Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Воронежский государственный лесотехнический университет

имени Г.Ф. Морозова»

Кафедра вычислительной техники и информационных систем

(название кафедры)

|  |  |
| --- | --- |
|  | **Допустить к защите**  Заведующий кафедрой  \_\_\_\_\_\_\_\_\_\_\_\_\_ \_ В.К. Зольников\_\_\_\_  (подпись) (инициалы и фамилия)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (число, месяц, год) |

**ВыпускнАЯ квалификационнАЯ работА**

Разработка модуля обучающей информационной системы по   
дисциплине

«Типы информационных систем и их администрирование»

(тема)

\_\_\_\_\_\_\_\_09.03.02 Информационные системы и технологии\_\_\_\_\_\_\_

(код и наименование направления подготовки)

|  |  |  |
| --- | --- | --- |
| Студент группы \_\_\_\_ИС2-151-ОБ\_\_\_\_\_\_  (обозначение группы)  Руководитель, \_\_\_к.т.н.\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (ученая степень, ученое звание)  Консультант по оформлению, к.т.н., доцент | \_\_\_\_\_\_\_\_\_\_\_  (подпись)  \_\_\_\_\_\_\_\_\_\_\_  (подпись)  \_\_\_\_\_\_\_\_\_\_\_  (подпись) | \_М.О.\_Чайка\_\_  (инициалы и фамилия)  \_Н.Ю. Юдина\_\_  (инициалы и фамилия)  С.А. Евдокимова  (инициалы и фамилия) |

Воронеж 2019

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«Воронежский государственный лесотехнический университет

имени Г.Ф. Морозова»

Кафедра вычислительной техники и информационных систем

(название кафедры)

|  |  |
| --- | --- |
|  | **согласовано**  Заведующий кафедрой  \_\_\_\_\_\_\_\_\_\_\_\_\_ \_В.К. Зольников\_  (подпись) (инициалы и фамилия)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (число, месяц, год) |

**ЗАДАНИЕ**

**на выпускную квалификационную работу**

Студенту \_\_\_\_\_\_\_\_Чайка Максим Олегович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия и полные инициалы)

Группы ИС2-151-ОБ

Разработка модуля обучающей информационной системы по   
дисциплине

«Типы информационных систем и их администрирование»

(тема)

\_\_\_\_\_\_\_09.03.02 Информационные системы и технологии\_\_\_\_

(код и наименование направления подготовки)

Утвержденараспоряжениемдеканата механического факультета  
 № 36 от «28» февраля 2019 г.

Срок представления работы к защите «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2019 г.

Исходные данные: учебно-методический материал по дисциплине «Типы информационных систем и их администрирование» для направления подготовки 09.03.02 – Информационные системы и технологии (методички, лекции, лабораторные работы, вопросы для проверки остаточных знаний, упражнения);

Перечень вопросов, подлежащих разработке: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1.Проанализировать принципы построения обучающих систем.

2.Разработать функциональную схему и компоненты автоматизированной

обучающей системы.

3.Разработать программный модуль автоматизированной обучающей системы

по дисциплине «Типы информационных систем и их администрирование».

Перечень графических документов или иллюстративного материала:

Плакат 1 : Актуальность.

Плакат 2: Цель и задачи.

Плакат 3: Анализ существующих информационных обучающих систем.

Плакат 4: Диаграммы прецедентов.

Плакат 5: Диаграммы последовательностей.

Плакат 6: Схема базы данных.

Плакат 7: Интерфейс программы.

Плакат 8: Интерфейс для работы с базой данных.

Плакат 9: Интерфейс модуля преподавателя.

Плакат 10: Интерфейс модуля студента.

Плакат 11: Заключение.

Руководитель, \_\_\_\_к.т.н., доцент\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.Ю. Юдина

(ученая степень, ученое звание) (подпись) (инициалы и фамилия)

Консультант по оформлению С.А. Евдокимова

(инициалы и фамилия)

Задание принял студент \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_М.О. Чайка\_\_

(подпись) (число, месяц, год) (инициалы и фамилия)

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 5](#_Toc11947517)

[1 Анализ принципов построения информационных обучающих систем 7](#_Toc11947518)

[1.1 Общая характеристика информационных обучающих систем 7](#_Toc11947519)

[1.2 Обзор обучающих информационных систем 9](#_Toc11947520)

[1.3 Стандарты в области образовательных технологий 11](#_Toc11947521)

[2 Разработка концепции, логической и физической модели информационной обучающей системы 14](#_Toc11947522)

[2.1 Концепция информационной обучающей системы 14](#_Toc11947523)

[2.2 Функциональная структура системы 15](#_Toc11947524)

[2.3 Описание объекта автоматизации 16](#_Toc11947525)

[2.4 Проект логической реализации информационной обучающей системы 23](#_Toc11947526)

[2.5 Разработка реляционной модели базы данных обучающей системы 29](#_Toc11947527)

[3 Разработка программного модуля обучающей информационной системы 41](#_Toc11947528)

[3.1 Обоснование выбора языка и среды программирования 41](#_Toc11947529)

[3.2 Обоснование выбора системы управления базами данных 41](#_Toc11947530)

[3.3 Работа обучающей информационной системы 42](#_Toc11947531)

[ЗАКЛЮЧЕНИЕ 50](#_Toc11947532)

[СПИСОК ЛИТЕРАТУРЫ 51](#_Toc11947533)

[Приложение А. Листинг программы 52](#_Toc11947534)

# ВВЕДЕНИЕ

Развитие информационных технологий оказывает влияние на все сферы жизни современного общества. Изменяются потребности людей, их образ жизни и особенности восприятия информации. Все это отражается на такой неотъемлемой части жизни человека как образование.

Для того что бы соответствовать современным тенденциям и особенностям спроса современные образовательные системы должны обладать рядом черт:

1. Гибкость – возможность подстраиваться под нужды студента
2. Технологичность – использование возможностей, предоставляемых современными технологиями.
3. Адаптивность – способность к изменению и модернизации системы
4. Масштабируемость – возможность применения в рамках одной дисциплины или интеграции в систему управления образованием на уровне ВУЗа

В настоящее время разработка обучающих систем, отвечающих этим требованиям, достаточно распространена, однако не покрывает в полной мере возрастающий спрос [1, 5, 8, 9].

Существующие информационные обучающие системы обладают высокой стоимостью и зачастую имеют избыточный функционал, который усложняет использование и освоение обучающей системы. Таким образом, разработка обучающей информационной системы для конкретных условий использования является актуальной.

Целью выпускной квалификационной работы является разработка программного модуля информационной обучающей системы.

Для этого необходимо решить следующие задачи:

1. Проанализировать принципы построения информационных обучающих систем.
2. Разработать логическую модель, функционал и информационное обеспечение информационной обучающей системы.
3. Разработать программный модуль информационной обучающей системы.

Данная[**38**](https://vgltu.antiplagiat.ru/report/print/265?c=0) выпускная квалификационная работа состоит из введения, трех глав[**38**](https://vgltu.antiplagiat.ru/report/print/265?c=0), в[**17**](https://vgltu.antiplagiat.ru/report/print/265?c=0)которых проведены: анализ использования информационных систем в сфере образования, разработана модель реляционной базы данных, а также логическая модель информационной обучающей системы и, наконец разработан программный модуль ИОС по дисциплине «Типы информационных систем и их администрирование». Завершает ВКР заключение о проделанной работе, список использованной литературы, а также приложение А, содержащее листинг программы.

# 1 Анализ принципов построения информационных обучающих систем

## 1.1 Общая характеристика информационных обучающих систем

Системы автоматизированного обучения развиваются уже более тридцати лет. На ранних порах развития это были программные комплексы для изучения студентами информационных технологий, в то же время получили развитие системы автоматического оценивания знаний обучающихся. Во второй половине восьмидесятых годов появляются автоматизированные обучающие системы на основе учебников с использованием графических технологий[5,17,21].

Появление Internet и web технологий в девяностых годах создало предпосылки для создания концепции открытого образования, хотя существующие на тот момент аналоги АОС не предоставляли достаточный функционал для осуществления идей дистанционного образования. Отсутствовали единые стандарты представления и обмена данных между обучающими системами. Отсутствовала гибкость систем и возможности подстраиваться под нужды пользователя. Сопровождение обучающих систем и обновление информации представляло собой громоздкий и неэффективный механизм.[8,12,13,23]

В отечественной системе образования первые информационные системы начали создаваться во второй половине 20 века.

В системе образования управленческой деятельности можно выделить следующие уровни с использованием ЭВМ:

1) Индивидуальный – управление развитием и обучением отдельно взятого учащегося

2) Комплексный – управление работой группы родственных учебных заведений

3) Локальный – управление процессом в рамках одного учебного заведения

4) Региональный – управление учебными заведениями по территориальному принципу

5) Федеральный – управление системой образования страны[1,22,16]

На первом уровне управленческой деятельности задачи управления совпадают с задачами обучения при помощи компьютеров. На втором уровне действительные успехи достигаются в вузах. Во-первых, это связано с тем, что государственные высшие учебные заведения имеют большой контингент учащихся и преподавателей, а также большую материальную базу, чтобы использование компьютеров было экономически оправданно, во-вторых, в вузах, особенно технических, работают профессионально подготовленные и квалифицированные кадры, которые при своем опыте могут решать проблемы информатизации управления.[11]

В рамках автоматизированных информационных обучающих систем на сегодняшний день решается ряд задач обучения. В первую группу можно отнести задачи проверки уровня знаний, умений и навыков студентов до и после обучения, их индивидуальных способностей, склонностей и мотиваций. Для таких проверок обычно используют соответствующие системы тестов и экзаменационных вопросов

Вторая группа задач связана с учетом и статическим анализом показателей усвоения учебного материала: баллы, оценки, определение времени, отведенного на решения задач, процент ошибок и т.д.

Третья группа задач связана с решением задач подготовки и предъявления учебного материала адаптации материала по уровням сложности, контрольных заданий, лабораторных работ самостоятельных работ студентов. Проще говоря, использование подобных программных продуктов дает возможность оценить успеваемость студентов, формирует различные статистические данные по лабораторным работам и др.[11,22,19]

## 1.2 Обзор обучающих информационных систем

Разрабатывать обучающие информационные системы является не простым делом, это весьма трудоемкий и дорогостоящий процесс, в силу своей высокой наукоемкости и требовательности к совместной разработке высококвалифицированных специалистов: преподавателей-предметников, дизайнеров, психологов.[4]

Несмотря на это, крупные компании и организации финансируют проекты создания обучающих систем в учебных заведениях и ведут собственные разработки в этой области.

Изменения качественного инструментария в разработке средств обучения предполагают разработку абсолютных новых методических подходов в организации процессов, это должно позволить подготавливать специалистов в различные сферы деятельности. И благодаря этому, с одной стороны, сделает возможным систематизировать уже накопленный опыт в рассматриваемой области, а с другой стороны – поможет сформулировать главные требования к вновь разрабатываемым системам автоматизированного обучения.[12]

Вполне очевидно, что при развитии компьютерных средств учебного назначения в основе новых разрабатываемых информационных технологии должна находиться практика уже существующих результатов в рассматриваемой области средств обучения.

Среди существующих информационных обучающих систем можно выделить следующие:

1. WebTutor– Это закрытый пакет программного обеспечения от компании WebSoft. Его основными преимуществами являются возможность запуска на разных платформах (Web, PS, Android, iOS), наличие готовых курсов и удобный интерфейс. К минусам стоит отнести высокую стоимость, (в т.ч. наличие большого количества функционала, продающегося отдельно), продолжительный процесс внедрения, сложная система выгрузки отчетов.[9]
2. Moodle – Представляет собой свободное веб-приложение, предоставляющее возможность создавать сайты для онлайн-обучения. Среди явных достоинств стоит выделить бесплатность, набор основных модулей для организации обучения. Так же стоит отметить не очень удобный, но простой интерфейс. К недостаткам относится необходимость продолжительной настройки и доработки. (На основе этой системы построен сайт дистанционного обучения CDOVGLTU.RU)[10]
3. Teachbase компании Интернет-школа – российский интернет-сервис дистанционного обучения, сервис строго ориентирован на рынок корпоративного обучения, обладает большим и разнообразным функционалом, реализуется по дорогостоящей подписке.[11]

Таблица 1.1 – Обзор обучающих информационных систем

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Производитель | Описание | Стоимость |
| TeachBase | Интернет-школа  (Россия) | Ориентированный на рынок корпоративного обучения сервис | 4000р/мес. – 35000р/мес. |
| iSpring | iSpring  (Россия) | Онлайн платформа дистанционного обучения | От 5000р/мес. |
| Acrobat ConnectPro | Adobe Systems Incorporated (США) | Платформа для проведения вебинаров и онлайн обучения | От 25 000 р./мес. |
| Moodle | MoodlePty LTD | Система управления обучением с открытым кодом | бесплатно |

## 1.3 Стандарты в области образовательных технологий

Над решением проблемы унификации форматов данных и представления информации в информационных обучающих системах работают следующие международные организации:

IMS Global Learning Consortium– международный образовательный консорциум, развивающий концепцию, технологии и стандарты обучения на базе системы управления обучением IMS (Instructional Management System);

IEEE LTSC – IEEE Learning Technology Standards Committee –комитет стандартизации в области технологий обучения, созданный в IEEE (Institute of Electrical and Electronics Engineers);

AICC – Aviation Industry CBT Committee– комитет компьютерного обучения в авиационной промышленности;

ADL – Advanced Distributed Learning Initiative Network– организация распределенного обучения, основанная департаментом политики в области науки и технологий в администрации президента США (OSTP –White House Office of Science and Technology Policy) и министерством обороны США (DoD), как сеть распределенного обучения, обеспечивающая широкомасштабный доступ к образовательным ресурсам многих пользователей.[23]

Результатом работы этих организаций являются следующие работы:

1. Спецификации IMS– включает в себя компоновку содержимого учебников, описание данных обучающихся, описание учебных материалов, описание типичных вопросов и средств тестирования а так же описание хранилищ цифровых данных. Процедуры учебного процесса в концепции IMSпредставлены на рисунке 1.1

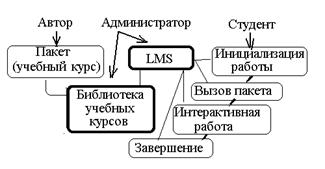


Рисунок 1.1 – Процедуры учебного процесса и роли участников  
 в концепции IMS

1. Спецификация IEEELTSC – набор документов комитета по стандартизации образовательных технологий, описывающих модели архитектуры образовательных систем, терминологические словари, управление обучением, протоколы обмена данными и профили платформ и сред.
2. Модель SCORM – промышленный стандарт для обмена учебными материалами на базе адаптированных спецификаций ADL, IEEE, IMS, DublinCore, andvCard. Предназначена для многократного использования образовательных модулей, облегченного сопровождения и коррекции курсов, модификации деталей отдельных модулей в соответствии с требованиями пользователей.
3. Модель сетевого дистанционного обучения E-Lerning – Модель взаимодействия объекта обучения (студента) и преподавателя в рамках среды дистанционного обучения. Представлена на рисунке 1.2[23]

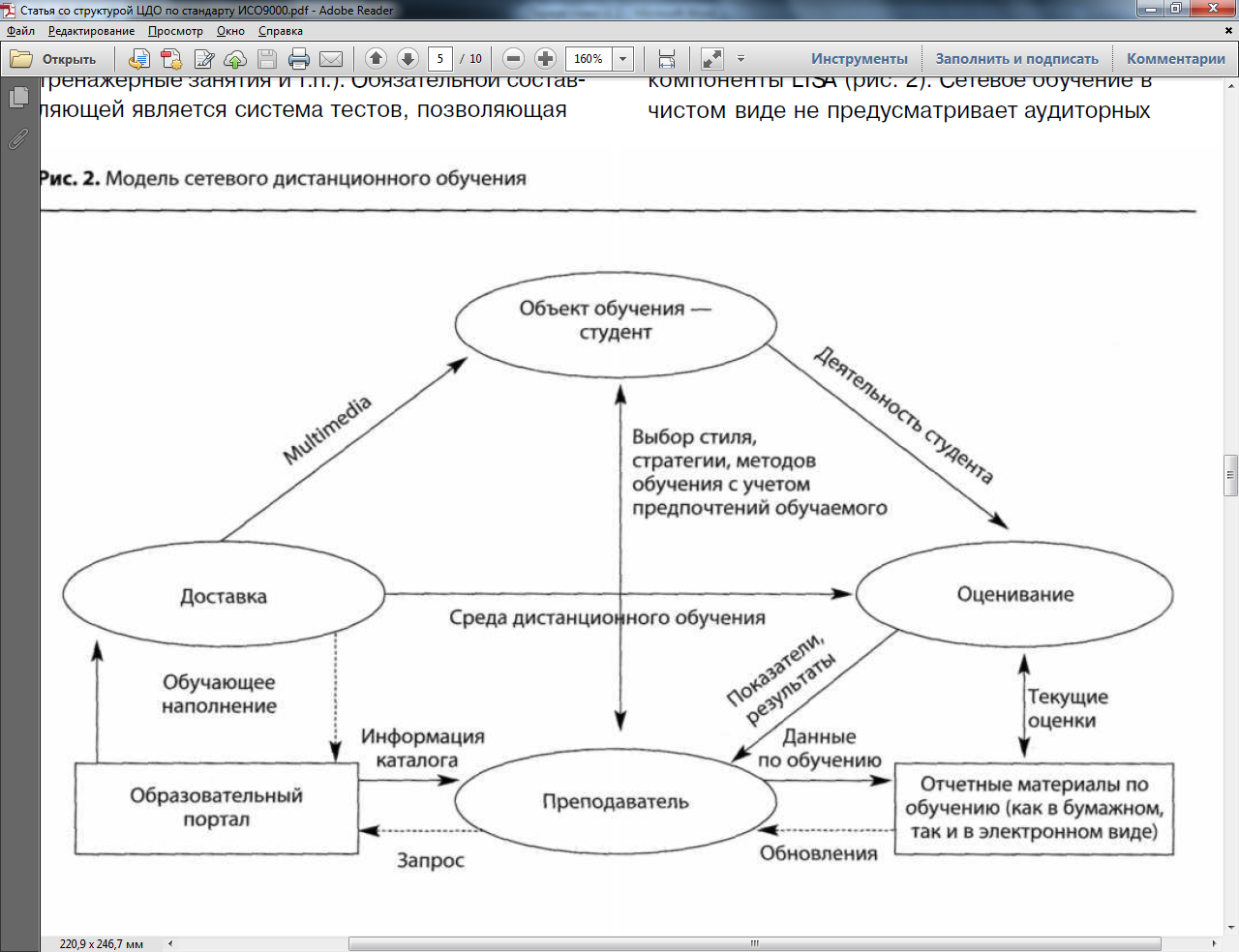


Рисунок 1.2 – Модель сетевого дистанционного обучения

Данная модель предоставляет свободный выбор временных рамок и места обучения. Она предполагает самостоятельную работу студента в рамках образовательной дисциплины нацеленную на изучение теории и выполнение тестовых заданий для формирования компетенций.[17]

# 2 Разработка концепции, логической и физической модели информационной обучающей системы

## 2.1 Концепция информационной обучающей системы

Целью выпускной квалификационной работы является разработка модуля обучающей информационной системы по дисциплине «Типы информационных систем и их администрирование» Это программный комплекс, который обеспечит возможность создания и управления лекционным курсом, предоставления его для изучения студентам и контроля усвоения материала.

На основе анализа имеющихся систем управления обучением с учетом особенностей разрабатываемой информационной системы ее проектирование состоит в следующих этапах:

1. Разработка функциональной структуры информационной системы
2. Разработка базы данных
3. Разработка подсистем: Администрирования программы, Управления курсами и лекциями и подсистемы изучения материала.

В разрабатываемой информационной системе предусмотрены 3 вида пользователей:

1) Администратор – обладает полными полномочиями для просмотра и редактирования базы данных и управления пользователями.

2) Преподаватель – Управляет определенным количеством дисциплин и входящего в них материала, отслеживает успеваемость студентов

3) Студент – Имеет доступ к изучению дисциплин и материалов, может проходить тестирования и выполнять практические задания, предусмотренные курсом.

Перечисленные выше подсистемы должны решать следующие задачи:

1. Подсистема администрирования содержит функционал для управления пользователями и контроля базы данных программы.
2. Подсистема преподавателя предоставляет функционал управления дисциплинами и лекциями, создания тестовых и практических заданий, контроля успеваемости студентов и создания отчетности.
3. Подсистема студента позволяет изучать лекции, выполнять практические и тестовые задания, получать статистику успеваемости по конкретному студенту (пользователю).

## 2.2 Функциональная структура системы

Информационная обучающая система предоставляет платформу для создания и изучения обучающего курса по дисциплине «Типы информационных систем и их администрирование».

Обучающие системы – автоматизированные информационные системы на основе информационных технологий. Каждый обучающий курс включает в себя лекционный материал для изучения, промежуточные тестирования для отслеживания успеваемости студентов, практические задания и дополнительные материалы для дополнительного (необязательного) изучения.

Обобщенная схема обучающего курса представлена на рисунке 2.1

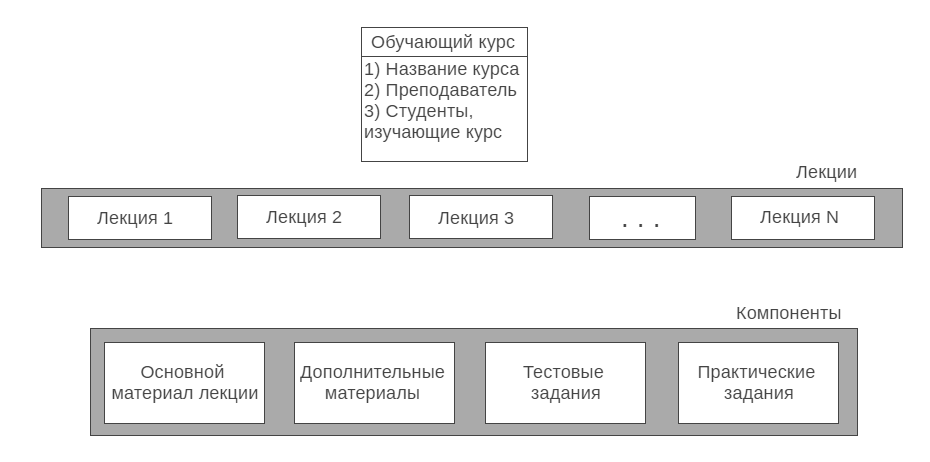


Рисунок 2.1 – Схема обучающего курса

## 2.3 Описание объекта автоматизации

Объектом автоматизации является процесс образования в рамках определенной дисциплины. Данный проект разработан с помощью методологии IDEF0. IDEF0 – методология функционального моделирования и графическая нотация, предназначенная для формализации и описания бизнес-процессов. Отличительной особенностью IDEF0 является её акцент на соподчинённость объектов. В IDEF0 рассматриваются логические отношения между работами, а не их временная последовательность (поток работ).

Контекстная диаграмма представляет требования к системе на самом верхнем уровне – уровне взаимодействия с окружением.

На контекстной диаграмме специфицируемая система представляется в виде одного единственного процесса, связанного с внешними сущностями потоками данных. Контекстная диаграмма процесса обучения представлена на рисунке 2.2.

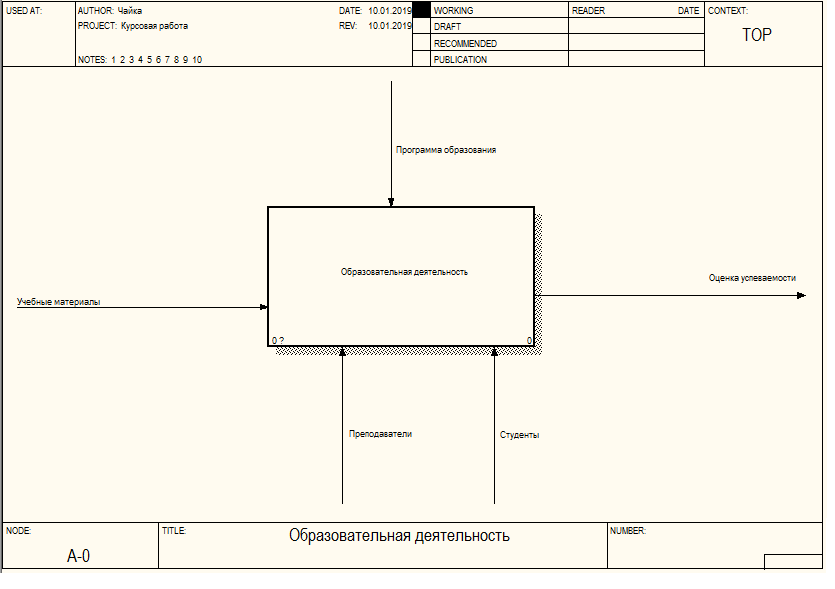


Рисунок 2.2 – Контекстная диаграмма А-0

Работы и стрелки контекстной диаграммы А-0 находятся в таблицах 2.1 и 2.2.

Таблица 2.1 – Стрелки контекстной диаграммы А-0

|  |  |
| --- | --- |
| *Arrow Name* | *Arrow Definition* |
| Преподаватель | Преподаватель, преподающий данный обучающий курс |
| Студенты | Студенты, проходящие обучения по данному курсу |
| Учебные материалы | Рекомендуемые учебники и пособия |
| Программа обучения | Требования к обучению по данной программе |
| Оценка успеваемости | Результаты прохождения студентами итоговых тестов и степень усвоения материала |

Таблица 2.2 – Работы контекстной диаграммы А-0

|  |  |
| --- | --- |
| *Activity Name* | *Definition* |
| Изучение теории | Изучение студентами теоретической информации |
| Практическая работа | Выполнение практических и лабораторных работ |
| Оценивание успеваемости | Выполнение тестов и подведение статистики успеваемости |

Диаграммы, которые описывают каждый фрагмент контекстной диаграммы, называются диаграммами декомпозиции. Диаграмма декомпозиции образовательной деятельности А0 представлена на рисунке 2.3.

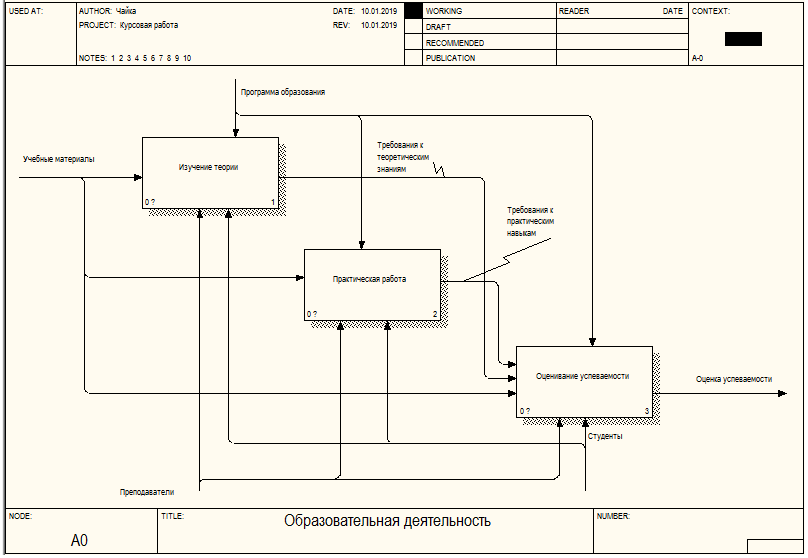


Рисунок 2.3 – Диаграмма декомпозиции А0

Внутренние стрелки диаграммы декомпозиции А0 находятся в таблице 2.3.

Таблица 2.3 – Внутренние стрелки диаграммы декомпозиции А0

|  |  |
| --- | --- |
| *Arrow Name* | Блок начала стрелки |
| Требования к практическим навыкам | Требования к практическим навыкам |
| Требования к теоретическим знаниям | Требования к теоретическим знаниям |

Диаграмма декомпозиции образовательной деятельности А1 представлена на рисунке 2.4.

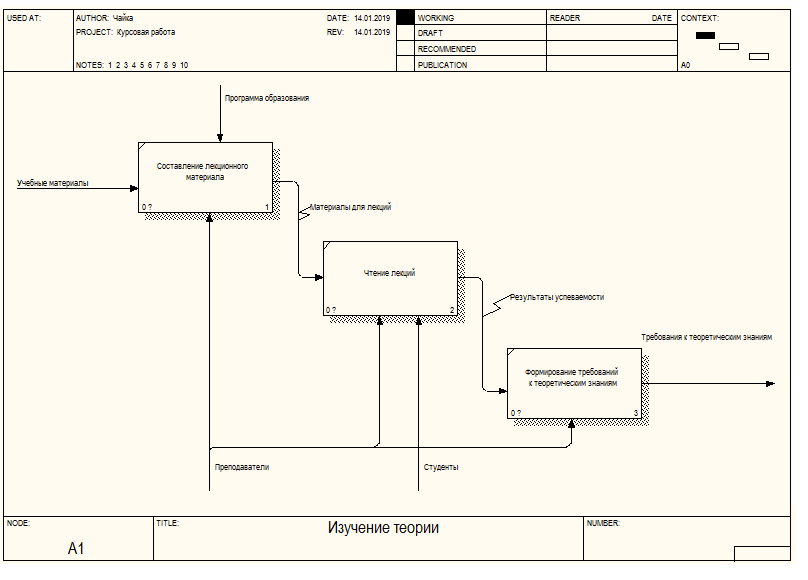


Рисунок 2.4 – Диаграмма декомпозиции А1

Диаграмма декомпозиции образовательной деятельности А2 представлена на рисунке 2.5.

Стрелки диаграммы декомпозиции А1 находятся в таблице 2.4.

Таблица 2.4 – Стрелки диаграммы декомпозиции А1

|  |  |
| --- | --- |
| *Arrow Name* | Блок начала стрелки |
| Материалы для лекций | Готовый лекционный материал в электронной форме |
| Результаты успеваемости | Оценка посещаемости и работы на лекциях |

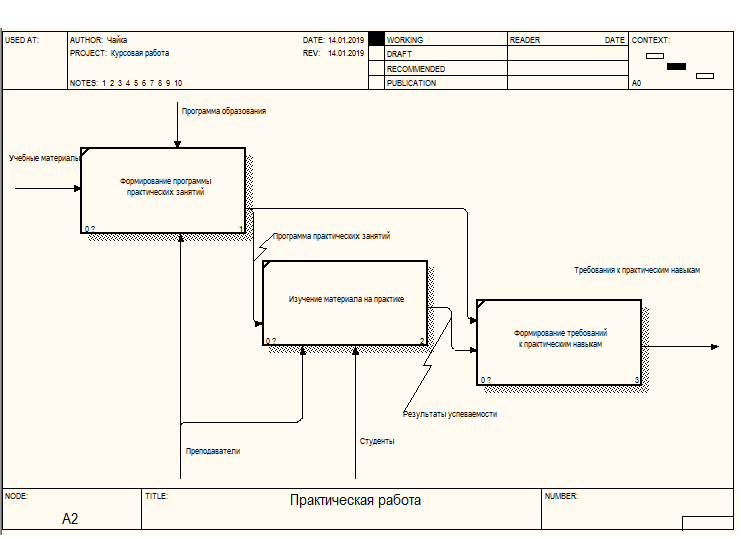


Рисунок 2.5 – Диаграмма декомпозиции А2

Работы диаграммы декомпозиции А2 находятся в таблицах 2.5.

Таблица 2.5 – Работы диаграммы декомпозиции А2

|  |  |
| --- | --- |
| Имя работы | Определение |
| Формирование материала для практических занятий | Разработка на основе учебных материалов и требований к специальности практических и лабораторных работ |
| Изучение материала на практике | Выполнение практических и лабораторных работ |
| Формирование требований к практическим навыкам | Формирование требований к практическим навыкам студента |

Диаграмма декомпозиции образовательной деятельности А3 представлена на рисунке 2.6.

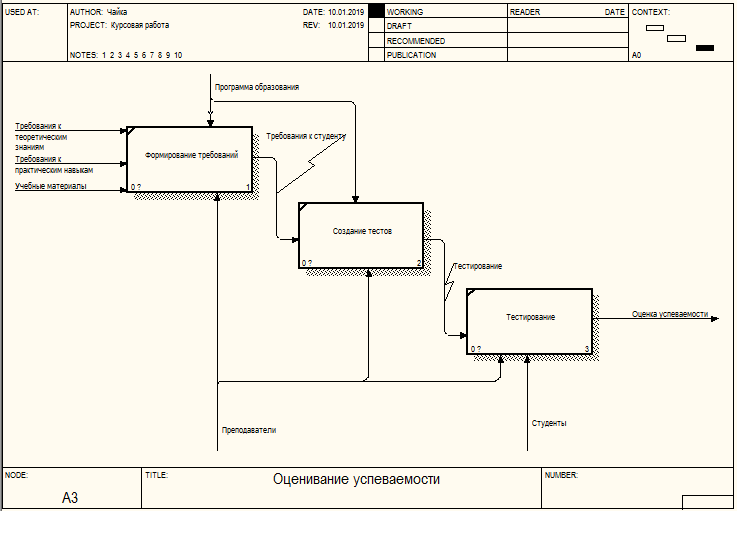


Рисунок 2.6 – Диаграмма декомпозиции А3

Диаграммы потоков данных (DFD) – используются для описания документооборота и обработки информации. Модель информационной системы определятся как иерархия диаграмм потоков данных, описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи пользователю.

Диаграммы верхних уровней иерархии (контекстные диаграммы) определяют основные процессы или подсистемы ИС с внешними входами и выходами. Они детализируются при помощи диаграмм нижнего уровня. Такая декомпозиция продолжается до тех пор, пока не будет достигнут такой уровень декомпозиции, на котором процессы становятся элементарными и детализировать их далее невозможно. DFD можно использовать как дополнение к модели IDEF0 для более наглядного отображения текущих операций документооборота. Диаграмма потоков данных «просмотр подходящих клиенту предложений» представлена на рисунке 2.7.

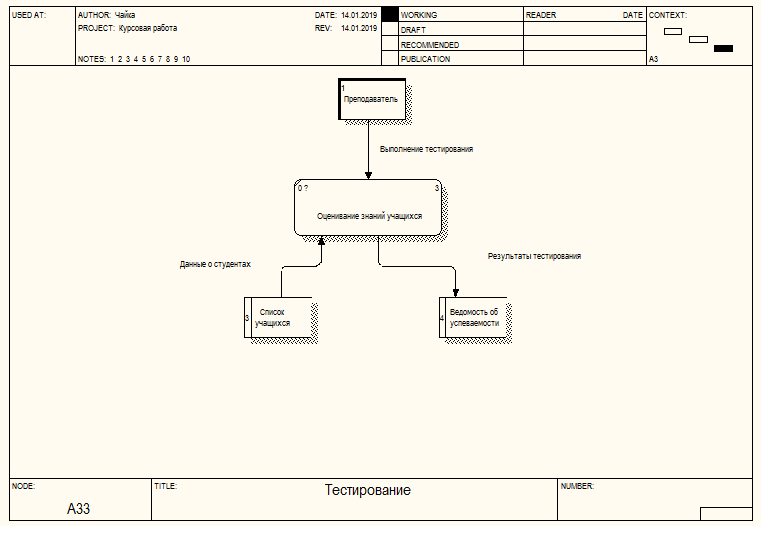


Рисунок 2.7 – Диаграмма потоков данных А33

Объекты и потоки диаграммы потоков данных А33 находятся в таблицах 2.6 и 2.7 .

Таблица 2.6 – Объекты DFD-диаграммы А3

|  |  |
| --- | --- |
| Вид объекта | Имя объекта |
| Внешнаяя сущность | Преподаватель |
| Работа | Оценивание знаний учащихся |
| Хранилище данных | Список учащихся |
| Хранилище данных | БД успеваемости |

Таблица 2.7 - Потоки данных DFD-диаграммы А3

|  |  |  |
| --- | --- | --- |
| Имя стрелки | Начало стрлки | Конец стрелки |
| Выполнение тестирования | Преподаватель | Оценивание знаний учащихся |
| Данные о студентах | Список учащихся | Оценивание знаний учащихся |
| Результаты тестирования | Оценивание знаний учащихся | БД успеваемости |

Диаграмма дерева узлов показывает иерархию работ в модели и позволяет рассмотреть всю модель целиком, но не показывает взаимосвязи между работами. Диаграмма дерева узлов представлена на рисунке 2.8.

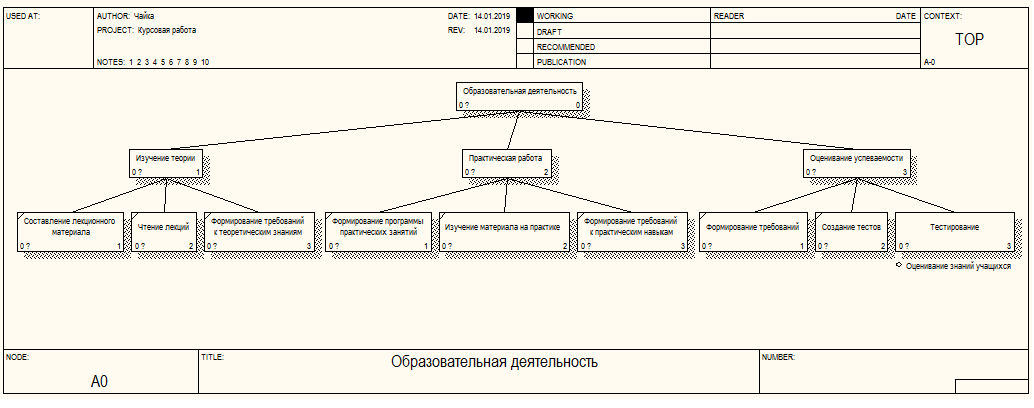


Рисунок 2.8 – Диаграмма дерева узлов

## 2.4 Проект логической реализации информационной обучающей системы

Чтобы описать АОС на концептуальном уровне рассмотрим диаграммы прецедентов представленные ниже.

Диаграмма прецедентов –диаграмма, отражающая отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Прецедент – возможность моделируемой системы (часть её функциональности), благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Варианты использования обычно применяются для спецификации внешних требований к системе.

Рассмотрим диаграмму прецедентов для неавторизированного пользователя (Рис.2.9)

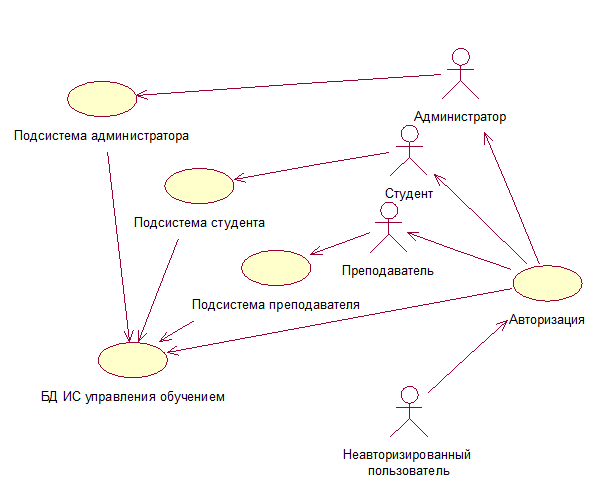


Рис.2.9– Диаграмма прецедентов для неавторизированного пользователя

Единственная прецедент доступный неавторизированному пользователю – это авторизация, происходящая при помощи логина и пароля. После нее в зависимости от типа пользователя (студент, преподаватель, администратор) пользователю предоставляется доступ к соответствующей подсистеме.

Далее рассмотрим диаграммы прецедентов соответственно для студента (Рис.2.10), преподавателя (Рис.2.11) и администратора (Рис.2.12)

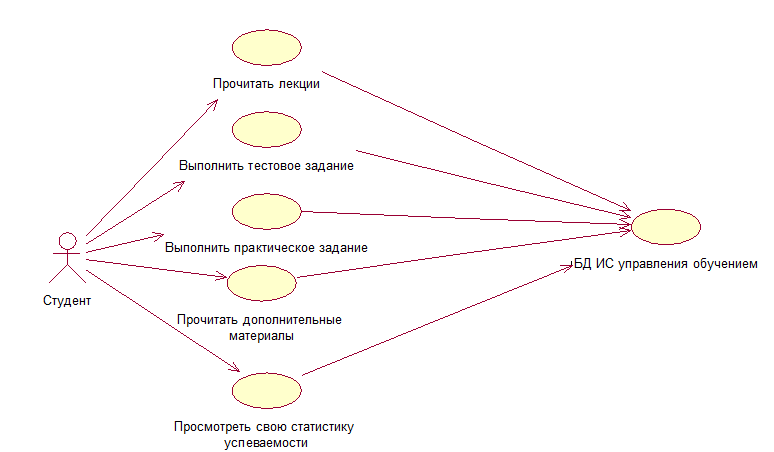


Рис.2.10 – Диаграмма прецедентов для студента

Студенту доступны следующие прецеденты: Прочитать материалы лекций, выполнить прилагающиеся к ним тестовые и практические задания, прочитать дополнительные материалы, предложенные преподавателем и изучить свою статистику успеваемости.

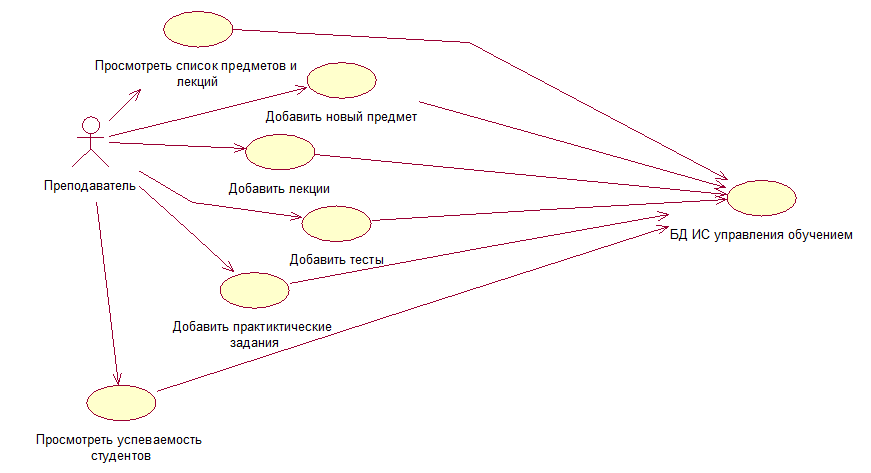


Рис.2.11 – Диаграмма прецедентов для преподавателя

Преподавателю доступны следующие прецеденты: Просмотр относящихся к нему предметов и входящих в них лекций, добавление новых предметов, лекций, тестов и практических заданий, просмотр успеваемости всех студентов.

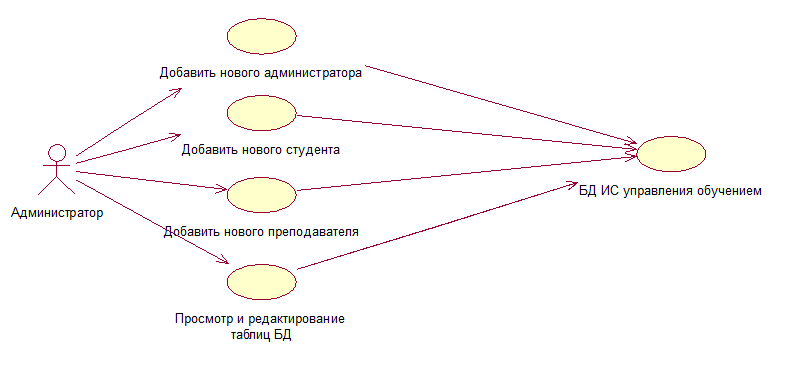


Рис.2.12 – Диаграмма прецедентов для администратора

Администратору доступны следующие прецеденты: Добавление новых пользователей (Студента, Преподавателя, Администратора), просмотр и редактирование таблиц базы данных.

Остановимся более подробно на основных процессах этих диаграмм:

На рис.2.13 находится диаграмма последовательностей добавления нового пользователя администратором (аналогична для любого типа пользователей).



Рис.2.13 – Добавление нового пользователя администратором.

1) Администратор заполняет и отправляет запрос на добавление нового пользователя.

2) Модуль администратора проверяет введенные данные. Если все введено верно происходит переход к пункту 3, иначе отображается сообщение об ошибке.

3) Модуль администратора отправляет запрос к базе данных для того что бы проверить существование в БД пользователя с таким же логином. Если дубликат не обнаружен продолжается выполнение процесса.

4) Модуль администратора отправляет все данные в таблицы БД

5) БД возвращает в качестве отклика количество обработанных строк.

6) Если число обработанных строк равно 1 – администратору выводится сообщение об успешном добавлении нового пользователя.

На рис.2.6 находится диаграмма последовательностей прохождения студентом теста

1. Студент в интерфейсе программы выбирает необходимый предмет, открывает лекцию и приступает к выполнению относящегося к ней теста.
2. Подсистема студента делает запрос о информации к необходимому тесту в БД
3. БД отправляет искомую информацию в подсистему студента
4. Подсистема студента выводит тест студенту.
5. Студент отвечает на вопросы теста.
6. Подсистема студента производит проверку ответов студента
7. Вносит информацию в БД
8. В случае успешного выполнения запроса подсистема студента получает отклик от БД.
9. Подсистема студента выводит результат студенту.

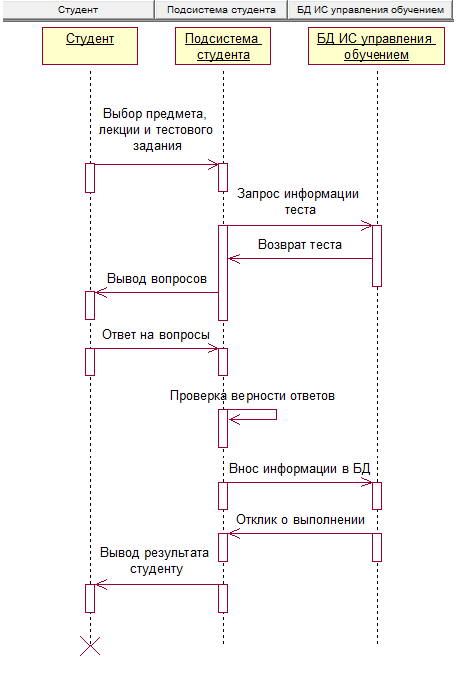


Рис.2.14 – Прохождение теста студентом

## 2.5 Разработка реляционной модели базы данных обучающей системы

Реляционная модель базы данных – это набор взаимосвязанных отношений (таблиц), каждая часть которой хранит информацию о объекте какого-либо типа. Строка таблицы содержит информацию о объекте, а каждый столбец содержит какую-либо характеристику этого объекта и имеет определенный тип данных

Схема данных базы данных обучающей системы представлена на Рис.2.15.

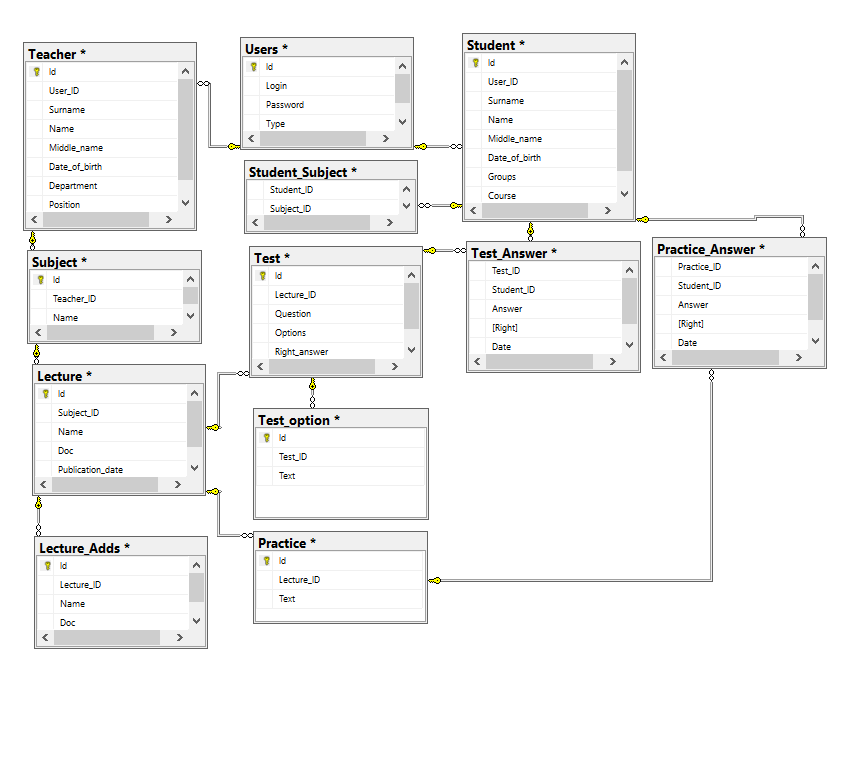


Рис.2.15 – Схема данных базы данных в СУБД MS SQLServer.

Для каждого отношения реляционной модели базы данных определим типы данных полей (таблицы 2.9 – 2.21)

Основной таблицей базы данных является таблица 2.9, используемая для авторизации пользователей. Любая дальнейшая работа с базой данных выполняется посредством обращения к этой таблице и уже после выделяются связанные с указанным пользователем данные. В таблице хранятся только логин и пароль пользователя, его уникальный идентификационный номер и тип, которых в системе предусмотрено три.

Таблица 2.9 – Users

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор |
| Login | Nvarchar(50) | Нет | Логин пользователя, 50 символов |
| Password | Nvarchar(50) | Нет | Пароль пользователя, 50 символов |
| Type | Char(7) | Нет | Тип пользователя, 7 символов |

После авторизации программа определяет тип пользователя и предоставляет относящиеся к нему материалы. Для хранения персонализированной информации о пользователях с типом Учитель создана специальная таблица 2.10. Ключевым полем в ней является уникальный идентификатор учителя, с которым связаны дальнейшие таблицы. Остальные столбцы таблицы хранят обычную персональную информацию: ФИО, год рождения, должность и кафедра.

Таблица 2.10 – Teacher

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип  данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор учителя |
| User\_ID | Int | Нет | Ссылка на идентификатор пользователя |
| Surname | Nchar(40) | Нет | Фамилия |
| Name | Nchar(40) | Нет | Имя |
| Middle\_name | Nchar(40) | Да | Отчество |
| Date\_of\_birth | Date | Да | Дата рождения |
| Departament | Nchar(150) | Да | Кафедра |
| Position | Nchar(150) | Да | Должность |

Таблица 2.11 так же создана для хранения личной информации другого типа пользователей. В ней хранятся данные студентов, пользующихся обучающей системой. По уникальному идентификатору студента определяется его успеваемость и хранятся результаты выполнения тестовых и практических заданий.

Таблица 2.11 – Student

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор студента |
| User\_ID | Int | Нет | Ссылка на идентификатор пользователя |
| Surname | Nchar(40) | Нет | Фамилия |
| Name | Nchar(40) | Нет | Имя |
| Middle\_name | Nchar(40) | Да | Отчество |
| Date\_of\_birth | Date | Да | Дата рождения |
| Group | Nchar(40) | Да | Группа |
| Course | Int | Да | Курс обучения |

Следующей ключевой таблицей базы данных является таблица 2.12. На основе ее содержимого строится вся структура обучающих курсов. Она хранит идентификаторы предметов, изучаемых в информационной обучающей системе, ссылки на преподавателей, которые ведут эти предметы и названия самих предметов. Непосредственно на нее ссылается таблица с лекциями.

Таблица 2.12 – Subject

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор предмета |
| Teacher\_ID | Int | Нет | Ссылка на идентификатор преподавателя |
| Name | NCHAR (150) | Нет | Название предмета |

Больше всего информации хранится в таблице 2.13. Это лекции для каждого из предметов. Уникальный идентификатор каждой лекции используется для присоединения дополнительных материалов, тестовых и практических заданий. Сами лекции хранятся в виде ссылок на документы с текстом.

Таблица 2.13 – Lecture

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор лекции |
| Subject\_ID | Int | Нет | Ссылка на идентификатор предмета |
| Name | NCHAR (100) | Нет | Имя лекции |
| Doc | Varchar (MAX) | Нет | Адрес файла лекции |
| Publication\_date | DATE | Да | Дата публикации |

Для расширения возможностей преподавателя ему предоставлена возможность прилагать дополнительные материалы к каждой лекции. Это могут быть методические указания, книги или лабораторные работы. Все это хранится в таблице 2.14 в таком же формате, как хранятся лекции – в виде ссылок на файлы.

Таблица 2.14 – Lecture\_Adds

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор дополнительного материала |
| Lecture\_ID | Int | Нет | Ссылка на идентификатор лекции |
| Name | NCHAR (100) | Нет | Имя дополнительного материала |
| Doc | Varchar (MAX) | Нет | Адрес файла дополнительного материала |

Для хранения тестовых заданий каждой лекции используется таблица 2.15. в ней каждый кортеж данных представляет собой отдельный тестовый вопрос, которые соединяются вместе через общую ссылку на лекцию.

Таблица 2.15 – Test

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| 1 | 2 | 3 | 4 |
| Id | Int | Нет | Уникальный идентификатор тестового задания |
| Lecture\_ID | Int | Нет | Ссылка на идентификатор лекции |

Продолжение таблицы 2.15

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Question | VARCHAR (MAX) | Нет | Текст вопроса |
| Options | BIT | Нет | Наличие вариантов ответа (да/нет) |
| Right\_answer | Int | Да | Номер правильного ответа |

Варианты ответов на тестовые вопросы хранятся в отдельной таблице Test\_Option (Таблица 2.16). Они имеют собственный индекс, однако, в приложение загружаются посредством общей ссылки на тестовый вопрос.

Таблица 2.16 – Test\_Option

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор варианта ответа |
| Test\_ID | Int | Нет | Ссылка на вопрос теста |
| Text | NChar(150) | Нет | Текст варианта ответа |

При изучении дисциплины студенты выполняют тестовые задания. Для хранения их результатов создана дополнительная таблица 2.17–Test\_Answer. В ней хранятся, выбранные студентами, варианты ответа, правильность выбранного ответа и дата выполнения задания.

Таблица 2.17 – Test\_Answer

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор ответа |
| Test\_ID | Int | Нет | Ссылка на вопрос теста |
| Student\_ID | Int | Нет | Ссылка на студента |
| Answer | Int | Да | Вариант ответа студента |
| Right | BIT | Нет | Правильность ответа |
| Date | Date | ДА | Дата выполнения задания |

Помимо тестовых заданий информационная обучающая система предусматривает практические задания, требующие развернутого ответа. Для хранения информации о таких вопросах были созданы следующие таблицы:

Таблица 2.18 – Practice предназначена для хранения непосредственно практических заданий и состоит из трех столбцов.

Таблица 2.18 – Practice

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор практического задания |
| Lecture\_ID | Int | Нет | Ссылка на лекцию |
| Text | Varchar(MAX) | Нет | Текст вопроса |

Таблица 2.19 – Practice\_Answer выполняет функцию, аналогичную таблице Test\_Answer– Она хранит ответы студентов на практические задания. Поле Right заполняется преподавателем или администратором вручную, после проверки ответа студента.

Таблица 2.19 – Practice\_Answer

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| 1 | 2 | 3 | 4 |
| Id | Int | Нет | Уникальный идентификатор ответа на практическое задание |
| Practice\_ID | Int | Нет | Ссылка на вопрос практического задания |
| Student\_ID | Int | Нет | Ссылка на студента |

Продолжение таблицы 2.19

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| Answer | VARCHAR (MAX) | Да | Ответ студента |
| Right | BIT | Нет | Верность ответа студента |
| DATE | Date | Нет | Дата выполнения задания |

Помимо основного лекционного материала в системе существует дополнительные материалы, хранящиеся подобным образом в отдельной таблице.

Таблица 2.20 – Lecture\_Adds в отличии от основной лекционной таблицы имеет привязку ключевым полем к самим лекциям.

Таблица 2.20 – Lecture\_Adds

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Id | Int | Нет | Уникальный идентификатор дополнительного материала |
| Lecture\_ID | Int | Нет | Ссылка на лекцию |
| Name | Nchar(100) | Нет | Имя дополнительного материала |
| Doc | VARCHAR (MAX) | Нет | Ссылка на файл |

Последней по порядку, но не последней по важности идет таблица для назначения студентам предметов на изучение. Таблица 2.21 представляет собой промежуточное звено для организации связи многие-ко-многим между таблицами Студенты и Предметы.

Таблица 2.21 – Students\_Subjects

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Допустимость NULL | Комментарий |
| Student\_Id | Int | Нет | Ссылка на студента |
| Subject\_ID | Int | Нет | Ссылка на предмет |

# 3 Разработка программного модуля обучающей информационной системы

## 3.1 Обоснование выбора языка и среды программирования

Для разработки обучающей информационной системы по дисциплине «Типы информационных систем и их администрирование» была выбрана среда разработки MicrosoftVisualStudio 2017. Она обладает широким функционалом, который существенно облегчает разработку больших проектов и интеграцию в них баз данных.

VisualStudioдоступна для студентов, бесплатно в рамках программы MicrosoftImagine, действующей с 2008 г и доступной боле чем в 80 странах.

Основными преимуществами VS2017 является большое количество элементов для создания визуального интерактивного интерфейса, удобный редактор и отладчик приложения. Работа с базами данных облегчена и интегрирована в VisualStudio. Большую часть взаимодействий можно настроить при помощи визуального интерфейса, не углубляясь в программирование.

Среди языков программирования был выбран С#, для него существует большое количество библиотек в среде VisualStudio, он является объектно-ориентированным языком программирования и представляет широкий функционал разработчику. В данном проекте он существенно облегчил разработку, за счет использования перегруженных функций, классов и прямого выполнения запросов к базе данных.

## 3.2 Обоснование выбора системы управления базами данных

Система SQL Server 2008 отталкивается от концепции платформы данных Майкрософт: она упрощает управление любыми данными в любом месте и в любой момент времени. Она позволяет хранить в базах данных информацию, полученную из структурированных, полуструктурированных и неструктурированных источников, таких как изображения и музыка. В SQL Server 2008 имеется большой набор интегрированных служб, расширяющих возможности использования данных: можно составлять запросы, выполнять поиск, проводить синхронизацию, делать отчеты, анализировать данные. Все данные хранятся на основных серверах, входящих в состав центра обработки данных. К ним осуществляется доступ с настольных компьютеров и мобильных устройств. Таким образом, осуществляется контроль над данными вне зависимости от того, где они хранятся.

Система SQL Server 2008 позволяет обращаться к данным из любого приложения, разработанного с применением технологий Microsoft .NET и VisualStudio. Сотрудники, отвечающие за сбор и анализ информации, могут работать с данными, не покидая привычных приложений, которыми они пользуются каждый день, например приложений выпуска 2007 системы MicrosoftOffice. SQL Server 2008 позволяет создать надежную, производительную, интеллектуальную платформу, отвечающую всем требованиям по работе с данными.

Серьезным аргументом в пользу MSSQLServer 2008 также является то, что он развернут на серверах ФГБОУ ВО ВГЛТУ, где предполагается использование программы.

## 3.3 Работа обучающей информационной системы

В информационнойобучающей системе предусмотрены три функциональных модуля. Доступ к ним предоставляется посредством единого окна авторизации (Рисунок – 3.1).

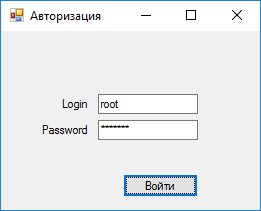


Рисунок 3.1 – Окно авторизации пользователя

Первоначальная настройка обучающей системы выполняется администратором. Необходимо зарегистрировать аккаунты для преподавателей, студентов и других администраторов. Это выполняется в форме регистрации пользователей (Рисунок 3.2), которая является частью модуля администрирования информационной обучающей системы.

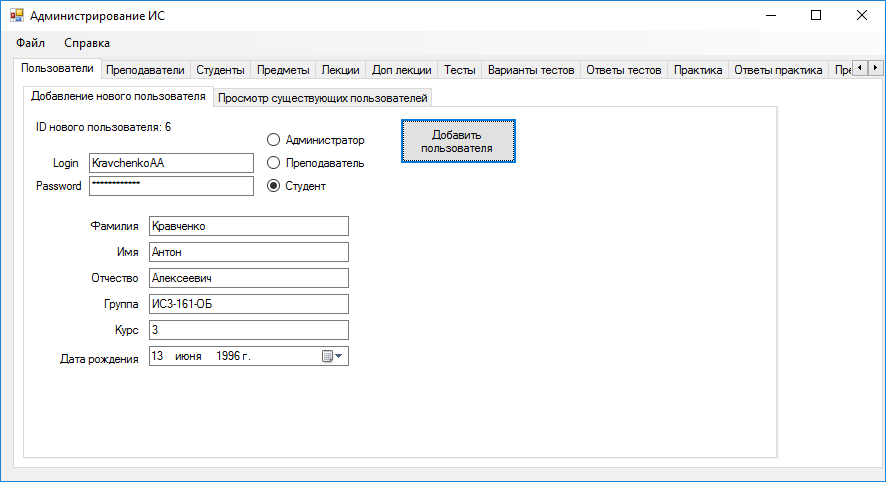


Рисунок 3.2 – Форма регистрации нового пользователя

Дополнительно модуль администрирования предоставляет возможность прямого управления содержимым таблиц базы данных, для непосредственной корректировки ошибок. Окно для работы с таблицей Студенты представлено на рисунке 3.3

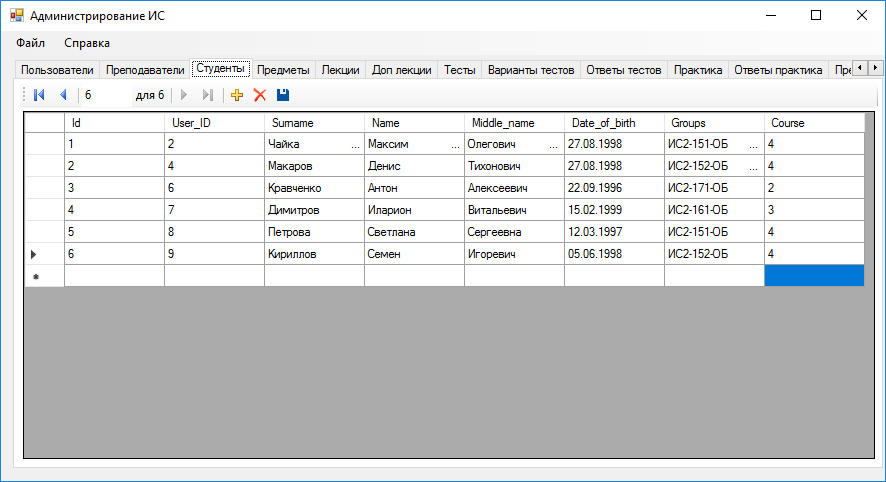
****

Рисунок 3.3 – Форма управления базой данных

Дальнейшая подготовка к работе и непосредственная загрузка обучающих курсов в информационную систему производится каждым преподавателем отдельно при помощи функционала модуля преподавателя.

В первую очередь преподавателю предоставляется доступ к форме управления предметами (Рисунок 3.4). С ее помощью возможно просмотреть список курсов, подотчетных преподавателю, добавить или удалить предметы.

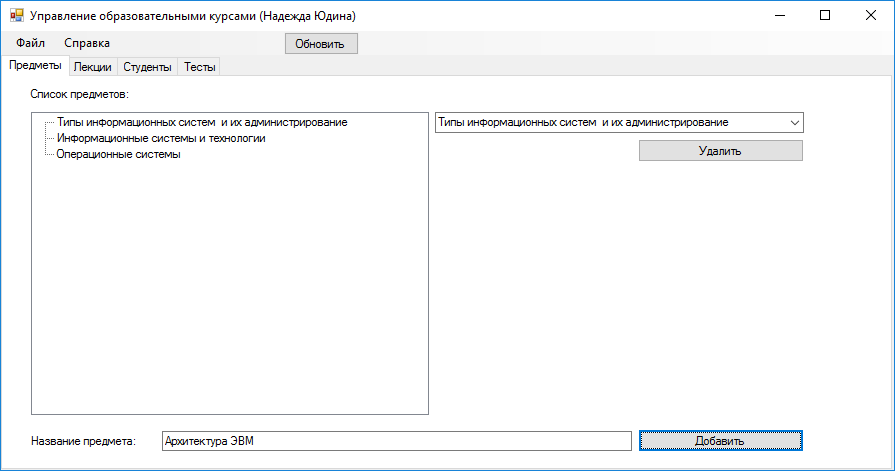
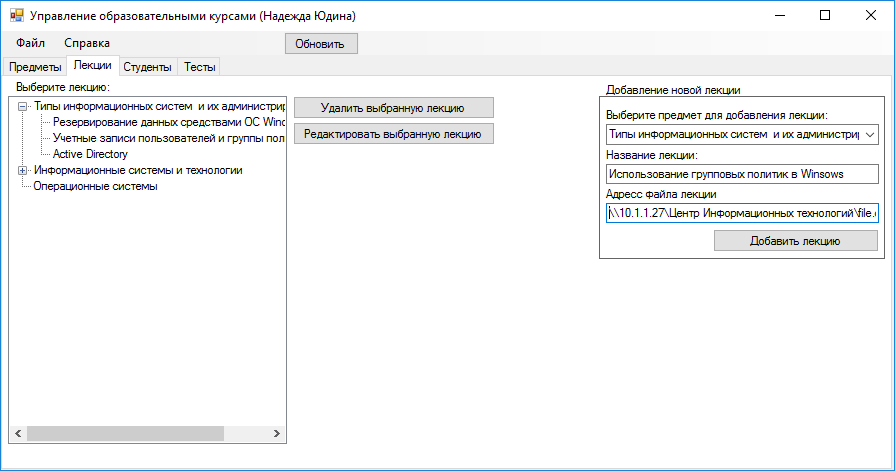


Рисунок 3.4 – Форма управления предметами

После регистрации в программном обеспечении всех необходимых предметов преподаватель может приступить к следующему шагу – наполнению курса.

Для управления лекционным материалом курса предусмотрена отдельная форма. В ней предоставляется список существующих лекций, распределенных по предметам и интерфейс для удаления, редактирования и добавления новых лекций. Данная форма представлена на рисунке 3.5

 Рисунок 3.5 – Форма управления лекциями

Функционал для просмотра студентов (Рисунок 3.6) вынесен в отдельную форму ввиду его возможного расширения. На текущий момент он предоставляет возможность просмотреть разделенный на группы список студентов, записанных на обучающие курсы которые ведет преподаватель.

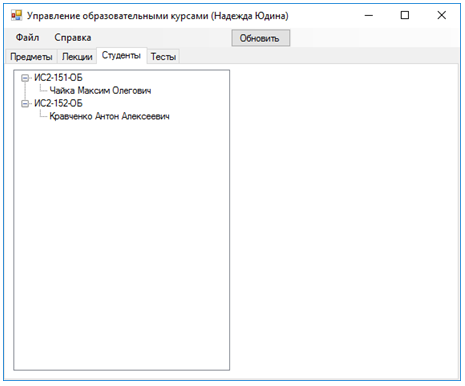


Рисунок 3.6 – Форма управления обучающимися

На текущий момент последней частью функционала этого модуля является возможность создания тестовых заданий к лекциям при помощи наглядного интерфейса, представленного на рисунке 3.7

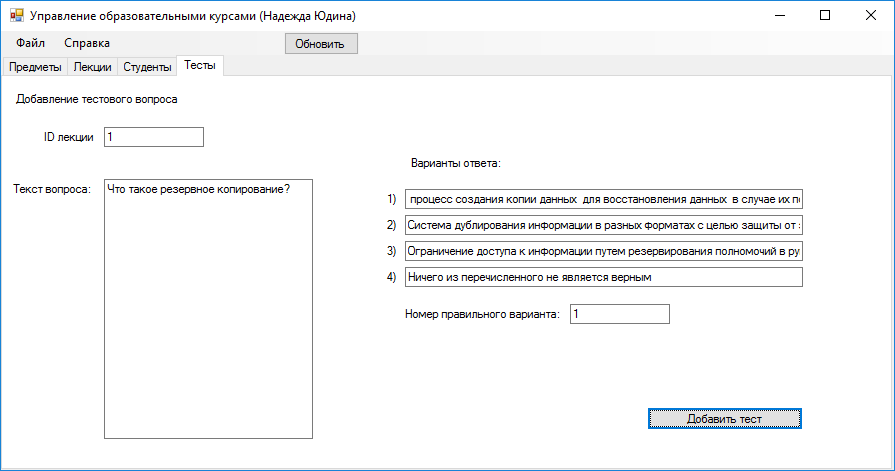


Рисунок 3.7 – Форма создания тестовых вопросов

После заполнения преподавателями информационная обучающая система может считаться настроенной и готовой к использованию.

Основную часть пользователей составляют студенты, использующие систему для дистанционного и дополнительного образования. Функционал, предоставляемый модулем для работы студента (Рисунок 3.8), не отличается разнообразием и призван, в первую очередь, решить основные задачи, стоящие перед студентом. К таким задачам относятся изучение теоретического материала и возможность тестовой оценки знаний студента.

В виду того что лекции хранятся в виде отдельных документов на файловом сервере при нажатии на кнопку “Открыть лекцию” произойдет перенаправление пользователя на файл лекции, который будет открыт в текстовом редакторе, установленном на компьютере пользователя.

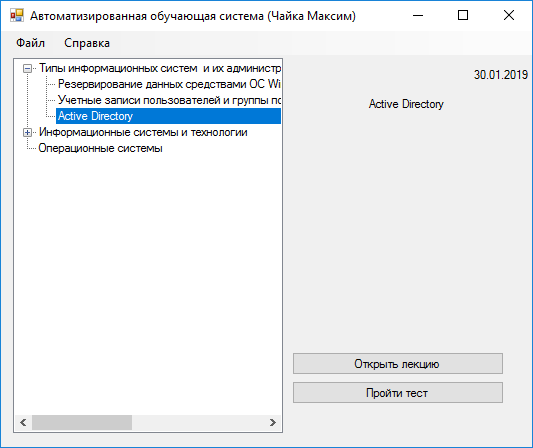


Рисунок 3.8 – Интерфейс модуля студента

Для проведения тестового оценивания знаний студента ему необходимо выбрать лекцию, и нажать на кнопку “Пройти тест”. После чего он будет перенаправлен в окно с вопросами теста (Рисунок 3.9), последовательно отвечая на которые он сразу же будет узнавать верность своих ответов и совершенные ошибки. После выполнения теста, результат которого автоматически заносится в базу данных, студент будет возвращен в главное меню модуля программы.

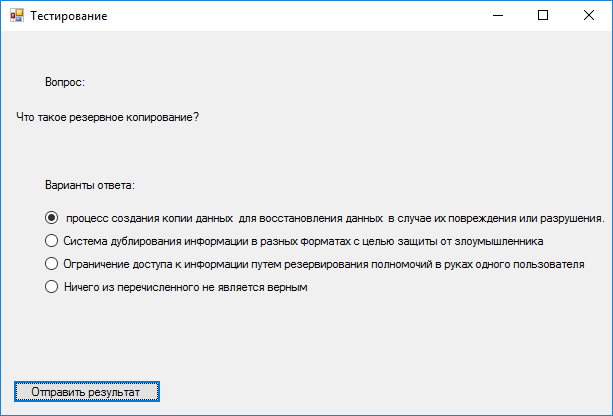


Рисунок 3.9 – Интерфейс ответа на тестовые вопросы

# ЗАКЛЮЧЕНИЕ

В выпускной квалификационной работе был разработан модуль информационной обучающей системы по дисциплине «Типы информационных систем и их администрирование». Для этого были решены следующие задачи:

1. Проанализированы принципы построения информационных обучающих систем, выделены основные особенности и преимущества такого подхода к обучению. Собрана информация по международным стандартам, регламентирующим ИОС.
2. Разработаны логическая и функциональная модели информационной системы, описывающие ее основной функционал и паттерны поведения.
3. Разработана информационная обучающая система по дисциплине «Типы информационных систем и их администрирование»

# СПИСОК ЛИТЕРАТУРЫ

1. Башмаков, М. И. Информационная среда обучения / М. И.Башмаков, С. Н. Поздняков, Н. А. Резник. – Спб.: СВЕТ, 1997. – 400с.
2. Изергин, Н. Д. Разработка электронных учебных изданий. Создание и использование информационных средств обучения: учеб.-практ. пособие / Н. Д. Изергин [и др.]. – Коломна : 2005.– 160 с.
3. Кириллов, В. В. Введение в реляционные базы данных/ В.В. Кириллов, Г. Ю. Громов. – СПб. : БХВ-Петербург, 2012. – 464 c.
4. Малыхина, М. П. Базы данных: основы, проектирование, использование / М. П. Малыхина. – СПб. : БХВ-Петербург, 2006. – 512 с.
5. Полат, Е. С. Новые педагогические и информационные технологии в системе образования : учеб.пособие для студ. высш. учеб. заведений / Е. С. Полатина [и др.]. – М.: Академия, 2008. – 272 с.
6. Интуит – национальный открытый университет: Диаграмма последовательностей [Электронный ресурс]. – Режим доступа: WorldWideWeb. URL:https://www.intuit.ru/studies/courses/1007/229/lecture/5954?page=3. –Загл. с экрана.
7. Лященко, Н.И. Диагностика образования и методика формирования структуры учебного курса в автоматизированной обучающей системе повышения квалификации специалистов / Н.И.Лященко // Современные проблемы науки и образования. – 2012. - №5.
8. Кривицкий, Б.Х. Обучающие компьютерные программы: психология разработки преподавателями обучающих курсов в АСО / Б.Х. Кривицкий // Образовательные технологии и общество. – 2007. – №3. – Том 10. – С.395-406.
9. Русских, Л.В. К вопросу о критериях качества высшего образования / Л.В. Русских // Вестник Южно-Уральского государственного университета. Серия: Социально-гуманитарные науки. – 2012. – №32. – С. 178-181.
10. Селиванцев, О.И. Информационная поддержка применения стандартов в области электронного обучения / О.И.Селиванцев, М.В.Сутягин // Открытое образование. – 2015. – №1. – С.50-54.
11. Малинин, Н.В. Модели оценки качества электронного образования / Н.В.Малинин // Преподаватель ХХI век. – 2014. – №3. – С.93-98.
12. Водолазкина, Н.А. Принципы разработки автоматизированной обучающей системы / Н.А.Водолазкина, А.Е.Закшевская // Международный научный журнал «инновационная наука». – 2015. - №6. – С.48-52.
13. Афанасьев, А.Н. Интеллектуальная обучающая система концептуальному проектированию автоматизированных систем / А.Н.Афанасьев, Н.Н. Войт // Известия Самарского научного центра Российской академии наук. – 2010. – №4-2. – С.465-468.
14. Бартасевич, И.Г. Система критериев качества обучения студентов в высшем учебном заведении / И.Г. Бартасевич // Вестник Астраханского государственного технического университета. – 2008. – №3.– С.217-220.
15. Мулик, А.Б. Разработка и определение эффективности метода качественной и количественной оценки потребностей человека /А.Б. Мулик, М.В.Постнова, Ю.А.Мулик, В.В.Новочадов //Российский медико-биологический вестник имени академика И.П. Павлова. – 2010. – №4. – С.1-8.
16. Шиянов, Б.А. Вероятностно - статистические методы количественной оценки рисков в системе регулирования неравновесными состояниями экономических систем / Б.А. Шиянов, О.В. Силютина, В.С. Неженец // Вестник ВГТУ. – 2010. – №8. – С.164-170.
17. Аль-Шаеби, Р.А.А. Методика автоматизированного управления процессом формирования индивидуализированных учебных планов / Р.А.А Аль-Шаеби // Известия Волгоградского государственного технического университета. – 2012. – №4 (91). – С.87-91.
18. Султанова, Т.А. Профессиональная образовательная среда: сущность, структура, критерии качества / Т.А. Султанова // Наука и современность. – 2011. – №8-1. – С.334-339.
19. Елькин, В.А. Совершенствование региональной системы анализа социально-экономического развития муниципальных образований / В.А. Елькин, Д.Ю. Хорохонов // Известия байкальского государственного университета. – 2011. – №1. – С.118-120.
20. Ясвин, В.А. Образовательная среда от моделирования к проектированию / В.А.Ясвин. –М.: Смысл, 2001. – 365с.
21. Тулупова Т. В. Автоматизация проектирования обучающих систем / Т.В.Тулупова // Науковедение. – 2013. – №5. – С.64.
22. Норенков, И. П. Технологии разделяемых единиц контента и удаленного доступа к учебным ресурсам [Электронный ресурс] / И. П. Норенков, А. М. Зимин // Научное издание МГТУ им. Н. Э .Баумана «Наука и образование» (материал доклада к конференции «Высокие технологии XXI века»). – Режим доступа: <http://technomag.bmstu.ru/doc/44146.html>
23. Волкова, А.В. Подход к формированию обобщенного критерия оценки качества для электронных учебно-методических комплексов / А. В. Волкова, В. В. Лавлинский, Т. В. Песецкая // Моделирование систем и процессов. – 2016. – Т.9. № 3. – С. 12-15.

# Приложение А Листинг программы

usingSystem;

usingSystem.Collections.Generic;

usingSystem.ComponentModel;

usingSystem.Data;

usingSystem.Drawing;

usingSystem.Linq;

usingSystem.Text;

usingSystem.Threading.Tasks;

usingSystem.Windows.Forms;

usingSystem.Data.SqlClient; //SQL Запросы

usingSystem.Diagnostics; // Прерывание текущего процесса для закрытия

namespace WindowsFormsApp1

{

public partial class Teacher\_Form : Form

{

//----------------------------------Область глобальных переменных------------------------------------

intTeacher\_User\_ID;

SqlConnectionsqlconn = new SqlConnection(@"Data Source=10.1.1.30;Initial Catalog=chaykamo;User ID=sa;Password=fvthbrflthmvj@2");

//Строка подключения БД

intid;

string surname;

string name;

stringmiddlename;

stringdate\_of\_birth;

stringdepartament;

string position;

stringlecture\_file\_address = ""; //Адресвыбраннойлекциивтривиев1

intselected\_subject\_id; //ДлявкладкиЛекции

intselected\_lecture\_id;// -//-

DataTable lectures = new DataTable(); //Таблицасовсемилекциями

DataTable subjects = new DataTable(); //Таблицасовсемипредметами

DataTable students = new DataTable(); //Таблицасовсемистудентами

//----------------------------------Область глобальных переменных------------------------------------

publicTeacher\_Form()

{

InitializeComponent();

}

publicTeacher\_Form(intteacher\_ID)

{

InitializeComponent();

Teacher\_User\_ID = teacher\_ID; //ПередачаUserIDИзформыавторизации

}

private void Teacher\_Form\_FormClosed(object sender, FormClosedEventArgs e)

{

//Закрытие приложения в случае Закрытия Формы.

Process.GetCurrentProcess().Kill();

}

//Обновлениевсего

voidUpdate\_All() {

//Добавлять новые функции в конец!

Tree\_Wiev1\_Update();

Tree\_Wiev2\_Update();

Tree\_Wiev3\_Update();

ComboBox1\_Update();

ComboBox2\_Update();

}

//Заполнениестудентовв tree wiev2 (students)

int Tree\_Wiev2\_Update() {//Заполнениестудентовв tree wiev2 (students)

//Предметы, лекции в tree wiev1.

/\*ВЫПОЛНЯЕТСЯ ОБНОВЛЕНИЕ ТАБЛИЦ ЛЕКЦИИ И ПРЕДМЕТЫ!!!\*/

int Tree\_Wiev1\_Update() { //Предметы, лекции и Доп задания.

// --------------------------------Загрузка Лекций и предметов в treeview.---------------------------------------

treeViewN.Nodes.Clear();

stringsql\_select\_subject = "Select Name, id FROM Subject Where Teacher\_ID =" + id.ToString() + ";";

stringsql\_select\_lectures = "Select \* FROM Lecture";

SqlDataAdaptersubject\_tableAdapter = new SqlDataAdapter(sql\_select\_subject, sqlconn);

//Выполнение запроса, результат в ДатаАдаптер

//DataTablesubjects = newDataTable();

subjects.Clear();

subject\_tableAdapter.Fill(subjects);

SqlDataAdapterlectures\_tableAdapter = new SqlDataAdapter(sql\_select\_lectures, sqlconn);

lectures.Clear();

//Выполнение запроса, результат в ДатаАдаптер

lectures\_tableAdapter.Fill(lectures);

int i = 0;

do

{

treeViewN.Nodes.Add(subjects.Rows[i].Field<String>(0).Trim()); //ДобавлениеПредмета

int j = 0;

do

{

if (lectures.Rows[j].Field<int>(1) == subjects.Rows[i].Field<int>(1)) //Важносравниватьзначениеидсзначениемизвведенногоужепредмета

{

treeViewN.Nodes[i].Nodes.Add(lectures.Rows[j].Field<String>(2).Trim()); // Добавлениелекции

}

j++;

} while (j <lectures.Rows.Count);

i++;

} while (i <subjects.Rows.Count);

//panel1.Visible = false; // Нужноли?

return 0;

}

//Загрузка Предметов в Комбобокс1 (Вкладка предметы)

int ComboBox1\_Update() {

comboBox1.Items.Clear();

int i = 0;

do

{

comboBox1.Items.Add(subjects.Rows[i].Field<String>(0).Trim()); //ДобавлениеПредмета

i++;

} while (i <subjects.Rows.Count);

return 0;

}

//Получение информации о учителе и загрузка базовой информации

private void Teacher\_Form\_Load(object sender, EventArgs e)

{

string query = "Select \* from Teacher Where User\_ID=" + Teacher\_User\_ID.ToString();

//Строка запроса к БД, использует поля текстбокс1 и текстбокс2

SqlDataAdaptersda = new SqlDataAdapter(query, sqlconn);

//Выполнение запроса, результат в ДатаАдаптер

DataTable dtb1 = newDataTable();

sda.Fill(dtb1);

//Создание ДатаТЕйбл и заполнение его данными из ДатаАдаптера

if (dtb1.Rows.Count == 1)

{

//Загрузка личных данных преподавателя из БД

id = dtb1.Rows[0].Field<int>(0);

surname = dtb1.Rows[0].Field<String>(2).TrimEnd();

name = dtb1.Rows[0].Field<String>(3).TrimEnd();

middlename = dtb1.Rows[0].Field<String>(4).TrimEnd();

date\_of\_birth = dtb1.Rows[0].Field<DateTime>(5).ToString();

departament = dtb1.Rows[0].Field<String>(6).TrimEnd();

position = dtb1.Rows[0].Field<String>(7).TrimEnd();

// MessageBox.Show("Добропожаловать, " + name.TrimEnd() + " " + middlename.TrimEnd() + "!");

this.Text = "Управление образовательными курсами (" + name.TrimEnd() + " " + surname.TrimEnd() + ")"; //Вывод имени преподавателя в имени формы

}

else{ MessageBox.Show("Ошибка Базы данных! Обратитесь к администратору");

this.Close();

};

//Обновление всех компонентов

Update\_All();

}

//Событие при выборе элемента в treeview Подгружает материалы лекции и тп.

private void treeView1\_AfterSelect(object sender, TreeViewEventArgs e)

{

//Событие при выборе элемента в treeview Подгружает материалы лекции и тп.

intsubject\_index=0, lecture\_index=0;

stringlecture\_name;

lecture\_index = treeViewN.SelectedNode.Index;

lecture\_name = treeViewN.SelectedNode.Text;

if (treeViewN.SelectedNode.Level> 0) {

int i = 0;

do {

if (subjects.Rows[i].Field<String>(0).Trim() == treeViewN.SelectedNode.Parent.Text) {

subject\_index = subjects.Rows[i].Field<int>(1);

}

i++;

} while (i <subjects.Rows.Count);

int j = 0;

do {

if (lectures.Rows[j].Field<int>(1) == subject\_index) { //Выборлекциипоиндексупредмета

if (lectures.Rows[j].Field<string>(2).Trim() == lecture\_name.Trim()) //Выборлекциипоимени

{

lecture\_file\_address = lectures.Rows[j].Field<string>(3);

selected\_lecture\_id = lectures.Rows[j].Field<int>(0);

// MessageBox.Show(lecture\_file\_address);

// Process.Start(lecture\_file\_address); //ОткрытиефайлавВорде

lecture\_index = j;

//panel1.Visible = true;

//label1.Text = lectures.Rows[j].Field<DateTime>(4).ToShortDateString();

//label2.Text = lectures.Rows[j].Field<String>(2).TrimEnd();

break;

}

}

j++;

} while (j <lectures.Rows.Count);

// label1.Text = lectures.Rows[subject\_index].Field<String>(lecture\_index);

}

}

private void button3\_Click(object sender, EventArgs e)

{

Update\_All();

}

//Кнопкадобавленияновогопредмета

private void button10\_Click(object sender, EventArgs e)

{

if (textBox1.Text !="") {

stringsql\_insert\_query\_new\_Subject = "INSERT INTO Subject (Teacher\_ID, Name) VALUES (" + id.ToString() + ", '" + textBox1.Text + "' )";

sqlconn.Open();

SqlCommandadd\_Subject = new SqlCommand(sql\_insert\_query\_new\_Subject, sqlconn);

intSubject\_query\_result = add\_Subject.ExecuteNonQuery();

if (Subject\_query\_result == 1)

{

MessageBox.Show("Успешнодобавлен 1 предмет");

}

elseMessageBox.Show("Ошибка!");

sqlconn.Close();

}

elseMessageBox.Show("Введите название предмета!");

}

//Кнопкаудаленияпредмета

private void button8\_Click(object sender, EventArgs e)

{

if (comboBox1.SelectedIndex>=0) {

//УДАЛЕНИЕПРЕДМЕТА

stringsql\_delete\_Subject = "DELETE FROM Subject WHERE id = " + subjects.Rows[comboBox1.SelectedIndex].Field<int>(1).ToString() + " ";

sqlconn.Open();

SqlCommanddelete\_Subject = new SqlCommand(sql\_delete\_Subject, sqlconn);

intSubject\_delete\_result = delete\_Subject.ExecuteNonQuery();

if (Subject\_delete\_result == 1)

{

MessageBox.Show("Успешноудален 1 предмет");

}

elseMessageBox.Show("Ошибка!");

sqlconn.Close();

Update\_All();

}

}

//Добавлениелекции

private void button11\_Click(object sender, EventArgs e)

{

if (textBox2.Text != "" && textBox3.Text != "" && comboBox2.SelectedIndex >=0)

{

DateTimecurDate = DateTime.Now;

stringsql\_insert\_query\_new\_Lecture = "INSERT INTO Lecture (Subject\_ID, Name, Doc, Publication\_date) VALUES (" + selected\_subject\_id.ToString() + ", '" + textBox2.Text + "', '" + textBox3.Text + "', '" + curDate.ToString() + "' )";

sqlconn.Open();

SqlCommandadd\_Lecture = new SqlCommand(sql\_insert\_query\_new\_Lecture, sqlconn);

intLecture\_query\_result = add\_Lecture.ExecuteNonQuery();

if (Lecture\_query\_result == 1)

{

MessageBox.Show("Успешнодобавлена 1 лекция");

}

elseMessageBox.Show("Ошибка!");

sqlconn.Close();

}

elseMessageBox.Show("Введите название предмета!");

}

//Редактированиелекции

private void button2\_Click(object sender, EventArgs e)

{

if (lecture\_file\_address != "")

{

Process.Start(lecture\_file\_address); //ОткрытиефайлавВорде

}

elseMessageBox.Show("Выберителекцию!");

}

//Удалениелекции

private void button1\_Click\_1(object sender, EventArgs e)

{

//УДАЛЕНИЕлекции

stringsql\_delete\_Lecture = "DELETE FROM Lecture WHERE id = " + selected\_lecture\_id.ToString() + " ";

sqlconn.Open();

SqlCommanddelete\_Lecture = new SqlCommand(sql\_delete\_Lecture, sqlconn);

intLecture\_delete\_result = delete\_Lecture.ExecuteNonQuery();

if (Lecture\_delete\_result == 1)

{

MessageBox.Show("Успешноудалена 1 лекция");

}

elseMessageBox.Show("Ошибка!");

sqlconn.Close();

Update\_All();

}

}

}