Частное учреждение образования

«Колледж бизнеса и права»

ДОПУЩЕН К ЗАЩИТЕ

Заместитель директора

по учебной работе

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.В.Малафей

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022

Создание обучающей программы

по учебной дисциплине «Страховое дело»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ДП Т.891062.401

Председатель цикловой комиссии (Т.Г. Багласова )

Руководитель проекта (К.О. Якимович )

Консультант по экономической части (Е.А. Андреева )

Консультант по охране труда (В.С. Кудласевич )

Учащаяся (М.С Новикова )

Рецензент ( )

2022

Содержание

Изм.

Изм.

Изм.

Изм.

Лист

Лист

Лист

Лист

№ докум.

№ докум.

№ докум.

№ докум.

Подпись

Подпись

Подпись

Подпись

Дата

Дата

Дата

Дата

Лист

Лист

Лист

Лист

3

3

3

3

ДП Т.891062.401 ПЗ

КП Т.716007.401 ПЗ

КП Т.716009.401 ПЗ

КП Т.716007.401 ПЗ

Разраб.

Разраб.

Разраб.

Разраб.

*Новикова М.С.*

*Голубев И.В.*

*Гринь А.М.*

*Голубев И.В.*

Провер.

Провер.

Провер.

Провер.

Якимович К.О.

Банцевич. С.В

Банцевич. С.В

Банцевич. С.В

Т. контр.

Т. контр.

Т. контр.

Т. контр.

Якимович К.О.

Н. контр.

Н. контр.

Н. контр.

Н. контр.

Багласова Е.В.

Утверд.

Утверд.

Утверд.

Утверд.

Багласова Т.Г.

Создание обучающей программы по учебной дисциплине «Страховое дело»

Лит.

У

Лит.

У

Лит.

У

Лит.

У

Листов

Листов

Листов

Листов

48

50

55

50

КБП

КБП

КБП

КБП

[Введение 4](#_Toc98752984)

[1 Объектно-ориентированный анализ и проектирование систем 5](#_Toc98752985)

[1.1 Сущность задачи 5](#_Toc98752986)

[1.2 Проектирование модели 5](#_Toc98752987)

[2 Вычислительная система 8](#_Toc98752988)

[2.1 Требования к аппаратным и операционным ресурсам 8](#_Toc98752989)

[2.2 Инструменты разработки 8](#_Toc98752990)

[3 Проектирование задачи 10](#_Toc98752991)

[3.1 Требования к приложению 10](#_Toc98752992)

[3.2 Концептуальный прототип 10](#_Toc98752993)

[3.3 Организация данных 13](#_Toc98752994)

[3.4 Функции и элементы управления 15](#_Toc98752995)

[3.5 Проектирование справочной системы приложения 16](#_Toc98752996)

[4 Описание программного средства 17](#_Toc98752997)

[4.1 Общие сведения 17](#_Toc98752998)

[4.2 Функциональное назначение 17](#_Toc98752999)

[4.3 Входные и выходные данные 18](#_Toc98753000)

[5 Методика испытаний 19](#_Toc98753001)

[5.1 Технические требования 19](#_Toc98753002)

[5.2 Функциональное тестирование 19](#_Toc98753003)

[6 Применение 22](#_Toc98753004)

[6.1 Назначение программы 22](#_Toc98753005)

[6.2 Условия применения 22](#_Toc98753006)

[6.3 Справочная система 22](#_Toc98753007)

[7 Охрана труда и окружающей среды 23](#_Toc98753008)

[7.1 Правовые, нормативные, социально-экономические и организационные вопросы охраны труда 23](#_Toc98753009)

[7.2 Разработка мер по нормализации вентиляции в помещении 25](#_Toc98753010)

[7.3 Пожарная безопасность 26](#_Toc98753011)

[7.4 Охрана окружающей среды 28](#_Toc98753012)

[8 Экономический раздел 30](#_Toc98753013)

[8.1 Обоснование необходимости выведения на рынок программного продукта 30](#_Toc98753014)

[8.2 Составление плана по разработке программного продукта 30](#_Toc98753015)

[8.3 Расчет затрат на разработку программного продукта 31](#_Toc98753016)

[8.4 Экономический эффект у разработчика программного продукта 35](#_Toc98753017)

[8.5 Экономический эффект у пользователя программного продукта 37](#_Toc98753018)

[Заключение 39](#_Toc98753019)

[Список информационных источников 40](#_Toc98753020)

[Приложение А Текст программных модулей 41](#_Toc98753021)

Введение

При составлении лекций и тестов можно столкнуться с рядом проблем:

– множество информации в рукописном виде;

– индивидуальная проверка знаний учеников.

Поэтому важно автоматизировать учебный процесс, тем самым предоставляя возможность тратить меньше времени на составление лекций, тестов, проведение проверки знаний учащихся.

Целью дипломного проекта является разработка программы «Страховое дело.exe» для автоматизации работы учебного процесса, которая будет осуществлять ведение базы данных, содержащей информацию о разделах, темах, лекциях, тестов, вопросы к тестам по учебной дисциплине «Страховое дело».

Решение поставленных задач отражено в пояснительной записке, которая состоит из восьми разделов и содержит необходимую и достаточную информацию по использованию данного программного средства [2].

В первом разделе «Объектно-ориентированный анализ и проектирование системы» рассматривается сущность и актуальность поставленной задачи, описание существующих аналогов, проектирование модели, отображающей функциональную структуру объекта.

Второй раздел «Вычислительная система» содержит описание вычислительной системы: технические характеристики персонального компьютера, требования, которые будут предъявляться к персональному компьютеру, описание операционной системы, языка реализации и языка моделирования.

Третий раздел «Проектирование задачи» включает требования к программному средству, концептуальный прототип, логическую и физическую организацию данных в контексте среды разработки, описание функций пользователя в рамках среды разрабатываемого программного средства, проектирование справочной системы программного средства.

Четвертый раздел «Описание программного средства» отражает общие сведения о программе, функциональное назначение, структуру входных и выходных данных.

В пятом разделе «Методика испытаний» рассматриваются требования к техническим средствам для проведения испытания, требования к характеристикам программы применительно к условиям эксплуатации, требования к информационной и программной совместимости. Также описывается порядок проведения функционального тестирования.

Шестой раздел «Применение» содержит информацию, необходимую в процессе эксплуатации программного средства: его назначение, условия применения, а также приводится описание справочной системы.

Седьмой раздел «Охрана труда и окружающей среды» содержит информацию по вопросам трудового законодательства, требования пожарной безопасности и охраны окружающей среды, а также информацию о разработке мер по нормализации вентиляции в помещениях.

В восьмом экономическом разделе описывается технико-экономическое обоснование разработки программного средства, составляется план по разработке программного средства, определяется цена программного средства и экономическая эффективность разработки.

В заключении описывается выполнение поставленной задачи, степень соответствия проектных решений задания, причины несоответствия, если таковые имеются.

В приложении А представлен текст программных модулей и в приложении Б – формы выходных документов.

Графическая часть представлена диаграммами вариантов использования, классов, последовательности, деятельности и компонентов.

1. Объектно-ориентированный анализ и проектирование системы
   1. Сущность задачи

Темой дипломного проекта является «Создание обучающей программы по учебной дисциплине «Страховое дело».

Предметной областью решаемой задачи является электронное средство обучения «Страховое дело». Объектом решаемой задачи является автоматизация учебного процесса Республиканского Института Профессионального образования.

Компьютеры проникли во все сферы деятельности человека, начиная с начального образования и заканчивая изучением новейших технологий, изучения новых видов материи, неизвестных пока человечеству. Применение компьютерных технологий облегчает процесс образования в средних и высших учебных заведениях как самих учеников, студентов, так и рабочего персонала.

Благодаря разнообразию программного и аппаратного обеспечения сегодня возможно использование всех потенциальных возможностей компьютерных технологий. Это позволяет хранить огромное количество информации, занимая при этом минимальное место. Также компьютерные технологии позволяют быстро эту информацию обрабатывать и держать ее в защищенном виде.

Суть программного средства «Страховое дело» состоит в том, чтобы облегчить работу учителей и учеников от ненужной деятельности. Учителям быстро и удобно добавлять лекции и проверять усвоенный материал, ученикам изучать материал и проходить тестирование на проверку знаний.

Разрабатываемый модуль должен обеспечивать следующие функции:

* ведение базы данных (добавление записей пользователей), содержащей уникальные ключи доступа пользователей;
* добавление лекций, тестов, ответов на тесты;
* просмотр лекций;
* просмотр глоссария, поиск нужного слова;
* прохождение тестов;
* автоматическое формирование результатов тестирования.

Аналогов данного продукта нет либо он не распространяется для общего пользования.

* 1. Проектирование модели

Цель моделирования данных состоит в обеспечении разработчика информационной системы концептуальной схемой базы данных в форме одной или нескольких локальных моделей, которые относительно легко могут быть отображены в любую систему без данных [3].

Наиболее распространенным средством моделирования данных является диаграмма «Сущность-связи» (ERD). С их помощью определяется важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи). ERD непосредственно используются для проектирования реляционных баз данных. Нотация ERD была впервые введена П. Ченом и получил дальнейшее развитие в работах Баркера.

Диаграмма «Сущность-связи» представлена на рисунке 1.

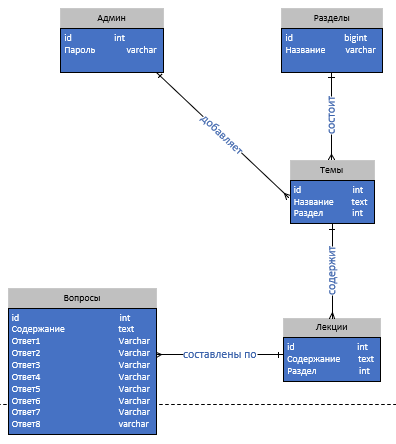


Рисунок 1

Исходя из предметной области можно выделить следующие сущности разработки: «Разделы», «Темы», «Лекции», «Вопросы», «Админ».

Сущность склад «Разделы» содержит атрибут название.

Сущность «Темы» содержит атрибуты:

* название;
* раздел.

Сущность «Лекции» содержит атрибуты:

* содержание;
* раздел.

Сущность «Вопросы» содержит атрибуты:

* содержание;
* ответ1;
* ответ2;
* ответ3;
* ответ4;
* ответ5;
* ответ6;
* ответ7;
* Ответ8.

Сущность «Админ» содержит атрибут пароль.

Суть диаграммы вариантов использования заключается в том, что проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью, так называемых вариантов использования.

К «extend» относится входная информация:

* подключение базы данных лекций, разделов, тем, вопросов, ответов;
* справочная система.

К «include» относится:

* просмотр лекций;
* просмотр тестов.

Диаграмма вариантов использования представлена на листе 1 графической части.

Диаграмма классов служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Диаграмма может отражать различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывать их внутреннюю структуру и типы отношений. Для электронного средства обучения «Страховое дело» в программном средстве реализовано пять классов: «Разделы», «Темы», «Лекции», «Вопросы» и «Админ». Класс «Вопросы» содержит методы и функции необходимые для работы тестирования. Классы «Лекции», «Разделы», «Темы» содержат в себе функции для просмотра информации. Класс «Админ» содержит в себе методы для входа в программное средство от имени администратора. Описанные классы представлены на листе 2 графической части.

При моделировании поведения проектируемой или анализируемой системы возникает необходимость детализировать особенности алгоритмической и логической реализации выполняемых системой операций. Для моделирования процесса выполнения операций в языке UML используются так называемые диаграммы деятельности.

Диаграммы деятельности – частный случай диаграмм состояний. Основная цель использования таких диаграмм – визуализация особенностей реализации операций классов, когда необходимо представить алгоритмы их выполнения. Диаграмма деятельности представлена на листе 3 графической части.

Временной аспект поведения имеет существенное значение при моделировании синхронных процессов, описывающих взаимодействия объектов. Именно для этой цели и используются диаграммы последовательности, в которых ключевым моментом является динамика взаимодействия объектов во времени. Например, администратор переходит с вкладки на вкладку и добавляет данные, что является временным аспектом поведения администратора. Он включает программу, вводит данные, сохраняет данные – последовательность действий. Диаграмма последовательности представлена на листе 4 графической части.

Диаграмма компонентов описывает объекты реального мира – компоненты программного обеспечения. Эта диаграмма позволяет определить архитектуру разрабатываемой системы, установив зависимости между программными компонентами, в роли которых может выступать исходный, бинарный и исполняемый код.

Определим компоненты для данного программного средства и построим диаграмму компонентов. Главным исполняемым файлом является файл «Страховое дело.exe». Этот файл является результатом компиляции проекта «Страховое дело.sln». Компонент «Страховое дело.sln» реализован в среде разработки Visual Studio 2019. В состав компонента «Страховое дело.sln» входят следующие файлы:

* файлы проекта с расширениями \*.sln, \*.exe;
* файл базы данных \*.sql, созданный в phpMyAdmin.

Вид диаграммы компонентов для данной проектируемой системы представлен в графической части на листе 5.

1. Вычислительная система
   1. Требования к аппаратным и операционным ресурсам

Минимальными требованиями к аппаратному и программному обеспечению, необходимыми для корректной работы программы являются:

* операционная система Windows 10;
* процессор Intel core i5-8250u;
* оперативная память 8 Гбайт;
* мышь;
* монитор;
* клавиатура.

Наиболее удобной операционной системой для проведения испытаний является Windows 10, так как она ориентирована на максимальное использование всех возможностей персонального компьютера, сетевых ресурсов и обеспечение комфортных условий работы.

На компьютере должны быть установлены программы phpMyAdmin, Visual Studio 2019.

* 1. Инструменты разработки

Ниже описаны инструменты разработки, которые будут использоваться для написания программного средства.

Персональный компьютер со следующей аппаратной конфигурацией:

* процессор Intel Core i7 7600U 2800 МГц;
* оперативная память DDR4 8 Гбайт;
* встроенная видеокарта Intel HD Graphics 620;
* винчестер SSD 512 Гбайт;
* клавиатура и мышь.

Программное обеспечение:

* операционная система Windows 10;
* среда программирования Visual Studio 2019;
* язык программирования C#;
* система управления базами данных MySQL;
* веб-приложение phpMyAdmin;
* программная платформа Microsoft .NET Framework 4.7.2;
* офисный пакет приложений Microsoft Word (MS Word) 2016;
* программное средство для создания UML диаграмм Microsoft Visio Pro 2016.

Операционная система – это набор управляющих программ, предназначенных для управления ресурсами вычислительной системы как единого комплекса, другими словами, операционная система – это набор программного обеспечения, который обеспечивает работу компьютера.

При разработке программного средства использовалась операционная система Windows 10, так как на данный момент эта операционная система является самой распространённой операционной системой. В Windows 10 были исправлены практически все недостатки предыдущих операционных систем. Аппаратные требования Windows 10 скромнее, она способна работать даже на маломощных компьютерах и ещё добавлено множество функций, существенно облегчающих работу за компьютером.

Visual Studio 2019 – это лучший инструмент для разработки любого приложения под любую платформу. Система управления версиями в этом выпуске делает разработку гибкой, а совместную работу – эффективной [13].

Язык программирования C# – современный объектно-ориентированный и типобезопасный язык программирования. C# позволяет разработчикам создавать множество типов безопасных и надежных приложений, работающих в экосистеме .NET. В C# особое внимание уделяется управлению версиями для обеспечения совместимости программ и библиотек при их изменении [12].

Система управления базами данных MySQL –  [свободная](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D0%BE%D0%B5_%D0%9F%D0%9E) [реляционная система управления базами данных](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94). Разработку и поддержку MySQL осуществляет корпорация [Oracle](https://ru.wikipedia.org/wiki/Oracle), получившая права на торговую марку вместе с поглощённой [Sun Microsystems](https://ru.wikipedia.org/wiki/Sun_Microsystems), которая ранее приобрела шведскую компанию [MySQL AB](https://ru.wikipedia.org/wiki/MySQL_AB). Продукт распространяется как под [GNU General Public License](https://ru.wikipedia.org/wiki/GNU_General_Public_License), так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм [репликации](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BF%D0%BB%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F_(%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%82%D0%B5%D1%85%D0%BD%D0%B8%D0%BA%D0%B0)) [9].

.NET Framework – [программная платформа](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B0#%D0%BA%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), выпущенная компанией [Microsoft](https://ru.wikipedia.org/wiki/Microsoft) в [2002 году](https://ru.wikipedia.org/wiki/2002_%D0%B3%D0%BE%D0%B4). Основой платформы является общеязыковая среда исполнения [Common Language Runtime (CLR)](https://ru.wikipedia.org/wiki/Common_Language_Runtime), которая подходит для разных языков программирования. Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду [6].

Microsoft Word 2019 — [текстовый процессор](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80), предназначенный для создания, просмотра, редактирования и форматирования текстов статей, деловых бумаг, а также иных [документов](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D1%8B%D0%B9_%D1%84%D0%B0%D0%B9%D0%BB), с локальным применением простейших форм [таблично](https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D0%B0)-[матричных](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%82%D1%80%D0%B8%D1%86%D0%B0_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)) алгоритмов. Выпускается [корпорацией Microsoft](https://ru.wikipedia.org/wiki/Microsoft) в составе [пакета](https://ru.wikipedia.org/wiki/%D0%9E%D1%84%D0%B8%D1%81%D0%BD%D1%8B%D0%B9_%D0%BF%D0%B0%D0%BA%D0%B5%D1%82) [Microsoft Office](https://ru.wikipedia.org/wiki/Microsoft_Office). Первая версия была написана [Ричардом Броди](https://ru.wikipedia.org/w/index.php?title=%D0%91%D1%80%D0%BE%D0%B4%D0%B8,_%D0%A0%D0%B8%D1%87%D0%B0%D1%80%D0%B4&action=edit&redlink=1) ([Richard Brodie](https://en.wikipedia.org/wiki/Richard_Brodie)) для [IBM PC](https://ru.wikipedia.org/wiki/IBM_PC), использующих [DOS](https://ru.wikipedia.org/wiki/DOS), в 1983 году. Позднее выпускались версии для [Apple Macintosh](https://ru.wikipedia.org/wiki/Macintosh) (1984), [SCO UNIX](https://ru.wikipedia.org/wiki/SCO_UNIX) и [Microsoft Windows](https://ru.wikipedia.org/wiki/Microsoft_Windows) (1989). Текущей версией является Microsoft Office Word 2021 для [Windows](https://ru.wikipedia.org/wiki/Windows) и [macOS](https://ru.wikipedia.org/wiki/MacOS), а также веб-версия Word Online ([Office Online](https://ru.wikipedia.org/wiki/Office_Online)), не требующая установки программы на компьютер [8].

1. Проектирование задачи
   1. Требования к приложению

Разрабатываемое приложение должно иметь понятный и удобный в использовании интерфейс, чтобы взаимодействие между программой и пользователем было максимально упрощено.

Согласно общим требованиям, графический интерфейс разрабатываемого приложения должен:

* ориентироваться на пользователя, который общается с программой на внешнем уровне взаимодействия;
* сохранять стандартизированное назначение и местоположение на экране графических объектов, работающих в среде Windows.

Интерфейс программного приложения будет разрабатываться с учетом общих требований к пользовательскому интерфейсу.

Существуют общие требования, которые предъявляются к программным приложениям:

* соответствие стандартам организации интерфейса: использование многооконного подхода, реализация управления работой программного приложения с помощью элементов управления;
* требования к выбранной цветовой схеме рабочего экрана;
* выполнение одной функции с помощью разных элементов управления.

Пользователю необходимо установить пакеты phpMyAdmin для работы с базами данных.

Для обучения пользователя необходимо разработать справочную систему, в которой должны быть раскрыты все аспекты работы с программой, возможные трудности, возникшие во время работы и пути их решения.

Для удобства работы пользователя с программным средством необходимо при разработке окон придерживаться единого стиля оформления. Окна не должны быть перегружены излишней информацией или содержать информацию, не относящуюся к данному окну. Также необходимо предусмотреть защиту данных от удаления и изменения, а также от ввода некорректных данных. В случае ввода некорректных данных или попытке совершить запрещенные действия, пользователь должен быть проинформирован о своих действиях с помощью диалоговых окон.

* 1. Концептуальный прототип

Концептуальный прототип состоит из описания внешнего пользовательского интерфейса, а именно, элементов управления.

При создании данного приложения важную роль играют окна, так как они являются основным средством работы с пользователем. Разрабатываемое приложение будет содержать несколько окон.

Все окна будут содержать стандартные пользовательские элементы управления.

В рабочем режиме программы, пользователю, для удобной навигации, будет предоставлено меню.

Главное окно будет использоваться как главное окно программного средства, на которое пользователь попадает в самом начале. Макет главного окна представлен на рисунке 2.

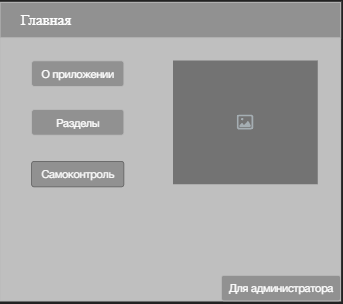


Рисунок 2

После пользователь может перейти в окно «О приложении», с последующей информацией для него. Макет данного окна представлен на рисунке 3.

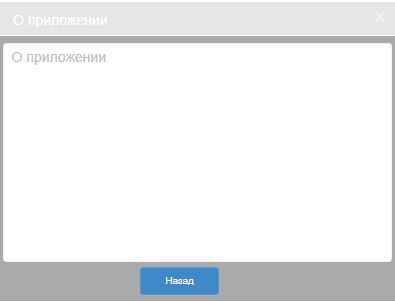


Рисунок 3

Так же пользователь, находясь в главном окне, может перейти в раздел «Лекции» и посмотреть нужную ему информацию, нажав на кнопку в меню «Лекции». Макет данного окна представлен на рисунке 4.

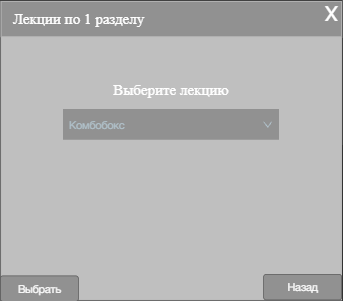


Рисунок 4

Так же, вновь воспользовавшись меню, пользователь сможет перейти в окно «Самоконтроль» и пройти тестирование, получив результат. Макет данного окна представлен на рисунке 5.

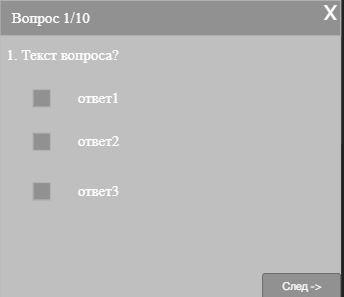


Рисунок 5

На окнах будут располагаться различные элементы управления, предназначенные для определенных задач. За компонентом Button будут закреплены разные действия для выполнения определенных функций, таких как: добавление данных и переходы между окнами. Элементы управления Textbox и Label будут использованы для ввода различных данных. Элементы Datagrid и ComboBox будут выводить данные.

* 1. Организация данных

Организация данных подразумевает создание модели данных, главными элементами которой являются сущности и их связи.

Реляционная модель основана на математическом понятии отношения, представлением которого является таблица. В реляционной модели отношения используются для хранения информации об объектах, представленных в базе данных. Отношение имеет вид двухмерной таблицы, в которой строки соответствуют записям, а столбцы – атрибутам. Каждая запись должна однозначно характеризоваться в таблице. Для этого используют первичные и внешние ключи. Достоинством реляционной модели является простота и удобство физической реализации. Структура базы данных разрабатываемого программного средства включает пять таблиц. Описание таблиц приводится в таблицах 1-5.

Таблица «Разделы» хранит информацию о разделах, структура которой приведена в таблице 1.

Таблица 1 – Структура таблицы «Разделы»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Размер,  байт | Описание |
| id | INT | 4 | Уникальный номер раздела |
| Название | VARCHAR | 254 | Название раздела |

Таблица «Темы» хранит информацию о темах, структура которой приведена в таблице 2.

Таблица 2 – Структура таблицы «Темы»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Размер,  байт | Описание |
| id | INT | 4 | Уникальный номер темы |
| Название | TEXT | 300 | Название темы |
| Раздел | INT | 4 | Номер раздела |

Таблица «Лекции» хранит информацию о лекциях, структура которой приведена в таблице 3.

Таблица 3 – Структура таблицы «Лекции»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Размер,  байт | Описание |
| id | INT | 4 | Уникальный номер лекции |
| Содержание | TEXT | 3000 | Содержание лекции |
| Раздел | INT | 4 | Номер раздела |

Таблица «Вопросы» хранит информацию о вопросах теста, структура которой приведена в таблице 4.

Таблица 4 – Структура таблицы «Вопросы»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Размер,  байт | Описание |
| id | INT | 4 | Уникальный номер теста |
| Содержание | TEXT | 300 | Название теста |
| Ответ1 | VARCHAR | 300 | Первый вариант ответа |
| Ответ2 | VARCHAR | 300 | Второй вариант ответа |
| Ответ3 | VARCHAR | 300 | Третий вариант ответа |
| Ответ4 | VARCHAR | 300 | Четвертый вариант ответа |
| Ответ5 | VARCHAR | 300 | Пятый вариант ответа |
| Ответ6 | VARCHAR | 300 | Шестой вариант ответа |
| Ответ7 | VARCHAR | 300 | Седьмой вариант ответа |
| Ответ8 | VARCHAR | 300 | Восьмой вариант ответа |

Таблица «Админ» хранит пароля для входа администратора, структура которой приведена в таблице 5.

Таблица 5 – Структура таблицы «Админ»

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Размер,  байт | Описание |
| id | INT | 4 | Уникальный номер пароля |
| пароль | VARCHAR | 20 | Пароль администратора |

Cхема базы данных представлена на рисунке 6.

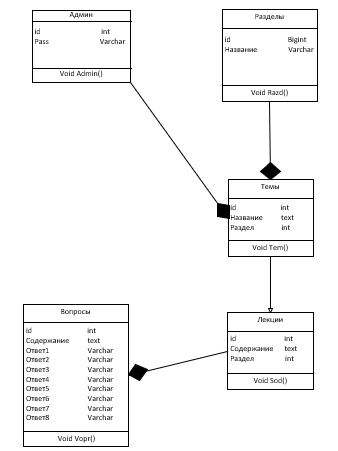


Рисунок 6 – Физическая схема базы данных

* 1. Функции и элементы управления

На основании диаграммы вариантов использования в программном средстве «Страховое дело.exe» реализованы различные функции.

В окне «Лекции» имеется возможность перейти к содержанию лекции (для этого необходимо выбрать нужную лекцию). За данную возможность отвечает Combobox. Код реализации данной функции представлен ниже.

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

{

if (comboBox1.SelectedIndex == 0) //первый элемент списка

{

L1 l = new L1();

l.Show();

this.Hide();

}

if (comboBox1.SelectedIndex == 1) //2 элемент списка

{

L2 l = new L2();

l.Show();

this.Hide();

}

if (comboBox1.SelectedIndex == 2) //3 элемент списка

{

L3 l = new L3();

l.Show();

this.Hide();

}

if (comboBox1.SelectedIndex == 3) //4 элемент списка

{

L4 l = new L4();

l.Show();

this.Hide();

}

if (comboBox1.SelectedIndex == 4) //5 элемент списка

{

L5 l = new L5();

l.Show();

this.Hide();

}

if (comboBox1.SelectedIndex == 5) //6 элемент списка

{

L6 l = new L6();

l.Show();

this.Hide();

}}

В окне администратора при добавлении лекций есть проверка на существование такой лекции в базе данных. Код реализации данной функции представлен ниже.

public Boolean check() // проверка есть ли такая тема в бд

{ DataBase db = new DataBase();

DataTable table = new DataTable();

MySqlDataAdapter adapter = new MySqlDataAdapter();

MySqlCommand command = new MySqlCommand("SELECT \* FROM `темы` WHERE `Название` = @tem AND `Раздел` = @razd", db.getConnection());

command.Parameters.Add("@tem", MySqlDbType.Text).Value = textBox1.Text;

command.Parameters.Add("@razd", MySqlDbType.VarChar).Value = textBox3.Text;

adapter.SelectCommand = command;

adapter.Fill(table);

if (table.Rows.Count > 0)

{

MessageBox.Show("Такая тема уже есть");

return true;

}

else

return false;}

Полный текст программы представлен в приложении А.

* 1. Проектирование справочной системы приложения

Для эффективной работы пользователя с приложением необходимо обеспечить его качественной справочной системой, в которой должны быть приведены методы и приемы работы с приложением, включающие данные о том, что произойдет после нажатия на определенную кнопку или при выборе пункта меню, сведения о том, какую информацию и в каком виде следует вводить в соответствующие поля, каким образом можно работать.

1. Описание программного средства
   1. Общие сведения

Электронное средство обучения «Страховое дело» разрабатывалось с целью автоматизации учебного процесса и улучшения эффективности работы учреждения образования за счет уменьшения времени, затрачиваемого на оформление преподавателем лекций и тестов для проверки знаний благодаря простому, лаконичному интерфейсу и функциональным возможностям приложения. Установка данного программного средства позволяет проделывать различные операции без ошибок. Данная программа была разработана на персональном компьютере со следующей конфигурацией:

* процессор Intel Core i5-7600U 3,9 Ггц;
* оперативная память DDR4 8 ГБ;
* встроенная видеокарта Intel HD Graphics.

Программа создана в средстве разработки Microsoft Visual Studio 2019 на языке программирования C# в операционной системе Windows 10.

В программе присутствует база данных «эсо.sql». Программа не требовательна к системным ресурсам, также проста в использовании и не требует специальных навыков при работе. Для работы данного программного средства необходима предварительная установка и настройка следующих программных продуктов:

* интегрированная среда phpMyAdmin;
* среда разработки Microsoft Visual Studio 2019;
* платформа Microsoft Net Framework 4.7.2.

В состав программных файлов входят:

* программа «Страховое дело.exe»;
* база данных «эсо.sql».

Объем установленного приложения не превышает 20 МБ, файлов – 10 МБ.

Программное средство не требует логина и пароля.

* 1. Функциональное назначение

Программное средство разработано с целью упрощения изучения учебных материалов по учебной дисциплине «Страховое дело».

Программное средство позволяет добавлять и удалять лекции, проходить тестирование с получением результатов.

Программа использует стандартные элементы управления, такие как кнопки, меню, списка, поля ввода, что обеспечивает единство интерфейса системы и программного средства.

В работе программного средства предусмотрены некоторые ситуации, которые должны предупреждать пользователя, чтобы он выполнял все необходимые требования по эксплуатации программы. Для этого существуют сообщения системы, например, если пользователь повторно пытается добавить уже существующую лекцию, или другие данные, которые не должны повторяться, то в этом случае на экран выводится сообщение с информацией о характерной ошибке.

Таким образом, программа может применяться в реальных условиях, представляя собой достаточно удобный помощник.

* 1. Входные и выходные данные

Входными являются данные, введённые пользователем в таблицы базы данных: «Лекции», «Самоконтроль», «Разделы», «Темы».

Входными данным при добавлении данных в таблицу «Лекции»:

* содержание;
* раздел.

Входными данным при добавлении данных в таблицу «Вопросы»:

* содержание;
* ответ1;
* ответ2;
* ответ3;
* ответ4;
* ответ5;
* ответ6;
* ответ7;
* ответ8.

Вся вносимая в программное средство информация хранится в локальной базе данных.

Выходными данными являются:

* лекции;
* разделы;
* темы.

1. Методика испытаний
   1. Технические требования

Основными минимальными требованиями, выдвигаемыми к аппаратному обеспечению персонального компьютера, являются:

* процессор Intel Core i7 7600U 2800 МГц;
* оперативная память DDR4 8 Гбайт;
* встроенная видеокарта Intel HD Graphics 620;
* винчестер SSD 512 Гбайт;
* клавиатура и мышь.

Для работы с программой необходимо наличие монитора, клавиатуры и мыши.

Компьютер должен работать под управлением операционной системы, начиная с Windows 10. Необходимо наличие phpMyAdmin и Microsoft .NET Framework 4.7.2.

* 1. Функциональное тестирование

Функциональное тестирование – это тестирование функций приложения на соответствие требованиям. Оценка производится в соответствии с ожидаемыми и полученными результатами (на основании функциональной спецификации), при условии, что функции отрабатывали на различных значениях.

Тестирование программы будет производиться последовательно, переходя из одной части программы в другую. Во время теста будут проверяться все действия с программой, навигация пунктам меню, которые может произвести пользователь. После чего, все собранные и найденные ошибки будут исправлены. Тест-кейсы для проведения функционального тестирования представлены в таблицах 6 – 8.

Таблица 6 – Тест-кейс для переключения в меню «Самоконтроль»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| Test-1 | Позитивный | Переход в меню «Самоконтроль» | 1. Перейти на главную страницу приложения.  2. Выбрать пункт в меню «Самоконтроль». | Ожидаемый результат: пользователь переходит в нужное окно «Самоконтроль».  Фактический результат:  результат соответствует ожидаемому. Все условия соблюдены. Результат перехода в другое окно представлен на рисунке 7. |

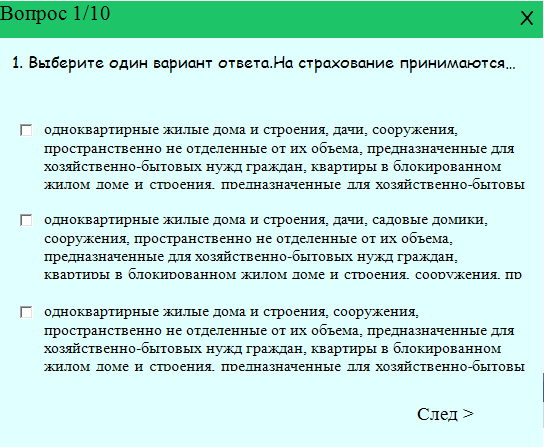


Рисунок 7 – Окно самоконтроля

Таблица 7 – Тест-кейс для добавления лекции

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| Test-2 | Позитивный | Добавление лекции | 1. Перейти на главную страницу приложения.  2. Выбрать пункт в меню «Для администратора».  3. Ввести пароль и нажать на кнопку «Вход».  4. В окне редактирования радела нажать на кнопку «Добавить».  5. Ввести название лекции и ее содержание в предложенные поля и нажать на кнопку «Добавить». | Ожидаемый результат: Данные добавляются в базу данных. Пользователь видит добавленную строку.  Фактический результат:  результат соответствует ожидаемому. Все условия соблюдены. Результат добавления информации представлен на рисунке 8. |

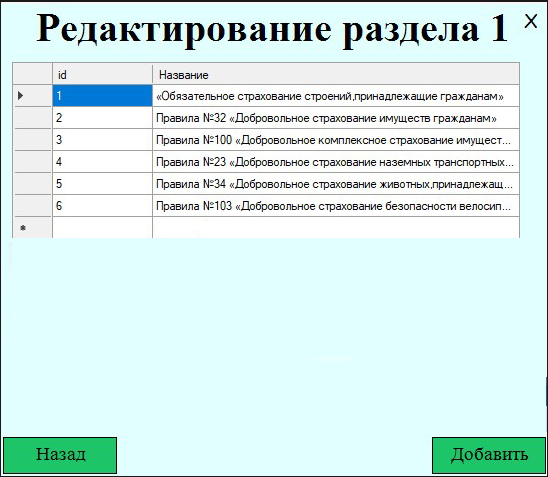


Рисунок 8 – Добавление данных

Таблица 8 – Тест-кейс для получения результата тестирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тип теста | Описание | Шаги воспроизведения | Результат |
| Test-3 | Позитивный | Получение результата | 1. Перейти на главную страницу приложения.  2. Выбрать пункт в меню «Самоконтроль».  3. Выбрать нужный раздел и нажать на кнопку «Выбрать».  4. Выбрать нужную тему и нажать на кнопку «Выбрать».  5. Пройти тестирование и нажать на кнопку «Получить результат».  6. Получить результат. | Ожидаемый результат: Баллы тестирования подсчитываются и пользователь получает результат.  Фактический результат:  результат соответствует ожидаемому. Все условия соблюдены. Результат получения результата представлен на рисунке 9. |

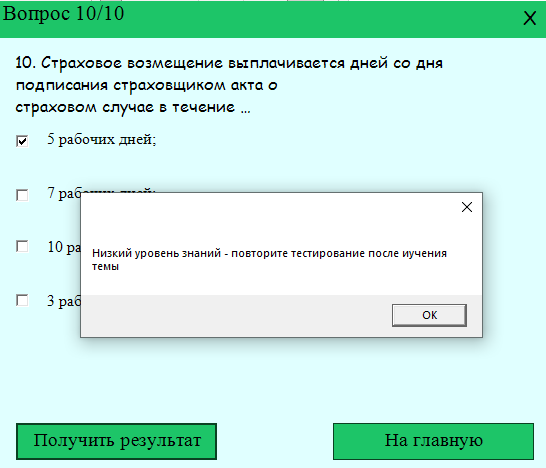


Рисунок 9 – Получение результата тестирования

1. Применение
   1. Назначение программы

Приложение «Страховое дело.exe» предназначено для автоматизации процесса обучения по данной специальности в государственном учреждении образования «Республиканский Институт Профессионального образования». Программное средство предоставляет пользователю возможности для просмотра лекций по разделам и темам, прохождения тестирования, а также добавления и удаления лекций.

* 1. Условия применения

Для работы с программой необходимо наличие программного обеспечения:

* операционная система, начиная с Windows 10;
* система управления базами данных phpMyAdmin;
* библиотека Microsoft .NET Framework 4.7.2.

На случай редактирования проекта программы необходимо наличие программного обеспечения:

* интегрированная среда phpMyAdmin;
* среда разработки приложений Microsoft Visual Studio 2019.
  1. Справочная система

Справочная система представлена в виде встроенного окна в приложении, содержащая информацию по эксплуатации программного средства.

Структура страницы справочной системы представлена на рисунке 10.

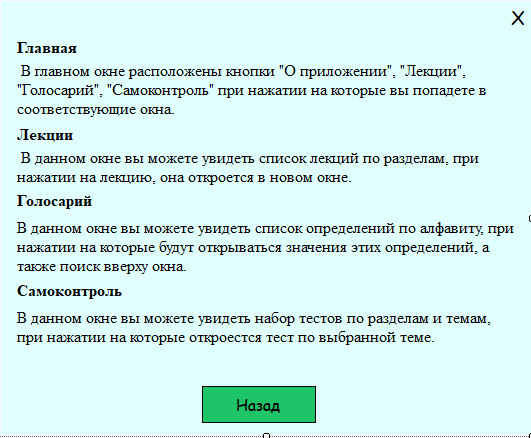


Рисунок 10

Справочная система будет вызываться при нажатии на кнопку «О приложении» на главной форме.

7 Охрана труда и окружающей среды

7.1 Правовые, нормативные, социально-экономические и организационные вопросы охраны труда

В «РИПО» работает в соответствии основными законами, регулирующими правоотношение в области охраны труда, такими как Закон РБ «Об охране труда».

Основными задачами службы охраны труда являются:

– подготовка, планирование и организация улучшающих мероприятий;

– санитарные нормы и правила «Требования при работе с видео дисплейными терминалами и электронно-вычислительными машинами», Гигиенический норматив «Предельно допустимые уровни нормируемых параметров при работе с видео дисплейными терминалами и электронно-вычислительными машинами», утвержденные постановлением Минздрава Республики Беларусь;

– организация работы по обеспечению выполнения работниками требований охраны труда, в том числе ее руководителя, по вопросам охраны труда;

– осуществление контроля за соблюдением требований актов законодательства, технических нормативных правовых актов и локальных нормативных правовых актов, содержащих требования охраны труда.

Главный инженер по техническому обеспечению в РИПО организует выполнение мероприятий по устранению недостатков и нарушений по охране труда, выявленных в ходе проведения ежемесячного контроля. О нарушениях, которые не могут быть устранены работниками структурного подразделения, сообщается ректору.

Организация, оперативное руководство и координация работы по охране труда, общее руководство управлением охраной труда осуществляется ректором Республиканского института профессионального образования.

Ответственным за осуществления контроля по охране труда – главный инженер.

Инженер по охране труда осуществляет организационно-методическую работу по управлению охраной труда, подготовку управленческих решений, контроль за их реализацией под руководством главного инженера.

Функции инженера по охране труда:

– проводит анализ состояния и причин производственного травматизма и профессиональной заболеваемости, разрабатывает мероприятия по их предупреждению;

– оказывает методическую помощь подразделениям в разработке инструкций по охране труда;

– проводит вводный инструктаж по охране труда, организует пропаганду охраны труда на предприятии, участвует в расследовании несчастных случаев на производстве и профессиональных заболеваний;

– оказывает помощь в организации обучения работников по вопросам охраны труда;

– подготовка государственной статистической отчетности по охране труда;

– пропаганда и распространение передового опыта в области охраны труда в организациях, расположенных на подведомственной территории, в том числе через средства массовой информации, организацию выставок, смотров-конкурсов, семинаров, совещаний и других мероприятий;

– консультирование, подготовка ответов на запросы организаций, рассмотрение в установленном порядке писем, обращений граждан по вопросам охраны труда;

– анализ причин производственного травматизма и профессиональной заболеваемости в организациях, расположенных на подведомственной территории, разработка мер по их профилактике;

– подготовка государственной статистической отчетности по охране труда;

– участие в установленном законодательством порядке в расследовании несчастных случаев на производстве;

– информационное обеспечение организаций, расположенных на подведомственной территории, по вопросам охраны труда.

В соответствии с требованиями ст.17 Закона от 23.06.2008 №356-З «Об охране труда» и ст.226 Трудового кодекса работодатель обязан осуществлять обучение, стажировку, инструктаж и проверку знаний, работающих по вопросам охраны труда.

Обучение по охране труда работников предприятия производится в соответствии с действующими Государственными стандартами, межотраслевыми положениями, типовыми инструкциями и программами.

Обучение руководителей и специалистов осуществляется по учебным планам и программам, составленным на основании типового перечня вопросов для обучения и проверки знаний по вопросам охраны труда руководителей и специалистов в соответствующих учреждениях.

Руководитель организации утверждает перечень должностей специалистов, которые должны проводить стажировку и определяет ее продолжительность.

Не позднее месяца со дня назначения на должность и периодически в соответствии с требованиями нормативных правовых актов, но не реже одного раза в три года, руководители и специалисты проходят проверку знаний по вопросам охраны труда и комиссиях для проверки знаний по вопросам охраны труда.

Ответственность за организацию своевременного и качественного проведения обучения, инструктажа, стажировки и проверки знаний работников по вопросам охраны труда в институте возлагается на главного инженера.

Расследование и учет несчастных случаев, профессиональных заболеваний в РИПО осуществляется в соответствии с «Правилами расследования и учета несчастных случаев на производстве и профессиональных заболеваний», утвержденными Постановлением Совета Министров Республики Беларусь от 15.01.2004 г. №30.

Расследование несчастного случая, профессионального заболевания, на производстве проводится с целью выявления вызвавших их причин и принятия соответствующих мер для предотвращения повторения подобных ситуаций, а именно:

– установления причин несчастного случая и определение мер по экстренному устранению причин травм;

– установления должностных и иных лиц, виновных в нарушениях, которые привели к несчастному случаю, какие требования норм, правил были нарушены;

– привлечения к ответственности виновных лиц на основании законодательства и иных нормативных правовых актов.

Основными видами контроля на предприятии являются:

– оперативный контроль руководителя работ и других должностных лиц;

– контроль по охране труда, осуществляемый службой охраны труда предприятия в соответствии с Типовым положением о службе охраны труда организации;

– периодический контроль за соблюдением законодательства об охране труда, осуществляемый представителями нанимателя с участием общественных инспекторов профсоюзов по охране труда.

В РИПО периодический контроль осуществляется представителями нанимателя с участием общественных инспекторов профсоюзов по охране труда, проводится:

– ежедневно;

– ежемесячно;

– ежеквартально.

Ежедневно контроль за состоянием охраны труда осуществляется руководителем структурного подразделения с участием общественного инспектора профсоюза по охране труда в отделе материально технического обеспечения, участке полиграфических работ и в производственном отделе.

Ежемесячно контроль проводится главным инженером предприятия с участием общественного инспектора профсоюза по охране труда, ведущим инженером по охране труда.

Ежеквартально контроль осуществляется на предприятии в целом ректором предприятия с участием руководителей структурных подразделений, общественных инспекторов по охране труда.

## 7.2 Разработка мер по нормализации вентиляции в помещении

Главенствующая роль в нормализации воздушной среды на рабочих местах и в помещениях принадлежит вентиляции. Вентиляция – это совокупность мероприятий и устройств, необходимых для обеспечения заданного качества воздушной среды в рабочих помещениях.

По способу перемещения воздуха вентиляция подразделяется на естественную и механическую, а по способу организации воздухообмена бывает местная и общеобменная.

При естественной вентиляции воздухообмен происходит за счет разности температур, т.е. под влиянием теплового напора. Естественная вентиляция производственных помещений может быть неорганизованной и организованной. При неорганизованной вентиляции (проветривании) поступление и удаление воздуха происходит через окна, форточки, фрамуги, специальные приемы, а также через неплотности наружных ограждений (инфильтрация). Организованная (регулируемая) естественная вентиляция производственных помещений называется аэрацией фонарей. При отсутствии аэрационных фонарей естественная вентиляция может быть улучшена с помощью специальных каналов или шахт. С целью повышения эффективности ветрового напора эти шахты снабжаются специальными насадками – дефлекторами.

В отличии от естественной механическая вентиляция позволяет производить предварительную обработку приточного воздуха – увлажнение, нагрев или охлаждение и очистку от пыли, газов, аэрозолей и других примесей. К установкам местной механической вентиляции относятся местные отсосы открытого типа, включающие защитно-обеспылевающие кожухи, вытяжные шкафы, бортовые отсосы, шарнирнотелескопические отсосы, перемещаемые отсосы, а также вытяжные зонты, укрытия-боксы, камеры и кабины.

Общеобменная вентиляция применяется в тех случаях, когда вредные вещества, избыточное тепло и влага выделяются рассредоточено по всему рабочему помещению, и удалить их с помощью местных отсосов не предоставляется возможным. Принцип действия общеобменной вентиляции основан на разбавлении загрязненного, перегретого или переувлажненного воздуха до уровней, соответствующих гигиеническим нормативам.

Приточная вентиляция предназначена для обработки воздуха: его подогрев, охлаждение, очистка от пыли или увлажнение. Рециркуляция воздуха в системах приточно-вытяжной вентиляции применяется в холодное и переходное время года в целях экономии тепловой энергии, затрачиваемой на подогрев воздуха.

Кондиционирование воздуха – это процесс создания и автоматического регулирования в помещениях заданных параметров микроклимата.

Основные санитарно-гигиенические требования к вентиляции производственных помещений определены гигиеническими нормативами, а также строительными нормами и правилами. Для эффективной работы вентиляции важно, чтобы еще на стадии ее проектирования было предусмотрено выполнение ряда санитарно-гигиенических и технических требований. Количество воздуха, необходимого для вентиляции производственных помещений и обеспечения требуемых параметров воздушной среды в рабочей зоне, устанавливается расчетным способом. Расчет ведется по избытку тепла, влаги или по количеству выделяющихся вредных веществ (пыли, газов, паров). При одновременном выделении в помещении тепла, ваги и вредных веществ необходимый воздухообмен должен устанавливаться по превалирующей вредности.

Количество наружного воздуха, подаваемого в помещения на одного работника, по санитарным нормам должно составлять:

* больше 30 м3/ч при объеме помещения не менее 20 м3 на одного работника;
* больше 20 м3/ч при объеме помещения более 20 м3 на одного работника.

В помещениях объемом более 40 м3 на каждого работающего при наличии окон (или окон и аэрационных фонарей) и при отсутствии выделения вредных или неприятно пахнущих веществ допускается периодически устраивать проветривание. В помещениях без естественной вентиляции подача воздуха на одного человека должна составлять не менее 60 м3/ч. Система вентиляции не должна быть источником шума и загрязнения окружающей среды. В процессе эксплуатации вентиляционные системы должны обслуживаться, очищаться от загрязнений, ремонтироваться в соответствии с установленным графиком подготовленным персоналом.

## 7.3 Пожарная безопасность

По взрывопожарной и пожарной опасности помещения и здания Республиканского Института Профессионального образования относятся по ТКП 474-2013 к категории B1-B4, так как в помещениях обращаются горючие вещества и материалы, способные при взаимодействии с кислородом воспламенятся.

По классификации пожароопасных зон ПУЭ, организация относится к зоне П-IIa, так как помещения не имеют оборудования, реагирующего на нормальные условия.

Строительные конструкции здания имеют следующий предел огнестойкости и класс пожарной опасности:

– несущие элементы здания (R45-K1) – предел огнестойкости 45 минут – по потере несущей способности, мало пожароопасные;

– самонесущие стены (RE 30-K1) – предел огнестойкости 30 минут – по потере несущей способности и потере целостности независимом от того, какое из двух предельных состояний наступит ранее, мало пожароопасные;

– наружные несущие стены (E 15-K2) – предел огнестойкости 15 минут – по потере целостности, умеренно пожароопасные;

– перекрытия междуэтажные (RE1 45-K1) – предел огнестойкости 45 минут – по потере несущей способности, целостности и теплоизолирующей способности независимо от того, какое из трех предельных состояний наступит ранее, мало пожароопасные;

– лестничные клетки – внутренние стены (REI 60-KO) – предел огнестойкости 60 минут – по потере несущей способности, целостности и теплоизолирующей способности независимо от того, какое их тех предельных состояний наступит ранее, не пожароопасные;

– лестничные клетки – марши и площадки лестниц (R 45-KO) – предел огнестойкости 45 минут – по потере несущей способности, не пожароопасные.

Согласно требованиям пожарной безопасности, предприятие обеспеченна первичными средствами пожаротушения:

– огнетушители (водные (ОВ), углекислотные (ОУ-2));

– пожарные краны (ПК).

Первичные средства пожаротушения расположены в коридорах, проходах, не препятствуют безопасной эвакуации людей. Их располагают на видных местах вблизи от выходов из помещений на высоте не более 1,5 м.

Расстояние от возможного очага возгорания до места размещения огнетушителя не должно превышать 20 метров.

В каждом кабинете имеется один огнетушитель ОУ-1 с диоксидом углерода, расположен рядом с выходной дверью на высоте метра от уровня пола.

Так же во всех помещениях установлены тепловые пожарные извещатели. Извещатель предназначен для обнаружения загораний, сопровождающихся появлением дыма в закрытых помещениях зданий и сооружений, и формирования электрического сигнала о возникшем пожаре и передачи его на приемно-контрольные приборы.

Выход из здания осуществляется поэтапно:

– на первом этапе люди покидают внутренние помещения;

– второй этап — выбор пути движения к выходу.

В высотных зданиях принят следующий порядок действий:

– первым эвакуируется этаж, на котором произошло возгорание, затем выводят людей с этажа выше и ниже, потом с оставшихся этажей, начиная с верхних;

– третий этап относится к этажам, расположенным выше первого;

– четвёртый этап – перемещение людей на безопасное расстояние от горящего здания.

Порядок создания и организации деятельности пожарной команды:

Пожарная команда может создаваться:

– в организации (независимо от формы собственности) – решением этой организации (далее – организация, в которой создана пожарная команда);

– на базе организации (независимо от формы собственности) – решением местного Совета депутатов, исполнительного и распорядительного органа (далее – организация, в которой создана пожарная команда).

Решением, указанным в части первой настоящего пункта, определяются:

– организация, на базе которой создается пожарная команда;

– сроки создания пожарной команды;

– источники финансирования пожарной команды;

– место дислокации пожарной команды;

– разряд пожарной команды;

– иная информация по усмотрению организации или органа, принявших решение о создании пожарной команды.

Пожарная команда может создаваться в структурных подразделениях организации, в которой или на базе которой создана пожарная команда.

Пожарная команда считается созданной с момента включения ее в план привлечения сил и средств на тушение пожаров и ликвидацию других чрезвычайных ситуаций.

Руководство пожарной командой, организация ее деятельности и обеспечение взаимодействия с органами и подразделениями по чрезвычайным ситуациям, добровольными пожарными организациями, иными организациями, обеспечение соблюдения законности в деятельности пожарной команды осуществляются руководителем организации, в которой создана пожарная команда.

Пожарные команды подразделяются на следующие разряды: первый разряд – с круглосуточным пребыванием водителя транспортного средства пожарной команды в месте нахождения пожарной автоцистерны или иной приспособленной для тушения пожаров техники организации, в которой создана пожарная команда, и круглосуточной готовностью других членов пожарной команды прибыть к месту нахождения пожарной автоцистерны или иной приспособленной для тушения пожаров техники организации, в которой создана пожарная команда, или к месту возникновения пожара (к месту сбора); второй разряд – с круглосуточной готовностью водителя транспортного средства пожарной команды прибыть к месту нахождения пожарной автоцистерны или иной приспособленной для тушения пожаров техники организации, в которой создана пожарная команда, а других членов пожарной команды – к месту нахождения пожарной автоцистерны или иной приспособленной для тушения пожаров техники организации, в которой или на базе которой создана пожарная команда, или к месту возникновения пожара; третий разряд – с круглосуточной готовностью членов пожарной команды прибыть к месту нахождения пожарной автоцистерны подразделения по чрезвычайным ситуациям или к месту возникновения пожара.

Требования по обеспечению постоянной готовности пожарной команды к тушению пожаров, в том числе обеспечению безопасности членов пожарной команды и оперативности реагирования пожарной команды, устанавливаются Министерством по чрезвычайным ситуациям. Ответственность за соответствие пожарной команды требованиям, указанным в части первой настоящего пункта, возлагается на руководителя организации, в которой создана пожарная команда, и начальника пожарной команды

Пожарная команда состоит из работников организации, в которой она создана.

Членами пожарной команды могут быть граждане Республики Беларусь, иностранные граждане и лица без гражданства, достигшие восемнадцатилетнего возраста, способные по своим деловым, моральным качествам и состоянию здоровья выполнять обязанности члена пожарной команды.

Начальник пожарной команды определяется из числа ее членов руководителем организации, в которой создана пожарная команда.

Члены пожарной команды не позднее одного месяца со дня принятия организацией, в которой создана пожарная команда, решения о зачислении в пожарную команду проходят подготовку по программам пожарнотехнического минимума в соответствии с законодательством. Подготовка членов пожарной команды также осуществляется при проведении тактико-специальных учений, занятий и объектовых тренировок. В период прохождения подготовки по программам пожарно-технического минимума, а также участия в тактико-специальных учениях, занятиях и объектовых тренировках членам пожарной команды предоставляются гарантии и компенсации, предусмотренные законодательством при повышении квалификации, переподготовке, профессиональной подготовке и стажировке работников по направлению нанимателя.

Пожарная команда может иметь символику, зарегистрированную в порядке, установленном законодательством, а также свое наименование.

Упразднение пожарной команды производится решением организации или органа, принявших решение о создании пожарной команды.

Организация или орган, принявшие решение о создании пожарной команды, не позднее 15 календарных дней до принятия решения об упразднении пожарной команды направляет уведомление в соответствующий городской (районный) отдел по чрезвычайным ситуациям. Пожарная команда считается упраздненной со дня исключения ее из плана привлечения сил и средств на тушение пожаров и ликвидацию других чрезвычайных ситуаций в населенных пунктах.

## 7.4 Охрана окружающей среды

В организации Республиканском Институте Профессионального образования за охрану окружающей среды отвечает инженер по охране окружающей среды, который ведет постоянный контроль деятельности всех служб на соблюдение требований инструкции по охране окружающей среды. Ответственность природопользователей, направленная в первою очередь на рациональное использование природных ресурсов, модернизацию оборудования, а также разработка и выполнение ими мероприятий по охране окружающей среды помогут снизить вредное воздействие производства на окружающую среду и улучшить экологическую обстановку в зоне влияния их хозяйственной деятельности.

Основные направления разработки мероприятий по охране окружающей среды на предприятии:

Охрана атмосферного воздуха, защита озонового слоя:

– переход на виды топлива, сырья, материалы, обеспечивающие предотвращение и (или) снижение выбросов загрязняющих веществ, в т.ч. озоноразрушающих, и (или) парниковых газов в атмосферный воздух, приобретение оборудования для внедрения таких методов;

– внедрение наилучших доступных технических методов;

– перевод мобильных источников выбросов на использование видов топлива с улучшенными экологическими характеристиками;

– улучшение условий рассеивания выбросов, сокращение неорганизованных источников выбросов.

Рациональное использование и охрана водных ресурсов:

– строительство, реконструкция, модернизация, капитальный ремонт сетей и сооружений для сбора, транспортировки, очистки, отведения и выпуска сточных вод, приобретение оборудования для этих целей;

– внедрение наилучших доступных технических методов, обеспечивающих предотвращение и снижение сбросов загрязняющих веществ в составе сточных вод в окружающую среду, а также направленных на сокращение потребления водных ресурсов;

– строительство и обустройство скважин водоснабжения;

– создание автоматизированных систем контроля сбросов загрязняющих веществ в составе сточных вод, приобретение оборудования, приборов, средств, в т.ч. программных.

ликвидация источников загрязнения поверхностных и подземных вод сточными водами животноводческих ферм и комплексов (строительство, ремонт и реконструкция сооружений для сбора, накопления, обработки навоза и навозосодержащих сточных вод, реконструкция оросительных систем по утилизации животноводческих сточных вод и поливочного оборудования, подбуртовка, обваловка, транспортировка навоза и навозосодержащих сточных вод).

Мероприятия, направленные на предотвращение и снижение вредного воздействия отходов:

– внедрение современных технологий по использованию и обезвреживанию отходов, технологических процессов, направленных на уменьшение объемов и (или) предотвращение образования отходов производства;

– приобретение оборудования для использования и обезвреживания отходов;

– приобретение, изготовление контейнеров для сбора отходов и вторичных материальных ресурсов.

Охрана труда в институте может включать такие мероприятия:

– замена старого производственного оборудования на новое;

– контроль соблюдения режима труда и отдыха;

– установка противопожарных систем;

– приобретение строительной техники с более низкими вибрационными и шумовыми характеристиками и т.п.

На основании составленного списка природоохранных мер создается официальный документ – перечень мероприятий по охране окружающей среды. Все мероприятия по сохранению природной среды, которые предприятие отражает в соответствующих документах, обязательны для выполнения.

8 Экономический раздел

# 8.1 Обоснование необходимости выведения на рынок программного продукта

Цель экономического раздела дипломного проекта – рассчитать затраты на разработку программного продукта и определить экономическую эффективность от его внедрения. Также необходимо рассчитать экономическую эффективность разработки программного средства.

Данное приложение сможет помочь ученикам и преподавателям автоматизировать учебный процесс путем имеющихся в приложении тесов и лекций по разделам. Использование данной программы будет ежедневное для людей, которые будут работать и обучаться в данном учебном заведении.

Аналогами данного приложения является «Страхование». Данное программное средство используется учениками и преподавателями РИПО.

В технико-экономическом обосновании будут рассмотрены следующие вопросы:

* составление плана по разработке программного средства;
* расчет стоимости разработки;
* экономическая эффективность разработки.

В первой части технико-экономического обоснования необходимо рассчитать срок разработки по созданию программного средства электронного средства обучения по предмету «Страховое дело» для государственного учреждения образования РИПО.

Во второй части технико-экономического обоснования необходимо рассчитать стоимость разработки программного средства с момента получения первого варианта технического задания и заканчивая оформлением документации и сдачей разработки.

В третьей части необходимо оценить экономическую эффективность программного средства.

# 8.2 Составление плана по разработке программного продукта

План разработки программного продукта представлен в таблице 9.

Таблица 9 - План разработки программного продукта

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование этапов и видов работ | Исполнитель  (должность,  квалификация) | Количество исполнителей | Трудоемкость, человеко-дни, мес |
| Инициация | Техник-программист | 1 | 3 |
| Планирование | Техник-программист | 1 | 2 |
| Разработка | Техник-программист | 1 | 25 |
| Реализация и тестирование | Техник-программист | 1 | 5 |
| Мониторинг и завершение проекта | Техник-программист | 1 | 5 |

Результаты расчетов отображены в гистограмме и представлены на рисунке 11.

Рисунок 11 – Гистограмма распределения времени работы при разработке программного продукта

# 8.3 Расчет затрат на разработку программного продукта

Стоимостная оценка программного продукта и определение экономического эффекта у разработчика предполагает расчет затрат, которые включают следующие элементы:

* материальные затраты;
* затраты на оплату труда;
* отчисления на социальные нужды;
* амортизация основных средств и нематериальных активов;
* прочие затраты.

В материальных затратах*.* отражаются затраты на материалы и принадлежности, необходимые для проведения научно-исследовательских работ, магнитные носители, бумагу, красящие ленты и другие материалы, необходимые для разработки программного продукта. Затраты определяются по действующим отпускным ценам. Здесь так же отражаются затраты на электроэнергию.

Расчет осуществляется по формуле (1)

 (1)

где Ктр - коэффициент, учитывающий транспортно-заготовительные расходы (в работе принимаем от 1,05 до 1,10);

Нрi - норма расхода i-го вида материалов;

Цi- действующая отпускная цена за единицу i-го вида материала, руб.;

n – количество применяемых видов материалов.

Расчет целесообразно представить в табличной форме, представлено в таблице 10.

руб.

Таблица 10 - Расчет материальных затрат

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование материалов | Единица измерения | Коли-чество | Цена приобретения, руб. | Сумма, руб. |
| Флэш-накопитель | Шт. | 1 | 7,50 | 7,50 |
| Картридж | Шт. | 1 | 12,80 | 12,80 |
| Папка скоросшиватель | Шт. | 1 | 0,37 | 0,37 |
| Шариковая ручка | Шт. | 1 | 0,50 | 0,50 |
| Бумага | Пачка | 1 | 11,30 | 11,30 |
| Всего расходов | | | | 32,47 |
| Всего с транспортно-заготовительными расходами | | | | 33,37 |

Затраты на электроэнергию находятся исходя из продолжительности периода разработки программного продукта, количества кВт/ч, затраченных на его проектирование и тарифа за 1 кВт/ч по формуле (2)

(2)

гдеКэ – стоимость одного кВт/ч,руб;

Тр – количество кВт/ч.

Тарифы на электроэнергию применяются, согласно, приложения к Декларации *«*Об уровне тарифов на электроэнергию, отпускаемую РИПО» на соответствующий период времени, когда разрабатывается программный продукт. Время реализации проекта 40 дней, среднее потребление энергии в месяц составляет 75 кВт/ч, то есть необходимое потребление энергии в дипломном проекте принимается 143 кВт/ч.

руб.

Затраты на оплату исчисляются исходя из численности различных категорий исполнителей и трудоемкости выполнения отдельных видов работ, тарифных ставок за один день или месячных должностных окладов, премиальных систем оплаты труда исполнителей по формуле (3)

(3)

где  - тарифная ставка за день (месячный оклад) i-й категорий работников;

tфi - время фактической работы работника i-й категории по теме, дн. или мес.;

Кпр - коэффициент премий по премиальным системам, Кпр от 1,10 до 1,30.

Расчет данных прямых расходов также целесообразно представить в табличной форме, как представлено в таблице 11. Для этого необходимо рассчитать заработную плату исполнителя, исходя из тарифной ставки первого разряда, которая в организации составляет 301,80 руб. Разряд техника программиста 7, тарифный коэффициент-2,03.

руб.

Таблица 11 - Расчет затрат на основную заработную плату персонала

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование категорий работников и должностей | Количество штатных единиц, чел. | Заработная плата за 1 день (месяц), руб. | Трудозатраты, дн. или мес. | Сумма, руб. |
| Техник-программист | 1 |  | 2 | 1225,31 |
| Всего с коэффициентом премий | | | | 1469,31 |

Определяется дополнительнаязаработная плата исполнителей, включающая разнообразные предусмотренные трудовым законодательством выплаты, по формуле (4)

(4)

где - норматив дополнительной заработной платы, от 10 до 20 %.

руб.

В отчислениях на социальные нужды отражается сумма обязательных страховых взносов, взносов на профессиональное пенсионное страхование в бюджет государственного внебюджетного фонда социальной защиты населения Республики Беларусь.

Рассчитываются отчисления органам социальной защиты по формуле (5)

(5)

где - норма отчислений на социальную защиту, =34%.

=599,48 руб.

Также рассчитываются отчисления на страхование от несчастных случаев на производстве и профессиональных заболеваний (Рстр) по ставке действующего законодательства. Для расчетов среднее значение Нбгс принимается равным 0,6 %.

Рстр = (6)

руб.

Амортизация основных средств и нематериальных активов рассчитываются в установленном законодательством Республики Беларусь порядке, по одному из методов начисления амортизации (линейный, нелинейный, производительный), исходя из целесообразности его применения.

По статье «Амортизация основных средств и нематериальных активов» рассчитываются амортизационные отчисления (АО), исходя из стоимости основных средств (ОС), используемых в процессе разработки программного обеспечения, сроков эксплуатации оборудования (Тс) и годовой нормы амортизации (На).

Для определения затрат по данному элементу будет использоваться линейный способ начисления амортизации. Нормативные сроки службы машин и оборудования составляют пять лет.

Норма амортизации для линейного способа начисления вычисляется по формуле (7)

(7)

гдеТс – срок службы оборудования, лет.

Для линейного способа начисления амортизационные отчисления равномерно распределены на весь период службы оборудования и вычисляются на один год. Так как разработка программного продукта длилась 40 дней, то сумма амортизационных отчислений (АО) за этот период составит

. (8)

руб.

Прочие затратына конкретное программное обеспечение включают затраты на: арендную плату; вознаграждения за рационализаторские предложения и выплата авторских гонораров; затраты на гарантийный ремонт и обслуживание изделий;начисленные налоги, сборы (пошлины), платежи, включаемые в затраты на производство продукции (работ, услуг); связь; на оплату услуг рекламы и маркетинга и пр.

В расчетах принимаем условно размер прочих затрат равных от 30 до 50 % от суммы всех остальных затрат на разработку

(9)

≈ от 20 до 30%.

руб.

На основании полученных данных по отдельным статьям затрат рассчитывается общая сумма затрат на разработку программного продукта.

Общая сумма затрат по элементам на разработку программного продукта рассчитывается по формуле(10)

(10)

руб.

Результаты расчетов оформляются таблицей 12.

Таблица 12 - Расчет затрат на разработку программного продукта

|  |  |
| --- | --- |
| Элемент затрат | Затраты, руб. |
| Материальные затраты (Рм) | 33,37 |
| Электроэнергия (Рэ) | 55,76 |
| Затраты на оплату труда (Роз) |  |
| Дополнительная заработная плата (Рдз ) |  |
| Отчисления органам социальной защиты (Рос) | 599,48 |
| Отчисления на страхования от несчастных случаев (Рстр) |  |
| Амортизация (РАо) |  |
| Прочие затраты (Рпр) |  |
| Общая сумма затрат (З) | 1787,25 |

Таблицу 12 удобно представить в виде диаграммы, отображающей элементы затрат программного продукта. Диаграмма структуры затрат представлена на рисунке 12.

Рисунок 12 – Структура затрат

Исходя из расчётов затрат на разработку программного продукта, можно сделать вывод, что самыми затратными являются выплаты зарплаты сотруднику и отчисление органам социальной защиты. Также не мало средств уходит на прочие затраты и выплату дополнительной заработной платы сотруднику. Незначительное количество средств уходит на электроэнергию, материальные расходы, амортизацию и страховку.

# 8.4 Экономический эффект у разработчика программного продукта

Заказчик оплачивает разработчику всю сумму расходов по проекту, включая прибыль. После уплаты налогов из прибыли в распоряжении заказчика остается чистая прибыль от проекта. Ввиду того, что программное обеспечение разрабатывается для одного объекта, чистую прибыль можно считать в качестве экономического эффекта разработчика от реализованного программного продукта.

В дипломном проекте отпускная цена программного продукта, представляет собой не цену за единицу продукции, а цену проекта, за которую его можно продать и получить определённую выгоду.

Отпускная цена продукции формируется исходя из плановой себестоимости производства продукции, всех видов установленных налогов и прибыли, а также качества, потребительских свойств продукции и конъюнктуры рынка.

С учетом действующих в республике нормативных документов отпускная цена на продукцию рассчитывается по формуле (11)

(11)

гдеОЦ– отпускная цена разработчика, руб.;

З – затраты на разработку, руб.;

П – прибыль, руб.

руб.

Прибыль рассчитывается по формуле(12)

, (12)

гдеR –уровень рентабельности (от 10 до 30 %), %

руб.

Стоимость проекта с учётом НДС, представляет собой сумму отпускной цены и налога на добавленную стоимость.

, (13)

где ставка НДС-20%

руб.

(14)

руб.

Таким образом, разработчик программного продукта может продать заказчику программный продукт, что покроет затраты и обеспечит прибыль за разработку проекта.

Сумма расходов на разработку и содержание программного средства представлена в таблице 13.

Таблица 13 – Сумма расходов на разработку и содержание программного средства

|  |  |
| --- | --- |
| Затраты | Величина, руб. |
| Отпускная цена разработчика |  |
| Обслуживание разработчиком | 400 |
| Итого | 2544,70 |

Предполагается, что с внедрением программного средства выручка организации увеличится на 20%, по сравнению с предыдущим годом, за счёт увеличения количества новых клиентов. Выручка в 2021 году составила 600000 рублей, следовательно, дополнительная выручка будет равна 120000 рублей.

= 120000,00 руб.

Теперь найдём дополнительную прибыль по формуле (15)

, (15)

гдеОЦ – отпускная цена разработчика, руб.;

– дополнительная выручка за год, руб.;

З – затраты на содержание сайта за год;

R – уровень рентабельности.

=23571,06 руб.

Дополнительная прибыль составляет 23571,06 руб., что значительно превышает расходы на содержание программного средства. Это является хорошим экономическим показателем и так же является обоснованием создания долгосрочной услуги.

# 8.5 Экономический эффект у пользователя программного продукта

Экономический эффект у пользователя программного продукта выражается в виде экономии трудовых, материальных и финансовых ресурсов, получаемой от:

* повышение производительности сервиса (увеличение числа выводимых за единицу времени документов, уменьшение среднего времени подготовки отчета и т.д.), что выражается в снижении трудоемкости выполнения операций, решении задач, подготовки данных, обработки информации и анализа результатов;
* сокращение затрат на оплату машинного времени и расходных материалов;
* повышения уровня сервиса (сокращение времени на устранение инцидентов);
* улучшения показателей основной деятельности предприятия в результате использования программного продукта и т.д.

Так же можно проводить сравнение затрат на производство по статьям затрат (если учащийся располагают такой информацией по изделию-аналогу). За базовый вариант следует принимать аналогичный программный продукт, используемое в действующей автоматизированной системе или ручной вариант, если автоматизация отсутствует.

При сравнении базового и нового варианта программного продукта в качестве экономического эффекта будет выступать общая сумма экономии всех видов ресурсов относительно базового варианта.

Приведем расчет экономии некоторых видов ресурсов в связи с использованием нового программного обеспечения.

Экономия затрат на заработную плату (Эз) при использовании нового программного продукта в расчете на объем выполненных работ определяется по формуле(16)

(16)

где Эзе – экономия затрат на заработную плату при решении задач с использованием нового программного продукта в расчете на 1 задачу, руб4

Q *-* количество задач, решаемых за год с использованием нового программного продукта (задач). Уточняет руководитель дипломного проекта.

руб.

Экономия затрат на заработную плату в расчете на одну задачу (Эзе)рассчитывается по формуле(17)

(17)

гдеЗсм – среднемесячная заработная плата одного программиста, руб.

Тс1, Тс2 – снижение трудоемкости на одну задачу в базовом и новом варианте соответственно, человеко – часов (соответственно 1,71 и 0,87 чел-час);

Тч -количество часов работы в день, часов (8 часов);

Др – среднемесячное количество рабочих дней.

руб.

Экономия с учетом начислений на заработную плату Эн

(18)

где Кнз – коэффициент начислений на заработную плату принимаем 1.2.

руб.

Экономия за счет сокращения простоя сервиса Эс

(19)

гдеП1,П2 *-* время простоя сервиса, обусловленное программным продуктом в день;

Дрг- плановый фонд работы сервиса (дней);

Сп– стоимость одного часа работника для которого разрабатывается программный продукт, руб.

руб.

Общая экономия от применения нового программного продукта рассчитывается по формуле(20)

, (20)

руб.

Внедрение нового программного продукта позволит сэкономить на текущих затратах, т.е. практически получить на эту сумму дополнительную прибыль. Для пользователя в качестве экономического эффекта выступает лишь чистая прибыль – дополнительная прибыль, остающаяся в его распоряжении, которая определяется по формуле(21)

(21)

руб.

В процессе использования нового программного продукта чистая прибыль в конечном итоге возмещает капитальные затраты. Она составляет 1837,37 рублей.

В данном разделе представлено экономическое обоснование для дипломного проекта по теме реализация программного средства электронное средство обучения по предмету «Страховое дело» для государственного учреждения образования РИПО.

В технико-экономическом обосновании были рассмотрены следующие вопросы:

* составление плана по разработке программы;
* расчет стоимости разработки.

В первой части технико-экономического обоснования был рассчитан срок разработки по созданию программного средства.

Во второй части технико-экономического обоснования была рассчитана стоимость разработки программного продукта, которая составила 2573,64 рублей.

Основное преимущество разработки программного средства в том, что оно сокращает время изучения программы, а также её подачи и проверки знаний учащихся.

Заключение

В процессе разработки программного средства были применены и закреплены знания по уже изученному материалу, были отработаны навыки владения методами надёжного программирования и эффективности разработки программного обеспечения в «Visual Studio 2019» с использованием языка программирования C#. Было составлено техническое задание, разработан технический проект, проведено испытание программы.

Для достижения цели дипломного проектирования были решены следующие задачи:

* определена вычислительная система, необходимая для создания программного средства;
* разработаны логическая и физическая модели данных;
* по модели выполнено проектирование задачи;
* разработано программное средство;
* описано созданное программное средство;
* выбрана методика испытания;
* описан процесс функционального тестирования;
* приведены примеры в области применения.

Разработка имеет интуитивно понятный графический интерфейс, позволяющий с минимальным знанием компьютера пользоваться данным программным средством.

При обучении возникают трудности с предоставлением информации и проверки знаний преподавателями. Для этого предназначено программное средство «Страховое дело». Оно позволит быстро работать с информацией и проводить проверку знаний учеников с моментальным результатом, тем самым сокращая время. Программа реализована в полном объеме и в соответствии с заданными требованиями. Полостью отлажена и протестирована. Поставленные задачи выполнены.

Для обеспечения надежности функционирования программы, была проведена опытная эксплуатация. Программное средство может модифицировать, в соответствии с пожеланиями заказчика.

В процессе разработки использовался в большом объеме материал по программированию и алгоритмизации, что способствовало закреплению наработанных навыков и умений в этих областях.

Также были изучены вопросы по охране труда и окружающей среды: Правовые, нормативные, социально-экономические и организационные вопросы охраны труда; Разработка мер по нормализации вентиляции в помещении; Пожарная безопасность; Охрана окружающей среды.

Вопросы по экономической части: Обоснование необходимости выведения на рынок программного продукта; Составление плана по разработке программного продукта; Расчет затрат на разработку программного продукта; Экономический эффект у разработчика программного продукта; Экономический эффект у пользователя программного продукта.

Список информационных источников

1. Андруш, В.Г. Охрана труда / В.Г. Андруш, Л.Т. Ткачёва, К.Д. Яшин. – Минск, 2019.
2. Багласова, Т.Г. Методические указания по выполнению дипломного проекта для учащихся по специальности 2-40 01 01 «Программное обеспечение технологий» / Т.Г. Багласова. – Минск : КБП, 2017. – 24 c.
3. Багласова, Т.Г. Методические указания по оформлению курсовых и дипломных проектов / Т.Г. Багласова, К.О. Якимович. – Минск : КБП, 2013. – 29 c.
4. Головачев, А.С. Экономика предприятия / А.С. Головачев. – Минск : Вышэйшая школа, 2015. – 687 c.
5. Общие требования к тестовым документам : ГОСТ 2.105-95. – Введ. 01.01.1996 – Минск : Межгос. совет по стандартизации, метрологии и сертификации, 1995. – 84 с.
6. .NET Framework [Электронный ресурс]. – Режим доступа : https://docs.microsoft.com/en-us/dotnet/framework/. – Дата доступа : 23.02.2022.
7. Dr.Explain [Электронный ресурс]. – Dr.Explain, 2022. – Режим доступа : https://www.drexplain.ru/. – Дата доступа : 11.02.2022.
8. MS Word [Электронный ресурс]. – Режим доступа : https://officeapplications.net/microsoft-word/. – Дата доступа : 24.01.2022.
9. phpMyAdmin [Электронный ресурс]. – Режим доступа : https://docs.microsoft.com/ru-ru/sql/tutorials/tutorials-for-phpMyAdmin?view= phpMyAdmin/. – Дата доступа : 27.01.2022.
10. Smart Install Maker [Электронный ресурс]. – Режим доступа : https://www.ixbt.com/news/soft/index.shtml?10/52/79/. – Дата доступа : 28.01.2022.
11. Visio [Электронный ресурс]. – Режим доступа : https://www.visio.com/. – Дата доступа : 20.01.2022.
12. C# [Электронный ресурс]. – Режим доступа : https://docs.microsoft.com/en-us/dotnet/c-sharp/. – Дата доступа : 18.02.2022.
13. Visual Studio [Электронный ресурс]. – Режим доступа : https://visualstudio.microsoft.com/vs/. – Дата доступа : 18.02.2022.

Приложение А

(обязательное)

Текст программных модулей

namespace Страховое дело

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public static Главна k;

public static Голосарий z;

public static Самоконтроль s;

public static О приложении c;

public MainWindow()

{

InitializeComponent();

}

private void Главная\_Click(object sender, RoutedEventArgs e)

{

if (k == null)

{

k = new Главная();

this.Hide();

k.Show();

}

else k.Activate();

}

private void Голосарий\_Click(object sender, RoutedEventArgs e)

{

if (z == null)

{

z = new Голосарий();

this.Hide();

z.Show();

}

else z.Activate();

}

private void Самоконтроль\_Click(object sender, RoutedEventArgs e)

{

if (s == null)

{

s = new Самоконтроль();

this.Hide();

s.Show();

}

else s.Activate();

}

private void Оприложении\_Click(object sender, RoutedEventArgs e)

{

if (c == null)

{

c = new Оприложении();

this.Hide();

c.Show();

}

else c.Activate();

}

}}

namespace Страховое дело

{

/// <summary>

/// Логика взаимодействия для Window1.xaml

/// </summary>

public partial class Window1: Window

{

DataClasses1DataContext dc = new DataClasses1DataContext(Properties.Settings.Default.РИПОConnectionString);

public int idd;

DateTime b;

public Window1(DateTime a)

{

InitializeComponent();

dateLabel.Content = a.ToString();

b = a;

}

private void Назад\_Click(object sender, RoutedEventArgs e)

{

MainWindow w1 = new MainWindow();

w1.Show();

}

private void Назад\_Click1(object sender, RoutedEventArgs e)

{

string[] Date\_3 = b.ToString().Split(' ');

string[] Date4 = Date\_3[0].Split('.');

string[] Time = time.Text.Split(':');

int a = dc.ExecuteQuery<int>("Select \* from Голосарий").Count();

dc.ExecuteQuery< Голосарий>($"insert into Голосарий(ID,Слово,Лекция) values ({a + 1},'{work.Text}','{ }')");

}

private void TextBox\_TextChanged(object sender, TextChangedEventArgs e)

{ }

private void work\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{ }

}

}

namespace Страховое дело

{

/// <summary>

/// Логика взаимодействия для Голосарий.xaml

/// </summary>

///

public partial class Голосарий : Microsoft.Office.Interop.Excel.Window

{

DataClasses1DataContext dc = new DataClasses1DataContext(Properties.Settings.Default.ЗАГСConnectionString);

public static MainWindow m;

public Голосарий()

{

InitializeComponent();

phonesGrid.ItemsSource = dc.ExecuteQuery<Голосарий>("Select \* from Голосарий");

}

private void Назад\_Click(object sender, RoutedEventArgs e)

{

if (m == null)

{

m = new MainWindow();

this.Hide();

m.Show();

}

else m.Activate();

}

private void Поиск\_Click(object sender, RoutedEventArgs e)

{

try

{

int a = Convert.ToInt32(search.Text);

phonesGrid.ItemsSource = dc.ExecuteQuery<Заявка>($"Select \* from Голосарий where ID = {a}");

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void Сортировка\_1\_Click(object sender, RoutedEventArgs e)

{

(phonesGrid.ItemsSource as DataView).Sort = "Тип";

}

private void Сортировка\_2\_Click(object sender, RoutedEventArgs e)

{

phonesGrid.ItemsSource = dc.ExecuteQuery<Заявка>($"Select \* from Голосарий order by Часы, Минуты");

}

private void Удалить\_Click(object sender, RoutedEventArgs e)

{

if (phonesGrid.SelectedItems != null)

{

int a = dc.ExecuteQuery<int>("select id from Самоконтроль ").ElementAt(phonesGrid.SelectedIndex);

dc.ExecuteQuery<Заявка>($"delete from Самоконтроль where id ={a}");

}

phonesGrid.ItemsSource = dc.ExecuteQuery< Самоконтроль >("Select \* from Самоконтроль ");

}

private void Отчет\_Click(object sender, RoutedEventArgs e)

{

Excel.Application excel = new Excel.Application();

excel.Visible = true;

Workbook workbook = excel.Workbooks.Add(System.Reflection.Missing.Value);

Worksheet sheet1 = (Worksheet)workbook.Sheets[1];

for (int j = 0; j < phonesGrid.Columns.Count; j++)

{

Range myRange = (Range)sheet1.Cells[1, j + 1];

sheet1.Cells[1, j + 1].Font.Bold = true;

sheet1.Columns[j + 1].ColumnWidth = 15;

myRange.Value2 = phonesGrid.Columns[j].Header;

}

for (int i = 0; i < phonesGrid.Columns.Count; i++)

{

for (int j = 0; j < phonesGrid.Items.Count; j++)

{

TextBlock b = phonesGrid.Columns[i].GetCellContent(phonesGrid.Items[j]) as TextBlock;

Microsoft.Office.Interop.Excel.Range myRange = (Microsoft.Office.Interop.Excel.Range)sheet1.Cells[j + 2, i + 1];

myRange.Value2 = b.Text;

}

}

}

private void Сортировка\_3\_Click(object sender, RoutedEventArgs e)

{

phonesGrid.ItemsSource = dc.ExecuteQuery<Заявка>($"Select \* from Самоконтроль order by Дата");

}

private void Сортировка\_4\_Click(object sender, RoutedEventArgs e)

{

type.Visibility = Visibility.Visible;

}

dynamic Excel.Window.Activate()

{

throw new NotImplementedException();

}

dynamic Excel.Window.ActivateNext()

{

throw new NotImplementedException();

}

dynamic Excel.Window.ActivatePrevious()

{

throw new NotImplementedException();

}

bool Excel.Window.Close(object SaveChanges, object Filename, object RouteWorkbook)

{

throw new NotImplementedException();

}

dynamic Excel.Window.LargeScroll(object Down, object Up, object ToRight, object ToLeft)

{

throw new NotImplementedException();

}

Excel.Window Excel.Window.NewWindow()

{

throw new NotImplementedException();

}

dynamic Excel.Window.\_PrintOut(object From, object To, object Copies, object Preview, object ActivePrinter, object PrintToFile, object Collate, object PrToFileName)

{

throw new NotImplementedException();

}

dynamic Excel.Window.PrintPreview(object EnableChanges)

{

throw new NotImplementedException();

}

dynamic Excel.Window.ScrollWorkbookTabs(object Sheets, object Position)

{

throw new NotImplementedException();

}

dynamic Excel.Window.SmallScroll(object Down, object Up, object ToRight, object ToLeft)

{

throw new NotImplementedException();

}

int Excel.Window.PointsToScreenPixelsX(int Points)

{

throw new NotImplementedException();

}

int Excel.Window.PointsToScreenPixelsY(int Points)

{

throw new NotImplementedException();

}

dynamic Excel.Window.RangeFromPoint(int x, int y)

{

throw new NotImplementedException();

}

void Excel.Window.ScrollIntoView(int Left, int Top, int Width, int Height, object Start)

{

throw new NotImplementedException();

}

dynamic Excel.Window.PrintOut(object From, object To, object Copies, object Preview, object ActivePrinter, object PrintToFile, object Collate, object PrToFileName)

{

throw new NotImplementedException();

}

Excel.Application Excel.Window.Application => throw new NotImplementedException();

XlCreator Excel.Window.Creator => throw new NotImplementedException();

dynamic Excel.Window.Parent => throw new NotImplementedException();

Range Excel.Window.ActiveCell => throw new NotImplementedException();

Chart Excel.Window.ActiveChart => throw new NotImplementedException();

Pane Excel.Window.ActivePane => throw new NotImplementedException();

dynamic Excel.Window.ActiveSheet => throw new NotImplementedException();

dynamic Excel.Window.Caption { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayFormulas { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayGridlines { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayHeadings { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayHorizontalScrollBar { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayOutline { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.\_DisplayRightToLeft { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayVerticalScrollBar { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayWorkbookTabs { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayZeros { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.EnableResize { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.FreezePanes { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.GridlineColor { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

XlColorIndex Excel.Window.GridlineColorIndex { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

double Excel.Window.Height { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.Index => throw new NotImplementedException();

double Excel.Window.Left { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

string Excel.Window.OnWindow { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

Panes Excel.Window.Panes => throw new NotImplementedException();

Range Excel.Window.RangeSelection => throw new NotImplementedException();

int Excel.Window.ScrollColumn { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.ScrollRow { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

Sheets Excel.Window.SelectedSheets => throw new NotImplementedException();

dynamic Excel.Window.Selection => throw new NotImplementedException();

bool Excel.Window.Split { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.SplitColumn { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

double Excel.Window.SplitHorizontal { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.SplitRow { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

double Excel.Window.SplitVertical { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

double Excel.Window.TabRatio { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

double Excel.Window.Top { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

XlWindowType Excel.Window.Type => throw new NotImplementedException();

double Excel.Window.UsableHeight => throw new NotImplementedException();

double Excel.Window.UsableWidth => throw new NotImplementedException();

bool Excel.Window.Visible { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

Range Excel.Window.VisibleRange => throw new NotImplementedException();

double Excel.Window.Width { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.WindowNumber => throw new NotImplementedException();

XlWindowState Excel.Window.WindowState { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

dynamic Excel.Window.Zoom { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

XlWindowView Excel.Window.View { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayRightToLeft { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

SheetViews Excel.Window.SheetViews => throw new NotImplementedException();

dynamic Excel.Window.ActiveSheetView => throw new NotImplementedException();

bool Excel.Window.DisplayRuler { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.AutoFilterDateGrouping { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

bool Excel.Window.DisplayWhitespace { get => throw new NotImplementedException(); set => throw new NotImplementedException(); }

int Excel.Window.Hwnd => throw new NotImplementedException();

private void О\_приложении\_Click(object sender, RoutedEventArgs e)

{

Оприложении w1 = new Оприложении ();

w1.Show();

this.Close();

}

private void type\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

string[] a = type.SelectedValue.ToString().Split(' ');

phonesGrid.ItemsSource = dc.ExecuteQuery<Заявка>($"Select \* from Самоконтроль where Тип = '{a[1]}'");

type.Visibility = Visibility.Hidden;

}

}

}

namespace Страховое дело

{

/// <summary>

/// Логика взаимодействия для Голосарий.xaml

/// </summary>

public partial class Голосарий : Window

{

public static MainWindow m;

private void calendar\_SelectedDatesChanged(object sender, SelectionChangedEventArgs e)

{

DateTime? selectedDate = calendar1.SelectedDate;

Window1 a = new Window1(Convert.ToDateTime(selectedDate));

a.Show();

}

public Голосарий ()

{

InitializeComponent();

}

private void Назад\_Click(object sender, RoutedEventArgs e)

{

if (m == null)

{

m = new MainWindow();

this.Hide();

m.Show();

}

else m.Activate();

}

private void Выбрать\_Click(object sender, RoutedEventArgs e)

{

}

}

}

namespace Страховое дело

{

/// <summary>

/// Логика взаимодействия для О\_приложении.xaml

/// </summary>

public partial class О\_приложении: Window

{

public static MainWindow m;

public О\_приложении ()

{

InitializeComponent();

}

private void Назад\_Click(object sender, RoutedEventArgs e)

{

if (m == null)

{

m = new MainWindow();

this.Hide();

m.Show();

}

else m.Activate();

}

}

}



*Инв.№дубл.*

*Взам.инв.№*

*Подп. и дата*

*Н. Контр*

*Багласова .В.*

*Утверд.*

*Багласова Т.Г.*

*Реценз.*

*.*

*КБП*

*У*



Создание обучающей программы по учебной дисциплине «Страховое дело»

*Диаграмма вариантов использования*

*№ докум.*

*Подпись*

*Дата*

*Изм.*

*Лист*

*Масса*

*Лит.*

*Масштаб*

*Листов 5*

*Лист 1*

*Разраб.*

*Новикова М.С*

*Инв.№подл.*

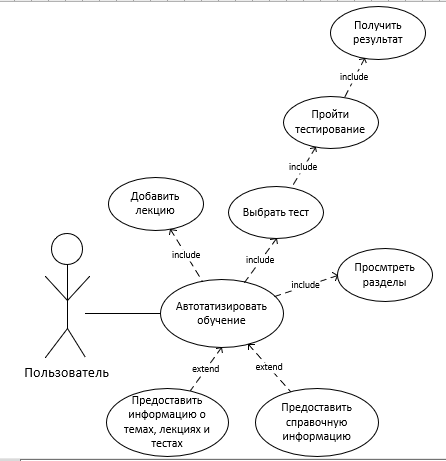
*Подп. и дата*

*Т. Контр.*

*Якимович К.О.*

*Провер.*

*Якимович К.О.*





*Инв.№дубл.*

*Взам.инв.№*

*Подп. и дата*

*Н. Контр*

*Багласова Е.В.*

*Утверд.*

*Багласова Т.Г.*

*Реценз.*

*.*

*КБП*

*У*



Создание обучающей программы по учебной дисциплине «Страховое дело»

*Диаграмма классов*

*№ докум.*

*Подпись*

*Дата*

*Изм.*

*Лист*

*Масса*

*Лит.*

*Масштаб*

*Листов 5*

*Лист 2*

*Разраб.*

*Новикова М.С.*

*Инв.№подл.*

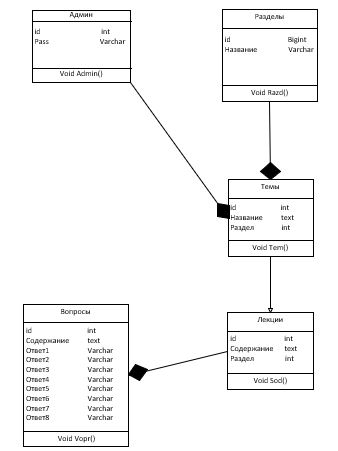
*Подп. и дата*

*Т. Контр.*

*Якимович К.О.*

*Провер.*

*Якимович К.О.*





*Инв.№дубл.*

*Взам.инв.№*

*Подп. и дата*

*Н. Контр*

*Багласова Е.В.*

*Утверд.*

*Багласова Т.Г.*

*Реценз.*

*.*

*КБП*

*У*



Создание обучающей программы по учебной дисциплине «Страховое дело»

*Диаграмма деятельности*

*№ докум.*

*Подпись*

*Дата*

*Изм.*

*Лист*

*Масса*

*Лит.*

*Масштаб*

*Листов 5*

*Лист 3*

*Разраб.*

*Новикова М.С.*

*Инв.№подл.*

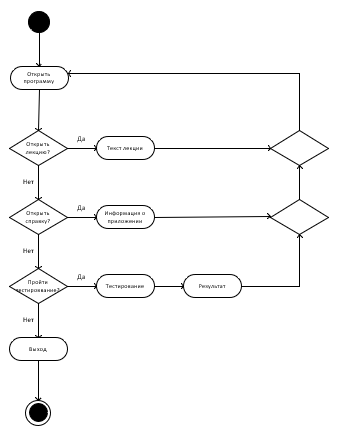
*Подп. и дата*

*Т. Контр.*

*Якимович К.О.*

*Провер.*

*Якимович К.О.*





*Инв.№дубл.*

*Взам.инв.№*

*Подп. и дата*

*Н. Контр*

*Багласова Е.В.*

*Утверд.*

*Багласова Т.Г.*

*Реценз.*

*.*

*КБП*

*У*

*ДП Т.89106.401 ГЧ*

Создание обучающей программы по учебной дисциплине «Страховое дело»

*Диаграмма последовательности*

*№ докум.*

*Подпись*

*Дата*

*Изм.*

*Лист*

*Масса*

*Лит.*

*Масштаб*

*Листов 5*

*Лист 4*

*Разраб.*

*Новикова М.С.*

*Инв.№подл.*

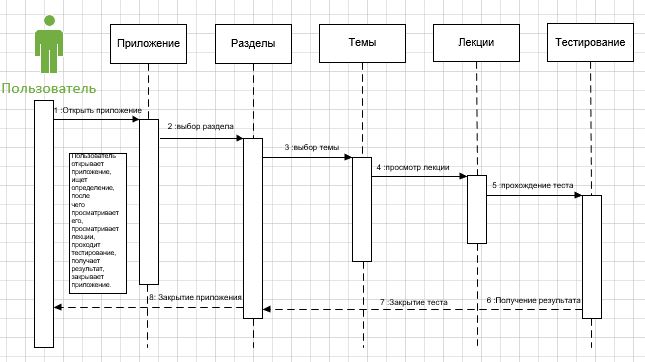
*Подп. и дата*

*Т. Контр.*

*Якимович К.О.*

*Провер.*

*Якимович К.О.*





*Инв.№дубл.*

*Взам.инв.№*

*Подп. и дата*

*Н. Контр*

*Багласова Е.В.*

*Утверд.*

*Багласова Т.Г.*

*Реценз.*

*.*

*КБП*

*У*

*ДП Т.89106.401 ГЧ*

Создание обучающей программы по учебной дисциплине «Страховое дело»

*Диаграмма компонентов*

*№ докум.*

*Подпись*

*Дата*

*Изм.*

*Лист*

*Масса*

*Лит.*

*Масштаб*

*Листов 5*

*Лист 5*

*Разраб.*

*Новикова М.С.*

*Инв.№подл.*

*Подп. и дата*

*Т. Контр.*

*Якимович К.О.*

*Провер.*

*Якимович К.О.*

