

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ
ГУМАРБЕКА ДАУКЕЕВА»
Кафедра «IT - Инжиниринг»

«ДОПУЩЕН К ЗАЩИТЕ»
Зав. кафедрой PhD, доцент Досжанова А.А.
« ____ » _____ 2020 г.

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка информационной системы «Кафедра»

Специальность 5В060200 Информатика

Выполнил Сақан Ерлану Мұхтарханұлы

Группа: ИНФ-16-2

(Ф.И.О.)

Научный руководитель: PhD, доцент Досжанова А.А.

(ученая степень, звание, Ф.И.О.)

Консультанты:

по экономической части: к.э.н профессор Габелашвили Кахабер Ревазович

(ученая степень, звание, Ф.И.О.)

« ____ » _____ 2020 г.

(подпись)

по безопасности жизнедеятельности: ассистент Тыщенко Елена Михайловна

(ученая степень, звание, Ф.И.О.)

« ____ » _____ 2020 г.

(подпись)

по применению вычислительной техники:

ст. преп. Майкотов Мухит Нурдаулетович

(ученая степень, звание, Ф.И.О.)

« ____ » _____ 2020 г.

(подпись)

Нормоконтролер: ст. преп. Абсатарова Бибигуль Рыскуловна

(ученая степень, звание, Ф.И.О.)

« ____ » _____ 2020 г.

(подпись)

Рецензент: _____

(учёная степень, звание, Ф.И.О.)

« ____ » _____ 2020 г.

(подпись)

Алматы 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ
ГУМАРБЕКА ДАУКЕЕВА»

Институт систем управления и информационных технологий»

Кафедра «IT - Инжиниринг»

Специальность 5В060200-Информатика

ЗАДАНИЕ

на выполнение дипломного проекта

Студенту: Сақан Ерлану Мұхтарханұлы

Тема работы: Разработка информационной системы «Кафедра»

Утверждена приказом по университету № 147 от «11» ноября 2019 г.

Срок сдачи законченного проекта «___» _____ 202__ г.

Исходные данные к проекту (требуемые параметры результатов исследования (проектирования) и исходные данные объекта): справочник по разработке программ на C#, интегрированная среда Visual Studio, данные преддипломной практики.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта: современное состояние системы на предприятии, исследование предметной области, проектирование информационной системы, разработка программного продукта, вопросы безопасности жизнедеятельности и охраны труда, экономическая эффективность работ по стандартизации.

Перечень графического материала (с точным указанием обязательных чертежей): представлены 10 таблиц, 69 иллюстраций.

Основная рекомендуемая литература:

1 Перлова, О.Н. Проектирование и разработка информационных систем: Учебник / О.Н. Перлова, О.П. Ляпина, А.В. Гусева. - М.: Academia, 2017. - 416 с;

2 Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 с;

3 Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 с.

Консультации по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
Нормоконтроль	Абсатарова Б.Р.	18.05.2020	
Программное обеспечение	Майкотов М.Н.	14.05.2020	
Подготовка экономической части дипломного проекта	Габелашвили К.Р.	24.04.2020	
Подготовка части БЖД	Тыщенко Е.М.	24.04.2020	

График

подготовки дипломного проекта:

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
Исследование предметной области	01.11.2019-20.12.2019	
Проектирование ориентированной информационной системы	21.12.2019-20.02.2020	
Разработка программного продукта	21.02.2020-30.04.2020	

Дата выдачи задания «01» ноября 2019 г.

Заведующий кафедрой _____ Досжанова А.А.
(подпись) (Ф.И.О.)

Руководитель работы _____ Досжанова А.А.
(проекта) (подпись) (Ф.И.О.)

Задание принял к исполнению _____ Сақан Е.М.
студент (подпись) (Ф.И.О.)

Аңдатпа

Кез келген ұйымның жетістігінің басты шарты - басқару әдістерін жетілдіру. Құралдардың бірі - бұл бизнестің тиімділігін арттыруға мүмкіндік беретін ақпараттық жүйелерді енгізу.

Бұл дипломдық жоба «IT-инжиниринг» кафедрасы үшін ақпараттық жүйені дамытуды талқылайды. Жүйе әдіснамалық жұмыстардың тізімін құру, дерекқордан экспорттау және импорттау, сонымен қатар жүйеде өңдеуді автоматтандыруға мүмкіндік береді, бұл өнімділікті арттырады және жұмыс процесін жылдамдатады.

Жүйе C # тіліндегі Visual Studio-да жасалған.

Аннотация

Основным условием успешной деятельности любой организации является совершенствование методов управления. Одним из инструментов является внедрение информационных систем, которые позволяют повысить эффективность деятельности.

В данном дипломном проекте рассмотрена разработка информационной системы для кафедры “IT-Инжиниринг”. Система позволяет автоматизировать процесс создания списка методических трудов, экспорт и импорт из базы данных, а также редактирование в самой системе, что увеличивает продуктивность и ускоряет процесс работы.

Система разработана в среде Visual Studio на языке C#.

Abstract

The main condition for the success of any organization is the improvement of management methods. This allows you to increase the efficiency of activities.

This graduation project discusses the development of a system for the department of IT Engineering. The system allows you to automate the process of creating a list of methodological works, as well as edit in the system itself.

System available in Visual Studio in C #.

Содержание

Введение	8
1 Теоретическое обоснование разработки информационной системы	9
1.1 Анализ предметной области	9
1.2 Существующие систем и их проблемы	9
1.3 Решение проблем	17
2 Описание и обоснование выбора языка программирования и СУБД	22
2.1 Проектирование информационной системы	22
2.2 Анализ современных языков программирования	26
2.3 Выбор языка программирования С#	26
2.4 Особенность использования СУБД	28
3 Разработка информационной системы	29
3.1 Программная реализация	29
3.2 Системные требования	39
3.3 Интерфейс	40
4 Обоснование эффективности внедрения проекта	47
4.1 Техничко - экономическое обоснование дипломных работ, связанных с разработкой программного продукта (ПП)	47
4.2 Трудоемкость разработки ПП	48
4.3 Расчет затрат на разработку ПП	48
4.4 Определение договорной цены ПП	51
4.5 Расчет результатов от создания и использования ИС	52
4.6 Расчет основных показателей экономической эффективности	54
5 Безопасность жизнедеятельности	56
Заключение	61
Список литературы	62
Приложение А	63
Приложение Б	Ошибка! Закладка не определена.

Введение

Информационные системы в данный момент активно связаны со сферой деятельности человека. По данной причине есть необходимость внедрения информационных систем в сферы, где они мало развиты или слабо используются. Такое внедрение должно позволить уменьшить затраты, увеличить производство труда, а также время на обработку данных.

Информационная система в дипломной работе выполнена по заказу кафедры “IT-Инжиниринг” Алматинского Университета Энергетики и Связи имени Гумарбека Даукеева.

Целью дипломного проекта является разработка системы учета методических трудов ППС кафедры. Для кафедры тема стала наиболее актуальна, так как начался переход в цифровую эпоху. Сейчас нет необходимости хранения в бумажной форме документов. Информационная система предназначена для хранения в цифровом формате.

Программа должна обеспечивать:

- хранение методических трудов;
- структурирование информации;
- формирование отчетов.

Для достижения вышеуказанной цели необходимо решить следующие задачи:

- провести анализ существующих систем;
- разработать программное обеспечение;
- провести экономическую оценку эффективности информационной системы.

В настоящее время существуют похожие информационные системы, которые помогают автоматизировать процесс хранения методических трудов, а также вывода методических трудов.

Большинство из них имеют довольно широкий функционал, но в них есть недостатки.

Основные недостатки:

- функционал программы не удовлетворяет всем требованиям работы;
- информационные системы требуют доработку и на это могут уйти дополнительные затраты.

После изучения структуры кафедры, а также выявления основных проблем, было принято решение о создании информационной системы “Кафедра” в виде базы данных с интерфейсом.

1 Теоретическое обоснование разработки информационной системы

1.1 Анализ предметной области

Кафедра IT-инжиниринга входит в состав Института систем управления и информационных технологий НАО “Алматинский Университет Энергетики и Связи имени Гумарбека Даукеева”, является общеобразовательным, учебно-научным подразделением, созданным в целях осуществления учебной, методической, научно-исследовательской и воспитательной работы среди обучающихся.

Кафедра ITE осуществляет свою деятельность в соответствии с действующим законодательством Республики Казахстан, приказами и распоряжениями Министерства образования и науки РК, Уставом Университета, решениями Ученого совета Университета, приказами и распоряжениями руководства, другими нормативными документами внешнего и внутреннего характера, определяющими порядок деятельности Университета, настоящим Положением и должностными инструкциями.

Основными задачами Кафедры ITE являются:

- 1) учебно-методическое обеспечение учебных дисциплин кафедры, подготовка учебников, учебных пособий и методических указаний по изучению дисциплин, материалов текущего, промежуточного и итогового контроля знаний по всем формам обучения;
- 2) рекомендация для публикаций утвержденных научных работ;
- 3) изучение, обобщение и распространение опыта работы лучших преподавателей;
- 4) проведение работ по развитию, модернизации и укреплению материально-технической базы кафедры включая постоянное совершенствование учебно-лабораторного оборудования.

Функциями Кафедры ITE являются:

- 1) рассмотрение и утверждение индивидуальных планов учебной, научной, учебно-методической и других работ ППС кафедры;
- 2) методическое обеспечение и организация самостоятельной работы студентов по дисциплинам Кафедры ITE.

1.2 Существующие систем и их проблемы

Информационные системы является системой, которая создана для хранения, поиска и обработки информации. Информационные системы созданы для обеспечения информацией определенный круг людей. В основном данная система предназначена для оптимизации рабочего процесса и его автоматизации. Еще одной важной отличительной возможностью является уменьшение бумажной волокиты. На сегодняшний день есть достаточно много систем, которые позволяют хранить и обрабатывать информацию. Правда есть некоторые сложности и проблемы в

использовании этих систем. Мы имеем такие сервисы, как Google Drive, Dropbox и Platonus.

Главной проблемой данных сервисов является:

- безопасность хранения;
- универсальность;
- объем;
- постоянное подключение к интернету.

Для крупных организаций безопасность является неотъемлемо важным элементом и непозволительно допускать утечку информации. Сегодня информация имеет важный характер. Любые действия совершаются за счет имеющейся информации. Утечка информации может привести к краху организации, так как другие организации могут собрать всю информацию и использовать ее против этой же организации. Загрузка данных в сервисы может привести к утечке, так как данные находятся во всемирной паутине. Человек, поставивший цель получения этих данных, может найти пути их получения. В данный момент поток информации, которые проходят, шифруются разными методами. Главная задача организации является надежность потока информации[1].

Для физических лиц Google Drive и Dropbox являются наиболее предпочитаемыми и удобными. Сервисы универсальные и позволяют работать в разных сферах деятельности человека. В основном главная задача физического лица является хранение личных файлов в данных сервисах, куда входят: личные фото, документы, видео и прочее, но вот данное преимущество в универсальности идет в минус для государственных учреждений, организаций, частных компаний и для остальных структур. Функционал данных сервисов может не удовлетворять всем требованиям работы. Каждая организация имеет свою уникальную сферу деятельности, и хранения данных может быть не единственной нужной функцией для обеспечения работоспособности сферы. Возьмем Google Drive.

Сервис достаточно удобный и имеет ряд полезных функций, вроде встроенного инструмента создания документации: документы, таблицы, презентации и формы (рисунок 1.1).

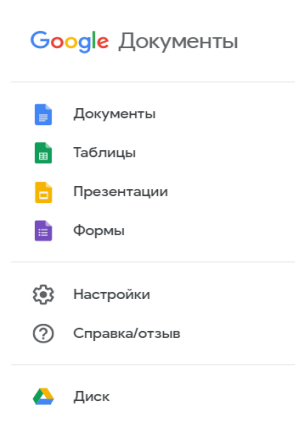


Рисунок 1.1 – Элементы в Google Drive

В документах есть возможность создавать печатные материалы: резюме, письмо и доклады (рисунок 1.2).

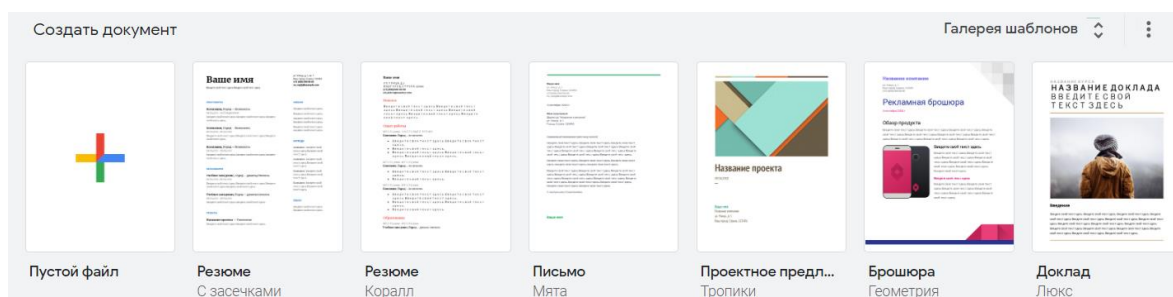


Рисунок 1.2 – Шаблоны создания документа

Разные таблицы: расходы, календари, список дел (рисунок 1.3).

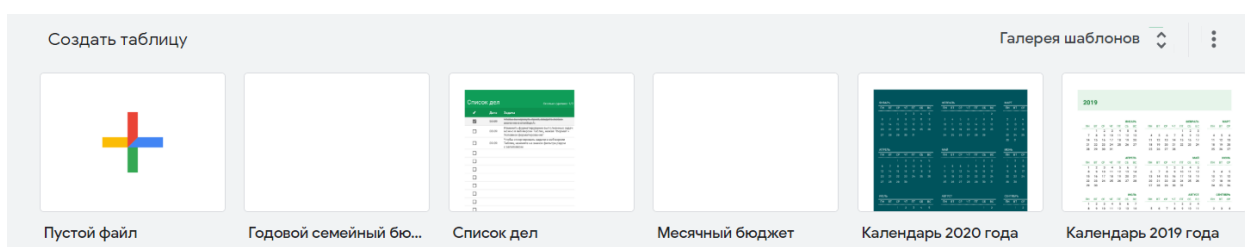


Рисунок 1.3 – Шаблоны создания таблиц

Презентации: портфолио, представление, фотоальбом (рисунок 1.4).

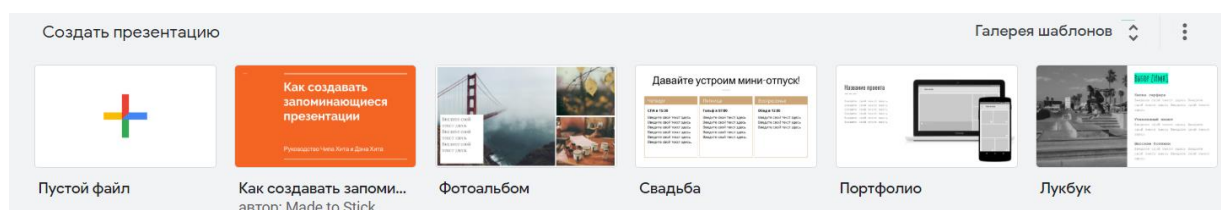


Рисунок 1.4 – Шаблоны создания презентаций

Формы: заявки, заказы (рисунок 1.5).

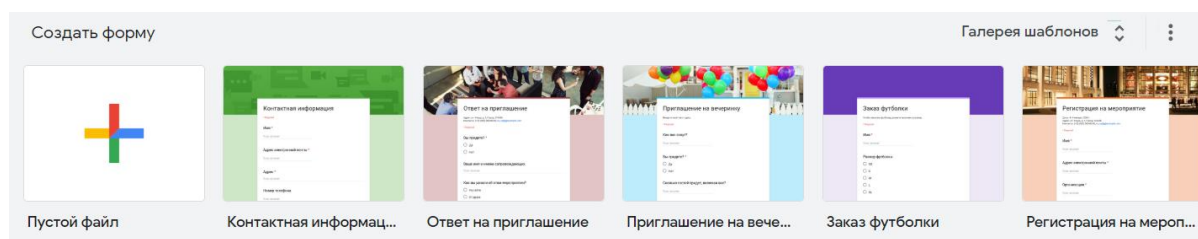


Рисунок 1.5 – Шаблоны создания форм

В этом и кроется проблема. Сервис слишком универсален и не адаптирован для определенной среды, что не очень эффективно. Возьмем, к примеру, компьютеры средней сборки. Такими ноутбуками могут

пользоваться абсолютно все желающие, так как адаптированы под повседневную жизнь пользователя.

Обычному пользователю должно хватить использование интернета, офисных приложений и предустановленного функционала: калькулятор, медиаплеер и просмотр картинок (рисунок 1.6).

Имя	Состояние	ЦП	Память	Диск	Сеть	GPU
Calculator (2)	φ	0%	2,3 МБ	0 МБ/с	0 Мбит/с	0%
Discord (32 бита) (3)		0,3%	106,5 МБ	0 МБ/с	1,8 Мбит/с	0%
Google Chrome (48)		3,3%	1 252,7 МБ	0,4 МБ/с	0 Мбит/с	0,1%
Groove Music (3)		0%	42,2 МБ	0,1 МБ/с	0 Мбит/с	0%
Microsoft Edge (19)	φ	0%	15,6 МБ	0 МБ/с	0 Мбит/с	0%

Рисунок 1.6 – Диспетчер задач (демонстрация работы процессов)

Теперь стоит упомянуть отдельную категорию людей: разработчики, монтажеры. Сейчас компьютеры играют важную роль в сфере деятельности человека. Для каждого из них нужны свои собственные комплектующие для эффективности их действий. Разработчики и монтажеры требуют более мощное железо для эффективной работы. В свою очередь монтажеры требуют более мощный графический процессор, так как приходится часто работать с аудио и видео файлами, что требует много вычислительной мощности. Особенно сейчас, когда разрешение экранов достигает 3840x2160 точек, которое в четыре раза превышает количество пикселей в формате Full HD, если еще вспомнить, что в ЭЛТ-телевизорах разрешение было 640x480, а может и меньше. Монтажеры не смогут эффективно работать с компьютерами со встроенным графическим процессором (рисунок 1.7).

Имя	Состояние	ЦП	Память	Диск	Сеть	GPU
Приложения (9)		29%	92%	3%	0%	62%
> Pr (5)		4,4%	1 620,1 МБ	0,1 МБ/с	0 Мбит/с	61,6%

Рисунок 1.7 – Загруженность системы процессом Adobe Premiere Pro

У разработчиков проблема похожа. Для эффективной работы больше нужен центральный процессор, чем графический (рисунок 1.8).

Имя	Состояние	ЦП	Память	Диск	Сеть	GPU
> Microsoft Visual Studio 2019 (3...		1,9%	316,7 МБ	0,1 МБ/с	0 Мбит/с	0%

Рисунок 1.8 – Загруженность системы процессом Visual Studio Code

Расход ресурсов приложения в среде разработки Visual Studio (рисунок 1.9).

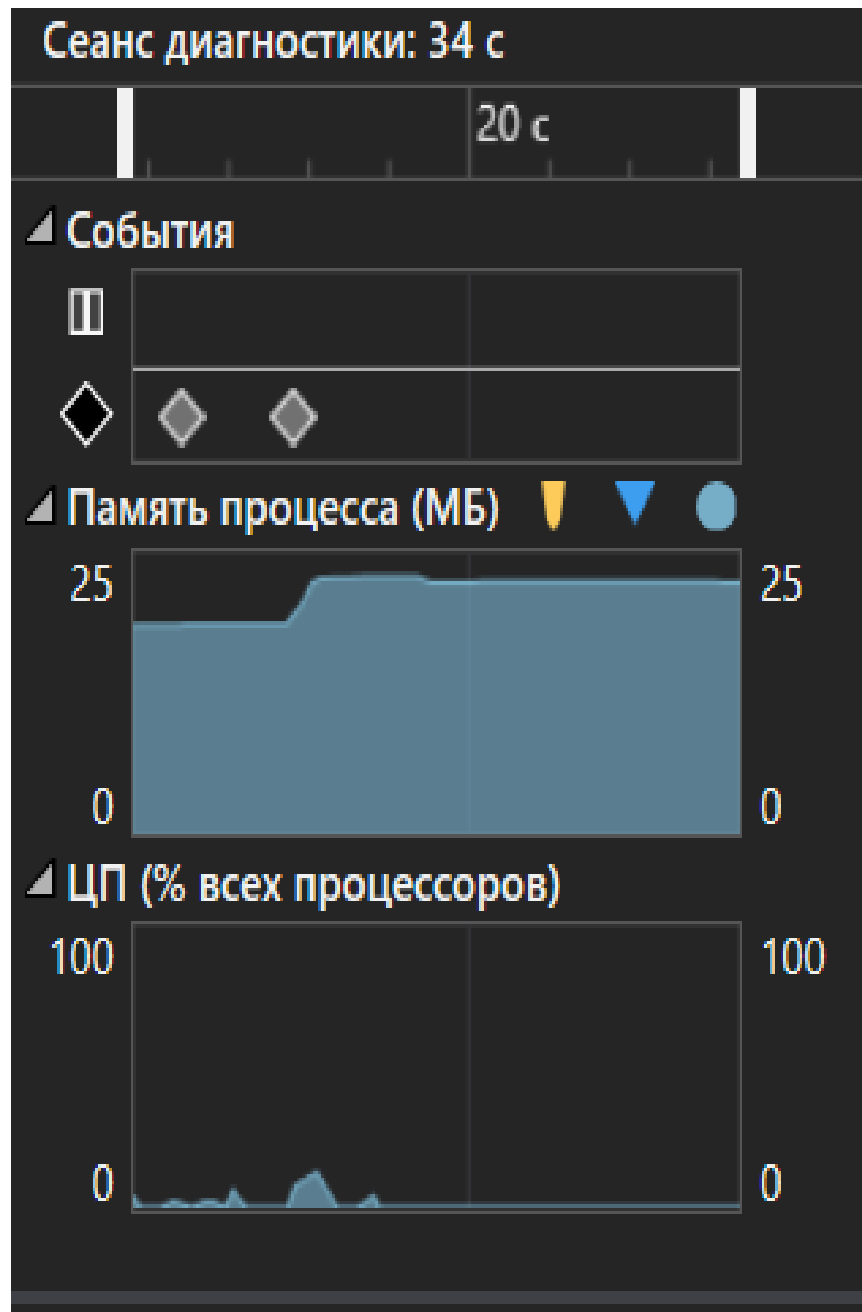
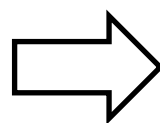


Рисунок 1.9 – Расход ресурсов в Visual Studio

Здесь приведены примеры для сравнения комплектующих разных сфер в целях понятия эффективности (рисунок 1.10).



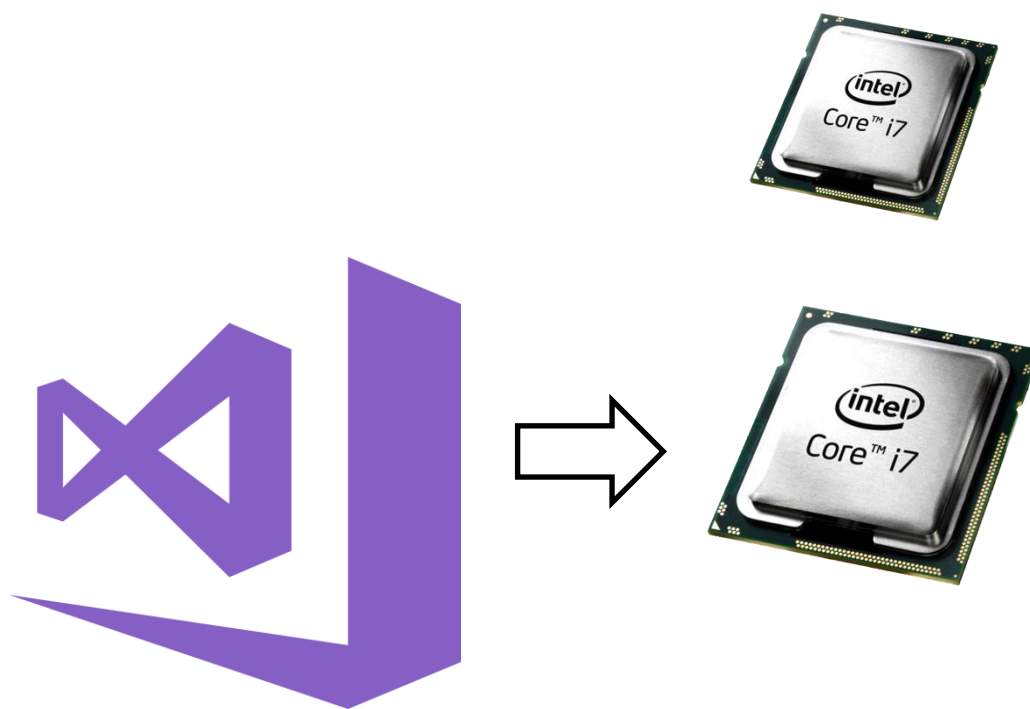


Рисунок 1.10 – Требование к определенным приложениям

Разработчикам эффективно не тратить средства на графический процессор и могут вложить больше средств на центральный процессор. Здесь ситуация имеет некоторую схожесть. В этой ситуации дается абсолютно пустой бланк, который нужно вручную заполнить, так как он не адаптирован под сферу деятельности предприятия. Необходимо тратить много времени на выполнение работы, которую можно было потратить более эффективно. К примеру, кафедре приходится вручную заполнять список трудов преподавателей. У каждого преподавателя может быть больше сотни трудов.

Вам нужно заполнить (рисунок 1.11):

- 1) наименование;
- 2) издательство, журнал или номер авторского свидетельства, патента;
- 3) количество печатных листов или страниц;
- 4) соавторы.

СПИСОК
научных и научно-методических трудов
к.т.н., доцента кафедры «IT-инжиниринг» НАО «Алматинский университет энергетик
и связи имени Гумарбека Даукевича»
Иванова Иван Иванович

№ п/п	Наименование научного или методического труда	Издательство, журнал (название, номер, год, страницы) или номер авторского свидетельства, патента	Кол-во печатных листов или страниц	Соавторы
Учебники, рекомендованные Министерством образования и науки Республики Казахстан				
1	Исследования в области информатики	Алматы: КазННТУ имени К.И.Сатпаева, 2016 – 256 с.	236	Петрова И.В.
Учебные пособия, рекомендованные Республиканским учебно-методическим советом Министерства образования и науки Республики Казахстан				
2	Прикладная информатика	Алматы: КазННТУ имени К.И.Сатпаева, 2015. – 496 с.	496	Сипоров В.П.
Монографии				
3	Информатика	Алматы: Издательство LEM, 2015.–130с.	132	Иванов А.И.
Учебники и учебные пособия, рекомендованные ученым советом вуза				
4	Алгоритмы и программирование	Издательство LEM, 2015. – 104 с.	104	Кузнецова О.Ю.
Публикации в изданиях, имеющих печатную или электронную версию в базах данных информационной компании Thomson Reiter (WebOfScience, ThomsonReuters):				
5	Developing and customizing university business intelligence cloud	International Conference on Cloud Computing Technologies, Applications and Management (ICCS TAM), Dubai, 2015. – P. 229-233. Available online at: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6488104&number=6488030	3	Ахметов В.В., Ибраева Г.А., Иванова И.И., Малигулин А.А., Мухоморов К.К.
Публикации в базах данных Scopus, Pubmed, zbMath, MathSciNet, Agric, Georef, Astrophysical Journal:				
6	Systems of Protection of Computer Networks from Distributed Network Attacks to Denial of Service	Research Journal of Applied Sciences (RJAS), Dubai, 2015. – Vol. 10, Issue 2. – P. 49-53. DOI:10.3923/rjas.2015.49.53	5	Иванова И.И., Малигулин А.А., Мухоморов К.К., Оспанов Е.А.
Публикации в материалах конференций, индексруемых в базах Web of Science, Scopus:				
7	Perspectives of Multiple Reduction of Biometrical Test Selection Scopes While Increasing Network Dimension of Pearson's Private Criteria	15th International Conference on Control, Automation and Systems (ICCAS 2015), Busan, Korea, 2015. – P. 1333-1336	4	А. Иванов, А. Малигулин, З. Абдыгазыр, К. Мухоморов
Публикации в изданиях, рекомендуемых ККСОН МОН РК (кроме материалов конференций):				
8	Разработка структуры интегрированной on-line информационной системы возобновляемых источников энергии	Вестник КазННТУ имени К.И.Сатпаева, 2015. – №1(107). – С. 150-157.	8	Балтабаева П.Ш., Киселева О.В., Харитонов П.П.

Автор

И.И.Иванов

№ п/п	Наименование научного или методического труда	Издательство, журнал (название, номер, год, страницы) или номер авторского свидетельства, патента	Кол-во печатных листов или страниц	Соавторы
Публикации в других научных изданиях Республики Казахстан (кроме материалов конференций):				
9	Базовые модели эталонных величин для систем обнаружения вторжений	Вестник МКТУ им. А.Ясаи. Серия общественные науки. 2015. – №5-6. – С.15-16	2	Абдраманов Р. Корченко А.А., Жумагалиева Н.
В научных изданиях других стран:				
10	Моделирование длинных биометрических кадров, воспроизводимых корреляционных связи выходящих данных нейросетевых преобразователей	Нейрокомпьютерная техника. – М.: Изд-во «Радиоотехника». 2012. – №3. – С. 40-43.	4	Болчикин В.И., Куликов С.В., Малигулин Е.А.
Публикации в конференциях стран дальнего и ближнего зарубежья:				
11	Statistical Description of Output States of the Neural Network "Biometrics-code" Transformers	Progress In Electromagnetics Research Symposium (PIERS), Moscow, Russia, 2012. – P. 62-66	5	А. Иванов, В. Фунтиков, А. Малигулин, Т. Урне
12	Application of Iterative Algorithm of Training of Single Neuron in Biometric Appendices	International Conference on Innovative Trends in Multidisciplinary Academic Research (ITMAR-2014), Istanbul, Turkey, 2014. – Vol. 1. – P. 596-601. Available online at: http://www.globaliummatters.org/wp-content/uploads/2014/12/ITMAR-14-596.pdf	6	З. Абдыгазыр, А. Иванов, А. Малигулин, К.Мухоморов
Публикации в конференциях, симпозиумах, семинарах, проведенных на территории Республики Казахстан:				
13	Анализ методов экспертного оценивания для систем обнаружения вторжений	Група II Международной научно-практической конф. «Информационные и телекоммуникационные технологии: образование, наука, практика» – Алматы: КазННТУ им. К.И.Сатпаева, 2015. Том II. – С. 28-31.	4	Корченко А.А., Ахметова С.Т., Жумагалиева Н.
Зарубежные патенты				
14	Анализ методов экспертного оценивания для систем обнаружения вторжений	Патент №123456789 и т.д. от 12.12.2015		Корченко А.А., Ахметова С.Т., Жумагалиева Н.
Патенты и авторские свидетельства				
15	Анализ методов экспертного оценивания для систем обнаружения вторжений	Патент №123456789 и т.д. от 12.12.2015 г.		Корченко А.А., Ахметова С.Т., Жумагалиева Н.
Методические указания, сборники лекций				
16	Инициальное моделирование. Методические указания к выполнению лабораторных работ для студентов	Алматы: АУЭС, 2019. – 33 с.	2,062 5	

Автор

И.И.Иванов

Рисунок 1.11 – Шаблон документации методических трудов

На каждую строку примерно уходит минута. В этом документе есть больше сотни работ. Человеку требуется для этого больше часа для заполнения таблицы, когда можно было бы все сделать быстрее.

Стоит отметить, что для работы таких сервисов требуется постоянное подключение к интернету. Отсутствие или плохое соединение делает просмотр и редактирование файлов практически невозможным, что также снижает эффективность работы и может привлечь к снижению трудоспособности, а также есть шанс потери доступа к своей документации, если все было бы загружено в сервис.

И самой главной проблемой является стоимость таких сервисов (рисунок 1.12).

15 ГБ	100 ГБ	200 ГБ	2 ТБ
Бесплатно	690 KZT в месяц	990 KZT в месяц	3 390 KZT в месяц
Текущий план	Вы можете заплатить за год вперед: 6 900 KZT в год	Вы можете заплатить за год вперед: 9 900 KZT в год	Вы можете заплатить за год вперед: 33 900 KZT в год
Что включено	Преимущества Google One	Преимущества Google One	Преимущества Google One
✓ Объем хранилища: 15 ГБ	✓ Объем хранилища: 100 ГБ	✓ Объем хранилища: 200 ГБ	✓ Объем хранилища: 2 ТБ
	✓ Поддержка от Google	✓ Поддержка от Google	✓ Поддержка от Google
	✓ Доступ для семейной группы	✓ Доступ для семейной группы	✓ Доступ для семейной группы
	✓ Доп. преимущества подписки	✓ Доп. преимущества подписки	✓ Доп. преимущества подписки

Рисунок 1.12 – Тарифы в Google Drive

Moodle является системой управления электронным обучением. Система известна своим управлением обучения или виртуальной обучающей средой. Представляет собой веб-приложение, предоставляющее возможность создавать сайты для онлайн-обучения.

Интерфейс системы (рисунок 1.13).

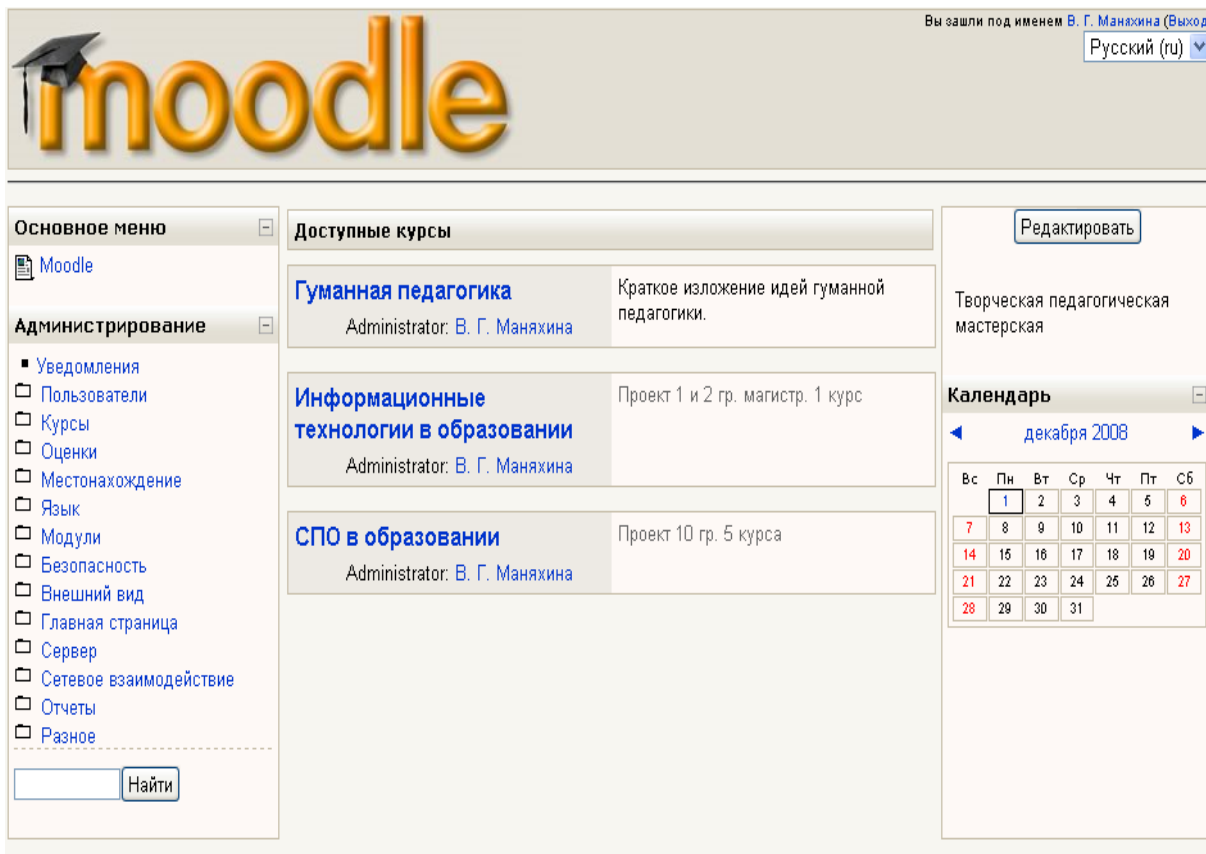


Рисунок 1.13 – Интерфейс Moodle

Вместе с тем, Moodle имеет значительный недостаток: в системе не предусмотрены группы уровня сайта, что делает очень сложным учет студентов разных специальностей. Группы в Moodle существуют не для управления правами доступа к курсам, а для разделения групп слушателей в одном курсе. Чтобы одни слушатели не видели активность других. Группы создаются внутри курса и не могут быть перенесены в другие.

Кроме этого, оценками слушателя можно оперировать только внутри курса. Нет возможности составить итоговую ведомость, например, по всем дисциплинам семестра, да и само понятие семестра в базовой версии системы отсутствует.

Из сказанного можно сделать вывод, что Moodle является системой, ориентированной на западную модель обучения: изучение одного курса несколькими группами слушателей, в то время как для организации и управления учебным процессом отечественного ВУЗа, система дистанционного обучения должна быть ориентированной на приоритетное использование учебных групп.

Одно из достоинств системы Moodle, вынесенное в название – модульность, делает вышеуказанные недостатки несущественными, так как есть возможность разработать надстройку любой сложности (модуль), которая позволит управлять большими контингентами студентов, обучающимися по нескольким специальностям.

Система Moodle оперирует объектами “курс”, “пользователь, роль пользователя в курсе “слушатель”, “преподаватель”, чего недостаточно для автоматизации учебного процесса отечественного ВУЗа. Для сохранения целостности структуры Moodle и возможности использования новых версий, необходимо было создать систему Деканат в виде отдельного блока, сосредоточив в нем все функции управления учебным процессом.

В Moodle предусмотрен штатный инструмент, предназначенный для оперирования группами студентов за пределами одного курса. Этот инструмент называется “метакурс”, который позволяет подписывать и отписывать всех студентов одного курса на другой курс в одно действие. При этом система учебных единиц становится двухуровневой:

Первый уровень – собственно «Курс», содержащий произвольную информацию, разбитую на блоки, на который регистрируются студенты.

Второй уровень – метакурсы, на которые регистрируются дочерние курсы.

Достоинство этой схемы заключается в том, что при регистрации курса, в “метакурсе” автоматически оказываются зарегистрированными все студенты, записанные на курс. Недостатками – то, что студенты из разных курсов регистрируются все вместе, без разбивки на группы, что делает работу преподавателей очень затруднительной.

1.3 Решение проблем

Одной из особенностей специальности “Информатика” является овладения навыками разработки информационной системы. Данная система позволяет уменьшить бумажную волокиту кафедры. Данная проблема является существенной на сегодняшний день. Есть много способов решения данной проблемы. Одним из них является создание информационной системы. Создание информационной системы должна позволить существенно уменьшить работу с бумагой, так как все документы расположены повсюду и должна обеспечить освобождения некоторой площади от них.

Не стоит еще забывать, что такой подход еще является экологической катастрофой. Каждый год уходит тонны бумаг для хранения документов. Приходится тратить еще куча средств на ее закупку, что в свою очередь не очень дешево. Разработка информационной системы решает все проблемы. Такая система позволяет осуществить множества функций адаптированных под деятельность кафедры.

Функционал информационных систем представляет собой совокупность из нескольких возможностей, представленных ниже (рисунок 1.14):

- возможность загрузки литературы и трудов;
- хранение литературы и трудов;
- скачивание литературы и трудов;
- проведение анализа литературы и трудов;
- редактирование файлов внутри программы без использования проводника;
- облегчение работы и организация документов персонала;
- создание таблицы трудов персонала.

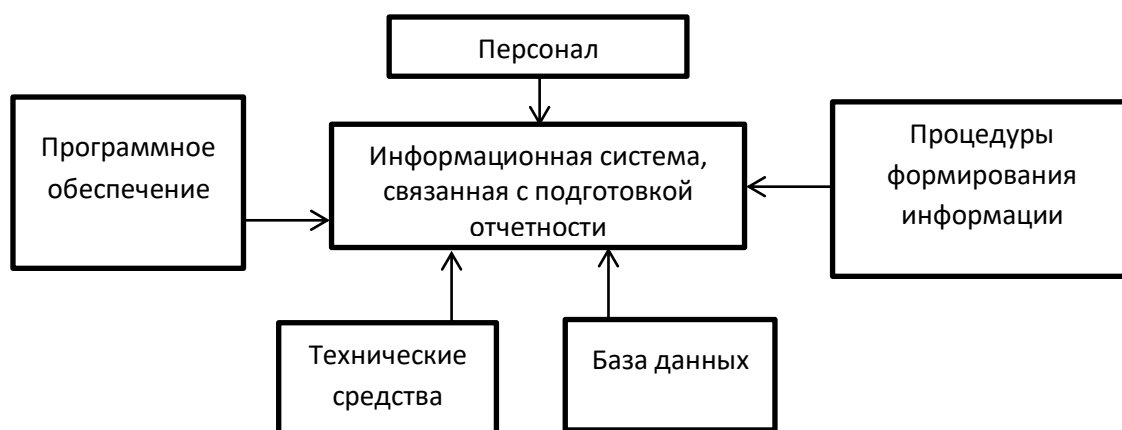


Рисунок 1.14 – Схема работы информационной системы

В современных условиях основным техническим средством обработки информации является персональный компьютер, который считается важным компонентом информационной технологии.

По степени обработки информации выделяют:

- ручную обработку данных;
- механизированную обработку данных;
- автоматизированную обработку данных;
- автоматическую обработку данных.

Информационные системы классифицируются:

- функциональное назначение – финансовые, коммерческие;
- объект управления – офис, фирма;
- характер использования – сбор, хранение, выдача информации.

Работа информационной на предприятии выглядит следующим образом. Необходимо подключить сервер, которому будут подключены все компьютеры кафедры. Это необходимо для совместного использования одной базы данных (рисунок 1.14).

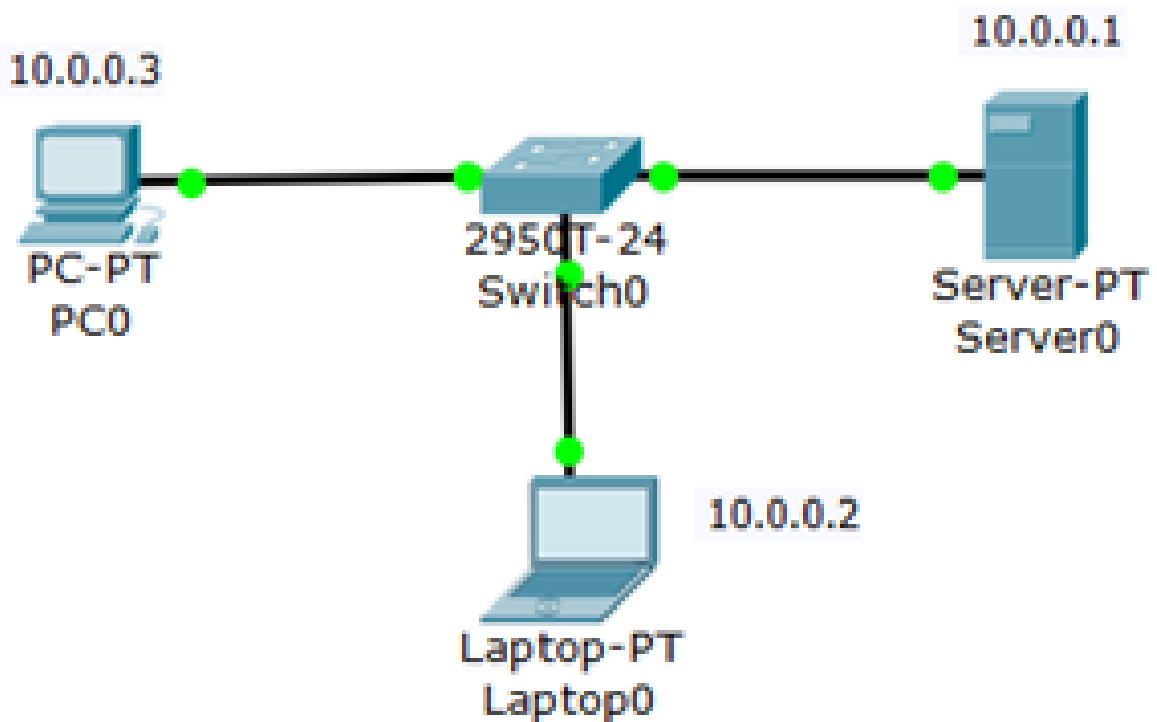


Рисунок 1.14 – Сеть на предприятии

Любая информационная системы имеет свою структуру. Она состоит из отдельных частей, которые называются подсистемами. Одни подсистемы выполняют функцию автоматизацию системы, другие – обеспечивают функционирование (рисунок 1.15).

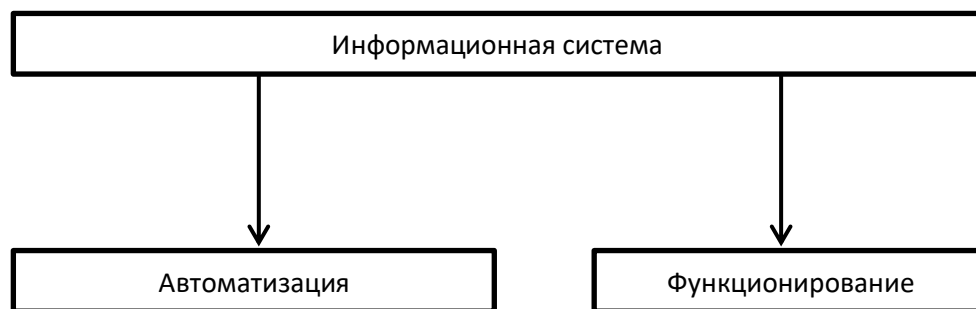


Рисунок 1.15 – Сеть на предприятии

Автоматизация процесса – способ работы систем, позволяющих осуществить управление процессом без участия человека.

Функционирование – возможность действовать система по назначенному алгоритму.

Мы также имеем информационные ресурсы. Информационные ресурсы хранятся на машинных носителях, архивах, фондах, библиотеках. Это весь имеющийся объем информации, зафиксированный на материальных носителях и предназначенный для общественного пользования (рисунок 1.16).



Рисунок 1.16 – Жесткий диск сервера

Самое важное в работе информационной системе является еще хранение данных в базе данных. Базы данных – это совокупность сведений о конкретных объектах реального мира в какой-либо объектной области.

Преимуществом базы данных является структурирование информации, которая является соглашением о способе представления данных (таблица 1.1).

Таблица 1.1 – Построение базы данных

Неструктурированные данные					
Автор: Иванов И.И., Название: Нейронные сети, Издательство: Алматы: КазНИТУ имени К.И.Сатпаева, Год: 2015, Страниц: 256, Соавтор: Петров И.В.					
Структурированные данные					
Автор	Название	Издательство	Год	Страниц	Соавтор
Иванов И.И.	Нейронные сети	Алматы: КазНИТУ имени К.И.Сатпаева	2015	256	Петров И.В.

База данных подобие электронной картотеки, которая хранится в компьютере в виде одного или нескольких файлов и имеет ряд особенностей[5]:

- большой объем хранения информации;
- возможность извлечения из базы данных разнообразной информации;
- удобная для пользователя вид и форма извлекаемой информации;
- высокая скорость доступа к данным;
- надежность хранения данных;

- простота запросов, форм и отчетов для выборки данных;
- характер использования – сбор, хранение, выдача информации.

Какие можно проводить операции с БД:

- добавление новой информации;
- изменение информации;
- поиск информации;
- удаление информации.

Зависимость данных – все данные размещены только в одном месте, и все изменения вносятся только в одном месте[6].

Автоматизация. Автоматизированные информационные системы являются важнейшим элементом современных БД[7].

Важно отметить, что использование информационных систем должен сократить бумажный оборот.

Хранения всей документации на бумаге является огромной проблемой:

- трудности отслеживания движения документа на всех этапах его жизненного цикла;
- длительность сроков подготовки и согласования документов;
- медленный поиск документов;
- сложность организации документооборота, если с одними и теми же документами одновременно работает несколько пользователей;
- трудоемкость получения сводных отчетов и журналов;
- в бумажном архиве нет возможности гибкого управления правами доступа к документам.

Традиционный документооборот оказывается неэффективным. Все эти минусы устраняются с введением систем электронного документооборота. Для таких организаций, где количество документов и сложность их ведения велики, становится жизненно важной задачей автоматизации документооборота с целью устранения вышеперечисленных недостатков

2 Описание и обоснование выбора языка программирования и СУБД

2.1 Проектирование информационной системы

На рисунок 2.1 представлена диаграмма возможностей для пользователя.

Информационные системы дают возможность использовать следующие функции пользователю[8]:

- импорт методических трудов в базу данных (путем заполнения нужных ячеек в самой информационной системе);
- экспорт из БД (возможность экспорта загруженных работ в базу данных);
- редактирование (редактирование методического труда в самой информационной системе);
- создание таблиц (информационная система полностью автоматизирована и присутствует автоматическое заполнение таблиц);
- анализ (анализ методических работ по годам).

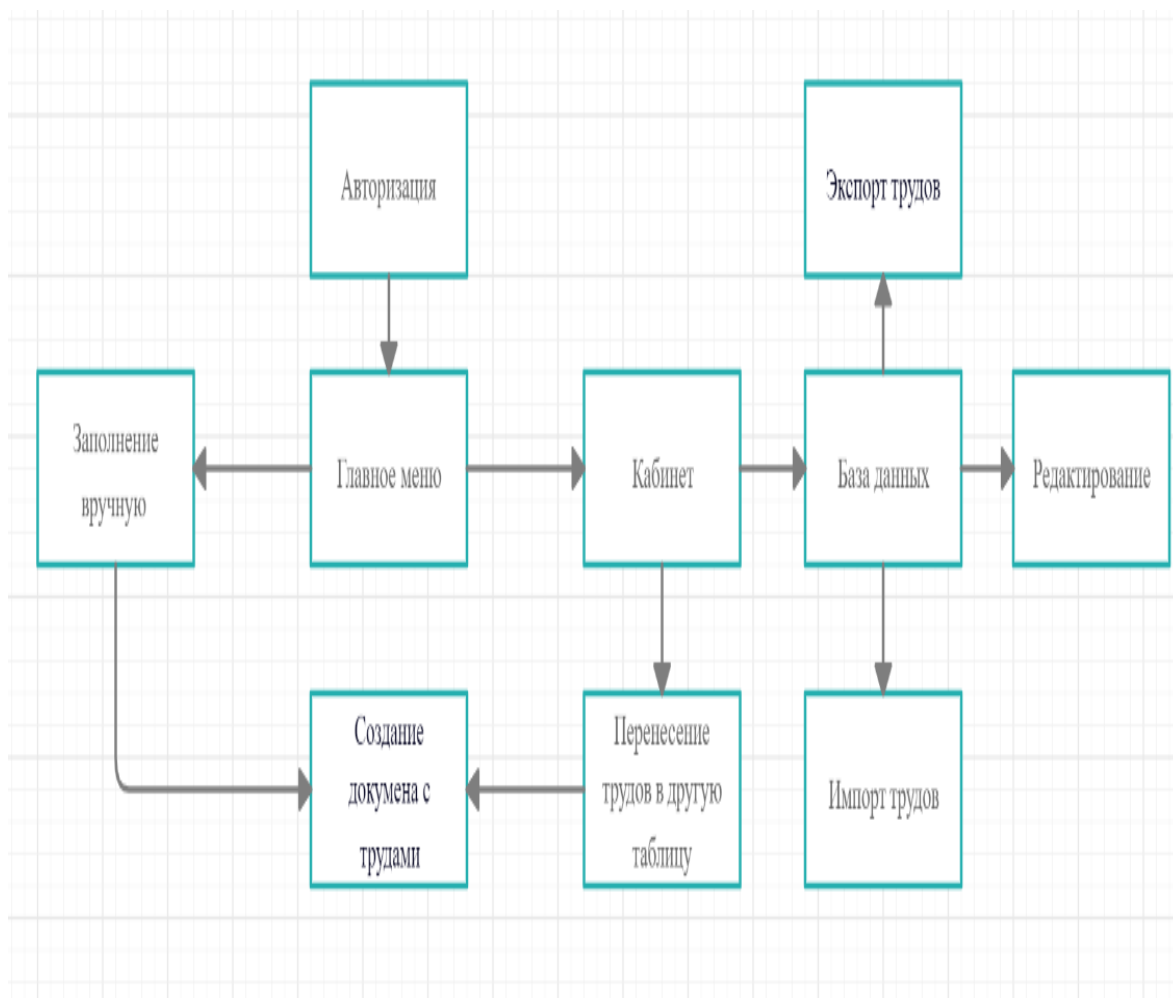


Рисунок 2.1 – Диаграмма возможностей для пользователя

На рисунке 2.2, 2.3, 2.4 представлена физическая модель, схема и диаграмма взаимодействия с базой данных. Рисунок содержит детали физической модели базы данных.

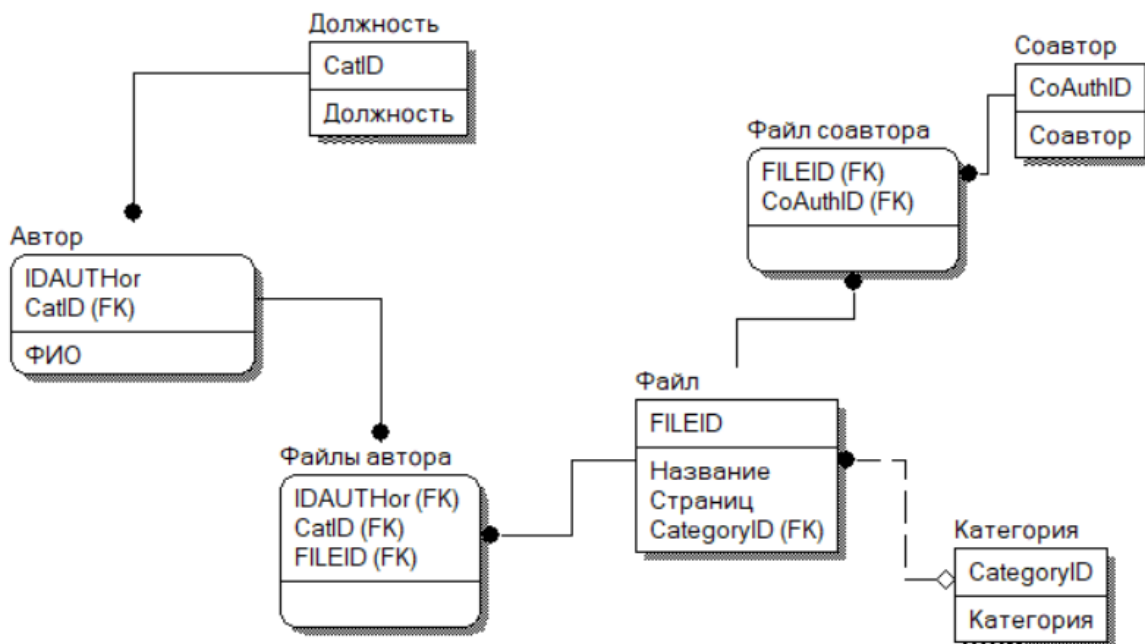


Рисунок 2.2 – Физическая модель базы данных



Рисунок 2.3 – Схема базы данных

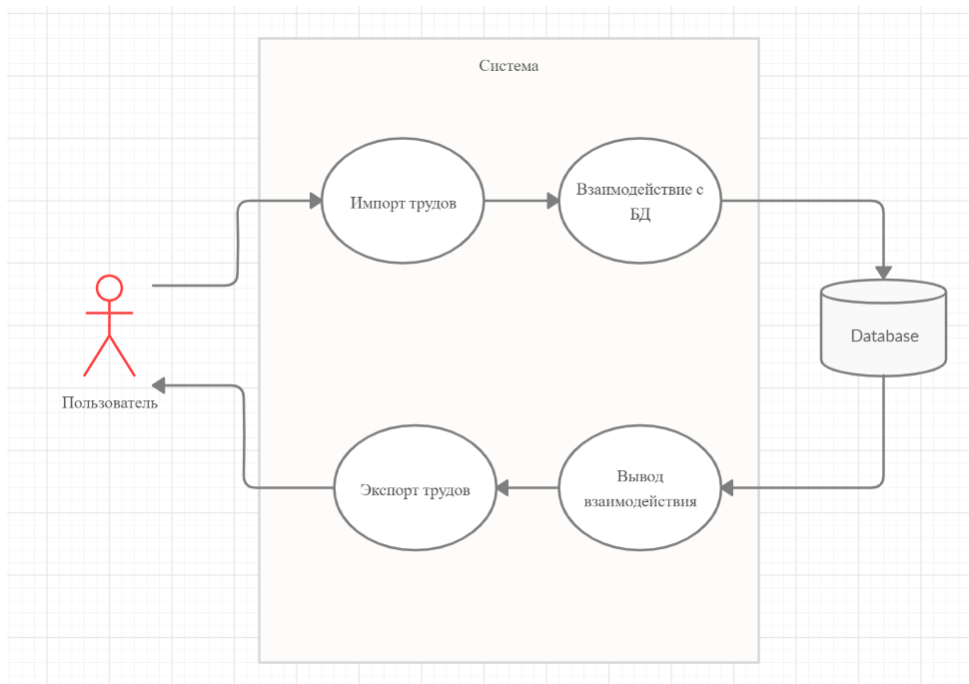


Рисунок 2.4 – Диаграмма взаимодействия с базой данных

На рисунке 2.5 представлена диаграмма последовательности работы информационной системы.

Диаграмма последовательности

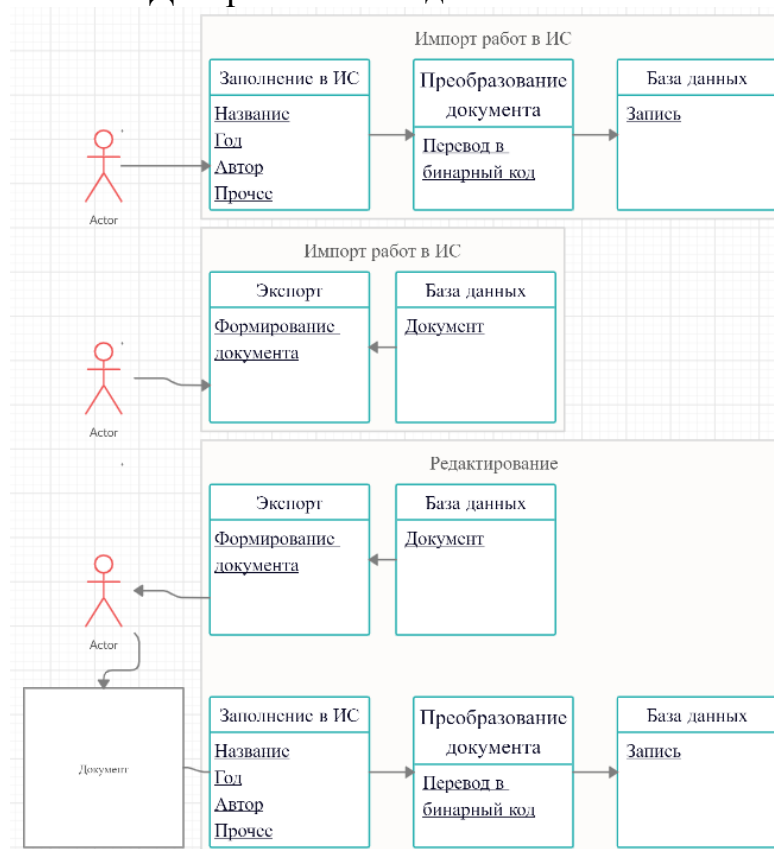
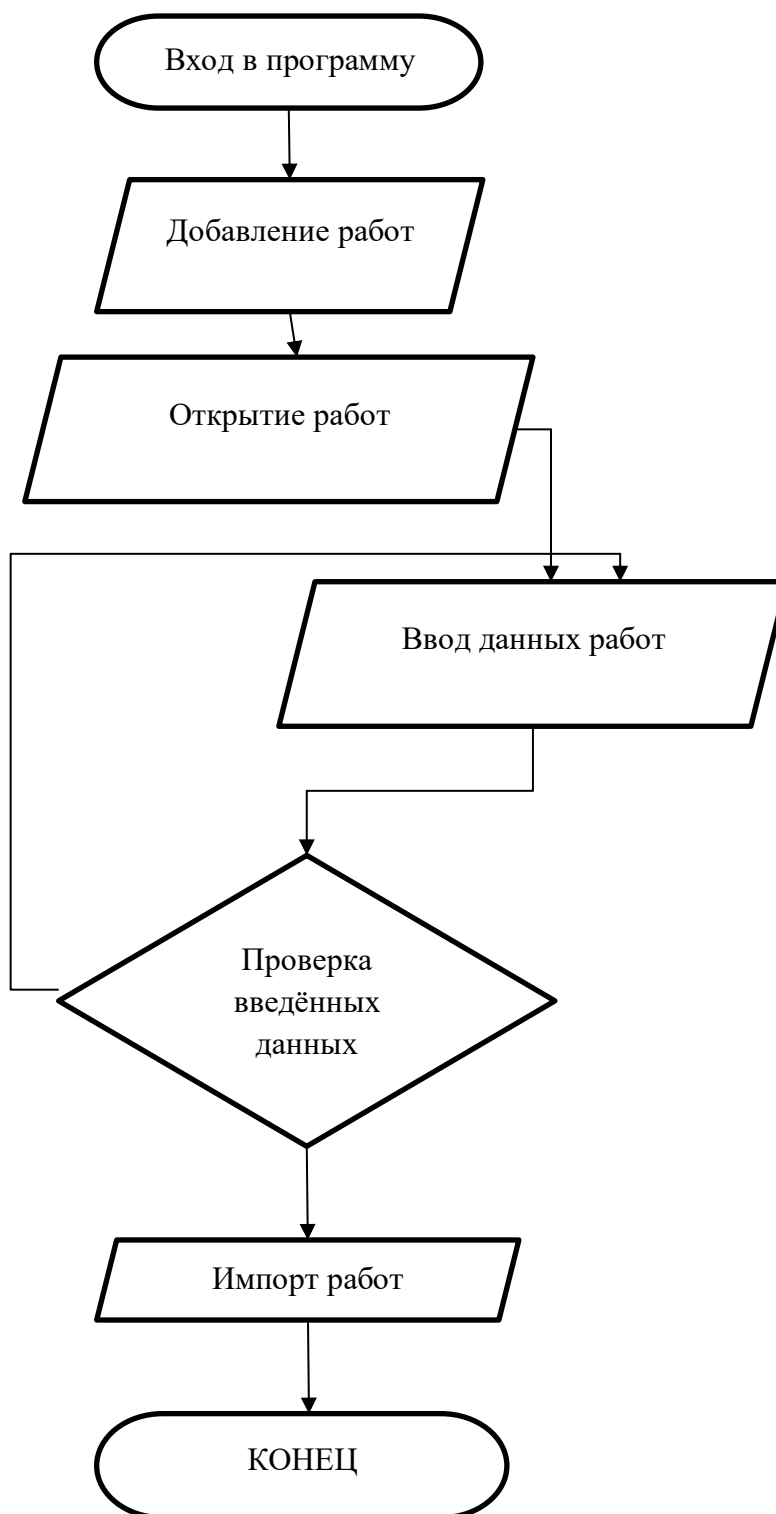


Рисунок 2.5 – Диаграмма последовательности

Ниже приведена блок-схема работы данного элемента сайта:



2.2 Анализ современных языков программирования

Среди разработчиков существует множество сред языков программирования[10].

Среда разработки программного обеспечения используется программистами для разработки программного обеспечения. Обычно в среде разработки входят компоненты, компилятор и отладчик. В такой среде можно производить весь цикл разработки[11]. Также существуют среды для разработки на нескольких языках – Microsoft Visual Studio и прочие.

Современные языки программирования:

- Delphi;
- C/C++;
- C#;
- Visual Basic;
- Java;
- JavaScript;
- PHP.

Рассмотрим плюсы и минусы разных языков программирования.

C++, C плюсы:

- возможность полного контроля;
- полный контроль над памятью;
- вы лучше понимаете все ошибки во время компиляции

C++, C минусы:

- требуется заранее оптимизировать;
- бедная библиотека.

JAVA, C# плюсы:

- можно не управлять памятью;
- богатая библиотека;
- компилируется в байт-код.

Одним из минусов языков JAVA и C# является наибольшая сложность выявления ошибки среди сравниваемых языков.

2.3 Выбор языка программирования C#

Язык программирования C# был выбран по причине легкости и доступности. C# является современным объектно-ориентированным и безопасным языком программирования. C# относится к широко известному семейству языков C, и покажется хорошо знакомым любому, кто работал с C, C++, Java или JavaScript.

C# является объектно-ориентированным языком, который поддерживается также и компонентно-ориентированным программированием[2]. Разработка приложений сильнее тяготеет к производству программных компонентов в виде автономных пакетов, исполняющих отдельные функциональные возможности[12]. Важная

особенность этих компонентов в том, что выражают из себя модель программирования со свойствами, методами и событиями. У них есть атрибуты, предоставляющие декларативные сведения о компоненте[13]. Эти компоненты включают в себя собственную документацию. С# дает возможность использовать языковые конструкции, которые поддерживают такую концепцию работы. Благодаря этому С# подходит для создания и применения программных компонентов.

Функции языка С#, обеспечивающих надежность и устойчивость приложений:

- сборка мусора автоматически освобождает память, занятую недостижимыми неиспользуемыми объектами;

- обработка исключений предоставляет структурированный и расширяемый подход к обнаружению ошибок, а также к восстановлению;

- безопасная структура языка делает невозможным чтение из неинициализированных переменных, индексацию массивов за пределами их границ или выполнение непроверенных приведений типов.

В С# есть единая система типов. Все типы С#, включая типы-примитивы, такие как `int` и `double`, наследуют от одного корневого типа `object`[14]. Это говорит о том, что все типы используют общий набор операций, и значения любого типа можно хранить, передавать и обрабатывать схожим образом. Таким образом, С# поддерживает пользовательские ссылочные типы и типы значений, позволяя как динамически выделять память для объектов, так и хранить упрощенные структуры в стеке.

При разработке С# много внимания было уделено управлению версиями, чтобы обеспечить совместимость программ и библиотек С# при дальнейшем развитии[15]. Большинство языков программирования стараются не обращать внимания на этот вопрос[3]. Из-за чего эти программы на этих языках ломаются чаще, чем хотелось бы, при выходе новых версий зависимых библиотек. Вопросы управления версиями существенно повлияли на такие аспекты разработки С#, как отдельные модификаторы `virtual` и `override`, правила разрешения перегрузки методов и поддержка явного объявления членов интерфейса.

Элементы языка С#[16]:

- структура С# основывается на программах, пространствах имен, типах, членах и сборках;

- можно получить сведения, такие как типы значения, ссылочные типы и переменные в языке С#;

- выражения производятся из операндов и операторов (выражения также могут возвращать значения);

- можете использовать инструкции, чтобы описать действия, которые выполняются программой;

- классы – это самый важный тип в языке С# (объекты представляют собой экземпляры классов, классы производятся путем описания их членов, которые описаны в этой статье);

- массив является структурой данных, которая содержит несколько переменных, доступ к которым можно осуществить по вычисляемым индексам;

- интерфейс реализован классами и структурами (интерфейс может содержать методы, свойства, события и индексы);

- тип delegate выражает из себя ссылки на методы и типом возвращаемого значения (делегаты позволяют использовать методы как сущности, сохраняя их в переменные и передавая в качестве параметров);

- атрибуты дают возможность программам указывать дополнительные описательные данные о типах, членах и других сущностях.

2.4 Особенности использования СУБД

Информационная система требует использования базы данных. Для работы требуется хранения методических трудов непосредственно в базе данных. Visual Studio имеет в себе локальную базу данных, которая очень удобна для использования[17]. Все синтаксисы запросов является общей среди всех языков базы данных и можно без проблем адаптироваться[4].

MySQL – свободная СУБД, где гибкость СУБД MySQL выражается поддержкой большого количества типов таблиц:

- пользователи имеют возможность выбрать как таблицы типа MyISAM;

- поддерживает текстовый поиск, так и таблицы InnoDB;

- поддерживает транзакции на уровне отдельных записей.

СУБД MySQL ставится с особым типом таблиц EXAMPLE, показывающие принципы производства новых типов таблиц. В СУБД MySQL перманентно появляются новые типы таблиц.

MySQL имеет API и коннекторы для многих языков программирования[18]:

- Delphi;

- C/C++/C#;

- JAVA;

- PHP;

- Python.

При этом – это ещё не полный список языков.

3 Разработка информационной системы

3.1 Программная реализация

Для программной реализации выбран язык программирования C#.

Главной задачей программы является загрузка методических трудов.

В первую очередь нужно выбрать файл для загрузки (рисунок 3.1).

Данную возможность можно реализовать следующими строками кода:

- OpenFileDialog отображает диалоговое окно, позволяющий пользователю открыть файл;
- Filter определяет варианты, которая определяет варианты, доступные в поле диалогового окна;
- Title задает заголовок диалогового окна.

```
OpenFileDialog fDialog = new OpenFileDialog();  
fDialog.Title = "Выберите файл";  
fDialog.Filter = "docx files (*.docx)|*.docx|doc files (*.doc)|*.doc";
```

Рисунок 3.1 – Работа с диалоговым окном

Выбранный файл для загрузки в базу данных нужно конвертировать в бинарный код.

FileStream предоставляет stream в файле, поддерживая асинхронные и синхронные операции чтения и записи. Класс позволяет использовать путь файла, открытие файла, а также его чтение для дальнейшего прогона в программе (рисунок 3.2);

```
FileStream FS = new FileStream(path, System.IO.FileMode.Open, System.IO.FileAccess.Read);
```

Рисунок 3.2 – Работа с файлом (открытие, чтение)

BinaryReader считывает примитивные типы данных как двоичные значения в заданной кодировке. Вся нужная информация хранится в переменной FS и отсюда добавляет в бинарный считыватель (рисунок 3.3);

```
BinaryReader BR = new BinaryReader(FS);
```

Рисунок 3.3 – Работа с бинарным считывателем

FileInfo предоставляет свойство и методы экземпляра для создания, копирования, удаления, перемещения и открытия файлов, а также позволяет создавать объекты FileStream. Отсюда можно получить общую длинную байта файла (рисунок 3.4);

```
long allbytes = new FileInfo(path).Length;
```

Рисунок 3.4 – Получения длины байта файла

ReadBytes считывает указанное количество байтов из текущего потока в массив байтов и перемещает текущую позицию на это количество байтов. Length получает размер текущего файла в байтах (рисунок 3.5).

```
FileBytes = BR.ReadBytes((Int32)allbytes);
```

Рисунок 3.5 – Работа с ReadBytes

InsertFile является ссылкой на метод. В данном случае он вносит данные из переменных. FileBytes содержит бинарный код загруженного файла, который нужно занести в базу данных (рисунок 3.6).

```
InsertFile(filename, FileBytes,
```

Рисунок 3.6 – Работа с InsertFile

Вызов InsertFile позволяет импортировать файлы в базу данных.

SqlCommand представляет хранимую процедуру, выполняемую над базой данных SQL. В данном случае SqlCommand требуется для выполнения запроса select. Такая необходимость требуется для вывода информации из базы данных. В переменной sCom хранится запрос на вывод информации из базы данных (рисунок 3.7).

```
string sql = "SELECT * FROM uploadfile";  
SqlCommand sCom = new SqlCommand(sql, sqlcon);  
int max = 0;  
sqlcon.Open();
```

Рисунок 3.7 – Работа с запросами в локальной базе данных VS

SqlDataReader предоставляет способ чтения потока строк последовательного доступа из базы данных SQL. Из предыдущего пункта выводится информация из базы данных, которая потребовалась для чтения потока строк столбца "Id" из базы данных, чтобы при каждом прогоне давать каждому новому файлу новое значение ID, которое на одно больше, чем предыдущее (рисунок 3.8).

```
using (SqlDataReader read = sCom.ExecuteReader())
{
    while (read.Read())
    {
        if (int.Parse(read["Id"].ToString()) > max)
        {
            max = int.Parse(read["Id"].ToString());
        }
    }
}
max++;
sqlcon.Close();
```

Рисунок 3.8 – Предоставление нового значения ID

Переменная max потребовалось для импорта id в базу данных нового файла. В запросе insert происходит импорт наименования id, имени, бинарного кода и прочее (рисунок 3.9).

```
sqlcon.Open();
SqlCommand sqlcmd = new SqlCommand("insert into uploadfile(Id, Имя, fcontent,
sqlcmd.Parameters.AddWithValue("@Id", max);
sqlcmd.Parameters.AddWithValue("@FN", FN);
sqlcmd.Parameters.AddWithValue("@FB", FB);
sqlcmd.Parameters.AddWithValue("@FD", FD);
sqlcmd.Parameters.AddWithValue("@FA", FA);
sqlcmd.Parameters.AddWithValue("@FK", FK);
sqlcmd.Parameters.AddWithValue("@FP", FP);
sqlcmd.Parameters.AddWithValue("@FS", FS);
sqlcmd.Parameters.AddWithValue("@FC", FC);
sqlcmd.Parameters.AddWithValue("@FO", FO);
sqlcmd.Parameters.AddWithValue("@FG", FG);
sqlcmd.ExecuteNonQuery();
sqlcon.Close();
```

Рисунок 3.9 – Внесения данных в базу

Для работы с базой данных в среде разработки Visual Studio C# была использована локальную базу данных и подключена для этого библиотека System.Data.SqlClient (рисунок 3.10).

```
using System.Data.SqlClient;
```

Рисунок 3.10 – Использование директивы System.Data.SqlClient

Для импорта количества страниц в базу данных требуется подключить пакеты NuGet – SautinSoft. Данный пакет дает возможность для создания загрузки и изменения файлов (рисунок 3.11).

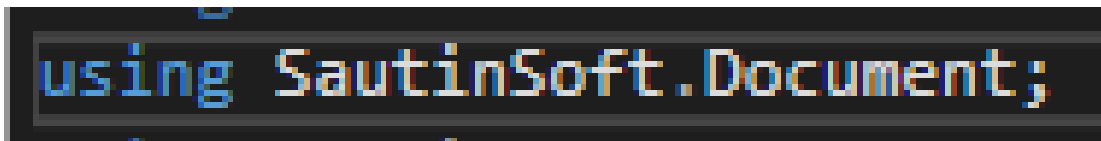


Рисунок 3.11 - Использование директивы SautinSoft.Document

Doc и Docx являются абсолютно разными форматами. Разница между Doc и Docx заключается в работе пакетов программ с данными форматами. Решается проблема с помощью дополнительного конвертера. ConvertFile является удобным способом конвертировать из Doc в Docx. CalculateStats не работает со старым форматом Doc. Поэтому требуется конвертация. После конвертации программа сохраняет в dc статистику файла: количество страниц, символов и т.д. Для вывода страниц требуется `dc.Document.Properties.BuiltIn[BuiltInDocumentProperty.Pages]` (рисунок 3.12)

```
SautinSoft.UseOffice u = new SautinSoft.UseOffice();

string inpFile = Path.GetFullPath(path);
string outFile = Path.GetFullPath("D:\\temp\\temporary.docx");

int ret = u.InitWord();

if (ret == 1)
{
    Console.WriteLine("Error! Can't load MS Word library in memory");
    return;
}

// Perform the conversion.
ret = u.ConvertFile(inpFile, outFile, SautinSoft.UseOffice.eDirection.DOC_to_DOCX);

u.CloseWord();

DocumentCore dc = DocumentCore.Load("D:\\temp\\temporary.docx");

dc.CalculateStats();

File.Delete("D:\\temp\\temporary.docx");
```

Рисунок 3.12 – Работа с разными форматами документа

Программа имеет возможность скачивания файлов из базы данных. В переменную name нужно внести значения из таблицы при нажатии по строке. Это требуется, чтобы программа не путалась и скачала нужный файл из базы данных. С помощью запроса select он выводит один файл с именем равной из переменной name (рисунок 3.13).


```

if (e.ColumnIndex == 0)
{
    //string id;
    string name;
    FileStream FS = null;
    byte[] dbbyte;

    name = dataGridView1.Rows[e.RowIndex].Cells[e.ColumnIndex+1].Value.ToString();
    //id = Convert.ToString(e.RowIndex + 1);

    sqlcon.Open();
    sqlcmd = new SqlCommand("select * from uploadfile where Имя = @name ", sqlcon);
    sqlcmd.Parameters.AddWithValue("@name", name);
    da = new SqlDataAdapter(sqlcmd);
    dt = new System.Data.DataTable();
    da.Fill(dt);
    sqlcon.Close();
}

```

Рисунок 3.13 – Поиск нужного файла по имени

После экспорта бинарного кода из базы данных, программа начинает создания файла. Write позволяет записать блок байтов в файловый поток (рисунок 3.14).

```

string filetype = dataGridView1.Rows[e.RowIndex].Cells[1].Value.ToString().Substring(0, 4);
string filename = dataGridView1.Rows[e.RowIndex].Cells[1].Value.ToString().Substring(4);
dbbyte = (byte[])dt.Rows[0]["fcontent"];
string filepath = Path.Combine(Properties.Settings.Default.DownloadPath, filename);

FS = new FileStream(filepath, FileMode.Create);
FS.Write(dbbyte, 0, dbbyte.Length);
FS.Close();

Process Proc = new Process();
Proc.StartInfo.FileName = filepath;
Proc.Start();

```

Рисунок 3.14 – Процесс формирования документа

Главной особенностью проекта является создание таблиц. В первую очередь требуется внесение данных из первой таблицы во вторую. Add позволяет добавить в коллекцию новую строку (рисунок 3.15).

```

if (dataGridView2.Rows.Count > 0)
{
    for (int i = 0; i < dataGridView1.Rows.Count; i++)
    {
        dataGridView2.Rows.Add();
        int l = 1;
        foreach (DataGridViewRow row in dataGridView2.Rows)
        {
            row.Cells[0].Value = 1;
            l++;
        }
        //dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[0].Value = dataGridView1.Rows[i].Cells[0].Value.ToString();
        dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[1].Value = dataGridView1.Rows[i].Cells[1].Value.ToString();
        dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[2].Value = dataGridView1.Rows[i].Cells[3].Value.ToString();
        dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[3].Value = dataGridView1.Rows[i].Cells[4].Value.ToString();
        dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[4].Value = dataGridView1.Rows[i].Cells[5].Value.ToString();
        dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[5].Value = dataGridView1.Rows[i].Cells[7].Value.ToString();
        dataGridView2.Rows[dataGridView2.Rows.Count - 1].Cells[6].Value = dataGridView1.Rows[i].Cells[8].Value.ToString();
    }
}

```

Рисунок 3.15 – Перенос данных из одной таблицы в другую

Как только программа перенесет данные из одной таблицы в другую, строки из combobox удаляются и заносятся в массив List <string> category через команду Add. Также количество строк заносятся в массив countsPages (рисунок 3.16).

```

category.Add(Convert.ToString(comboBox1.SelectedItem));
countsPages.Add(dataGridView1.RowCount);
comboBox1.Items.Remove(comboBox1.SelectedItem);
int num1 = int.Parse(lblPageCount.Text);
int sum = num1 + 1;
lblPageCount.Text = sum.ToString();
vivod.Enabled = true;

```

Рисунок 3.16 – Занесение категорий в массив

Если нет необходимости в работе из таблицы, то есть возможность без проблем удалите через RemoveAt (рисунок 3.17).

```

if (IndexIsChanged == true)
{
    dataGridView2.Rows.RemoveAt(selectedIndex);
}

```

Рисунок 3.17 – Удаление работы из таблицы

Массивы countsPages и category требовались для формирования таблицы (рисунок 3.18).

```
html += @"<tr><td colspan=""5"" style = 'font-family: ""Times New Roman""><center><font size=""2""><center><b> + category[i]
if (i == 0)
{
    for (int f = 0; f < countsPages[i]; f++)
    {
        html += "<tr>";
        for (int j = 0; j < dataGridView2.ColumnCount - 2; j++)
        {
            html += @"<td style = 'font-family: ""Times New Roman""><center><font size=""2""> + dataGridView2.Rows[f].Cells[j
```

Рисунок 3.18 – Формирование таблиц

Для создания таблиц формата docx требуется создать таблицу в формате html. WriteAllText создает новый файл, записывает в него указанную строку и затем закрывает файл (рисунок 3.19).

```
string htmlFilePath = @"C:\DataGridView.htm";
File.WriteAllText(htmlFilePath, html);
object oMissing = System.Reflection.Missing.Value;
```

Рисунок 3.19 – Временная запись документа в HTML

Итоговый файл должен быть в формате docx и для этого требуется конвертация из html в docx (рисунок 3.20).

```
Microsoft.Office.Interop.Word._Application word = new Microsoft.Office.Interop.Word.Application();
Microsoft.Office.Interop.Word._Document wordDoc = word.Documents.Open(FileName: htmlFilePath, ReadOnly: false);

wordDoc.SaveAs(FileName: sfd.FileName, FileFormat: Microsoft.Office.Interop.Word.WdSaveFormat.wdFormatRTF);
((Microsoft.Office.Interop.Word._Document)wordDoc).Close();
((Microsoft.Office.Interop.Word._Application)word).Quit();
Process.Start(sfd.FileName);
```

Рисунок 3.20 – Конвертация документа из HTML в DOCX

Для работы с dll офиса от Майкрософта требуется подключить требуемую библиотеку, а также пакеты через ссылки (рисунок 3.21).

```
using Microsoft.Office.Interop.Word;
```

Рисунок 3.21 – Использование директивы Microsoft.Office.Interop.Word

Создание таблиц вручную имеет схожий алгоритм, что и автоматическое, но использует другой способ внесения данных. Из названия очевидно, что требуется внести вручную всю информацию для вывода.

Для создания сперва требуется выбрать категории, которые требуются внести в таблицу и занести в массив для использования в следующих шагах (рисунок 3.22).

```
if (checkBox1.Checked == true || checkBox2.Checked == true || checkBox3.Checked == true || checkBox4.Checked == true || checkBox5.Che
{
    if (checkBox1.Checked == true) { One = checkBox1.Text; massiv = massiv.Concat(new string[] { checkBox1.Text }).ToArray(); }
    if (checkBox2.Checked == true) { Two = checkBox2.Text; massiv = massiv.Concat(new string[] { checkBox2.Text }).ToArray(); }
    if (checkBox3.Checked == true) { Three = checkBox3.Text; massiv = massiv.Concat(new string[] { checkBox3.Text }).ToArray(); }
    if (checkBox4.Checked == true) { Four = checkBox4.Text; massiv = massiv.Concat(new string[] { checkBox4.Text }).ToArray(); }
    if (checkBox5.Checked == true) { Five = checkBox5.Text; massiv = massiv.Concat(new string[] { checkBox5.Text }).ToArray(); }
    if (checkBox6.Checked == true) { Six = checkBox6.Text; massiv = massiv.Concat(new string[] { checkBox6.Text }).ToArray(); }
    if (checkBox7.Checked == true) { Seven = checkBox7.Text; massiv = massiv.Concat(new string[] { checkBox7.Text }).ToArray(); }
    if (checkBox8.Checked == true) { Eight = checkBox8.Text; massiv = massiv.Concat(new string[] { checkBox8.Text }).ToArray(); }

    pageList = pageList.Concat(massiv).ToArray();
}
```

Рисунок 3.22 – Выбор категорий

При указании количества работ в текстовом поле, изменяется количество вкладок для внесения данных, которые требуются для вывода (рисунок 3.22).

```
if (!String.IsNullOrEmpty(txtKolvoRabot.Text))
{
    if (Convert.ToInt32(txtKolvoRabot.Text) <= 10 && Convert.ToInt32(txtKolvoRabot.Text) > tabControl1.TabPages.Count)
    {
        for (int i = tabControl1.TabPages.Count; i < Convert.ToInt32(txtKolvoRabot.Text); i++)
        {
            tabControl1.TabPages.Add(new TabPage { Text = "Работа " + (i + 1), Name = "tabPage" + (i + 2) });
            tabControl1.TabPages[i].BackColor = Color.White;

            TextBox txt1 = new TextBox();
            txt1.Name = "txta" + i;
            txt1.Location = new Point(txtName.Location.X, txtName.Location.Y);
            txt1.Size = new Size(620, 26);
            txt1.Anchor = (AnchorStyles.Bottom | AnchorStyles.Left);
            tabControl1.TabPages[i].Controls.Add(txt1);
            txt1.BringToFront();

            TextBox txt2 = new TextBox();
            txt2.Name = "txtb" + i;
            txt2.Location = new Point(txtIzdatelstvo.Location.X, txtIzdatelstvo.Location.Y);
            txt2.Multiline = true;
            txt2.Size = new Size(350, 70);
            txt2.Anchor = (AnchorStyles.Bottom | AnchorStyles.Left);
            tabControl1.TabPages[i].Controls.Add(txt2);
            txt2.BringToFront();

            TextBox txt3 = new TextBox();
            txt3.Name = "txtc" + i;
            txt3.Location = new Point(txtPages.Location.X, txtPages.Location.Y);
            txt3.Size = new Size(350, 26);
            txt3.Anchor = (AnchorStyles.Bottom | AnchorStyles.Left);
            tabControl1.TabPages[i].Controls.Add(txt3);
            txt3.BringToFront();
        }
    }
}
```

Рисунок 3.22 – Создание новых вкладок

Как только данные текстовые поля будут заполнены, то можно будет перейти на следующую вкладку, но текущая будет заблокирована (рисунок 3.23).

```
if (tabControl1.TabPages.Count <= currentTab) { currentTab = tabControl1.TabPages.Count - 1; }
if (tabControl1.SelectedIndex - currentTab == 1)
{
    if (CheckPage() == true)
    {
        //MessageBox.Show("Страница перевернута, индекс = " + tabControl1.SelectedIndex);
        EnableTab(tabControl1.TabPages[currentTab], false);
        currentTab = tabControl1.SelectedIndex;
    }
    else { tabControl1.SelectedIndex = currentTab; MessageBox.Show("Не все поля заполнены! Переход на следующую");
}
else if (tabControl1.SelectedIndex > currentTab)
{
    tabControl1.SelectedIndex = currentTab;
    MessageBox.Show("Открыть можно только следующую вкладку! Страница осталась = " + (currentTab + 1), "Ошибка!");
}
GC.Collect();
```

Рисунок 3.23 – Переход на новую вкладку

При окончании заполнения одной категории, то есть возможность перейти к следующей вкладке (рисунок 3.24).

```
currentPage++;
label6.Text = ChooseCategories.pageList[currentPage];

txtKolvoRabot.Text = "1";
currentTab = 0;
EnableTab(tabControl1.TabPages[0], true);
for (int i = 0; i < textBoxCount(tabControl1.TabPages[0]); i++)
{
    textboxes[i].Text = "";
}
```

Рисунок 3.24 – Переход к следующей при заполнении

Для создания таблиц формата docx требуется создать таблицу в формате html. WriteAllText создает новый файл, в которую записывает указанную строку, и, после завершения, закрывает (рисунок 3.25).

```
string htmlFilePath = @"C:\DataGridView.htm";
File.WriteAllText(htmlFilePath, html);
object oMissing = System.Reflection.Missing.Value;
```

Рисунок 3.25 – Создание документа в HTML

Итоговый файл должен быть в формате docx и для этого требуется конвертация из html в docx (рисунок 3.26).

```
Microsoft.Office.Interop.Word._Application word = new Microsoft.Office.Interop.Word.Application();
Microsoft.Office.Interop.Word._Document wordDoc = word.Documents.Open(fileName: htmlFilePath, ReadOnly: false);

wordDoc.SaveAs(fileName: sfd.FileName, FileFormat: Microsoft.Office.Interop.Word.WdSaveFormat.wdFormatRTF);
((Microsoft.Office.Interop.Word._Document)wordDoc).Close();
((Microsoft.Office.Interop.Word._Application)word).Quit();
Process.Start(sfd.FileName);
```

Рисунок 3.26 – Конвертация документа в DOCX из HTML

WriteAllBytes создает новый файл, в которую записывает указанный массив байтов и, после завершения, закрывает. Через GetDataFromDatabase программа достает выбранный файл из таблицы, а затем присваивает значение к переменной data, что позволяет создать временный файл для редактирования (рисунок 3.27).

```
byte[] data = GetDataFromBase();
if (toggle == true)
{
    string fileformat = dataGridView1.SelectedRows[0].Cells[1].Value.ToString().Substring(Convert.ToInt32(dataGrid
formatForRedakt = fileformat.Replace(" ", "");
    File.WriteAllBytes("c:\\temp\\RedaktFile" + fileformat.Replace(" ", ""), data);

    startWatch.Start();

    Process Proc = new Process();
    Proc.StartInfo.FileName = "c:\\temp\\RedaktFile" + fileformat.Replace(" ", "");
    Proc.Start();

    ScanClosedApplications();
}
```

Рисунок 3.27 – Создание временного файла

Принцип работы GetDataFromDatabase (рисунок 3.28).

```
bool toggle = true;
private static readonly string CONNECT_STRING = "Data Source=(LocalDB)\\MSSQLLocalDB;AttachDbFilename=" + Environment.
public byte[] GetDataFromBase()
{
    string selectCMD = "SELECT fcontent FROM uploadfile WHERE Имя = '" + nameForRedakt + "'";

    using (SqlConnection cnn = new SqlConnection(CONNECT_STRING))
    {
        cnn.Open();
        using (SqlCommand cmd = new SqlCommand(selectCMD, cnn))
        {
            cmd.CommandType = CommandType.Text;
            cmd.CommandText = "SELECT fcontent FROM uploadfile WHERE Имя = @NAME";
            cmd.Parameters.AddWithValue("@NAME", nameForRedakt);
            using (SqlDataReader dr = cmd.ExecuteReader())
            {
                try
                {
                    dr.Read();
                    byte[] data = (byte[])dr["fcontent"];
                    toggle = true;
                    return data;
                }
                catch (Exception)
                {
                    toggle = false;
                    return null;
                }
            }
        }
    }
}
```

Рисунок 3.28 – Извлечение байтов из базы данных для временного файла

Происходит создание бинарного кода и внесение в переменную FileBytes (рисунок 3.29).

```
FileStream FS = new FileStream("c:\\temp\\RedaktFile" + formatForRedakt.Replace(" ", ""), System.IO.FileMode.Open, System.IO.FileAccess.Read);
BinaryReader BR = new BinaryReader(FS);
long allbytes = new FileInfo("c:\\temp\\RedaktFile" + formatForRedakt.Replace(" ", "")).Length;
FileBytes = BR.ReadBytes((Int32)allbytes);

FS.Close();
FS.Dispose();
BR.Close();
```

Рисунок 3.29 – Создание бинарного кода и внесение в FileBytes

Через запрос UPDATE, программа обновляет fcontent, в котором хранился предыдущий бинарный код и меняет содержимое в ячейке на новый код из переменной FileBytes (рисунок 3.30).

```
string connectionString = "Data Source=(LocalDB)\\MSSQLLocalDB;AttachDbFilename=D:\\Uni
SqlConnection connection = new SqlConnection(connectionString);
SqlCommand sCommand = new SqlCommand();
sCommand.Connection = connection;
connection.Open();
sCommand.CommandType = CommandType.Text;
sCommand.CommandText = "UPDATE uploadfile SET fcontent = @FCONTENT WHERE Имя = @NAME";
sCommand.Parameters.AddWithValue("@FCONTENT", FileBytes);
sCommand.Parameters.AddWithValue("@NAME", nameForRedakt);
sCommand.ExecuteNonQuery();
connection.Close();
```

Рисунок 3.31 – После завершения происходит обновление ячейки в БД

3.2 Системные требования

Рекомендуемые системные требования:

- операционная система: Windows 7/8/10 - 64-bit;
- процессор (CPU): Intel Celeron (Desktop);Л
- оперативная память (RAM): 1 Гб (или больше);
- свободное место на жёстком диске: ~300 МВ.

Рекомендуемые системные требования:

- операционная система: Windows 7/8/10 - 64-bit;
 - процессор (CPU): Intel Core i3 (Desktop);
 - оперативная память (RAM): 2 Гб (или больше);
- свободное место на жёстком диске: ~300 МВ.

3.3 Интерфейс

Необходимо рассмотреть приложения на контрольном примере (рисунок 3.32).

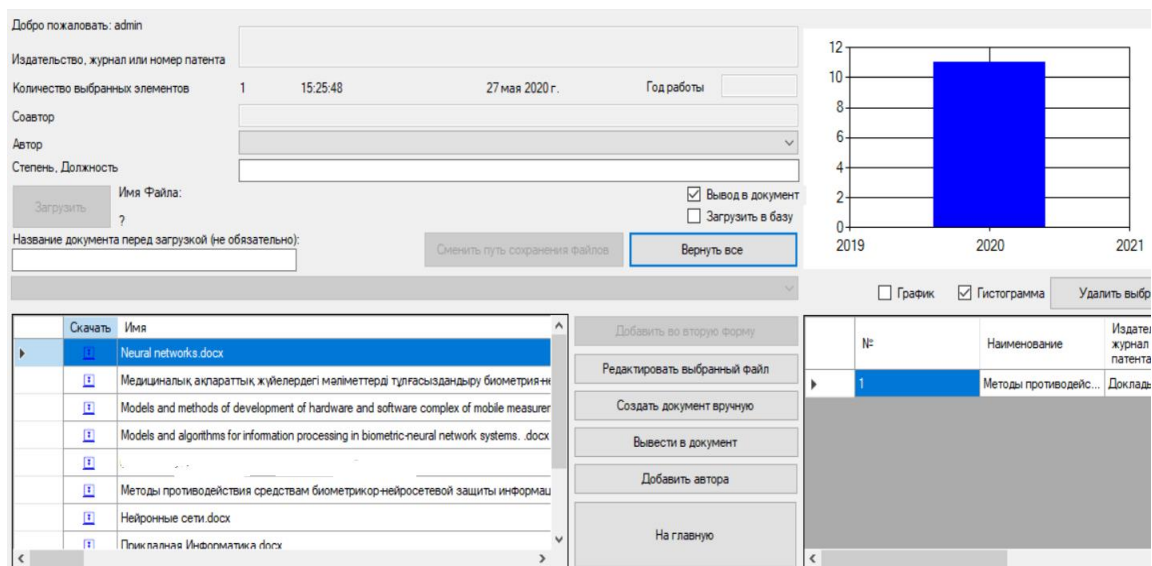


Рисунок 3.32 - Интерфейс

В базу данных программы можно загружать файлы любых форматов, но, в основном, программа рассчитана на файлы документов.

Пример загрузки файла в БД (рисунокрисунок 3.33).

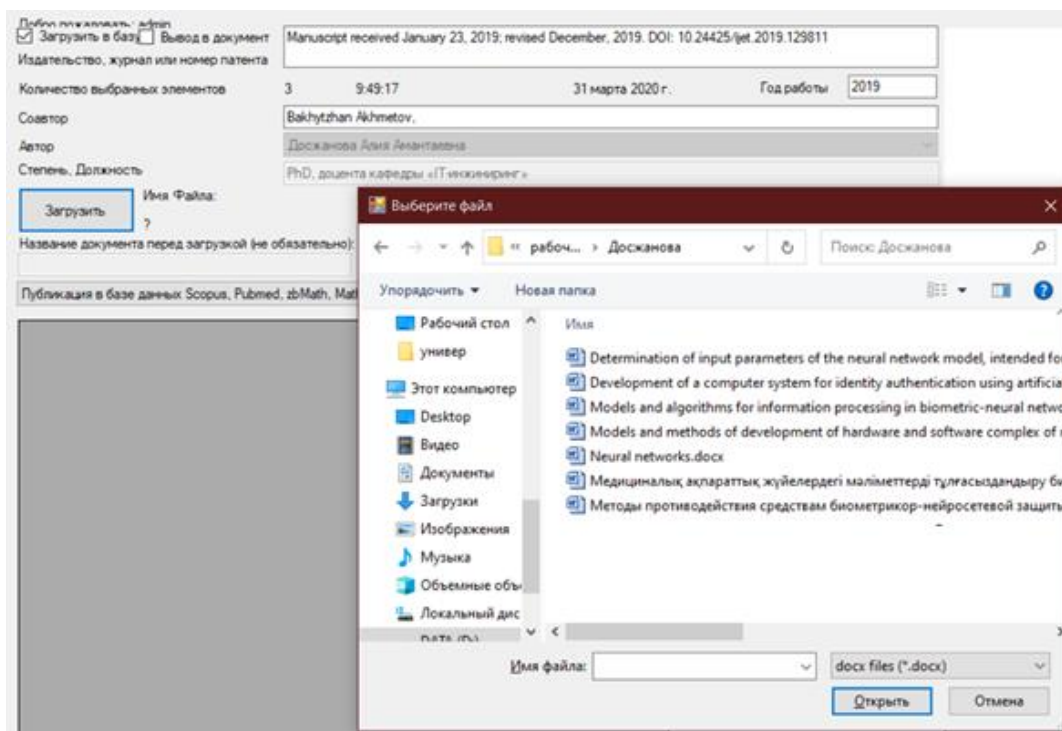


Рисунок 3.33 – Загрузка в базу данных

Все загруженные документы хранятся в базе данных программы, с которой и происходит дальнейшее взаимодействие (рисунок 3.34).

Id	Имя	fcontent	Дата	Аккаунт	Категория	Принадлежно...	Страниц	Соавтор	Автор	Год
1	Neural network...	0x50480304140...	30 марта 2020 ...	admin	Учебные посо...	Tutorial, Almaty...	10	Akhmetov B.S., ...	Досжанова Ал...	2019
2	Медициналық ...	0x50480304140...	30 марта 2020 ...	admin	Монография	Алматы: АУЭС, ...	10	Malikova F.U., K...	Досжанова Ал...	2018
3	Models and me...	0x50480304140...	30 марта 2020 ...	admin	Монография	Алматы: ИИСТ, 20...	10	Malikova F.U., K...	Досжанова Ал...	2018
4	Models and alg...	0x50480304140...	30 марта 2020 ...	admin	Монография	Алматы: ИИСТ, 20...	10	Malikova F.U. ...	Досжанова Ал...	2019

Рисунок 3.34 – таблица в базе данных

Анализ происходит в виде вывода графика и гистограммы, строящихся по выбранному полю в таблице документов. Пример вывода гистограммы по годам выпуска литературы (вся литература 2020 года) (рисунок 3.35).

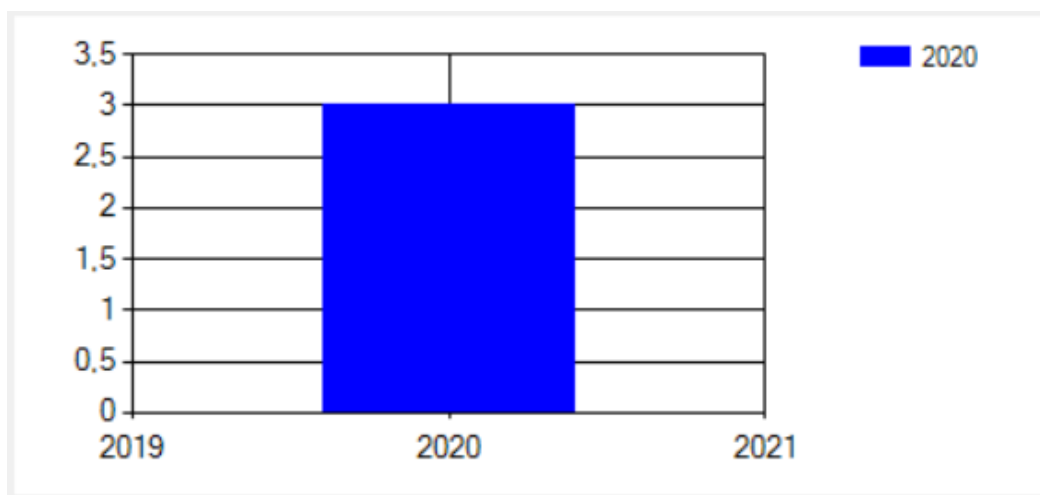
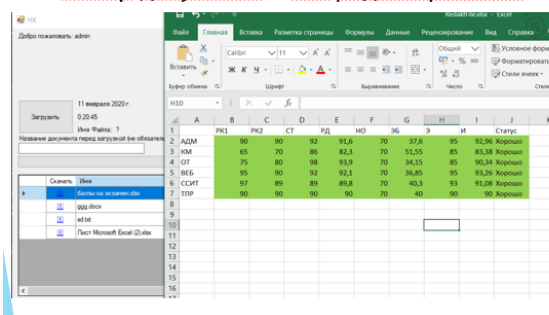


Рисунок 3.35 – гистограмма работ по годам

Данная функция позволяет изменять содержимое документов прямо внутри базы данных программы, не выходя и не загружая новые файлы снова. Этапы редактирования показаны ниже (рисунок 3.36).

1. Выбор документа и его редактирование.



2. Сохранение документа и его закрытие.

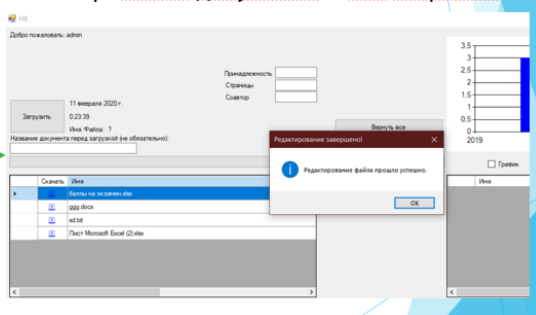


Рисунок 3.36 – Редактирование документа

Каждый загруженный документ можно скачать (выгрузить) на жесткий диск компьютера, если возникнет такая необходимость. Скачивание выполнится по указанному программе пути после нажатия на кнопку скачивания рядом с именем файла (рисунок 3.37).

	Скачать	Имя
▶		Neural networks.docx
		Медициналық ақпараттық жүйелердегі мәліметтерді тұлғасыздандыру биометриянк
		Models and methods of development of hardware and software complex of mobile measurer
		Models and algorithms for information processing in biometric-neural network systems. .docx

Рисунок 3.37 – Загруженные работы

Для данной цели был создан дружественный интерфейс, который продолжает совершенствоваться.

Каждый преподаватель может создать свой удобный список документов (рисунок 3.38).

Скачать	Имя
▶	Neural networks.docx
	Медициналық ақпараттық жүйелердегі мәліметтерді тұлғасыздандыру биометриянк
	Models and methods of development of hardware and software complex of mobile measurer
	Models and algorithms for information processing in biometric-neural network systems. .docx

Добавить во вторую форму		
Редактировать выбранный файл		
Создать документ вручную		
Вывести в документ		
	№	Наименование
▶	1	Методы противодейс... Доклады
		Издатель журнал и патента

Рисунок 3.38 – Перенос в другую таблицу

Программа имеет возможность создания таблиц научных трудов. Пример:

С начала вам нужно выбрать категории, которые хотите внести в таблицу (рисунок 3.39).

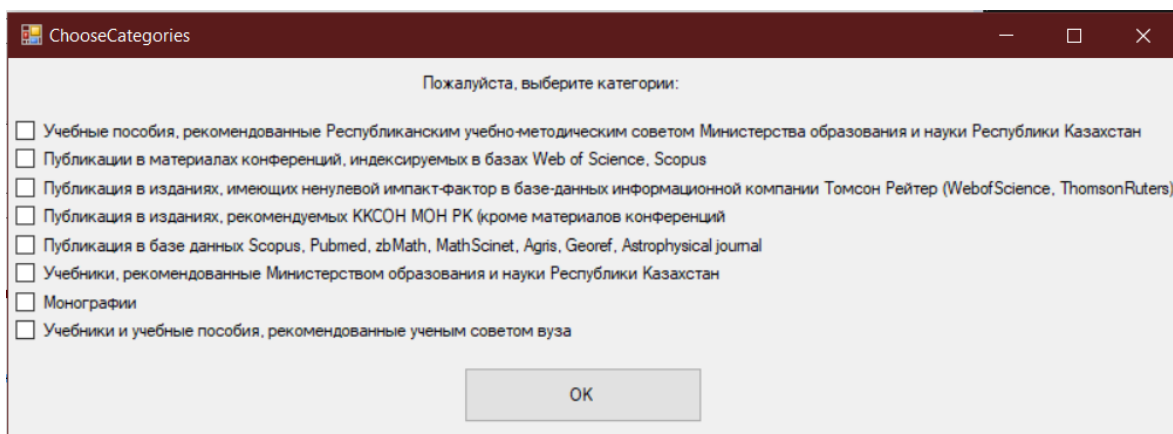


Рисунок 3.39 – Выбор категорий для ручного создания таблиц

Следующим шагом является внесение данных для таблицы (рисунок 56).

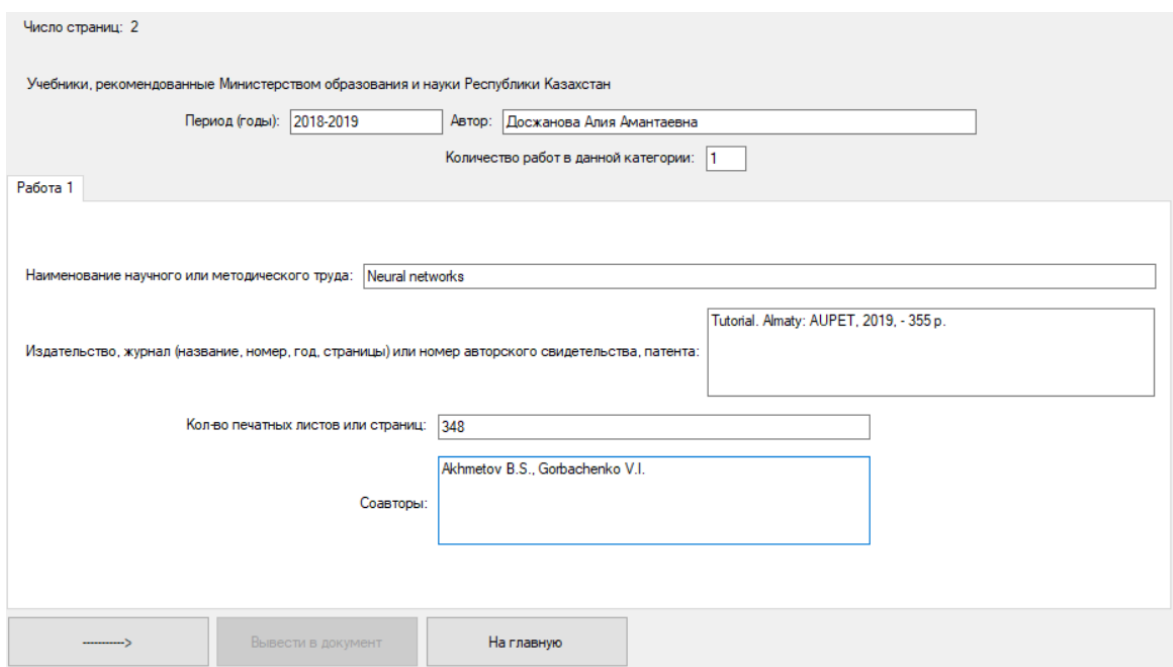


Рисунок 3.40 – Внесение данных в таблицу

Обратите внимание, что есть возможность выбрать количество работ для каждой категории. Только нет возможности перейти к следующей вкладке, пока не заполните все поля, а также нет возможности вернуться назад, после завершения заполнения и перехода (рисунок 3.41).

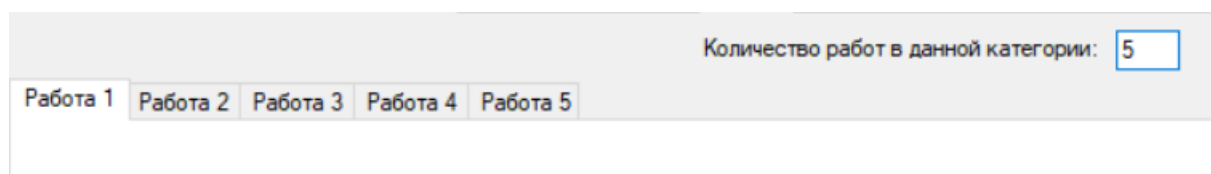


Рисунок 3.41 - Вкладки

После заполнения категории, то появляется возможность нажать на кнопку (рисунок 3.42).

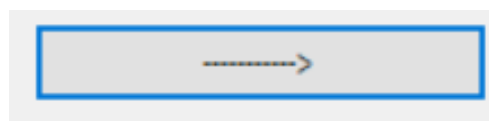


Рисунок 3.42 – Кнопка для перехода на следующую категорию

При завершении заполнения всех вкладок, появляется возможность нажать кнопку “Вывести в документ” (рисунок 3.43).

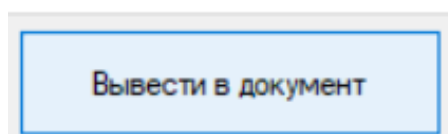


Рисунок 3.43 – Доступ к кнопке после завершения внесения данных

После нажатия на кнопку, выбирайте путь сохранения. Программа откроет документ. Есть возможность редактировать документ через программу в форме, где сохраняются файлы (рисунок 3.44).

СПИСОК
научных и научно-методических трудов за период 2018-2019 гг.
PhD, доцента кафедры «IT-инжиниринг»
НАО «Алматинский университет энергетики и связи имени Гумарбека Давкеева»
Досжановой Алии Амантаевны

№	Наименование	Издательство, журнал или номер патента	Кол-во страниц	Соавторы
Публикация в базе данных Scopus, Pubmed, zbMath, MathScinet, Agris, Georef, Astrophysical journal				
1	Neural networks	Tutorial. Almaty: AUPET, 2019, - 355 p.	348	Akhmetov B.S., Gorbachenko V.I.
2	Медициналық ақпараттық жүйелердегі мәліметтерді тұлғасыздандыру биометрия-нейрожелділік технологиялары	Алматы: АУЭС, 2018, – 108 с.	108	-
Монографии				
3		Алматы: КазГосЖенПУ , 2018. – 230 с.	111	Балабекова М.Ж., Кожамкулова Ж.Ж.
4	Models and algorithms for information processing in biometric-neural network systems.	Almaty: ПСТ, 2019. – 104 p.	104	Malikova F.U.
5	Development of a computer system for identity authentication using artificial neural networks	International Society for Stereology & Image Analysis (Thomson Reuters) Vol 36, No 1 (2017) EISSN1854-5165 ISSN 1580-3139 https://www.ias-iss.org/ojs/IAS/article/view/1612 Импакт-фактор - 1,424	124	T.Kartbayev, B.Akhmetov, K.Mukapil, A.Kalizhanova, G.Nabiyeva

Рисунок 3.44 – Созданная таблица

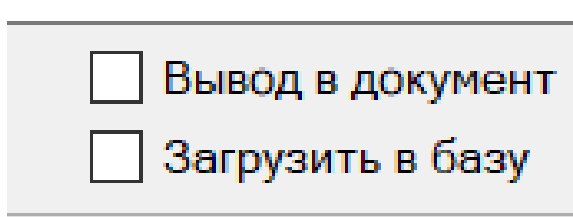
Программа автоматически ставит автора в родительный падеж

Досжановой Алии Амантаевны

Рисунок 3.45 – Перевод ФИО в родительный падеж

Также есть возможность создать таблицу из загруженных работ, которые хранятся в БД. На форме есть два режима (рисунок 3.45).

- вывод в документ;
- загрузить в бд.



Вывод в документ

Загрузить в базу

Рисунок 3.46 – Выбор между двумя режимами

По прошлым примерам можно было обратить внимание на примеры загрузки в бд. Включается режим “Вывод в документ”, открывается другой функционал, который немного схож с рисунок 3.39, но больше ориентирован на автоматизацию работу. Вам нужно выбрать автора и категорию (рисунок 3.46).

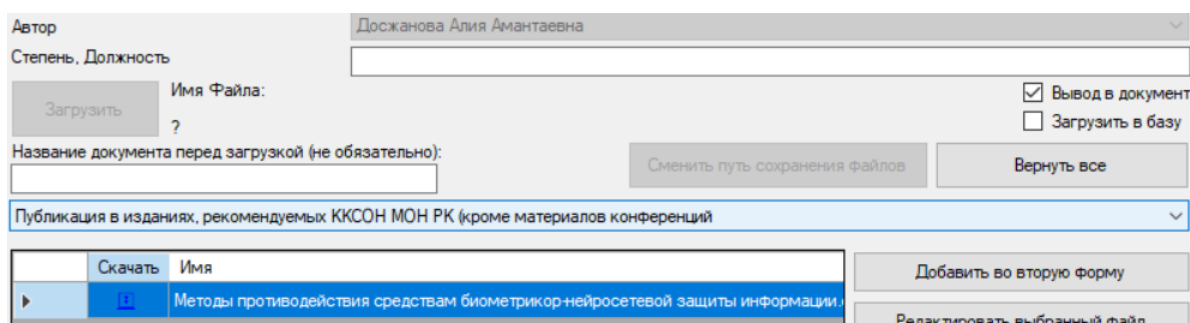


Рисунок 3.47 – Интерфейс при выборе “Вывод в документ”

Добавить все нужные работы во вторую таблицу (рисунок 3.47).



№	Наименование	Издатель журнал патента
1	Методы противодейс...	Доклады

Рисунок 3.48 – Добавление во вторую таблицу

Нажать на кнопку (рисунок 3.48).

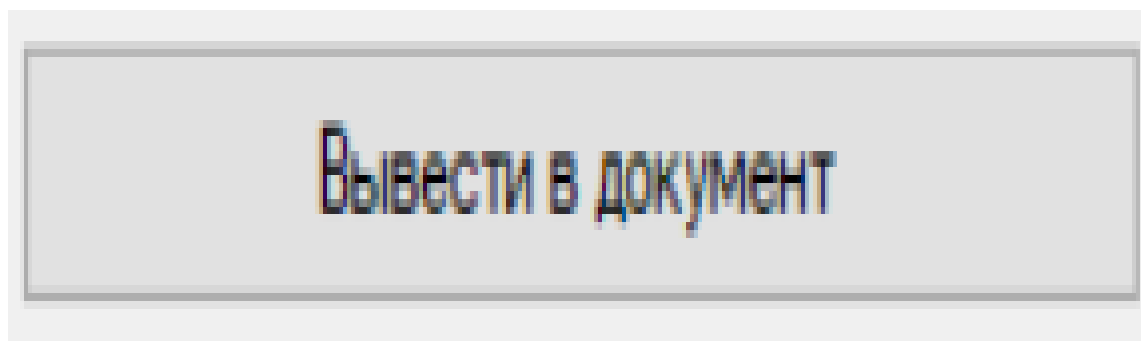


Рисунок 3.49 – Кнопка вывода в документ

Документ готов (рисунок 3.49).

СПИСОК
научных и научно-методических трудов за период 2017 - 2019 гг.
PhD, доцента кафедры «IT-инжиниринг»
НАО «Алматинский университет энергетикки и связи имени Гумарбека Даукеева»
Досжановой Алии Амантаевны

№	Наименование	Издательство, журнал или номер патента	Кол-во страниц	Соавторы
Учебники, рекомендованные Министерством образования и науки Республики Казахстан				
1		Алматы: КазГосЖенПУ, 2018. – 230 с.	10	Балабекова М.Ж., Кожамкулова Ж.Ж.
Монография				
2	Медициналық ақпараттық жүйелердегі мәліметтерді тұлғасыздандыру биометрия-нейрожелілік технологиялары.docx	Алматы: АУЭС, 2018, – 108 с.	10	Malikova F.U., Kartbayev T.S.,
3	Models and methods of development of hardware and software complex of mobile measurement of volume liquid.docx	Almaty: ПСТ, 2018, – 100 р.	10	Malikova F.U., Kartbayev T.S.,
4	Models and algorithms for information processing in biometric-neural network systems. .docx	Almaty: ПСТ, 2019. – 104 р.	10	Malikova F.U.
Публикация в изданиях, рекомендуемых ККСОН МОН РК (кроме материалов конференций)				
5	Методы противодействия средствам биометрикор-нейросетевой защиты информации.docx	Доклады НАН РК, – Алматы, - 2017, №4.- С. 29-34	10	Ахметов Б.С.,

Рисунок 3.50 – Готовый документ

Необходимо обратить внимание на столбец с количеством страниц. В каждом из столбцов количество страниц равно десяти. Программа достает количество страниц из документа. Нет необходимости ручного заполнения.

4 Обоснование эффективности внедрения проекта

4.1 Техничко - экономическое обоснование дипломных работ, связанных с разработкой программного продукта (ПП)

Важную роль в организации работы любого предприятия, в частности кафедры “ИТ-Инжиниринг”, является разработка собственной информационной системы. В настоящее время не существует такой информационной системы, которая в данный момент используется на кафедре “ИТ-Инжиниринг”.

Информационная система - это совокупность программно-аппаратных средств, способов и людей, которые обеспечивают сбор, хранение, обработку и выдачу информации для обеспечения подготовки и принятия решений. К основным компонентам информационных систем, используемых в экономике, относятся: программно-аппаратные средства, бизнес-приложения и управление информационными системами. Назначение информационных систем - создание современной информационной инфраструктуры для управления компанией.

Основной задачей информационной системы является:

- импорт литературы и трудов;
- хранение литературы и трудов;
- экспорт литературы и трудов;
- проведение анализа литературы и трудов;
- редактирование файлов внутри программы без использования проводника;
- облегчение работы и организация документов персонала.

Данная информационная система повышает эффективность и качество работы кафедры “ИТ-Инжиниринг”.

Главными задачами информационной системы является:

- облегчить работу персонала;
- ускорить процесс работы.

Информационная система – обязана при работе соблюдать интересы кафедры и обеспечивать полное, а также ритмичное выполнение планов работы персонала.

Предприятия, в частности кафедры “ИТ-Инжиниринг”, обязаны систематически улучшать свою экономическую работу для снижения себестоимости.

Основным объектом экономической части выступает информационная система, которая предоставляет решение задач, возникающих в экономических системах, и конечная цель функционирования является эффективное управление экономической системой.

Технико - экономическое обоснование разработки должно содержать:

- определение трудоемкости разработки ПП;
- расчет затрат на разработку ПП;
- определение возможной цены разработанного ПП;
- оценку социально - экономических результатов функционирования

ПП.

4.2 Трудоемкость разработки ПП

Трудоемкость – это экономический показатель, характеризующий показатель затраты рабочего времени на производство единицы продукции или на выполнение конкретной технологической операции. Создание ПП занимает примерно 112 часов для одной продукции.

Таблица 4.1 - Распределение работ по этапам и видам и оценка их трудоемкости

Этап разработки ПП	Вид работы на данном этапе	Трудоемкость разработки ПП, чел.× ч.
Планирование	Перед началом разработки нужно получить техническое задание и составить план по разработке: интерфейс, функции и прочее.	16
Разработка	Данный этап подразумевает создание самого ПП. Здесь происходит реализация всех идей из этапа планирования.	80
Тестирование	После создания ПП, нужно обязательно его протестировать для выявления ошибок	16
ИТОГО	трудоемкость выполнения дипломной работы	112

4.3 Расчет затрат на разработку ПП

Определение затрат на разработку ПП производится путем составления соответствующей сметы, которая включает следующие статьи:

- 1) Материальные затраты.
- 2) Затраты на оплату труда.
- 3) Социальный налог.
- 4) Амортизация основных фондов.
- 5) Прочие затраты.

Когда происходит процесс создания ПП, требуется обеспечить персонал материальными ресурсами для комфортной работы.

Таблица 4.2 - Затраты на материальные ресурсы

Наименование материального ресурса	Количество израсходованного материала	Цена за единицу, тг	Сумма, тг
Бумага	73	5	365
Картридж	1	20230	1300
Тонер	1	260	260
Ручка	5	150	750
Карандаш	5	100	500
Ластик	2	50	100
Корректор	2	150	300
Линейка	1	300	300
Visual Studio 2019	1	600000	600000
HP Pavilion	1	380000	380000
Принтер	1	37000	37000
Windows 10 Pro	1	75000	75000
ИТОГО затраты на материальные ресурсы			1095875

Общая сумма затрат на материальные ресурсы (ЗМ) определяется по формуле:

$$Z_c = 73 * 5 + 20230 + 260 + 5 * 150 + 5 * 100 + 2 * 50 + 2 * 150 + 300 + 600000 + 380000 + 37000 + 75000 = 1095875$$

Формула 4.1

Таблица 4.3 - Затраты на электроэнергию

Наимен. оборудов.	Мощность, кВт	Кэф. использ. мощности	Вре.раб. обор ПП, ч	Цена элек/энер, тг/кВт×ч	Кол-во	Сумма, тг
HP Pavilion	0,55	0,9	112	19,17	1	1180,87
Лампочка	0,08	0,9			3	515,29
Принтер	0,31	0,9			1	665,58
ИТОГО затраты на электроэнергию						2361,74

Общая сумма затрат на электроэнергию (ЗЭ) рассчитывается по формуле:

$$Z_3 = 0,55 * 112 * 19,17 + 0,08 * 112 * 19,17 + 0,31 * 112 * 19,17 = 2361,74$$

Формула 4.2

При разработке ПП требуется сотрудники имеющие знание в данной сфере и оплата их труда.

Затраты на оплату труда рассчитаны в таблице 4.4.

Таблица 4.4 - Затраты на оплату труда

Категория работника	Количество	Квалификация	Трудоемкость разработки ПП, чел.×ч	Часовая ставка, тг/ч	Сумма, тг
Разработчик	1	Бакалавриат	112	892,86	93750
ИТОГО затраты на оплату труда					93750

Таблица 4.5 - Социальный налог

СО(Социальные отчисления)	3,5	(ЗП - ОПВ)*3,5%	2 953,13
ВОСМСЮ (Отчисления на ВОСМСЮ)	2,0	ЗП*2%	1 875,00
СН (Социальный налог)	9,5	(ЗП-ОПВ-ВОСМС)*9,5%-СО	4 884,37
Всего уплаченные налоги			9 712,5

$$НР = ЗП \cdot \frac{Н_{НР}}{100} = 93\,750 \cdot 0,7 = 65\,625 \text{ тенге} \quad (4.3)$$

Амортизационные отчисления — это денежные средства, предназначенные для возмещения износа предметов, относящихся к основным средствам предприятия

Таблица 4.6 - Амортизация основных фондов (ОФ)

Наим. оборуд ПО	Стоимость обору. и ПО, тг	Годовая норма амортиз., %	Врем. Раб. ПП, ч	Сумма, тг
НР Pavilion	380000	14,29	16	4525,17
Принтер	37000	14,29	16	440,61
Windows 10 Pro	75000	33,33	16	2083,125
Visual Studio	600000	33,33	16	16665
ИТОГО амортизация основных фондов				23713,905

Применение основных фондов колеблется от 3 до 10 лет. Оборудование используется в течение 7 лет, а программное обеспечение – 3 года. Применяя формулу, заполним Таблицу 4.6 для отображения амортизации основных фондов.

$$H_{A1} = 100/7 = 14,29\%.$$

$$H_{A3} = 100/3 = 33,33\%.$$

$$Z_{ам} = \frac{380000 * 0,1429 * 16}{1 * 12 * 16} = 4525,17 \text{ тг}$$

$$Z_{ам} = \frac{37000 * 0,1429 * 16}{1 * 12 * 16} = 440,61 \text{ тг}$$

$$Z_{ам} = \frac{75000 * 0,3333 * 16}{1 * 12 * 16} = 2083,125 \text{ тг}$$

$$Z_{ам} = \frac{600000 * 0,3333 * 16}{1 * 12 * 16} = 16665 \text{ тг}$$

Также стоит учитывать прочие расходы:

- аренда помещения - 75000;
- коммунальные услуги – 5492,94.

Таблица 4.7 - Смета затрат на разработку ПП

Статьи затрат	Сумма, тг
1) Материальные затраты, в том числе:	
- материалы;	1 095 875
- электроэнергия.	2 361,74
2) Затраты на оплату труда;	93 750
3) Отчисления на социальные нужды;	14 748,6
4) Амортизация основных фондов;	23 713,905
5) Прочие затраты.	80 492,94
ИТОГО по смете	1 310 942,185

4.4 Определение договорной цены ПП

Величина возможной (договорной) цены ПП должна устанавливаться с учетом эффективности, качества и сроков ее выполнения на уровне, отвечающем экономическим интересам заказчика (потребителя) и исполнителя.

Договорная цена (ЦД) для прикладных ПП рассчитывается по формуле:

$$Ц_{д} = Z_{нир} * \left(1 + \frac{P}{100}\right) = 1419342,185 * \left(1 + \frac{30}{100}\right) = 1845144,84 \quad (4.4)$$

$Z_{\text{нпр}}$ - затраты на разработку ПП (из таблицы 1.6), тг;

P - средний уровень рентабельности ПП. %

Далее определяется цена реализации с учетом налога на добавленную стоимость (НДС), ставка НДС устанавливается законодательно Налоговым Кодексом РК. На 2013 год ставка НДС установлена в размере 12%.

Цена реализации с учетом НДС рассчитывается по формуле:

$$C_p = C_d + C_d * \text{ндс} = 1845144,84 + 1845144,84 * 0,12 = 2066562,221 \quad (4.5)$$

Данная информационная система является не единственной в своем роде, но имеет больше преимуществ, чем свои аналоги. ИС является с экономической точки зрения выгодным проектом. Цена является наименьшей на рынке. Средняя стоимость разработки аналогичных систем обойдется в 2 300 000 тг. Разрабатываемая система обойдется в 2 066 562 тг.

4.5 Расчет результатов от создания и использования ИС

Информационная система с целью управления проектированием специализирована в основную очередь с целью контролирования деятельности рабочего персонала. ИС облегчает процесс принятия решений, но кроме того может увеличить эффективность персонала, за счет уменьшения времени требуемой работы.

Для оценки экономии от использования разрабатываемой ИС необходимо сравнить эксплуатационные расходы с ее применением и без.

Статьи затрат при применении ИС включают в себя:

- заработная плата специалиста, осуществляющего поддержку и сопровождение системы;
- износ оборудования;
- накладные расходы.

Данная система не предполагает расхода каких-либо материалов, так как работа полностью перейдет из бумажной в цифровой. Это значит, что расходы на расходные предметы пропадут.

В расходные материалы входит бумага, картридж.

Пачка бумаги стоит – 1400 тг.

Картридж – 16000.

В месяц уходит 5 пачек бумаги и один картридж.

$$P = ((1400 \cdot 5) + (16000)) \cdot 12 = 276\,000 \text{ тенге}$$

В распоряжении находятся семь персональных компьютера (стоимость одного ~200 000 тенге), принтер (90 000 тенге) и прочая компьютерная периферия (~30 000 тенге); итого затрат на оборудование – 320 000 тенге.

Износ оборудования рассчитывается исходя из 25 % амортизационных отчислений за год.

$$A = (1\,400\,000 + 90\,000 + 30\,000) \cdot 0,25 = 380\,000 \text{ тенге}$$

Данной работой занимаются 2 сотрудника, нанятые для контроля бумажного оборота методических трудов и прочих документов.

Каждый сотрудник получает – 100 000 тенге

$$\text{ЗП} = 2\,400\,000 \text{ тенге}$$

$$\text{ОПВ} = 2\,400\,000 \cdot 0,1 = 240\,000$$

$$\text{СО} = (2\,400\,000 - 240\,000) \cdot 0,035 = 75\,600$$

$$\text{ВОСМС} = 2\,400\,000 \cdot 0,02 = 48\,000$$

$$\text{СН} = (2\,400\,000 - 240\,000 - 48\,000) \cdot 0,095 - 75\,600 = 125\,040$$

$$\text{ОТ} = \text{СО} + \text{СН} + \text{ВОСМС} = 75\,600 + 125\,600 + 48\,000 = 249\,200 \text{ тенге}$$

Общие накладные расходы составят:

$$\text{НР} = \text{ЗП} \cdot \frac{N_{\text{НР}}}{100} = 2\,400\,000 \cdot 0,7 = 1\,680\,000 \text{ тенге} \quad (4.6)$$

Статьи затрат без применения ИС включают в себя:

- заработная плата сотрудников;
- износ используемого ими оборудования;
- расход материалов (к примеру, канцелярия – бумага, картридж и т.д.);
- накладные расходы.

Таблица 4.8 – Годовые эксплуатационные затраты

Статьи	Без применения ИС (1 сотрудник – куратор проекта)	С применением ИС
Год з/п	2 400 000	1 125 000
Соц. отч. и налоги	249 200	116 550
Расходуемые материалы	276 000	0
Амортизационные отчисления	380 000	23715
Накладные расходы	1 680 000	787 500
Всего	4 985 200	2 052 765

Ожидаемая условно-годовая экономия определяется по формуле:

$$\text{Э}_{\text{уг}} = C_1 - C_2 + \sum \text{Э}_i \quad (4.7)$$

где $\text{Э}_{\text{уг}}$ - величина экономии, тенге;

C_1 и C_2 – показатели текущих затрат по базовому и внедряемому вариантам, тенге:

$\Sigma \Delta_i$ – ожидаемый дополнительный эффект от различных факторов, тенге.

$$\Delta_{\text{уг}} = C_1 - C_2 = 2\,932\,435 \text{ тенге}$$

4.6 Расчет основных показателей экономической эффективности

Так как разработанная информационная система несет более социальный эффект, чем экономический, целесообразно оценивать эффективность за счет экономии в сравнении с предыдущим периодом работы без использования ИС.

Величина ожидаемого годового экономического эффекта от внедрения ИС рассчитывается по формуле:

$$\Delta_r = \Delta_{\text{уг}} - K \cdot E_H \quad (4.7)$$

где Δ_r - ожидаемый годовой экономический эффект, тенге;

$\Delta_{\text{уг}}$ — ожидаемая условно-годовая экономия, тенге;

K — капитальные вложения, тенге;

E_H - нормативный коэффициент экономической эффективности капитальных вложений.

Нормативный коэффициент экономической эффективности капитальных вложений определяется по формуле:

$$E_H = \frac{1}{T_H} \quad (4.7)$$

где T_H — нормативный срок окупаемости капитальных вложений, лет.

Нормативный срок окупаемости капитальных вложений. принимается исходя из срока морального старения - технических средств и проектных решений ИС ($T_H=1,2,3\dots n$), для программных продуктов срок окупаемости принимаем равным 4 года.

$$E_H = \frac{1}{4} = 0,25$$

$$\Delta_r = 2\,932\,435 - 2\,066\,562 \cdot 0,25 = 2\,415\,795 \text{ тенге}$$

Расчетный коэффициент экономической эффективности капитальных вложений составляет:

$$E_p = \frac{\Delta_{\text{уг}}}{K}$$

где E_p - расчетный коэффициент экономической эффективности капитальных вложений;

$\mathcal{E}_{\text{уг}}$ — ожидаемая условно-годовая экономия, тенге;
 K — капитальные вложения на создание системы, тенге.

$$E_p = \frac{2\,932\,435}{2\,066\,562} = 1,42$$

Расчетный срок окупаемости капитальных вложений составляет:

$$T_p = \frac{1}{E_p} \quad (4.8)$$

где E_p - коэффициент экономической эффективности капитальных вложений.

$$T_p = \frac{1}{1,42} = 0,7 \text{ года} \approx 8,4 \text{ месяцев}$$

Таблица 4.9 – Показатели сравнительной экономической эффективности от внедрения программного продукта

Наименование показателей	Значение
Условная годовая экономия затрат, тенге	2 415 795
Коэффициент экономической эффективности капитальных вложений (E_p)	1,42
Срок окупаемости капитальных вложений (T_p)	0,7

В итоге имеется, что разработанная информационная система позволяет упростить процесс работы персонала, а также сэкономить на различных факторах. Для использования ИС больше не требуются сотрудники контроля бумажного оборота методических трудов и прочих документов. Стоит еще учесть, что можно значительно снизить расход материалов, так как уже не требуется распечатывания всех документов и всей документацией можно работать напрямую в программе, а также хранения в ИС.

Ожидаемый годовой экономический эффект составил 2 932 435 тенге. Информационная система окупится за 8 месяцев с момента начала использования.

5 Безопасность жизнедеятельности

Анализ опасных и вредных производственных факторов при работе на ПК

Общие положения:

Определение класса условий труда на рабочих местах проводится с целью:

- определения значимости оздоровительных мероприятий;
- формирование базы данных согласно имеющимся условиям труда;
- установление выплат, а также компенсации за вредные условия труда.

Необходимо комплексно оценить условия труда, чтобы провести аттестации рабочего места. Оценка условий труда выполняется по особой методике, на основе анализа уровней вредных и опасных факторов на данном рабочем месте.

Вредоносный производственный фактор - причина среды и рабочего процесса, что способен спровоцировать сокращение трудоспособности, патологию, послужить причиной к нарушению здоровья потомства.

Вредоносными могут быть:

- физиологические причины: степень нагретости, сырость и движение воздуха, вызывающие и не вызывающие ионизацию излучения, фоновый шум, колебания, недостаток света освещенность;
- химические причины: загрязненность и пыленность воздуха;
- биологические причины: вредные бактерии;
- причины тяжести труда: физиологическая, подвижная и неподвижная нагрузка; огромное количество одинаковых движений, большое число наклона тела, не комфортная рабочая поза;
- причины напряженности труда: умственные, чувственные, эмоциональные нагрузки, однообразие и длительность работы.

Угрожающие производственные причины - причины среды и рабочего процесса, который может привести к резкому ухудшению состояния здоровья, повреждения, а также кончине.

Угрожающий производственный фактор состоит из: электрический ток, пламя, горячая поверхность, подвижные части оборудования, чрезмерное давление, острые кромки предметов, высота и.т.п.).

Классы условий труда

В основном все условия труда, которые могут встретиться на практике, подразделяются на четыре класса по уровням вредных и опасных факторов.

1-й класс - оптимальный (совокупность причин позволяет сохранять здоровье, поддерживать высокую работоспособность).

2-й класс - допустимый (причины среды и рабочего процесса не превышают поставленных норм, а допустимые изменения функционального состояния организма, вызванные усталостью, утомлением,

восстанавливаются во время регламентированного отдыха или к началу следующей смены).

1 и 2 классы соответствуют безопасным условиям труда.

3 класс - вредный (наличие вредных факторов, оказывающих неблагоприятное действие на организм работающего и/или его потомство).

Вредные условия труда по степени изменений в организме работающих подразделяются на 4 степени:

- 1 степень 3-го класса (3.1) - вызывает обратимые изменения в организме и обуславливает риск развития заболевания;

- 2 степень 3-го класса (3.2) - вызывает стойкие функциональные нарушения, временную утрату трудоспособности, начальные признаки профессиональной патологии;

- 3 степень 3-го класса (3.3) - вызывает развитие профессиональной патологии в легкой форме, рост общей хронической заболеваемости;

- 4 степень 3-го класса (3.4) - вызывает выраженные формы профессиональных заболеваний, высокий уровень общей заболеваемости;

4 класс - экстремальный, опасный (4) - производственные факторы даже в течение части рабочей смены создают угрозу для жизни, создают высокий риск острых профессиональных поражений.

На практике в первую очередь для оценки класса условий труда устанавливают, соответствует ли нормам санитарно-гигиенических показателей:

- содержание вредных веществ в воздухе рабочей зоны;
- значения параметров микроклимата;
- уровни шума и вибрации, инфра- и ультразвука;
- наличие электромагнитных и ионизирующих излучений;
- параметры световой среды производственных помещений.

Необходимо исследовать класс условия труда по показателям тяжести и напряженности трудового процесса.

Комфорт рабочей позы можно определить в первую очередь параметрами основных элементов рабочего места: стола, стула, оборудования и т.д. В настоящее время большинство из этих параметров стандартизированы и включены в санитарно-гигиенические и эргономические нормативно-правовые акты. Чтобы обеспечить комфортную рабочую позу, элементы рабочего места должны удовлетворять требованиям санитарных норм и правил. В таблице 3 приведены оптимальные размеры основных элементов рабочего места. Размещение оборудования должно отвечать следующим требованиям:

- экран видеомонитора должен находиться на расстоянии 600 - 700 мм от глаз пользователя;

- клавиатуру следует располагать на поверхности стола на расстоянии 100;

- 300 мм от края, обращенного к пользователю или на специальной выдвижной панели стола.

Рабочее место пользователя не считается удобным, если рабочее место не отвечает указанным требованиям, а рабочая поза - свободной. Пользователь находится в вынужденной, неудобной позе 75 % рабочего времени, при работе за компьютером непрерывно в течение 6 часов за время рабочей смены, отсюда следует, что условия труда по фактору рабочей позы оцениваются по классу 3.2. Вынужденная рабочая поза вызывает у пользователей ПК статическое перенапряжение мышц шейно-плечевой области и спины, создающее предпосылки для быстрой утомляемости и развития патологии опорно-двигательного аппарата.

По напряженности и тяжести труда СанПиН устанавливает три категории работ: А, Б, В. Напряженность этих работ не должна превышать класс условий труда 3.1 по следующим критическим параметрам:

а) работа по считыванию информации с экрана ПЭВМ по предварительному запросу (работа пользователя программных комплексов, операторы технологических процессов) - по максимальному количеству считываемых знаков: не более 60000 знаков за смену;

б) работа по вводу информации (работа секретаря, наборщика издательства, бухгалтера, программиста при наборе текста программы, студента при оформлении расчетно-пояснительной записки) - по суммарному количеству вводимых знаков: не более 40000 в смену;

в) творческая работа с компьютером в режиме диалога (менеджеры, дизайнеры, творческие работники, студенты в процессе обучения) - по суммарному количеству времени работы за экраном видеотерминала: не более 4 часов в смену.

Общая оценка условий труда с учетом комбинированного действия производственных факторов проводится следующим образом:

1) На основе результатов измерений и экспертных заключений классы условий труда для каждого рассматриваемого фактора сводятся в таблицу;

2) Наиболее высокий класс и степень вредности определяют общую оценку;

3) В случае, если три и более факторов относятся к классу 3.1, то общая оценка условий труда соответствует классу 3.2;

4) При наличии двух и более факторов классов 3.2; 3.3 и 3.4 - условия труда оцениваются на одну степень выше.

При сокращении времени контакта вредными факторами (защита временем) условия труда могут быть оценены как менее вредные, но не ниже класса 3.1.

Рабочее место считается аттестованным, если класс условий труда не превысил 2.

Схема рабочего места с ПК

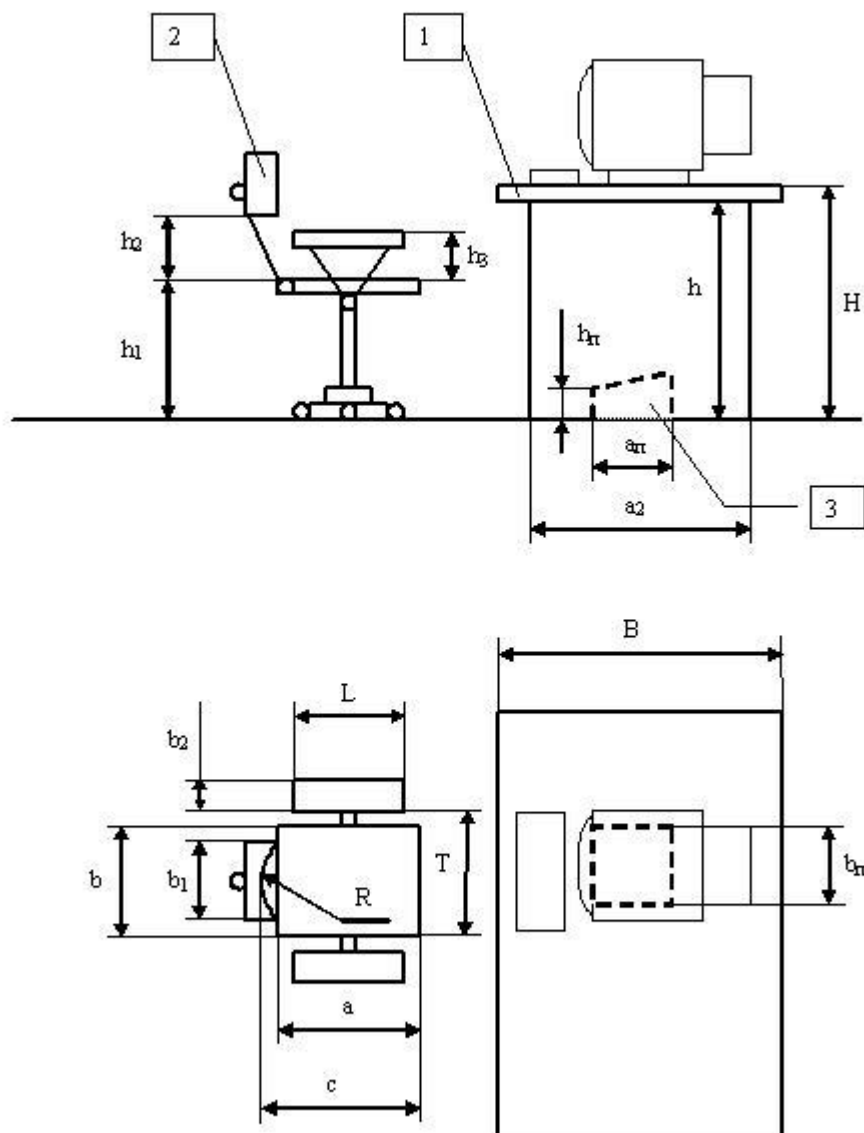


Рисунок 5.1 – Добавление во вторую таблицу

Таблица 5.1 - Результаты оценки условий труда на рабочем месте, оснащённом ПК

Показатели Тяжести и напряженности трудового процесса	Характеристика показателя по результатам фактической оценки	Класс условий труда
При локальной нагрузке	Служащему требуется работать кистями и пальцами рук для заполнения необходимой документации в ИС	1

Продолжение таблицы 5.1

Рабочая поза	Персонал имеет комфортные условия труда.	1
Содержание работы	При работе с ИС требуется заполнения данных по инструкции	3.1
Степень сложности задания	Работы является не сложной, но внимательность необходима, чтобы все верно заполнить.	2
Характер выполняемой работы	Персонал состоит из преподавателей, которым необходимо проводить параллельно занятия, а также успевать выполнять работы по кафедре.	3.1
Плотность сигналов (звуковых, световых) и сообщений в среднем за 1 час работы	Во время работы исполнитель получает множество сигналов во время работы.	2
Наблюдение за экранами видеотерминалов (часов в смену)	В основном наблюдение идет с перерывом на занятия и не требует долго времени	2
Степень ответственности	Исполняющему главное правильно внести или вынести данные	1
Степень риска для собственной жизни	Нет факторов, которые могут привести к смерти.	1
Степень риска для жизни других лиц	Нет факторов, которые могут привести к смерти других лиц	1
Число элементов (приемов), необходимых для выполнения простого задания или в многократно повтор. операциях	Программа состоит из множества элементов.	2
Продолжительность в секундах выполнения простых произв. зад. или повт. операций	Каждое метод. пособ. требует прим. минуту для импорта в базу данных	2
Фактическая продолжительность рабочего дня	Каждый сотрудник работает примерно 8-9 часов	2
Сменность работы	Работа в одну смену	1

Заключение

В данном дипломной проекте рассматривается информационная система. Информационная система позволяет производить загрузку, скачивание, редактирование и создание таблиц. Вы можете создавать таблицы вручную и автоматически.

Главные отличия этой программы от сервисов, которые использует кафедра:

- 1) возможность автономной работы;
- 2) программа адаптирована под нужды кафедры;
- 3) автоматизация рабочего процесса.

Сервисы, которая использовала кафедра, не была адаптирована под работу кафедры и предназначена для широкого круга лиц. В этих сервисах присутствовала встроенная документация, но требовала ручного заполнения, что неэффективно и тратит много времени, а также сил.

В проекте все было реализовано для эффективной работы кафедры.

В первой главе рассматриваются альтернативные системы, которые не столь эффективны по сравнению с разрабатываемой системой.

Во второй главе описываются причины выбора среды разработки и языка программирования – Visual Studio и C#, которые имеют богатый функционал для полной реализации разрабатываемой системы, а также ее полное проектирование и продумывание концепции системы.

Третья глава состоит из описания процесса разработки информационной системы. В этой главе присутствует описание интерфейса и кода программы. Программа не требует сильного железа и может работать на бюджетных компьютерах.

В четвертой главе демонстрируется эффективность данной программы. В итоге мы имеем, что разработанная информационная система позволяет упростить процесс работы персонала, а также сэкономить на различных факторах. Для использования ИС больше не требуются сотрудники контроля бумажного оборота методических трудов и прочих документов. Стоит еще учесть, что можно значительно снизить расход материалов, так как уже не требуется распечатывания всех документов и всей документацией можно работать напрямую в программе, а также хранения в информационной системе. Ожидаемый годовой экономический эффект составил 2 932 435 тенге. Информационная система окупится за 8 месяцев с момента начала использования.

В пятой главе производится анализ опасных и вредных производственных факторов при работе на ПК. Необходимые мероприятия проведены и устранены все возникшие проблемы, которые могли случиться с персоналом.

Список литературы

- 1 Перлова, О.Н. Проектирование и разработка информационных систем: Учебник / О.Н. Перлова, О.П. Ляпина, А.В. Гусева. - М.: Academia, 2017. - 416 с;
- 2 Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 с;
- 3 Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 с;
- 4 Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. - М.: Вильямс, 2015. - 266 с;
- 5 Бишоп, Дж. C# в кратком изложении / Дж. Бишоп, Н. Хорспул. - М.: Бином. Лаборатория знаний, 2013. - 472 с;
- 6 Вагнер, Билл C# Эффективное программирование / Билл Вагнер. - М.: ЛОРИ, 2013. - 320 с;
- 7 Зиборов, В.В. Visual C# 2012 на примерах / В.В. Зиборов. - М.: БХВ-Петербург, 2013. - 480 с;
- 8 Зиборов, Виктор Visual C# 2010 на примерах / Виктор Зиборов. - М.: "БХВ-Петербург", 2011. - 432 с;
- 9 Ишкова, Э. А. Самоучитель C#. Начала программирования / Э.А. Ишкова. - М.: Наука и техника, 2013. - 496 с;
- 10 Касаткин, А. И. Профессиональное программирование на языке си. Управление ресурсами/А.И. Касаткин. - М.: Высшая школа, 2012. - 432 с;
- 11 Лотка, Рокфорд C# и CSLA .NET Framework. Разработка бизнес-объектов / Рокфорд Лотка. - М.: Вильямс, 2010. - 816 с;
- 12 Мак-Дональд, Мэтью Silverlight 5 с примерами на C# для профессионалов / Мэтью Мак-Дональд. - М.: Вильямс, 2013. - 848 с;
- 13 Марченко, А. Л. Основы программирования на C# 2.0 / А.Л. Марченко. - М.: Интернет-университет информационных технологий, Бином. Лаборатория знаний, 2011. - 552 с;
- 14 Подбельский, В. В. Язык C#. Базовый курс / В.В. Подбельский. - М.: Финансы и статистика, Инфра-М, 2011. - 384 с;
- 15 Прайс, Джейсон Visual C# 2.0. Полное руководство / Джейсон Прайс, Майк Гандэрлой. - М.: Век +, Корона-Век, Энтроп, 2010. - 736 с;
- 16 Рихтер, Джеффри CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C# / Джеффри Рихтер. - М.: Питер, 2013. - 928 с;
- 17 Смоленцев, Н. К. MATLAB. Программирование на Visual C#, Borland JBuilder, VBA (+ CD-ROM) / Н.К. Смоленцев. - М.: ДМК Пресс, 2011. - 456 с;
- 18 Троелсен, Эндрю Язык программирования C# 5.0 и платформа .NET 4.5 / Эндрю Троелсен. - М.: Вильямс, 2015. - 486 с.

Приложение А

Техническое задание

1 Общие сведения

1.1 Наименование системы

Полное наименование системы:

Информационная система “Кафедра”

1.2 Сроки начала и окончания работ

Дата начала: 15.01.2020

Дата окончания: 31.04.2020

2 Назначение и цели создания системы

2.1 Назначение системы

Информационная система предназначена для импорта, экспорта, редактирования методических и автоматизации процессов работы персонала.

3 Рекомендации к разработке программы

Информационная система может быть разработана на языке С# с использованием локальной базой данных в среде Visual Studio.

4 Требование к внешнему виду системы

Обязательным условием является простота и понятливость интерфейса.

5 Технические требования

Информационная система не требует мощных персональных компьютеров и может работать на более простом железе.

6 Экономические требования

- возможная (договорная) цена продукта составила 1 851 845 тг;
- стоимость разработки продукта составила 1 310 942 тг.