



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра Прикладной информатики

БАКАЛАВРСКАЯ РАБОТА

На тему

Разработка информационной системы учета успеваемости и посещаемости студентов

Исполнитель Филоненко Александр Сергеевич

(фамилия, имя, отчество)

Руководитель Д.т.н., профессор, декан факультета ИСиГТ

(ученая степень, ученое звание)

Истомин Евгений Петрович

(фамилия, имя, отчество)

«К защите допускаю»

Заведующий кафедрой

(подпись)

кандидат технических наук
Слесарева Людмила Сергеевна

«22» 06 2016 г.

Санкт-Петербург
2016



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра Прикладной информатики

БАКАЛАВРСКАЯ РАБОТА

На тему

Разработка информационной системы учета успеваемости и
посещаемости студентов

Исполнитель Филоненко Александр Сергеевич

(фамилия, имя, отчество)

Руководитель Д.т.н., профессор, декан факультета ИСиГТ

(ученая степень, ученое звание)

Истомин Евгений Петрович

(фамилия, имя, отчество)

«К защите допускаю»

Заведующий кафедрой

(подпись)

кандидат технических наук
Слесарева Людмила Сергеевна

«__» _____ 20__ г.

Санкт–Петербург
2016

ОГЛАВЛЕНИЕ

Введение	3
1. Описание предметной области	5
1.1 Объект автоматизации.....	5
1.2 Постановка задачи	12
1.3 Входные и выходные данные	13
1.4 Требования к автоматизированной информационной системе	15
2. Проектирование информационной системы.....	16
2.1 Методология проектирования базы данных	16
2.2 Выбор СУБД для информационной системы.....	19
2.3 Этапы разработки базы данных	20
2.4 Проектирование базы данных.....	21
3. Реализация информационной системы	27
3.1 Создание базы данных	27
3.2 Проверка работоспособности информационной системы	32
3.3 Параметрические запросы.	35
4. Экономический расчет разработки информационной системы	37
4.1 Трудоемкость разработки информационной системы	37
4.2 Экономическая эффективность внедрения программного продукта....	42
Заключение.....	48
Список источников литературы.....	50

Введение

В настоящее время к модернизации всех сфер жизнедеятельности общества привело развитие сферы информационных технологий.

Значительные объёмы информационных потоков, обрабатываемых в всевозможных учреждениях, направлены в основном на создание документов необходимых в управлении. Такие управленческие документы, в свою очередь, ориентированы на принятие управленческих решений, то есть являются главной функцией любого учреждения.

Введение автоматизированных информационных систем в деятельность различных учреждений формирует полномочия для роста качества документационного снабжения управления и позволяет повысить производительность и качество управленческого труда.

На сегодняшний день достаточно актуальным считается вопрос автоматизации учебного процесса, в том числе в средних учебных заведениях (СУЗах).

Объектом дипломной работы является предметная область государственного бюджетного профессионального образовательного учреждения Краснодарского края «Туапсинский социально-педагогический колледж».

Цель исследования- разработка и проектирование автоматизированная информационной системы учета успеваемости и посещаемости студентов.

Разработанная в ходе дипломной работы система должна выполнять следующие процессы:

- проводить анализ списка студентов для деканата и внутренних отделов колледжа;
- хранить все рабочие файлы в едином хранилище с обеспечением удобного быстрого поиска;
- производить резервное копирование документов;
- обеспечение доступа сотрудников ко всей необходимой

информации.

Научной задачей исследования является разработка комплексной модели информационного обеспечения автоматизированного учета успеваемости и посещаемости студентов, формирование единой информационной базы и оптимизация потоков информации в единой информационной базе.

Методы исследования. Теоретическое исследование проводилось с использованием методов аналитического и имитационного моделирования, теории вероятности, математической статистики, теории массового обслуживания и теории управления в технических системах.

Также применялись в качестве инструментария методы анализа научной и информационной базы, синтеза полученных данных в теоретические выводы и практические рекомендации и методы экономического анализа.

Практическое значение исследования состоит в том, что оно содержит разработку предложения по совершенствованию системы управления учебным заведением.

Теоретической основой написания работы является литература в данной области. Список использованных литературных источников приведён в конце работы.

При выполнении данной работы было использовано программное обеспечение класса текстовый процессор, а также система управления базами данных.

1. Описание предметной области

1.1 Объект автоматизации

Образовательная организация государственное бюджетное профессиональное образовательное учреждение Краснодарского края «Туапсинский социально-педагогический колледж» (далее - колледж) находится в подведомственном подчинении министерству образования науки и молодежной политики Краснодарского края.

Учредителем колледжа является министерство образования, науки и молодежной политики Краснодарского края.

Функции собственника имущества Бюджетного учреждения осуществляют департамент имущественных отношений Краснодарского края.

Колледж является юридическим лицом и функционирует в соответствии с законодательством Российской Федерации и Уставом.

Колледж в своей деятельности руководствуется Конституцией Российской Федерации, Законом Российской Федерации от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации», приказом Министерства образования и науки российской Федерации от 14.06.2013 № 464 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам среднего профессионального образования», Законом Краснодарского края от 16.07.25013 № 2770-КЗ «Об образовании в Краснодарском крае», иными нормативно-правовыми актами, регламентирующими образовательную деятельность, Уставом колледжа.

В 2015 году колледж успешно прошел государственную аккредитацию, в соответствии с чем, министерством образования и науки Краснодарского края было выдано свидетельство об аккредитации № 03429 от 03.06.2015 (бланк серия 23А01 № 0001170), действующее до 03.06.2021.

25 января 2016 года колледжем были внесены изменения в Устав в связи с изменением наименования учредителя - министерства образования, науки и молодежной политики Краснодарского края.

В соответствии с действующей лицензией (бессрочной), образовательная деятельность в колледже осуществляется по следующим основным профессиональным образовательным программам подготовки квалифицированных рабочих и специалистов среднего звена среднего профессионального образования:

В колледже реализуется 11 основных образовательных программ среднего профессионального из них: 2 основные образовательные программы подготовки квалифицированных рабочих, служащих (далее - ГГГЖРС); 9 основных образовательных программ подготовки специалистов среднего звена (далее ППССЗ).

Численность контингента обучающихся по состоянию на 01.04.2016 года по всем реализуемым образовательным программам подготовки специалистов среднего звена составляет 978 человек, в том числе по очной форме - 799 чел., (внебж.) - Пчел, по заочной форме (бюдж.) - 162 чел., (внебж.) - 6 чел.,

В целом контингент колледжа формируется за счет обучающихся на бюджетной основе.

Движение контингента и выпускников за последние 3 года свидетельствует о заинтересованности граждан в образовательных программах колледжа, стабильном наборе и выпуске специалистов.

Отчисления из колледжа производятся по причине академической неуспеваемости, переводы в другие образовательные учреждения, по уважительным причинам. Педагогическим коллективом предпринимаются усилия для снижения количества отчисляемых: проводятся малые педсоветы, встречи с родителями, заседания совета профилактики по вопросам посещаемости и успеваемости, консультации с психологами, работа социального педагога.

С целью снижения количества отчисленных по причинам, связанным с пропусками учебных занятий и академическими задолженностями, в колледже разработана и внедрена система контроля успеваемости, посещаемости и качества знаний обучающихся. Она включает в себя не только фиксирование

опоздавших и отсутствующих на учебных занятиях, но и ежедневный анализ данных, проводимый кураторами учебных групп, руководителями структурных подразделений, директором. После анализа разрабатывается система воспитательных мероприятий по каждому нарушителю.

Воспитательные мероприятия включают в себя: анализ личных дел; проведение профилактических бесед со студентами и их родителями с целью недопущения пропусков занятий без уважительных причин; извещение родителей студентов по почте об успеваемости, посещаемости; рассмотрение на совете профилактики; информирование Отдела МВД России по Туапсинскому району о сложных подростках, а также комиссии по делам несовершеннолетних.

По итогам семестра в колледже проводится педагогический совет, анализируется работа всего педагогического коллектива. Итоги и результаты анализа доводятся до сведения родителей на родительских собраниях. Проводится большая индивидуальная работа с обучающимися и их родителями. Ежегодно проводятся мероприятия по профориентационной работе среди учащихся школ г. Туапсе и Туапсинского района в этом году в рамках празднования 50 лет ГБПОУ КК ТСГЖ в день открытых дверей посетило колледж (600 учащихся из 32 школ). Учащихся ознакомили с правилами приема в колледж.

Система управления колледжем направлена на совершенствование работы по организации образовательного процесса с целью обеспечения реализации основных профессиональных образовательных программ в соответствии с требованиями федеральных государственных образовательных стандартов.

Управление колледжем осуществляется на принципах сочетания единоначалия и коллегиальности в соответствии с законодательством Российской Федерации, уставом колледжа, локальными актами, регулирующими деятельность всех структурных подразделений колледжа.

Возглавляет колледж директор, назначаемый на эту должность и

освобождаемый от нее министерством образования, науки и молодежной политики Краснодарского края.

Директор действует от имени колледжа без доверенности, представляет его интересы во всех органах государственной власти, органах власти субъекта Российской Федерации, органах местного самоуправления, в отношении с юридическими и физическими лицами. Руководство финансово-экономической деятельностью возлагается на главного бухгалтера колледжа. Заместители директора осуществляют общее руководство структурными подразделениями и руководство реализацией программ и планов по соответствующим направлениям деятельности. Их компетенции и полномочия определены должностными инструкциями.

Организационная структура колледжа соответствует его уставу, нормативным документам и функциональным задачам (рис. 1.1).

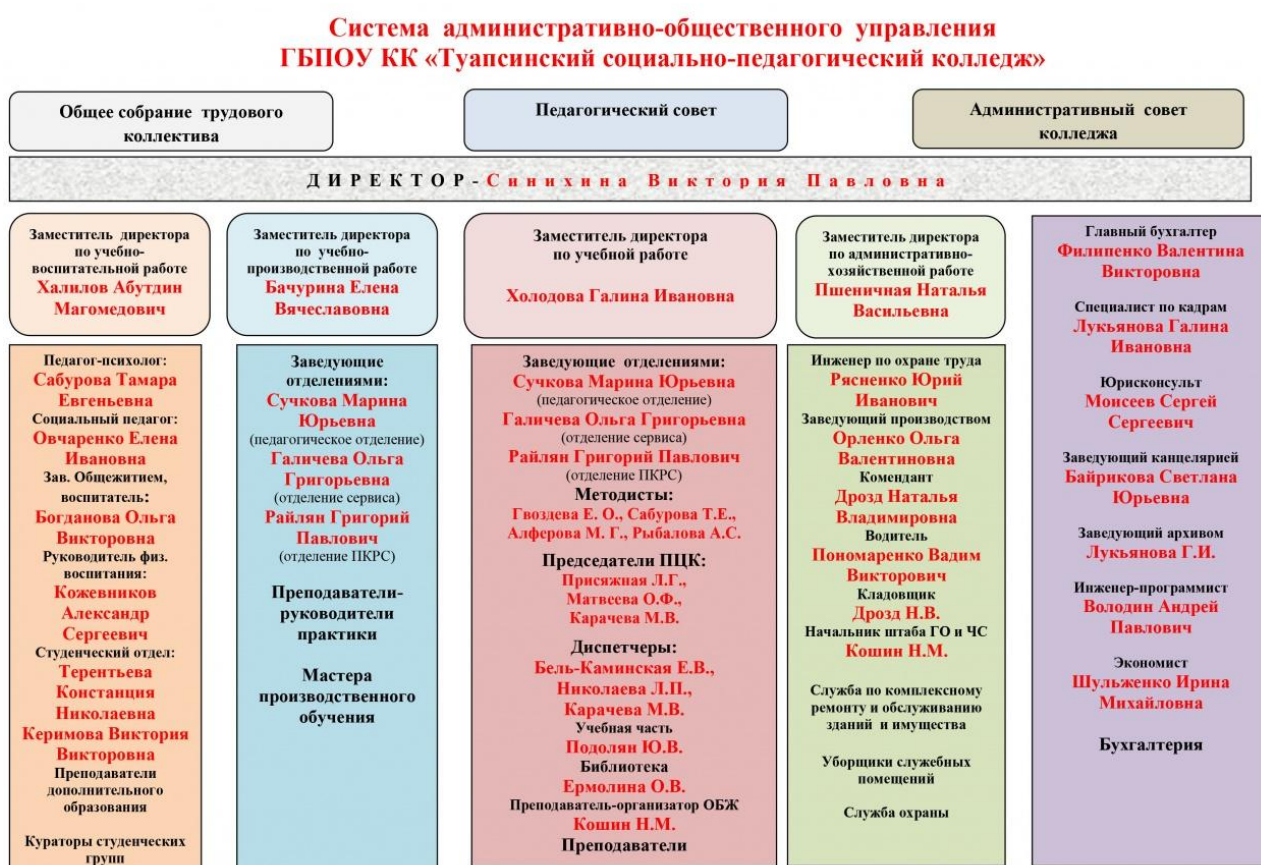


Рисунок 1.1 Организационная структура колледжа

В работе всех служб колледжа действует единая локальная сеть. В 2015

году в колледже была обновлена программа 1С Колледж Проф. В настоящий момент идет заполнение справочников программы всей необходимой информацией. В конце 2015 года также была установлена программа сбора информации ФИС ФРДО и обновлена программа ФИС ЕГЭ.

Организована система контроля за исполнением приказов и распоряжений. Внутренний контроль осуществляется заместителями директора, руководителями структурных подразделений. Регулярно проводятся административные совещания при директоре, где обсуждаются текущие вопросы образовательного процесса, перспектив развития колледжа, вопросы совершенствования и развития материально-технической базы колледжа.

В целях организации управления учебным процессом и его информационной поддержки в создана и принята информационно-аналитическая система (ИАС) «Университет», позволяющая построить единое информационное пространство в рамках учебного процесса, включающее необходимые функции по учету и организации движения студентов, формированию учебных планов факультетов и кафедр, формированию расписаний сессий, учета её результатов. Для удобства функционирования ИАС «Университет» построен по модульному принципу, с дифференциацией на сегментированные автоматизированные системы и рабочие места:

- автоматизированное рабочее место (АРМ) «Деканат»;
- «Приемная комиссия»;
- автоматизированные системы «Рейтинг студентов», «Планы и отчеты», «Аспирантура».

По возможностям использования информационных технологий в управлении образовательной деятельностью можно судить по следующей схеме (рис.1.2).

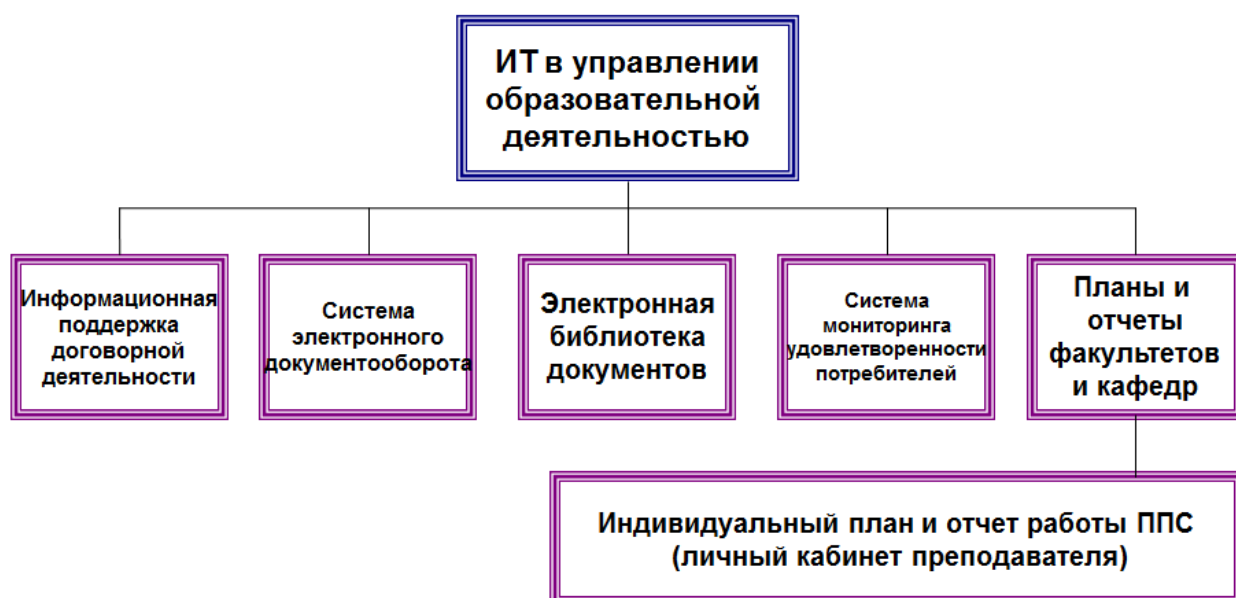


Рисунок 1.2 - Информационные технологии в управлении образовательной деятельностью

Современные образовательные технологии по сути являются неотъемлемой и обязательной частью учебного процесса. Уникальный и специфичный программно-технический комплекс «УМКа» обеспечивает беспрепятственный доступ к материалам учебно-методических комплексов через сеть Интернет (тестам, семинарским заданиям, лекциям), а также организация обратной связи с преподавателем посредством комментариев студента и форума в рамках каждой дисциплины кафедры. В настоящее время в УМКе зарегистрировано уже более 12 тысяч пользователей, размещено более 40 тысяч документов по почти 7 тысячам учебным курсам.

Далее рассмотрим основные технологии электронного документооборота:

СЭД «1С:Управление делами». В настоящее время уже внедрена система электронного документооборота «1С:Управление делами», направленная на автоматизацию деловых процессов, согласование проектов, планов и мероприятий, организацию потока бумажных и электронных документов.

Использование данной системы электронного документооборота позволяет в СУЗе повысить исполнительскую дисциплину, прозрачность и управляемость работы всех факультетов колледжа, наладить учет бумажных и

электронных документов, номенклатуры дел, проектов и процессов СУЗа. «1С:Управление делами» позволяет наладить поток задач и поручений между сотрудниками колледжа, надзор за ходом исполнения поручений, планированием и учетом мероприятий, учетом движения и местоположения различного рода бумажных документов (делопроизводство).

Рассматриваемая система построена на основе стандартов делопроизводства, принятых в настоящее время в РФ, и реализует все необходимые этапы прохождения документов, что дает возможность значительно повысить общее качество управленческой деятельности в колледже (рис.1.3).

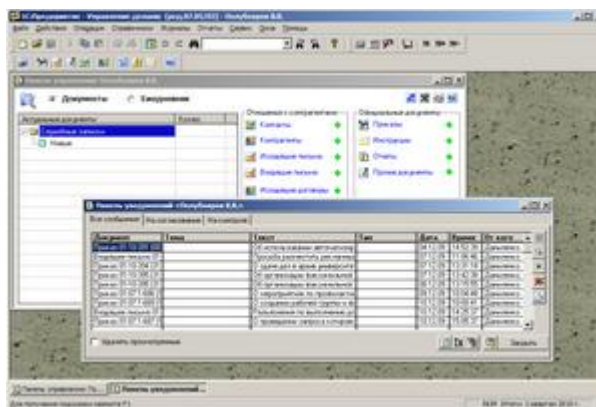


Рисунок 1.3 - Образец интерфейса 1С:Управление делами

АС «Договор». В рамках увеличения эффективности работы отделов и подразделений СУЗа, участвующих в договорной работе и создания банка данных о договорах в колледже была создана и реализована автоматизированная система «Договор». Целевыми функциями данной системы является создание и хранения данных по каждому договору, а также составление реестра договоров за определенный период. Реализация этой системы позволила улучшить контроль над исполнением обязательств по договорам.

Наименование издательства	Предмет договора	Сумма	Дата заключения	Исполнитель	Дата окончания	Отметка об исполнении	Примечания	Дополнительные замечания
Иванов Иван Емельевич	на тиражирование буклета в количестве 500 шт.	5 544.00	26.02.2010	Иванов Иван Емельевич	27.02.2010	исполнен в соответствии со сроками		

Рисунок 1.4 - Образец интерфейса АС «Договор»

ТГМТ в данное время использует систему электронного документооборота и управления взаимодействием «DIRECTUM» в процессе управленческой деятельности факультетов и кафедр. Преподаватели ТСПК всегда стремятся использовать в учебном процессе передовые методические разработки, в том числе и последние достижения в области бизнес-приложений. С этой целью, для привлечения современных научно-технических средств в учебный процесс, ТСПК и компания DIRECTUM заключили договор на использование системы электронного документооборота и управления взаимодействием DIRECTUM в управленческом и учебном процессе. Выбор в сторону именно данной системы электронного документооборота был сделан не случайно - система DIRECTUM объединяет все современные ECM-тренды и может использоваться, как универсальное учебное пособие не только для студентов, изучающих ИТ, но и для курсов по документоведению.

1.2 Постановка задачи

Для ведения деятельности учебного заведения крайне важно иметь точную и достоверную информацию о студентах, их оценках и посещаемости.

В данной работе разрабатывается информационная система учета

успеваемости и посещаемости студентов в государственном бюджетном профессиональном образовательном учреждении Краснодарского края «Туапсинский социально-педагогический колледж» с помощью СУБД Access.

1.3 Входные и выходные данные

Входными данными являются: во-первых, изменения, вносимые пользователем и простановка оценок, во-вторых, запросы пользователя на поиск информации о студенте! Данные сведения вводятся при помощи аппаратных средств ввода – клавиатуры и мыши (будут показаны на рисунке 1.5). Посредством монитора система информирует о своей работе и направляет дальнейшие действия пользователя.

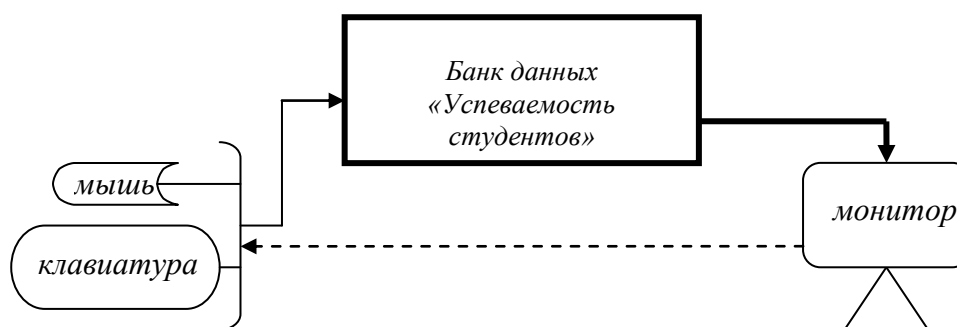


Рисунок 1.5 – Работа банка данных

В процессе работы с данным программным продуктом формируются и при необходимости распечатываются финансовые или иные документы.

Выходными данными являются: во-первых, обновленные базы данных и информация на мониторе, отображающая результаты изменений; во-вторых, информация на мониторе по результатам запроса на поиск. Посредством монитора система информирует о своей работе и направляет дальнейшие действия пользователя.

Процесс учета успеваемости и посещаемости в ТСПК включает в себя несколько этапов. Изначально диспетчеры деканатов в течение семестра контролируют ведение журналов посещаемости групп. В конце семестра сотрудник деканата, формируя ведомости на текущую сессию, актуализирует

допуски и список студентов. По окончании сессии заместитель декана по учебной работе может рассчитать статистику контрольных недель и сессий в разрезе каждой группы или факультета. Также результаты сессий являются исходными данными для печати дипломов и приложений к ним.

Поскольку бизнес-процесс учета успеваемости не автоматизирован, все операции совершаются вручную, что, в свою очередь, увеличивает время выполнения работ, также существует вероятность ошибки.

В результате анализа бизнес-процесса «Учет успеваемости и посещаемости» была построена диаграмма потоков данных, на которой можно увидеть, какие участки подлежат автоматизации. Диаграмма потоков данных «Учет успеваемости и посещаемости» представлена на рисунке 1.6.

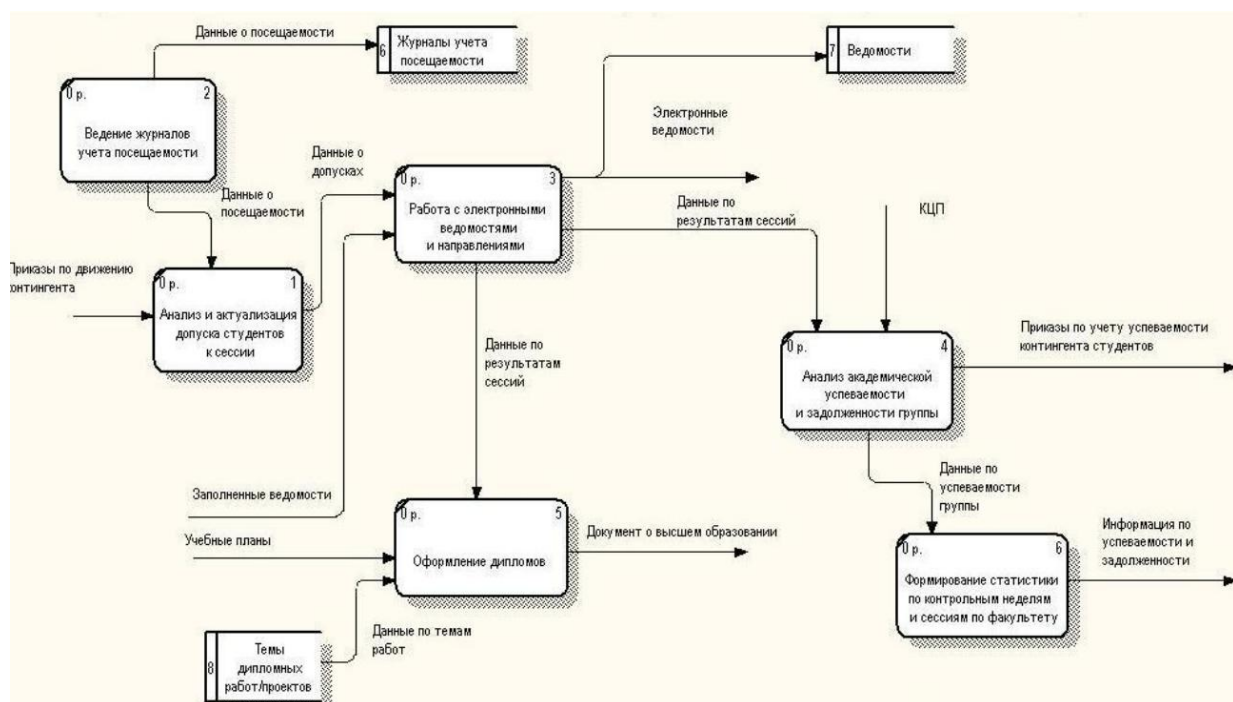


Рисунок 1.6 – Диаграмма потоков данных «Учет успеваемости и посещаемости»

Если раньше процесс работы с ведомостями выполнялся вручную, то теперь предполагается, что данные о результатах сессий должны вноситься в систему сотрудником деканата, для того, чтобы формировать статистику успеваемости, задолженности, а также дипломы и приложения к ним. Для осуществления изменений целесообразно использовать единый программный

модуль по учету успеваемости, который будет взаимодействовать с другими модулями комплексной информационной системы управления СУЗом. Реализация позволит повысить эффективность работы персонала деканатов, которая будет выражаться в сокращении времени на обработку данных и получении статистики с учетом всех необходимых параметров.

1.4 Требования к автоматизированной информационной системе

Цветовую палитру интерфейсов выдержать в стиле Windows. Разрешение экрана: 1024*768 — средний шрифт.

Интерфейс должен быть защищенным от неправильных действий пользователя.

Для ввода и корректировки данных в интерактивном режиме необходимо использовать несколько компьютеров, объединенных в локальную сеть.

Требования к программному обеспечению — операционная система Windows XP и более поздних версий, MicrosoftOffice не ниже 2007.

Требования к техническому обеспечению — компьютер Pentium 4, 1 ГГц или более поздних моделей стандартной конфигурации, лазерный принтер LaserJet 1100.

Информационная система должна обеспечивать следующие требования к надежности:

- устойчивая работа системы в целом;
- контроль на ошибочность вводимых данных;
- проверка на допустимость числовых данных.

2. Проектирование информационной системы

2.1 Методология проектирования базы данных

Информационная система состоит из базы данных и клиентского приложения, которое обеспечивает удобную работу с данными. В свою очередь база данных состоит из таблиц с наиболее важными данными о преподавателях, студентах.

База данных создана по технологии клиент-сервер, а клиентский интерфейс на совместимом с базой данных языке программирования. Технология клиент-сервер выбрана по той причине, что позволяет оперативно вносить изменения в базу данных, быстро менять ее структуру, является менее дорогостоящей, нежели технология файл-сервер. База данных разработана на базе системы управления базами данных (СУБД), которая поддерживает множественное подключение, чтобы сразу несколько пользователей могли получать доступ к данным и работать с ними, если будут иметь на это права. Разграничен доступ к данным: студенты могут видеть только свою посещаемость и успеваемость; преподаватели имеют возможность просмотра посещаемости и успеваемости студентов, добавлять, изменять и удалять данные, которые входят в их полномочия (выставление оценок, ведение таблицы посещаемости); сотрудники колледжа (работники кафедр, учебной части) могут просматривать данные, а также изменять их, но только в поле своих должностных обязанностей (добавление информации о студентах, преподавателях).

Система ускоряет процесс работы с данными, делает его более удобным и отвечает требованиям заказчика. Система имеет должный уровень безопасности данных.

В ходе выполнения дипломной работы можно выделить три этапа, выполнение которых способствовало достижению цели работы. Решение каждой из задач приближало к достижению итогового результата. Этапы и их результаты приведены в таблице 2.1.

Таблица 2.1 - Этапы создания информационной системы

Этап	Результат
1. Создание базы данных	В результате решения данной задачи получили рабочую базу данных, которая содержит минимальный объем информации.
2. Создание клиентских приложений и серверного приложения	Результатом данного этапа является клиентское приложение, которое обеспечивает быструю и удобную работу с информацией в базе данных. Добавление данных с его помощью.
3. Обеспечение безопасности информационной системы	В результате разграничения доступа группы пользователей могут выполнять только те операции с базой данных, которые им разрешены.

Для повышения эффективности работы над дипломным проектом был составлен календарный план (выделены основные этапы создания и заданы сроки выполнения каждого из них).

На этапе 1 выполнялись следующие задачи:

- 1.1. формулировка цели создания информационной системы;
- 1.2. определение задач, решение которых позволит достичь поставленной цели;
- 1.3. составление требований к информационной системе;
- 1.4. обследование предприятия,
- 1.5. сбор необходимой информации;
- 1.6. анализ полученной информации.

На этапе 2 выполнялись следующие задачи:

- 2.1. проектирование информационной системы;
- 2.2. создание модели будущей информационной системы;
- 2.3. создание минимального объема данных, которые понадобятся для тестирования информационной системы.

На этапе 3 выполнялись следующие задачи:

- 3.1. разработка информационной системы;
- 3.2. первоначальное тестирование информационной системы, устранение ошибок;

3.3. ввод в эксплуатацию на предприятии;

3.4. сопровождение информационной системы (документация, обновление системы).

Так как на отделении колледжа уже имеется компьютерная сеть с использованием сервера, то покупка нового оборудования не потребовалась.

Большинство существующих продуктов для автоматизации учебного процесса позволяют автоматизировать лишь отдельные участки учебного процесса, например, хранение личных карточек студентов, или формирование выписок в диплом, или только кадровый учет и так далее. Но очевидным является тот факт, что все эти процессы очень тесно связаны друг с другом, и отделить их друг от друга невозможно, а значит и автоматизировать их отдельно друг от друга – неэффективно. Раздельная автоматизация не решает основной задачи, а именно обеспечение оперативного доступа к любой информации, касающейся автоматизируемой предметной области. С моей точки зрения, все процессы внутри учебного заведения должны быть автоматизированы в комплексе, а вся информация, сопровождающая их, должна находиться в единой базе данных. Внедрение разрабатываемой системы призвано:

1. Руководителям учебного заведения:

- уменьшить временные затраты в процессе планирования и управления деятельностью учебного заведения;
- получить оперативный доступ к информации;
- повысить эффективность управления образовательным процессом и образовательным заведением в целом;
- предоставить учащимся и их родителям дополнительные информационные ресурсы, доступ к которым можно организовать на платных основаниях, что может стать дополнительным источником для заработка.

2. Преподавателям, сотрудникам учебной части и сотрудникам кафедр учебного заведения:

- получить оперативный доступ к документам, регламентирующим

образовательный процесс: учебным планам, графикам, расписанию занятий, спискам контрольных мероприятий;

- получить оперативный доступ к информации об учащихся, их успеваемости, посещаемости занятий;
- организовать автоматизированный сбор статистических данных по контингенту учащихся;
- организовать внутри учебного заведения систему электронного документооборота;
- уменьшить временные издержки на согласование различных документов;
- организовать централизованную проверку качества преподавания посредством тестов.

3. Учащимся:

- получать быстрый, простой и оперативный доступ к касающейся учебного процесса информации посредством web-интерфейса: расписания занятий, экзаменов, зачетов;
- получать информацию о собственной успеваемости, задолженности по оплате.

4. Родителям:

- контролировать успеваемость своих детей;
- получить быстрый, простой и оперативный доступ к касающейся учебного процесса информации посредством web-интерфейса: расписания занятий, экзаменов, зачетов;
- получить возможность общения с сотрудниками и администрацией учебного заведения в режиме online.

2.2 Выбор СУБД для информационной системы

При проектировании таблиц лучше разработать структуру на бумаге и

только затем начинать работу с СУБД Access.

MicrosoftAccess - это функционально полная реляционная СУБД. В ней предусмотрены все необходимые инструментальные средства для определения и обработки данных, а также для управления ими при работе с большими объемами информации. Что касается легкости использования, то MicrosoftAccess совершил здесь настоящий переворот, и многие для создания своих собственных баз данных и приложений обращаются именно к нему.

MicrosoftAccess предоставляет максимальную свободу в задании типа ваших данных (текст, числовые данные, даты, время, денежные значения, рисунки, звук, документы, электронные таблицы). Вы можете задать также форматы хранения (длина строки, точность представления чисел и даты времени) и предоставления этих данных при выводе на экран или печать. Для уверенности, что в базе данных хранятся только корректные значения, можно задать условия на значения различной степени сложности.

MicrosoftAccess может работать с большим числом самых разнообразных форматов данных, включая файловые структуры других СУБД. СУБД позволяет работать с данными, применяя различные способы. Например, вы можете выполнить поиск информации в отдельной таблице или создать запрос со сложным поиском по нескольким связанным между собой таблицам или файлам. С помощью одной единственной команды можно обновить содержание отдельного поля или нескольких записей. Для чтения и корректировки данных, вы можете создать процедуры, использующие функции СУБД. У многих систем имеются развитые возможности для ввода данных и генерации отчетов.

2.3 Этапы разработки базы данных

1. Этап сбора и анализа информации. Первым шагом разработки информационной системы является формулировка цели, с которой она разрабатывается. На втором шаге были выбраны средства, которые помогли достичь поставленной цели. Главным требованием к создаваемой

информационной системе являлось то, чтобы она позволяла избежать дублирования данных, ускоряла процесс поиска нужной информации. Перед созданием информационной системы было произведено обследование учебного заведения и собрана информация, которая после была проанализирована. После анализа полученной информации появилась возможность перейти ко второму этапу создания информационной системы.

2. Этап проектирования информационной системы. На этапе проектирования, на основе анализа полученной информации, было решено, что система будет состоять из проекций. Были определены таблицы, которые будут входить в каждую из проекций, какую информацию они будут содержать. Была создана модель базы данных, определены таблицы базы данных, определены связи между таблицами. В базу данных были добавлены первичные данные, которые понадобятся на третьем этапе для тестирования созданной информационной системы. Так как этап проектирования базы данных очень важен, то его я рассмотрю далее более подробно.

3. Этап реализации информационной системы. После успешного выполнения первых двух этапов создания информационной системы перешли к непосредственной реализации информационной системы. После того, как информационная система была реализована, началось ее тестирование с использованием первичных данных. Были устранены ошибки и на данный момент система готова к вводу в эксплуатацию в колледже.

2.4 Проектирование базы данных

Очень важно выполнить процесс проектирования базы данных от начала и до самого конца, так как частичное проектирование даст плохой результат. Из-за недостаточной проработки стандартных вопросов возникают многие проблемы с базами данных. Частичное выполнение этапа проектирования – это то же самое, что и пропустить его в целом. Полностью выполненный этап проектирования базы данных позволит создать соответствующую требованиям

логическую структуру базы данных и целостность ее данных.

Нужно помнить о том, что, чем ответственнее отнестись к этапу проектирования, тем выше будет качество структуры и целостность данных в базе данных. Если уделять мало времени этапу проектирования базы данных, то тем больше риск того, что при работе с базой данных, мы столкнемся с проблемами. Конечно, даже большое количество времени, которое уделено проектированию базы данных, не может гарантировать того, что мы не столкнемся с проблемами, но это поможет минимизировать проблемы. Нужно также помнить, что реализовать хорошо спроектированную базу данных проще, чем реализовать базу данных, которая спроектирована плохо.

1. Формулировка проблемы и постановка задачи. Первым шагом на этапе проектирования базы данных является формулировка проблемы и постановка задачи. Нужно определиться с тем для чего создается база данных, каких целей нужно достигнуть и определиться с функциональностью создаваемой базы данных. Каждая база данных создается для определенных нужд, не важно каких, будь это решение конкретной проблемы или создание базы данных, которая будет частью общей информационной системы. Когда сформулирована цель, с которой создается база данных, то это может гарантировать, что мы разрабатываем нужную структуру базы данных, и значит, что мы занимаемся сбором данных, которые необходимы для того, чтобы мы достигли намеченной цели. На первом шаге проектирования базы данных мы также определяемся с функциональностью создаваемой базы данных. Мы определяем, что смогут делать пользователи нашей базы данных и данными, которые в ней будут находиться. Для определения функциональности базы данных, разработчик базы данных общается с заказчиком, а именно руководителями организации, а также с конечными пользователями базы данных.

2. Анализ имеющейся базы данных. Вторым шагом на этапе проектирования базы данных является анализ имеющейся базы данных. Таким образом, мы получим ценную информацию о том, как организация собирает и

использует данные. Неотъемлемой частью анализа является общение с пользователями и руководителями организации с той целью, чтобы узнать, как они взаимодействуют с базой данных. Нужно узнать какие информационные требования у них в настоящее время. Такое интервью проводится именно с двумя группами людей – пользователи и руководители, чтобы нужно понять, как пользователи взаимодействуют с базой данных и какую информацию из нее получают руководители. Только проведя интервью с будущими пользователями проектируемой базы данных, мы сможем разработать такую базу данных, которая будет удовлетворять потребности заказчика.

Затем, используя информацию, полученную в процессе интервьюирования сотрудников организации, мы сможем составить начальный список нужных полей. Этот список в дальнейшем будет меняться по мере того, как мы будем разрабатывать проект базы данных. После того, как начальный список полей будет составлен, мы посылаем его на рассмотрение пользователям и руководителям. Обратная связь – приветствуется. Облегчит задачу проектирования и присланные от них предложения и изменения. Если присланные предложения хорошо изложены и обоснованы, то нужно переделать список, снова выслать на рассмотрение и в случае получения одобрения, переходим к третьему шагу проектирования базы данных.

3. Создание структуры данных. Третий шаг этапа проектирования базы данных – создание структуры данных. На этом шаге мы определяемся с тем, какие таблицы будет содержать проектируемая база данных, какие типы таблиц будут использованы. Определяемся с полями в таблицах и ключами.

Таблицы – это основа разрабатываемой базы данных. Сначала нужно определиться с темами, которые будут представлять таблицы. Их определение должно исходить из формулировки проблемы и задач, которые были определены на первом шаге этапа проектирования. Далее таблицы заполняются полями, которые были получены на втором шаге проектирования. После этого все таблицы должны быть просмотрены еще раз, нужно удостовериться в том, что каждая таблица отражает только одно событие, и что в ней нет

повторяющихся полей.

Далее подробнее работать нужно с полями таблиц, разбивать составные поля на несколько. Следует добиться того, чтобы поля содержали единственное значение, для этого разбиваем поля на несколько полей. Нужно добиться того, чтобы таблица содержала только те поля, которые относятся к ее теме. Затем определяем первичные ключи для каждой таблицы. Главная задача заключается в том, чтобы каждая таблица имела определенный первичный ключ, значение которого однозначно определяет любую запись в таблице.

И в конце третьего шага определяем спецификации для каждого поля проектируемой базы данных. После мы документируем спецификации для каждого поля. Затем делаем обзор структур таблиц и спецификаций с пользователями и руководителями. После того, как структура базы данных утверждена заинтересованными лицами, переходим к следующему шагу процесса проектирования базы данных.

4. Определение связей между таблицами. Четвертый шаг проектирования – определение связей между таблицами. Снова проводится интервьюирование сотрудников и руководителей, и после определяем связи и характеристики связей между таблицами, обеспечиваем целостность данных. Процесс интервьюирования достаточно сложен. Пользователи и руководители могут помочь (могут и не помочь) в определении зависимостей во множестве данных. Так как мы не в состоянии знать каждый нюанс использования данных в организации, поэтому любые сведения о том, как используют данные пользователи и руководители, будет очень полезна. После того определены зависимости, мы задаем логические связи между таблицами с помощью первичных ключей или связующих таблиц. Что использовать – зависит от типа связей, которые существуют между таблицами. Затем определяется степень участия таблиц в каждом отношении.

5. Нахождение и задание правил работы. Пятым шагом проектирования базы данных является – нахождение и задание правил работы. На этом шаге будут снова проводиться интервьюирование, определяться

ограничения, задаваться правила работы, определяться и реализовываться проверочные таблицы. То, какими способами организация собирает, хранит и использует данные, определяет множество ограничений и требований, которые мы должны встроить в базу данных. И интервью с пользователями и руководителями поможет определить ограничения, которые будут накладываться на данные, структуры данных. Далее нужно утвердить и документировать эти правила.

Уровень целостности, который задает правила работы в этой точке этапа проектирования, существенен, потому что он прямо связан с тем способом, которым организация представляет и использует свои данные. По мере роста организации, ее отношение к данным меняется, что означает, что правила работы также должны измениться. Определение и реализация правил работы в базе данных итеративный процесс, и мы должны постоянно отслеживать изменения, если вы собираетесь поддерживать заданный уровень целостности данных.

6. Нахождение и задание представлений. Шестым шагом этапа проектирования является нахождение и задание представлений. Опять-таки помогут с этим определиться интервью с сотрудниками и руководителями. Беря интервью у пользователей и руководителей, мы должны найти и встроить найденные представления в базу данных. Некоторым сотрудникам нужна детальная информация для выполнения своей работы, а другим – итоговая, на основе которой они могут принимать стратегические решения для организации. Каждая из групп пользователей должна иметь доступ к информации в соответствии со своими потребностями, и мы можем использовать эти представления для решения проблем доступа. Далее определяются представления, которые были выделены в процессе интервьюирования, используя подходящие таблицы и поля. Устанавливаются критерии для тех представлений, которые требуются, чтобы получить конкретную информацию.

7. Обзор целостности данных. И заключительным, седьмым, шагом, является проверка конечной структуры базы данных на предмет целостности

данных. Сначала просматриваем еще раз каждую таблицу, что удостовериться в том, что она отвечает требованиям разработанной базы данных. Далее проверяем структуру полей каждой таблицы. Если обнаруживаются проблемы, то нужно их исправить и сделать обзор структуры еще раз. После проверяем спецификации полей и делаем необходимые усовершенствования. Далее проверяем связи между таблицами, подтверждаем тип связей. И, наконец, мы делаем обзор правил работы, которые были определены ранее на этапе проектирования базы данных. Подтверждаем ограничения, которые наложили на различные аспекты базы данных. Если имеются новые ограничения, то они реализуются как новые правила работы и прибавляются к уже существующему набору правил работы. После того, как мы прошли все шаги этапа проектирования базы данных, мы готовы реализовать логическую структуру разрабатываемой базы данных.

После рассмотрения существующей системы, анализа имеющихся в ней проекций и рассмотрения системы с учетом требований к создаваемой информационной системе, были сделаны выводы, и теперь можно перечислить те проекции, которые нужны в разрабатываемой системе, которые могут понадобиться в будущем. Для удобства представления, проекции разрабатываемой системы, а также различные дополнения, представлены в таблице 1.2.

Таблица 1.2 - Разрабатываемая система и ее возможное обновление

Проекции разрабатываемой системы	Желательные дополнения	Возможные дополнения в будущем
Административная деятельность; Учебный процесс; Успеваемость и посещаемость;	Библиотека;	Хранилище документов; Тестирование.

3. Реализация информационной системы

3.1 Создание базы данных

Индексация применяется для решения проблемы быстрого поиска в больших массивах данных. На определенные поля устанавливается индекс, записи сортируются по этому индексу, и поиск по полям сводится к поиску по индексу.

Оперативные таблицы, в отличие от словарей и справочников, используют информацию из других таблиц, связанных с данной через вторичные индексы.

Главный ключ – это уникальный идентификатор для каждой записи, значение которого однозначно определяет только эту запись.

Главный ключ должен обладать двумя свойствами:

1. однозначной идентификацией записи;
2. отсутствием избыточности - никакое поле нельзя удалить из ключа, не нарушая при этом однозначности (первого свойства).

В спроектированной БД главным ключом для всех таблиц является специально введенный id(номер), т.е., например:

Для таблицы «public_directory_chair» – chair_id

Для таблицы « public_directory_faculty» – faculty_id

Для таблицы « public_directory_subject» – subject_id

Для таблицы « public_student_personal» – student_id

Существует несколько видов связей между таблицами (сущностями):

- 1.«Один к одному» – записи в одной таблице однозначно соответствуют записи в другой таблице.

- 2.«Один ко многим» – одной записи в первой таблице может соответствовать много записей в другой таблице.

- 3.«Многие к одному» – многим записям в первой таблице может

соответствовать одна запись в другой таблице.

4.«Многие ко многим» – многим записям в одной таблице соответствует много записей в другой. Как правило, этот вид связи реализуется с помощью промежуточных таблиц.

Часто в таблицах БД используется подстановка значений в поле одной таблицы из полей другой таблицы, при условии что эти таблицы связаны по определенным полям (рис.3.1).

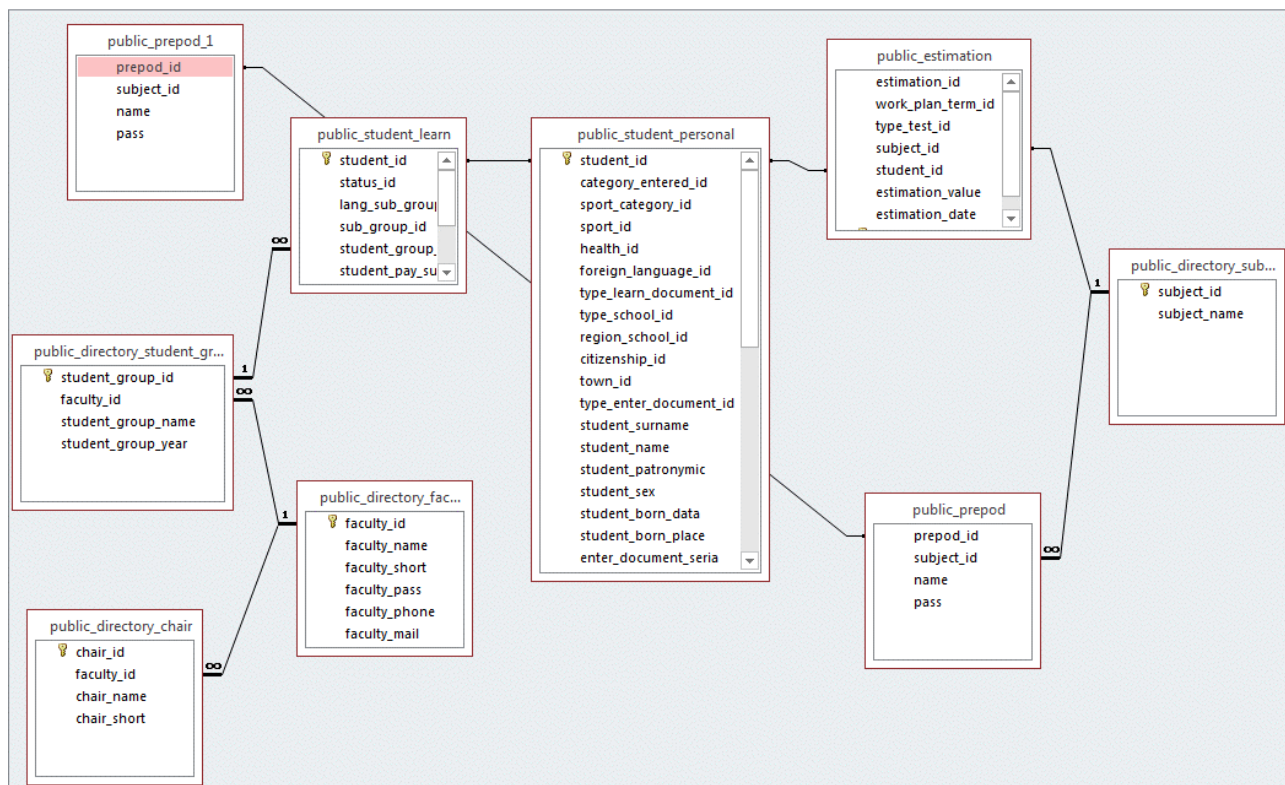


Рисунок 3.1 – Схема данных информационной системы

Таким образом, таблицы «public_directory_student_group» и «public_directory_chair» связаны отношением «многие ко многим» при помощи промежуточной таблицы «public_directory_faculty», связанной с предыдущими таблицами отношением «один ко многим» по полям faculty_id.

Условие целостности не позволяет ввод во внешнее ключевое поле связанной таблицы значения, отсутствующего в ключевом поле главной таблицы, и не допускается удаление любой из таблиц или связанных полей до удаления связи. Однако возможен ввод во внешнее ключевое поле пустого

значения, показывающего, что записи не являются связанными. Например, невозможно ввести в таблицу сделку с несуществующим клиентом, однако, допускается ввод сделки, не относящейся ни к одному из клиентов с пустым значением в поле «Код клиента».

Формы- это экранные бланки, предназначенные для визуализации и ввода данных. Для имеющихся таблиц можно создать форму (-ы). Корешок «Форма» в окне Базы данных, создать. Затем можно указать вариант: Конструктор, Автоформы и т.д. «Мастер форм».

Затем выбираем стиль формы:

- в один столбец;
- ленточная;
- табличная;
- выровненная;
- сводная диаграмма;
- сводная таблица.

Чтобы отредактировать форму в соответствии с нашими требованиями – переходим в режим конструктора. Здесь имеются все необходимые инструментальные средства для редактирования форм.

Подчиненная форма- это форма внутри главной формы, позволяющая вводить и отображать данные со связями «один ко многим». Главная форма представляет часть «один», подчиненная- «многим».

В подчиненной форме выводятся только те записи, которые связаны с текущей записью в главной форме.

Главная форма может быть выведена только как простая. Подчиненная- как таблица, ленточная, простая.

Для создания вложенных форм: в диалоге «Новая форма» выбрать «Мастер». В окне источника данных- таблицу главной формы. Выбираются поля для главной формы. Затем выбирается вторая таблица (для подчиненной формы) и из нее выбираются поля и добавляются в список выбранных полей. Следующее диалоговое окно предлагает выбрать тип представления данных:

«Подчиненные формы», «Связанные формы». Access на основе анализа типа связи сам устанавливает «Подчиненные формы» и можно пройти «Далее».

Остается уточнить вид (ленточный или табличный). По умолчанию принимается табличный. И указать имена форм (лучше дать похожие, чтобы легче ориентироваться в окне БД).

При открытии БД загружается главная кнопочная форма, содержащая основное меню управления БД.



Рисунок 3.2 – Главное меню

Кнопка «Преподаватель» ведёт на форму «Успеваемость студентов» (рисунок 3.3).

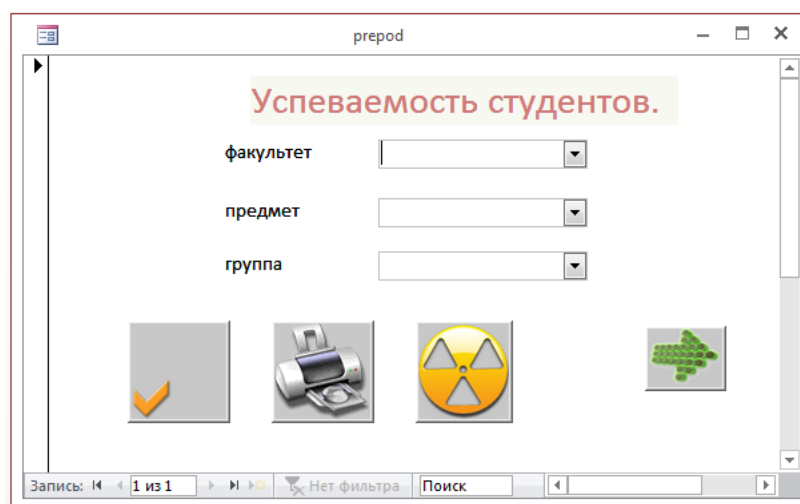


Рисунок 3.3 – Форма «Преподаватель»

Здесь мы выбираем нужную группу для ввода оценок. Если мы выбрали нужную группу, то переходим на форму «оценка», где и вводим нужные нам параметры для каждого студента

На форме `prerod` есть вложенная форма которая отображает среднюю оценку группы (она показана стрелкой). На этой же форме мы можем сохранять, удалять и вставлять новые записи.

С главной формы можно попасть на форму «stud» (рисунок 3.4).

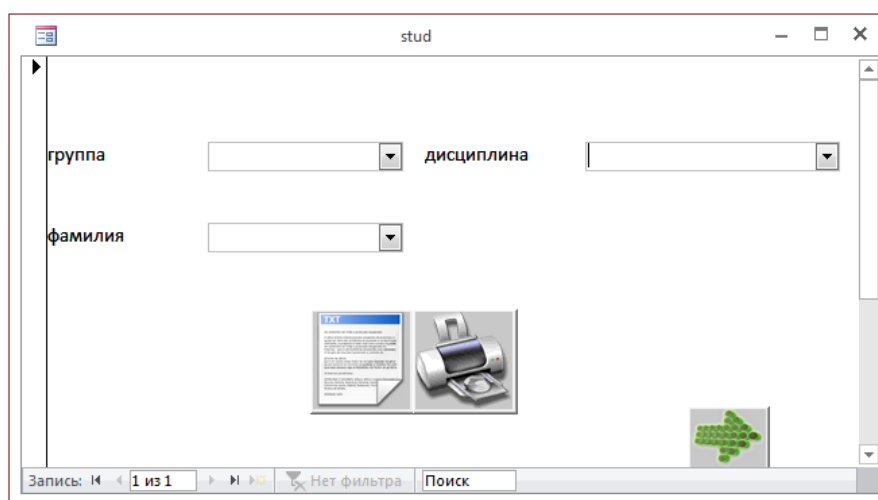


Рисунок 3.4 – Форма «Студент»

Также имеется форма «Руководство пользователя» (рис. 3.5.).

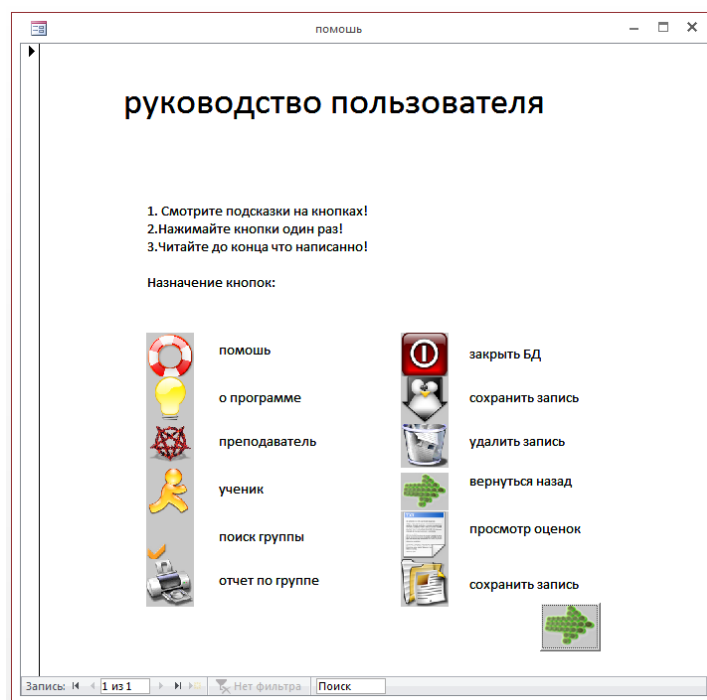


Рисунок 3.5 – Руководство пользователя

3.2 Проверка работоспособности информационной системы

Выполнить несколько запросов к базе данных.

1. запрос на выборку нужной группы
- 2.

```

SELECT          public_directory_student_group.student_group_name,
public_directory_subject.subject_name,   public_student_personal.student_surname,
public_estimation.estation_value, public_estimation.estation_date
FROM          public_directory_student_group      INNER      JOIN
(public_directory_subject INNER JOIN ((public_student_personal INNER JOIN
public_estimation                                ON
public_student_personal.student_id=public_estimation.student_id)  INNER JOIN
public_student_learn                                ON
public_student_personal.student_id=public_student_learn.student_id) ON
public_directory_subject.subject_id=public_estimation.subject_id) ON
public_directory_student_group.student_group_id=public_student_learn.student_group_id
WHERE

```

```

(((public_directory_student_group.student_group_id)=Forms!prepod!Combo_group)
And ((public_directory_subject.subject_id)=Forms!prepod!Combo_name))
GROUP BY public_student_personal.student_surname,
public_directory_student_group.student_group_name,
public_directory_subject.subject_name, public_estimation.estimation_value,
public_estimation.estimation_date;

```

3. запрос на выборку оценок

```

SELECT public_directory_student_group.student_group_name,
public_directory_subject.subject_name, public_student_personal.student_surname,
public_estimation.estimation_value, public_estimation.estimation_date
FROM public_directory_subject INNER JOIN ((public_student_personal
INNER JOIN public_estimation
ON public_student_personal.student_id=public_estimation.student_id) INNER JOIN
(public_directory_student_group INNER JOIN public_student_learn ON
public_directory_student_group.student_group_id=public_student_learn.student_group_id) ON public_student_personal.student_id=public_student_learn.student_id) ON
public_directory_subject.subject_id=public_estimation.subject_id
WHERE
(((public_directory_student_group.student_group_id)=Forms!stud!Combo_group2))
And public_directory_subject.subject_id=Forms!stud!Combo_name2
ORDER BY public_student_personal.student_surname;

```

4. Запрос на вычисления средней оценки

```

SELECT public_directory_student_group.student_group_name,
Avg(public_estimation.estimation_value) AS оценка
FROM public_directory_student_group INNER JOIN
(public_directory_subject INNER JOIN ((public_student_personal INNER JOIN
public_estimation
ON public_student_personal.student_id=public_estimation.student_id) INNER JOIN
public_student_learn
ON public_student_personal.student_id=public_student_learn.student_id)

```

public_directory_subject.subject_id=public_estimation.subject_id) ON
public_directory_student_group.student_group_id=public_student_learn.student_group_id

WHERE public_directory_subject.subject_id=Forms!prepod!Combo_name
GROUP BY public_directory_student_group.student_group_name;

Отчет используется для представления данных в виде, характерном для печатных документов.

Они позволяют получить итоговую информацию, как результат сложной (часто статистической) обработки, группировки, сложной сортировки. Часто применяется графическое представление информации в виде диаграмм и разнообразных графиков

Здесь пользователь имеет возможность просмотреть и распечатать один из представленных отчетов:

student_surname	subject_name	student_group_name	estimation_value	estimation_date
Рейсет	_Математика	03-БИО	45	5/26/2005
Абдулueva	_Математика	03-БИО	42	5/26/2005
Балдаева	_Математика	03-БИО	25	5/26/2005
Балдаева	_Математика	03-БИО	35	4/18/2005
Васильева	_Математика	03-БИО	35	4/18/2005
Васильева	_Математика	03-БИО	0	5/26/2005
Васюковых	_Математика	03-БИО	35	4/18/2005
Васюковых	_Математика	03-БИО	45	5/26/2005
Гиматдинова	_Математика	03-БИО	35	4/18/2005
Гиматдинова	_Математика	03-БИО	45	5/26/2005
Голубева	_Математика	03-БИО	45	5/26/2005
Голубева	_Математика	03-БИО	35	4/18/2005
Емельяна	_Математика	03-БИО	45	5/26/2005

Рисунок 3.6 - Отчёт успеваемости

student_group_id) ON
public_student_personal.student_id=public_student_learn.student_id) ON

```

public_directory_subject.subject_id=public_estimation.subject_id
WHERE
(((public_directory_student_group.student_group_id)=Forms!stud!Combo_group2))
And public_directory_subject.subject_id=Forms!stud!Combo_name2
ORDER BY public_student_personal.student_surname;
Запрос на вычисления средней оценки
SELECT
public_directory_student_group.student_group_name,
Avg(public_estimation. estimation_value) AS оценка
FROM
public_directory_student_group
INNER JOIN
(public_directory_subject INNER JOIN ((public_student_personal INNER JOIN
public_estimation
ON
public_student_personal.student_id=public_estimation.student_id)
INNER JOIN
public_student_learn
ON
public_student_personal.student_id=public_student_learn.student_id)
ON
public_directory_subject.subject_id=public_estimation.subject_id)
ON
public_directory_student_group.student_group_id=public_student_learn.student_group_id
WHERE public_directory_subject.subject_id=Forms!prepod!Combo_name
GROUP BY public_directory_student_group.student_group_name;

```

3.3 Параметрические запросы.

Запрос с параметрами дает качественно новую возможность, а не экономию времени. Эта возможность отделяет сферу действия ПОЛЬЗОВАТЕЛЯ от сферы действия РАЗРАБОТЧИКА приложения Access. Разработчик заранее разрабатывает запрос, выводя задание ряда параметров его условий пользователю. Пользователь вообще понятия не имеет о том, что такое запрос и прочие вещи Access.

Итак, запрос с параметрами- это такой запрос, при запуске которого открывается диалоговое окно с приглашением ввести условие отбора. Условий

(и окон) может быть несколько.

Рассмотрим принцип создания запроса с параметрами с одним условием:

Запрос создается в конструкторе запросов обычным способом, но в строке бланка с названием «Условия отбора», перед теми подписями, где должна быть поставлено конкретное значение параметра, нужно написать текст приглашения для диалогового окна в квадратных скобках.

Если речь идет о диапазоне дат (что также типично для таких запросов), то условие отбора может выглядеть:

Between [Введите начальную дату:] And
[Введите конечную дату:]

4. Экономический расчет разработки информационной системы

4.1 Трудоемкость разработки информационной системы

Оценку трудоемкости разработки программного продукта выполним для ПФ на основе вариантов исследования и на материалах компании rationalsoftware. Для определения весовых показателей действующих лиц все действующие лица системы делятся на три типа: простые, средние и сложные (таблица 4.1). Простое действующее лицо представляет внешнюю систему с четко определённым программным интерфейсом. Среднее действующее лицо представляет либо внешнюю систему, взаимодействующую с данной системой посредством протокола наподобие TCP/IP, либо личность, пользующуюся текстовым интерфейсом. Сложное действующее лицо представляет личность, пользующуюся графическим пользовательским интерфейсом. (таблица 4.2).

Таблица 4.1-Весовые коэффициенты действующих лиц

Тип действующего лица	Весовой коэффициент
Простое	1
Среднее	2
Сложное	3

Для разработанного программного обеспечения, действующие лица и их тип сложности приведен в таблице 4.2.

Таблица 4.2 -Типы действующих лиц

Действующее лицо	Тип
Администратор	Сложное
Преподаватель	Сложное
Студент	Сложное
Общий весовой показатель А	9

Для определения весовых показателей вариантов использования разделим их на три типа: простые, средние и сложные в зависимости от количества транзакций в потоках событий. Общее количество вариантов использования

каждого типа умножается на соответствующий весовой коэффициент, затем вычисляется общий весовой показатель (таблица 4.3).

Таблица 4.3- Весовые коэффициенты вариантов использования

Тип варианта использования	Описание	Весовой коэффициент
Простой	3 или менее транзакций	5
Средний	От 4 до 7 транзакций	10
Сложный	Более 7 транзакций	15

Для разрабатываемой программы сложность вариантов использования приведена в таблице 4.4.

Таблица 4.4-Сложность вариантов использования

Вариант использования	Тип
Инициализация параметров подключения к серверу	Простой
Авторизация	Простой
Создание новой сессии для тестирования	Сложный
Создание новых тестов и вопросов	Сложный
Проведение тестирования	Сложный
Общий весовой показатель UC	55

Рассчитаем показатель UUCP (UnadjustedTedUseCasePoints) по формуле 4.1:

$$UUCP = A + UC, \quad (4.1)$$

где A – общий весовой показатель действующих лиц;

UC – общий весовой показатель вариантов использования.

В итоге получаем $UUCP = 64$.

Техническая сложность проекта (TCF – Technical Complexity Factor) вычисляется с учетом показателей технической сложности (таблица 3.5).

Таблица 4.5 – Показатели технической сложности проекта TCF

Показатель	Описание	Вес
T1	Распределенная система	2
T2	Высокая производительность	1
T3	Работа конечных пользователей в режиме он-лайн	0
T4	Сложная обработка данных	2
T5	Повторное использование кода	1
T6	Простота установки	0,5
T7	Простота использования	0,5
T8	Переносимость	2
T9	Простота внесения изменений	1
T10	Параллелизм	1
T11	Специальные требования к безопасности	1
T12	Непосредственный доступ к системе со стороны внешних пользователей	1
T13	Специальные требования к обучению пользователей	1

Каждому показателю присваивается значение в диапазоне от 0 до 5 (0 означает отсутствие значимости показателя для данного проекта, 5 – высокую значимость).

Значение TCF вычисляется по формуле 4.2:

$$TCF = 0.6 + (0.01 \cdot (\sum T_i \cdot \text{Вес}_i)), \quad (4.2)$$

где T_i – показатель технической сложности проекта.

Для подсистемы учета и анализа состояния средств измерения, согласно весам (таблица 3.6), получим:

$$TCF = 0.6 + (0.01 \cdot 45) = 1.05$$

Таблица 4.6 – Показатели технической сложности разработанной системы

Показатель	Вес	Значение	Значение с учетом веса
T1	2	4	8
T2	1	5	3
T3	1	2	5
T4	1	4	3
T5	1	1	3
T6	0,5	3	1,5
T7	0,5	5	1,5
T8	2	2	8
T9	1	4	3
T10	1	2	1
T11	1	2	2
T12	1	5	5
T13	1	1	1
Z			45

Уровень квалификации разработчиков (EF - EnvironmentalFactor) вычисляется с учетом следующих показателей (таблица 4.7).

Таблица 4.7 – Показатели уровня квалификации разработчиков

Показатель	Описание	Вес
F1	Знакомство с технологией	1,5
F2	Опыт разработки приложений	0,5
F3	Опыт использования объектно-ориентированного подхода	1
F4	Наличие ведущего аналитика	0,5
F5	Мотивация	1
F6	Стабильность требований	2
F7	Частичная занятость	-1
F8	Сложные языки программирования	-1

Каждому показателю присваивается значение в диапазоне от 0 до 5.

Для показателей F1 – F4, 0 означает отсутствие, 3 – средний уровень, 5 – высокий уровень. Для показателя F5, 0 означает отсутствие мотивации, 3 – средний уровень, 5 – высокий уровень мотивации. Для F6, 0 означает высокую нестабильность требований, 3 – среднюю, 5 – стабильные требования. Для F7, 0 означает отсутствие специалистов с частичной занятостью, 3 – средний уровень, 5 – все специалисты с частичной занятостью. Для показателя F8, 0 означает простой язык программирования, 3 – среднюю сложность, 5 – высокую сложность.

Значение EF вычисляется по формуле 4.3:

$$EF = 1.4 + (-0.03 \cdot (\sum F_i \cdot \text{Вес}_i)), \quad (4.3)$$

где F_i – показатель уровня квалификации работников.

Вычислим EF для разработанной программы, данные для которой приведены в таблице 21, по формуле 4.3:

$$EF = 1.4 + (-0.03 \cdot 24,5) = 0.665$$

Результаты расчета показателей уровня квалификации разработчиков разработанной системы представлены в таблице 4.8.

**Таблица 4.8 Показатели уровня квалификации разработчиков разра-
ботанной системы**

Показатель	Вес	Значение F_i	Значение с учетом веса
F1	1,5	4	6
F2	0,5	5	2,5
F3	1	4	5
F4	0,5	1	1
F5	1	3	5
F6	2	4	8
F7	-1	2	0
F8	-1	3	-3
I			24,5

В результате получаем окончательное значение показателя UCP, которое рассчитывается по формуле 4.4:

В результате получаем по формуле 4.4 окончательное значение UCP (UseCasePoints):

$$UCP = UUCP \cdot TCF \cdot EF, \quad (4.4)$$

где UUCP – общий весовой показатель;

TCF – техническая сложность проекта;

EF – уровень квалификации разработчиков.

В итоге получаем $UCP = 44,688$

В качестве начального значения предлагается использовать 20 чел.-ч на одну UCP. Эта величина может уточняться с учетом опыта разработчиков. Приведем пример возможного уточнения.

Рассмотрим показатели F1 – F8 и определим, сколько показателей F1 – F6 имеют значение меньше 3 и сколько показателей F7 – F8 имеют значение больше 3. Если общее количество меньше или равно 2, следует использовать 20 чел./ч на одну UCP, если 3 или 4 – 28. Если общее количество равно 5 или более, следует внести изменения в сам проект, в противном случае риск провала слишком высок.

Для системы учета средств измерения получаем 20 человеко-часов на одну UCP, таким образом, общее количество человеко-часов на весь проект

равно 701,018, что составляет 22 недели при 40-часовой рабочей неделе. Команда разработчиков состоит из одного человека, и добавим 1 неделю на различные непредвиденные ситуации, тогда в итоге получим 23 недели на весь проект.

4.2 Экономическая эффективность внедрения программного продукта

В результате внедрения программного продукта ожидается снижение трудоёмкости работ при системном контроле объекта и условно-годовая экономия фонда заработной платы в организации при использовании разработанной системы.

Для расчета экономического эффекта, который будет получен, при разработке программного компонента для информационной системы образовательного процесса в среде Microsoft Visual Studio 2012, необходимо рассчитать ряд показателей, входящих в понятие экономической эффективности, таблица 4.9.

Таблица 4.9 – Показатели экономической эффективности

Наименование показателя	Обозначение	Значение	Ед. измерения
Срок разработки программы	tr	5	мес.
Основная заработная плата разработчиков	Взп	17000	руб./мес.
Количество пользователей	N	3	чел.
Количество рабочих дней за 20 недель	Кдн	100	день
Режим использования компьютера	P	8	час/сут.
Стоимость компьютера	Zэвм	20000	руб.
Время отладки, установки и внедрения	to	20	час.
Стоимость одного часа работы ЭВМ	Зчк	8	руб.

За время разработки программного обеспечения были израсходованы материалы, стоимость которых приведена в таблице 4.10.

Таблица 4.10 – Перечень израсходованных материалов

Наименование	Количество, шт.	Цена, руб.	Сумма, руб.
Тонер для картриджа принтера	1	250	250
Бумага (пачка 500 листов)	2	130	270
Диск CD-R	1	20	20
Итого, Вт:			540

Рассчитаем затраты на заработную плату, ЗП, руб по следующей формуле 4.5:

$$\text{ЗП} = t_p \cdot \text{Взп}, \quad (4.5)$$

где t_p – время разработки программы, мес.;

Взп – основная заработная плата разработчиков, руб.

Рассчитаем показатель ЗП по формуле (4.5)

$$\text{ЗП} = 5 \cdot 17000 = 85000 \text{ руб.}$$

Рассчитаем эксплуатационные затраты на разработку программного продукта, Зэксп, руб по следующей формуле 4.6:

$$\text{Зэксп} = T_{mv} \cdot \text{Зчк}, \quad (4.6)$$

где T_{mv} – время использования ПК при разработке программного продукта, час./сут.;

Зчк – стоимость одного часа работы ПК, руб.

Рассчитаем время использования ПК при разработке программного продукта, T_{mv} , час., по следующей формуле 4.7:

$$T_{mv} = P \cdot K_{дн} \quad (4.7)$$

где P – режим использования компьютера, час/сут.;

К_{дн} – количество рабочих дней за 20 недель, дней.

Рассчитаем показатель Т_{мв} по формуле 4.7:

$$T_{мв} = 8 \cdot 100 = 800 \text{ час.}$$

Рассчитаем показатель З_{эксп} по формуле 4.6:

$$З_{эксп} = 800 \cdot 8 = 6400 \text{ руб.}$$

Рассчитаем полную себестоимость программного продукта, С_{пп}, руб. по следующей формуле:

$$C_{пп} = З_{зп} + W_{н} + З_{эксп} + З_{пр} \quad (4.8)$$

где З_{зп} – затраты, связанные с заработной платой, руб.;

W_н – затраты на расходные материалы, руб.;

З_{эксп} – эксплуатационные затраты на разработку программного продукта, руб.;

З_{пр} – прочие затраты (амортизация, командировки и прочие расходы), руб.

Рассчитаем себестоимость программного продукта, С_п, руб. по следующей формуле:

$$C_{п} = (З_{зп} + W_{н} + З_{эксп} + З_{пр}) / N \quad (4.9)$$

где З_{зп} – затраты, связанные с заработной платой, руб.;

W_н – затраты на расходные материалы, руб.;

З_{эксп} – эксплуатационные затраты на разработку программного продукта, руб.;

З_{пр} – прочие затраты (амортизация, командировки и прочие расходы), руб.;

N – количество пользователей программы, чел.

Налог на доходы физических лиц НДФЛ = 13% и отчисления в фонды ОФ = 28%, из них ПФР = 20%, ФСС = 2,9%, ФФОМС – 5,1%, W_о = 0.34.

Рассчитаем затраты, связанные с заработной платой, $Z_{зп}$, руб. по следующей формуле:

$$Z_{зп} = ЗП + (ЗП \cdot W_{дд}) \quad (4.10)$$

где $ЗП$ – заработная плата, руб.

Рассчитаем прочие затраты (амортизация, командировки и прочие расходы), $Z_{пр}$, руб. по следующей формуле:

$$Z_{пр} = (Z_{зп} + W_{н} + Z_{эксп}) \cdot 0,3 \quad (4.11)$$

где $Z_{зп}$ – затраты, связанные с заработной платой, руб.;

$W_{н}$ – затраты на расходные материалы, руб.;

$Z_{эксп}$ – эксплуатационные затраты на разработку ПО, руб.

Для расчета полной себестоимости программного продукта необходимо определить затраты связанные с заработной платой и прочие затраты.

Рассчитаем затраты связанные с заработной платой по формуле 4.10.

$$Z_{зп} = 85000 + (40000 \cdot 0,13) = 96050 \text{ руб.}$$

Рассчитаем прочие затраты по формуле 4.11:

$$Z_{пр} = (96050 + 540 + 64000) \cdot 0,3 = 30897 \text{ руб.}$$

Рассчитаем полную себестоимость программного продукта по формуле (4.8)

$$C_{пп} = 96050 + 540 + 30897 + 6400 = 133887 \text{ руб.}$$

Рассчитаем себестоимость программного продукта по формуле 4.9:

$$C_{п} = (96050 + 540 + 30897 + 6400) / 3 = 44629 \text{ руб.}$$

Рассчитаем цену программного продукта, рассчитывая по заданной норме рентабельности ($R=30\%$), $Ц_{пп}$, руб. по следующей формуле:

$$Ц_{пп} = C_{пп} + (R \cdot C_{пп}) \quad (4.12)$$

где $C_{пп}$ – полная себестоимость программного продукта, руб.;

R – норма рентабельности, %.

Рассчитаем цену программного продукта по формуле 4.12:

$$Ц_{пп} = 133887 + (133887 \cdot 0,3) = 174053,1 \text{ руб.}$$

Рассчитаем балансовую прибыль разработчика, $БП$, руб. по следующей формуле:

$$БП = N \cdot (Ц_{пп} - C_{пп}) \quad (4.13)$$

где N – количество пользователей программы, чел.;

$Ц_{пп}$ – полная цена программного продукта, руб.;

$C_{пп}$ – полная себестоимость программного продукта, руб.

Рассчитаем балансовую прибыль разработчика по формуле 4.13:

$$БП = 3 \cdot (174053,1 - 133887) = 120498,3 \text{ руб.}$$

Рассчитаем чистую прибыль разработчика, $ЧП$, руб. по следующей формуле:

$$ЧП = БП - (N_{ст} \cdot БП) \quad (4.14)$$

где $N_{ст}$ – ставка налога на прибыль ($N_{ст} = 20\%$);

$БП$ – балансовая прибыль, руб.

Рассчитаем чистую прибыль разработчика по формуле 4.14:

$$ЧП = 120498,3 - (0,2 \cdot 120498,3) = 96398,64 \text{ руб.}$$

Рассчитаем капитальные затраты, связанные с разработкой программного продукта, $КЗ$, руб. по следующей формуле:

$$КЗ = (N \cdot Ц_{пк}) + КЗ_{пр} \quad (4.15)$$

где Цпк – цена компьютера, руб.;

N – количество компьютеров, шт.;

КЗпр – прочие капитальные затраты руб.

Рассчитаем капитальные затраты, связанные с разработкой программного продукта по формуле 4.15:

$$КЗ = (1 \cdot 20000) + 3500 = 23500 \text{ руб.}$$

Рассчитаем срок окупаемости, Ток, по следующей формуле:

$$Ток = КЗ/ЧП \text{ (4.16)}$$

где КЗ – капитальные затраты, связанные с разработкой программного продукта, руб.;

ЧП – чистая прибыль разработчика, руб.

Рассчитаем срок окупаемости для двух пользователей информационной системы по формуле 4.16:

$$Ток = 23500/96398,64 = 0,24 \text{ г.}$$

Т.о., срок окупаемости составляет 0,24 года или 2,88 месяца. Выделим основные рассчитанные экономические показатели в таблицу 4.11 .

Таблица 4.11 - Основные экономические показатели для 3-х пользователей

Показатель	Значение
Цена программного продукта, руб.	174053,1
Полная себестоимость программного продукта, руб.	133887
Чистая прибыль, руб.	96398,64
Срок окупаемости, лет	0,24

В заключение отметим, что, что затраты окупаются практически за полгода. Чистая прибыль (для 3-х пользователей) составляет 96398,64 руб. на 1 единицу программного продукта, а чистая прибыль (для 1-го пользователя) составляет 32132,88 руб. на 1 единицу программного продукта.

Заключение

При выполнении дипломной работы была смоделирована ИС «учета успеваемости и посещаемости студентов» в ТСПК. Разработанная система демонстрирует основные преимущества визуального моделирования. Моделирование обеспечивает более точную оценку необходимых ресурсов, четкую проработку планов и эффективное функционирование создаваемых систем.

Данная дипломная работа позволила изучить особенности создания баз данных, познакомиться с ней на концептуальном и понятийном уровне, а также приобрести навыки работы с пакетом программных продуктов MS |Office.

На основании подготовленных материалов было разработано программное обеспечение с рабочим названием: автоматизированная информационная система «учета успеваемости и посещаемости студентов».

После разработки данного программного обеспечения были получены следующие результаты:

1. повышение эффективности исполнения процессов, перечисленных выше, путем сокращения непроизводительных и дублирующих операций, операций, выполняемых «вручную», оптимизации информационного взаимодействия участников процессов;
2. повышение качества принятия управленческих решений за счет оперативности представления, полноты, достоверности и удобства форматов отображения информации;
3. развитие единого информационно-коммуникационного пространства сотрудников колледжа;
4. оптимизация и автоматизация процессов системы деятельности деканата;
5. предоставление доступа (в соответствии с предоставленными правами) к информации, содержащейся в различных информационных

системах, эксплуатирующихся в деканате;

6. обеспечение целостности, доступности и конфиденциальности информации.

При изучении данной предметной области, были рассмотрены различные системы по проблеме автоматизации деятельности деканата а так же СУЗа. Похожей системы выполняющей все выше описанные функции найдено не было. Система рассчитывающая статистические формы найдена не была.

В связи с чем и создана АИС, которая будет выполнять все функции и удовлетворять всем требованиям необходимым для автоматизации деятельности деканата, а так же даст возможность сэкономить деньги. С учетом затрат на создание данного ПП, общая смета программного продукта составляет 174053,1 руб.

Информационный база данных является актуальной на сегодняшний день, так как все СУЗы нуждаются в автоматизации управления. Разработанная программа устойчиво выполняет все свои функции, но для использования ее в сузах, она требует некоторых доработок: программным путем обеспечить безопасность данных, расширить возможности поиска (поиск по двум и более полям), и т.д.

Также были рассмотрены технологии, по которым создаются информационные системы.

Созданная информационная система разбита на проекции, которые не сильно связаны между собой, что позволяет пользоваться ей, даже если какая-то из проекций системы не нужна учебному заведению; также можно добавлять дополнительные проекции, которые, я думаю, в дальнейшем обязательно появятся. Таким образом, все поставленные задачи были проработаны и цель дипломной работы достигнута.

Список источников литературы

1. Баженов Р.И. О разработке информационной системы оценки учебных достижений студентов / Баженов Р.И., Баженова Н.Г., Белов И.В., Кардаш А.С. // Современные научные исследования и инновации. 2014. №12 [Электронный ресурс]. URL: <http://web.snauka.ru/issues/2014/12/41514> (дата обращения 25.09.2015).
2. Беззуб М.Д., Чуйко О.И. Сравнительный анализ наиболее популярных облачных серверов // Информационные системы и технологии: материалы городской конференции студентов и школьников. 15 апреля 2014 года. Хабаровск: РИЦ ХГАЭП, 2013. С. 7-11.
3. Белозёров И.Л., Белозёрова С.И. Применение бально-рейтинговой системы оценки знаний студентов в техническом высшем учебном заведении. // Педагогика и психология: проблемы, идеи, инновации: материалы II
4. Введение в UML. Лекции. [Электронный ресурс] - Режим доступа: <http://www.intuit.ru.-30.10.2010>
5. Вендров А.М. Проектирование программного обеспечения
6. Вендров, А.М. CASE-технологии. Современные методы и средства проектирования информационных систем : учебник / А.М. Вендров. - М. : Финансы и статистика, 2006. - 478 с.
7. Глушаков, С.В. База данных : учебник / С.В. Глушаков. - М. : Фолио, 2006. - 209 с.
8. Ефимова И.С., Калитин С.В. К вопросу о создании наглядного инструмента для автоматического расчёта рейтинга успеваемости студентов // Перспективы развития информационных систем и технологий на Дальнем Востоке: материалы городской науч.-ислед. конференции студентов и школьников. 10 апреля 2015 года. Хабаровск: РИЦ ХГАЭП, 2015. С. 37-45.
9. Ещенко Р.А., Чуйко О.И. Применение облачных технологий для контроля успеваемости студентов в учебном процессе // сборник статей IV Международной науч.-практ. конф. - Пенза: Приволжский дом знаний, 2014. -

С. 41-45.

10. Избачков, Ю.С. Информационные системы для вузов: учебное пособие / Ю.С. Избачков, В.Н. Петров. - СПб. : Питер, 2005. - 656 с.

11. Карташов, С.С. Образование и XXI век: информационные и коммуникационные технологии : учебное пособие / С.С. Карташов. - М.: Наука, 2005. - 191 с.

12. Кравченко К.В. Преимущества организации работы с контингентом студентов в вузе на основе единой корпоративной системы / К.В. Кравченко // Проблемы управления качеством образования: сб.ст./Пенза: МНИЦ ПГСХА, 2014. С 49-52

13. Красильникова М.В.. Проектирование информационных систем: Учебное пособие - М.: МИСиС, 2004. - 106 с.

14. Маклаков С.В. Erwin расширяет свои возможности. - Компьютер пресс, 3, 1998. - 200 с.

15. Марка Д.А., МакГоуэн К. Методология структурного анализа и проектирования. М., "МетаТехнология", 1993. -213 с.

16. Марков, А.С. Введение в теорию и методологию : учебник / А.С. Марков, К.Ю. Лисовский. - М.: Учебник, 2006. - 512 с.

17. Международной заочной научно-практической конференции. 10 февраля 2014 г. - Чебоксары: ЦДИП «INet», 2014. - С. 7-9.

18. Океанова З. К. Основы экономической теории -Москва, 2013 г.

19. Орлов С. А. Технологии разработки программного обеспечения, 2-е изд. – СПб.: «Питер», 2013 г.

20. Орлов С. К. Технологии разработки программного обеспечения - Санкт-Петербург, 2012 г.

21. Отчет о результатах самообследования государственного бюджетного профессионального образовательного учреждения Краснодарского края «Туапсинский социально-педагогический колледж» за 2015 год.

22. Официальный сайт государственного бюджетного профессионального образовательного учреждения Краснодарского края

«Туапсинский социально-педагогический колледж»: <http://www.tspk.tuapse.ru/>

23. Павлова Ж.Г. Модульно-рейтинговая система - средство мотивации учебной деятельности студентов вуза // Инновационный подход к обучению и воспитанию: Материалы V Международного фестиваля методических идей. 24 октября 2014 г. Негосударственное образовательное учреждение дополнительного профессионального образования «Экспертно-методический центр». 2014. С. 483-485.

24. Петелин Е.А., Белянкин Ю.В., Стрельникова Е.О. Автоматизация контроля текущей успеваемости и аттестации студентов // Образовательные технологии. 2014. №2. С. 102-106.

25. Полещук Н. Н. AutoCAD 2005, Санкт – Петербург, 2014 г.

26. Правосудов Р.Н., Акамова Н.В., Голяев С.С. автоматизация балльно-рейтинговой системы ВУЗа // Фундаментальные исследования. – 2015. – № 11-2. – С. 303-307

27. Рудаков А. В. Технология разработки программных продуктов - Москва, 2012 г.

28. Семакин И. Г., Шестаков А. П. Основы программирования - Москва, 2011 г.

29. Сироткин А.Ю. Преимущества использования облачных технологий при подготовке специалистов в вузе // Вестник ТГУ, т.18, вып.1, 2013. С. 243-244.

30. Скоз Е.Ю. Программные средства моделирования в САПР. Конспект лекций. 1999. - 245 с.

31. Соколова Т.Н., Демина А.В. Использование облачных сервисов в образовательном процессе // Информационная безопасность регионов. 2013. №2 (13). С. 14-16.

32. Устав государственного бюджетного профессионального образовательного учреждения Краснодарского края «Туапсинский социально-педагогический колледж» Принят общим собранием работников и представителей обучающихся протокол от «07»ноября 2013 г. № 3

33. Фролкин П.П., Шатило Я.С. Вычисления на облаках // Информационная безопасность регионов. 2013. №2 (13). С. 10-13.
34. Чуйко О.И., Белозерова С.И. Разработка информационной системы учета успеваемости студентов на основе облачных технологий // Интернет-журнал «НАУКОВЕДЕНИЕ» Том 7, №5 (2015)
35. Чуйко О.И., Ещенко Р.А. Электронный журнал: анализ применения в школах и перспективы внедрения в вузах // Международный академический вестник, 2014. №6. С. 27-31.
36. Шмуллер Д. Освой самостоятельно UML за 24 часа.[Текст]: М. У. Боггс, М. Боггс «UML и Rational Rose 2002»-Издательство «ЛЮРИ», 2004.-301 с.