системно-аналитической точки зрения, специфицирует не только функциональные требования к проектируемой системе, но и выполняет исходную структуризацию предметной области.

Диаграмма вариантов использования отражает организационный аспект системы. Разработка диаграммы преследует следующие цели:

- определить контекст моделируемой предметной области и общие границы;
- сформулировать к функциональному поведению проектируемой системы общие требования;
- разработать для ее последующей детализации в форме логических и физических моделей исходную концептуальную модель системы;
- подготовить для взаимодействия разработчиков системы с ее заказчиками и пользователями исходную документацию.

Изучив все правила построения данного типа диаграмм, получена диаграмма вариантов использования бизнес-процесса организации дополнительного образования, представленная на рисунке 3.3.

Как следует из диаграммы, актер Администратор выполняет функции по созданию/редактированию расписания, работе с сообщениями и организации курсов.

Преподаватель занимается работой, включающей полный функционал по информационному обеспечению учебного процесса.

Студент имеет права доступа к тестированию, курсам, работе с сообщениями, просмотру расписания и своей успеваемости.

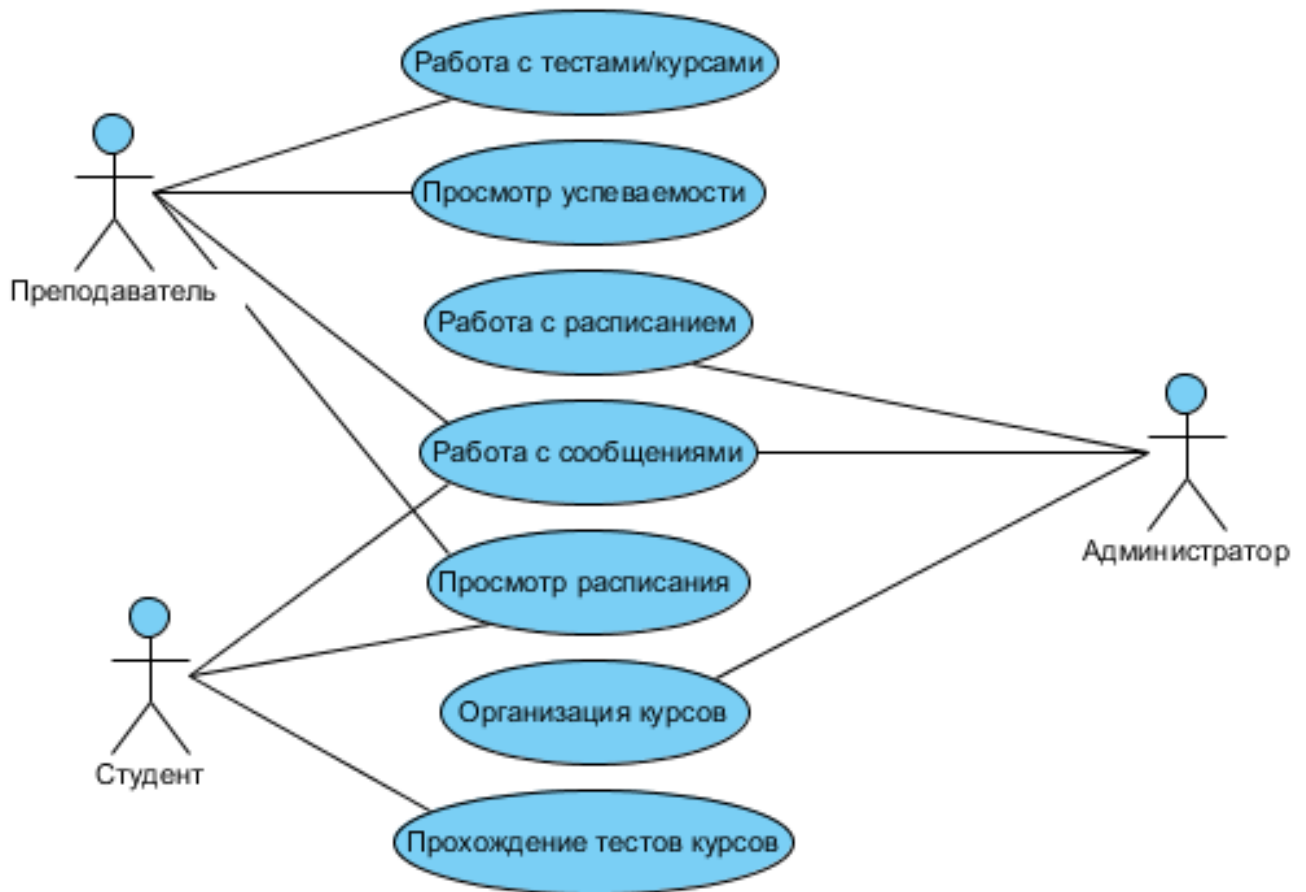


Рисунок 3.3 – Диаграмма вариантов использования бизнес-процесса
организации дистанционного образования

При моделировании поведения проектируемой или анализируемой системы возникает необходимость не только представить процесс изменения ее состояний, но и детализировать особенности алгоритмической и логической реализации выполняемых системой операций. Для этой цели используются блок-схемы или структурные схемы алгоритмов. Каждая такая схема акцентирует внимание на последовательности выполнения определенных действий или элементарных операций, которые в совокупности приводят к получению желаемого результата. Для визуализации порядка действий используется диаграмма деятельности. Для проектирования работы разрабатываемой системы был описан алгоритм авторизации пользователя в СДО на рисунке 3.4.

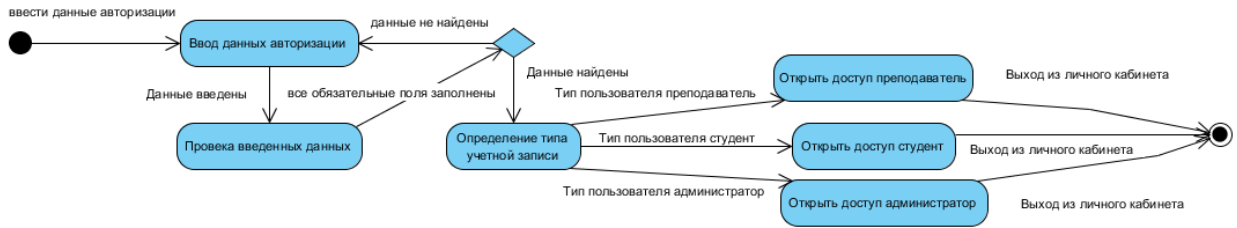


Рисунок 3.4 – Диаграмма авторизации пользователя в нотации UML
(диаграмма деятельности)

В результате прохождения процедуры авторизации, пользователь начинает работу с функциями приложения в зависимости от назначенных прав доступа данной учетной записи.

Функция формирования курса является одной из самых важных в системе так как именно она позволяет студентам выбрать уровень обучения, наиболее подходящий для себя. Диаграмма формирования курса представлена на рисунке 3.5. Диаграмма выполнена в нотации UML (диаграмма деятельности).

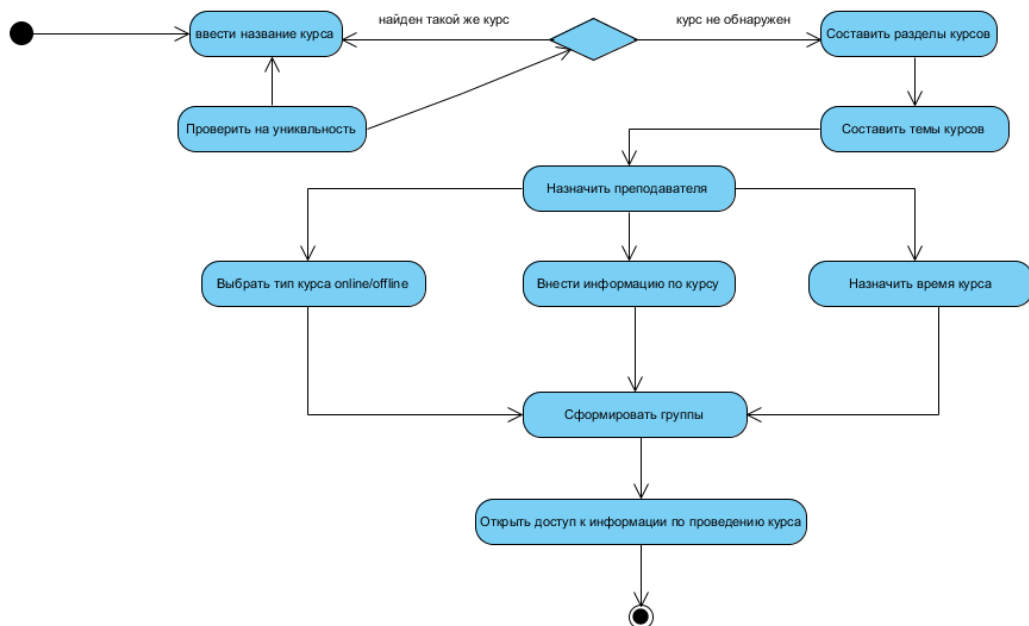


Рисунок 3.5 – Диаграмма формирования курса в нотации UML (диаграмма деятельности)

Таким образом, проведя дополнительный анализ разрабатываемой системы, были определены основные модули системы, продемонстрированы алгоритмы и процессы их работы. Были построены диаграмма вариантов

использования, диаграмма авторизации пользователя, диаграмма формирования курса в нотации UML.

3.5 Функциональное моделирование организации процесса обучения в вузе

В условиях все большего инвестирования каждого работника образовательной сферы в информатизацию общества в настоящее время имеются различные варианты автоматизирования организации обучения. Можно встретить как коммерческие системы, автоматизирующие деятельность образовательных учреждений, которые содержат в своем составе модуль, касающийся обучения, так и бюджетные (бесплатные).

Большинство программ содержат в себе такие модули как: автоматизация кадровой работы, систематизации данных об учениках, создание, заполнение справочников, поддержку содержания образования, администрирование образовательного и воспитательного процесса, автоматизация финансовой деятельности учреждения образования.

Для защиты персональных данных имеются возможности выгрузки базы данных с паролем, хранение, заполнение и ведения листов согласия учащихся и сотрудников.

При конфигурации программы первым этапом является заполнение базовой информации справочников образовательного учреждения, заполнение его структуры, а в последствии заполнение справочников детализированной информацией. При этом формируется общий доступ и периодизированные компоненты. Последовательность работы варьируется из соображений руководителя учреждения образования. Далее происходит ведение и управление базовой информацией и справочниками. Программный комплекс содержит расширенный модуль отчетности как промежуточной, внешней, так и итоговой. В данном программном комплексе также возможно осуществить рассылку родителям об успеваемости за выбранный период. Рассылка осуществляется по электронной почте.

В ходе анализа существующих разработок сложилась концепция создания собственного программного обеспечения, которое будет покрывать недостатки

существующих программных комплексов, внесет свои новые полезные идеи и разработки, будет направлена на более узкую область в целях ее большей автоматизации.

Построение эффективной информационной системы является первым этапом исследования. В течение процесса проектирования применяется несколько типов моделей. Каждая модель сформирована для удовлетворения отдельных целей. Их информационные источники и последовательность создают логическую модель.

Модель «ТО-ВЕ» («как должно быть») – функциональная модель, отражающая структуру системы, внешние информационные связи, информационные потоки между подразделениями, наличие вычислительной сети, степень компьютеризации и т.п.

Модель бизнес-процесса «как должно быть» описывает принципы и механизмы функционирования ИС как единого организма. На рисунке 3.1 представлена контекстная диаграмма бизнес-процесса «Организация процесса обучения в ВУЗе».

Входные данные: информация о студенте, УМКД.

Выходные данные: выполненные задачи, отчеты и график работы сотрудников.

На рисунке 3.6 изображен процесс «Организация курса дистанционного обучения», состоящий из подпроцессов: «Сбор материалов», «Разработка курса», «Подведение итогов», «Формирование отчетов» содержит бизнес-процессы, относящиеся к систематизации данных, проверка результатов обучения и формирования отчетности.



Рисунок 3.6 – Контекстная диаграмма «Организация курса дистанционного обучения» в нотации IDEF0

На рисунке 3.7 изображена декомпозиция процесса «Организация дистанционного обучения». Входные данные у подпроцесса «Сбор материалов»: УМКД, выходные данные: материалы для курса. Управление осуществляется с помощью учебного плана. Исполняющим механизмом является преподаватель.

Следующий подпроцесс «Разработка курса», где входные данные: материалы для курса, выходные данные: созданный курс. Управление осуществляется с помощью стандартов и нормативных документов. Исполняющим механизмом является преподаватель и ЭОС.

Третий подпроцесс «Подведение итогов», где входные данные: созданный курс, выходные данные: информация о студенте, результаты. Управление осуществляется с помощью нормативных документов. Исполняющим механизмом является преподаватель и ЭОС.

Последний подпроцесс «Формирование отчетов», где входные данные: информация о результатах, выходные данные: результаты обучения. Управление осуществляется с помощью нормативных документов. Исполняющим механизмом является преподаватель и ЭОС.

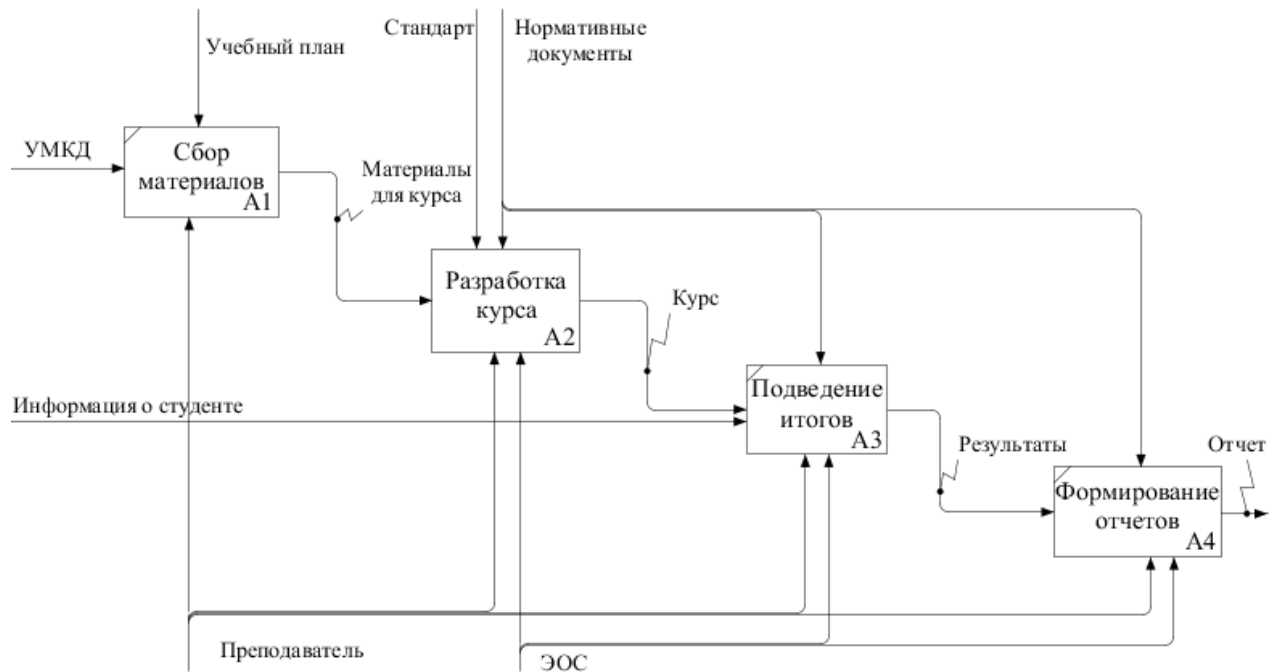


Рисунок 3.7 – Диаграмма декомпозиции бизнес-процесса «Организация курса дистанционного обучения»

Бизнес-процесс «Разработка курса» разбивается на подпроцессы «Разбиение материала по уровням сложности», «Заполнение курса по уровням сложности», «Проведение дополнительных занятий».

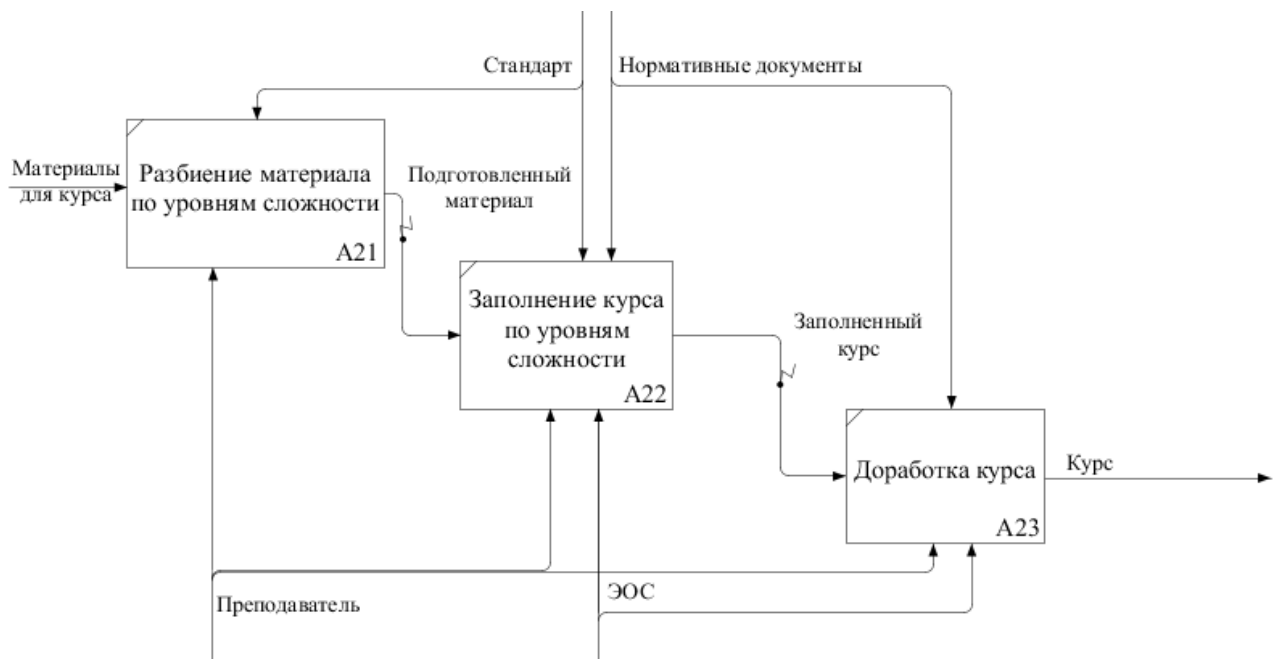


Рисунок 3.8 – Диаграмма декомпозиции бизнес-процесса «Разработка курса»

Для наглядности опишем модель ТО-ВЕ с помощью диаграммы потоков данных (DataFlowDiagrams), методология графического структурного анализа, описывающая внешние по отношению к системе источники и адресаты данных,

логические функции, потоки данных и хранилища данных, к которым осуществляется доступ.



Рисунок 3.9 – Диаграмма DFD «Организация процесса обучения в ВУЗе»



Рисунок 3.10 – Декомпозиция диаграммы DFD «Организация курса дистанционного обучения»

На рисунке в представленной диаграмме потоков данных видно, что в системе будут использоваться 2 роли: «Студент» и «Преподаватель». Основными функция системы будут: «Разработка курса», «Работа с курсом», «Добавление результатов обучения», «Работа с отчетами».

3.6 Инфологическая модель данных электронно-образовательной среды обучения уровневой дифференциации

Информационная модель – модель объекта, представленная в виде информации, описывающей существенные, для данного рассмотрения, параметры и переменные величины объекта, связи между ними, входы и выходы объекта и позволяющая, путём подачи на модель информации об изменениях входных величин, моделировать возможные состояния объекта (иногда ее также называют информационно-логической или концептуальной моделью).

Концептуальная модель – отображает информационные объекты, их свойства и связи между ними без указания способов физического хранения информации.

На основе диаграммы потоков данных были определены основные сущности предметной области:

- курс;
- сообщения;
- пользователь;
- группа;
- журнал.

Инфологическая модель данных электронно-образовательной среды обучения уровневой дифференциации представлена на рисунке 1 в нотации Чена.

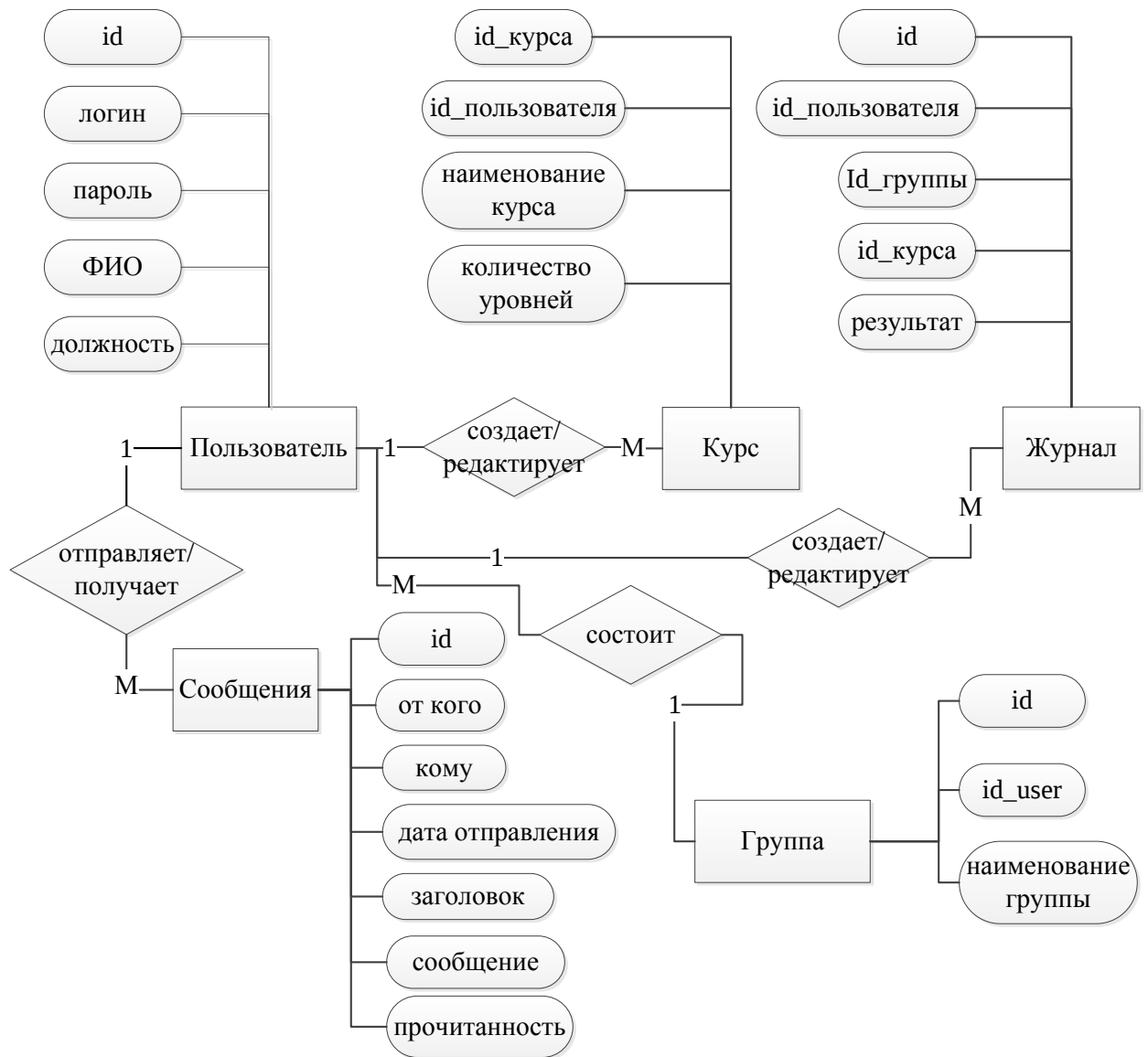


Рисунок 3.11 – Инфологическая модель данных СДО

На рисунке 3.11 изображена ER-модель в нотации Чена, на которой представлены 5 сущностей, отношения между которыми приведены к третьей нормальной форме.

Сущности «Пользователь» и «Сообщения» имеют связь один-ко-многому; так же сущности «Пользователь» и «Курс» имеют связь один-ко-многому, сущности «Пользователь» и «Журнал» имеют связь один-ко-многому, сущности «Пользователь» и «Группа» имеют связь один-ко-многому.

3.7 Даталогическая модель данных электронно-образовательной среды обучения уровневой дифференциации

Логическая модель данных является начальным прототипом будущей базы данных. Логическая модель описывает понятия предметной области, их

взаимосвязь, а также ограничения на данные, налагаемые предметной областью. Логическая модель строится в терминах информационных единиц, но без привязки к конкретной СУБД. Типы даталогических моделей – это есть не что иное, как модели представления данных, таким образом даталогическая модель данных может быть реляционной, иерархической или сетевой.

В качестве даталогической модели базы данных была выбрана реляционная модель, потому что в этой модели все данные хранятся в виде таблиц и в упорядоченном виде, и записи не повторяются, их уникальность обеспечивается первичным ключом, содержащий несколько полей, однозначно определяющих запись. Для быстрого поиска в реляционных таблицах создаются индексы по одному или нескольким полям таблицы. Значения индексов хранятся в упорядоченном виде и содержат ссылки на записи таблицы. Для автоматической поддержки целостности связанных данных, находящихся в разных таблицах, используются первичные и внешние ключи. Для выборки данных из нескольких связанных таблиц используются значения одного или нескольких совпадающих полей.

Основным средством разработки логической модели данных в настоящий момент являются различные варианты ER-диаграмм (Entity-Relationship, диаграммы сущность-связь). ER-диаграмма – это графическое представление взаимосвязей сущностей. Одну и ту же ER-модель можно преобразовать как в реляционную модель данных, так и в модель данных для иерархических и сетевых СУБД, или в постреляционную модель данных.

Для того чтобы построить логическую и физическую модели мы использовали ERwin - CASE-средство для проектирования и документирования баз данных, которое позволяет создавать, документировать и сопровождать базы данных, хранилища и витрины данных.

Даталогическая модель данных представлена на рисунке 3.12.

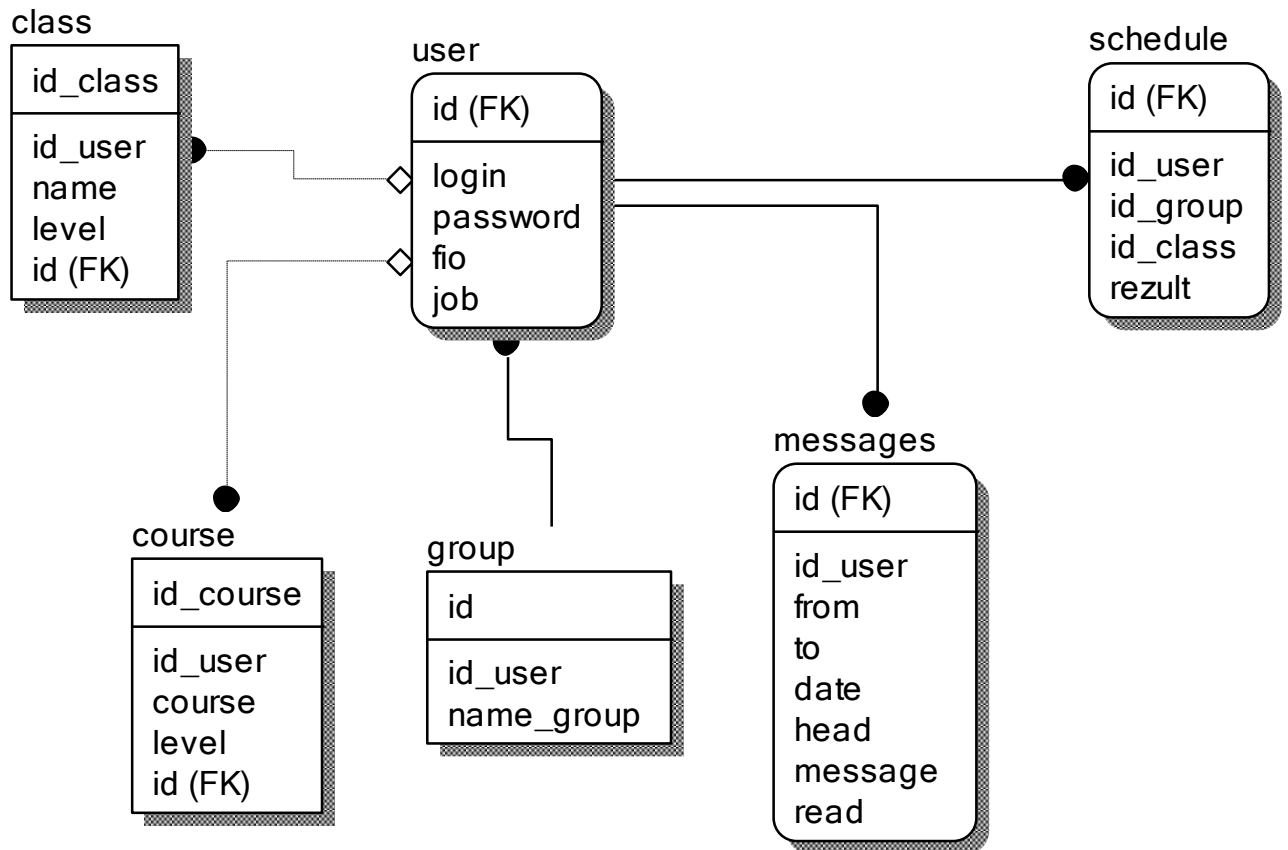


Рисунок 3.12 – Даталогическая модель данных «Электронно-образовательной среды обучения уровневой дифференциации»

Данные в БД должны обладать свойством целостности. Под целостностью данных понимается корректность данных и их непротиворечивость в любой момент времени. Поддержание целостности базы данных может рассматриваться как защита данных от неверных изменений или разрушения (этот вопрос не относится к незаконным изменениям и разрушениям, которые являются проблемой безопасности).

В разрабатываемой структуре базы данных учтены основные правила целостности. Каждая сущность идентифицируется уникальным ключом и разработана система внешних ключей. БД не содержит несогласованных значений внешних ключей, т.е. при работе с записями происходит каскадное обновление связанных полей и каскадное удаление связанных записей.

3.8 Разработка физической модели базы данных электронно-образовательной среды обучения уровневой дифференциации

Физическая модель данных зависит от конкретной СУБД, фактически являясь отображением системного каталога. В физической модели содержится информация обо всех объектах БД. Поскольку стандартов на объекты БД не существует, физическая модель зависит от конкретной реализации СУБД. Следовательно, одной и той же логической модели могут соответствовать несколько разных физических моделей.

Таблица 3.3 – Описание реляционной модели базы данных

Наименование таблицы	Наименование поля	Тип поля
Schedule	id	TinyInt
	id_user	TinyInt
	id_group	Byte
	id_class	Byte
	result	VarChar
User	id	TinyInt
	login	VarChar(20)
	fio	VarChar(20)
	password	VarChar(20)
	job	VarChar(20)
Class	id	TinyInt
	id_user	TinyInt
	name	VarChar(20)
	lavel	VarChar(20)
Message	id	TinyInt
	from	TinyInt
	to	TinyInt
	dispatch date	Date
	head	Text
	message	Text
	read	TinyInt
Course	id_course	TinyInt
	id_user	VarChar(20)
	course	VarChar(20)
	level	TinyInt
	id	TinyInt
Group	id	TinyInt
	id_user	TinyInt
	name_group	VarChar(20)

Если в логической модели не имеет значения, какой конкретно тип данных имеет атрибут, то в физической модели важно описать всю информацию о конкретных физических объектах.

Целостность, определяемая пользователем, поддерживается ограничениями в таблицах базы данных на ввод неотрицательных значений, а также обеспечением выбора значений внешних ключей из списков без разрешения варианта ввода недопустимого значения.

Описание реляционной модели базы данных приведено в таблице 3.3.

Данная модель показывает типы данных и длины полей таблиц базы данных разрабатываемой информационной системы. С помощью системы управления базами данных MySQL была реализована структура таблиц в соответствии с физической моделью данных.

3.9 Выбор архитектуры электронно-образовательной среды обучения уровневой дифференциации

Предполагаются следующие условия использования системы:

- сотрудники компании находятся на своих рабочих местах;
- одновременно с системой могут работать несколько пользователей;
- всеми пользователями используется единая база данных (БД), хранящая информацию обо всех сообщениях и документах;
- каждый пользователь имеет доступ в соответствии со своими правами доступа, определяемыми должностью (руководитель, менеджер).

Таблица 3.4 – Права доступа пользователей системы

Функция	Администратор	Преподаватель	Студент
Создание пользователей	+	-	-
Редактирование курсов	-	+	-
Составление расписания	+	-	-
Составление отчетов	+	+	+
Обмен сообщениями	+	+	+
Просмотр курсов	+	+	+

Права доступа пользователей системы представлены в таблице 3.4.

Исходя из вышеуказанных условий, выбрана распределенная архитектура системы: БД находится на сервере, а пользователи получают к ней доступ посредством клиентского приложения, установленного на их рабочем месте.

Под клиент-серверным приложением понимается информационная система, основанная на использовании серверов баз данных.

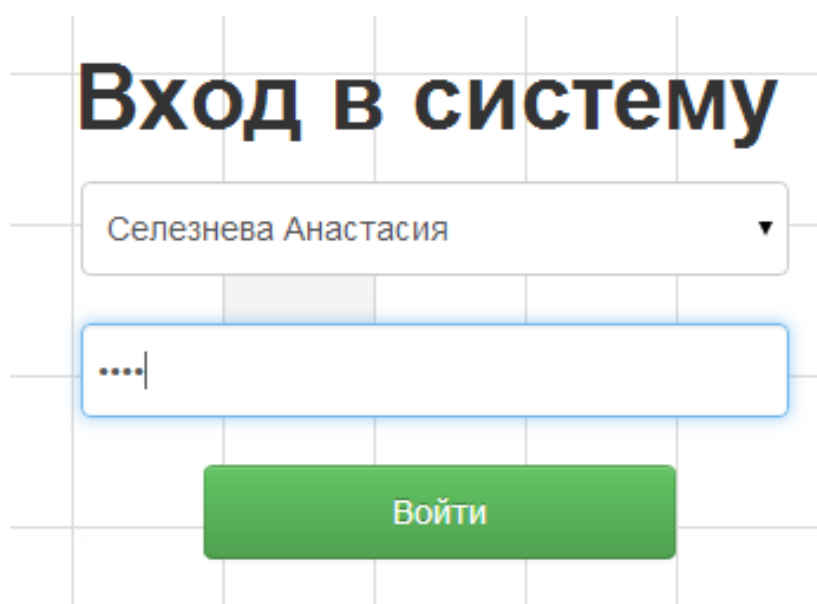
На стороне клиента выполняется код приложения, в который обязательно входят компоненты, поддерживающие интерфейс с конечным пользователем, производящие отчеты, выполняющие другие, специфичные для приложения функции.

Клиентская часть приложения взаимодействует с клиентской частью программного обеспечения управления базами данных, которая, фактически, является индивидуальным представителем СУБД для приложения.

Исходя из этого, делается вывод, что разрабатываемая электронная системы является клиент-серверным приложением.

3.9 Описание технической и программной структуры

Основным назначением системы является автоматизация ввода и хранения данных о курсах. Система позволяет изменять, дополнять, вести поиск и просмотр информации о курсах. Система должна облегчить создание и изучение курса.



The image shows a login window titled "Вход в систему". It contains a dropdown menu with the text "Селезнева Анастасия" and a small downward arrow. Below this is a password field with masked characters "....". At the bottom is a green button with the text "Войти".

Рисунок 3.13 – Окно авторизации

При открытии системы пользователь должен авторизоваться на форме, приведенной на рисунке 3.13. Форма содержит поля для ввода ФИО и пароля, а также кнопку ввода данных.

Работая с системой преподавателю необходимо взаимодействовать со студентами по средствам переписки.

The screenshot shows a web application interface with a dark teal header. On the right side of the header, it says "Вы зашли под пользователем: Селезнева Анастасия". Below the header is a navigation bar with tabs: "Задачи", "Сообщения" (highlighted), "Календарь событий", "Отчеты", "Расписание", and "Клиенты". The main content area is titled "Мои сообщения" and contains three sub-sections: "Написать" (active), "Входящие", and "Отправленные". The "Написать" section includes a dropdown menu labeled "Выберите получателя", a text input field labeled "Введите заголовок сообщения", a larger text area labeled "Введите текст сообщения", and a green button labeled "Отправить".

Рисунок 3.14 – Форма обмена сообщениями в электронно-образовательной среде обучения уровневой дифференциации

Для того чтобы увидеть необходимый подраздел из раздела, нужно на него кликнуть. Т.е. в подразделе «Написать», можно отправить сообщение пользователю. В подразделе «Входящие» отображаются сообщения, которые были отправлены другими пользователями, а в «отправленные» соответственно отображаются сообщения, которые были отправлены другим пользователям.

The screenshot shows a web application interface with a dark teal header. The navigation bar has tabs: "Личная страница", "Сообщения", "Рабочие курсы", "Создание курса" (highlighted), and "Журнал". The main content area is titled "Введите наименование дисциплины" and contains a text input field with the value "Линеал". Below this is another section titled "Количество уровней сложности" with a text input field containing the value "2". At the bottom is a green button labeled "Создать".

Рисунок 3.15 – Форма создания курса в электронно-образовательной среде обучения уровневой дифференциации

Вы зашли под пользователем: Селезнева Анастасия [Выход](#)

Задачи Сообщения Календарь событий Отчеты Расписание Студенты **Курсы**

1 уровень

Множества, элементы, операции

Над вещественными (действительными) числами a, b выполняются следующие алгебраические операции и для них используются следующие обозначения:

сложение $a+b$

вычитание $a-b$

умножение $a \cdot b, a \cdot b, a \times b$

деление $a/b, a/b, a/b$

обратная величина $b^{-1}, 1/b, 1/b$

Добавить подуровень

2 уровень

Введите заголовок

Добавить подуровень

Сохранить изменения

Рисунок 3.16 – Форма создания курса в электронно-образовательной среде обучения уровневой дифференциации

При прохождении курсов, студенту необходимо отметить пройденность курса, так же есть возможность просмотреть его еще раз.

1 Множества, элементы, операции		
1.1 Вещественные и комплексные числа		
1.1.1 Вещественные числа		
1.1.2	Алгебраические свойства вещественных чисел	☒
1.1.3	Свойства чисел 0, 1, -1	☒
1.1.4	Пара вещественных чисел и угол	☒
1.1.2 Геометрическое представление комплексных чисел		
1.1.5	Комплексное число	☒
1.1.11	Комментарий (комплексные и вещественные числа)	☒
1.1.12	Отображение вещественных чисел на комплексные	☒
1.1.20	Комментарий (комплексные и вещественные числа)	☒
1.1.7 Общепринятые названия		
1.1.42	Поле вещественных чисел	☒
1.1.44	Связь полей между собой	☒
1.2 Множества и элементы		
1.2.1 Множества и операции над ними		
1.2.8	Известные множества чисел	☒
5 Расстояния, углы, объемы		
5.1 Скалярное произведение		
5.1.1 Евклидово и унитарное пространство		
5.1.2	Евклидово пространство	☐
5.1.4	Комментарий (сравнение пространств)	☐

Рисунок 3.17 – Форма прохождения курсов в электронно-образовательной среде обучения уровневой дифференциации

В третьей главе выпускной квалификационной работы спроектирована модель То-Ве, были определены концептуальная и логическая модели, сущности БД и связи между ними. Также была определена основная структура информационной системы, основные модули системы.

Глава 4 Экспериментальная оценка качества обучения с использованием информационно-образовательной среды с технологией уровневой дифференциации обучения

Экспериментальную оценку качества обучения, полученного по завершении обучения с использованием СДО, можно реализовать разными методами, в зависимости от того, какой параметр требуется оценить.

Методика измерения профессиональных знаний, умений человека дело не простое. При отсутствии измерений оценка традиционно строится на интуиции, на анализе документов (дипломов, характеристик и т.п.), результатах «испытательного срока», контрольных работ, рефератов и т.п. При создании хорошо управляемых технологий обучения необходимо создать систему тестов, позволяющих объективно оценивать различные стороны процесса обучения, его результаты. Анализ работ по диагностике В.С. Аванесова [2–4], А. Анастаси [8], Э.Р. Ахмеджанова [12], М.С. Бернштейна [21], К.М. Гуревича [59], Р. Римской [159] показал, что диагностика связана с разработкой и применением исследовательских инструментов – тестов. При разработке тестов придерживались определения, данного В.С. Аванесовым [2, С.42], что педагогический тест – это совокупность взаимосвязанных заданий возрастающей сложности, позволяющих надежно и валидно оценить знания и другие интересующие педагога характеристики личности». Очень существенно, что в определении основной упор делается на систему взаимосвязанных заданий возрастающей сложности.

Объективный, надежный и валидный тестовый контроль имеет важное образовательное и развивающее значение, способствуя всестороннему изучению программы, расширению, углублению и совершенствованию знаний, умений и навыков, развитию познавательных интересов студентов. Каждый студент вынужден активно, мотивированно участвовать в процессе тестового контроля, так как технология научно-организованного педагогического теста не допускает невнимательного отношения испытуемых к выполнению заданий.

Основными видами контроля, используемыми нами, являются:

- входной контроль;
- тестовый экспресс-контроль, организуемый на каждом лекционном занятии;
- текущий контроль (защита лабораторных работ);
- промежуточный контроль (контроль формируемой информационной компетентности).
- самоконтроль студентов.

Для доказательства эффективности использования в образовательном процессе разработанной СДО с технологией уровней дифференциации необходимо было провести эксперимент.

Целью эксперимента являлась апробация разработанной электронно-образовательной среды с технологией уровневой дифференциации для доказательства эффективности ее использования в образовательном процессе.

На основании цели эксперимента были выделены следующие задачи:

1. Выбор экспериментальной и контрольной групп.
2. Осуществление практического этапа внедрения и реализации СДО.
3. Анализ результатов апробации с использованием разработанного информационного обеспечения.

В качестве критериев оценки хорошо построенного образовательного процесса можно выделить следующие:

- результаты входного тестирования;
- промежуточные результаты, по окончании каждого модуля;
- результаты итогового тестирования.

В ходе проверки гипотезы проводилось сравнение результатов тестирования экспериментальной и контрольной групп. При подборе экспериментальной и контрольной групп при проведении эксперимента учитывались следующие требования: количество студентов в группах должно быть в идеале идентичным; уровень начальных знаний по дисциплине должен в идеале находиться в одном диапазоне.

СДО предполагает прохождение входного тестирования студента с целью выявления начального уровня обучения. В связи с этим был проведен анализ на то, как влияет выбор уровня на результаты обучения.

В качестве эксперимента были сформированы две группы испытуемых по 10 человек, в первой, для каждого студента был определен начальный уровень, а, во второй, начальный уровень был для всех одинаковый.

В таблице 4.1 представлена статистика результатов обучения студентов контрольной и экспериментальной групп.

Таблица 4.1 – Статистика результатов студентов по модулям

№ испытуемого	Результаты начального теста		Результаты 1 модуля		Результаты 2 модуля		Результаты 3 модуля		Результаты итогового теста	
	ЭГ	КГ	ЭГ	КГ	ЭГ	КГ	ЭГ	КГ	ЭГ	КГ
1	30	28	63	60	70	55	75	61	78	59
2	41	46	43	42	85	75	79	56	86	77
3	41	43	55	49	63	46	79	66	88	69
4	17	14	53	46	58	46	76	76	87	80
5	12	16	83	82	68	56	75	65	99	75
6	43	46	61	60	87	71	80	53	86	75
7	48	51	41	32	60	51	70	64	83	64
8	31	31	44	42	81	81	85	66	87	73
9	17	12	41	40	84	65	79	70	97	87
10	45	43	50	41	51	49	74	72	85	75

Данные по среднестатистическим результатам двух групп испытуемых отобразим на гистограмме (рисунок 4.1), на которой видно, что среднестатистические результаты экспериментальной группы выше результатов контрольной группы. Из таблицы 4.2 видно, что с каждым пройденным тестом результаты экспериментальной группы становятся лучше, чем у контрольной группы.

Таблица 4.2 – Среднестатистические результаты студентов по модулям

	Результаты начального теста	Результаты 1 модуля	Результаты 2 модуля	Результаты 3 модуля	Результаты итогового теста
Среднее значение экспериментальной группы	32,5	53,4	70,7	77,2	87,6
Среднее значение контрольной группы	33	49,4	59,5	64,9	73,4

По представленному графику на рисунке 4.1 можно заметить, что группа испытуемых, которая проходила обучение с определением начального уровня подготовки к курсу после прохождения тестирования имеет лучшую успеваемость, чем вторая группа испытуемых.

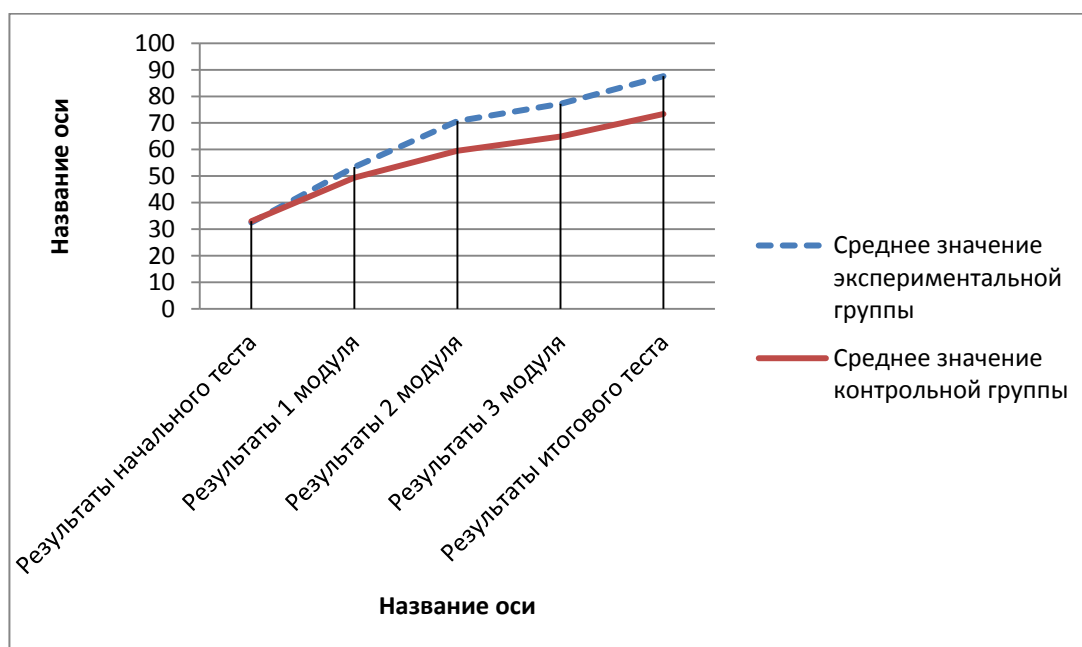


Рисунок 4.1 – График среднестатистических результатов двух групп

Таким образом, СДО можно считать отличным способом организации дистанционного обучения. Наблюдается хороший среднестатистический показатель успеваемости, в связи с выбором стартового уровня обучения, студенту легче начать обучение и разобраться с последующим материалом. Анализ эффективности обучения при помощи СДО приводится ниже с использованием непараметрического критерия Манна-Уитни.

Критерий предназначен для оценки различий между двумя выборками по уровню какого-либо признака, количественно измеренного. Он позволяет

выявлять различия между малыми выборками, когда $n_1, n_2 \geq 3$ или $n_1 = 2, n_2 \geq 5$. В каждой выборке должно быть не более 60 наблюдений.

Этот метод определяет, достаточно ли мала зона перекрещивающихся значений между двумя рядами. Положим, что первым рядом (выборкой, группой) мы называем тот ряд значений, в котором значения, по предварительной оценке, выше, а вторым рядом - тот, где они предположительно ниже.

Чем меньше область перекрещивающихся значений, тем более вероятно, что различия достоверны. Иногда эти различия называют различиями в расположении двух выборок.

Эмпирическое значение критерия U отражает то, насколько велика зона совпадения между рядами. Поэтому чем меньше $U_{\text{эмп}}$, тем более вероятно, что различия достоверны.

Гипотезы:

H_0 : Уровень признака в группе 2 не ниже уровня признака в группе 1.

H_1 : Уровень признака в группе 2 ниже уровня признака в группе 1.

Алгоритм расчета критерия Манна-Уитни:

1. Объединить все данные в единый ряд, пометив данные, принадлежащие разным выборкам.
2. Проранжировать значения, приписывая меньшему значению меньший ранг. Всего рангов получится $(n_1 + n_2)$.
3. Подсчитать сумму рангов отдельно для каждой выборки.
4. Определить большую из двух ранговых сумм.
5. Определить значение U по формуле:

$$U = n_1 * n_2 + \frac{n_x(n_x+1)}{2} - T_x, \quad (4.1)$$

где n_1 – объем выборки №1;

n_2 – объем выборки №2;

T_x – большая из двух ранговых сумм;

n_x – объем максимальной выборки: $n_x = \max(n_1, n_2)$.

6. Определить критические значения $U_{кр}$ по таблице. Если $U_{эмн} > U_{кр}(0,05)$. H_0 принимается. Если $U_{эмн} \leq U_{кр}(0,05)$ H_0 отвергается. Чем меньше значения U , тем достоверность различий выше.

Гипотеза H_0 о незначительности различий между выборками принимается, если $U_{кр} < U_{эмн}$. В противном случае H_0 отвергается и различие определяется как существенное, где $U_{кр}$ – критическая точка, которую находят по таблице Манна-Уитни.

Найдем критическую точку $U_{кр}$, по таблице находим $U_{кр}(0,05) = 45$. Так как $U_{кр} > U_{эмн}$ – принимаем альтернативную гипотезу H_1 ; различия в уровнях выборок можно считать существенными.

Ограничения применимости критерия:

1. В каждой из выборок должно быть не менее 3 значений признака. Допускается, чтобы в одной выборке было два значения, но во второй тогда не менее пяти.
2. В выборочных данных не должно быть совпадающих значений (все числа – разные) или таких совпадений должно быть очень мало.

U-критерий Манна-Уитни - статистический критерий, используемый для оценки различий между двумя независимыми выборками по уровню какого-либо признака, измеренного количественно. Позволяет выявлять различия в значении параметра между малыми выборками.

Для проверки одностороннего критерия определим критическое значение для $p = 0,05$ и уровня значимости 5%, которое равно (таблица критических значений U-критерия Манна-Уитни приведены в приложении Б). Так как вычисленное значение критерия меньше табличного, нулевая гипотеза отвергается на выбранном уровне значимости, и различия между выборками признаются статистически значимыми. Таким образом, вывод о существовании различий подтверждается с помощью данного непараметрического теста.

Применим данный критерий для разработанной СДО.

Выдвигаются две гипотезы.

H_0 : Уровень среднестатистической оценки в контрольной группе не ниже уровня среднестатистической оценки в экспериментальной группе.

H_1 : Уровень среднестатистической оценки в экспериментальной группе не ниже уровня среднестатистической оценки в контрольной группе.

Для этого отобразим результаты обучения в таблице 4.3.

Таблица 4.3 – Результаты итогового теста по пятибалльной шкале

№ испытуемого	Результаты итогового теста	
	Экспериментальная группа	Контрольная группа
1	4	3
2	5	4
3	5	4
4	5	5
5	5	4
6	5	4
7	5	4
8	5	4
9	5	5
10	5	4

Найдем критическую точку $U_{кр}$, по таблице находим $U_{кр} (0.05) = 27$. Так как $U_{кр} > U_{эмп}$ - принимаем альтернативную гипотезу H_1 ; различия в уровнях выборок можно считать существенными.

Произведя вычисления, $U_{эмп} = 21,5$. Что говорит о том, что мы принимает гипотезу H_1 .

На основании статистической оценки эксперимента можно констатировать, что использование СДО способствует повышению качества и эффективности обучения.

Таким образом, в 4 главе была произведена поэтапная оценка СДО. На первом этапе проводился анализ образовательного процесса с использованием СДО по среднестатистическим результатам двух групп испытуемых.

На втором этапе была проведена обработка результатов педагогического эксперимента, направленного на апробацию созданного курса с использованием СДО, которая позволила оценить педагогический эффект, получаемый после обучения слушателей с использованием СДО.

На основании полученных с помощью СДО данных сотрудники ВУЗа способны корректировать выявленные недостатки и тем самым улучшать показатели обучения, а, следовательно, качество получаемых в ВУЗе знаний.

Таким образом, практическая реализация СДО подтвердила правильность разработанной теоретической модели и может быть внедрена в ВУЗы в качестве системы, помогающей упростить образовательный процесс и улучшить качество образования.

Заключение

В ходе диссертационной работы была проанализирована литература на тему технологии уровневой дифференциации в обучении, использования информационных технологий в образовании.

Таким образом, проанализировав исследования Грузик Н.П., Селевко Г.К. была смоделирована информационная система, организующая образовательный процесс в среде обучения уровневой дифференциации.

В соответствии с этим были сформулированы цели и задачи для будущей системы, а именно было принято решение о создании «Технологии уровневой дифференциации обучения в построении информационно-образовательной среды вуза».

Первая и вторая глава диссертационной работы представляет собой теоретическую доказательную базу для подтверждения сформулированной гипотезы. Согласно собранной в процессе анализа предметной области информации была подтверждена актуальность темы научной работы, в рамках которой была разработана информационно-образовательная среда. Были определены текущая ситуация в сфере образования.

Была сформулирована цель диссертационной работы - разработка информационно-образовательной среды вуза с технологией уровневой дифференциации обучения.

Было дано определение уровневой дифференциации, разработана технология уровневой дифференциации для электронно-образовательной среды.

При определении функциональных возможностей ИС и выборе средств построения была определена структура информационной системы. Были определены основные требования к системе, позволяющие обеспечить ее полное функционирование при решении поставленных задач, также были разработаны основные модули системы.

Перед тем, как приступить к реализации системы необходимо было как можно более точно спроектировать базу данных во избежание ошибок и избыточности данных в процессе реализации. Для проектирования базы данных

использовалась ER-модель, а программный продукт ERWIN позволил перенести спроектированную базу данных в физическую структуру данных.

Реализация системы происходила на языке PHP с использованием JavaScript и языка запросов SQL. Информационное обеспечение представляет собой WEB-интерфейс. Оно включает в себя сайт-визитку, личный кабинет студента, преподавателя и администратора. Каждый из пользователей обладает определенными доступными функциями. Например, студент может записываться на курс, администратор конфигурировать расписание, преподаватель проводить занятия.

Но для того, чтобы определить насколько информационное обеспечения соответствуют стандартам и насколько удобно оно в использовании, необходимо было провести экспериментальную апробацию.

В процессе экспериментальной апробации дорабатывались некоторые детали, а впоследствии проводился эксперимент на то, как информационное обеспечение повлияет на образовательный процесс.

Эксперимент доказал, что использование информационного обеспечения приводит к улучшению показателей качества обучения.

В результате проделанной диссертационной работы была подтверждена выдвинутая гипотеза:

Является предположение о зависимости эффективности дистанционного обучения от применения технологии уровневой дифференциации.

В результате проектирования информационной системы была разработана информационно-образовательная среда вуза с технологией уровневой дифференциации обучения, которая позволит облегчить работу преподавателя. Разработанная информационно-образовательная среда позволяет автоматизировать процесс сбора, обработки и хранения информации от преподавателей и студентов. Основным преимуществом разработанной информационно-образовательной среды является возможность выбора уровня обучения студенту.

Информационно-образовательная среда нуждается в доработке, но уже выполняет основные функции. В перспективе предусматривается решение ряда дополнительных задач, таких как: введение вебинаров в расписание, увеличение форм отчетности. Это позволит расширить не только возможности, но и повысить качество и легкость обучения.

Список используемой литературы

1. Brooks F. No Silver Bullet. Information Proceeding of the IFIP 10th World Computing Conference, 1986, p. 1069-1076
2. Czarnecki K., Eisenecker, U.W Generative programming: Methods, Tools, and Applications Addison-Wesley, 2010. - 832 p.
3. Havey M. Essential Business Process Modeling: O'REILLY, 2015, 332 p
4. Journal of Visual Languages and Computing. 15 (2014), Preface, P. 207-209
5. Learning Management System Modular Object-Oriented Dynamic Learning Environment [Электронный ресурс]. – Режим доступа: www.moodle.org. (дата обращения: 15.09.2014).
6. Sommerville I. Software Engineering. 6th Edition: Addison-Wesley, 2011, 693 p
7. Аалдерс, Р. ИТ аутсорсинг. Практическое руководство [Текст]. — М.: Альпина Бизнес Букс, 2009. — 300 с.
8. Акимова, И.В. Использование информационных технологий в образовании: учебное пособие для студентов и учителей [Текст] / И. В. Акимова, И. А. Баландин. - Пенза, 2010.
9. Акофф, Р. О целеустремленных системах [Текст] / Р. Акофф, Ф. Эмери. М. – 2008. -С.248.
10. Андреева, И.А. Теория и практика использования информационных ресурсов (на примере малого бизнеса) [Текст]/ ИРР. 2009. - №8(63).
11. Апатова, Н.В. Информационные технологии в школьном образовании[Текст]. - М., 2010.
12. Апатова Н.В. Информационные технологии в дошкольном образовании[Текст]. — М., 2009. — 127 с.
13. Бенкен, Е.С. PHP, MySQL, XML программирование для Интернета [Текст] / Е. С.Бенкен. - 3-е изд. - Санкт-Петербург : БХВ-Петербург, 2011. - 288 с.
14. Боженюк, А. В. Интеллектуальные интернет-технологии : учеб.для вузов / А. В. Боженюк, Э. М. Котов, А. А. Целых. - Гриф УМО. - Ростов н/Д: Феникс, 2009. – 381 с.

15. Будилов, В.А. Основы программирования для Интернета : Стат. HTML - станицы. CGI - программирование с применением яз. PHP. Разраб. динам.страниц с помощью JavaScript. Технология создания актив.серве. страниц ASP / Вадим Алексеевич Будилов. - СПб. : БХВ-Петербург, 2009 (Акад. тип.Наука РАН). - 716 с.
16. Вендров, А.М. Проектирование программного обеспечения экономических информационных систем: Учебник / А.М. Вендров. – М.: Финансы и статистика, 2011.-544 с.
17. Гаврилов, А.А. Методическое пособие по работе с ЭОР в среде Moodle/ А.А. Гаврилов. – Псков, 2010. – 38с.
18. Гагарина, Л.Г. Разработка и эксплуатация автоматизированных информационных систем М.: ИД «ФОРУМ», ИНФРА-М, 2009. 384с.
19. Гагарина, Л.Г. Технология разработки программного обеспечения : учеб.пособие для студ. вузов, обуч. по напр. 230100 - "Информатика и вычислительная техника" / Л. Г. Гагарина, Б. Д. Виснадул ; под ред. Л. Г. Гагариной. - Гриф УМО. - М. : ФОРУМ - ИНФРА-М, 2010. - 399 с.
20. Гвоздева, Т.В. Основы построения автоматизированных информационных систем. М.: ФОРУМ, 2009. 320с.
21. Гвоздева, Т.В. Проектирование информационных систем / Т. В. Гвоздева, А. Б. Баллод. – М.: Феникс, 2010. – 508 с.
22. Грекул, В.И. Проектирование информационных систем : учеб.пособие / В.И. Грекул, Г.Н. Денищенко, Н.Л. Коровкина. - 2-е изд., испр. - М. : Интернет-Ун-т Информ. Технологий : БИНОМ. Лаб. знаний, 2011. - 299 с.
23. Грузик, Н.П. Учить учиться: учеб. для вузов / Н.П. Грузик - М.: Феникс, 1981. - 400 с.
24. Гусятников, В.Н. Стандартизация и разработка программных систем : учебное пособие для студентов высших учебных заведений, обучающихся по специальности 080801 "Прикладная информатика" и другим экономическим специальностям / В. Н. Гусятников, А. И. Безруков. - Москва : Финансы и статистика : ИНФРА-М, 2010. - 284, [2] с.

25. Дейк, Т.В. К определению дискурса. [Электронный ресурс]. – Режим доступа: <http://psyberlink.flogiston.ru/internet/bits/vandijk2.htm>(дата обращения: 02.12.2014).
26. Дейт, К. Дж. Введение в системы баз данных 8-е изд./ Пер. с англ. Птицын К. А. М.: Вильямс. 2010. 328с.
27. Дроздова, Т.В. Научный текст и проблемы его понимания [Текст] / Т.В. Дроздова, – М.: МГЛУ, 2007. – 390 с.
28. Заботина, Н.Н. Проектирование информационных систем [Текст] : учебное пособие для студентов высших учебных заведений, обучающихся по специальности 080801 "Прикладная информатика (по областям)" и другим экономическим специальностям / Н. Н. Заботина. - Москва: ИНФРА-М, 2011. - 329, [1] с. : ил.; 22 см. - (Высшее образование).
29. Информация и образование [Электронный ресурс]. – Режим доступа: <http://hotuser.ru/>(дата обращения: 15.09.2014).
30. Карпова, Т.С. Базы данных: модели, разработка реализация. СПб.: Питер, 2009. 304с.
31. Колисниченко, Д.Н. Самоучитель PHP 5. 2-е изд. СПб.: Наука и Техника, 2008. 576с.
32. Колисниченко, Д.Н. PHP 5/6 и MySQL 6. Разработка Web-приложений / Денис Колесниченко. - 2-е изд. - Санкт-Петербург : БХВ-Петербург, 2010. - 540 с.
33. Комолова, Н.В. HTML, XHTML и CSS [Текст] / Н. Комолова, Е. Яковлева. - Санкт-Петербург [и др.] : Питер, 2012. - 300 с.
34. Коннолли, Т. Б. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. / Пер. с англ. Имамутдинова Р. Г., Птицын К. А. М.: Вильямс. 2010. 440с.
35. Кузин, А.В. Базы данных: учеб. пособие для вузов / А.В. Кузин, С.В. Левонисова. – 3-е изд., стер.; Гриф УМО. – М.: Академия, 2011. – 315 с.
36. Кузнецов, М.И. PHP. Практика создания Web-сайтов: учеб. пособие / М. И. Кузнецов, И. М. Симдянов И.В. – СПб. – БХВ-Петербург, 2009. – 1264 с.

37. Кузнецов, С.Д. Основы баз данных. 2-е изд. М.: «Интернет-университет информационных технологий - ИНТУИТ.ру», 2009. 484с.
38. Кузнецов, М.И. Самоучитель по РНР 5 [Текст] / М.И. Кузнецов, И.В. Симдянов. – СПб.: БХВ-Петербург, 2010. – 560с.
39. Ларина, Ю.А. Основы объектно ориентированного моделирования с использованием языка UML [Текст]: учебное пособие / Ю. А. Ларина ; М-во образования и науки Российской Федерации, Федеральное агентство по образованию, Ярославский гос. ун-т им. П. Г. Демидова. - Ярославль : Ярославский гос. ун-т им. П. Г. Демидова, 2010. - 150 с.
40. Леоненков, А.В. Самоучитель UML / Александр Леоненков. - 6-е изд. - СПб. : БХВ-Петербург, 2010 (Акад. тип.Наука РАН). - 427 с. : ил.; 23 см. - (Самоучитель).
41. Логанов, С.В. Объектно-ориентированное проектирование. Язык UML и основы объектно-ориентированного программирования ИС [Текст] : учебное пособие для студентов, обучающихся по направлению "Информационные системы" / С. В. Логанов ; М-во образования и науки Российской Федерации, Гос. образовательное учреждение высш. проф. образования Нижегородский гос. технический ун-т им. Р. Е. Алексеева. - Нижний Новгород : НГТУ им. Р. Е. Алексеева, 2010. - 140, [1] с.
42. Ломов, А.Ю. Apache, Perl, MySQL: практика создания динамических сайтов. Самоучитель. СПб: БХВ-Петербург, 2007. 368 с.
43. Мазуркевич, А.Е. РНР: настольная книга программиста СПб: BHV-Санкт-Петербург 2010 г. 479 стр.
44. Малков, О.Б. Работа с СУБД MYSQL: учебное пособие / О. Б. Малков, М. В. Девятерикова. – Омск: Изд-во ОмГТУ, 2010. - 84 с.
45. Матросов, А.В. HTML 4.0 (в подлиннике) [Текст] / А.В. Матросов, А.О. Сергеев. Санкт-Петербург: Издательство BHV, 2008. – 668 с.
46. Методология функционального моделирования IDEF0. Руководящий документ. – М.: ИПК Издательство стандартов, 2000. – 75 с.

47. Науменко, Е.Н., Исследование технологий дистанционного обучения при подготовке специалистов в области автоматизации и управления [Текст] / Е. Н. Науменко // Молодой ученый. — 2013. — №6. — С. 98.
48. Национальный Открытый Университет «ИНТУИТ» [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru>(дата обращения: 15.09.2014).
49. Пауэлл, Т. Полный справочник по JavaScript. М.: Издательский дом «Вильямс», 2009. 960 с.
50. Портал по PHP, MySQL и другим веб-технологиям [Электронный ресурс]. – Режим доступа: <http://www.php.su>(дата обращения: 11.12.2014).
51. Проектирование информационных систем [Электронный ресурс]. – Режим доступа: http://citforum.ru/database/oraclepr/oraclepr_02.shtml (дата обращения: 25.11.2014).
52. Прохоренок, Н.А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера. – СПб.:БХВ-Петербург, 2010.
53. Развитие науки и образования в современном мире: Сборник научных трудов по материалам Международной научно-практической конференции 30 сентября 2014 г. В 7 частях. Часть VI. [Текст] - М.: «АР-Консалт», 2014 г.- 151 с.
54. Селевко, Г.К. Энциклопедия образовательных технологий / Г.К. Селевко, - М.: Народное образование, 2005. -556 с.
55. Система дистанционного образования Moodle [Электронный ресурс]. – Режим доступа: <https://moodle.org/> (дата обращения: 14.12.2014).
56. Системы дистанционного образования в Мире [Электронный ресурс]. – Режим доступа<http://www.dlearn.org/directory/> (дата обращения: 12.12.2014).
57. Советов, Б.Я. Базы данных: теория и практика : учебник для студентов вузов, обучающихся по направлениям "Информатика и вычислительная техника" / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовский. - 2-е изд. - Москва: Юрайт, 2012. - 462, [1] с.
58. Советов, Б.Я. Базы данных: теория и практика : учебник для студентов вузов, обучающихся по направлениям "Информатика и вычислительная

техника" и "Информационные системы" / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовский. - Москва: Юрайт, 2012. - 462с.

59. Сухомлин, В.А. Современные информационные технологии и ИТ-образование / В.А. Сухомлин – М., 2012. – 1050с.
60. Сухомлин, В.А. Современные информационные технологии и ИТ-образование / В.А. Сухомлин – М., 2013. – 829с.
61. Тольяттинский государственный университет [Электронный ресурс]. – Режим доступа: <http://site.tltsu.ru/> (дата обращения: 02.12.2014).
62. Черников, Б.В. Информационные технологии управления: учеб. для вузов / Б.В. Черников. – Гриф УМО. – М.: ФОРУМ, 2012. – 351с.

ПРИЛОЖЕНИЕ А

Исходный код системы

Index.php

```

<?session_start()?>
<html>

<head>
    <meta charset="utf-8">
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <link rel="stylesheet" href="css/style.css">
    <script src="js/bootstrap.min.js"></script>
    <script src="js/jquery-1.11.0.min"></script>
    <style>input::-webkit-input-placeholder {color:#555}input:-moz-placeholder {color:#555;}</style>
    <title>Электронно-образовательная среда</title>
</head>
<body background="img/bg.gif">
<div class="container" style="margin-top: 50px;">
    <div class="row">
        <div class="span6" id = "login">
            <h1 style="margin-left: 135px;padding:5px">Вход в систему</h1>
            <form class="form-horizontal" action="auth.php" method="POST">
            <div class="control-group">
                <div class="controls">
                    <select name="fio" id="username">
                    <option value=" disabled selected style='display:none;'>Выберите фамилию</option>
                    <?
$adb = mysql_connect("localhost","root","");
mysql_select_db("info", $adb);
$qqr_result = mysql_query("select * from info.user");
while($user = mysql_fetch_array($qqr_result)){ ?>
                <option value="<?echo $user[1]?>"><?echo
$user[1]?></option><?}></select></div>
            </div>
            <div class="control-group">
                <div class="controls">
                    <input type="password" name="password" id="password" placeholder="Введите пароль">
                </div>
            </div>
            <div class="control-group">
                <div class="controls">
                    <input class="btn btn-success" type="submit" id="enter" name="auth" value = "Войти">
                </div>
            </div></form>
        </div></div></div>
</body></html>

```

Auth.php

```

<?php session_start();
$fio = $_POST['fio'];
//$password = "";
$adb = mysql_connect("localhost","root","");
mysql_select_db("info", $adb);
$qqr_result = mysql_query("select * from info.user where user.fio = '$fio'");
$user = mysql_fetch_array($qqr_result);
if(isset($_POST['auth'])){
    if($user['fio'] == $_POST['fio']){
        if($user['password'] == $_POST['password']){

```

```

$_SESSION['id'] = $user['id'];
$_SESSION['fio'] = $user['fio'];
$_SESSION['job'] = $user['job'];
header("Location: main.php");
}else $error;
} else $error;
if(isset($error))
echo "<h2>Ошибка! Не правильно введен пароль</h2><br>";
echo "<a href='index.php'>Попробуйте ещё раз!</a>";
}??>

```

Main.php

```

<?session_start();?>
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<link rel="stylesheet" href="../css/bootstrap.min.css">
<link rel="stylesheet" href="css/style.css">
<script src="../js/jquery-1.11.0.min.js"></script>
<script src="../js/bootstrap.min.js" type="text/javascript"></script>
<script type="text/javascript">
function showOrHide(show, hide) {
show = document.getElementById(show);
hide = document.getElementById(hide);
if (show.checked) hide.style.display = "block";
else hide.style.display = "none";
}
</script>
<title>Дипломная работа</title>
</head>
<body>
<div class="hi">Вы зашли под пользователем: <?echo $_SESSION[fio]?> <span class="label label-
success"><a href="del.php">Выход</a></span></div>
<div class="navbar navbar-inverse">
<div class="container">
<ul class="nav nav-tabs">
<li class="active"><a href="main.php">Задачи</li></a>
<li><a href="messages.php">Сообщения</li></a>
<li><a href="kalen.php">Календарь событий</li></a>
<li><a href="reports.php">Отчеты</li></a>
<li><a href="#">Расписание</li></a>
<li><a href="client.php">Клиенты</li></a>
</ul>
</div>
</div>
<div class="content">
<div class="tabbable tabs-left">
<ul class="nav nav-tabs">
<li>Мои задачи</li>
<li><a href="#all" data-toggle="tab">Все</a></li>
<li><a href="#new" data-toggle="tab">Новые</a></li>
<li><a href="#operating" data-toggle="tab">Выполняемые</a></li>
<li><a href="#complete" data-toggle="tab">Завершённые</a></li>
<?if ($_SESSION[job]=="Администратор")?><li><a href="#create" data-toggle="tab">Добавить
задачу</a></li><?>
<?if ($_SESSION[job]=="Администратор")?><li><a href="#admin" data-
toggle="tab">Администрирование</a></li><?>

```

```

</ul>
<div class="tab-content">
<div class="tab-pane active" id="hello">
<p>Выберите нужный вам раздел в списке слева.</p>
</div>
<div class="tab-pane" id="all">
<table class="table table-hover">
<thead><tr><th>#</th><th>Ом</th><th>Задача</th><th>Дамы</th><th>Состояние</th></tr>
</thead>
<tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$c=0;
$sql = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_otp`, `user`.`fio`, `zadacha`.`id_pol`,
`zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM `zadacha`,`user`
WHERE `zadacha`.`id_otp`=`user`.`id` and `zadacha`.`id_pol`=`$id` order by `zadacha`.`id_zadacha`
desc");
while($task = mysql_fetch_array($sql)){ $c++;?>
<?if($task[7]==0){?><tr class="error"><td><?echo $task[0]?></td><td><?echo
$task[2]?></td><td><?echo $task[4]?></td><td><?echo $task[5]?></td><td>He
прочитано</td></tr><?>
if(($task[7]!=0) and ($task[6]==0)){?><tr class="warning"><td><?echo $task[0]?></td><td><?echo
$task[2]?></td><td><?echo $task[4]?></td><td><?echo
$task[5]?></td><td>Выполняется</td></tr><?>
if(($task[7]!=0) and ($task[6]!=0)){?><tr class="success"><td><?echo $task[0]?></td><td><?echo
$task[2]?></td><td><?echo $task[4]?></td><td><?echo
$task[5]?></td><td>Готово</td></tr><?><?>
if($c==0){?><tr><td style="text-align:center;padding:10px" colspan="5">У вас нет
задач</td></tr><?><?></tbody>
</table>
</div>
<div class="tab-pane" id="new">
<form method="post" action="edit_task.php">
<table class="table table-hover"><thead>
<tr><th>#</th><th>Ом</th><th>Задача</th><th>Дамы</th><th>Состояние</th></tr>
</thead><tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$i=0;
$sql_new = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_otp`, `user`.`fio`,
`zadacha`.`id_pol`, `zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM
`zadacha`,`user` WHERE `zadacha`.`id_otp`=`user`.`id` and `zadacha`.`prochten`='0' and
`zadacha`.`id_pol`=`$id` order by `zadacha`.`id_zadacha` desc");
while($task_new = mysql_fetch_array($sql_new)){ $i++;?>
<tr class="error"><td><?echo $task_new[0]?></td><td><?echo $task_new[2]?></td><td><?echo
$task_new[4]?></td><td><?echo $task_new[5]?></td><td>He прочитано</td><td><input
type="checkbox" class="task" name="check<?echo $i?>"></td></tr>
<input type="hidden" name="id_zadacha<?echo $i?>" value="<?echo $task_new[0]?>"> <?>
if($i==0){?><tr><td style="text-align:center;padding:10px" colspan="5">У вас нет новых
задач</td></tr><?><?>
</tbody></table>
<?if($i!=0){?><input class="btn" type="submit" name="read_task" value="Принять"><?>
</form>
</div>
<div class="tab-pane" id="operating">

```

```

<form method="post" action="edit_task.php">
<table class="table table-hover">
<thead>
<tr><th>#</th><th>Ом</th><th>Задача</th><th>Дамы</th><th>Состояние</th></tr>
</thead><tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$j=0;
$sql_op = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_otp`, `user`.`fio`,
`zadacha`.`id_pol`, `zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM
`zadacha`,`user` WHERE `zadacha`.`id_otp`=`user`.`id` and `zadacha`.`prochten`='1' and
`zadacha`.`vipol`='0' and `zadacha`.`id_pol`='$id' order by `zadacha`.`id_zadacha` desc");
while($task_op = mysql_fetch_array($sql_op)){ $j++;?>
<tr class="warning"><td><?echo $task_op[0]?></td><td><?echo $task_op[2]?></td><td><?echo
$task_op[4]?></td><td><?echo $task_op[5]?></td><td>Выполняется</td><td><input
type="checkbox" class="task" name="check"<?echo $j?>"></td></tr>
<input type="hidden" name="id_zadacha"<?echo $j?>" value="<?echo $task_op[0]?>"> <?}
if($j==0){?><tr><td style="text-align:center;padding:10px" colspan="5">У вас нет выполняемых
задач</td></tr><?}
?>
</tbody></table>
<?if($j!=0){?><input class="btn" type="submit" name="complete_task" value="Завершить"><?}?>
</form>
</div>
<div class="tab-pane" id="complete">
<table class="table table-hover">
<thead>
<tr><th>#</th><th>Ом</th><th>Задача</th><th>Дамы</th><th>Состояние</th></tr>
</thead><tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$k=0;
$sql_com = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_otp`, `user`.`fio`,
`zadacha`.`id_pol`, `zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM
`zadacha`,`user` WHERE `zadacha`.`id_otp`=`user`.`id` and `zadacha`.`prochten`='1' and
`zadacha`.`vipol`='1' and `zadacha`.`id_pol`='$id' order by `zadacha`.`id_zadacha` desc");
while($task_com = mysql_fetch_array($sql_com)){ $k++;?>
<tr class="success"><td><?echo $task_com[0]?></td><td><?echo $task_com[2]?></td><td><?echo
$task_com[4]?></td><td><?echo $task_com[5]?></td><td>Готово</td></tr>
<?}
if($k==0){?><tr><td style="text-align:center;padding:10px" colspan="5">У вас нет завершенных
задач</td></tr><?}
?>
</tbody>
</table>
</div>
<?if($_SESSION[job]=="Администратор"){?>
<div class="tab-pane" id="create">
<form method="POST" action="edit_task.php">
<select name="to_user" class="span4">
<option value="" disabled selected style='display:none;'>Выберите сотрудника</option>
<?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];

```

```

$qr_result = mysql_query("select `id`,`fio` from `user` where `id`!='$id'");
while($user = mysql_fetch_array($qr_result)){?>
<option value="<?echo $user[id]?>"><?echo $user[fio]?></option><?}?>
</select>
<select name="client" class="span4">
<option value=" disabled selected style='display:none;'>Выберите клиента</option>
<?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$qr_result1 = mysql_query("select `fio` from `client`");
while($client = mysql_fetch_array($qr_result1)){?>
<option value="<?echo $client[fio]?>"><?echo $client[fio]?></option><?}?>
</select><br>
<textarea class="span8" name="name" rows="7" placeholder="Введите описание
задачи"></textarea><br>
<label class="checkbox">
<input type="checkbox" name="event" id="show" onchange='showOrHide("show", "hide");'>Добавить
задачу в календарь событий сотрудника
</label><br>
<div id='hide' style='display:none'>
<label class="checkbox">
Дата события <input type="date" class="span3" name="date" placeholder="Введите дату события">
</label><br>
</div>
<br>
<input type="submit" class="span4 btn btn-large btn-success" style="margin-left:160px"
name="create_task" value="Добавить">
</form>
</div>
<div class="tab-pane" id="admin">
<form method="post" action="edit_task.php">
<table class="table table-hover">
<thead>
<tr><th>#</th><th>Ом</th><th>Задача</th><th>Дата</th><th>Состояние</th></tr>
</thead>
<tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$a=0;
$sql_admin = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_pol`, `user`.`fio`,
`zadacha`.`id_otp`, `zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM
`zadacha`,`user` WHERE `zadacha`.`id_pol`=`user`.`id` and `zadacha`.`id_otp`='$id' order by
`zadacha`.`id_zadacha` desc");
while($task_admin = mysql_fetch_array($sql_admin)){ $a++;?>
<?if($task_admin[7]==0){?><tr class="error"><td><?echo $task_admin[0]?></td><td><?echo
$task_admin[2]?></td><td><?echo $task_admin[4]?></td><td><?echo
$task_admin[5]?></td><td>Не прочитано</td></tr><?}
if(($task_admin[7]!=0) and ($task_admin[6]==0)){?><tr class="warning"><td><?echo
$task_admin[0]?></td><td><?echo $task_admin[2]?></td><td><?echo
$task_admin[4]?></td><td><?echo $task_admin[5]?></td><td>Выполняется</td></tr><?}
if(($task_admin[7]!=0) and ($task_admin[6]!=0)){?><tr class="success"><td><?echo
$task_admin[0]?></td><td><?echo $task_admin[2]?></td><td><?echo
$task_admin[4]?></td><td><?echo $task_admin[5]?></td><td>Готово</td><td><input
type="checkbox" name="check<?echo $a?>"></td></tr>
<input type="hidden" name="id_zadacha<?echo $a?>" value="<?echo $task_admin[0]?>"><?}?><?}

```

```

if($a==0){?><tr><td style="text-align:center;padding:10px" colspan="5">Вы не добавляли
задачи</td></tr><?> ?>
</tbody></table>
<?if($a!=0){?><input class="btn" type="submit" name="admin_task" value="Отправить на
доработку"><?>?>
<input type="submit" class="btn" name="del_all_task" value="Удалить все задачи в системе">
</form></div><?>?>
</div> </div></div></body></html>

```

Messages.php

```

<?session_start();
include("data_convert.php");?>
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<link rel="stylesheet" href="../css/bootstrap.min.css">
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/messages.css">
<script src="../js/jquery-1.11.0.min.js"></script>
<script src="../js/bootstrap.min.js" type="text/javascript"></script>
<title>Сообщения</title>
</head>
<body>
<div class="hi">Вы зашли под пользователем: <?echo $_SESSION[fio]?> <span class="label label-
success"><a href="del.php">Выход</a></span></div>
<div class="navbar navbar-inverse">
<div class="container">
<ul class="nav nav-tabs">
<li><a href="main.php">Задачи</li></a>
<li class="active"><a href="messages.php">Сообщения</li></a>
<li><a href="kalen.php">Календарь событий</a></li>
<li><a href="reports.php">Отчеты</a></li>
<li><a href="#">Расписание</a></li>
<li><a href="client.php">Клиенты</a></li>
</ul>
</div></div><div class="content">
<div class="tabbable tabs-left">
<ul class="nav nav-tabs">
<li>Мои сообщения</li>
<li><a href="#send_msg" data-toggle="tab">Написать</a></li>
<li><a href="#in_msg" data-toggle="tab">Входящие</a></li>
<li><a href="#out_msg" data-toggle="tab">Отправленные</a></li></ul>
<div class="tab-content">
<div class="tab-pane active" id="hello">
<p>Выберите нужный вам раздел в списке слева.</p></div>
<div class="tab-pane" id="send_msg">
<form method="POST" action="send_message.php">
<select name="to_user" class="span6">
<option value="" disabled selected style='display:none;'>Выберите получателя</option><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$q_r_result = mysql_query("select `id`,`fio` from `user` where `id`!='$id'");
while($user = mysql_fetch_array($q_r_result)){?>
<option value="<?echo $user[id]?>"><?echo $user[1]?></option><?>?>
</select><br>

```

```

<input class="span6" name="title" size="16" type="text" placeholder="Введите заголовок
сообщения"><br>
<textarea class="span6" name="message" rows="7" placeholder="Введите текст
сообщения"></textarea><br>
<input type="submit" class="span4 btn btn-large btn-success" style="margin-left:85px" name="send_msg"
value="Отправить">
</form>
</div>
<div class="tab-pane" id="in_msg">
<table class="table">
<thead><tr>
<th>#</th><th>От кого</th><th>Тема</th><th>Дата</th></tr>
</thead>
<tbody>
<form method="post" action="send_message.php"><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$i=0; // используется для проверки.если она будет равна 0, значит нету сообщений
$sql = mysql_query("SELECT `messages`.`id`,`messages`.`from_user`,`user`.`fio`,`messages`.`title`,
`messages`.`date`,`messages`.`message`,`messages`.`is_read`,`messages`.`del_fr_user`,
`messages`.`del_to_user` FROM `messages`,`user` WHERE `user`.`id`=`messages`.`from_user` and
`messages`.`del_to_user`='0' and `messages`.`to_user`='$id' order by `messages`.`id` desc");
while($in = mysql_fetch_array($sql)){ $i++; if ($in[6]=="0"){?>
<tr style='background: #e2e2e2; cursor:pointer'><td><input type="checkbox" name="check<?echo
$in[0]?>"></td><td><a href="#"<?echo $in[0]?>" data-toggle="modal"><?echo
$in[2]?></a></td><td><a href="#"<?echo $in[0]?>" data-toggle="modal"><?echo
$in[3]?></a></td><td><a href="#"<?echo $in[0]?>" data-toggle="modal"><?echo
data_convert($in[4],1,1,0)?></a></td></tr><?>
else {?><tr style='cursor:pointer'><td><input type="checkbox" name="check<?echo
$in[0]?>"></td><td><a href="#"<?echo $in[0]?>" data-toggle="modal"><?echo
$in[2]?></a></td><td><a href="#"<?echo $in[0]?>" data-toggle="modal"><?echo
$in[3]?></a></td><td><a href="#"<?echo $in[0]?>" data-toggle="modal"><?echo
data_convert($in[4],1,1,0)?></a></td></tr><?>
<!-- Модальное окно просмотра входящих сообщения -->
<div class="modal hide fade" id="<?echo $in[0]?>" tabindex="-1" role="dialog">
<div class="modal-header"><button class="close" type="button" data-
dismiss="modal">×</button></div>
<h4 class="text-muted" id="myModalLabel">От: <?echo $in[2]?></h4>
</div><div class="modal-body">
<?echo $in[5]?></div>
<div class="modal-footer">
<input type="hidden" name="msg_id<?echo $i?>" value="<?echo $in[0]?>">
<input type="hidden" name="kol" value="<?echo $i?>">
<button class="btn" data-dismiss="modal">Закрыть</button>
</div></div>
<!-- Конец модального окна просмотра входящих сообщения -->
<?if($i==0){echo "<tr><td colspan='4' style='text-align: center'>Список входящих сообщений
нуст</td></tr>";}?></tbody>
</table>
<?if($i!=0){?><input type="submit" class="btn" name="del_in_msg" value="Удалить выделенные
сообщения">
<input type="submit" name="message_is_read" class="btn" value="Отметить новые сообщения
прочтенными"><?>
</form></div>
<div class="tab-pane" id="out_msg">
<table class="table">

```

```

<thead><tr>
<th>#</th><th>Комы</th><th>Тема</th><th>Дата</th>
</tr></thead><tbody>
<form method="post" action="send_message.php"><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$j=0;
$sql = mysql_query("SELECT `messages`.`id`,`messages`.`from_user`,`user`.`fio`,`messages`.`title`,
`messages`.`date`,`messages`.`message`,`messages`.`is_read`,`messages`.`del_fr_user`,
`messages`.`del_to_user` FROM `messages`,`user` WHERE `user`.`id`=`messages`.`to_user` and
`messages`.`del_fr_user`='0' and `messages`.`from_user`='$id' order by `messages`.`id` desc");
while($out = mysql_fetch_array($sql)){ $j++;?>
<tr style='cursor:pointer'><td><input type="checkbox" name="check<?echo $out[0]?>"></td><td><a
href="#<?echo $out[0]?>" data-toggle="modal"><?echo $out[2]?></a></td><td><a href="#<?echo
$out[0]?>" data-toggle="modal"><?echo $out[3]?></a></td><td><a href="#<?echo $out[0]?>" data-
toggle="modal"><?echo data_convert($out[4],1,1,0)?></a></td></tr>
<!-- Модальное окно просмотра отправленных сообщения -->
<div class="modal hide fade" id="<?echo $out[0]?>" tabindex="-1" role="dialog">
<div class="modal-header"><button class="close" type="button" data-
dismiss="modal">×</button></div>
<h4 class="text-muted" id="myModalLabel">Комы: <?echo $out[2]?></h4>
</div>
<div class="modal-body">
<?echo $out[5]?>
</div>
<div class="modal-footer">
<button class="btn" data-dismiss="modal">Закрыть</button>
</div></div>
<!-- Конец модального окна просмотра отправленных сообщения -->
<?if($j==0){echo "<tr><td colspan='4' style='text-align: center'>Список отправленных сообщений
нечт</td></tr>";?>
</tbody></table>
<?if($j!=0){?><input type="submit" class="btn" name="del_out_msg" value="Удалить выделенные
сообщения"><?}>
</form></div>
</div></div></div>
</body>
</html>

```

Kalen.php

```

<?session_start();
if ($_SESSION['fio']==") {header("Location: ../index.php");}
?>
<html>
<head>
<meta charset="utf-8">
<link rel="stylesheet" href="css/bootstrap.min.css">
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/events.css">
<script src="js/bootstrap.min.js" type="text/javascript"></script>
<script src="js/jquery-1.11.0.min.js"></script>
<script>
$(document).ready(function(){
$.ajax({
url: "/events.php?def",
cache: false,
beforeSend: function() {

```

```

$('#Event').html('');
},
success: function(html){
$('#Event').hide();
$('#Event').html(html);
$('#Event').show();
}
});
return false;
});
</script>
<script>
$(document).ready(function(){
$.ajax({
url: "/events.php?def_client",
cache: false,
beforeSend: function() {
$('#Client').html('');
},
success: function(html){
$('#Client').hide();
$('#Client').html(html);
$('#Client').show();
}
});
return false;
});
</script>
<title> Электронно-образовательная среда</title>
</head>
<body>
<divclass="hi">Вы зашли под пользователем: <?echo $_SESSION['fio']?><span class="label label-
success"><a href="del.php">Выход</a></span></div>
<div class="navbar navbar-inverse">
<div class="container">
<ul class="nav nav-tabs">
<li><a href="main.php">Задачи</li></a>
<li><a href="messages.php">Сообщения</li></a>
<li class="active"><a href="#">Календарь событий</a></li>
<li><a href="reports.php">Отчеты</a></li>
<li><a href="#">Расписание</a></li>
<li><a href="client.php">Клиенты</a></li>
</ul></div></div>
<div class="calendar-box">
<?/* Функция генерации календаря */
function draw_calendar($month,$year){
/* Начало таблицы */
$calendar = '<table class="table table-bordered table-condensed">';
/* Заглавия в таблице */
$headings = array('Пн','Вт','Ср','Чт','Пт','Сб','Вс');
$calendar.= '<tr class="calendar-row"><td class="calendar-day-head">'.implode('</td><td
class="calendar-day-head">', $headings). '</td></tr>';
/* необходимые переменные дней и недель... */
$running_day = date('w',mktime(0,0,0,$month,1,$year));
if($running_day == 0)$running_day = 6;
else $running_day = $running_day - 1;
$days_in_month = date('t',mktime(0,0,0,$month,1,$year));

```

```

$days_in_this_week = 1;
$day_counter = 0;
$dates_array = array();
/* первая строка календаря */
$calendar.= '<tr class="calendar-row">';
/* вывод пустых ячеек в сетке календаря */
for($x = 0; $x < $running_day; $x++):
    $calendar.= '<td class="calendar-day-np"></td>';
$days_in_this_week++;
endfor;
/* дошли до чисел, будем их писать в первую строку */
for($list_day = 1; $list_day <= $days_in_month; $list_day++):
    $calendar.= '<td class="calendar-day">';
    /* Пишем номер в ячейку */
    if ($list_day==date('d')):
        $calendar.= '<div class="day-num">'.$list_day.'</div>';
    else:
        $calendar.= '<div class="day-number">'.$list_day.'</div>';
    endif;

    $db = mysql_connect("localhost","root","");
    mysql_select_db("info", $db);
    $id=$_SESSION['id'];
    $query = mysql_query("select * from event where id_user='$id'");
    if($list_day < 10) $list_day="0$list_day";
    $i=0;
    while($result = mysql_fetch_array($query))
        {
            $i++;
            $date_parts = explode("-", $result['date']);
            $day = $date_parts[2];
            if ($list_day==$day) { ?>
                <script>
$(document).ready(function(){
$('#getEvents<?echo $i?>').click(function(){
$.ajax({
url: "/events.php?date=<?echo $result[2]?>",
cache: false,
beforeSend: function() {
$('#Event').html('');
},
success: function(html){
$('#Event').hide();
$('#Event').html(html);
$('#Event').show();});
return false;});});
</script>
<? $calendar .= '<a style="cursor:pointer" id="getEvents'.$i.'">Просмотреть</a>'; break; }
else $calendar.= ";}
    $calendar.= '</td>';
if($running_day == 6):
    $calendar.= '</tr>';
if(($day_counter+1) != $days_in_month):
    $calendar.= '<tr class="calendar-row">';
endif;
    $running_day = -1;
    $days_in_this_week = 0;
endif;
    $days_in_this_week++; $running_day++; $day_counter++;
endfor;
/* Выводим пустые ячейки в конце последней недели */

```

```

if($days_in_this_week < 8):
for($x = 1; $x <= (8 - $days_in_this_week); $x++):
    $calendar.= '<td class="calendar-day-np"></td>';
endfor;
endif;
/* Закрываем последнюю строку */
$calendar.= '</tr>';
/* Закрываем таблицу */
$calendar.= '</table>';
/* Все сделано, возвращаем результат */
return $calendar;
}
/* СПОСОБ ПРИМЕНЕНИЯ */
$day = date('d');
$mont = date('m');
$ye = date('Y');
echo draw_calendar($mont, $ye);    ?>
</div>
<div class="create">
<div class="event-box-header">
<h4 style="text-align:center">Добавить новое событие</h4><hr>
</div>
<div class="event-box-body">
<form method="POST" action="edit_event.php" style="margin-left:5%;">
<select name="client" class="span4">
<option value=" disabled selected style='display:none;'>Выберите клиента</option><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$qr_result1 = mysql_query("select `fio` from `client`");
while($client = mysql_fetch_array($qr_result1)){?>
<option value="<?echo $client[fio]?>"><?echo $client[fio]?></option><?}?>
</select><br>
<input type="date" class="span5" style="height: 30px" name="date" placeholder="Введите дату
события"><br>
<textarea class="span5" name="text" rows="5" placeholder="Введите описание
события"></textarea><br>
<input type="submit" class="span3 btn btn-success" style="margin-left:20%" name="create_event"
value="Добавить">
<?if ($_SESSION[job]=="Администратор")?><br><br><input type="submit" class="btn"
style="margin-left:16%" name="del_all_events" value="Удалить все события в системе"><?}?>
</form></div></div>
<div class="events">
<div id="showEvent" class="event-box">
<div class="event-box-body">
<div id="Event"></div>
</div></div></div>
<?>
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$query = mysql_query("select * from `event`");
$i=0;
while($result = mysql_fetch_array($query))
{
    $i++;
    ?>
}
</script>
$(document).ready(function(){
    $('#getEvents<?echo $i?>').click(function(){
        $.ajax({

```

```

url: "/events.php?date1=<?echo $result[2]?>",
cache: false,
beforeSend: function() {
$( "#Client" ).html('');
},
success: function(html){
$( "#Client" ).hide();
$( "#Client" ).html(html);
$( "#Client" ).show();
}});
return false;});});
</script>
<?}?>
<div class="client">
<div class="client-box-body">
<div id="Client"></div>
</div></div></body></html>

```

Events.php

```

<?session_start();
include("data_convert.php");
if(isset($_GET['def'])) {
?>
<table class="table table-hover">
<thead>
<tr><th colspan="2"><h4 style="text-align:center;padding:0px 10px 0px 10px">Список событий на
сегодня</h4></th></tr>
</thead>
<tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$i=0;
$today = date("Y-m-d");
$id=$_SESSION['id'];
$sql = mysql_query("SELECT * FROM `event` WHERE `id_user`=' $id' and `date`=' $today'");
while($event = mysql_fetch_array($sql)){ $i++;?>
<tr><td><?echo $i?></td><td><?echo $event[text]?></td><td><?if($event[complete]=='1') {echo
"<i>выполнено</i>";} else echo "<form method='post' action='edit_event.php'><input type='hidden'
name='id' value='$event[id]'><input class='btn' type='submit' name='ok'
value='Готово'></form>";?></td></tr><?>
if($i==0){?><tr><td style="text-align:center;padding:10px" colspan="2">На этот день событий
нет</td></tr><?>
</tbody>
</table>
<?>
if(isset($_GET['def_client'])) {
?>
<table class="table table-hover">
<thead>
<tr><th colspan="2"><h4 style="text-align:center;padding:0px 10px 0px 10px">Информация о
клиентах</h4></th></tr>
</thead>
<tbody>
<?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$today = date("Y-m-d");

```

```

$sql = mysql_query("SELECT * FROM `event` WHERE `id_user`='$_id' and `date`='$_today'");
while($event = mysql_fetch_array($sql)){
    $fio_cut = explode("Клиент: ", $event['text']);
    $fio = $fio_cut[1];
    $sql1 = mysql_query("SELECT DISTINCT `fio`, `phone` FROM `client` WHERE `fio`='$_fio'");
    $i=0;
    while($client = mysql_fetch_array($sql1)){ $i++;?>
        <tr><td style="padding:10px"><?echo $client['fio']?></td><td style="padding:10px"><?echo
        $client['phone']?></td></tr><?>
        if($i==0){?><tr><td style="text-align:center;padding:10px" colspan="2">На этот день клиентов
        нет</td></tr><?break;}}?>
    </tbody>
</table>
<?>
if(isset($_GET[date])){
    ?>
    <table class="table table-hover">
    <thead>
    <tr><th colspan="2"><h4 style="text-align:center;padding:0px 10px 0px 10px">Список событий на
    <?echo data_convert($_GET[date],1,0,0)?></h4></th></tr>
    </thead>
    <tbody>
    <?>
    $db = mysql_connect("localhost","root","");
    mysql_select_db("info", $db);
    $i=0;
    $id=$_SESSION['id'];
    $day = $_GET['date'];
    $sql = mysql_query("SELECT * FROM `event` WHERE `id_user`='$_id' and `date`='$_day'");
    while($event = mysql_fetch_array($sql)){ $i++;?>
        <tr><td><?echo $i?></td><td><?echo $event[text]?></td><td><?if($event[complete]=='1') {echo
        "<i>выполнено</i>";} else echo "<form method='post' action='edit_event.php'><input type='hidden'
        name='id' value='$_event[id]'><input class='btn' type='submit' name='ok'
        value='Готово'></form>";?></td></tr><?>
        if($i==0){?><tr><td style="text-align:center;padding:10px" colspan="2">На этот день событий
        нет</td></tr><?>
    </tbody>
</table>
<?>
if(isset($_GET['date1'])){
    ?>
    <table class="table table-hover">
    <thead>
    <tr><th colspan="2"><h4 style="text-align:center;padding:0px 10px 0px 10px">Информация о
    клиенте</h4></th></tr>
    </thead>
    <tbody><?>
    $db = mysql_connect("localhost","root","");
    mysql_select_db("info", $db);
    $id=$_SESSION['id'];
    $day = $_GET['date1'];
    $sql = mysql_query("SELECT * FROM `event` WHERE `id_user`='$_id' and `date`='$_day'");
    while($event = mysql_fetch_array($sql)){
        $fio_cut = explode("Клиент: ", $event['text']);
        $fio = $fio_cut[1];
        $sql1 = mysql_query("SELECT DISTINCT `fio`, `phone` FROM `client` WHERE `fio`='$_fio'");
        $i=0;

```

```

while($client = mysql_fetch_array($sql1)){ $i++?>
<tr><td style="padding:10px"><?echo $client['fio']?></td><td style="padding:10px"><?echo
$client['phone']?></td></tr><?>
if($i==0){?><tr><td style="text-align:center;padding:10px" colspan="2">На этот день клиентов
нет</td></tr><?break;??>
</tbody>
</table>
<?}?>

```

Reports.php

```

<?session_start();
include("data_convert.php");?>
<!doctype html>
<html>
    <head>
        <meta charset="utf-8">
        <link rel="stylesheet" href="../css/bootstrap.min.css">
        <link rel="stylesheet" href="css/style.css">
        <link rel="stylesheet" href="css/messages.css">
        <script src="../js/jquery-1.11.0.min.js"></script>
        <script src="../js/bootstrap.min.js" type="text/javascript"></script>
        <title>Отчеты</title>
    </head>
    <body>
        <div class="hi">Вы зашли под пользователем: <?echo $_SESSION[fio]?><span class="label label-
        success"><a href="del.php">Выход</a></span></div>
        <div class="navbar navbar-inverse">
            <div class="container">
                <ul class="nav nav-tabs">
                    <li><a href="main.php">Задачи</li></a>
                    <li><a href="messages.php">Сообщения</li></a>
                    <li><a href="kalen.php">Календарь событий</li></a>
                    <li class="active"><a href="reports.php">Отчеты</li></a>
                    <li><a href="#">Расписание</li></a>
                    <li><a href="client.php">Клиенты</li></a>
                </ul></div></div>
                <div class="content">
                    <div class="tabbable tabs-left">
                        <ul class="nav nav-tabs">
                            <li>Отчеты по задачам</li>
                            <li><a href="#complete_tasks" data-toggle="tab">Завершенные задачи</a></li>
                            <li><a href="#performed_tasks" data-toggle="tab">Выполняемые задачи</a></li>
                            <?if ($ _SESSION[job]=="Администратор")?><li><a href="#worker_tasks" data-
                            toggle="tab">Задачи сотрудника</a></li><?>
                        </ul>
                        <div class="tab-content">
                            <div class="tab-pane active" id="hello">
                                <p>Выберите нужный вам отчет в списке слева.</p>
                            </div>
                            <div class="tab-pane" id="complete_tasks">
                                <h3 style="text-align:center">Отчет по завершенным задачам</h3><br>
                                <?
                                $db = mysql_connect("localhost","root","");
                                mysql_select_db("info", $db);
                                $id=$_SESSION['id'];
                                $j=0;
                                $sql_com = mysql_query("SELECT count(`id_zadacha`), min(`data`) FROM `zadacha` WHERE
                                `prochten`='1' and `vipol`='1' and `id_pol`='".$id"'");

```

```

$task_com = mysql_fetch_array($sql_com);
if($task_com[0]!=0){?><p style="text-indent:20px"><?echo $_SESSION['fio']?>, за период с <?echo
data_convert($task_com[1],1,0,0)?> по <?echo data_convert(date("Y-m-d"),1,0,0)?>, выполнил(а) <?echo
$task_com[0]?> задач.</p><?>
if($task_com[0]==0){?><p style="text-indent:20px"><?echo $_SESSION['fio']?> не выполнил(а) ни
одной задачи.</p><?>?>
<br><pstyle="text-indent:20px">Более подробная информация представлена в таблице.</p>
<table border="1">
<thead>
<tr><th>Ом</th><th>Задача</th><th>Дата получения</th><th>Состояние</th></tr>
</thead>
<tbody>
<?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$i=0;
$sql_com = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_otp`, `user`.`fio`,
`zadacha`.`id_pol`, `zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM
`zadacha`,`user` WHERE `zadacha`.`id_otp`=`user`.`id` and `zadacha`.`prochten`='1' and
`zadacha`.`vipol`='1' and `zadacha`.`id_pol`=`$id` order by `zadacha`.`id_zadacha` desc");
while($task_com = mysql_fetch_array($sql_com)){ $i++;?>
<tr><td style="padding:5px"><?echo $task_com[2]?></td><td style="padding:5px"><?echo
$task_com[4]?></td><td style="padding:5px"><?echo data_convert($task_com[5],1,0,0)?></td><td
style="padding:5px">Готово</td></tr><?>
if($i==0){?><tr><td style="text-align:center;padding:10px" colspan="5">У вас нет завершенных
задач</td></tr><?>?>
</tbody></table>
<br><br>
<p>Отчет сформирован сотрудником: <?echo $_SESSION['fio']?></p>
<p>Дата: <?echo data_convert(date("Y-m-d"),1,0,0)?></p>
<a class="btn" href="reports_complete_tasks.php" target="_blank">Распечатать</a>
</div>
<div class="tab-pane" id="performed_tasks">
<h3 style="text-align:center">Отчет по выполняемым задачам</h3><br>
<?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];
$j=0;
$sql_op = mysql_query("SELECT count(`id_zadacha`), min(`data`) FROM `zadacha` WHERE `prochten`='1'
and `vipol`='0' and `id_pol`=`$id`");
$task_op = mysql_fetch_array($sql_op);
if($task_op[0]!=0){?><p style="text-indent:20px">У сотрудника <?echo $_SESSION['fio']?>, за период
с <?echo data_convert($task_op[1],1,0,0)?> по <?echo data_convert(date("Y-m-d"),1,0,0)?>, на
выполнении находится <?echo $task_op[0]?> задач.</p><?>
if($task_op[0]==0){?><p style="text-indent:20px"><?echo $_SESSION['fio']?> не выполняет ни одной
задачи.</p><?>?>
<br><pstyle="text-indent:20px">Более подробная информация представлена в таблице.</p>
<table border="1">
<thead>
<tr><th>Ом</th><th>Задача</th><th>Дата получения</th><th>Состояние</th></tr>
</thead>
<tbody><?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$id=$_SESSION['id'];

```

```

$ج=0;
$جsql_op = mysql_query("SELECT `zadacha`.`id_zadacha`, `zadacha`.`id_otp`, `user`.`fio`,
`zadacha`.`id_pol`, `zadacha`.`name`, `zadacha`.`data`, `zadacha`.`vipol`, `zadacha`.`prochten` FROM
`zadacha`,`user` WHERE `zadacha`.`id_otp`=`user`.`id` and `zadacha`.`prochten`='1' and
`zadacha`.`vipol`='0' and `zadacha`.`id_pol`='$id' order by `zadacha`.`id_zadacha` desc");
while($task_op = mysql_fetch_array($جsql_op)){ $ج++;?>
<tr><td style="padding:5px"><?echo $task_op[2]?></td><td style="padding:5px"><?echo
$task_op[4]?></td><td style="padding:5px"><?echo $task_op[5]?></td><td
style="padding:5px">Выполняется</td></tr><?>
if($ج==0){?><tr><td style="text-align:center;padding:10px" colspan="5">У вас нет выполняемых
задач</td></tr><?>?>
</tbody></table>
<br><br>
<p>Отчет сформирован сотрудником: <?echo $_SESSION['fio']?></p>
<p>Дата: <?echo date_convert(date("Y-m-d"),1,0,0)?></p>
<a class="btn" href="reports_performed_tasks.php" target="_blank">Распечатать</a>
</div>
<?if ($_SESSION[job]=="Администратор")?>
<div class="tab-pane" id="worker_tasks">
<form method="POST" target="_blank" action="reports_worker_tasks.php">
<select name="id" class="span6">
<option value="" disabled selected style='display:none;'>Выберите сотрудника</option>
<?
$جdb = mysql_connect("localhost","root","");
mysql_select_db("info", $جdb);
$جqr_result = mysql_query("select `id`,`fio` from `user`");
while($جuser = mysql_fetch_array($جqr_result)){?>
<option value=""><?echo $جuser[id]?>><?echo $جuser[1]?></option><?>?>
</select>
<br>
<input type="submit" class="btn" value="Распечатать">
</form></div><?>?>
</div></div></div></body>
</html>

```

Client.php

```

<?session_start();
include("data_convert.php");?>
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<link rel="stylesheet" href="../css/bootstrap.min.css">
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/messages.css">
<script src="../js/jquery-1.11.0.min.js"></script>
<script src="../js/bootstrap.min.js" type="text/javascript"></script>
<title>Клиенты</title>
</head>
<body>
<div class="hi">Вы зашли под пользователем: <?echo $_SESSION[fio]?><span class="label label-
success"><a href="del.php">Выход</a></span></div>
<div class="navbar navbar-inverse">
<div class="container">
<ul class="nav nav-tabs">
<li><a href="main.php">Задачи</a></li>
<li><a href="messages.php">Сообщения</a></li>
<li><a href="kalen.php">Календарь событий</a></li>

```

```

</li><a href="reports.php">Отчеты</a></li>
</li><a href="#">Расписание</a></li>
<li class="active"><a href="client.php">Клиенты</li></a>
</ul></div></div><div class="content">
<div class="tabbable tabs-left">
<ul class="nav nav-tabs">
<li>Клиенты</li>
<li><a href="#list_client" data-toggle="tab">Список клиентов</a></li>
<li><a href="#add_client" data-toggle="tab">Добавить клиента</a></li>
</ul>
<div class="tab-content">
<div class="tab-pane active" id="hello">
<p>Выберите нужный вам раздел в списке слева.</p>
</div>
<div class="tab-pane" id="add_client">
<form method="POST" action="add_client.php">
<input class="span6" name="fio" size="16" type="text" placeholder="Введите ФИО клиента"><br>
<input class="span6" name="phone" size="16" type="tel" pattern="(\d\d\d\d) ?\d\d\d-\d\d\d-\d\d\d"
placeholder="(###)###-##-##"><br>
<textarea class="span6" name="text" rows="7" placeholder="Введите дополнительную информацию о
клиенте"></textarea><br>
<input type="submit" class="span4 btn btn-large btn-success" style="margin-left:85px" name="add"
value="Добавить">
</form></div>
<div class="tab-pane" id="list_client">
<table class="table">
<thead>
<tr><th>ФИО</th><th>Телефон</th></tr>
</thead>
<tbody>
<form method="post" action="add_client.php">
<?
$db = mysql_connect("localhost","root","");
mysql_select_db("info", $db);
$i=0; // используется для проверки.если она будет равна 0, значит нету клиентов
$sql = mysql_query("SELECT * FROM `client` ORDER BY `fio`");
while($client = mysql_fetch_array($sql)){ $i++;?>
<tr style='cursor:pointer'><td><a href="#<?echo $client[0]?>" data-toggle="modal"><?echo
$client[1]?></td><td><?echo $client[2]?></a></td></tr>
<!-- Модальное окно просмотра входящих сообщения -->
<div class="modal hide fade" id="<?echo $client[0]?>" tabindex="-1" role="dialog">
<div class="modal-header"><button class="close" type="button" data-dismiss="modal">×</button></div>
<h4 class="text-muted" id="myModalLabel"><?echo $client[1]?></h4>
</div>
<div class="modal-body">
<input type="text" name="edit_phone<?echo $i?>" value="<?echo $client[2]?>">
<input type="hidden" name="client_id<?echo $i?>" value="<?echo $client[0]?>">
<input type="hidden" name="kol" value="<?echo $i?>">
</div>
<div class="modal-footer">
<input type="submit" name="ok" value="Изменить" class="btn">
<button class="btn" data-dismiss="modal">Закрыть</button>
</div>
</div>
<!-- Конец модального окна просмотра входящих сообщения -->
<?}></tbody></table></form></div></div></div></div></body>
</html>

```

ПРИЛОЖЕНИЕ Б

Критические значения критерия U Манна-Уитни для уровней статистической значимости $p \leq 0,05$ и $p \geq 0,01$ (по Гублеру Е.В., Генкину А.А., 1973).

Различия между двумя выборками можно считать значимыми ($p < 0,05$), если $U_{\text{эмп}}$ ниже или равен $U_{0,05}$ и тем более достоверными ($p < 0,01$), если $U_{\text{эмп}}$ ниже или равен $U_{0,01}$.

n_1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
n_2	$p=0,05$																			
3	-	0																		
4	-	0	1																	
5	0	1	2	4																
6	0	2	3	5	7															
7	0	2	4	6	8	11														
8	1	3	5	8	10	13	15													
9	1	4	6	9	12	15	18	21												
10	1	4	7	11	14	17	20	24	27											
11	1	5	8	12	16	19	23	27	31	34										
12	2	5	9	13	17	21	26	30	34	38	42									
13	2	6	10	15	19	24	28	33	37	42	47	51								
14	3	7	11	16	21	26	31	36	41	46	51	56	61							
15	3	7	12	18	23	28	33	39	44	50	55	61	66	72						
16	3	8	14	19	25	30	36	42	48	54	60	65	71	77	83					
17	3	9	15	20	26	33	39	45	51	57	64	70	77	83	89	96				
18	4	9	16	22	28	35	41	48	55	61	68	75	82	88	95	102	109			
19	4	10	17	23	30	37	44	51	58	65	72	80	87	94	101	109	116	123		
20	4	11	18	25	32	39	47	54	62	69	77	84	92	100	107	115	123	130	138	
n_2	$p=0,01$																			
5	-	-	0	1																
6	-	-	1	2	3															
7	-	0	1	3	4	6														
8	-	0	2	4	6	7	9													
9	-	1	3	5	7	9	11	14												
10	-	1	3	6	8	11	13	16	19											
11	-	1	4	7	9	12	15	18	22	25										
12	-	2	5	8	11	14	17	21	24	28	31									
13	0	2	5	9	12	16	20	23	27	31	35	39								
14	0	2	6	10	13	17	22	26	30	34	38	43	47							
15	0	3	7	11	15	19	24	28	33	37	42	47	51	56						
16	0	3	7	12	16	21	26	31	36	41	46	51	56	61	66					
17	0	4	8	13	18	23	28	33	38	44	49	55	60	66	71	77				
18	0	4	9	14	19	24	30	36	41	47	53	59	65	70	76	82	88			
19	1	4	9	15	20	26	32	38	44	50	56	63	69	75	82	88	94	101		
20	1	5	10	16	22	28	34	40	47	53	60	67	73	80	87	93	100	107	114	