**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**федеральное государственное автономное образовательное   
учреждение высшего образования   
«Санкт-Петербургский политехнический университет Петра Великого»**

**ИНСТИТУТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

|  |  |
| --- | --- |
|  | **Допустить к защите**  Заместитель директора по  учебно-методической работе  \_\_\_\_\_\_\_\_\_\_\_ \_\_Е. Г. Конакина\_\_\_\_\_  (Подпись) (И.О.Ф.)  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. |
|  |

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

Тема Разработка информационной системы складского учета

специальность 09.02.07 группа 42919/4

Студент (ка) Прозоров А.Д.

(подпись) (ФИО)

Руководитель Славгородская А.А.

(подпись) (ФИО)

Санкт-Петербург

2022

СОДЕРЖАНИЕ

[Введение 4](#_Toc105061358)

[1 Общая часть 5](#_Toc105061359)

[1.1 Анализ предприятия 5](#_Toc105061360)

[1.3 Функции проектируемого приложения 5](#_Toc105061362)

[1.4 Анализ рынка существующих решений 5](#_Toc105061363)

[1.5 Обоснование и выбор методики, технологии и инструментальных средств разработки 6](#_Toc105061364)

[2 Специальная часть 10](#_Toc105061365)

[2.1 Постановка задачи 10](#_Toc105061366)

[2.2 Описание структуры приложения 11](#_Toc105061367)

[2.3 Проектирование информационной системы 14](#_Toc105061368)

[2.4 Архитектура и описание структуры базы данных 15](#_Toc105061369)

[2.5 Разработка пользовательского интерфейса 19](#_Toc105061370)

[3 Экономическая часть 24](#_Toc105061371)

[3.1 Область применения программного продукта и его преимущества перед аналогичным программным продуктом 24](#_Toc105061372)

[3.2 Трудоемкость разработки программного продукта, квалификация исполнителя и его оклад 24](#_Toc105061373)

[3.3 Расчет затрат на разработку 26](#_Toc105061374)

[3.4 Расчет затрат на внедрение 30](#_Toc105061380)

[3.5 Расчет цены и прибыли 33](#_Toc105061385)

[4 Техника безопасности и охрана труда 36](#_Toc105061386)

[4.1 Анализ условий труда программиста. 36](#_Toc105061387)

[4.2 Расчет искусственного освещения рабочего места программиста. 38](#_Toc105061388)

[4.3 Электробезопасность на предприятии. 40](#_Toc105061389)

[4.4 Пожарная безопасность на предприятии 42](#_Toc105061390)

[Заключение 45](#_Toc105061391)

[Список использованных источников 46](#_Toc105061392)

[Приложение А (справочное) Исходный код 48](#_Toc105061393)

# ВВЕДЕНИЕ

В настоящее время для повышения своей эффективности все больше организаций стараются не только повысить качество товаров, но и автоматизировать учет товара, продаж и поставок. Сейчас хранение, поиск и доступ к информации стали важным явлением для работников склада. Это помогает им сократить время на поиск, приобретение, хранение, обмен различными ресурсами.

Большинство складов по-прежнему используют огромные папки с документами для поиска данных о товарах, и если данные по какой-либо причине утеряны, то восстановление этих данных может стать настоящей проблемой.

Для предприятие занимающимся торговым бизнесом очень важен контроль материальных ценностей, так как материальные ценности являются основной прибылью. Каким бы честным не были сотрудники, бесконтрольность может привести к обману или невыполнению рабочих обязанностей.

Набор задач, выполняемых на разных складах, примерно одинаков. Это объясняется тем, что склады выполняют схожие функции в разных логических процессах, таких как: временное хранение и хранение.

С развитие прогресса и компьютерного рынка программного обеспечения, у большинства складов появилась необходимость в автоматизации складского учета. Автоматизация складского учета способна сократить человеческие затраты, оперативно выдавать результат необходимый работнику, а также поменять большие архивы на структурированное хранение информации в электронном виде.

Автоматизация складского учета влияет на качество и скорость выполнения ключевых процессов на складе, а также это способ оптимизации бизнес-процессов склада.

# Общая часть

## Анализ предприятия

Магазин Exist начал свою работу в 2014 году. Целью удачного бизнес-процесса было предложить клиентам самый широкий спектр автомобильных запасных частей и аксессуаров. С ростом числа клиентов увеличивалось и количество автозапчастей на складе. В определенный момент компания поняла, что без системы складского учета очень сложно контролировать склад и вовремя заказывать новые запчасти.

Была поставлена задача сделать информационную систему складского учета. Внедрение информационных систем на предприятии необходимо, так как персонал склада фиксирует все операции вручную, что очень трудоёмко и не исключает большого количества ошибок из-за человеческого фактора.

## Функции проектируемого приложения

Разработанная информационная система обладает следующим функционалом для администратора системы.

* Добавление и удаление информации с БД;
* Добавление в БД текстовой информации о товаре;
* Запросы (на удаление и добавление, на выборку) и формы (главная форма, формы на добавление информации в таблицы) для обработки хранимой информации;
* Удобный и наглядный интерфейс для пользователей, которые не имеют опыт работы с БД.

## Анализ рынка существующих решений

На рынке программного обеспечения сегодня представлен широкий спектр типовых программных решений, продуктов для малых предприятий.

Проанализировав рынок существующих решений, стало понятно, что многие системы не идеальны. Например, система программ «1С: Предприятие» предназначена для решения широкого спектра задач автоматизации учета и управления, но имеет неудобный и очень сложный для понимания интерфейс, который представлен на рисунке 1.

Была поставлена задача разработать простой и понятный интерфейс, который не требует длительного поиска необходимых функций и доступным расположением информации для того, чтобы в дальнейшем у работников склада не возникло проблема с адаптацией к информационной системе.

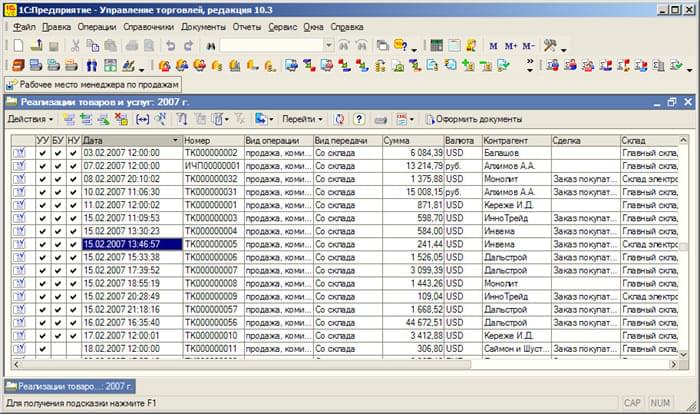


Рисунок 1 – «1С: Предприятие»

## Обоснование и выбор методики, технологии и инструментальных средств разработки

Анализ имеющегося на рынке программного обеспечения показал, что большинство информационных систем имеют достаточно сложный интерфейс.

Поэтому надо разработать систему, которая при наличии всех необходимых и некоторых дополнительных функций по ведению склада будет иметь максимально простой интерфейс.

В настоящий момент для прикладного программирования в среде операционный систем Windows созданы различные средства программирования.

К их числу можно отнести следующее:

* С++;
* Visual Basic;
* Delphi;
* Visual FoxPro.

Все указанные средства программирование объединяет одно общее т.е. использованием визуального интерфейса для создания исходного и выполняемого кода, а также при создании баз данных и работе с ними.

Выбрать язык для разработки корпоративных Windows-приложений в наше время уже не так просто, как это было раньше. Лучшим языком для коммерческой разработки всегда считался С++, но сейчас у перегруженных работой программистов не хватает времени и терпения на нудные циклы «написание кода — компиляция — отладка», многократно повторяемые из-за бесчисленных утечек памяти и прочих неявных ошибок, часто допускаемых при написании кода на С++. Несмотря на то, что программировать на С++ под Windows стало намного проще, темпы работы, задаваемые организацией, не оставляют программистам времени на «борьбу» со средствами разработки.

Java по сравнению с С++ сокращает цикл разработки, лучше защищает код и обеспечивает возможность запуска приложений на многих платформах. Java не идеален, он разительно отличается от С++, и в большинстве случаев подходит для Windows не лучше, чем перчатка для ноги. И все же Java остается единственным разумным выбором для корпоративных проектов.

Массовое бегство разработчиков Windows-приложений в стан Java не оставалось без внимания Microsoft, которая поспешила выпустить открытую предварительную бета-версию комплекта средств разработки для платформы .NET. Загрузив эти 86 Мбайт, можно посмотреть, как будет выглядеть Windows, и ознакомиться с компилятором С# — нового языка программирования.

Эта вышедшая из-под пера Microsoft вариация на тему С++ — настоящий бальзам на раны измученных разработчиков Windows-приложений. В отличие от С++ и Java, C# позволяет без затруднений осуществлять доступ к внутренним сервисам Windows — сетевым объектам, пользовательским интерфейсам. Как и Java, C# существенно повышает стабильность за счет автоматического управления ресурсами и предотвращения распространенных ошибок программирования.

Тем, кто уже знает С++, С# будет выучить гораздо проще, чем Java. C# это стандартизованный язык, причем находящийся в ведении не Microsoft, а Европейской ассоциации производителей компьютеров (ECMA), — той же независимой организации, которая контролирует JavaScript. И хотя в предварительной версии С# на это нет и намека, Microsoft уже сообщила о намерении перенести .NET и С# на платформы, отличные от Windows.

Windows Forms был выбран потому, что на нем можно писать кроссплатформенные приложения. Простые проекты, написанные на Windows Forms, можно довольно легко перенести на другую операционную систему, если на ней установлен .Net Framework нужной модели, на котором написан Ваш проект.

Для хранения данных я выбрал SQL Server, т.к. он обладает следующими преимуществами:

* Масштабирование системы. Взаимодействовать с ней можно как на простых ноутбуках, так и на ПК с мощным процессором, который способен обрабатывать большой объем запросов.
* Размер стариц до 8 Кб. Данные извлекаются быстро, а сложную информацию удобнее хранить. Система обрабатывает транзакции в интерактивном режиме, есть динамическая блокировка.
* Автоматизация рутинных административных задач. Например, управление блокировками и памятью, редактура размеров файлов. В программе продуманы настройки, можно создавать профили пользователей.
* Удобный поиск. Его можно осуществлять по фразам, словам, тексту.
* Поддержка работы с другими решениями Майкрософт, в том числе с Excel, Access.

# Специальная часть

## Постановка задачи

Целью данного проекта является создание информационной системы складского учета.

Задачей данного проекта является разработка специализированной системы управления контентом для информационной системы складского учета. Объем требований к данной информационной системе позволяет создать быструю и комфортную для пользователя систему управления. Каждая страница формируется из специализированных блоков.

Разрабатываемая информационная система склада должна иметь следующие возможности для Администратора системы.

* + Добавлять и удалять информацию с БД;
  + Добавлять в бд текстовую информации о товаре;
  + Содержать конструктор таблиц для добавления на страницы табличных данных;
  + Иметь необходимые запросы (на удаление и добавление, на выборку) и формы (главная форма, формы на добавление информации в таблицы) для обработки хранимой информации;
  + Обеспечение удобного и наглядного интерфейса администратора, который не имеет навыков работы с базами данных;
  + Обеспечение надежного хранения информации;
  + Обеспечивать защиту от несанкционированного доступа;
  + Контролировать избыточность, непротиворечивость, сохранность и достоверность хранимой в базе данных информации.

Для пользователей информационной системы управления должна предоставлять следующие возможности.

* + Возможность просматривать товары;
  + Возможность оформления прихода товаров;
  + Возможность оформления расхода;
  + Возможность добавления товарной группы;
  + Возможность добавления единиц измерения.

Управление содержимым информационной системы выполняется Администратором.

Администратор системы после входа в систему управления (после ввода логина и пароля) получает полный доступ к базе данных, имеет возможность добавлять и удалять содержимое всех таблиц.

## Описание структуры приложения

Структура приложения информационной система складского учета включает в себя несколько модулей:

1. Модуль авторизации, который представлен на рисунке 2.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 – Модуль авторизации

1. Модуль регистрации, который представлен на рисунке 3.

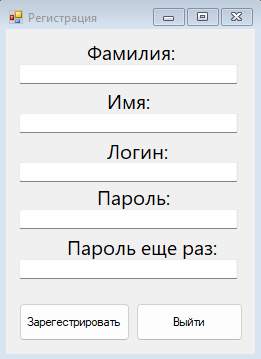


Рисунок 3 – Форма регистрации

1. Главный модуль, представленный на рисунке 4, который выполняется сразу после успешной авторизации пользователя.

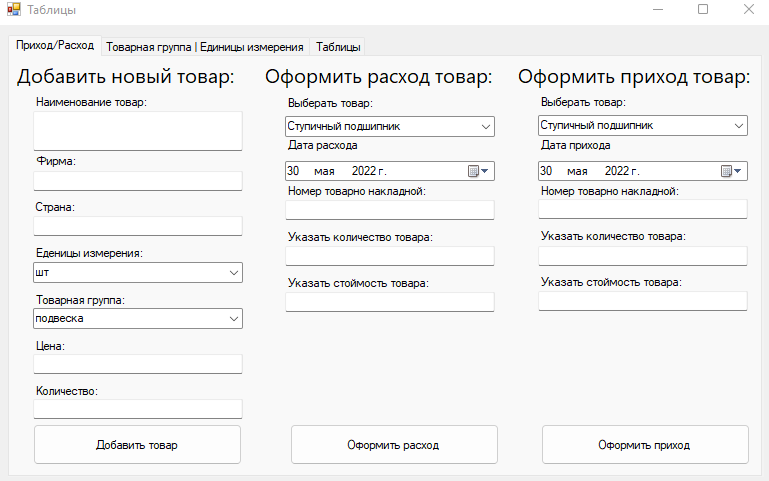


Рисунок 4 – Главный модуль

1. Модуль просмотра выбранной таблицы, который представлен на рисунке 5.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 5 – Раздел таблицы

Модуль авторизации содержит процедуры для проверки логина и пароля и при успешной проверки авторизует пользователя.

Модуль регистрации содержит процедуры добавления информации из формы регистрации, а также этот модуль проверяет корректность этой информации.

Главный модуль реализует процедуры добавления информации в таблицы: «Товары», «Приход», «Расход», «Товарные\_группы», «Единицы\_измерения».

Модуль просмотра выбранной таблицы реализует добавление и редактирование записей в выбранной таблицы, а также реализует поиск записей и настройку фильтра для удобного отображения записей.

## Проектирование информационной системы

Для проектирования информационной системы была выбрана методология объектно-ориентированного проектирования информационной системы. Для описания функционала системы необходимо использовать следующие диаграммы языка проектирования UML:

1. Диаграмма вариантов использования (прецедентов);
2. Диаграмма последовательности, представленная на рисунке 7.

Для отображения отношений между актерами и прецедентами, а также для описания системы на концептуальном уровне построена диаграмма вариантов использования, которая представлен на рисунке 6.

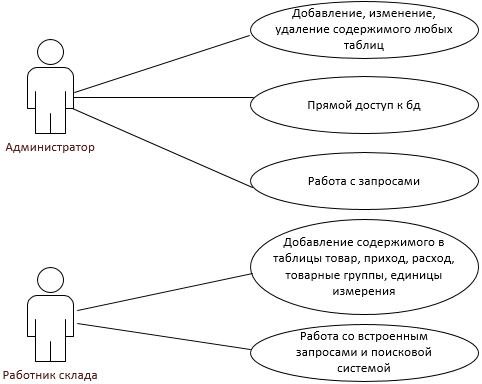


Рисунок 6 – Диаграмма прецедентов

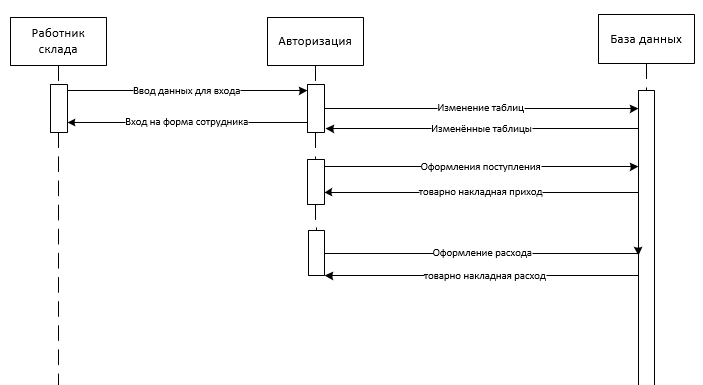


Рисунок 7 – Диаграмма последовательности

## Архитектура и описание структуры базы данных

База данных «Склад» создается для удовлетворения потребностей пользователей в информации. Данные о товаре, его характеристиках, приходе, расходе и остатках позволят более оперативно отслеживать потребности в товаре, а также проводить инвентаризацию на складе.

Последовательность решаемых БД задач можно представить, как:

* Единицы измерения товаров;
* Товарные группы;
* Товар;
* Поставщики;
* Получатели;
* Работники склада;
* Предоставление информации из этих справочников;
* Проведения прихода товара на склад от поставщиков;
* Ввод данных по товарной накладной;
* Проведение расхода товара со склада получателям;
* Ввод данных по товарной накладной;
* Просмотри связей товара и поставщика / получателя;
* По товару: информация о товаре, кто является поставщиком данного товара, кто является получателем данного товара;
* По поставщику: информация о поставщике, какой товар он поставляет на склад;
* По получателю: информация о получателе, какой товар он получает со склада.

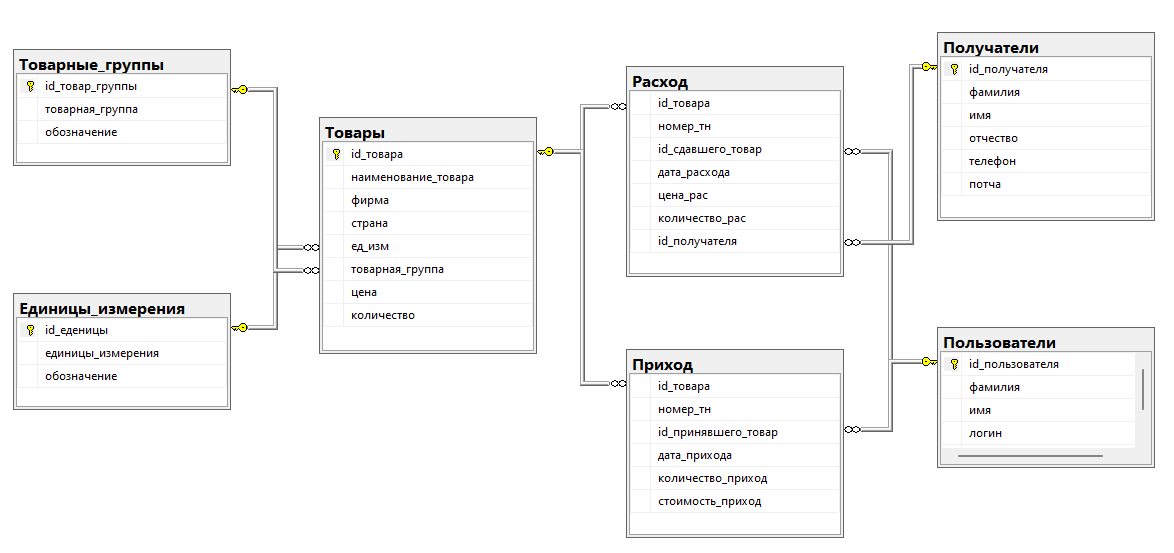


Рисунок 8 – Схема БД

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 9 – Таблица товары

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 10 – Таблица товарные группы

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 11 – Таблица расходы

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 12 – Таблица приход

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 13 – Таблица пользователи

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 14 – Таблицы получатели

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 15 – Таблица единицы измерения

## Разработка пользовательского интерфейса

В данной информационной системе складского учета были разработаны следующие формы: форма входа, форма регистрации, главная форма и форма таблицы.

Форма входа разработана для того, чтобы у каждый пользователь мог зайти в приложение под своим пользователем.

Форма регистрации разработана для регистрации пользователя.

Главная форма предназначена для удобного добавления записей в таблицу. На ней расположены поля для ввода данных, выпадающие списки для выбора данных и кнопка для добавления данных.

Форма таблицы предназначена для удобного добавления, редактирования и поиска записей в выбранной таблице.

Далее на рисунках 16–22 представлен интерфейс информационной системы. При запуске программы появится форма входа, которая представлена на рисунке 16, в котором пользователь должен указать свой логин и пароль, после чего пользователь должен нажать на кнопку «Войти». Пользователь также может зарегистрироваться, нажав на кнопку «Регистрация».

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 16 – Форма входа

При нажатии на кнопку «Регистрация», появится форма регистрации, которая представлена на рисунке 17. Пользователю необходимо указать все данные, которые требуются в форме регистрации и затем нажать на кнопку «Зарегистрироваться».

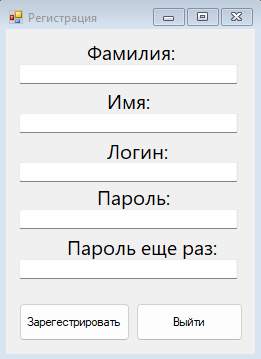


Рисунок 17 – Форма регистрации

При успешном входе пользователь попадает на главную форму, которая представлена на рисунке 19 в противном случае кнопка войти заблокируется на 10 секунд, пример предоставлен на рисунке 18.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 18 – Ошибка входа

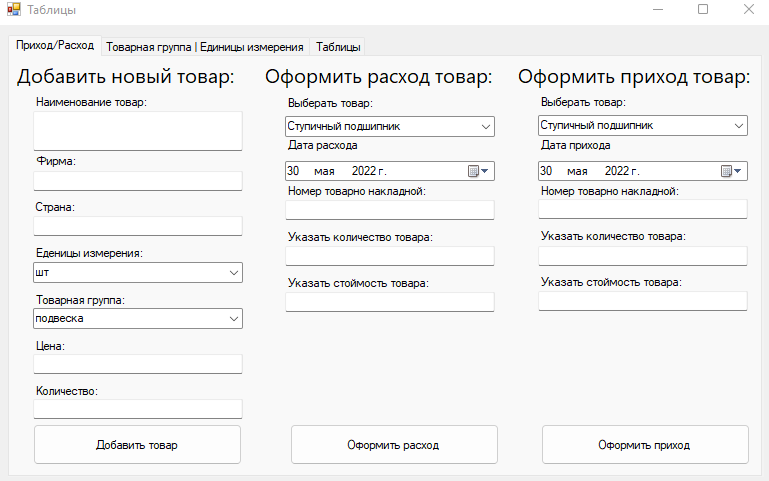


Рисунок 19 – Главная форма

На главной форме, которая представлена на рисунке 19 пользователь может работать с БД, а именно добавлять новый товар, оформлять приход и расход товара.

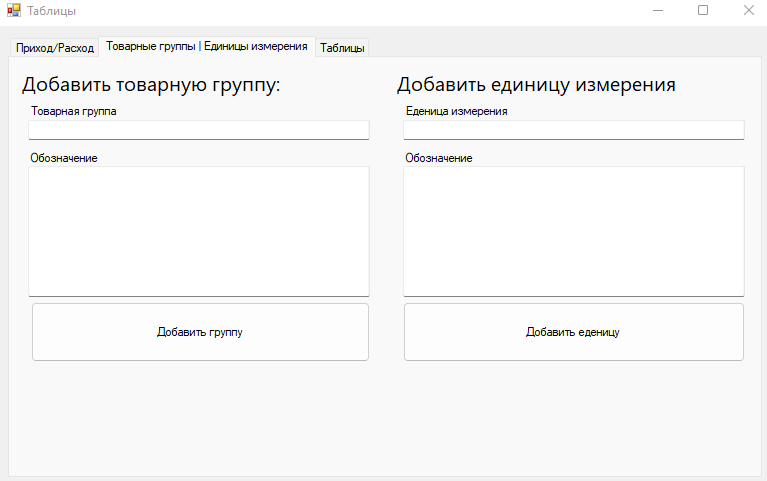


Рисунок 20 – Раздел товарные группы и единицы измерения

Нажав на вкладку «Товарные группы | Единицы измерения» перед пользователем появится раздел, который представлен на рисунке 20, в котором он может добавить новую товарную группу и единицу измерения.

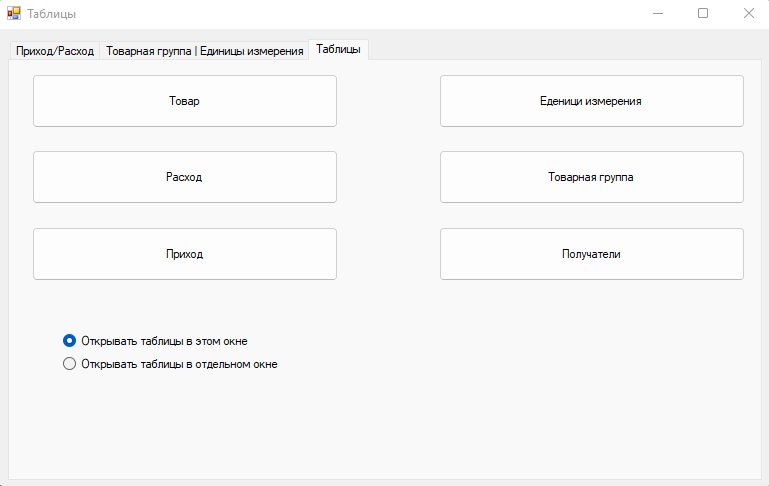


Рисунок 21 – Раздел таблицы

Во вкладке «Таблицы», которая представлена на рисунке 21, у пользователя появляется возможность просмотра, поиска и редактирования информации из таблиц. Для этого пользователю необходимо нажать на кнопку с названием нужной таблицы, так же пользователь может открывать таблицы в новом окне.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 22 – Раздел таблицы

На рисунке 22 изображена форма для работы с табличными данными.

Пользователь может настроить фильтр, выполнить поиск и отредактировать выбранную позиции в таблице.

1. **Экономическая часть** 
   1. Область применения программного продукта и его преимущества перед аналогичным программным продуктом

Система предназначена для пользования сотрудниками и клиентами турагентства, при создании новых туров, их заказе и оформлении путевки.

Достоинствами системы являются возможности подготовки квитанции об оформлении тура, а также формирования отчетной документации, чего нету в продуктах аналогах.

* 1. Трудоемкость разработки программного продукта, квалификация исполнителя и его оклад

Трудоемкость разработки можно определить в таблице 1. Строка «Всего» отображает общую трудоемкость разработки.

Таблица 1 – Трудоемкость разработки программного продукта

|  |  |  |
| --- | --- | --- |
| **Наименование этапа** | **Условное обозначение** | **Трудоемкость выполнения этапа, час** |
| Описание задания | То | 41 |
| Разработка UML - диаграмм | Тд | 33 |
| Разработка Базы данных | Тбд | 154 |
| Проектирование программы | Тпп | 310 |
| Отладка программы | Топ | 94 |
| Оценка качества программы | Тоц | 36 |
| Оформление документации | Тд | 108 |
| Всего | Тобщ | 776 |

Разработчики программного продукта представлены в таблице 2.

Таблица 2 – Разработчики программного продукта

|  |  |  |  |
| --- | --- | --- | --- |
| **Исполнители** | **Оклад, руб.** | **Часовая тарифная ставка, руб/час** | **Количество сотрудников** |
| Разработчик-программист | 46 000 | 273,81 | 1 |
| Руководитель проекта | 52 000 | 309,52 | 1 |
| Инженер по внедрению | 36 000 | 214,29 | 1 |

Часовая тарифная ставка ЧТС, руб./час., определяется исходя из месячного оклада, количества рабочих дней в месяце и продолжительности рабочего по формуле:

(4)

где Ом – оклад исполнителя в месяц, руб. /мес.;

Д – количество рабочих дней в месяце (для расчета Д = 21 раб. дня);

Тс – продолжительность рабочего дня (для расчета Тс = 8час.).

В таблице 3 показана стоимость технических средств для разработки системы.

Таблица 3 – Стоимость технических средств разработки

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование компонента** | **Цена, руб.** | **Количество, шт.** | **Стоимость, руб.** |
| Ноутбук Acer Aspire 3 A317-53-31HA | 45990 | 1 | 45990 |
| Мышь Logitech M170 | 709 | 1 | 709 |
| Итого: | | | 46699 |

В таблице 4 представлены затраты на расходные материалы.

Таблица 4 – Планируемые затраты на расходные материалы (Р)

| **Затраты** | **Стоимость** | **Количество** | **Сумма, руб.** |
| --- | --- | --- | --- |
| Интернет | 800 руб./месяц | 6 месяцев | 4800 |
| Электричество | 4,49 руб./КВт\*ч | 1256 КВт\*ч | 5639,44 |
| Итого: | | | 10439,44 |

* 1. Расчет затрат на разработку

Исходные данные, связанные с разработкой программного продукта приведены в таблице 5.

Таблица 5 – Исходные данные

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Обозначение** | **Значение** |
| Оклад разработчика | Ор | 98 000 руб. |
| Время разработки | Трп | 4 месяца |
| Машинное время разработки | Тмч | 3,5 месяца |
| Коэффициент дополнительной заработной платы | Кд | 0,17 |
| Коэффициент страховых взносов | Кст | 0,3 |
| Количество единиц техники | Q | 1 шт. |
| Себестоимость содержания техники | См/ч | 10 руб./час |
| Коэффициент готовности техники | Кгт | 0,97 |
| Число рабочих дней в месяце | ЧРД | 21 день |
| Продолжительность смены | Тсм | 8 часов |
| Коэффициент сменности | Ксм | 1 |
| Коэффициент транспортных расходов | Кт | 0,16 |
| Коэффициент накладных расходов | Кнр | 0,52 |

Оклад разработчика=сумме всех задействованных в разработке=разработчик+руководитель=46 000+52 000=98 000 руб.

Расчет полных затрат на разработку проектного решения (программного продукта) осуществляется по формуле:

Зрп = Зот + Зст + Зэвм + Зсп + Зрм + Знр, руб. (5)

где Зот - затраты на оплату труда разработчика (разработчиков), руб.;

Зст - страховые взносы по оплате труда во внебюджетные фонды, руб.;

Зэвм - затраты, связанные с содержанием вычислительной техники, руб.;

Зсп – затраты на технические средства, руб.;

Зрм - затраты на расходные материалы, необходимые при разработке программного продукта, руб.;

Знр - затраты по накладным расходам, приходящиеся на разработку программного продукта, руб.

* + 1. Затраты на оплату труда разработчиков (Зот), руб.

Размер фонда оплаты труда разработчиков (Зот) рассчитывается по формуле:

Зот = Ор \* Трп (1+Кд), руб. (6)

где Ор - месячный оклад разработчика проектного решения, руб./мес.;

Трп - время разработки проектного решения разработчиком, мес.(час.) включает в себя машинное время работы над проектом (Тмрп);

Кд - коэффициент дополнительной заработной платы разработчика.

Зот = 98 000\*4(1+0,17) = 458 640 руб.

* + 1. Затраты по страховым вносам (Зст), руб.

Сумма страховых взносов определяется по формуле:

Зст = Кст \* Зот, руб (7)

где Кст - коэффициент страховых взносов для расчета отчислений во внебюджетные фонды.

Зст = 0,3\*458 640= 137 592 руб.

* + 1. Затраты по содержанию ЭВМ (Зэвм), руб

Затраты, связанные с эксплуатацией и содержанием ЭВМ, определяются по формуле:

Зэвм = Тмрп \* Кгт \* Q\* Cм/ч, руб. (8)

где Тмрп - машинное время на разработку проектного решения, час;

Кгт - коэффициент готовности техники;

Q - количество условных единиц, используемой техники;

Cм/ч - стоимость машино-часа, эксплуатации оборудования, руб. в час.

Так как машинное время может измеряться в месяцах, а себестоимость машино-часа за один час, то машинное время необходимо перевести в часы.

Перевод рабочего времени в часы осуществляется по формуле:

Тмрп = Тмч \* Чрд \* Тсм \* Ксм , час. (9)

где Тмч - рабочее время в месяцах;

Чрд - число рабочих дней в месяце;

Тсм - продолжительность рабочей смены;

Ксм - количество рабочих смен.

Тмрп = 3,5\*21\*8\*1= 588 час

Зэвм = 588\*0,97\*1\*10= 5 703,60 руб.

* + 1. Затраты на расходные материалы (Зрм), руб.

Затраты на расходные материалы необходимые для разработки проектного решения определяются по формуле:

Зрм = , руб (11)

затраты на расходные материалы, руб.;

Кт - коэффициент транспортных расходов.

Зрм = 10 439,44\*(1+0,16) = 12 109, 75руб.

* + 1. Затраты по накладным расходам (Знр)

Затраты по накладным расходам определяются по формуле:

Знр = Кнр\*Зот, руб. (12)

где Кнр - коэффициент накладных расходов, принимается для расчета по данным предприятия;

Зот - затраты по оплате труда, руб.

Знр = 0,52\* 458 640= 238 492,8 руб.

Таблица 6 – Результаты вычислений

| **Наименование** | **Обозначение** | **Значение** |
| --- | --- | --- |
| Оплата труда | Зот | 458 640 руб.; |
| Страховые взносы | Зст | 137 592 руб.; |
| Содержание ЭВМ | Зэвм | 5 703,60 руб.; |
| Затраты на специальные программы | Зсп | 46 699 руб.; |
| Расходные материалы | Зрм | 12 109,75 руб.; |
| Накладные расходы | Знр | 238 492,80 руб.; |
| Итого затрат на разработку | Зрп | 899 237,15 руб. |

Произведя вычисления было выявлено, что полные затраты на разработку составляют 899 237,15 рублей. Промежуточные результаты вычислений представлены в таблице 6.

* 1. Расчет затрат на внедрение

Исходные данные, связанные с разработкой программного продукта приведены в таблице 7.

Таблица 7 – Исходные данные

| **Наименование** | **Обозначение** | **Значение** |
| --- | --- | --- |
| Оклад инженера по внедрению | Ор | 36000 руб. |
| Время необходимое для внедрения | Трп | 2 месяца |
| Коэффициент дополнительной заработной платы | Кд | 0,17 |
| Коэффициент страховых взносов | Кст | 0,3 |
| Затраты на разработку | Зрп | 899 237,15 руб. |
| Себестоимость содержания техники | См/ч | 10 руб./час |
| Коэффициент готовности техники | Кгт | 0,97 |
| Число рабочих дней в месяце | ЧРД | 21 день |
| Продолжительность смены | Тст | 8 часов |
| Коэффициент сменности | Ксм | 1 |
| Коэффициент накладных расходов | Кнр | 0,52 |

Затраты на внедрение являются для организации единовременными и формируют величину капиталовложений в данный проект.

Затраты на внедрение проектного решения (Квпр) рассчитываются по формуле:

Квпр= Зпо +Зотв +Зст +Зэвм +Знр, руб. (13)

где Зпо - затраты на приобретение программного обеспечения (включают стоимость разработанного ПП, а также других существующих ПП, необходимых для функционирования системы), руб.;

Зотв - затраты на оплату туда работников, занятых внедрением проекта, руб.;

Зст - отчисления во внебюджетные фонды с заработной платы работников, занятых внедрением проекта, руб.;

Зэвм - затраты, связанные с эксплуатацией ЭВМ при внедрении проектного решения, руб.;

Знр - накладные расходы, руб.

* + 1. Затраты на оплату работников, занятых внедрением проекта (Зот)

Затраты на оплату труда работников, занятых внедрением разработанного проекта, включают в себя заработную плату сотрудников, участвующих непосредственно в установке и отладке системы.

Зотв= Окл. Разработчика\*Те\*(1+Кд), руб. (14)

Те - трудоемкость внедрения, т.е. время необходимое для внедрения проекта, час., дни, мес.

Кд - коэффициент дополнительной заработной платы.

Зотв = 36 000\*2\*(1+0,17) = 82 800 руб.

* + 1. Затраты по страховым взносам (Зст)

Данный вид затрат определяется аналогично страховым вносам, учитываемым при расчете затрат на разработка программного продукта по формуле 7.

Зст = 0,3\*84 240= 25 272 руб.

* + 1. Затраты по содержанию ЭВМ (Зэвм), руб

Затраты, связанные с эксплуатацией и содержанием ЭВМ, определяются аналогично затратам по содержанию ЭВМ при разработке программного продукта по формуле 8.

Тмрп = 2\*21\*8\*1= 336 час

Зэвм = 336\*0,97\*1\*10= 3 259,20 руб.

* + 1. Затраты по накладным расходам (Знр)

Затраты по накладным расходам определяются по формуле:

Знр = КнрЗотв, руб (15)

где Кнр - коэффициент накладных расходов

Зотв - затраты по оплате труда работников, занятых внедрением.

Знр = 0,52\*84 240= 43 804,8 руб.

Таблица 8 – Результаты вычислений

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Обозначение** | **Значение** |
| Затраты на приобретение программного обеспечение | Зпо | 899 237,15 руб |
| Оплата труда | Зотв | 84 240 руб. |
| Страховые взносы | Зст | 25 272 руб. |
| Содержание ЭВМ | Зэвм | 3 259,20 руб. |
| Накладные расходы | Знр | 43 804,80 руб. |
| Итого затрат на внедрение | Зрпв | 1 055 813,15 руб. |

Произведя вычисления было выявлено, что полные затраты на внедрение составляют 1 055 813,15 рублей. Промежуточные результаты вычислений представлены в таблице 8.

* 1. Расчет цены и прибыли

Исходные данные, связанные с разработкой программного продукта приведены в таблице 9.

Таблица 9 – Исходные данные

| **Наименование** | **Обозначение** | **Значение** |
| --- | --- | --- |
| Коэффициент рентабельности | Кр | 0,28 |
| Коэффициент налога на добавочную стоимость | Кндс | 0,2 |
| Ставка налога | Кнп | 0,2 |
| Стоимость затрат на разработку и внедрение | Квпр | 1 055 813,15 руб. |

Цена программного продукта, который разработан одной организаций по заказу другой и не предназначен для тиражирования, определяется по формуле:

Цпп = Квпр + Ппл + НДС, руб. (16)

где Ппл - планируемая прибыль рассчитывается по формуле:

Ппл = Квпр × Кр, руб. исходя из нормативной рентабельности (коэффициента)

НДС- налог на добавленную стоимость определяется исходя из Кндс = 0.2 (ставка налога 20%) по формуле

НДС = (Квпр + Ппл)×Кндс, руб. (17)

Ппл = 1 055 813,15\*0,28= 295 627,68 руб.

НДС = (1 055 813,15+295 627,68) \* 0,2 = 270 288,17 руб.

Цпп = 1 055 813,15+295 627,68+270 288,17= 1621729 руб.

Каждое предприятие с полученной прибыли перечисляет государству налог на прибыль. На сегодня ставка налога 20% (Кнп = 0.2) от полученной прибыли, и определяется по формуле:

НП = Ппл × Кнп, руб. (18)

НП = 295 627,68\*0,2= 59 125,54 руб.

Чистая прибыль, остающаяся в распоряжении предприятия

ЧП = Ппл – НП, руб. (19)

ЧП = 295 627,68–59 125,54 = 236 502,14 руб.

Поступление в бюджет складываются из налога на прибыль и НДС

ПБ = НП + НДС, руб. (20)

ПБ = 59 125,54+270 288,17= 329 413,71 руб.

Таблица 10 – Результаты вычислений

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Обозначение** | **Значение** |
| Плановая прибыль | Ппл | 295 627,68 руб. |
| Налог на добавочную стоимость | НДС | 270 288,17 руб. |
| Цена программного продукта | Цпп | 1 621 729 руб. |
| Налог на прибыль | НП | 59 125,54 руб. |
| Чистая прибыль | ЧП | 236 502,14 руб. |
| Поступление в бюджет | ПБ | 329 413,71 руб. |

Произведя вычисления было выявлено, что чистая прибыль будет составлять 236 502,14 рублей. Промежуточные результаты вычислений представлены в таблице 10.

На основании произведенных расчетов можно сделать вывод, что на разработку продукта будет затрачено 899 273 ,15 рублей, а с учетом внедрения 1 055 813,15 руб*.* Цена готового продукта будет равна 1 621 729 руб., что в свою очередь позволит получить чистую прибыль в размере от 236 502,14 рублей.

1. **Техника безопасности и охрана труда**
   1. **Анализ условий труда программиста.**

Рабочее место программиста — это его рабочий кабинет, либо часть помещения, в котором он проводит большую часть работы. Совершенно очевидно, что чем лучше приспособлено рабочее место для выполнения функций программиста, тем производительнее и эффективнее будет его трудовая деятельность.

Программист имеет собственное рабочее место, оборудованное ПЭВМ. Рабочий стул (кресло) должен быть подъемно-поворотным и регулируемым по высоте и углам наклона сиденья и спинки, а также – расстоянию спинки до переднего края сиденья.

Рабочая поверхность стола сделана из дерева. На столе размещены ЭВМ, монитор и принтер. Площадь поверхности стола предполагает размещение монитора, устройств ввода/вывода и рабочей зоны с местом расположения оперативной документации.

Кабинет, в котором расположено рабочее место, представляет собой помещение площадью 30 м2, длиной 5 м, шириной 6 м и высотой 2,75 м.

Для создания благоприятных условий для зрительного восприятия кабинет оформлен в светло-бежевом цвете.

Для создания и поддержания независимо от наружных условий оптимальных значений температуры, влажности, чистоты воздуха, в холодное время используется водяное отопление, в теплое время года применяется кондиционирование воздуха.

Режим работы программиста - односменный с пятидневной рабочей неделей и нормированным рабочим днем. Продолжительность ежегодного отпуска - 28 календарных дней.

При идентификации вредных производственных факторов было установлено, что опасными могут являться:

* шум и вибрация, источником которого является оргтехника;
* электромагнитные поля и излучения от дисплея ПК;
* статическое электричество, накапливаемое на клавиатуре, дисплее, корпусе системного блока;
* электрический ток в электрических сетях;
* запыленность рабочей зоны, так влажная уборка рабочей поверхности проводиться не ежедневно, а только по мере загрязнения;
* статические перегрузки, так как работа программиста связана с неизменной статической позой, иногда неудобной позе (работа за ПК), что может привести к искривлению позвоночника, остеохондрозу, застою в органах малого таза;
* перенапряжение анализаторов, в большей степени зрительного, что может привести к различным нарушениям зрения (снижение остроты, спазм и уменьшение запаса аккомодации), помимо этого повышение зрительной нагрузки может приводить к общему утомлению, возникновению головной боли, ухудшению самочувствия;
* умственное перенапряжение, так как в процессе деятельности программист анализирует и перерабатывает огромное количество административных сетей, разрабатывает программные продукты, поддерживает технику в рабочем состоянии.

В целях выявления вредных и (или) опасных производственных факторов и осуществления мероприятий по приведению условий труда в соответствие государственными нормативными требованиями охраны трудаежегодно проводится аттестация рабочих мест по условиям труда - оценка условий труда на рабочих местах. Аттестации подлежат все имеющиеся в организации рабочие места.

При аттестации рабочих мест проводят оценку условий труда, гигиеническую оценку и оценку травмоопасности рабочих мест. При этом учитывают наличие средств коллективной защиты, обеспеченность работников средствами индивидуальной защиты и определяют эффективность этих средств.

Согласно санитарным правилам СП 2.2.3670-20 "Санитарно-эпидемиологические требования к условиям труда", дисплей должен располагаться на расстоянии 60-70 см, но не менее 50 см от глаз. Между боковыми поверхностями мониторов должно быть не менее 1,2 м. При использовании жидкокристаллических дисплеев на каждый компьютер должно приходиться не менее 4,5 м2 площади. На дисплей ПК не должен попадать прямой солнечный свет во избежание бликов и повышенной нагрузки на зрение.

Поэтому в кабинете, где работает программист, стоит только один компьютер, а комплектующая оргтехника расположена достаточно далеко непосредственно от самого специалиста. Это обусловлено тем, что все копировальные аппараты, принтеры и факс в периоды своей работы выделяют не только вредные токсичные газы, но и шумы и вибрацию. Данные факторы оказывают гораздо меньшее негативное влияние, нежели излучение от компьютера, но также являются вредными и могут нанести вред здоровью сотрудника.

Для снижения напряжения с глаз, а также для снятия мышечного напряжения во время рабочего дня проводится проветривание в течении 10 – 15 минут каждые 1,5 часа.

* 1. **Расчет искусственного освещения рабочего места программиста.**

Через зрительный анализатор человек получает около 80% из общего объема информации. Качество поступающей информации во многом зависит от освещения: неудовлетворительное количественно или качественно, оно не только утомляет зрение, но и вызывает утомление организма в целом. Кроме того, нерациональное освещение может явиться причиной травматизма: плохо освещенные опасные зоны, слепящие источники света и блики от них, резкие тени ухудшают видимость настолько, что вызывают полную потерю ориентировки работающих, снижают производительность труда и увеличивают брак продукции. Поэтому необходимо достаточное искусственное освещение для рабочего места.

Условия работы для расчета показателя освещенности рабочего места программиста являются:

* Помещение размером 30 квадратных метров;
* Высота помещения 2,75 метра, длина 5 метра, ширина 6 метров;
* Лампы люминесцентные (дневного света) в количестве 4 штук, новые;
* Крупногабаритная мебель отсутствует.

Расчет общего равномерного искусственного освещения горизонтальной рабочей поверхности выполняется методом коэффициента использования светового потока. Световой поток (лм) одной лампы рассчитывается по следующей формуле:



где Ен - нормируемая минимальная допустимая освещенность по СП 1.3330.2016, для помещения 2000 лк;

S - площадь освещаемого помещения, м2;

z - коэффициент неравномерности освещения, z = 1,1;

K - коэффициент запаса, учитывающий снижение освещенности в процессе эксплуатации вследствие загрязнения и старения ламп и светильников, а также снижения отражающих свойств поверхностей помещения для кабинета будет равен 1,3;

Nc - число светильников в помещении;

- коэффициент затенения, обычно 1;

=i - коэффициент использования светового потока.

Коэффициент использования светового потока, давший название методу расчета, определяется по индексу помещения i в зависимости от типа светильника и коэффициентов отражения света от потолка, стен и пола:



где А, В - длина и ширина помещения в плане, м;

Нс - высота подвеса светильников над рабочей поверхностью, м.

Согласно СП 1.3330.2016 величина полученного светового потока достаточно для данного помещения.

* 1. **Электробезопасность на предприятии.**

При поступлении на работу сотрудник должен пройти инструктаж по технике безопасности и электробезопасности. Работника знакомят с основными правилами по технике безопасности, предлагают внимательно прочитать действующие на предприятии инструкции, поясняя при этом отдельные правила и требования.

Инструктаж по технике безопасности при выполнении конкретной работы проводит руководитель соответствующего производственного участка, показывая инструктируемому правильные безопасные приемы работы.

Повторный инструктаж проводится для рабочих независимо от их квалификации, стажа и опыта работы, не реже 1 раза в 6 месяцев по программе инструктажа на рабочем месте.

При нарушении работающими правил и инструкций по технике безопасности, технологической и производственной дисциплины, а также в случаях изменения технологического процесса или вида работы проводят дополнительные инструктажи.

Все инструктажи оформляются записями в специальном журнале с указанием номеров или шифров инструкций. Журнал о проведении инструктажа хранится у руководителя подразделения.

Работник на рабочем месте должен соблюдать общие меры электробезопасности. При этом запрещается:

* прикасаться к задней панели системного блока при включенном питании;
* допускать попадание влаги на поверхность системного блока, монитора, рабочую поверхность клавиатуры, дисководов, принтеров и др. устройств;
* производить самостоятельно вскрытие и ремонт оборудования;
* пользоваться неисправными электроприборами и электропроводкой;
* ремонтировать электроприборы самостоятельно;
* подвешивать электропровода на гвоздях, металлических и деревянных предметах, перекручивать провод, закладывать провод и шнуры на водопроводные трубы и батареи отопления, вешать что-либо на провода, вытягивать за шнур вилку из розетки;
* прикасаться одновременно к персональному компьютеру и к устройствам, имеющим соединение с землей (радиаторы отопления, водопроводные краны, трубы и т.п.);
* пользоваться самодельными электронагревательными приборами и электроприборами с открытой спиралью;
* наступать на переносимые электрические провода, лежащие на полу.

При работе с электроприборами и оргтехникой (персональные компьютеры, принтеры, сканеры, копировальные аппараты, факсы, бытовые электроприборы, приборы освещения):

* автоматические выключатели и электрические предохранители должны быть всегда исправны.
* Изоляция электропроводки, электроприборов, выключателей, штепсельных розеток, ламповых патронов и светильников, а также шнуров, с помощью которых включаются в электросеть электроприборы, должны быть в исправном состоянии.
* Электроприборы необходимо хранить в сухом месте, избегать резких колебаний температуры, вибрации, сотрясений.
* Для подогрева воды пользоваться сертифицированными электроприборами с закрытой спиралью и устройством автоматического отключения, с применением несгораемых подставок.
  1. **Пожарная безопасность на предприятии**

Инструктаж по пожарной безопасности проводится по программе, разработанной инженером по охране пожарной безопасности. Продолжительность инструктажа устанавливается в соответствии с утвержденной программой. Инструктаж по пожарной безопасности, как правило, проводится совместно с инструктажем по технике безопасности и в те же сроки.

Инструктаж по пожарной безопасности проходят все работники организации, независимо от их образования, стажа работы по данной профессии или должности, временные работники, командированные, обучающиеся и студенты, прибывшие на практику.

О проведении инструктажа по пожарной безопасности работник, проводивший инструктаж, делает запись в журнале учета проведения инструктажей по пожарной безопасности, с обязательной подписью инструктируемого и инструктирующего. При регистрации внепланового инструктажа по пожарной безопасности указывают причину его проведения.

Все работники организаций должны допускаться к работе только после прохождения противопожарного инструктажа, а при изменении специфики работы проходить дополнительное обучение предупреждению и тушению возможных пожаров в порядке, установленном руководителем.

На проведение первичного противопожарного инструктажа необходимо отводить не менее 1 ч. Инструктируемые должны ознакомиться:

* с действующими на объекте правилами пожарной безопасности и инструкциями;
* с производственными участками, наиболее опасными в пожарном отношении, где запрещается курить, применять открытый огонь;
* с возможными причинами возникновения пожара и мерами его предупреждения;
* с практическими действиями в случае возникновения пожара - вызов пожарной помощи, использование первичных средств пожаротушения, место расположения ближайшего телефона и ознакомление с правилами поведения в случае возникновения пожара, эвакуации людей и материальных ценностей.

При первичном инструктаже инструктирующий обязан рассказать о производственных установках с повышенной пожарной опасностью, мерах предотвращения пожаров и загораний, указать место курения, ознакомить вновь поступившего с имеющимися на объекте средствами пожаротушения, показать ближайший телефон и объяснить правила поведения в случае возникновения пожара.

Проведение противопожарного инструктажа в обязательном порядке должно сопровождаться практическим показом способов использования имеющихся на объекте средств пожаротушения (противогазы, респираторы, огнетушители и т.д.).

Соблюдение рассмотренных в данном разделе мероприятий по охране труда и технике безопасности позволяет снизить утомляемость и травматизм, повысить производительность труда, обеспечить комфортные условия трудовой деятельности специалиста, работающего в должности программиста.

# ЗАКЛЮЧЕНИЕ

В данной работе был разработан и реализован проект информационной системы для складских нужд, а также рассмотрены методы автоматизации процесса введения складской работы. Был разработан простой и понятный, при эксплуатации пользователем интерфейс.

При разработке были исследованы потенциальные проблемы и произведен анализ аналогичных существующих решений, было приведено обоснование предлагаемого решения.

В экономической части приведены расчеты экономического эффекта, который может быть достигнут при внедрении разработанного программного обеспечения.

В разделе «Техника безопасности и охрана труда» были обоснованы инженерно-технические мероприятия по обеспечению безопасности и охране труда при использовании разработанного программного продукта на вычислительной технике.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Алекс Дэвис. Асинхронное программирование в C# 5.0 [Текст] / Алекс Дэвис. - М.: Библиотека программиста, 2022. - 122 с.
2. Арора Гаурав. Паттерны проектирование для C# и платформы .NET Core [Текст] / Арораа Гаурав, Чилберто Джеффри. - М.: Библиотека программиста, 2021. - 352 с.
3. Гагарина, Л.Г. Технология разработки программного обеспечения: учебное пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул; под ред. Л.Г. Гагариной. – Москва: ИД «ФОРУМ»: ИНФРА-М, 2017. – 400 с.
4. Грин Дженнифер. Head Firs изучаем С# [Текст] / Грин Дженнифер, Стиллмен Эндрю. - М.: Библиотека программиста, 2022. - 769 с.
5. Джон Скит. С# программирование для профессионалов [Текст] / Джон Скит. - М.: Библиотека программиста, 2022. - 351 с.
6. Зверева, В.П. Сопровождение и обслуживание программного обеспечения компьютерных систем: учебное пособие / В.П. Зверева, А.В. Назаров. – М: Академия, 2018. – 254 с.
7. Крис Дейт. Введение в системы баз данных [Текст]: Крис Дейт. - М.: Вильямс, 2018. - 1328 с.
8. Мюллер Джон. С# для чайников / Мюллкр Джон, Семпф Билл. -М: Диалектика-Вильямс, 2019. – 608 с.
9. Е. П. Моргунов. PostgreSQL Основы языка SQL [Текст] / Е. П. Моргунов. - М.: БХВ-Петербург, 2019. - 336 с.
10. Пселтис Эндрю. Потоковая обработка данных [Текст] /Пселтис Эндрю. - М.: ДМК Пресс, 2018. - 218 с.
11. Прайс Марк. С# 7 и NET Core. Кросс-платформенная разработка для профессионалов [Текст] / Прайс Марк. - М.: Библиотека программиста, 2019. - 640 с.
12. Петцольд, Ч. Программирование для Microsoft Windows 8: разработка приложений для Windows Store на C# и XAML / Чарльз Петцольд; [пер. с англ. Е. Матвеев]. – 6-е изд. – М [и др.]: Питер, 2014. – 1008 с.
13. Санжей Мишра, Секреты Oracle SQL. [Текст] / Снажей Мишра, Алан Бьюли - М.: Сивол-Плюс, 2018. - 368 с.
14. Троелсен Э. С# 7 и NET Core. Кросс-платформенная разработка для профессионалов [Текст] / Троелсен Э, Джепикс Ф. - М.: Библиотека программиста, 2022. - 1392 с.
15. Федорова, Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности: учебное пособие / Г.Н. Федорова. – М: Курс: ИНФРА-М, 2016. – 332 с.
16. Mentanit.com C# [Электронный ресурс] / Режим доступа: URL: https://metanit.com/sharp/
17. Работа с базой данных в C# [Электронный ресурс] / Режим доступа: URL: https://it-black.ru/rabota-s-bazoj-dannyh-v-ci\_sharp/
18. Документация по С# [Электронный ресурс] / Режим доступа: URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/
19. Начало работы в Visual Studio [Электронный ресурс] / Режим доступа: URL: https://metanit.com/sharp/tutorial/1.2.php
20. Руководство по С# [Электронный ресурс] / Режим доступа: URL: https://professorweb.ru/my/csharp/charp\_theory/level1/index.php

# Приложение А

(справочное)

**Исходный код**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

//Форма входа;

namespace FG

{

public partial class Вход : Form

{

Hash hash = new Hash();

DataBase dataBase = new DataBase();

Role role = new Role();

static int passwordShot = 0;

public Вход()

{

InitializeComponent();

textBox2.PasswordChar = '\*';

}

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

{

if (textBox1.Text != "" && textBox2.Text != "")

{

if (passwordShot < 2)

{

string login = textBox1.Text.Trim();

string password = hash.getHash(textBox2.Text.Trim());

SqlDataAdapter adapter = new SqlDataAdapter();

DataTable table = new DataTable();

string queryString = $"select логин, пароль from Пользователи where логин = '{login}' and пароль = '{password}'";

SqlCommand command = new SqlCommand(queryString, dataBase.getConnection());

//string user;

//user = hash.getHash(textBox2.Text.Trim());

adapter.SelectCommand = command;

adapter.Fill(table);

if (table.Rows.Count == 1)

{

GetLog.login = login;

role.getRol(login, password);

MessageBox.Show("Вход выполнен успешно!");

Hide();

}

else

{

passwordShot++;

MessageBox.Show("Неправильный логин или пароль");

}

}

else

{

timer1.Enabled = true;

}

}

else

{

MessageBox.Show("Введите логин и пароль");

}

}

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

{

Регистрация рег = new Регистрация();

рег.Show();

Hide();

}

int tik = 10;

private void timer1\_Tick(object sender, EventArgs e)

{

passwordShot = 0;

tik--;

if (tik > 0)

{

button2.Enabled = false;

}

else

{

button2.Enabled = true;

timer1.Enabled = false;

tik = 10;

}

}

}

}

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

//Главная страница

namespace FG

{

public partial class Таблицы : Form

{

DataBase dataBase = new DataBase();

public Таблицы()

{

InitializeComponent();

}

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

{

if (ttexBox.Text != "" && ttexBox2.Text != "" && ttexBox3.Text != "" && ttexBox4.Text != "" && ttexBox5.Text != "" && tcomboBox1.Text != "" && tcomboBox2.Text != "")

{

string Tname = ttexBox.Text.Trim();

string firm = ttexBox2.Text.Trim();

string country = ttexBox3.Text.Trim();

string ed = tcomboBox1.SelectedValue.ToString();

string tg = tcomboBox2.SelectedValue.ToString();

string cost = ttexBox4.Text.Trim();

string kall = ttexBox5.Text.Trim();

string queryString = $"insert into Товары(наименование\_товара, фирма, страна, ед\_изм, товарная\_группа, цена, количество) values('{Tname}', '{firm}', '{country}', '{ed}', '{tg}', '{cost}', '{kall}')";

SqlCommand command = new SqlCommand(queryString, dataBase.getConnection());

dataBase.openConnection();

if (command.ExecuteNonQuery() == 1)

{

MessageBox.Show("Товар добавлен");

ttexBox.Clear();

ttexBox2.Clear();

ttexBox3.Clear();

ttexBox4.Clear();

ttexBox5.Clear();

UpdateMet();

}

else

{

MessageBox.Show("Ошибка добавления!");

}

dataBase.closeConnection();

}

else

{

MessageBox.Show("Заполните пустые поля \n в добавлении товара");

}

}

GetLog getLog = new GetLog();

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

{

if (pcomboBox1.Text != "" && ptextBox2.Text != "" && ptextBox3.Text != "" && ptextBox4.Text != "")

{

string Tname = pcomboBox1.SelectedValue.ToString();

string namber = ptextBox2.Text.Trim();

string kall = ptextBox3.Text.Trim();

string cost = ptextBox4.Text.Trim();

string login = "asd";

login = getLog.GetUserID(login);

//login = getLog.GetUserID(GetLog.login);

string queryString = $"insert into Приход(id\_товара, номер\_тн, id\_принявшего\_товар, дата\_прихода, количество\_приход, стоимость\_приход) values('{Tname}', '{namber}', '{login}', '{dateTimePicker1.Value}', '{kall}', '{cost}')";

SqlCommand command = new SqlCommand(queryString, dataBase.getConnection());

dataBase.openConnection();

if (command.ExecuteNonQuery() == 1)

{

MessageBox.Show("Товар добавлен");

ptextBox2.Clear();

ptextBox3.Clear();

ptextBox4.Clear();

UpdateMet();

}

else

{

MessageBox.Show("Ошибка добавления!");

}

dataBase.closeConnection();

}

else

{

MessageBox.Show("Заполните пустые поля \n в приходно накладной");

}

}

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

{

if (cb1.Text != "" && tb2.Text != "" && tb3.Text != "" && tb4.Text != "")

{

string Tname = cb1.SelectedValue.ToString();

string namber = tb2.Text.Trim();

string kall = tb3.Text.Trim();

string cost = tb4.Text.Trim();

string login = "asd";

login = getLog.GetUserID(login);

//login = getLog.GetUserID(GetLog.login);

string queryString = $"insert into Расход(id\_товара, номер\_тн, id\_сдавшего\_товар, дата\_расхода, количество\_рас, цена\_рас) values('{Tname}', '{namber}', '{login}', '{dateTimePicker2.Value}', '{kall}', '{cost}')";

SqlCommand command = new SqlCommand(queryString, dataBase.getConnection());

dataBase.openConnection();

if (command.ExecuteNonQuery() == 1)

{

MessageBox.Show("Товар отправлен");

tb2.Clear();

tb3.Clear();

tb4.Clear();

UpdateMet();

}

else

{

MessageBox.Show("Ошибка отправления!");

}

dataBase.closeConnection();

}

else

{

MessageBox.Show("Заполните пустые поля \n в расходно накладной");

}

}

private void Таблицы\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "складDataSet1.Товары". При необходимости она может быть перемещена или удалена.

this.товарыTableAdapter.Fill(this.складDataSet1.Товары);

this.товарные\_группыTableAdapter.Fill(this.складDataSet1.Товарные\_группы);

this.единицы\_измеренияTableAdapter.Fill(this.складDataSet.Единицы\_измерения);

this.единицы\_измеренияTableAdapter.Fill(this.складDataSet.Единицы\_измерения);

}

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

{

try

{

this.единицы\_измеренияTableAdapter.FillBy(this.складDataSet.Единицы\_измерения);

}

catch (System.Exception ex)

{

System.Windows.Forms.MessageBox.Show(ex.Message);

}

}

public void UpdateMet()

{

this.товарыTableAdapter.Fill(this.складDataSet1.Товары);

this.товарные\_группыTableAdapter.Fill(this.складDataSet1.Товарные\_группы);

this.единицы\_измеренияTableAdapter.Fill(this.складDataSet.Единицы\_измерения);

this.единицы\_измеренияTableAdapter.Fill(this.складDataSet.Единицы\_измерения);

}

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

{

if (textBox10.Text != "" && textBox8.Text != "")

{

string name = textBox10.Text.Trim();

string coment = textBox8.Text.Trim();

string queryString = $"insert into Расход(id\_товара, номер\_тн, id\_сдавшего\_товар, дата\_расхода, количество\_рас, цена\_рас) values('{name}', '{coment}')";

SqlCommand command = new SqlCommand(queryString, dataBase.getConnection());

dataBase.openConnection();

if (command.ExecuteNonQuery() == 1)

{

MessageBox.Show("Товар отправлен");

tb2.Clear();

tb3.Clear();

tb4.Clear();

UpdateMet();

}

else

{

MessageBox.Show("Ошибка отправления!");

}

dataBase.closeConnection();

}

else

{

MessageBox.Show("Заполните пустые поля в Товарной группе");

}

}

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

{

if (textBox6.Text != "" && textBox7.Text != "")

{

string name = textBox6.Text.Trim();

string coment = textBox7.Text.Trim();

string queryString = $"insert into Расход(id\_товара, номер\_тн, id\_сдавшего\_товар, дата\_расхода, количество\_рас, цена\_рас) values('{name}', '{coment}')";

SqlCommand command = new SqlCommand(queryString, dataBase.getConnection());

dataBase.openConnection();

if (command.ExecuteNonQuery() == 1)

{

MessageBox.Show("Товар отправлен");

tb2.Clear();

tb3.Clear();

tb4.Clear();

UpdateMet();

}

else

{

MessageBox.Show("Ошибка отправления!");

}

dataBase.closeConnection();

}

else

{

MessageBox.Show("Заполните пустые поля в Еденице измерения");

}

}

//Открытие таблицы ТОВАРЫ

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

{

if (radioButton1.Checked)

{

FormOpen.logik = true;

Товары товар = new Товары();

товар.Show();

Hide();

}

else

{

FormOpen.logik = false;

Товары товар = new Товары();

товар.Show();

}

}

//Открытие таблицы ПРИХОД

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

{

if (radioButton1.Checked)

{

FormOpen.logik = true;

Приход товар = new Приход();

товар.Show();

Hide();

}

else

{

FormOpen.logik = false;

Приход товар = new Приход();

товар.Show();

}

}

//Открытие таблицы РАСХОД

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

{

if (radioButton1.Checked)

{

FormOpen.logik = true;

Расход товар = new Расход();

товар.Show();

Hide();

}

else

{

FormOpen.logik = false;

Расход товар = new Расход();

товар.Show();

}

}

//Открытие таблицы ЕДЕНИЦЫ ИЗМЕРЕНИЯ

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

{

if (radioButton1.Checked)

{

FormOpen.logik = true;

Еденицы\_измерения товар = new Еденицы\_измерения();

товар.Show();

Hide();

}

else

{

FormOpen.logik = false;

Еденицы\_измерения товар = new Еденицы\_измерения();

товар.Show();

}

}

//Открытие таблицы ТОВАРНЫЕ ГРУППЫ

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

{

if (radioButton1.Checked)

{

FormOpen.logik = true;

Товарная\_группа товар = new Товарная\_группа();

товар.Show();

Hide();

}

else

{

FormOpen.logik = false;

Товарная\_группа товар = new Товарная\_группа();

товар.Show();

}

}

//Открытие таблицы ПОЛУЧАТЕЛИ

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

{

if (radioButton1.Checked)

{

FormOpen.logik = true;

Получатели товар = new Получатели();

товар.Show();

Hide();

}

else

{

FormOpen.logik = false;

Получатели товар = new Получатели();

товар.Show();

}

}

}

}