Введение

В наше время автоматизация различных процессов, протекающих на предприятии, является неотъемлемой частью повышения производительности труда и позволяет сэкономить массу времени и средств.

Слишком большие материальные затраты и затраты по трудовому времени на ведение и контроль документов, содержание их в неактуальном состоянии, а также наличие множества ошибок и сложность формирования отчетов – все это говорит о необходимости автоматизации.

В настоящее время специалисты разрабатывают и внедряют на предприятия для автоматизации процессов такие программы, как информационные системы.

Информационная система (ИС) – это система, предназначенная для хранения, поиска и обработки информации, и соответствующие организационные ресурсы, которые обеспечивают и распространяют информацию.

ИС предназначена для своевременного обеспечения надлежащих людей надлежащей информацией, то есть для удовлетворения конкретных информационных потребностей в рамках определенной предметной области.

Регистрация в общежитии техникума является важным пунктом для выдачи студентам временной прописки. Обычно регистрация всех студентов проходит в один день, что значительно усложняет весь этот процесс, а заполнение документации в нескольких экземплярах еще и сильно затягивает его.

Заполняется и хранится вся документация в бумажном виде, что очень неудобно для хранения, поиска и выборки нужной сотрудникам информации. Также высока вероятность написания ошибочных данных, и как следствие, это приводит к большим проблемам.

Во избежание всех этих проблем было принято решение разработать и внедрить в Томской техникум информационных технологий свою информационную систему регистрации студентов в общежитии, которая бы ускорила и облегчила заполнение, редактирование и удаление данных, а также формирование различных отчетов.

Томской техникум информационных технологий – учреждение среднего специального технического профессионального образования.

«ОБЗАЦ ПРО ТЕХНИКУМ»

Цель дипломного проекта – разработать автоматизированную информационную систему для Томского техникума информационных технологий для регистрации студентов в общежитии.

В дипломном проекте представлены: эскизный проект, алгоритмы решения задач в виде блок схем, ERD диаграмма, листинг кода программы с комментариями и подробное руководство пользователя с иллюстрациями.

1 ОБЩАЯ ЧАСТЬ

1.1 Постановка задачи

Требуется разработать информационную систему для регистрации студентов в общежитии техникума. Система должна предусматривать хранение информации обо всех студентах, прошедших регистрацию (их паспортные и контактные данные, данные представителей и родственников, данные заключенного договора и др.), а также техническую информацию о самом общежитии (список комнат, список работников и пр.). Иметь возможность формирования ряда документации в различных форматах для дальнейшей их отправки на печать.

Система должна быть ориентирована как на обычного пользователя (сотрудника общежития), который будет работать со всеми имеющимися данными, не касающихся самих пользователей, а также и на администратора, в возможности которого входит управление списком пользователей, имеющим доступ к ПО. Иметь автоматизированный и ручной ввод данных. Должно быть организовано разграничение прав доступа к данным. Развитые механизмы поиска.

При добавлении данных, должна быть обеспечена целостность и непротиворечивость данных, а также предусмотрен контроль ввода некорректной информации. У системы должен быть простой, понятный и красивый интерфейс с анимацией. Нужно предусмотреть маски для ввода данных (телефон, паспортные данные и др.).

В системе должна быть предусмотрена авторизация пользователей с защитой от перебора паролей, сами пароли должны храниться в шифрованной виде, так как ПО хранит паспортные данные.

Обычный пользователь с помощью системы должен иметь возможность решать следующие задачи:

1. регистрировать студентов в общежитии;
2. просматривать, редактировать и удалять информацию по каждому студенту (его данные, представителей, родственников, данные договора);
3. просматривать список всех зарегистрированных студентов;
4. просматривать список всех заключенных договоров;
5. просматривать список родственников по каждому студенту;
6. экспортировать данные студентов в MS Word и MS Excel;
7. сортировать таблицы по любому полю;
8. производить поиск в таблицах;
9. просматривать список всех комнат и данных по ним, в том числе количество свободных мест на данный момент;
10. изменять список имеющихся комнат.

Пользователю с правами администратора требуется предоставить следующий функционал:

1. возможность создавать новых пользователей;
2. просматривать и редактировать список пользователей;
3. иметь возможность задать новый логин или пароль выбранному пользователю;
4. изменять данные своего профиля.

### Основные функции программы:

1. Функция блокировки кнопки входа на 1 минуту после трех неудачных попыток авторизации и последующее увеличение времени на 1 минуту после каждой следующей попытки;
2. Функция добавления студента и заполнение данных о нем (имя, фамилия, отчество, телефон, дата рождения, группа, и возможность загрузить фото);
3. Функция добавления паспорта студента (серия, номер, кем выдан, дата выдачи, адрес прописки, код подразделения);
4. Функция добавления представителей студента и заполнение данных о них (имя, фамилия, отчество, телефон, кем приходится, место работы, место жительства, домашний телефон);
5. Функция добавления паспорта представителя (серия, номер, кем выдан, дата выдачи, адрес прописки, код подразделения);
6. Функция добавления списка родственников студента и заполнение данных по каждому из них (имя, фамилия, отчество, телефон, степень родства, адрес);
7. Функция добавления договора и заполнение данных о нем;
8. Функция вывода списка всех зарегистрированных студентов с возможностью сортировки по любому полю;
9. Функция мягкого поиска студента в списке студентов по фамилии без учета регистра букв;
10. Функция удаления студента;
11. Функция редактирования данных студента и его паспортных данных;
12. Функция редактирования данных представителей студента и их паспортных данных;
13. Функция редактирования списка родственников студента и данных по каждому родственнику;
14. Функция редактирования данных договора студента;
15. Функция удаления родственника у студента;
16. Функция вывода списка родственников каждого студента с возможностью сортировки по любому полю;
17. Функция мягкого поиска нужного списка родственников по фамилии студента без учета регистра букв;
18. Функция вывода списка всех договоров с возможностью сортировки по любому полю;
19. Функция мягкого поиска нужного договора в списке договоров по фамилии студента без учета регистра букв или дате заключения;
20. Функция удаления договора студента;
21. Функция вывода списка всех комнат с выводом количества свободных мест по каждой комнате;
22. Функция удаления выбранной комнаты;
23. Функция добавления новой комнаты;
24. Функция формирования документа «Договор о найме жилого помещения в специализированном жилом фонде» в формате Word c экспортом в него данных выбранного студента;
25. Функция формирования документа «Дополнительное соглашение к договору о найме жилого помещения в специализированной жилом фонде» в формате Word с экспортом в него данных выбранного студента;
26. Функция формирования документа «Анкета студента» в формате Excel c экспортом в него данных выбранного студента;
27. Функция изменения данных профиля;
28. Функция вывода списка всех пользователей программы;
29. Функция изменения логина и/или пароля для выбранного пользователя;
30. Функция создания нового пользователя;

1.2 Структура данных

База данных информационной системы будет состоять из нескольких таблиц. Взаимосвязь таблиц представлена на рисунке 1. В таблицах 1-8 представлено подробное описание данных.

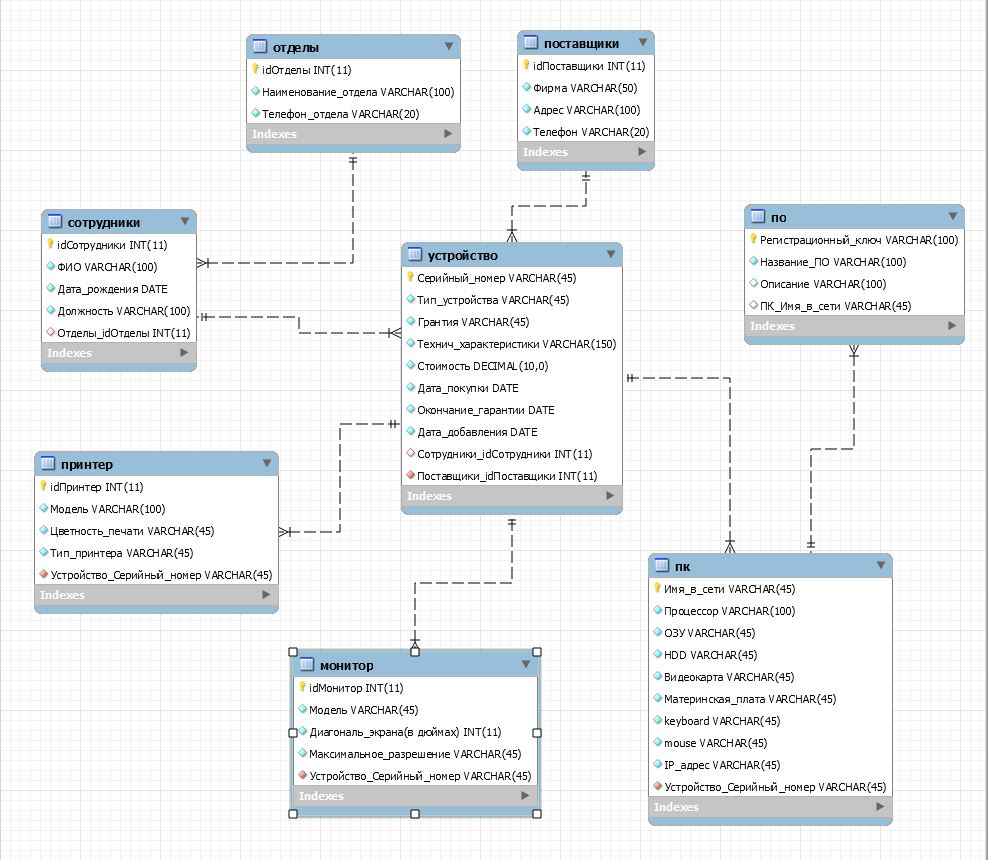


Рисунок 1 – Структура баз данных

Таблица 1 - Устройство

|  |  |  |  |
| --- | --- | --- | --- |
| №  п-п | Название поля | Тип данных | Описание |
| 1 | Серийный\_номер | VARCHAR(45) | Идентификатор устройства |
| 2 | Тип\_ устройства | VARCHAR(45) | Тип устройства |
| 3 | Гарантия | VARCHAR(45) | Гарантия устройства |
| 4 | Технич\_ характеристики | VARCHAR(150) | Тех.характеристики уст-ва |
| 5 | Стоимость | DECIMAL(10,0) | Стоимость устройства |
| 6 | Дата\_покупки | DATE | Дата покупки устройства |
| 7 | Окончание\_гарантии | DATE | Окончание гарантии |
| 8 | Дата\_добавление | DATE | Дата добавление устройства в БД |
| 9 | Сотрудники\_idСотрудники | INT(11) | Внешний ключ Сотрудника |
| 10 | Поставщики\_idПоставщики | INT(11) | Внешний ключ Поставщика |

Таблица 2 - Поставщики

|  |  |  |  |
| --- | --- | --- | --- |
| №  п-п | Название поля | Тип данных | Описание |
| 1 | idПоставщики | INT(11) | Идентификатор поставщика |
| 2 | Фирма | VARCHAR(50) | Наименование фирмы |
| 3 | Адрес | VARCHAR(100) | Адрес поставщика |
| 4 | Телефон | VARCHAR(20) | Телефон поставщика |

Таблица 3 - Сотрудники

|  |  |  |  |
| --- | --- | --- | --- |
| № п-п | Название поля | Тип данных | Описание |
| 1 | idСотрудники | INT(11) | Идентификатор сотрудника |
| 2 | ФИО | VARCHAR(100) | ФИО сотрудника |
| 3 | Дата\_рождения | DATE | Дата рождения сотрудника |
| 4 | Должность | VARCHAR(100) | Должность сотрудника |
| 5 | Отделы\_idОтделы | INT(11) | Внешний ключ отдела |

Таблица 4 - Заказы

|  |  |  |  |
| --- | --- | --- | --- |
| № п-п | Название поля | Тип данных | Описание |
| 1 | idОтделы | INT(11) | Идентификатор отдела |
| 2 | Наименование\_отдела | VARCHAR(100) | Наименование отдела |
| 3 | Телефон\_отдела | VARCHAR(20) | Телефон отдела |

Таблица 1 - ПК

|  |  |  |  |
| --- | --- | --- | --- |
| № п-п | Название поля | Тип данных | Описание |
| 1 | Имя\_в\_сети | VARCHAR(45) | Идентификатор  Имя в сети |
| 2 | Процессор | VARCHAR(100) | Модель процессора |
| 3 | ОЗУ | VARCHAR(45) | Оперативная память ПК |
| 4 | HDD | VARCHAR(45) | Жесткий диск ПК |
| 5 | Видеокарта | VARCHAR(45) | Стоимость устройства |
| 6 | Материнская\_плата | VARCHAR(45) | Материнская плата ПК |
| 7 | keyboard | VARCHAR(45) | Клавиатура ПК |
| 8 | mouse | VARCHAR(45) | Мышь ПК |
| 9 | IP\_адрес | VARCHAR(45) | IP адрес в сети |
| 10 | Устройство\_Серийный\_номер | VARCHAR(45) | Внешний ключ Серийный номер ПК |

Таблица 1 - Монитор

|  |  |  |  |
| --- | --- | --- | --- |
| № п-п | Название поля | Тип данных | Описание |
| 1 | idМонитор | INT(11) | Идентификатор  Монитора |
| 2 | Модель | VARCHAR(45) | Модель Монитора |
| 3 | Диагональ\_экрана(в дюймах) | INT(11) | Диагональ экрана |
| 4 | Максимальное\_разрешение | VARCHAR(45) | Максимальное разрешение монитора |
| 5 | Устройство\_Серийный\_номер | VARCHAR(45) | Внешний ключ Серийный номер монитора |

Таблица 1 - Принтер

|  |  |  |  |
| --- | --- | --- | --- |
| № п-п | Название поля | Тип данных | Описание |
| 1 | idПринтер | INT(11) | Идентификатор  Принтера |
| 2 | Модель | VARCHAR(45) | Модель Принтера |
| 3 | Цветность\_печати | INT(11) | Цвет печати |
| 4 | Тип\_принтера | VARCHAR(45) | Тип принтера(струйный, лазерный) |
| 5 | Устройство\_Серийный\_номер | VARCHAR(45) | Внешний ключ Серийный номер принтера |

Таблица 1 - ПО (программное обеспечение)

|  |  |  |  |
| --- | --- | --- | --- |
| № п-п | Название поля | Тип данных | Описание |
| 1 | Регистрационный\_ключ | VARCHAR(100) | Идентификатор  Регистрационный ключ |
| 2 | Название\_ПО | VARCHAR(100) | Название ПО |
| 3 | Описание | VARCHAR(100) | Описание |
| 4 | ПК\_Имя\_в\_сети | VARCHAR(45) | Внешний ключ ПК\_Имя\_в\_сети |

1.3 Выбор языка программирования и СУБД

Основным языком программирования для проекта, был выбран C#(Си Шарп).

Язык программирования C# был создан относительно не давно - в конце 1998 года в группе специалистов компании Майкрософт. Целью создания языка С# была возможность создания различных программ для платформы Microsoft.NET. Данная привязанность платформ Microsoft .NET и языка С# была устранена в дальнейшем, но сама платформа .NET во всех версиях в своем составе содержит компилятор кодов данного языка программирования, что позволяет запускать приложения без установки дополнительного программного обеспечения.

Наименование языка программирования С# связана с тем, что синтаксис очень напоминает таковой C++. Вообще, новый язык программирования С# привнял в себя много положительных черт своих предшественников - Delphi, C++, Java. При этом из языка программирования С# изъяты проблемные алгоритмы, которые показали свою не стабильность в перечисленных языках программирования.

Общеязыковая среда исполнения (CLR) и базовая библиотека классов являются основой для целого стека технологий, которые разработчики могут задействовать при построении тех или иных приложений. Например, для работы с базами данных в этом стеке технологий предназначена технология ADO.NET. Для построения графических приложений с богатым насыщенным интерфейсом - технология WPF. Для создания веб-сайтов - ASP.NET и т.д.

Реализация проекта осуществлялась в интегрированной среде разработки Microsoft Visual Studio 2015.

Microsoft Visual Studio обладает огромным набором средств и возможностей: позволяет разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения и веб-службы для всех поддерживаемых платформ: Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone, .NET Compact Framework и Silverlight. Простейшего рефакторинг кода.

Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня.

Выбор языка программирования обусловлен следующим образом: создать приложение на C# легче, чем на С++, поскольку синтаксис языка более простой, чем синтаксис С++. Тем не менее, C# является мощным языком программирования, и имеется мало вещей, которые можно сделать на С++ и нельзя на C#.

Иногда код на C# оказывается объёмнее, чем на С++. Это следствие того, что в C# (в отличие от С++) осуществляется контроль безопасности использования типов (если некоторые данный отнесены к определённому типу, то они не могут самостоятельно преобразоваться к другому типу). Существуют строгие правила, по которым следует осуществлять преобразование из одного типа в другой. Это часто приводит к необходимости писать на C# код более объёмный, чем на С++. Однако в замен мы получаем то преимущество, что программа становится более надёжной и её отладка упрощается.

Для взаимодействия и хранения данных использовалась система управления базами данных (СУБД) MYSQL.

Разработка базы данных MySQL началась в 1995 году, за это время над ней работали несколько компаний, и сейчас она принадлежит Oracle. Кроме версии с открытым исходным кодом, существует несколько коммерческих версий, в которых реализованы дополнительные возможности, такие как кластер гео-репликации и автоматическое масштабирование.

SQL-сервер – это программа, принимающая запросы, написанные на языке SQL, и отсылающая обратно определённые ответы. Ответами могут быть: данные, количество строк, задействованных в запросе, или просто строка. MySQL для поставленных задач является достаточно быстродействующей и устойчивой к ошибкам СУБД. Она прекрасно справляется с обработкой значительных массивов данных. Кроме того, MySQL в запросах к большим таблицам превосходит многие другие системы. Она стабильна и превосходно работает даже в случаях, когда несколько сотен пользователей нуждаются в доступе к одним и тем же данным. MySQL также масштабируема, работает на различных системах, – от персональных компьютеров до больших многопроцессорных систем.

MySQL имеет только необходимые функции. Она не поддерживает вложенных запросов, тригерров; отсутствует также встроенная поддержка XML, OLAP и конструктов (constraits). Но взамен этого есть новации (например, возможность кешировать запросы). Отсутствующие функции были убраны для увеличения быстродействия, которое обеспечивает эта СУБД.

MySQL относиться к типичным реляционным базам данных, все данные хранятся в таблицах и приложения могут очень быстро получить к ним доступ. Для запросов используется стандартный язык SQL, поддерживается большинство возможностей языка, определенных стандартом. При всем этом, она легка в использовании и развертывании.

За время своего развития MySQL получила поддержку различных типов таблиц, интеграцию во многие программы и языки программирования, а также имеет веб и графические интерфейсы для настройки.

Чтобы осуществлять связь между базой данных и приложением на C# необходим посредник. И именно таким посредником является технология ADO.NET.

ADO.NET предоставляет собой технологию работы с данными, которая основана на платформе .NET Framework. Эта технология представляет нам набор классов, через которые мы можем отправлять запросы к базам данных, устанавливать подключения, получать ответ от базы данных и производить ряд других операций.

Чтобы использовать один и тот же набор объектов для разных источников данных, необходим соответствующий провайдер данных. Собственно через провайдер данных в ADO.NET и осуществляется взаимодействие с базой данных. Причем для каждого источника данных в ADO.NET может быть свой провайдер, который собственно и определяет конкретную реализацию вышеуказанных классов. Соответственно для MYSQL нам нужен свой коннектор MYSQL Connector\Net загруженный с официального сайта MYSQL.

Для экспорта данных в excel таблицу мы использовали специализированные библиотеки, загруженные из Microsoft Visual Studio.

2 СПЕЦИАЛЬНАЯ ЧАСТЬ

2.1 Эскизный проект

В данной программе будут использоваться функции:

* “Графический интерфейс”;
* “Авторизация”;
* “Изменение пароля”;
* “Добавление записи”;
* “Редактирование записи”;
* “Удаление записи”;
* “Экспорт в Excel”;
* “Поиск”;
* “Сортировка”;

Подпрограмма Графический интерфейс

Предназначена для создания графического пользовательского интерфейса.

Входные данные: нет.

Выходные данные: нет.

Данная подпрограмма предназначена для доступа к основным функциям программы: “Авторизация”, “Изменение пароля”, “Добавление записи”, “Редактирование”, “Удаление”, “ Экспорт в Excel ”, “Поиск”, “Сортировка”. А также отображает необходимые компоненты для их вызова и работы с БД.

На главном окне программы расположены: меню, кнопки редактирования таблицы, экспорт в Excel. В главном окне содержится все необходимые компоненты для работы с функциями. Объект таблицы используются для вывода информации из БД.

Функция “Авторизация”

Выполняет авторизацию пользователя.

Входные данные: введенные с клавиатуры логин и пароль.

Выходные данные: переход на другую форму.

1. Считывается имя пользователя и пароль.
2. Если введены данные клиента, то выполняется пункт 3, иначе выполняется пункт 6.
3. Открывается окно с функциями, доступными клиенту и выполняется пункт 8.
4. Если введены данные администратора, то выполняется пункт 5, иначе выполняется пункт 6.
5. Открывается окно с функциями, доступными администратору и выполняется пункт 8.
6. Выводится сообщение о не правильно введённом пароле.
7. Конец работы функции.

Функция “Добавление записи”

Данная функция реализует добавление новых записей таблицу “Устройства” (Монитор).

Входные данные: введенные данные с клавиатуры в поля: «Тип устройства», «Модель», «Диагональ экрана», «Разрешение», «Гарантия», «Стоимость», «Серийный номер», «Дата покупки», «Окончание гарантии», «Дата добавления», «Сотрудник», «Поставщик».

Выходные данные: новая запись в таблице.

1. Считываем данные введенные с клавиатуры.
2. Если имеются не заполненные поля, тогда выводится предупреждающее сообщение и выполняется пункт 9, иначе пункт 3.
3. Переходим на первую запись в таблице.
4. Если ID(«Серийный номер») в таблице “Устройства” совпадает с введенным ID(«Серийный номер»), то выводится сообщение “ID существует” и выполняется пункт 10, иначе выполняется пункт 5.
5. Переходим на следующую запись.
6. Если конец таблицы то выполняется пункт 7, иначе пункт 4.
7. Производится запись введённых значений в поля таблицы.
8. Изменения сохраняются в базе данных.
9. Конец подпрограммы.

Аналогично функция “Добавление записи” работает для других таблиц.

Функция “Смена пароля”

Данная функция реализует смену пароля.

Входные данные: введенные данные с клавиатуры в поля «Пользователь», «Старый пароль», «Новый пароль».

Выходные данные: Новый пароль.

1. При нажатии на кнопку открывается диалоговое окно «Смена пароля».
2. Если старый пароль указан неверно, открывается диалоговое окно с сообщением об ошибке, далее выполняется пункт 4 иначе пункт 3.
3. Происходит смена пароля, со старого на новый.
4. Конец функции.

Функция “Редактирование записи”.

Данная функция реализует редактирование имеющихся данных.

Входные данные: запись таблицы.

Выходные данные: измененная запись таблицы.

1. Выбирается поле для редактирования.
2. Данные из полей таблицы вставляются в соответствующие поля редактирования.
3. Данные в полях изменяются пользователем.
4. Внесенные изменения считываются из полей для редактирования и записываются в соответствующие поля таблицы.
5. Выводится сообщение об успешном окончании операции.
6. Конец функции.

Функция “удаление записи”

Данная функция реализует удаление записи из таблиц.

Входные данные: Запись в таблице.

Выходные данные: Преобразованная таблица.

Выделяем запись в таблице.

1. Запрос к БД.
2. Если ошибка, то пункт 5 иначе пункт 4.
3. Каскадное удаление записей в БД.
4. Вывод обновленной таблицы.
5. Конец программы.

Аналогично функция “Удаление” работает для других таблиц.

Функция “Экспорт в Excel”

Данная функция позволяет экспортировать данные в Excel таблицу.

Входные данные: Записи в таблице «Устройства».

Выходные данные: Файл MS Excel.

1. Проверяем, пуста ли таблица «Устройства» если пустая выводим диалоговое окно «Таблица пустая» далее пункт 7 иначе пункт 2
2. Создаем объект Excelapp для работы с файлами MS Excel.
3. В цикле проходимся по объекту datagridview таблицы «Устройства» и записываем строки и столбцы в объект Excelapp.
4. Вывод окна для указания названия и пути для сохранения файла.
5. Объект Excelapp записывает данные в файл MS Excel.
6. Конец функции.

Функция “Поиск”

Данная функция выполняет поиск по таблице.

Входные данные: данные введенные с клавиатуры.

Выходные данные: результат поиска.

1. Считывается данные, введенные с клавиатуры в поля для поиска.
2. В цикле побегаем по объекту datagridview таблицы по которой осуществляем поиск, если не найдено то выводим диалоговое окно «Не найдено» далее пункт 4 иначе пункт 3.
3. Выделяем найденную запись в таблице
4. Конец функции.

Функция “Сортировка”

Данная функция позволяет сортировать поля таблицы в прямом порядке.

Входные данные: таблица.

Выходные данные: отсортированная таблица.

Алгоритм сортировки основан на внутренней реализации метода Sort объекта datagridview.

1. По нажатию на названия поля объекта datagridview создается обработчик события.
2. В метод Sort объекта datagridview передается объект обработчика события с названием поля.
3. На выходе получаем отсортированный datagridview.

2.2 Технический проект

“Графический интерфейс”

НАЧАЛО

Ввод имени команды

Авторизация

Автори-зация?

Добавление записи

Добавление записи?

Редактирова-ние

Редакти-рование?

Удаление

Удаление?



Добавление изображения

Добавление изоб.?

КОНЕЦ



Поиск

Поиск?

Сортировка

Сортировка?

Функция “Авторизация”

да

нет

НАЧАЛО

Ввод логина и пароля

Данные клиента

Данные Админист

Не верный пароль

Открытие окна клиента

КОНЕЦ

нет

да

Открытие окна администратора

Функция “Добавление записи”

да

НАЧАЛО

Добавление новой строки в таблицу

Данные для новой записи

КОНЕЦ

Зап. поля?

Не все поля заполнены

да

нет

ID совпадают?

да

нет

ID

совпадают

Конец таблицы?

нет

Запись введенных данных

Запись сохранена

Функция “Смена пароля”

НАЧАЛО

КОНЕЦ

Сохранение нового пароля

Ввод нового и старого пароля

Проверка старого пароля

да

нет

Функция “Редактирование”

НАЧАЛО

Вывод полей таблицы для редактирования

Выбор поля для редактирования

Запись изменений в таблицу

КОНЕЦ

Изменение полей

Успешное окончание операции

Функция “Удаление”

НАЧАЛО

КОНЕЦ

Выбор записи

В таблице

Ввод выбранных данных

Удаление записи из БД

Функция “Экспорт в Exсel”

НАЧАЛО

КОНЕЦ

Клавиша Export

Данные из таблицы

Данные пусты?

Формируем объект данных

Файл Excel

ДА

Нет

Функция “Поиска”

НАЧАЛО

КОНЕЦ

Клавиша поиска

Данные поиска

Выделение найденной записи в таблице

Объект пуст?

нет

да

Функция “Сортировка”

Алгоритм сортировки основан на внутренней реализации метода Sort объекта datagridview.

НАЧАЛО

Обработчик события по полю таблици

Объект обработчик

КОНЕЦ

Метод Sort объекта datagridwiev

Отсортированные данные

2.3 Руководство пользователя

В данном курсовом проекте реализовано приложение “Ведене базы компьютеров и оргтехники”, которое позволяет облегчить задачу поиска необходимой информации, как для пользователя, так и для администратора.

Для запуска приложения воспользуйтесь graduate\_work.exe. На рисунке 1 представлено окно авторизации пользователя, которое открывается при запуске программы. Для входа под Пользователем выберите пользователя «user». Чтобы войти в систему, как Администратор, выберите пользователя “admin” и введите пароль, затем нажмите на кнопку «ОК» или «Отмена» для выхода из программы.

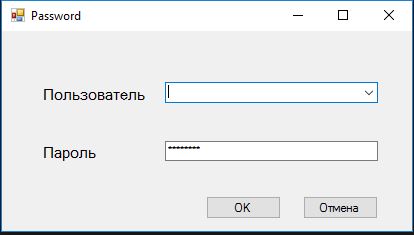


Рисунок 1 – Окно авторизации

На рисунке 2 показано, если при нажатии на кнопку входа появляется сообщение об ошибке, проверьте, выбран ли пользователь, также проверьте раскладку клавиатуры, не включен ли у вас «Caps Lock».

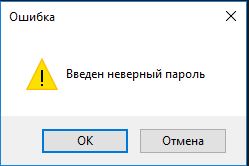


Рисунок 2 – Сообщение об ошибке ввода пароля

После входа в систему вы увидите главное окно программы, показанное на рисунке 3.

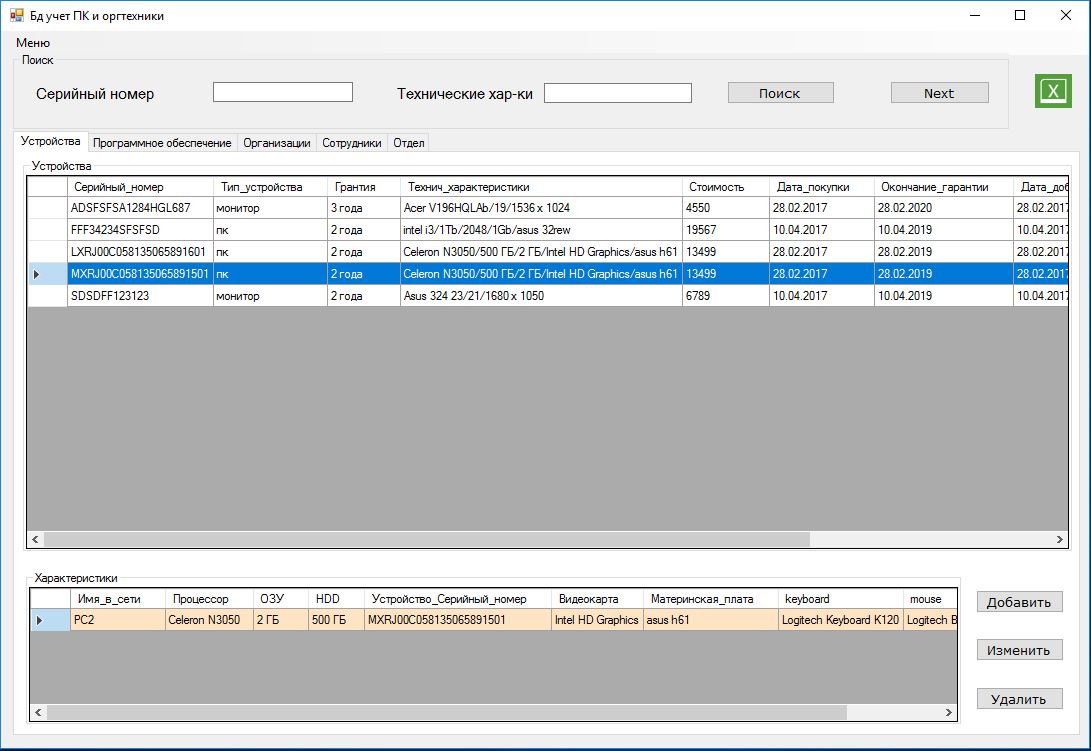


Рисунок 3 – Главное окно

На рисунке 4 показано главное окно, разделенное на вкладки:

1. Устройства;
2. Программное обеспечение;
3. Организации;
4. Сотрудники;
5. Отдел.

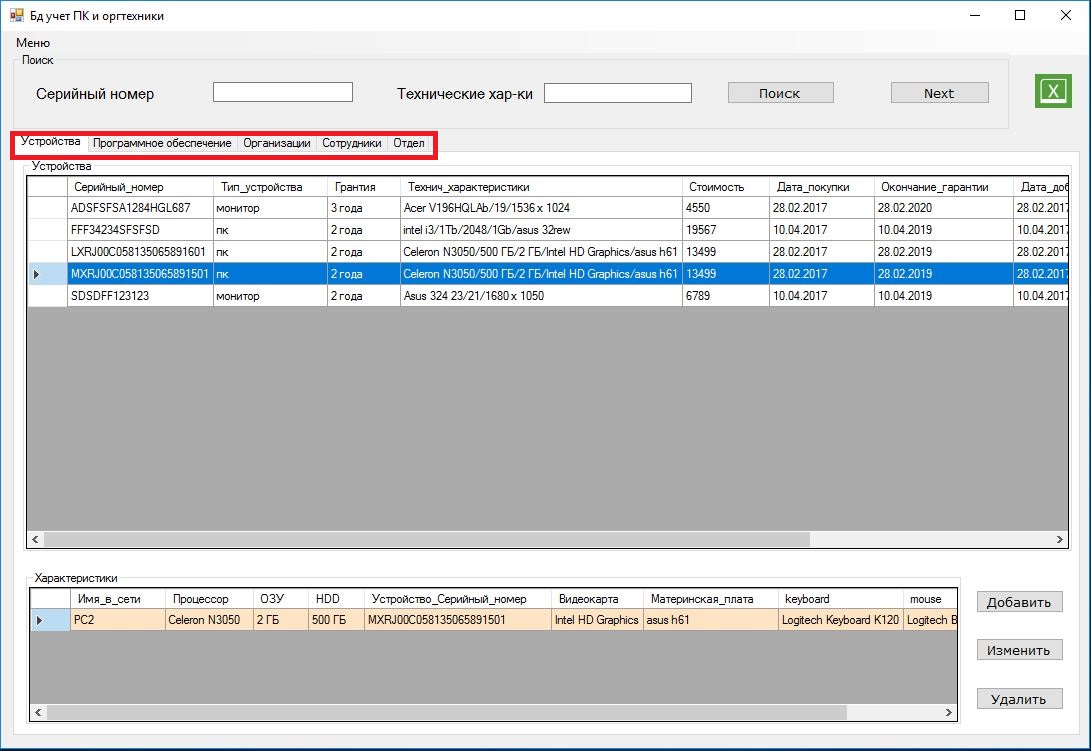


Рисунок 4 – Вкладки главной таблицы

Как показано на рисунке 5 каждой вкладке соответствует своя главная таблица. На рисунке 6 для более удобного поиска имеется дополнительная таблица, которая показывает записи соответствующие записи в выделенной записи в главной таблице.

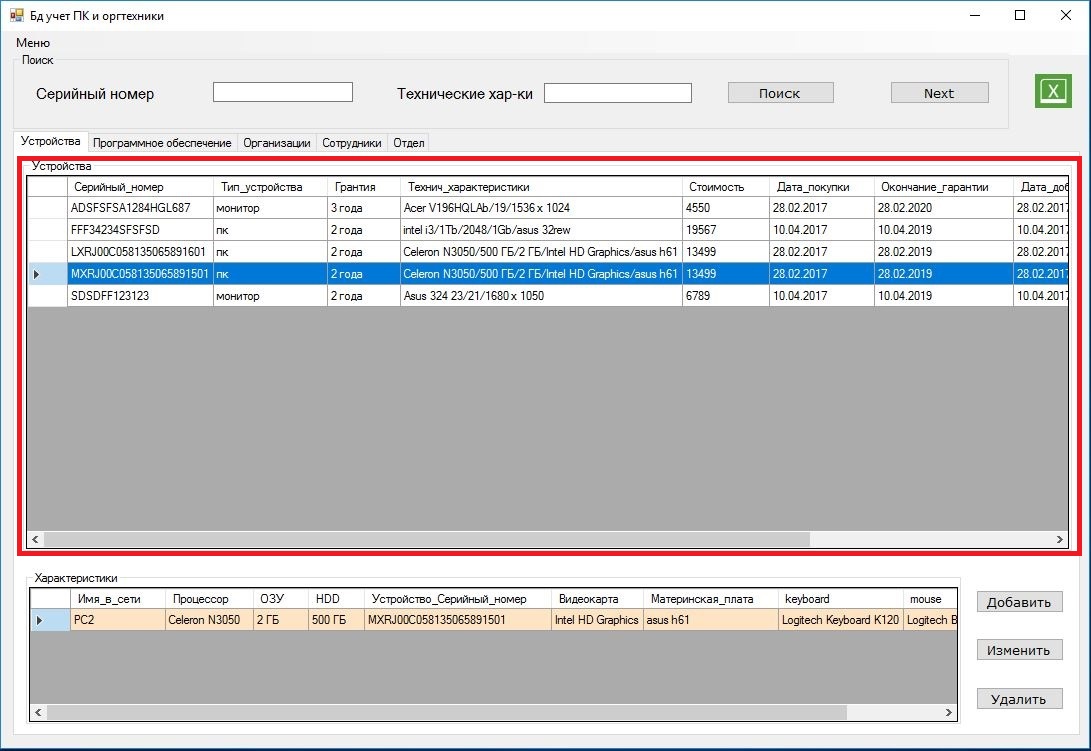


Рисунок 5 – Главная таблица вкладки

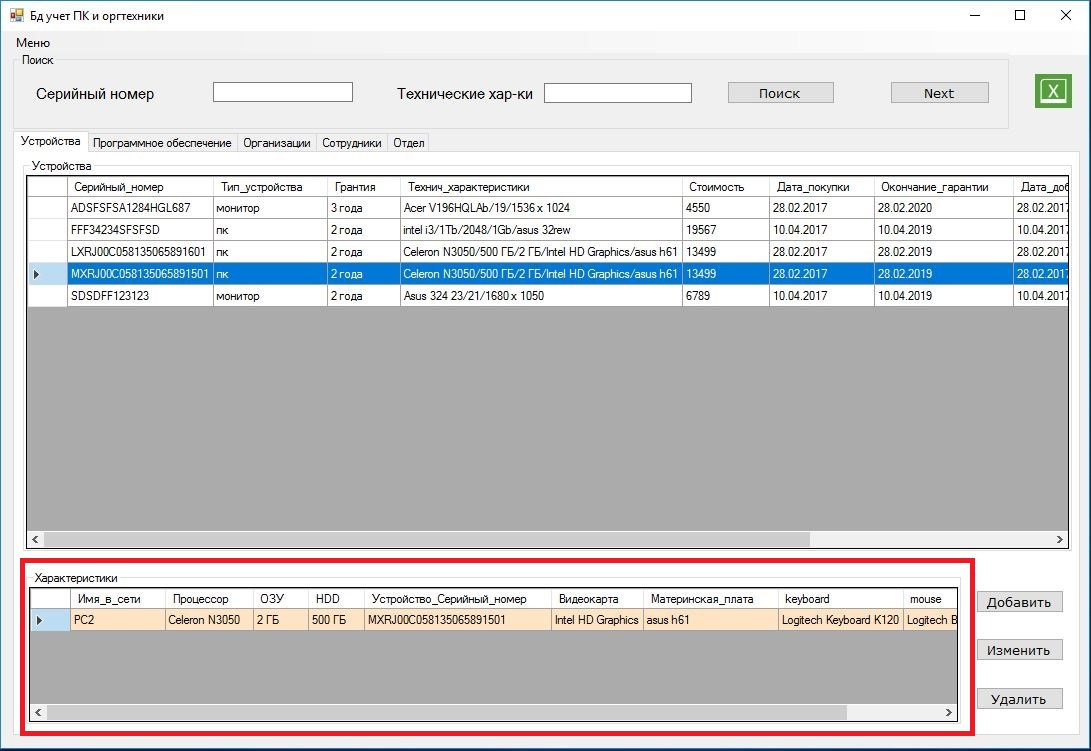


Рисунок 6 – Дополнительная таблица вкладки

В главном окне имеется поиск. Чтобы воспользоваться поиском нужно выбрать любую вкладку и вести данные которые хотите найти и нажать кнопку «Поиск» как показано на рисунке 7, для того чтобы пролистать все найденные строки в главной таблице нужно нажать кнопку «Next». В каждой вкладке имеются свои поля для поиска. Если не чего не найдено, выводится диалоговое окно (рисунок 8) «Не найдено».

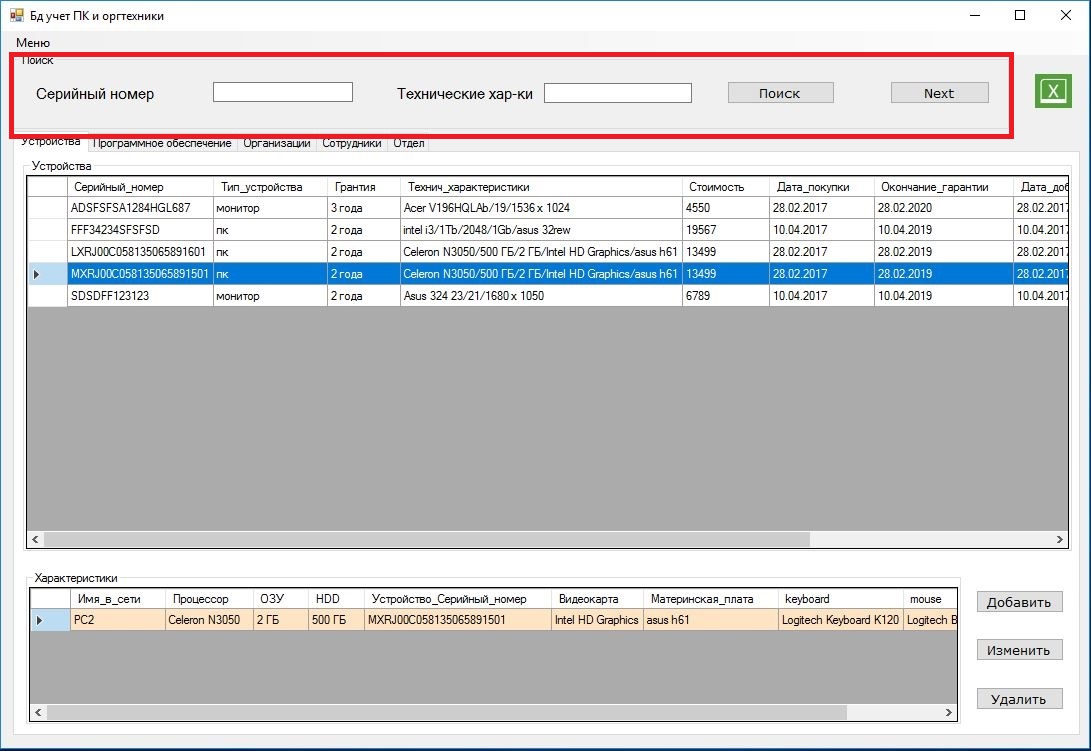


Рисунок 7 – Поиск по главной таблице

Если вы хотите сменить пароль для «Пользователя» или «Администратора», нужно выбрать в программе выпадающий список «Меню» (рисунок 9) и выбрать «сменить пароль» или «выход» для выхода из программы.

Внимание! Смена пароля осуществляется только под Администратором.

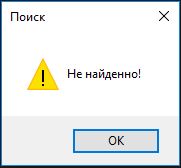


Рисунок 8 – Диалоговое окно сообщающее,  
 что при поиске ничего не найдено

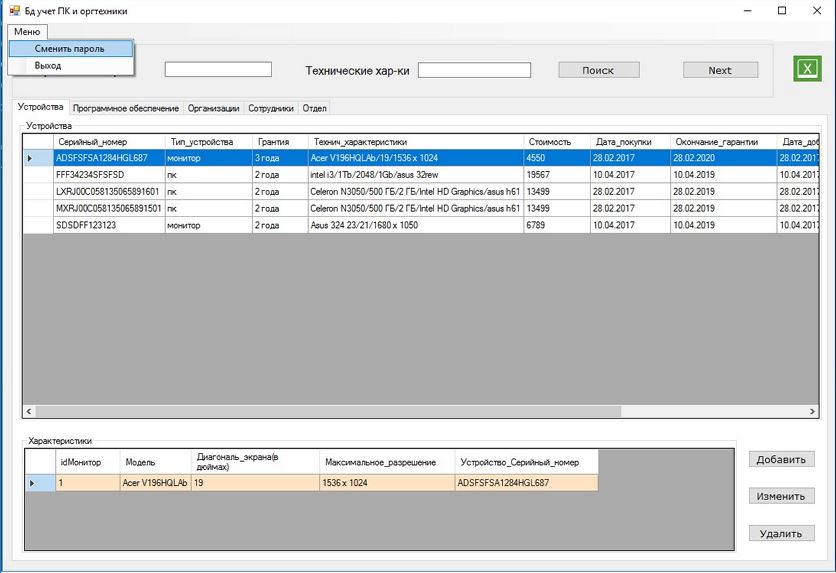


Рисунок 9 – Выпадающее меню

После нажатия «меню» и «сменить пароль» вы падает диалоговое окно «Смена пароля» (рисунок 10) где нужно выбрать пользователя для которого будет производиться смена пароля, и указать в текстовых полях старый пароль текущего выбранного пользователя, и указать новый пароль для пользователя после чего нажмите «OK» для смены пароля или «Отмена» для выхода и закрытия окна «Смена пароля». Если ввести не правильный старый пароль или не указать пользователя появится предупреждение «Введен неверный пароль или не указан пользователь» (рисунок 11).

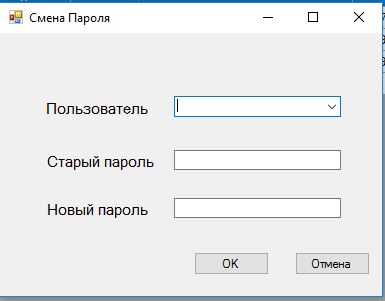


Рисунок 10 – Смена пароля

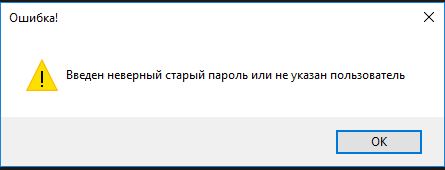


Рисунок 11 – Ошибка при смене пароля

Для добавления, удаления, изменения данных, выберете нужную вкладку в главном окне программы и выберете нужную кнопку (рисунок 12). Внимание! Изменять, добавлять, удалять и менять пароль, выполняется только под Администратором, под Пользователем можно только в режиме просмотра данных.

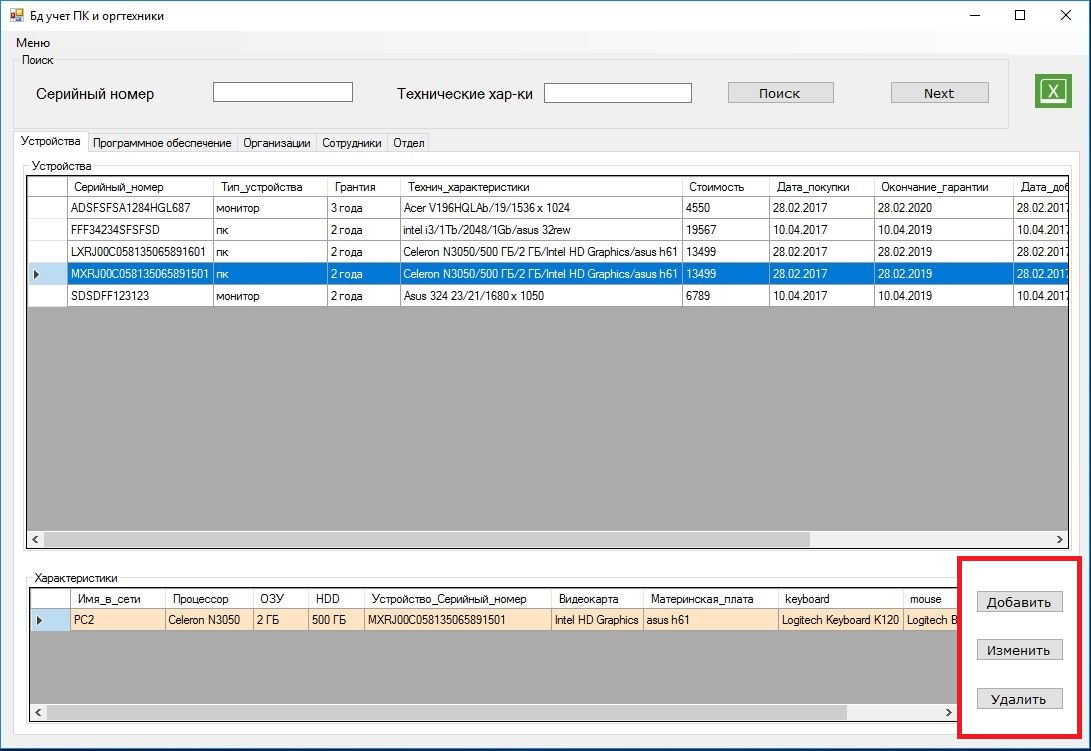


Рисунок 12 – Кнопки для добавления, изменения и удаления данных

Нажмите кнопку «Добавить» что бы добавить запись в таблицу. После нажатия появится форма для заполнения данных (рисунок 13). Каждой вкладке соответствует своя форма для заполнения данных. Если вы заполнили все поля, нажмите кнопку «Добавить» (рисунок 13). Каждое текстовое поле в форме для заполнения данных должно быть заполнено, если текстовое поле не заполнено возникает диалоговое окно с ошибкой (рисунок 14). Если текстовые поля заполнены, после успешного добавления данных в БД возникает диалоговое окно с сообщением «Данные добавлены» (рисунок 15). При этом главная таблица автоматически обновляется.

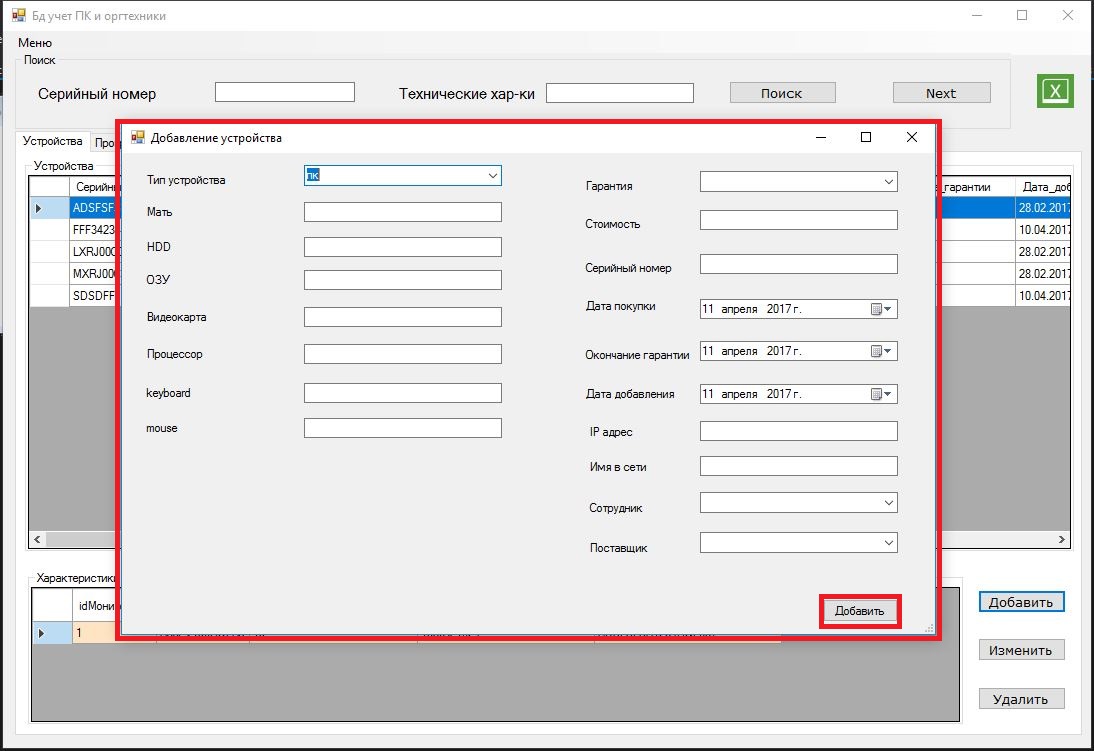


Рисунок 13 - Добавление данных на примере вкладки «Устройства»

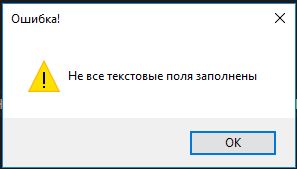


Рисунок 14 – Диалоговое окно при незаполненных полях

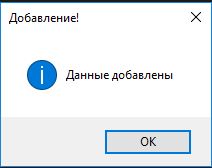


Рисунок 15 – Диалоговое окно при успешном добавлении данных в БД

Что бы изменить запись нужно выбрать нужную вкладку и выделить в главной таблице строку, которую хотите изменить и нажать кнопку «Изменить». После нажатия появится форма для изменения данных, как показано на рисунке 16. Форма автоматически заполняется данными. Текстовые поля в форме не должны быть пустыми иначе появляется диалоговое окно «Ошибка!», как показано на рисунке 14. При успешном изменение выпадает сообщение «данные изменены» (рисунок 17).

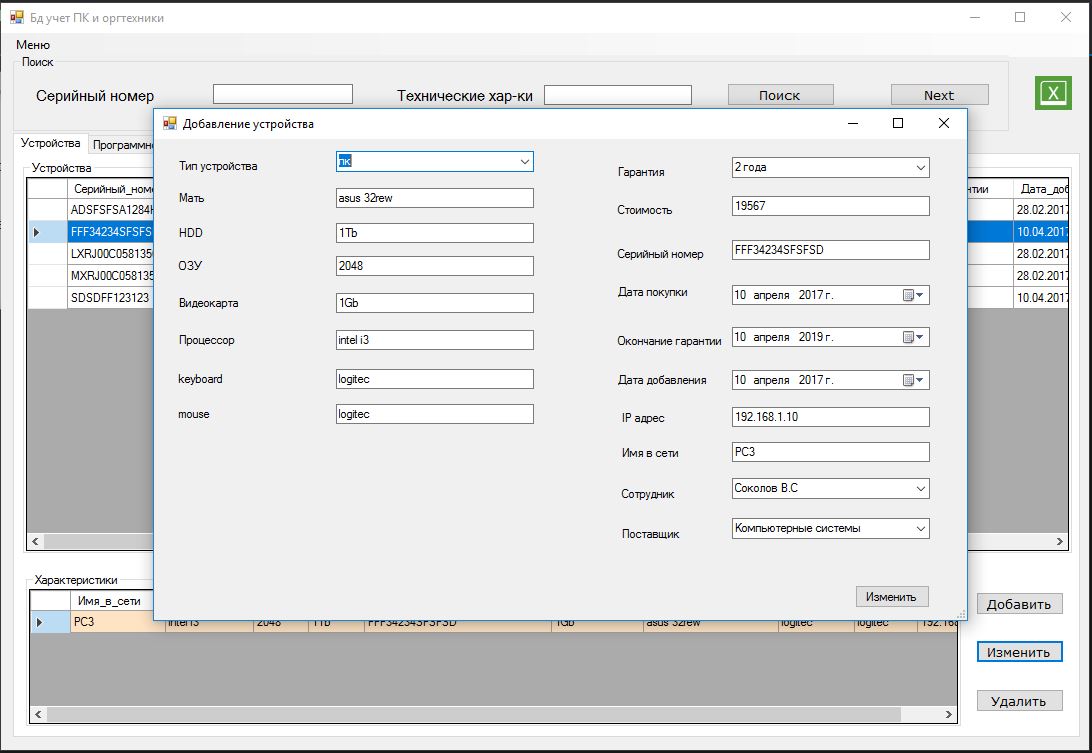


Рисунок 15 – Изменение данных

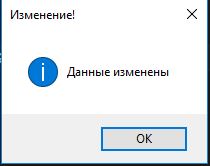


Рисунок 17 – Диалоговое окно при успешном изменение данных в БД

Чтобы удалить запись нужно выбрать нужную вкладку и выделить в главной таблице строку, которую хотите изменить и нажать кнопку «Удалить». После нажатия появится диалоговое окно с вопросом для подтверждения на удаление записи в главной таблице (рисунок 18). При нажатии на кнопку «OK» для подтверждения или «Отмена» для отмены удаления. На рисунке 19 показано диалоговое окно с информацией «Данные удалены».

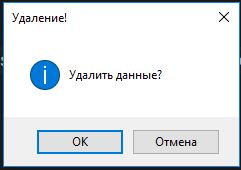


Рисунок 18 – Удаление данных

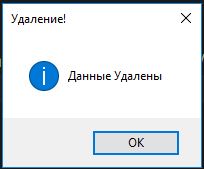


Рисунок 19 – Успешное удаление данных

Для экспорта данных таблицы в MS Excel. Нужно нажать кнопку Excel (рисунок 20). После нажатия появится окно выбора пути для сохранения Excel файла и указания формата(xls, xlsx) и названия файла (рисунок 21). После экспорта можно открыть файл Excel. Экспортные данные показаны на рисунок 22.

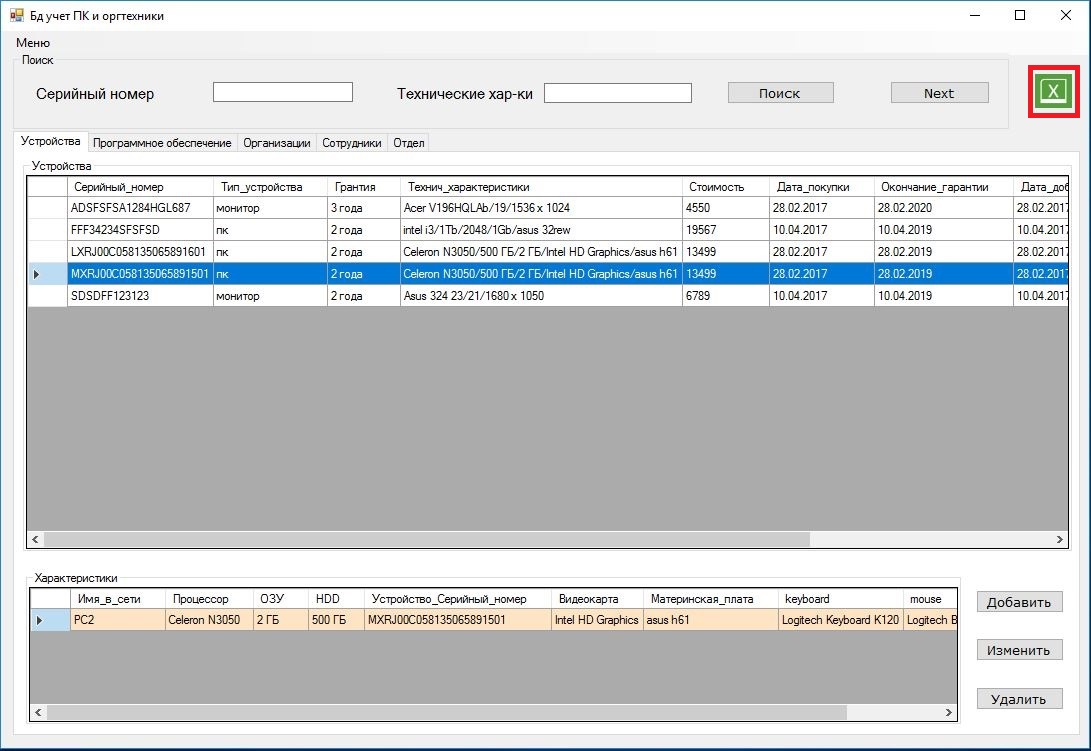


Рисунок 20 – Кнопка экспорта данных таблицы в MS Excel

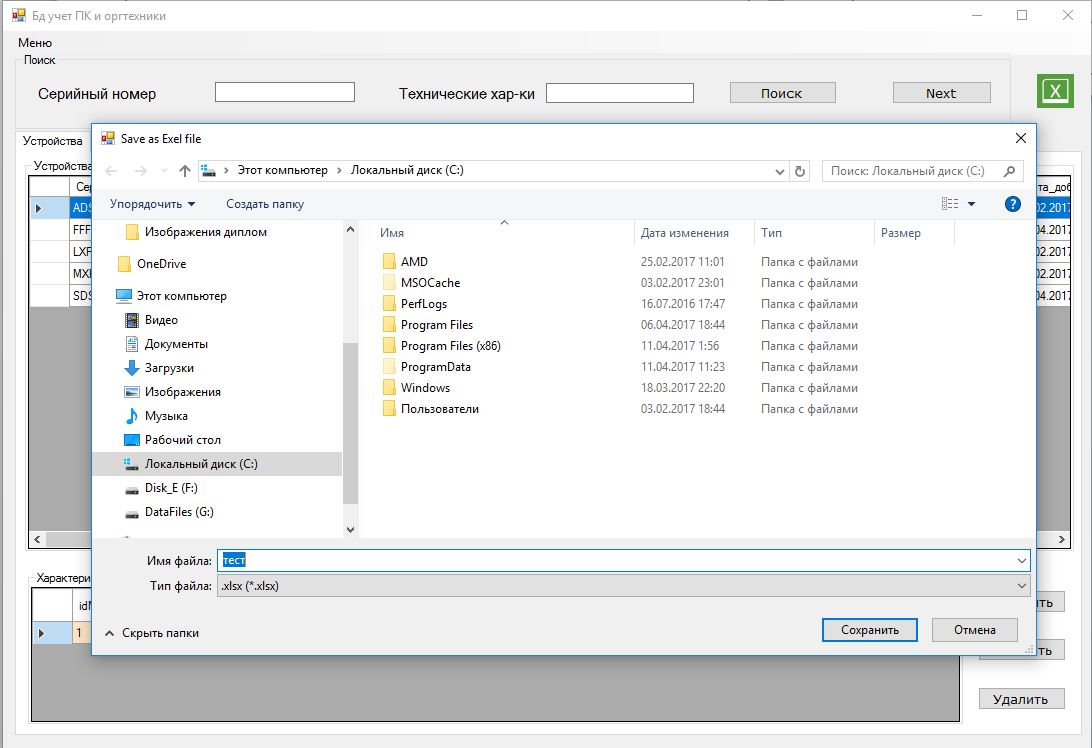


Рисунок 21 – Выбор пути для экспорта данных таблицы в MS Excel

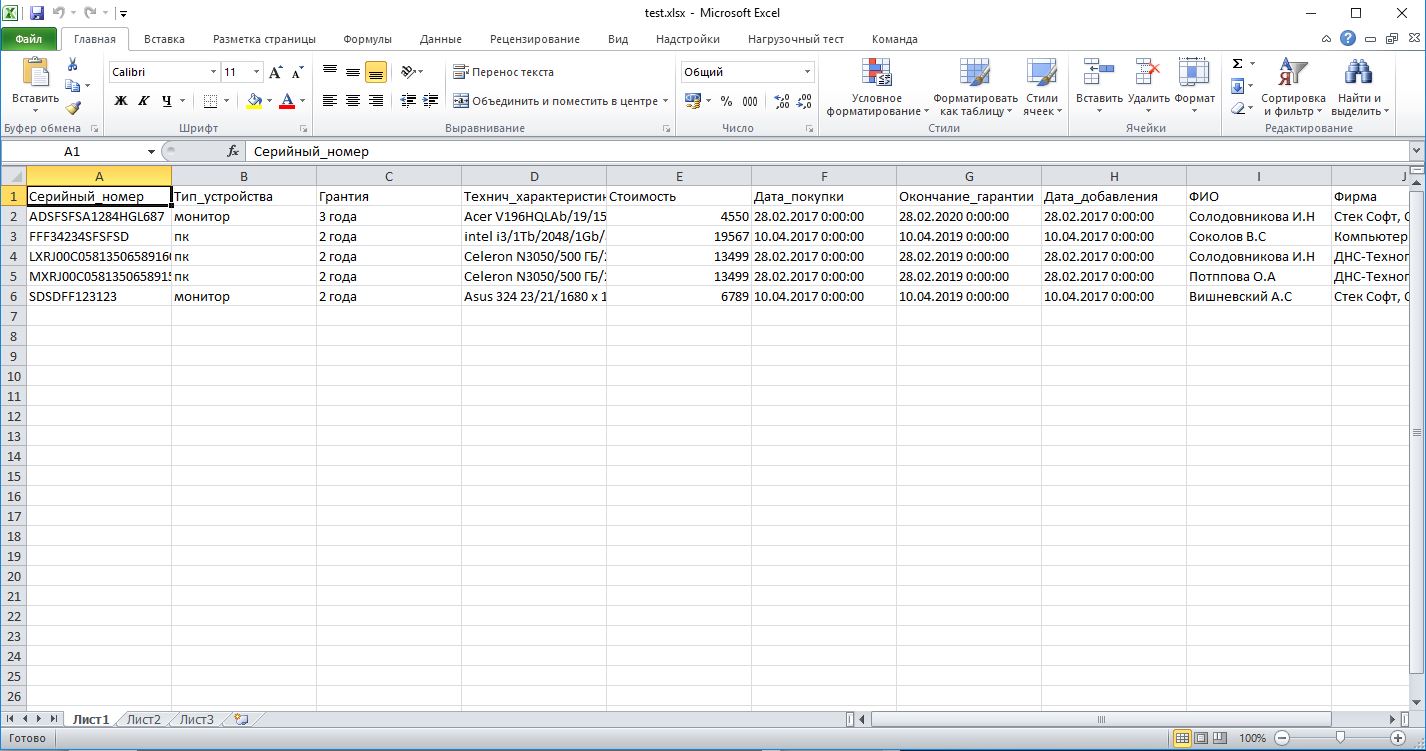


Рисунок 22 – Файл MS Excel экспортированный из таблицы

Для сортировки нужно нажать по любому полю в главной таблице. Форма таблицы показана на рисунке 23.

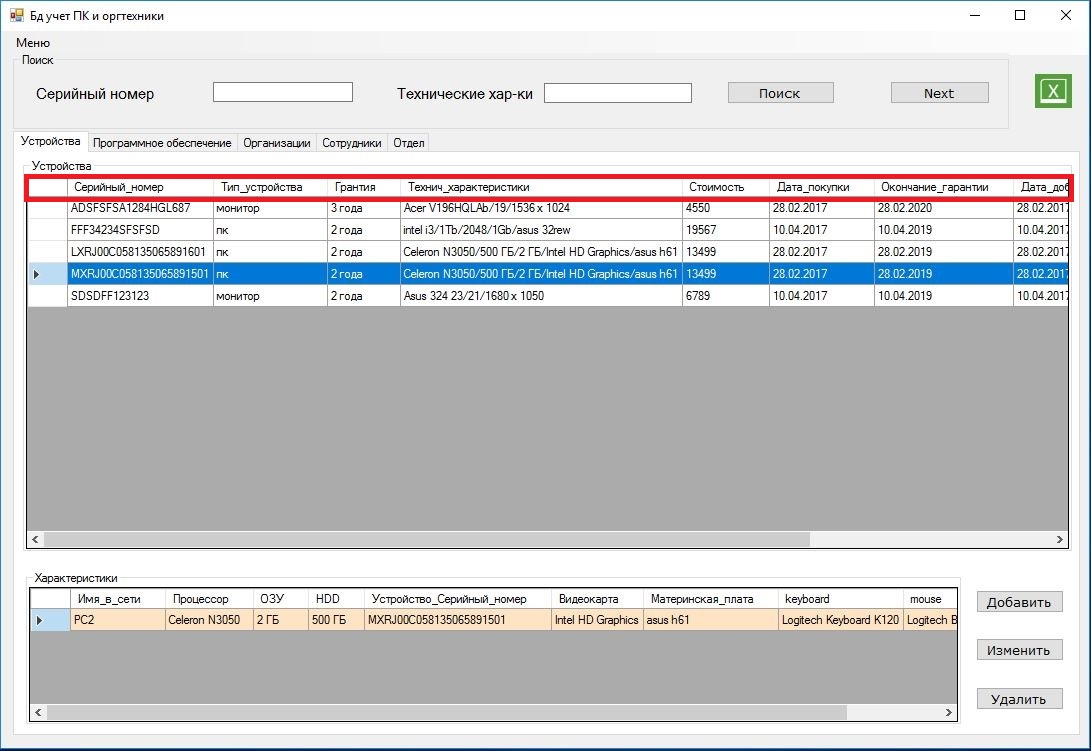


Рисунок 23– Сортировка по полю таблицы

3 ЭКОНОМИЧЕСКАЯ ЧАСТЬ

3.1 Расчет затрат на разработку программы и решение задачи на ЭВМ

Основными компонентами затрат на разработку программ и решение задачи на ЭВМ являются затраты, связанные с оплатой труда специалистов на разработку программы, обслуживание и эксплуатацию ЭВМ в период отладки программы и решения задачи, то есть рассчитываются прямые и косвенные затраты.

При определении полной себестоимости программы учтены все материальные расходы, расходы по заработной плате, отчисления в социальные статьи и составлена калькуляция затрат в следующей последовательности:

- основная заработная плата персонала;

- дополнительная заработная плата персонала;

- отчисления в социальные статьи;

- стоимость работ на ВЦ;

- расчет стоимости материалов;

- расчет косвенных затрат на разработку программы.

При расчете всех экономических показателей была составлена таблица 1, в которой указаны все этапы работы по разработке программы и решению задачи, исполнитель каждого этапа, трудоемкость и стоимость исполнения.

Стоимость каждого этапа определена, исходя из оклада исполнителей и времени выполнения этапа.

Количество рабочих часов в месяце равно 168 часов, то есть 21 рабочий день в месяце по 8 часов.

Стоимость часа работы руководителя и программиста определяется по следующим формулам 1 и 2.

Ст.ч.р = Оклад.р / К.р.ч, (1)

где Оклад.р – оклад руководителя, руб.;

Ст.ч.р. – стоимость часа работы руководителя;

К.р.ч. – количество рабочих часов в месяце, час.

*Стоимость часа работы руководителя*

Ст.ч.р = 25600/ 168 = 152,3 руб./час.

Ст.ч.п=Оклад.п / К.р.ч, (2)

где Ст.ч.п – стоимость часа работы программиста, руб.;

Оклад.п – оклад программиста;

К.р.ч – количество рабочих часов в месяце, час.

*Стоимость часа работы программиста*

Ст.ч.п = 15000 / 168 = 89,2 руб./час.

Таблица 1 – Этапы разработки

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование этапов работ | Исполнитель | Трудоемкость (час) | Оклад (руб.) | Плата за час (руб.) | Стоимость исполнения |
| Постановка задачи | Руководитель | 8 | 25600 | 152,3 | 1218,4 |
| Программист | 8 | 15000 | 89,2 | 713,6 |
| Изучение литературы | Программист | 46 | 15000 | 89,2 | 4103,2 |
| Технический проект | Программист | 16 | 15000 | 89,2 | 1427,2 |
| Руководитель | 8 | 25600 | 152,3 | 1218,4 |
| Эскизный проект | Программист | 23 | 15000 | 89,2 | 2051,6 |
| Руководитель | 15 | 25600 | 152,3 | 2284,5 |
| Кодирование | Программист | 79 | 15000 | 89,2 | 7046,8 |
| ЭВМ | 86 | 10 | 10 | 860 |
| Отладка программы  Тестирование | Программист | 35 | 15000 | 89,2 | 3122 |
| ЭВМ | 40 | 10 | 10 | 400 |
| Руководитель | 23 | 25600 | 152,3 | 3502,9 |
| Оптимизация программы | Программист | 26 | 15000 | 89,2 | 2319,2 |
| Руководитель | 8 | 25600 | 152,3 | 1218,4 |
| ЭВМ | 20 | 10 | 10 | 200 |
| Оформление сопроводитель  ной документации | Программист | 37 | 15000 | 89,2 | 3304,4 |
| Руководитель | 32 | 25600 | 152,3 | 4873,6 |
| ЭВМ | 32 | 10 | 10 | 320 |
| Итого: | Программист | 270 |  |  | 24106 |
| Руководитель  ЭВМ | 94  178 |  |  | 14316,2  1780 |

Основная заработная плата персонала рассчитывается по формуле

ЗПо = ЗПч × Тч × Кр, (3)

где ЗПч – заработная плата за час;

Тч – количество отработанных при разработке программы часов;

ЗПо  - основная заработная плата персонала;

Кр – районный коэффициент.

*Основная заработная плата руководителя*

ЗПо = 152,3× 94 × 1,3 = 18611 руб.

*Основная заработная плата программиста*

ЗПо = 89,2 × 270 × 1,3 = 31309,2 руб.

Дополнительная заработная плата персонала рассчитывается по формуле

ЗПдоп = ЗПо × 0,10, (4)

где ЗПдоп  – дополнительная заработная плата персонала;

ЗПо – основная заработная плата персонала, руб.

*Дополнительная заработная плата руководителя*

ЗПдоп = 18611× 0,1 = 1861.1 руб.

*Дополнительная заработная плата программиста*

ЗПдоп = 31309,2 × 0,1 = 3130.9 руб.

Отчисления в социальные статьи рассчитываются по формуле

Осс=(ЗПо+ЗПдоп)×0,30, (5)

где ЗПдоп  - дополнительная заработная плата;

ЗПо – основная заработная плата;

Осс  - отчисления в социальные статьи.

Отчисления в социальные статьи составляют 30%, из них:

- в пенсионный фонд(Опф) отчисляется 22%;

- в фонд социального страхования(Осстр) – 2,9%;

- в фонд медицинского страхования(Омс) – 5,1%.

Отчисления в социальные статьи от заработной платы руководителя рассчитывается по формуле 5

где Опф – отчисления в пенсионный фонд;

Осстр – отчисления в фонд социального страхования;

Омс – отчисления в фонд медицинского страхования;

Осс – отчисления в социальные статьи – это сумма отчислений в пенсионный фонд, отчислений в фонд социального страхования и отчислений в фонд медицинского страхования.

Опф= (18611 + 1861.1) × 0,22 = 4503.8 руб.;

Осстр= (18611 + 1861.1) × 0,029 = 593.6 руб.;

Омс= (18611 + 1861.1) × 0,051 = 1044 руб.;

Осс= 4503.8 + 593.6 + 1044 = 6141.4 руб.

Отчисления в социальные статьи от заработной платы программиста рассчитывается по формуле 5:

Опф= (31309,2 + 3130,9) × 0,22 = 7576,8 руб.;

Осстр= (31309,2 + 3130,9) × 0,029 = 998,7 руб.;

Омс= (31309,2 + 3130,9) × 0,051 = 1756,4 руб.;

Осс= 7576,8 + 998,7 + 1756,4 = 10331,9 руб.

Стоимость работ на ВЦ рассчитывается по формуле 6.

Срм=Смч×Тм, (6)

где Смч – стоимость машинного часа в рублях (10 руб./час);

Срм – стоимость работ на ВЦ;

Тм – общее время работы ЭВМ (час).

*Стоимость работ на ВЦ*

Срм = 10 × 178 = 1780 руб.

Расчет стоимости материалов приведен в таблице 2.

Таблица 2 – Материальные затраты

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование | Единица измерения | Количество | Цена за 1 ед., руб | Стоимость, руб. |
| Бумага (А4, 500 л, белая) | уп. | 1 | 300 | 300 |
| Тетрадь | шт. | 1 | 5 | 5 |
| Карандаш | шт. | 1 | 10 | 10 |
| Ручка | шт. | 2 | 15 | 30 |
| Степлер | шт. | 1 | 400 | 400 |
| Мультифора | шт. | 10 | 3 | 30 |
| Итого: |  |  |  | 875 |

Расчет косвенных затрат на разработку программы рассчитывается по формуле 7

Рк = ЗПо × 0,08 руб., (7)

где Рк – расчет косвенных затрат на разработку программы;

ЗПо – основная заработная плата персонала.

Рк = (18611 + 31309,2) × 0,08 = 3993.6 руб.

Полная себестоимость программы приведена в таблице 3.

Таблица 3 – Смета затрат на разработку

|  |  |
| --- | --- |
| Наименование статей расходов | Стоимость работ (руб.) |
| Основная заработная плата  - программиста  - руководителя | 31309,2  18611 |
| Дополнительная заработная плата  - программиста  - руководителя | 3130,9  1861.1 |
| Отчисления в социальные статьи  - программиста  - руководителя | 10331,9  6141.4 |
| Стоимость работ на ВЦ | 1780 |
| Стоимость материалов | 875 |
| Косвенные расходы | 3993.6 |
| Полная себестоимость: | 77989,1 |

1. Расчет годовых затрат на эксплуатацию программы

Стоимость одного непосредственного решения на ЭВМ определяется по формуле

Ср.м=Смч×Тр+ЗПо.п.×Q×Кр×Ккр, (8)

где Ср.м – стоимость одного непосредственного решения на ЭВМ;

Смч – стоимость работы на ЭВМ за час (10 руб./час);

Тр – время решения задачи на ЭВМ (0,005 час);

Q – трудоемкость менеджера (1 час);

Кр – районный коэффициент (1,3);

Ккр – коэффициент косвенных расходов (1,08);

ЗПо.п – заработная плата за час работника (89,2).

*Стоимость одного непосредственного решения на ЭВМ*

Ср.м = 10 × 0,005 + 89,2 × 1 × 1,3 × 1,08 = 125.2 руб.

Расчет годовых затрат на разработку программы необходимо провести для последующего анализа эффективности данного программного продукта.

Готовые затраты на эксплуатацию программы рассчитываются по формуле

Ср.м.год=N×Ср.м+Ен×С, (9)

где N – плотность потока заявок (3500);

Ср.м.год – годовые затраты на эксплуатацию программы;

Ср.м – стоимость одного непосредственного решения на ЭВМ;

Ен – нормальный коэффициент сложности (0,2);

С – себестоимость разработки программы (77989,1).

*Расчет годовых затрат на разработку программы*

Ср.м.год = 3500 × 125.2 + 0,2 × 77989,1= 453797.82 руб.

Расчет годовых затрат на выполнение работ без использования программы по учету материалов для полиграфической компании «4 Цвета».

Расходы на выполнение работ без использования программы по учету материалов для полиграфической компании «4 Цвета» рассчитываются по формуле

Ср.сп=ЗПсп×Тсп×Ккр×Кр, (10)

где ЗПсп – заработная плата работника (100 руб.);

Тсп – затраты времени специалиста на оформление денежного перевода (2 ч);

Кр – районный коэффициент (1,3);

Ккр – коэффициент косвенных расходов (1,08).

*Расходы на выполнение работ без использования электронного учебника*

Ср.сп= 100 × 2 × 1,08 × 1,3 = 280,8 руб.

Зная стоимость всех работ по выполнению одного расчета, определим годовые расходы без использования электронного учебника

Ср.сп.год=N×Ср.сп, (11)

где N – плотность потока заявок (3500);

Ср.сп.год – годовые расходы ранее употреблявшимся способом.

3.2 Расчет годовых затрат на выполнение работ без использования программы по учету материалов

Ср.сп.год = 3500 × 280,8 = 982800 руб.

Эгод = Ср.сп.год – Ср.м.год

где Ср.сп.год – годовые затраты на выполнение работ ранее употреблявшимся способом;

Ср.м.год – годовые затраты на эксплуатацию программы.

Определение коэффициента экономической эффективности программы.

Эгод = 982800 - 453797.82 = 529002.1

Коэффициента экономической эффективности показывает сколько на 1 руб. вложенных затрат в разработку и эксплуатацию, получаем экономии. Чем больше данное значение, тем эффективнее проект.

Ер = Эгод/(С+ Ср.м.год)

Ер = 529002.1 / (77989,1 + 453797.82) = 0.994

Таким образом, на один вложенный рубль в разработку и эксплуатацию, получаем 0.994 экономии.

Определение срока окупаемости

Срок окупаемости показывает через какой период времени возместятся затраты на разработку и эксплуатацию.

Ток = (С+ Ср.м.год)/ Эгод

Ток = (77989,1 + 453797.82) /529002.1= 1.005

Ток = 1/ Ер

Ток = 1/0.994 = 1.005

Срок окупаемости программы за счет экономии средств составил порядка 1(одного) года.

Заключение

Требовалось разработать информационную систему, представляющую собой программу для учета компьютеров, оргтехники и установленного ПО (Программное обеспечение).

В ходе разработки дипломного проекта были реализованы все поставленные задачи, а именно была реализована база данных с последующим приведением ее к третьей нормальной форме и реализован графический интерфейс информационной системы. Реализованы функции хранения, добавления, удаления и изменения данных. Так же реализован экспорт данных в таблицу Excel, поиск и сортировка.

За счет понятного графического интерфейса, программой удобно пользоваться как системному администратору, так и обычному сотруднику.

Программа реализована с учетом тех компаний, где имеется большой парк компьютеров и оргтехники.

Решена проблема которые присущи электронным таблицам ([Microsoft Excel](https://ru.wikipedia.org/wiki/Microsoft_Excel), [OpenOffice](https://ru.wikipedia.org/wiki/OpenOffice.org_Calc)) а именно размер данных, простота обновления, точность, безопасность и проблемы с избыточностью.

Список использованных источников

1. Виктор Гольцман. MySQL 5.0. Библиотека программиста. 1-е издание. - СПб.: Питер, 2016. – 546 с.
2. Джеффри Рихтер. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е издание. - СПб.: Питер, 2013. – 1245 с.

3) Эндрю Троелсен. Язык программирования C# 6.0 и платформа .NET 4.6. 7-e издание. – Москва.: Вильямс, 2016. – 1440 с.