**Содержание**

[Введение 4](#_Toc136070537)

[1 Общая часть 6](#_Toc136070538)

[1.1 Структура предприятия, место и функции информационных отделов 6](#_Toc136070539)

[1.2 Архитектура имеющихся технических средств и их основные параметры 7](#_Toc136070540)

[1.3 Используемое программное обеспечение 8](#_Toc136070541)

[2 Специальная часть 10](#_Toc136070542)

[2.1 Основание разработки 10](#_Toc136070543)

[2.2 Постановка задачи 11](#_Toc136070544)

[2.2.1 Построение модели базы данных 11](#_Toc136070545)

[2.2.2 Проектирование функциональных задач 15](#_Toc136070546)

[2.2.3 Спецификация входных и выходных данных 16](#_Toc136070547)

[2.3 Реализация программы 19](#_Toc136070548)

[2.3.1 Анализ средств разработки 19](#_Toc136070549)

[2.3.2 Разработка интерфейса программы 23](#_Toc136070550)

[2.3.3 Проектирование основного и вспомогательного алгоритмов 27](#_Toc136070551)

[2.3.4 Разработка программных модулей 31](#_Toc136070552)

[2.4 Тестирование приложения 34](#_Toc136070553)

[3 Организационно-экономическая часть 37](#_Toc136070554)

[3.1 Расчет сметы затрат на разработку программы 37](#_Toc136070555)

[3.1.1 Составление проекта выполнения работ 37](#_Toc136070556)

[3.1.2 Расчет материальных затрат 37](#_Toc136070557)

[3.1.3 Расчет затрат на оплату труда 39](#_Toc136070558)

[3.1.4 Расчет амортизационных отчислений 42](#_Toc136070559)

[3.1.5 Расчет стоимости разрабатываемой программы 43](#_Toc136070560)

[3.2 Расчет экономической эффективности внедрения программы 46](#_Toc136070561)

[4 Мероприятия по охране труда и окружающей среды 49](#_Toc136070562)

[4.1 Техника безопасности при работе с вычислительной техникой 49](#_Toc136070563)

[Заключение 55](#_Toc136070564)

[Библиография 56](#_Toc136070565)

# Введение

Во многих организациях учет большого объема данных ведется, в основном, в бумажном виде, что является очень трудоемким и неудобным процессом, для упрощения которого используют так называемые информационные системы.

Использование информационных систем в деятельности любой организации, где необходимо вести учет огромного количества информации, позволит выполнить следующие задачи:

* вести отчетность по данным в электронном виде;
* повысить скорость обработки данных;
* создать регистры по типам документации;
* сэкономить время на поиске важных документов или необходимой информации;
* уменьшить возникновение ошибок при заполнении документов.

Информационная система (ИС) — это система, предназначенная для хранения, поиска и обработки информации, и соответствующие организационные ресурсы (человеческие, технические, финансовые и т. д.), которые обеспечивают и распространяют информацию.

Основой любой информационной системы является база данных.

База данных (БД) — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных. Любую базу данных можно создать при помощи систем управления базами данных.

Во время прохождения практики было дано задание, которое заключалось в разработке системы, которая позволит вести учет данных об изделиях в электронном виде и осуществлять различные операции с ними.

В соответствии с заданием, полученным на практике, была определена цель дипломного проекта: «Создание информационной системы учета изделий».

Исходя из цели работы, был сформулирован следующий круг задач:

1. Изучить требования заказчика и провести анализ предметной области – производство изделий и их учет.
2. Изучить техническую литературу и интернет-ресурсы.
3. Построить модель базы данных.
4. Построить пользовательский интерфейс информационной системы.
5. Разработать функциональные возможности информационной системы.
6. Оформить техническую документацию в виде пояснительной записки к дипломному проекту.

# 1 Общая часть

## 1.1 Структура предприятия, место и функции информационных отделов

ООО ПКФ «Гофра-Пак» является коммерческой организацией, созданной в организационно-правовой форме общества с ограниченной ответственностью, в соответствие с действующим законодательством Российской Федерации. Общество является юридическим лицом и действует на основании законодательства РФ и устава.

Основной целью организации является получение прибыли в интересах акционеров Общества.

Предметом деятельности общества является насыщение потребительского рынка товарами и услугами.

Общество в установленном законом порядке осуществляет производство и реализацию продукции из гофрокартона.

Организационная структура предприятия изображена на схеме 1.

Бригадир

Директор

Бухгалтер

Секретарь

Начальник цеха

Менеджер по сбыту

Программист

Грузчик

Клеевар

Оператор КУ

Оператор линии

Собрание учредителей ООО ПКФ «Гофра-Пак»

Схема 1 – Организационная структура предприятия

На предприятии IT-отдела нет, но имеется наемный программист, работающий по договору аутсорсинга, который выполняет следующие обязанности:

1. Установка требуемого программного обеспечения на компьютеры организации.
2. Изучение соответствующей технической документации в сфере своей компетенции.
3. Тестирование программных решений.
4. Проведение отладки программных продуктов.
5. Сопровождение программных продуктов в ходе их использования.
6. Контроль над обновлением установленного программного обеспечения.
7. Мониторинг ситуации с вирусами и потенциально вредными файлами.
8. Установка требуемых параметров (пароли, права доступа и пр.) для работы локальной сети учреждения.
9. Мониторинг корректности работы используемых серверов.
10. Консультирование руководителей и коллег по инженерно-программным вопросам.

## 1.2 Архитектура имеющихся технических средств и их основные параметры

Компьютер – модульный прибор, состоящий из различных устройств, каждое из которых выполняет определенные задачи. В общем плане, это электронное устройство, которое выполняет операции ввода информации, хранения и обработки ее по определенной программе, вывод полученных результатов в форме, пригодной для восприятия человеком.

Аппаратные средства (Hardware) – совокупность всех устройств, которые составляют компьютер или могут к нему добавляться по мере необходимости.

Техническое средство (ТС) – обобщенное название объектов, с помощью которых человек реализует определенную потребительскую, производственную, эксплуатационную либо познавательную задачу в рамках той или иной технологии.

Состав технических средств в ООО ПКФ «Гофра-Пак»:

1. Системный блок (3 шт.) - процессор Intel Pentium Gold G6400, материнская плата GIGABYTE H410M H, ОЗУ DDR4 Patriot Signature Line 4Gb 2 шт., жесткий диск Toshiba P300 с 1 Тб памяти.
2. Монитор (3 шт.) - DELL E2420H 23.8 1920x1080@60 Гц, IPS.
3. Коммутатор - Hikvision DS-3E0510P-E/M.
4. Принтер (2 шт.) - Pantum M6500.

## 1.3 Используемое программное обеспечение

Совокупность программ и сопровождающей их документации, предназначенная для решения задач на ПК, называется программным обеспечением.

В организации ООО ПКФ «Гофра-Пак» используются следующие программные средства:

1. LibreOffice - кроссплатформенный, свободно распространяемый офисный пакет, содержащий в себе текстовый и табличный процессор, программу для подготовки и просмотра презентаций, векторный графический редактор, систему управления базами данных и редактор формул.
2. «1С: Предприятие» - программный продукт компании 1C, предназначенный для автоматизации деятельности на предприятии.
3. «1С: Бухгалтерия» - программный продукт для автоматизации бухгалтерской деятельности.
4. Интернет-браузер «Яндекс» - позволяет искать нужную информацию во всемирной сети Интернет.
5. Антивирусное ПО «Kaspersky Internet Security» - программный продукт, который обеспечивает защиту в реальном времени не только от вредоносных программ, а и от хакерских атак из сети, спама и прочих Интернет-угроз; обеспечивает безопасную навигацию по сети Интернет, проверяет степень безопасности программ и сайтов и пр.
6. «WinRAR» - это архиватор файлов в формат RAR и ZIP для 32-битных и 64-разрядных операционных систем Windows.
7. «ABBYY FineReader» - программа для распознавания текста, которая позволяет с максимальной точностью переводить документы и PDF-файлы в цифровой формат с возможностью дальнейшего редактирования текстов.

# 2 Специальная часть

## 2.1 Основание разработки

Студент 4 курса Медведев В.А. проходил производственную практику в ООО ПКФ «Гофра-Пак».

Основанием для разработки информационной системы учета изделий является задание, полученное во время прохождения производственной практики по профессиональным модулям ПМ.01 Разработка модулей программного обеспечения для компьютерных систем, ПМ.02 Осуществление интеграции программных модулей, ПМ.04 Сопровождение и обслуживание программного обеспечения компьютерных систем, ПМ.11 Разработка, администрирование и защита баз данных.

Руководителем дипломного проекта является преподаватель специальных дисциплин Чухарев Виталий Михайлович.

Заказчиком на разработку информационной системы учета изделий является ООО ПКФ «Гофра-Пак».

Требования заказчика заключались в создании информационной системы учета изделий с помощью, которой можно было вести электронный учет производства изделий на предприятии.

Преимуществами разработанной информационной системы учета изделий являются:

1. Уникальный и интуитивно-понятный дизайн приложения.
2. Возможность вести учет сотрудников, материалов и производимых изделий.
3. Осуществление добавления, редактирования или удаления информации.
4. Возможность осуществлять поиск, сортировку или фильтрацию данных.

Разработанная информационная система необходима для применения в коммерческих целях и будет использоваться для учета производимых предприятием изделий в электронном виде.

Созданная информационная система учета изделий предназначена для использования бухгалтером предприятия ООО ПКФ «Гофра-Пак».

## 2.2 Постановка задачи

## 2.2.1 Построение модели базы данных

Модель базы данных разработана на основе анализа требований заказчика и изучения предметной области – учет производимых на предприятии изделий.

В разработанной модели базы данных объектами являются созданные таблицы:

1. Таблица «Post» (Должность) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «name» (наименование должности).
2. Таблица «ProductType» (Тип изделия) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «name» (наименование типа изделия).
3. Таблица «Material» (Материал) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «name» (наименование материала).
4. Таблица «Product» (Изделие) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «id\_material» (идентификатор материала), «id\_productType» (идентификатор типа изделия), «name» (наименование изделия), «description» (описание изделия), «count» (количество изделий), «image» (изображение изделия).
5. Таблица «Employee» (Сотрудник) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «id\_post» (идентификатор должности), «name» (имя сотрудника), «lastname» (фамилия сотрудника), «middlename» (отчество сотрудника), «phone» (номер телефона сотрудника), «email» (электронная почта сотрудника), «birthDate» (дата рождения сотрудника), «employmentDate» (дата трудоустройства сотрудника), «image» (фото сотрудника).
6. Таблица «ProductionPlan» (План производства) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «name» (наименование плана производства), «description» (описание плана производства), «startDate» (дата старта плана производства), «finishDate» (дата завершения плана производства).
7. Таблица «ProductionPlanEmployee» (План производства и Сотрудник) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «id\_plan» (идентификатор плана производства), «id\_employee» (идентификатор сотрудника).
8. Таблица «ProductionPlanProduct» (План производства и Изделие) – данная таблица предназначена для хранения следующих данных: «id» (идентификатор), «id\_plan» (идентификатор плана производства), «id\_product» (идентификатор изделия), «productCount» (количество производимых изделий).

Вышеперечисленные объекты (таблицы) были созданы с помощью СУБД Microsoft SQL Server Management Studio и являются исходными таблицами.

На рисунке 1 представлена модель базы данных:

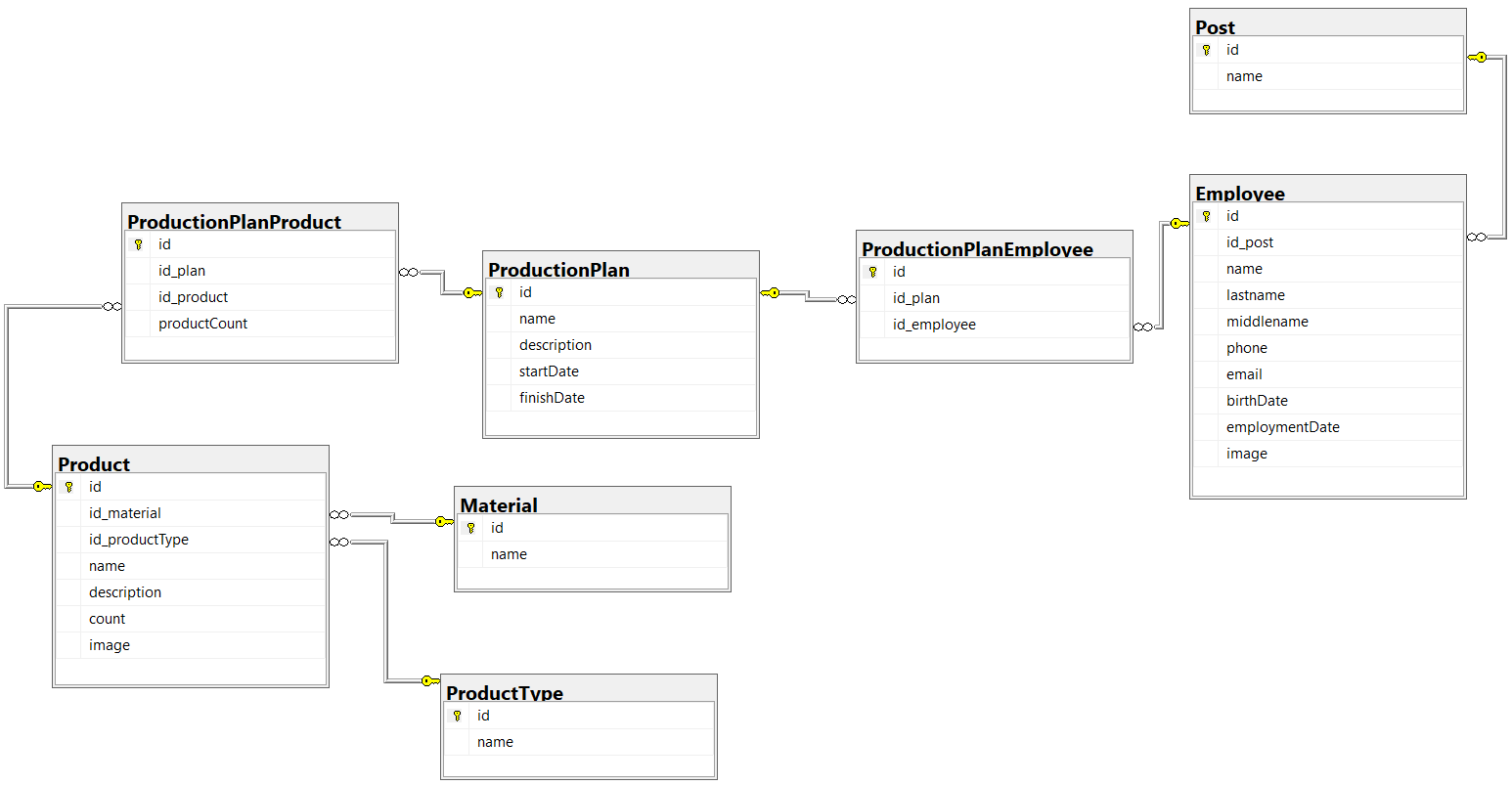


Рисунок 1 – Модель базы данных

Далее приводится описание всех отношений между сущностями базы данных с подробным их описанием.

Сущности «Post» и «Employee» связаны «один-ко-многим», так как одна должность может быть у нескольких сотрудников.

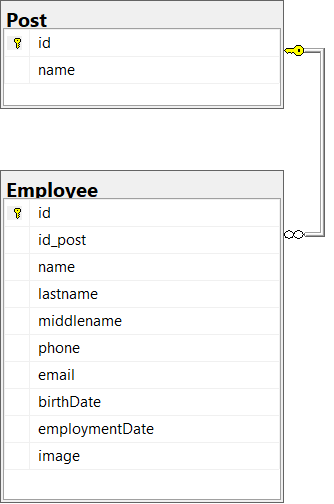


Рисунок 2 – Сущности «Post» и «Employee»

Сущности «ProductType» и «Product» связаны «один-ко-многим», так как один тип изделий может быть у нескольких изделий.

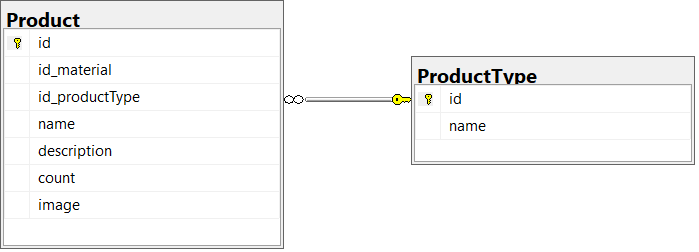


Рисунок 3 – Сущности «ProductType» и «Product»

Сущности «ProductType» и «Material» связаны «один-ко-многим», так как один материал может быть использован для производства различных изделий.

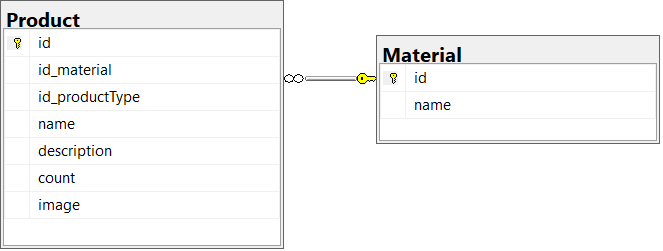


Рисунок 4 – Сущности «ProductType» и «Product»

Сущности «Product», «ProductionPlanProduct» и «ProductionPlan» связаны «многие-ко-многим», так как в одном плане производства могут быть несколько изделий одновременно.

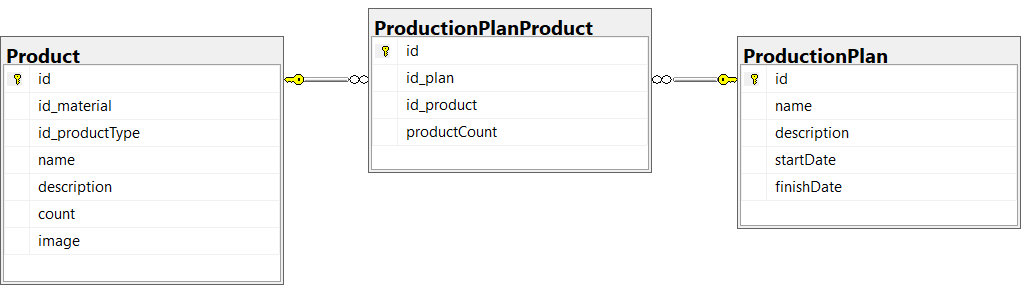


Рисунок 5 – Сущности «Product», «ProductionPlanProduct» и «ProductionPlan»

Сущности «Employee», «ProductionPlanEmployee» и «ProductionPlan» связаны «многие-ко-многим», так как в одном плане производства могут принимать участие несколько сотрудников одновременно.

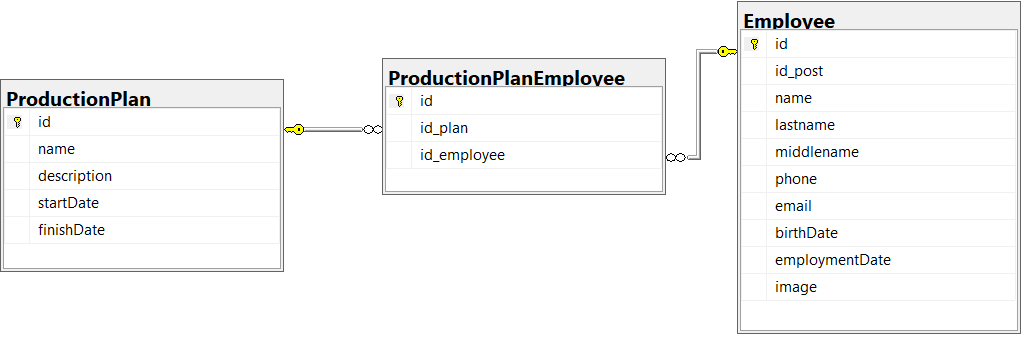


Рисунок 6 – Сущности «Employee», «ProductionPlanEmployee» и «ProductionPlan»

Созданная модель базы данных относится к третьей нормальной форме (3НФ) так как:

* находится во второй нормальной форме (2НФ);
* не имеет транзитивных зависимостей.

## 2.2.2 Проектирование функциональных задач

Созданная информационная система учета изделий обладает следующими функциональными возможностями:

* добавление новых данных, а также изменение и удаление существующих;
* поиск, сортировка и фильтрация данных;
* вывод отчетной информации: списки должностей, сотрудников, материалов, типов изделий, изделий и планов производства.

## 2.2.3 Спецификация входных и выходных данных

Входные данные разработанной программы организованы в виде информации, поступающей от пользователей.

Таблица 1 – «Post» (Должность)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код каждой должности |
| name | Текст | Наименование каждой должности |

Таблица 2 – «Employee» (Сотрудник)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код каждого сотрудника |
| id\_post | Счетчик | Код должности |
| name | Текст | Имя сотрудника |
| lastname | Текст | Фамилия сотрудника |
| middlename | Текст | Отчество сотрудника |
| phone | Текст | Номер телефона сотрудника |
| email | Текст | Электронная почта сотрудника |
| birthDate | Дата | Дата рождения сотрудника |
| employment\_date | Дата | Дата принятия сотрудника на работу |
| image | Изображение | Фото сотрудника |

Таблица 3 – «Material» (Материал)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код каждого материала |
| name | Текст | Наименование каждого материала |

Таблица 4 – «ProductType» (Тип изделия)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код каждого типа |
| name | Текст | Наименование каждого типа |

Таблица 5 – «Product» (Изделие)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код каждого изделия |
| id\_material | Счетчик | Код материала |
| id\_productType | Счетчик | Код типа изделия |
| name | Текст | Наименование изделия |
| description | Текст | Описание изделия |
| count | Числовой | Количество изделий |
| image | Изображение | Изображение изделия |

Таблица 6 – «ProductionPlan» (План производства)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код каждого плана |
| name | Текст | Наименование плана |
| description | Текст | Описание плана |
| startDate | Дата | Дата начала плана |
| finishDate | Дата | Дата завершения плана |

Таблица 7 – «ProductionPlanEmployee» (Связующая таблица между Планом производства и Сотрудником)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код связующего поля |
| id\_plan | Счетчик | Код плана производства |
| id\_employee | Счетчик | Код сотрудника |

Таблица 8 – «ProductionPlanProduct» (Связующая таблица между Планом производства и Изделием)

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Комментарий** |
| id | Счетчик | Код связующего поля |
| id\_plan | Счетчик | Код плана производства |
| id\_product | Счетчик | Код изделия |
| productCount | Числовой | Количество производимых изделий |

Выходная информация разработанной программы организована в следующем виде списков данных (рисунок 7) или различных диалоговых окон (рисунки 8 и 9).

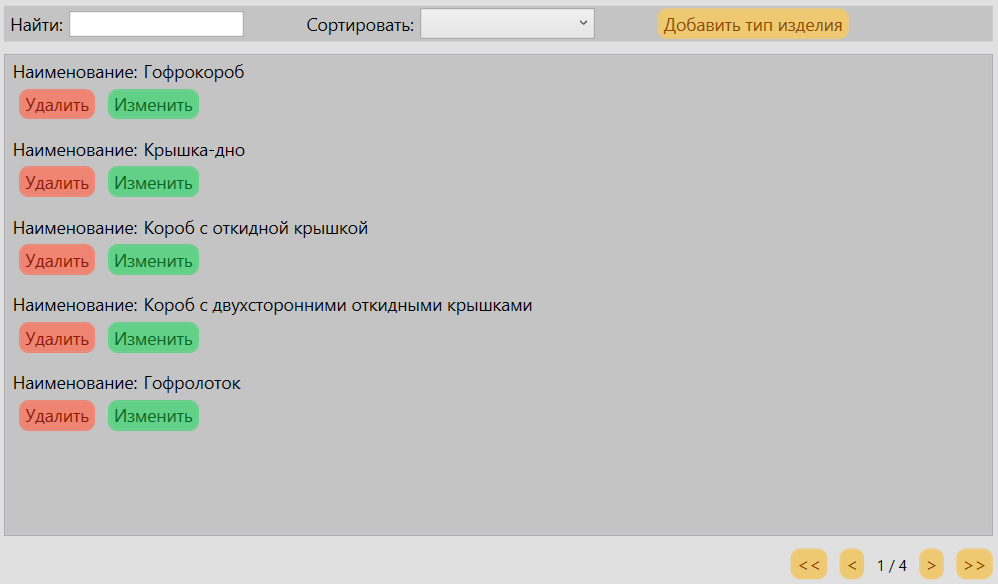


Рисунок 7 – Список типов изделий

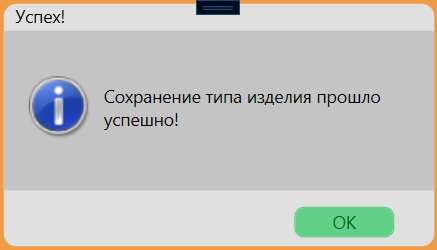


Рисунок 8 – Сообщение об успешности действия

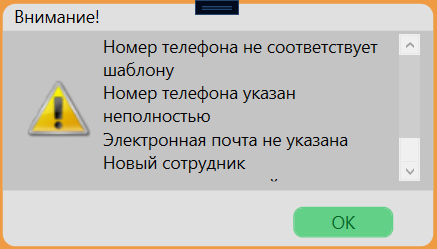


Рисунок 9 – Сообщение о некорректности введенных данных

## 2.3 Реализация программы

## 2.3.1 Анализ средств разработки

Информационные среды по учету данных можно разработать с помощью следующих решений:

1. Среды разработки.
2. Системы управления базами данных.
3. Технологии доступа к данным.

Интегрированная среда разработки – комплекс программных средств, используемых программистами для разработки программного обеспечения (ПО).

Среда разработки включает в себя:

* текстовый редактор;
* транслятор (компилятор и/или интерпретатор);
* средства автоматизации сборки;
* отладчик.

Иногда содержит также средства для интеграции с системами управления версиями и разнообразные инструменты для упрощения конструирования графического интерфейса пользователя. Многие современные среды разработки также включают браузер классов, инспектор объектов и диаграмму иерархии классов – для использования при объектно-ориентированной разработке ПО.

Microsoft Visual Studio является одной из лучших условно-бесплатных сред разработки. Это премиум IDE, стоимость которой зависит от редакции и типа подписки. Она позволяет создавать самые разные проекты, начиная с мобильных и веб-приложений и заканчивая видеоиграми. Microsoft Visual Studio включает в себя множество инструментов для тестирования совместимости – вы сможете проверить свое приложение на более чем 300 устройствах и браузерах. Благодаря своей гибкости, эта IDE отлично подойдет как для студентов, так и для профессионалов.

Eclipse – это бесплатная с открытым исходным кодом среда разработки, которая хорошо подойдет как новичкам, так и опытным разработчикам. Помимо инструментов отладки и поддержки Git/CVS, Eclipse поставляется с Java и инструментом для создания плагинов. Изначально Eclipse использовалась только для Java, но сейчас, благодаря плагинам и расширениям, ее функции значительно расширились. Именно из-за возможности расширить Eclipse своими модулями эта платформа и завоевала свою популярность среди разработчиков.

Система управления базами данных — совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных. Это комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными (вставлять, обновлять, удалять и выбирать).

MySQL – это самая часто используемая СУБД. Основные особенности системы – основана на модели клиент-сервер и имеет открытый исходный код. Предназначена для взаимодействия с базами данных, организованными на реляционной модели. MySQL постоянно совершенствуется и развивается. Она идеально подходит для проектов, которые ведутся в таблицах. Из-за небольшого количества доступных функций, с системой проще взаимодействовать. Разобраться в работе с этой СУБД сможет даже новичок. Управлять базой данных можно без использования программ, для этого достаточно открыть браузер.

PostgreSQL – это система предусмотрена для взаимодействия с базами данных, основанными на объектно-реляционной модели. Многие используют ее в качестве бэкенда для приложений. Основана система на языке структурированных запросов SQL. Она постоянно совершенствуется, что позволяет ей оставаться в топе уже несколько лет.

Язык программирования — формальный язык, предназначенный для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно — ЭВМ) под её управлением.

C# — язык широкого назначения, разработан Microsoft. C# используется для разработки десктопных приложений, веб-приложений, веб-сервисов, приложений Microsoft и игр в Unity. Более того, C# имеет богатый набор библиотек, которые делают его более быстрым и эффективным языком программирования.

Python – широко используется в машинном обучении и в работе с большими данными. В этих областях он зарекомендовал себя как один из лучших инструментов. Также его используют в веб-приложениях: Python прекрасно подходит для небольших и средних проектов.

Технологии доступа к данным являются прослойкой между API конкретного сервера и приложением пользователя, предоставляя программисту простой унифицированный механизм работы с данными.

На сегодняшний день существует множество технологий доступа к данным, таких как BDE, OLE, ODBC, DАО, АDО, и до сих пор разрабатываются новые, более надежные, удобные в работе и более быстродействующие технологии.

Механизмы доступа к базам данных снижают сложность обмена информацией с базами, однако интерпретация результатов их работы также достаточно трудоемка. Поэтому реализованы наборы компонентов, предназначенные для взаимодействия с механизмами обмена. Таким образом, можно выделить несколько субъектов, участвующих в движении информации между базой данных и приложением (например, пользовательским интерфейсом):

1. Интерфейсная часть приложения или его программная часть, манипулирующая информацией, хранимой в базе данных.
2. Компоненты, обеспечивающие связь приложения с механизмом доступа к базе данных.
3. Механизм доступа к базе данных.
4. База данных.

Технология OLE DB (от англ. Object Linking and Embedding DataBase – механизм доступа к базам данных с помощью технологии OLE) представляет собой набор СОМ-интерфейсов, обеспечивающих универсальный доступ приложений к базам данных и другим хранилищам информации (в том числе и нереляционным - файловым системам, системам электронной почты и др.), в которых результатом запросов является информация, представленная в табличной форме.

BDE (англ. Borland Database Engine – механизм доступа к базам данных фирмы Borland) – базовая технология доступа к БД от фирмы Borland, является аналогом ODBC и имеет схожую с ней архитектуру. Этот механизм позволяет получать доступ к реляционным БД с помощью специальных BDE драйверов или через ODBC драйверы.

Для разработки программного приложения была выбрана среда разработки Microsoft Visual Studio, так как данный программный продукт идеально работает с API Windows, а также включает в себя интегрированную среду разработки программного обеспечения и ряд других инструментов.

Для создания базы данных и хранения информации была выбрана СУБД Microsoft SQL Server, так как она реляционная, имеет удобный графический редактор таблиц и связей, а также имеет встроенную среду выполнения SQL-запросов.

Языком программирования был выбран C# - полнофункциональный объектно-ориентированный язык, который поддерживает все три «столпа» объектно-ориентированного программирования: инкапсуляцию, наследование и полиморфизм. Он имеет прекрасную поддержку компонентов, надежен и устойчив благодаря использованию «сборки мусора», обработки исключений, безопасности типов.

Для доступа к данным был выбран Entity Framework 6.0 – гибкая система работы с базами данных в языке C#, позволяет производить действия не только над данными, но и над базой данных в целом. Также EF обладает встроенной поддержкой LINQ для удобной работы с выборками данных.

## 2.3.2 Разработка интерфейса программы

Работа с информационной системой начинается с запуска заставки, во время показа которой запускается программа.



Рисунок 10 – Заставка программы

После запуска появляется основной интерфейс программы, на которой в левой части находится навигационная панель, а в правой – рабочая панель, в которой происходят все манипуляции с данными. С помощью навигационной панели можно перемещаться между различными страницами с данными.

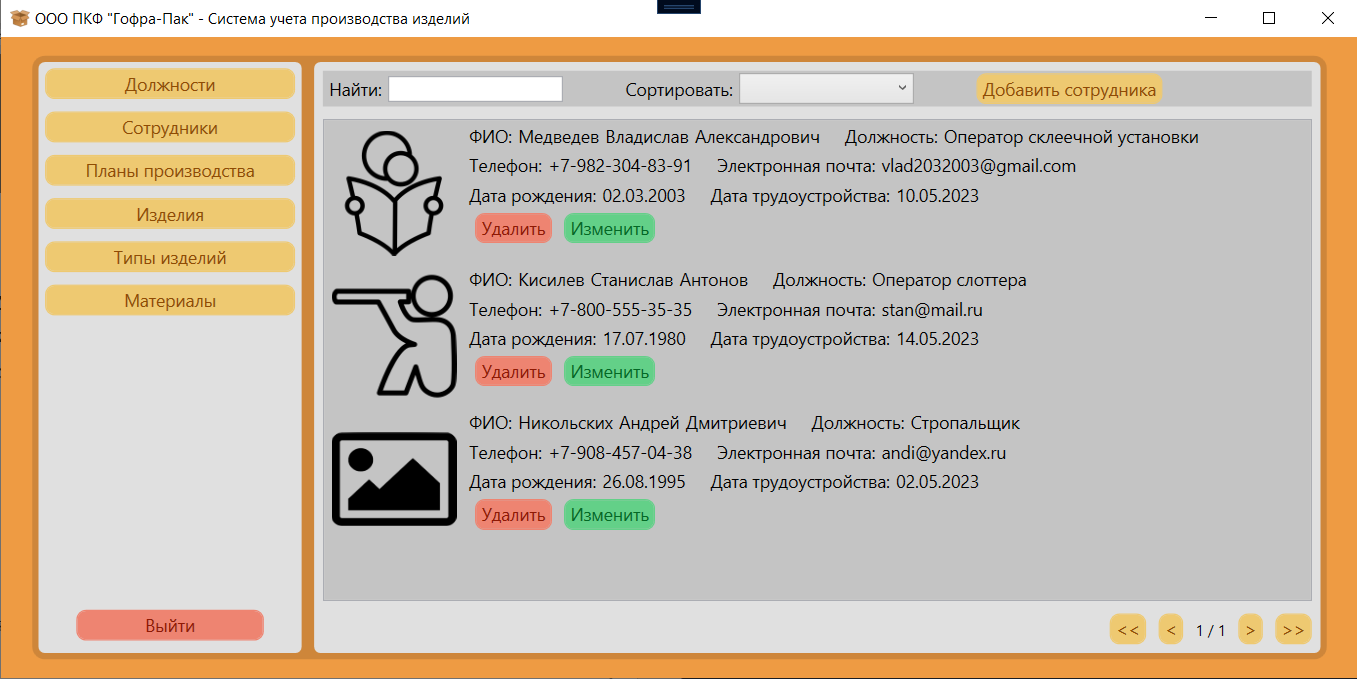


Рисунок 11 – Основной интерфейс программы

Кнопка «Добавить» позволяет внести в таблицу новую информацию. После ее нажатия рабочая панель изменит свое содержимое, что позволит внести в себя новые данные. После окончания процедуры ввода можно сохранить информацию или отменить действие.

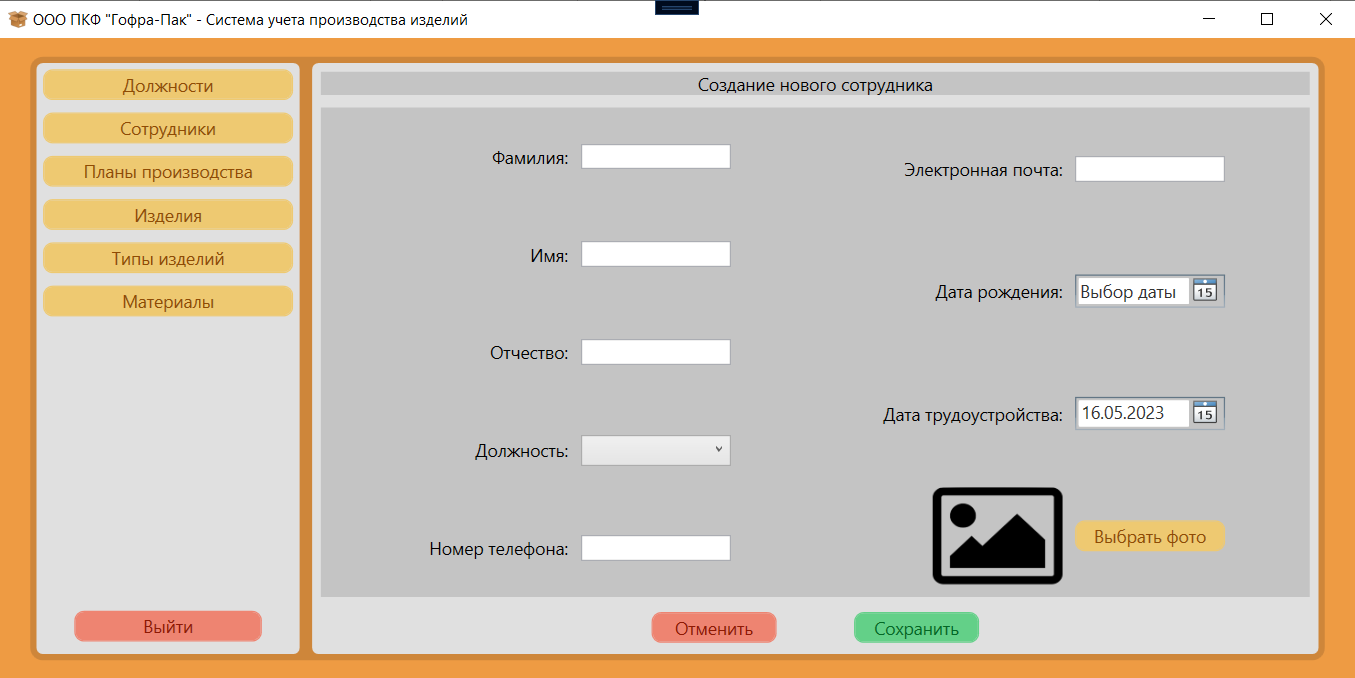


Рисунок 12 – Страница для добавления новых данных

Кнопка «Изменить» позволяет изменить уже существующие данные в таблице. После ее нажатия рабочая панель изменит свое содержимое, позволяя отредактировать существующие данные. После окончания процедуры ввода можно сохранить информацию или отменить действие. В случае отмены останутся предыдущие данные, до редактирования.

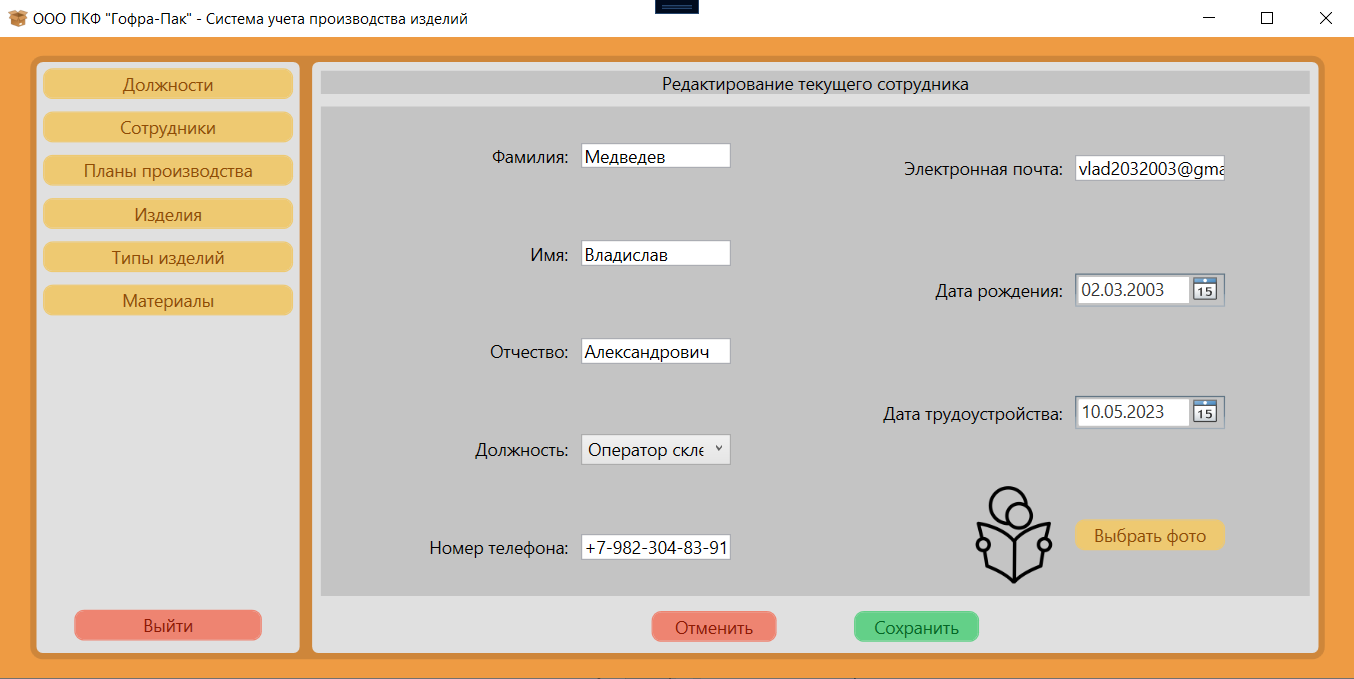


Рисунок 13 – Страница для изменения существующих данных

Кнопка «Удалить» позволяет удалить имеющиеся данные в таблице. После ее нажатия появится диалоговое окно, запрашивающее подтверждение действия (рисунок 14). После согласия появится следующее диалоговое окно, оповещающее об успешности удаления (рисунок 15). На этом этапе можно отменить удаление и тогда данные останутся.

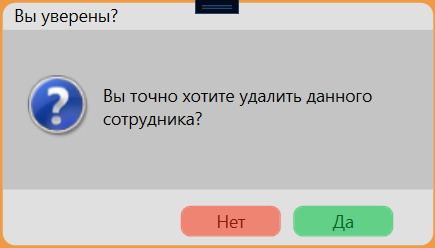


Рисунок 14 – Диалоговое окно подтверждения действия

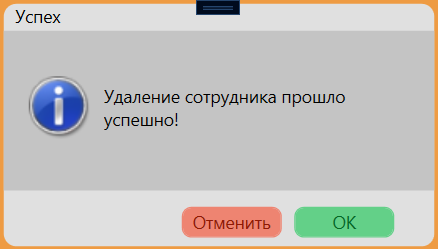


Рисунок 15 – Диалоговое окно успешности действия

## 2.3.3 Проектирование основного и вспомогательного алгоритмов

Алгоритм — конечная совокупность точно заданных правил решения произвольного класса задач или набор инструкций, описывающих порядок действий исполнителя для решения некоторой задачи.

Основные функции, реализованные в программе:

1. Добавление данных.
2. Редактирование данных.
3. Удаление данных.

Дополнительные функции, реализованные в программе:

1. Поиск данных.
2. Сортировка данных.
3. Фильтрация данных.

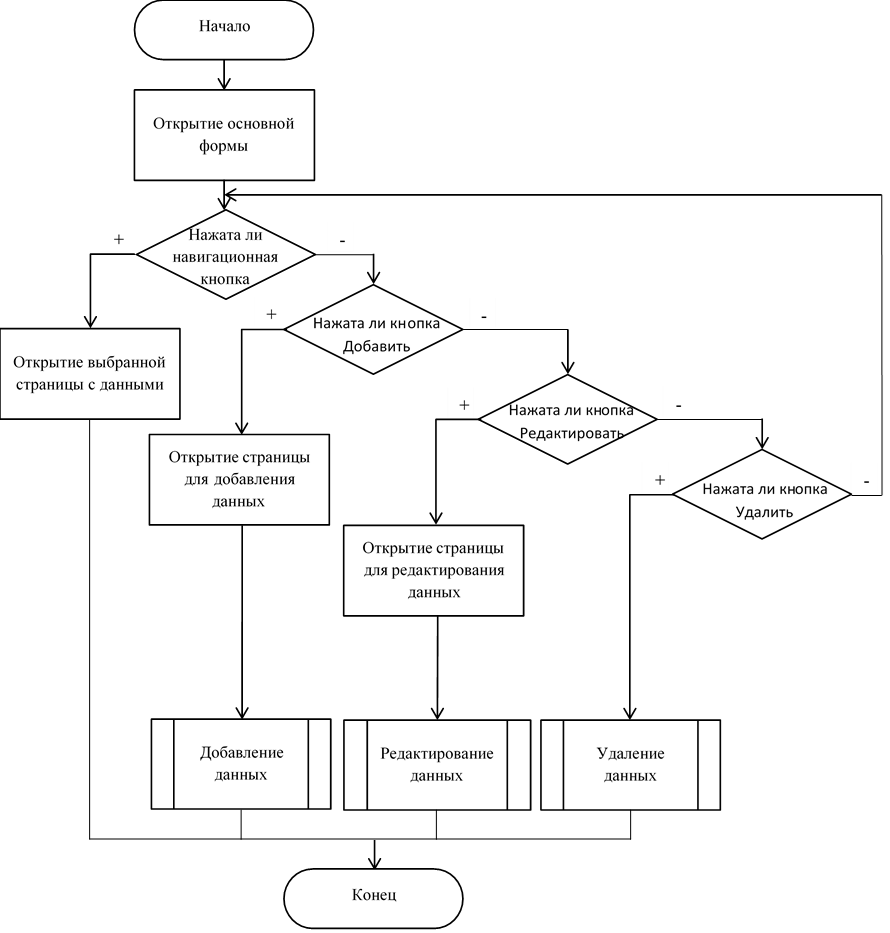


Схема 2 – Основная логика программы

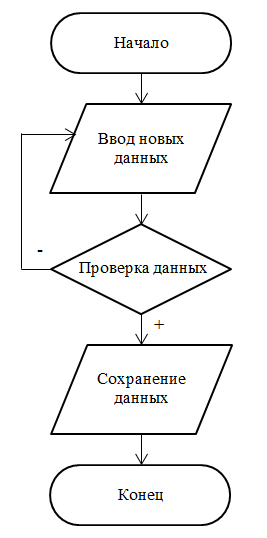


Схема 3 – Логика добавления данных

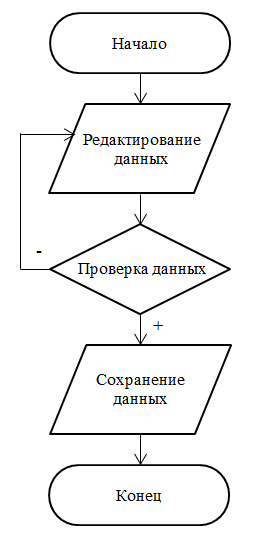


Схема 4 – Логика редактирования данных

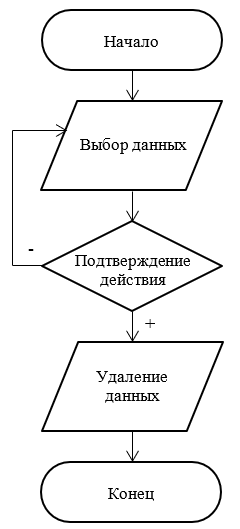


Схема 5 – Логика удаления данных

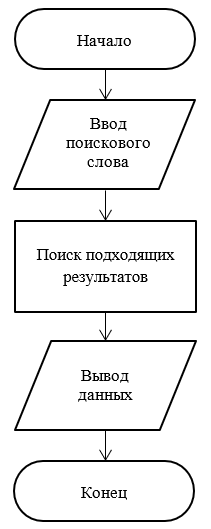


Схема 6 – Логика поиска данных

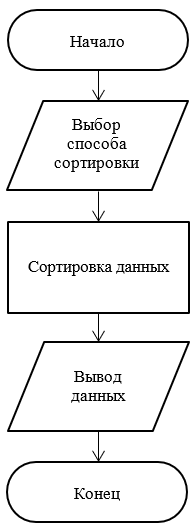


Схема 7 – Логика сортировки данных

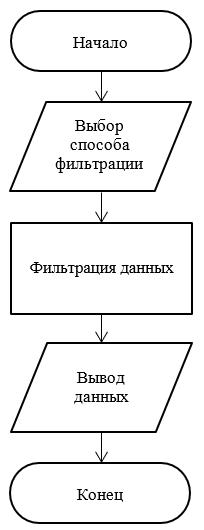


Схема 8 – Логика фильтрации данных

## 2.3.4 Разработка программных модулей

Модуль — функционально законченный фрагмент программы, оформленный в виде отдельного файла с исходным кодом или поименованной непрерывной её части (например, ActiveOberon), предназначенный для использования в других программах.

В программе реализованы следующие модули:

1. Добавление данных (рисунки 16, 19, 20).
2. Редактирование данных (рисунки 17, 19, 20).
3. Удаление данных (рисунок 18).
4. Поиск данных (рисунок 21).
5. Сортировка данных (рисунок 22).
6. Фильтрация данных (рисунок 23).

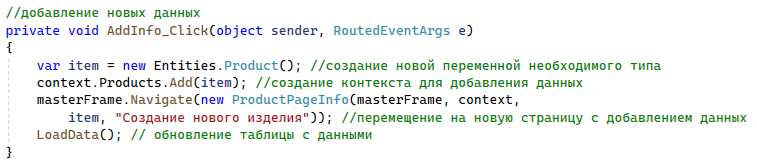


Рисунок 16 – Добавление данных

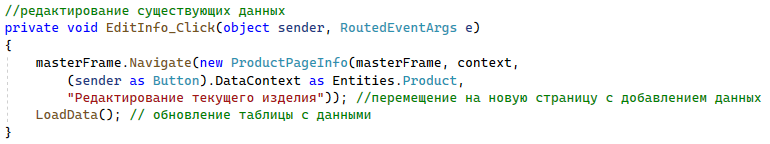


Рисунок 17 – Редактирование данных



Рисунок 18 – Удаление данных

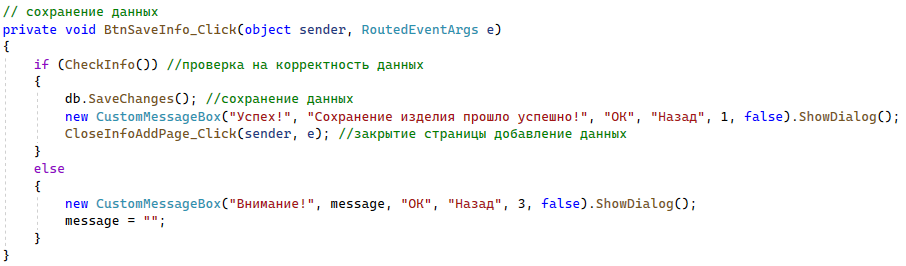


Рисунок 19 – Сохранение данных

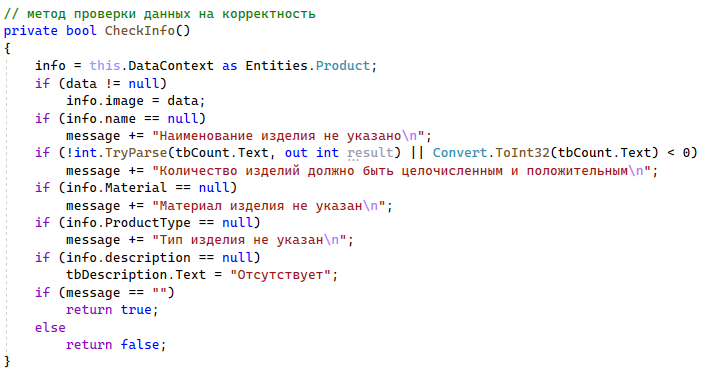


Рисунок 20 – Метод для проверки данных на корректность



Рисунок 21 – Поиск данных

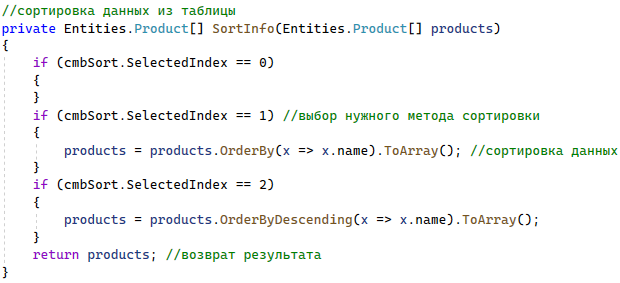


Рисунок 22 – Сортировка данных

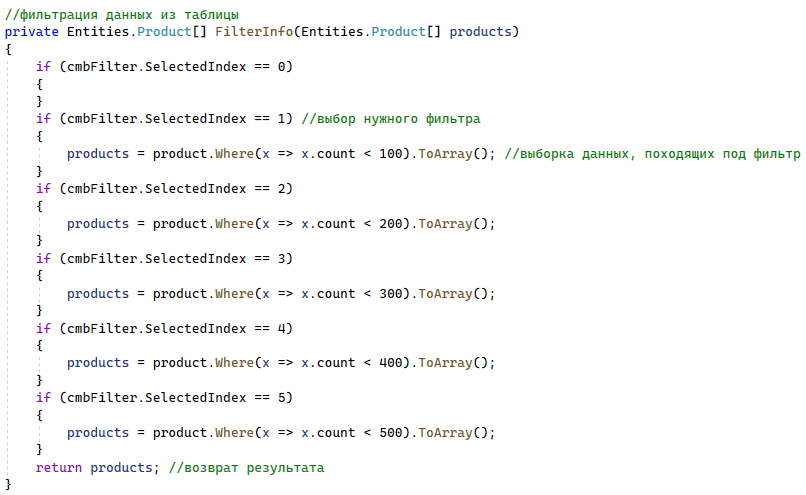


Рисунок 23 – Фильтрация данных

## 2.4 Тестирование приложения

Для тестирования программы были применены такие методы, как:

1. Smoke-тестирование.
2. Вид свободного тестирования (ad-hoc testing) - Monkey testing.
3. «Black Box» testing.
4. «White Box» testing

Smoke-тестирование проводилось в самом начале написания основного функционала программы в заранее подготовленной среде, чтобы убедиться в работоспособности этих функций и приступать к другим тестам.

Monkey testing - произвольное тестирование продукта с целью как можно быстрее, используя различные вариации входных данных, нарушить работу программы или вызвать ее остановку (простыми словами - сломать). «Monkey» тестирование проводилось во время добавления сложных конструкций кода в программу, чтобы выявить различного рода ошибки, такие как некорректность ввода данных или аварийные завершения. Например, в поле для чисел были введены буквы (рисунок 24).

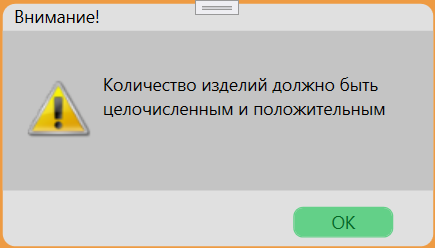


Рисунок 24 – Предупреждение о некорректности введенного значения

На протяжении всей разработки программы проводилось «Black Box» тестирование интерфейса программы с целью выявить неработоспособные элементы интерфейса или элементы, работающие неправильно. Например, в поля для ввода данных были вписаны корректные по типу данные, но некорректные по смыслу: была указана почта, но неполностью или дата рождения оказалась позже, чем дата трудоустройства (рисунок 25). Для предотвращения таких случаев были написаны специальные проверки для таких полей. Также были написаны проверки, требующие подтверждения для совершения действия (рисунок 26).

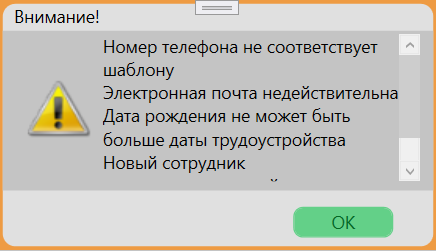


Рисунок 25 – Предупреждение о смысловой некорректности введенных данных

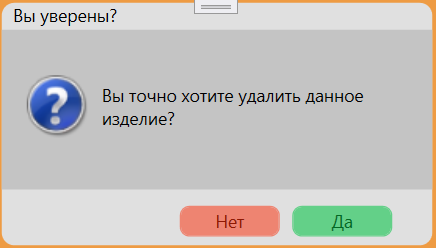


Рисунок 26 – Окно для подтверждения действия

«White Box» тестирование проводилось для устранения ошибок в коде программы. Неоднократно проводился рефакторинг кода.

Итогом всех тестирований стала простая в использовании программа с понятным интерфейсом и имеющая рабочий функционал.

# 3 Организационно-экономическая часть

## 3.1 Расчет сметы затрат на разработку программы

## 3.1.1 Составление проекта выполнения работ

Для расчета сметы затрат составлен проект выполнения работ по созданию программы. Он представляет собой перечень мероприятий, которые необходимо выполнить, чтобы разработать и внедрить систему учета изделий. Работы перечислены в требуемой последовательности с установленной продолжительностью каждого этапа, данные в таблице 9.

Таблица 9 – Проект выполнения работ по созданию программы.

|  |  |
| --- | --- |
| Наименование этапов | Продолжительность, дни |
| Получение задания на разработку программы | 1 |
| Сбор и изучение требований заказчика, анализ предметной области | 1 |
| Проектирование концептуальной модели | 2 |
| Проектирование графического интерфейса программного приложения | 2 |
| Разработка функциональных возможностей программы | 5 |
| Тестирование программы | 1 |
| Отладка программного продукта и встреча с заказчиком | 1 |
| Эксплуатация, сдача проекта заказчику | 1 |

Общие затраты времени на разработку программы определены как сумма продолжительности работ и составляют 14 дней.

## 3.1.2 Расчет материальных затрат

В составе материальных затрат по разработке программы отражена стоимость:

* приобретаемых материалов, которые являются необходимым компонентом при проведении работ;
* покупной энергии, расходуемой на производственные и хозяйственные нужды.

Рассчитываем затраты на эксплуатационные материалы, исходные данные представлены в таблице 10, в ней же и результаты расчетов.

Таблица 10 – Затраты на эксплуатационные материалы.

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование материала | Количество | Цена, руб. | Сумма, руб. |
| Бумага (лист) | 20 | 2 руб. | 40 руб. |
| Интернет (час) | 336 | 0,76 руб. | 255,36 руб. |
| Ручка (шт.) | 1 | 24 руб. | 24 руб. |
| Флеш-карта (шт.) | 1 | 300 руб. | 300 руб. |
| Итого: |  |  | См. = 619,36 руб. |

См. – сумма затрат на эксплуатационные материалы.

Расчёт стоимости электроэнергии.

ЭВМ будет использоваться 14 дней по 8 часов в день, то есть 112 часов.

Исходные данные:

* потребляемая мощность – 0,66 кВт/ч;
* время работы на ЭВМ – 112 ч;
* тариф по электроэнергии – 5,5 руб. /кВт

Рассчитываем стоимость электроэнергии (Сэл.) по формуле:

Сэл. = P × tраб. × Ц,

где P – потребляемая мощность, кВт/ч;

tраб. – время работы на ЭВМ, ч;

Ц – цена за 1 кВт/ч, руб.

Сэл. = 0,66044 × 112 × 5,5 = 406,83 руб.

Рассчитываем сумму материальных затрат (См.з.):

См.з. = См. + Сэл.,

См.з. = 619,36 + 406,83 = 1026,19 руб.

## 3.1.3 Расчет затрат на оплату труда

Для оплаты труда программиста чаше всего используется повременная зарплата.

Затраты на оплату труда включают:

* заработную плату программиста;
* оплату отпуска программиста за отработанное время.

Расчет заработной платы программиста.

Исходные данные:

* время работы над программой, 14 дн.;
* должностной оклад за месяц, 15000 руб.;
* количество рабочих дней за месяц, 22 дн.;
* процент премии, 50 %;
* районный коэффициент, 1,15.

Рассчитываем заработную плату по тарифу:

где Зпл.по тар. - заработная плата по тарифу, руб.;

T – время работы над программой, дн.;

ДО – должностной оклад за месяц, руб.;

tм-ц – количество дней за месяц, дн.

Рассчитываем сумму премии:

где ∑прем. – сумма премии, руб.;

%Пр. – процент премии.

Рассчитываем сумму зарплаты за отработанное время:

*.*,

Рассчитываем сумму зарплаты с районным коэффициентом:

*,*

где ∑з.пл.сРК – сумма зарплаты с районным коэффициентом, руб.,

РК – районный коэффициент, 1,15.

Для расчета очередного отпуска определяем дневную зарплату (Дн.з.пл.):

Исходные данные:

* дневная зарплата, 1176,14 руб.;
* средняя продолжительность отпуска, 28 дн.;
* годовой фонд рабочего времени, 247 дн.;
* время работы над программой, 14 дн.

Рассчитываем сумму оплаты за отпуск:

где ∑оп.от. – сумма оплаты отпуска, руб.;

tот. – продолжительность отпусков, дн.;

Фр.в. – годовой фонд рабочего времени, дн.;

Т – время работы над программой, дн.;

Дн.з.пл. – дневная заработная плата, руб.;

Рассчитываем сумму затрат на оплату труда (Зот.):

,

## 3.1.4 Расчет амортизационных отчислений

Для разработки информационной системы учета изделий использовалась вычислительная техника в виде ноутбука. Вычислительная техника входит в состав основных производственных фондов.

Основные производственные фонды – это совокупность средств труда, функционирующих в сфере материального производства в неизменной натуральной форме в течение длительного времени и переносящих свою стоимость на вновь созданный продукт по частям, по мере изнашивания.

Амортизационные отчисления – это постепенное погашение стоимости основных фондов предприятия (зданий, оборудования, подвижного состава и т.п.), изнашивающихся в процессе работы и от времени.

Рассчитываем амортизационные отчисления за год, исходные данные и результаты расчётов в таблице 11.

Таблица 11 - Амортизационные отчисления за год

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование основных производственных фондов | Стоимость ОПФ, руб. | Норма амортизации, % | Сумма амортизации, руб. |
| Ноутбук | 46000 | 17 | 7820 |
| Итого: |  |  | Ar = 7820 |

Ar - годовая сумма амортизационных отчислений.

Годовая сумма амортизационных отчислений (Ar) рассчитывается по формуле:

где Cn – стоимость первоначальная, руб.;

Ha – норма амортизации, %

Рассчитываем годовую сумму амортизационных отчислений ноутбука, Aг.н.:

Рассчитываем сумму амортизационных отчислений за время разработки программы:

где Ar – годовая сумма амортизационных отчислений, руб.;

n – количество дней на разработку программы;

365 – количество дней в году.

## 3.1.5 Расчет стоимости разрабатываемой программы

Себестоимость — стоимостная оценка текущих затрат на производство и реализацию продукции.

Себестоимость продукции формируется из следующих элементов:

* материальные затраты;
* затраты на оплату труда;
* отчисления на социальные нужды;
* амортизация основных фондов;
* накладные расходы.

В элементе «отчисления на социальные нужды» отражаются обязательные отчисления органам государственного и негосударственного социального страхования, Пенсионного фонда, Государственного фонда занятости и медицинского страхования, в процентах к оплате труда работников.

Общий процент отчислений в фонды остается прежним по сравнению с предыдущим 2022 годом – 30%, из них:

* 22% в пенсионный фонд;
* 2,9% в фонд социального страхования;
* 5,1% в фонд медицинского страхования.

Накладные расходы — это налоги, сборы, отчисления в специальные фонды, платежи по кредитам, затраты на командировки, оплата услуг связи и другие.

Рассчитываем себестоимость разрабатываемой программы по формуле:

где С/С – себестоимость, руб.;

См.з. – материальные затраты, руб.;

З.о.т. – затраты га оплату труда, руб.;

А – амортизационные отчисления, руб.;

От. – отчисления на социальные нужды, руб.;

НР – накладные расходы, руб.

Рассчитываем отчисления на социальные нужды.

Исходные данные:

* затраты на оплату труда -18332,5 руб.;
* установленный размер отчислений - 30 %.

где %отчисл. – процент отчислений, %

От. – отчисления на социальные нужды, руб.

Рассчитываем накладные расходы (80% от основной заработной платы):

Рассчитываем себестоимость.

Исходные данные:

* См.з – материальные затраты, 1026,19 руб.;
* З.о.т. – затраты на оплату труда, 18332,5 руб.;
* А – амортизационные отчисления, 299,95 руб.;
* От. – отчисления на социальные нужды, 5499,75 руб.;
* НР – прочие расходы (накладные расходы), 14666 руб.

Определяем структуру себестоимости разрабатываемой программы, результаты в таблице 12.

Структура – процентное содержание элементов затрат к общей сумме себестоимости.

Таблица 12 – Структура себестоимости

|  |  |  |
| --- | --- | --- |
| Наименование статей затрат | Сумма, руб. | Структура, % |
| Материальные затраты | 1026,19 | 2,58 |
| Затраты на оплату труда | 18332,5 | 46,03 |
| Отчисления на социальные нужды | 5499,75 | 13,81 |
| Амортизационные отчисления | 299,95 | 0,75 |
| Накладные расходы | 14666 | 36,83 |
| Итого: | 39823,94 | 100 |

Вывод: самую большую долю затрат на разработку программы составляют затраты на оплату труда, а самую малую долю – амортизационные отчисления. Амортизация самая малая доля, потому что стоимость ОПФ низкая и количество затраченных на разработку дней небольшое. Затраты на оплату труда занимают самую большую долю, потому что для разработки программы используется умственный труд, который дорого стоит.

## 3.2 Расчет экономической эффективности внедрения программы

Экономическая эффективность – это соотношение результатов с затратами.

Если разрабатываемая программа пользуется спросом, то возможна ее реализация. В этом случае эффективность определяется возможной прибылью от реализации разработанной программы.

Для этого необходимо установить цену на разработанную программу.

При установлении цены используют полную сбытовую себестоимость товарной продукции, она включает производственную себестоимость и внепроизводственные расходы.

Внепроизводственные (коммерческие) расходы включают затраты, связанные с реализацией продукции (расфасовка, упаковка, отгрузка, реклама, маркетинговые исследования, комиссионные; хранение, транспортировка), а также различного рода отчисления и платежи.

Полная себестоимость рассчитывается по формуле:

где С/Ск.– полная себестоимость единицы продукции, руб.;

С/С – себестоимость единицы продукции производственная, руб.;

Вн.р. – внепроизводственные расходы (7-10% от производственной себестоимости), руб.

Исходные данные:

* внепроизводственные расходы – 2787,68 руб. (7 % от производственной себестоимости);
* производственная стоимость – 39823,94 руб.

Оптовую цену предприятия можно определить расчетным путем, используя уровень рентабельности продукции.

Уровень рентабельности продукции устанавливают в зависимости от величины прибыли (20-40% от коммерческой (полной) себестоимости единицы продукции).

Рассчитываем цену разрабатываемой программы по формуле:

где Ц – цена разрабатываемой программы, руб.;

С/Ск. – себестоимость коммерческая (полная) разрабатываемой программы, руб.;

Pnp. – рентабельность продукции, %.

Исходные данные:

* рентабельность продукции – 23%;
* полная себестоимость продукции – 42611,62 руб.

Расчет прибыли от реализации программы.

Прибыль от реализации продукции (работ, услуг) определяется как разница между выручкой от реализации продукции (работ, услуг) в действующих ценах (без налога на добавленную стоимость, акцизного налога) и затратами на ее производство и реализацию, включаемыми в себестоимость продукции (работ, услуг).

Рассчитываем прибыль от реализации разрабатываемой программы:

где Пр. – прибыль от реализации программы, руб.

# 4 Мероприятия по охране труда и окружающей среды

## 4.1 Техника безопасности при работе с вычислительной техникой

Эксплуатирующий средства вычислительной техники и периферийное оборудование персонал может подвергаться опасным и вредным воздействиям, которые по природе действия подразделяются на следующие группы:

* поражение электрическим током;
* механические повреждения;
* электромагнитное излучение;
* инфракрасное излучение;
* опасность пожара;
* повышенный уровень шума и вибрации.

Для снижения или предотвращения влияния опасных и вредных факторов необходимо соблюдать Санитарные правила и нормы, гигиенические требования к видео-дисплейным терминалам, персональным электронно-вычислительным машинам и организации работы:

1. На рабочем месте монитор должен устанавливаться таким образом, чтобы исключить возможность отражения от его экрана в сторону пользователя источников общего освещения помещения.
2. Расстояние от экрана монитора до глаз пользователя должно составлять от 50 до 70 см.
3. Монитор должен быть установлен прямо перед пользователем и не требовать поворота головы или корпуса тела.
4. Рабочий стол и посадочное место должны иметь такую высоту, чтобы уровень глаз пользователя находился чуть выше центра монитора. На экран монитора следует смотреть сверху вниз, а не наоборот.
5. Клавиатура должна быть расположена на такой высоте, чтобы пальцы рук располагались на ней свободно, без напряжения, а угол между плечом и предплечьем составлял от 100° до 110°.
6. При работе с компьютерной мышью рука не должна находиться на весу. Локоть руки или хотя бы запястье должны иметь твердую опору. Если предусмотреть необходимое расположение рабочего стола и кресла затруднительно, рекомендуется применить коврик для мыши, имеющий специальный опорный валик.
7. Недопустимо размещать на системном блоке, мониторе и периферийных устройствах посторонние предметы.
8. Все питающие кабели и провода должны располагаться с задней стороны компьютера и периферийных устройств. Их размещение в рабочей зоне пользователя недопустимо.
9. Запрещается производить какие-либо операции, связанные с подключением, отключением или перемещением компонентов компьютерной системы без предварительного отключения питания.
10. Компьютер не следует устанавливать вблизи электронагревательных приборов и систем отопления.

Требования пожарной безопасности к территориям, зданиям, сооружениям определены Правилами пожарной безопасности (п.п.13, 25, 26, 32-34, 40-42, 44) и сводятся, в частности, к следующему.

Содержание территории:

* территория организаций в пределах противопожарных разрывов между зданиями, сооружениями и открытыми складами должна своевременно очищаться от горючих отходов, мусора, сухой травы. Противопожарные разрывы между зданиями и сооружениями не разрешается использовать для складирования материалов, оборудования и тары;
* дороги, проезды, подходы к зданиям и сооружениям, открытым складам и источникам водоснабжения, используемым для тушения пожаров, должны быть всегда свободными;
* территория организаций должна иметь наружное освещение в темное время суток для быстрого нахождения пожарных гидрантов, наружных пожарных лестниц;
* не разрешается курение на территории и в помещениях складов и баз, хлебоприемных пунктов, объектов торговли, добычи, переработки и хранения легковоспламеняющихся жидкостей (ЛВЖ), горючих жидкостей (ГЖ) и горючих газов (ГГ), производств всех видов взрывчатых веществ, взрывопожароопасных и пожароопасных участков, а также в не отведенных для курения местах иных организаций, в детских дошкольных и школьных учреждениях, в злаковых массивах;
* разведение костров, сжигание отходов и тары не разрешается в пределах установленных нормами проектирования противопожарных расстояний, но не ближе 50 м до зданий и сооружений. Сжигание отходов и тары в специально отведенных для этих целей местах должно производиться под контролем обслуживающего персонала;
* на территориях населенных пунктов и организаций не разрешается устраивать свалки горючих отходов.

Содержание зданий, сооружений, помещений:

* для всех производственных и складских помещений должна быть определена в соответствии с НПБ 105-03 категория взрывопожарной и пожарной опасности, а также класс зоны по Правилам устройства электроустановок, которые надлежит обозначить на дверях помещений;
* около оборудования, имеющего повышенную пожарную опасность, следует вывешивать стандартные знаки безопасности;
* противопожарные системы и установки (противодымная защита, средства пожарной автоматики, системы противопожарного водоснабжения, противопожарные двери, клапаны, другие защитные устройства в противопожарных стенах и перекрытиях и т.п.) помещений, зданий и сооружений должны постоянно содержаться в исправном рабочем состоянии;
* устройства для самозакрывания дверей должны находиться в исправном состоянии. Не допускается устанавливать какие-либо приспособления, препятствующие нормальному закрыванию противопожарных или противодымных дверей (устройств);
* наружные пожарные лестницы и ограждения на крышах (покрытиях) зданий и сооружений должны содержаться в исправном состоянии и не реже одного раза в пять лет подвергаться эксплуатационным испытаниям;
* в помещениях с одним эвакуационным выходом одновременное пребывание пятидесяти и более человек не допускается. В зданиях с массовым пребыванием людей, на случай отключения электроэнергии, у обслуживающего персонала должны быть электрические фонари. Количество фонарей определяется руководителем исходя из особенностей объекта, количества дежурного персонала, количества людей в здании, но не менее одного на каждого работника дежурного персонала;
* двери чердачных помещений, а также технических этажей и подвалов, в которых по условиям технологии не требуется постоянного пребывания людей, должны быть закрыты на замок. На дверях указанных помещений должна быть информация о месте хранения ключей. Окна чердаков, технических этажей и подвалов должны быть остеклены и постоянно закрыты;
* во всех производственных, административных, складских и вспомогательных помещениях на видных местах должны быть вывешены таблички с указанием номера телефона вызова пожарной охраны.

В зданиях и сооружениях организаций запрещается:

* хранение и применение в подвалах и цокольных этажах ЛВЖ и ГЖ, пороха, взрывчатых веществ, баллонов с газами, товаров в аэрозольной упаковке, целлулоида и других взрывопожароопасных веществ и материалов, кроме случаев, оговоренных в действующих нормативных документах;
* использовать чердаки, технические этажи, вентиляционные камеры и другие технические помещения для организации производственных участков, мастерских, а также хранения продукции, оборудования, мебели и других предметов;
* размещать в лифтовых холлах кладовые, киоски, ларьки и т.п.;
* устраивать склады горючих материалов и мастерские, размещать иные хозяйственные помещения в подвалах и цокольных этажах, если вход в них не изолирован от общих лестничных клеток;
* снимать предусмотренные проектом двери эвакуационных выходов из поэтажных коридоров, холлов, фойе, тамбуров и лестничных клеток, другие двери, препятствующие распространению опасных факторов пожара на путях эвакуации. Производить изменения объемно-планировочных решений, в результате которых ухудшаются условия безопасной эвакуации людей, ограничивается доступ к огнетушителям, пожарным кранам и другим средствам пожарной безопасности или уменьшается зона действия автоматических систем противопожарной защиты;
* загромождать мебелью, оборудованием и другими предметами двери, люки на балконах и лоджиях, переходы в смежные секции и выходы на наружные эвакуационные лестницы;
* проводить уборку помещений и стирку одежды с применением бензина, керосина и других ЛВЖ и ГЖ, а также производить отогревание замерзших труб паяльными лампами и другими способами с применением открытого огня;
* оставлять неубранным промасленный обтирочный материал;
* устанавливать глухие решётки на окнах и приямках у окон подвалов, за исключением случаев, специально оговоренных в нормах и правилах, утвержденных в установленном порядке;
* остеклять балконы, лоджии и галереи, ведущие к незадымляемым лестничным клеткам;
* устраивать в лестничных клетках и поэтажных коридорах кладовые (чуланы), а также хранить под лестничными маршами и на лестничных площадках вещи, мебель и другие горючие материалы. Под лестничными маршами в первом и цокольном этажах допускается устройство только помещений для узлов управления центрального отопления, водомерных узлов и электрощитовых, выгороженных перегородками из негорючих материалов;
* устанавливать дополнительные двери или изменять направление открывания дверей (в отступлении от проекта) из квартир в общий коридор (на площадку лестничной клетки), если это препятствует свободной эвакуации людей или ухудшает условия эвакуации из соседних квартир;
* устраивать в производственных и складских помещениях зданий (кроме зданий V степени огнестойкости) антресоли, конторки и другие встроенные помещения из горючих и трудногорючих материалов и листового металла.

# Заключение

В ходе выполнения дипломного проекта была достигнута поставленная цель - создание информационной системы для учета изделий.

Были решены следующие задачи дипломного проектирования:

1. Изучены требования заказчика и проведен анализ предметной области.
2. Изучена техническая литература и интернет-ресурсы.
3. Построена модель базы данных.
4. Разработан пользовательский интерфейс информационной системы.
5. Реализованы функциональные возможности информационной системы.
6. Оформлена техническая документация в виде пояснительной записки к дипломному проекту.

Итогом проектирования стала разработанная информационная система для учета изделий с помощью языка C# и системы управления базами данных Microsoft SQL Server. Подготовлена документация, отражающая процесс работы над проектом.

# Библиография

1. METANIT.COM: Реляционные базы данных и язык SQL: сайт. – URL: https//metanit.com/sql/tutorial/ (дата обращения: 02.05.2023).
2. Microsoft Docs: Документация по Microsoft SQL Server на русском языке / URL: https//docs.microsoft.com/ru-ru/sql/?view=sql-server-ver15 (дата обращения: 29.04.2023).
3. Гниденко И. Г.: Технология разработки программного обеспечения: учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров – М.: Юрайт, 2020. – 235 с. – Текст: электронный // ЭБС Юрайт [сайт]. – URL: https://urait.ru/bcode/453640 (дата обращения: 08.05.2023).
4. Стружкин Н. П.: Базы данных: проектирование. Практикум: учебное пособие для среднего профессионального образования / Н. П. Стружкин, В. В. Годин – М.: Юрайт, 2020. – 291 с. – Текст: электронный // ЭБС Юрайт [сайт]. – URL: https//urait.ru/bcode/455865 (дата обращения: 30.04.2023).
5. Albahari J.: C# 8.0 in a Nutshell [Текст] / Albahari J., Johannsen E. – Sebastopol: O'Reilly Media, 2020 – 1019 с. – ISBN: 978-1492051138.
6. Alls J.: Clean Code in C#. Refactor your legacy C# code base and improve application performance by applying best practices [Текст] / Alls J. – Birmingham: Packt Publishing, 2020. – 500 с. – ISBN: 9781838982973.
7. Албахари Д.: C# 7.0. Справочник. Полное описание языка [Текст] / Д. Албахари, Б. Албахари – М.: Вильямс, 2018. – 1024 с. – ISBN: 978-5-6040043-7-1.
8. Албахари Д.: C# 8.0. Карманный справочник [Текст] / Д. Албахари, Б. Албахари – М.: Диалектика, 2020. – 240 с. – ISBN: 978-5-907203-14-3.
9. Арораа Г.: Паттерны проектирования для C# и платформы .NET Core [Текст] / Г. Арораа, Д. Чилберто – СПБ: Питер, 2021. – 352 с. – ISBN: 978-5-4461-1523-5.
10. Вагнер Б.: Наиболее эффективное программирование на C#. 50 способов улучшения кода [Текст] / Б. Вагнер. - М.: Вильямс, 2018. - 240 c. – ISBN: 978-5-9908911-2-8.
11. Гриффитс И.: Программируем на C# 8.0. Разработка приложений [Текст] / И. Гриффитс – М.: Прогресс книга, 2021. – 944 с. – ISBN: 978-5-4461-1638-6.
12. Клири С.: Конкурентность в C#. Асинхронное, параллельное и многопоточное программирование [Текст] / С. Клири – М.: Прогресс книга, 2020. – 304 с. – ISBN: 978-5-4461-1572-3.
13. Кокоса К. Управление памятью в .NET для профессионалов [Текст] / К. Кокоса – М.: ДМК-Пресс, 2020. – 800 с. – ISBN: 978-5-97060-800-5.
14. Маркин А. В.: Программирование на SQL. Учебное пособие для СПО [Текст] / А. В. Маркин – М.: Юрайт, 2019. – 434 с. – ISBN: 978-5-534-11093-7.
15. Молинаро Э.: SQL. Сборник рецептов. Решения и методики построения запросов для разработчиков баз данных [Текст] / Э. Молинаро, Р. Грааф – Санкт-Петербург: БХВ, 2022. – 592 с. – ISBN: 978-5-9775-6759-6.
16. Прайс М.: C# 9 и .NET 5. Разработка и оптимизация [Текст] / М. Прайс – СПБ: Питер, 2022. – 832 с. – ISBN: 978-5-4461-2921-8.
17. Рихтер Д.: CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд. [Текст] / Д. Рихтер – СПБ: Питер, 2022. – 896 с. – ISBN: 978-5-4461-1102-2.
18. Скит Д.: C# для профессионалов. Тонкости программирования [Текст] / Д. Скит. – М.: Вильямс, 2019. – 608 с. – ISBN: 978-5-907114-62-3.
19. Троелсен Э.: Язык программирования C# 7 и платформы. NET и NET Core [Текст] / Э. Троелсен – М.: Диалектика/Вильямс, 2018. – 1328 с. – ISBN: 978-1-4842-3017-6.
20. Фленов М. Е.: Библия C#. 5-е издание [Текст] / М. Е. Фленов - Санкт-Петербург: БХВ, 2022. – 464 с. – ISBN: 978-5-9775-6827-2.