**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

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

**УНИВЕРСИТЕТСКИЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ**

|  |  |
| --- | --- |
|  | **Допустить к защите**  Заместитель директора по  учебно-методической работе  \_\_\_\_\_\_\_\_\_\_\_ М.Г. Рождественская\_\_  (Подпись) (И.О.Ф.)  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 201\_ г. |
|  |

**ДИПЛОМНЫЙ ПРОЕКТ**

Тема АИС инвентаризации программного обеспечения OC MS Windows

специальность 09.02.04 «Информационные системы (по отраслям)»

группа 4293/21

Студент (ка)

(подпись) (ФИО)

Руководитель

(подпись) (ФИО)

Санкт-Петербург­­

2016

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc452488715)

[1 ОБЩАЯ ЧАСТЬ 5](#_Toc452488716)

[1.1 Описание предметной области 5](#_Toc452488717)

[1.2 Анализ методов решения 6](#_Toc452488718)

[1.3 Обзор средств программирования 6](#_Toc452488719)

[1.3.1 СУБД 7](#_Toc452488720)

[1.3.2 Среда разработки 7](#_Toc452488721)

[1.4 Язык программирования 7](#_Toc452488722)

[1.4.1 Описание языка SQL 8](#_Toc452488723)

[1.4.2 Описание языка С# 8](#_Toc452488724)

[1.5 Вывод 8](#_Toc452488725)

[2 СПЕЦИАЛЬНАЯ ЧАСТЬ 9](#_Toc452488726)

[2.1 Постановка задачи 9](#_Toc452488727)

[2.1.1 Основания для разработки 9](#_Toc452488728)

[2.1.2 Назначение программы 9](#_Toc452488729)

[2.1.3 Требования к программе 9](#_Toc452488730)

[2.1.3.1 Требования к функциональным характеристикам 10](#_Toc452488731)

[2.1.3.2 Требования к надёжности 10](#_Toc452488732)

[2.1.3.3 Требования к условиям эксплуатации 10](#_Toc452488733)

[2.1.3.4 Требования к составу и параметрам технических средств 11](#_Toc452488734)

[2.2 Описание схем 11](#_Toc452488735)

[2.3 Текст программы 14](#_Toc452488736)

[2.4 Описание программы 14](#_Toc452488737)

[2.4.1 Общие сведения 14](#_Toc452488738)

[2.4.2 Функциональное назначение 14](#_Toc452488739)

[2.4.3 Описание логической структуры 15](#_Toc452488740)

[2.4.3.1 Описание функций программы 15](#_Toc452488741)

[2.4.4 Вызов и загрузка 16](#_Toc452488742)

[2.4.5 Входные и выходные данные 16](#_Toc452488743)

[2.5 Руководство оператора 16](#_Toc452488744)

[2.6 Программа и методика испытаний 17](#_Toc452488745)

[2.8 Результаты работы программы 17](#_Toc452488746)

[2.9 Вывод 17](#_Toc452488747)

[3 ОБОСНОВАНИЕ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ПРОЕКТА 18](#_Toc452488748)

[3.1 Область применения программного продукта и его преимущества перед аналогичным программным продуктом. 18](#_Toc452488749)

[3.2 Трудоемкость разработки программного продукта, квалификация исполнителя (лей) и его( их) оклад. 18](#_Toc452488750)

[3.3 Расчет затрат на разработку информационных технологий 19](#_Toc452488751)

[3.3.1 Затраты на оплату труда разработчиков (Зот), руб. 20](#_Toc452488752)

[3.3.2 Затраты по страховым вносам(Зст), руб. 21](#_Toc452488753)

[3.3.3 Затраты по содержанию ЭВМ (Зэвм), руб. 21](#_Toc452488754)

[3.4.4 Затраты на специальные программные продукты (Зсп), руб. 21](#_Toc452488755)

[3.3.5 Затраты на расходные материалы (Зрм), руб. 22](#_Toc452488756)

[3.3.6 Затраты по накладным расходам(Знр) 22](#_Toc452488757)

[3.4 Определение цены программного продукта, который разработан одной организаций по заказу другой. 23](#_Toc452488758)

[4 МЕРОПРИЯТИЯ ПО ОБЕСПЕЧЕНИЮ БЕЗОПАСНОСТИ ЖИЗНЕДЕЯТЕЛЬНОСТИ НА ПРЕДПРИЯТИИ АО «НИИ «Рубин» 25](#_Toc452488759)

[4.1 Анализ условий труда администратора 25](#_Toc452488760)

[4.2 Расчет искусственного освещения в помещении 29](#_Toc452488761)

[4.3 Электробезопасность на предприятии 31](#_Toc452488762)

[4.4 Пожарная безопасность на предприятии 33](#_Toc452488763)

[ЗАКЛЮЧЕНИЕ 36](#_Toc452488764)

[СПИСОК ЛИТЕРАТУРЫ 37](#_Toc452488765)

[ПРИЛОЖЕНИЕ А 38](#_Toc452488766)

[ПРИЛОЖЕНИЕ Б 39](#_Toc452488767)

[ПРИЛОЖЕНИЕ В 40](#_Toc452488768)

[Приложение Г 41](#_Toc452488769)

[ПРИЛОЖЕНИЕ Д 52](#_Toc452488770)

[ПРИЛОЖЕНИЕ Е 55](#_Toc452488771)

[ПРИЛОЖЕНИЕ Ж 58](#_Toc452488772)

**Перечень принятых сокращений**

АИС – Автоматизированная Информационная Система.

ОС – Операционная Система.

MS – Microsoft.

ПО – Программное Обеспечение.

ПК – Персональный Компьютер.

СУБД – Система Управления Базами Данных.

UML – Unified Modeling Language.

SQL – Structured Query Language.

ISO – International Organization for Standardization.

БД – База Данных.

# ВВЕДЕНИЕ

Производители программного обеспечения (ПО) тратят много средств и времени на создание и поддержку своих программных продуктов, поэтому их труд зачастую не бесплатен. Но люди не хотят платить за то, что им нужно, особенно во время, когда интернет – повсеместен, и в нем можно найти практически все, что угодно. Поэтому найти пиратское программное обеспечение с обходом лицензии вполне реально. Помимо людей, пользующихся пиратским программными продуктами дома, встречаются организации и/или люди в организациях, которые пользуются умышленно пиратским контентом, что ведет компанию-производителя программного обеспечения в убыток. Получается, что производитель тратит время и деньги за бесплатно, вполне логично, что это плохо и надо пресекать. Для обнаружения изменений в легальном программном обеспечении, а так же мониторинга всех действий, производимых с программами на компьютере (под управление операционной системы (ОС) Microsoft (MS) Windows), было решено написать агента по сбору данных о программах из реестра компьютера. Дальнейший анализ будет проводиться специалистом по информационной безопасности, который будет вести мониторинг изменений в значениях реестра, показываемые данным продуктом. Тем самым выявлять попытки установки лишнего программного обеспечения и/или установка не легального программного обеспечения. Последнее время интернет все же начал вести борьбу с пиратством, выпустив законы, блокируя сайты, но в такой огромной сети нельзя стопроцентно убрать нелегальное программное обеспечения. Разработанный программный продукт можно использовать не только для просмотра изменений в лицензионном ПО, но и для простой проверки на добавленные программные продукты, которых не было ранее. Так можно пресекать какие либо ненужные для работы программы и игры, что повышает производительность работника.

# 1 ОБЩАЯ ЧАСТЬ

## 1.1 Описание предметной области

Трудно представить современную компанию, в которой не нет компьютеров, которые не будут нуждаться в качественном программном продукте. Системный администраторы устанавливают на компьютеры работников нужное программное обеспечение. Но зачастую оно либо устаревает и не обновляется, потому что пользователь не следит за этим, либо пользователь устанавливает дополнительные программные обеспечения, не предусмотренные администратором, что может повлечь за собой проблемы с информационной безопасностью и безопасности данных организации в целом.

АО «НИИ «Рубин» организация с большим штатом, а так же с большим количеством компьютеров, на которых установлено множество программ. Для такой большой организации, что бы облегчить работу системному администратору и не терять время зря, нужно автоматизировать процесс инвентаризации. Целью данного дипломного проектирования является решение данной проблемы инвентаризации ПО для АО «НИИ «Рубин».

Задачи:

* выявление установки пиратского ПО в организации;
* выявление своевольной установки стороннего ПО в организации;
* регистрация изменений с ПО без ведома администратора.

Чтобы решить эти задачи, нужно:

* создать агента по сбору информации о ПО с ПК;
* создание удобного интерфейса для администратора;
* оформить графически.

## 1.2 Анализ методов решения

Существует несколько вариантов для инвентаризации ПО внутри организации:

1. Ручной – данный вариант подразумевает под собой ручную проверку информации о ПО на компьютерах в организации.
2. Автоматический – разработка АИС которая будет в автоматическом режиме вести инвентаризацию ПО на всех компьютерах в организации.

Недостаток ручного метода в том, что для инвентаризации большой организации, с большим количеством ПК, администратор потратит в сотни раз больше времени и сил, чем в случае использования автоматического метода.

Use Case диаграмма работы АИС инвентаризации ПО в OC MS Windows для АО «НИИ «Рубин» представлена в приложении А.

## 1.3 Обзор средств программирования

При разработки информационной системы важно решить какие инструментальные программные средства будут использоваться. В первую очередь будет создана база данных, для хранения данных. Для разработки программ, систем программ, работающих с базами данных, используются специальные средства – системы управления базами данных (СУБД). Далее определиться с языком программирования и средой разработки.

### 1.3.1 СУБД

Существует множество СУБД, но проанализировав поставленную задачу, было принято решение в качестве СУБД выбрать MySQL, т.к. в ней есть все необходимое для поставленной задачи, а именно:

* удаленный доступ;
* разграничение прав доступа;
* высокая скорость работы с запросами;
* поддержка мульти-запросов.

Таблица 1 – Сравнение СУБД

|  |  |  |
| --- | --- | --- |
| Возможности | MySQL | MS Access |
| Кроссплатформенность | + | - |
| Мультизапросы | + | - |
| Удаленный доступ | + | - |
| Формы | - | + |

### 1.3.2 Среда разработки

В качестве среды разработки АИС была выбрана Visual Studio от Microsoft, которая имеет серьезную поддержку, на данный момент интенсивно продвигающая среда разработки, с полной совместимостью с OC Windows.

## 1.4 Язык программирования

Описание языка выполнено в соответствии с ГОСТ 19.506-79 ЕСПД.

### 1.4.1 Описание языка SQL

MySQL имеет очень сложный, но интуитивно понятный и легкий в изучении интерфейс, основанный на SQL (структурированный язык запросов). Язык SQL (Structured Query Language - структурированный язык запросов) представляет собой стандартный высокоуровневый язык описания данных и манипулирования ими в системах управления базами данных (СУБД), построенных на основе реляционной модели данных. Полное описание SQL можно найти в литературе [6].

### 1.4.2 Описание языка С#

C# - объектно-ориентированный язык программирования. Разработан в   
1998—2001 годах в компании Microsoft как язык разработки приложений для платформы Microsoft .NET Framework. Полное описание языка можно найти в литературе [7].

### 1.5 Вывод

Проанализировав предметную область, благодаря анализу решений было принято решение выбрать в качестве СУБД – MySQL, в качестве среды разработки и языка программирования – Visual Studio и C# соответственно.

# 2 СПЕЦИАЛЬНАЯ ЧАСТЬ

## 2.1 Постановка задачи

Постановка задачи выполнена в соответствии с ГОСТ 19.201-78 и заключается в достижении автоматическом сборе информации и разработки АИС для инвентаризации ПО в MS OC Windows.

### 2.1.1 Основания для разработки

Основанием для разработки послужило задание, выданное руководителем дипломного проектирования.

### 2.1.2 Назначение программы

Программа предназначена для инвентаризации программного оборудования и регистрации нелегального ПО.

### 2.1.3 Требования к программе

Основные требования к программе выполнены в соответствии с ГОСТ 19.201-78.

#### 2.1.3.1 Требования к функциональным характеристикам

Программа должна выполнять следующие функции:

* Собирать данные из реестра;
* Вносить собранные данные в базу;
* Различать компьютеры, на которых установлена программа-агент;
* Выводить информацию обо всех значениях реестра;
* Выводить измененные значения;
* Показывать период времени, когда были произведены изменения;
* Иметь интуитивно понятный интерфейс.

#### 2.1.3.2 Требования к надёжности

Данная АИС должна обеспечивать надежный сбор информации, не меняя параметров в реестре во избежание отказа и/или поломки всей ОС Windows. Надежный вывод отчетов из полученных данных, не меняя их первоначальные значения для получения точных результатов.

#### 2.1.3.3 Требования к условиям эксплуатации

Для клиентской стороны основным требованием является наличие ПК с установленной OC Windows.

Для серверной части основным требованием является установленный MySQL сервер, и наличие созданной оболочки, для вывода информации администратору.

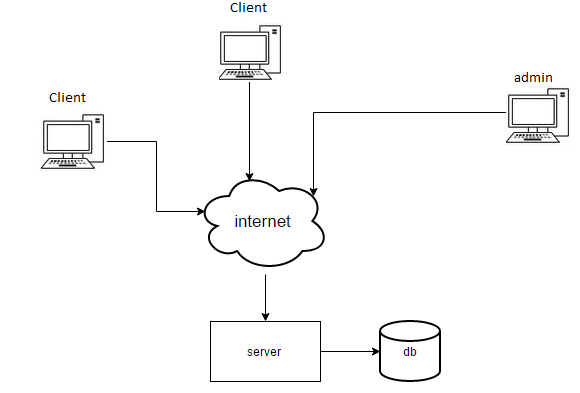


Рисунок 1 – Схема клиент-сервер

#### 2.1.3.4 Требования к составу и параметрам технических средств

Для функционирования АИС рекомендуется следующая конфигурация пользовательского компьютера:

1. Компьютер с процессором не менее Pentium IV 1ГГц (рекомендуется от 1.5ГГц) ОЗУ: 128 Мб или выше;
2. операционная система Windows XP или выше;
3. доступ в интернет.

Данные требования были выявлены на этапе отладки программ и соответствию специфики работы программного продукта.

## 2.2 Описание схем

Схема данных представлена в Приложении Б.

Схема работы агента представлена в Приложении В.

Для реализации функционирования программного комплекса была разработана следующая структура данных, реализованная в виде взаимосвязанных таблиц:

1. Таблица «History»

Таблица «History» предназначена для хранения информации, при изменении в реестровых параметров программных продуктов в OC Windows. Связана с таблицей «PCInfo» полем «id\_pc» с вторичным ключом.

Таблица 1 – History

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Назначение |
| \*ID | int | Id ПО в БД |
| Id\_pc | int | Id компьютера |
| DisplayName | text | Название программы |
| DisplayVersion | text | Версия ПО |
| Publisher | text | Издательство |
| Version | text | Программная версия ПО |
| InstallDate | date | Дата установки ПО |
| ReviewDate | date | Дата проверки |
| Note | text | Заметки |

1. Таблица «PCInfo»

Таблица «PCInfo» предназначена для хранения информации о компьютерах на которых хоть раз была запущенна программа-агент. Связанна с таблицами «History» «LogError» полем «id» с первичным ключом.

Таблица 2 – Авторы

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Назначение |
| \*id | int | Id компьютера в БД |
| HostName | text | Имя компьютера |
| IP | text | IP компьютера |
| MAC | text | MAC компьютера |
| Note | text | Заметки |

1. Таблица «LogError»

Таблица «LogError» необходима для сохранений логов ошибок при работе Агента. Связана с таблицей «PCInfo» полем «id\_pc» с вторичным ключом.

Таблица 3 – LogError

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Назначение |
| \*id | int | Id ошибки в БД |
| \*id\_pc | int | Id компьютера |
| ErrorText | text | Текс ошибки |

## 2.3 Текст программы

Текст программы выполнен в соответствии с ГОСТ 19.101-77 и ГОСТ 19.401-78 и представляет собой запись программы на исходном языке программирования с необходимыми комментариями. АИС разработана на языке C#. Листинг кода представлен в приложении Г.

## 2.4 Описание программы

## 2.4.1 Общие сведения

В дипломном проекте реализована автоматизированная информационная система (АИС) для компании АО «НИИ «Рубин».

Для функционирования АИС необходимы следующие характеристики ПК:

* Компьютер с процессором не менее Pentium IV 1ГГц (рекомендуется от 1.5ГГц) ОЗУ: 128 Мб или выше;
* операционная система Windows XP или выше;
* доступ в интернет.

.

### 2.4.2 Функциональное назначение

АИС предназначена для инвентаризации программного обеспечения в операционных системах семейства Windows. Основной задачей данного продукта является получение информации о программах и меткой времени их изменения, если таковы были зарегистрированы программой-агентом.

АИС обеспечивает следующие возможности:

* ведение базы данных с информацией об изменениях в ПО;
* ведение базы данных с информацией о ПК;
* вывод отчетов для администратора.

### 2.4.3 Описание логической структуры

#### 2.4.3.1 Описание функций программы

Программный продукт должен обеспечивать возможность выполнения перечисленных ниже функций:

* Агент:

1. Собирать информацию из реестра о программах;
2. Отправлять на сервер собранные данные;
3. Фиксировать время проверки (временная метка);
4. При отсутствии сети сохранять информацию для дальнейшей пересылке на сервер;
5. Регистрировать компьютер, на котором происходит сбор информации.

* Программа «оболочка»:

1. Выводить информацию о компьютерах на которых стоит (стояла) программа Агент;
2. Выводить отчет об изменениях в программных продуктах на выбранных компьютерах;
3. Фильтровать программы по названию, производителю и версии;
4. Выводить изменения за выбранный период времени.

### 2.4.4 Вызов и загрузка

АИС вызывается нажатием на иконку программы из любого места в системе.

Программа-агент запускается единожды, после чего будет вызываться автоматически благодаря автозапуску.

### 2.4.5 Входные и выходные данные

Входными данными являются данные для фильтрации при выводе отчетов:

* название программного обеспечения;
* версия программного обеспечения;
* автор программного обеспечения;
* Дата, от которой выводить изменения и дата, до которой выводить изменения.

Выходными данными являются информация о ПО, такая как: название, версия, автор, программная версия, дата установки и дата инвентаризации – все данные выводятся в отчет. Выводимые данные имеют разновидную фильтрацию, что упрощает поиск нужного ПО.

## 2.5 Руководство оператора

Данный пункт по структуре соответствует программному документу «Руководство оператора» (ГОСТ 19.505-79). Документ «Руководство оператора» представлен в Приложении Д.

## 2.6 Программа и методика испытаний

Программа и методика испытаний оформлена в соответствии с 19.301-79 ЕСПД. Документ «Руководство оператора» представлен в Приложении Е.

## 2.8 Результаты работы программы

Результатами работы программы являются сбор информации при изменениях на компьютере «клиента» с дальнейшей передачей их на сервер и сохранением в БД, так же отчеты об этих изменениях в графическом виде для администратора и фильтрацией информации. Результаты работы представлены в приложение Ж.

## 2.9 Вывод

Реализованный программный комплекс позволяет вести наблюдения за программными продуктами на компьютерах внутри организации, что позволяет ускорить инвентаризацию всех ПК в организации.

# 3 ОБОСНОВАНИЕ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ПРОЕКТА

## 3.1 Область применения программного продукта и его преимущества перед аналогичным программным продуктом.

АИС инвентаризации ПО в MS OC Windows применяется в организации «НИИ «Рубин», которая является непосредственно заказчиком данного продукта, весь функционал выполнен по требованию заказчика. Существующие аналоги, такие как: ClearApps, Total Network Inventory 3 – имеют сложный интерфейс и много не нужных функций, которые лишние для заказчика, так же они работают через удаленный доступ, что сильно сказывается на безопасности способа, и могут повлечь нарушение целостности реестра, что крайне неприятно может отразиться на работе всей системы, к которой подключились удаленно.

## 3.2 Трудоемкость разработки программного продукта, квалификация исполнителя (лей) и его( их) оклад.

Трудоемкость разработки можно определить в таблице 3.1, т.е. строка «всего» показывает общую трудоемкость разработки.

Таблица 3.1 - Трудоемкость разработки программного продукта

|  |  |  |
| --- | --- | --- |
| **Наименование этапа** | **Условное обозначение** | **Трудоемкость выполнения этапа, (час., день, месяц)** |
| Описание задания | То | 3 |
| Написание технического задания | Ттз | 5 |
| Анализ методов решений | Тмр | 4 |
| Создание БД | Tбд | 3 |
| Создание Агента | Та | 10 |
| Создание основной программы | Тпо | 15 |
| Оформление документации | Тд | 4 |
| Всего (общая трудоемкость разработки) | | 44 |

Таблица 3.2 - Разработчики программного продукта

|  |  |  |
| --- | --- | --- |
| **Исполнители** | **Оклад, руб.** | **Часовая тарифная ставка, руб.** |
| Системный администратор | 4000 | 150 |
| Разработчик | 6000 | 150 |

Часовая тарифная ставка определяется исходя из месячного оклада, количества рабочих дней в месяце и продолжительности рабочего дня по формуле:

**ЧТС = , руб./час. (1)**

Где**: ЧТС**- часовая тарифная ставка, руб./час.;

**Ом**- оклад исполнителя в месяц, руб. /месс.;

**Д-** количество рабочих дней в месяце(для расчета принимаем

(Д=22раб.дня);

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

## 3.3 Расчет затрат на разработку информационных технологий

Под информационными технологиями понимаются экономические информационные системы (ЭИС), программные продукты (ПП), информационные базы данных и т.д.

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

**Зрп = Зот + Зст + Зэвм + Зсп + Зрм + Знр, руб. (2)**

Где: **Зот** – затраты на оплату труда разработчика (разработчиков), руб.;

**Зст** – страховые взносы по оплате труда во внебюджетные фонды,

руб.;

**Зэвм** – затраты, связанные с содержанием вычислительной

техники, руб.;

**Зсп** – затраты на специальное программное обеспечение, руб.;

**Зрм** – затраты на расходные материалы, необходимые при

разработке программного продукта, руб.;

**Знр** – затраты по накладным расходам, приходящиеся на

разработку программного продукта, руб.

### 3.3.1 Затраты на оплату труда разработчиков (Зот), руб.

Размер фонда оплаты труда разработчиков (Зот) рассчитывается по

формуле:

**Зот = Ор \* Трп (1+Кд), руб. (3)**

Где: **Ор** - месячный оклад разрабоцтчика проектного решения,

руб./мес.;

**Трп-** время разработки проектного решения разработчиком, мес.

включает в себя машинное время работы над проектом (**Тмрп**)

**Кд**–коэффициент дополнительной заработной платы

разработчика;

Если в разработке принимают участие несколько человек, то затраты по оплате труда определяются по формуле:

**Зот = , руб. (4)**

Где: месячный оклад разработчика проектного

решения, руб.;

**-** количество разработчиков, чел.

### 3.3.2 Затраты по страховым вносам(Зст), руб.

Сумма страховых взносов определяется по формуле:

**Зст = Кст \*Зот, руб или Зст = Зот \* (%ставка/100%), руб**. **(5)**

Где: **Кст**- коэффициент страховых взносов или процентная ставка

установленная государством, для расчета отчислений во

внебюджетные фонды.

### 3.3.3 Затраты по содержанию ЭВМ (Зэвм), руб.

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

**Зэвм = Тмрп \* Кгт \* Q\* Cм-ч, руб. (6)**

Где: **Тмрп-** машинное время на разработку проектного решения, час;

**Кгт**- коэффициент использования техники;

**Q-** – количество условных единиц используемой техники;

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

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

Величина стоимости машинного часа(**Cм-ч**) принимается по данным организации. Перевод рабочего времени в часы осуществляется по формуле:

**Тмрп = Тмч \* Чрд \* Тсм \* Ксм , час. (7)**

Где: **Тмч** – рабочее время в месяцах;

**Чрд** – число рабочих дней в месяце;

**Тсм** – продолжительность рабочей смены;

**Ксм** – количество рабочих смен.

### 3.4.4 Затраты на специальные программные продукты (Зсп), руб.

Затраты на специальные программные продукты, необходимые для разработки проектного решения определяются по формуле:

**Зсп= , руб. (8)**

Где ***Цρ*** – цена *ρ*-го специального программного средства, руб.

**n –** количество видов программных продуктов, шт.

### 3.3.5 Затраты на расходные материалы (Зрм), руб.

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

**Зсп = , руб. (9)**

Где: -количество материала, шт.;

цена вида материала, руб.;

**n** – количество видов материалов.

Затраты на расходные материалы удобнее определять в таблице

Таблица 3.3 Затраты на расходные материалы

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование** | **Цена (без НДС), руб.** | **Количество, шт.** | **Стоимость, руб.** |
| 1 | 2 | 3 | 3=2\*3 |
| Бумага | 1 | 50 | 50=1\*50 |
| Диск | 1 | 100 | 100=1\*100 |
| Ручка | 1 | 60 | 60=1\*60 |
| Флешка | 1 | 400 | 400=1\*400 |
| Итого | | | 610 |
| ТЗР (15% от Итого) | | | 91,5 |
| Всего затрат с учетом ТЗР | | | 701,5 |

### 3.3.6 Затраты по накладным расходам(Знр)

Затраты по накладным расходамопределяются по формуле:

**Знр = Кнр\*Зот, руб. (10)** Где: **Кнр**- коэффициент накладных расходов (или процент)

принимается для расчета по данным предприятия;

**Зот**- затраты по оплате труда, руб., см.формулу (2).

**Зрп - полные затраты на разработку программного продукта определяются по формуле (1) и сводятся в таблицу.**

Таблица 3.4 - Смета затрат на разработку программного продукта

|  |  |  |
| --- | --- | --- |
| **Перечень затрат** | **Условное обозначение** | **Значение, руб.** |
| Оплата труда | Зот | 23000 |
| Страховые взносы | Зст | 6900 |
| Содержание ЭВМ | Зэвм | 95 |
| Специальные программы | Зсп | 0 |
| Расходные материалы | Зрм | 701,5 |
| Накладные расходы | Знр | 3450 |
| Итого затрат на разработку | Зрп | 34146,5 |

## 3.4 Определение цены программного продукта, который разработан одной организаций по заказу другой.

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

**Цпп = Зрп+Пп +НДС, руб. (26)**

Цпп = 34146+10243+7990=52379 руб.

Где: **Цпп** – цена программного продукта, руб.;

**Зрп** – затраты на разработку проектного решения(итог раздела 3.3.

см. таб. 3.4.), руб.;

**Пп** – планируемая прибыль(принимается25% от суммы затрат на разработку программного продукта, т.е. от Зрп) руб.;

**НДС** – налог на добавленную стоимость(ставка НДС=18%), руб.;

Планируемая прибыль также мотет быть рассчитана исходя из нормативной рентабельности программного продукта, определяемая организацией и рассчитывается по формуле:

**Пп = Зрп Rп, руб. (27)**

Пп=34146\*0,3=10243 руб.

Где: Rп –коэффициент рентабельности, определенный предприятием.

Налог на добавленную стоимость определяется по ставке установленной государством от суммы затрат на разработку программного продукта и величины плановой прибыли по формуле:

**НДС = (Зрп + Пп)Кндс, руб. (28)**

НДС = (34146+10243)\*0,18=7990 руб.

Где **Кндс –**коэффициент НДС( Кндс=ставка НДС%/100%).

В результата полученного положительного финансового результата (прибыли) предприятие должно начислить налог на прибыль(ставка налога-20% от прибыли) по формуле:

**НП = Пп, руб. (30)**

НП = 10243\*0,2=2048 руб.

Прибыль оставшаяся на предприятии после уплаты налога на прибыль, определяется по форсуле:

**Пч = Пп НП, руб. (31)**

Пч = 10243-2048=8194 руб.

В целом поступления в бюджет от реализации программного продукта составят:

**БП = НП +НДС, руб. (32)**

БП = 2048+7990 =10038 руб.

# 4 МЕРОПРИЯТИЯ ПО ОБЕСПЕЧЕНИЮ БЕЗОПАСНОСТИ ЖИЗНЕДЕЯТЕЛЬНОСТИ НА ПРЕДПРИЯТИИ АО «НИИ «Рубин»

## 4.1 Анализ условий труда администратора

Рабочее место IT-специалиста - это его рабочий кабинет, в котором он проводит большую часть времени. Совершенно очевидно, что чем лучше приспособлено рабочее место для выполнения функций администратора, тем производительнее и эффективнее будет его трудовая деятельность.

Рабочее место оборудовано компьютерным столом с дополнительными секциями для документов, полочками для клавиатуры и системного блока, тумбой со "скрытыми" выдвижными ящиками, подставкой для ног, имеющей ширину 350 мм, глубину 450 мм, регулировку по высоте в пределах до 150 мм и по углу наклона опорной поверхности подставки до 20 град. Поверхность подставки рифленая и имеет по переднему краю бортик высотой 10 мм. Так же рабочее место оборудовано подъемно-поворотным креслом с регулируемым по высоте и углам наклона сидением и спинкой, а также расстоянием спинки от переднего края сиденья. При этом регулировка каждого параметра независимая, легко осуществляется и имеет надежную фиксацию. **Поверхности сиденья**, спинки и других элементов кресла полумягкие, с нескользящим, слабо электризующимся и воздухопроницаемым покрытием, обеспечивающим легкую очистку от загрязнений.

Рабочая поверхность стола сделана из ДСП с износостойкой поверхностью. На столе расположен монитор компьютера, клавиатура, мышь, все необходимые канцелярские принадлежности, настольная лампа и телефон. Клавиатура и мышь подключены к системному блоку через инфракрасные порты, что исключает наличие проводов и повышает мобильность этих устройств.

На тумбе слева от стола размещено многофункциональное устройство (сканер, принтер, ксерокс).

Вдоль стен стоят книжные шкафы для хранения печатной продукции.

Площадь помещения, в котором располагается рабочее место библиотекаря, 30 м2, высота – 3 м. Форма кабинета прямоугольная.

Кабинет оформлен в светло-зеленых теплых тонах, так как зеленый цвет полезен для глаз. Все поверхности в помещении матовые, т. блестящие поверхности вредны для здоровья.

Для снижения шума используются ковровые покрытия на пол. Для поддержания нормального микроклимата применяются кондиционеры. Для снижения солнечного света на окнах жалюзи.

Режим работы IT-специалиста - односменный с пятидневной рабочей неделей и нормированным рабочим днем. Продолжительность ежегодного отпуска 24 рабочих дня.

К качествам библиотекаря предъявляются следующие требования:

* зрение и слух без значительных отклонений;
* функционирование опорно-двигательного аппарата и верхних конечностей без значительных отклонений;
* развитая функция внимания, особенно во взаимосвязи со зрительным и слуховым восприятием;
* хорошая память, развитые творческие качества: воображение, мышление, творческое мышление;
* находчивость, инициативность, энергичность, общительность, самостоятельность, умение мыслить самостоятельно;
* умение быстро ориентироваться в обстановке.

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

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

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

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

Согласно санитарно-гигиеническим нормам (СанПиН 2.2.4.723-98), дисплей должен располагаться на расстоянии 60-70 см, но не менее 50 см от глаз. Между боковыми поверхностями мониторов должно быть не менее 1,2 м. При использовании жидкокристаллических дисплеев на каждый компьютер должно приходиться не менее 4,5 м2 площади. На дисплей ПК не должен попадать прямой солнечный свет во избежание бликов и повышенной нагрузки на зрение.

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

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

## 4.2 Расчет искусственного освещения в помещении

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

Условия работы для расчета показателя освещенности рабочего места являются:

* помещение размером 30 квадратных метров;
* высота помещения 3 метра, длина 6 метров, ширина 5 метров;
* лампы люминесцентные (дневного света) в количестве шести штук, новые.

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

 (1)

где

Ен - нормируемая минимальная допустимая освещенность по СНиП 23-05-95,для помещения 2000 лк;

S - площадь освещаемого помещения, м2;

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

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

Nc - число светильников в помещении;

- коэффициент затенения, обычно 1;

=i - коэффициент использования светового потока

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

 (2)

где

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

Нс - высота подвеса светильников над рабочей поверхностью, м.

 (3)

 (4)

Согласно СНиП 23-05-95 величина полученного светового потока достаточна для данного помещения.

## 4.3 Электробезопасность на предприятии

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

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

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

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

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

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

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

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

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

## 4.4 Пожарная безопасность на предприятии

Инструктаж по пожарной безопасности проводится по программе, разработанной инженером по охране труда организации (Зам. Генерального директора), с учетом требований стандартов, правил, норм и инструкций о мерах пожарной безопасности. Продолжительность инструктажа устанавливается в соответствии с утвержденной программой. Инструктаж по пожарной безопасности, как правило, проводится совместно с инструктажем по технике безопасности и в те же сроки.

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

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

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

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

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

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

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

Соблюдение рассмотренных в данном разделе мероприятий позволяет снизить утомляемость и травматизм, повысить производительность труда, обеспечить комфортные условия трудовой деятельности.

# ЗАКЛЮЧЕНИЕ

В рамках дипломного проекта была создана автоматизированная информационная система для АО «НИИ «Рубин». Данный программный продукт сокращает в сотни раз время, средства и силы, затраченные на инвентаризацию ПО. Благодаря созданной АИС «Рубин» лишится проблемы с инвентаризацией и будет направлять освободившиеся ресурсы на дальнейшее свое развитие и продвижение на мировом рынке.

Дипломный проект удовлетворяет требованиям, поставленным заказчиком, система прошла тестирование, успешно функционирует в организации АО «НИИ «Рубин».

# СПИСОК ЛИТЕРАТУРЫ

1. АО «НИИ «Рубин» - http://www.rubin-spb.ru
2. Работа с реестром [Электронный ресурс] - http://ab57.ru/reestr.html
3. КиберФорум [Электронный ресурс] - http://www.cyberforum.ru/csharp-net/thread104649.html
4. SQL синтаксис [Электронный ресурс] - http://www.site-do.ru/db/sql5.php
5. MySQL manual [Электронный ресурс] - http://www.mysql.ru/docs/man
6. ISO/IEC 9075 [Электронный ресурс] -http://www.iso.org/iso/iso\_catalogue/catalogue\_tc/catalogue\_detail.htm?csnumber=53685
7. ISO/IEC 23270 [Электронный ресурс] - <http://www.iso.org/iso/catalogue_detail.htm?csnumber=36768>
8. Александреску, А. Язык программирования D / А. Александреску. — М.: Символ, 2013. — 536 с.
9. Александреску, А. Язык программирования D / А. Александреску. — СПб: Символ-Плюс, 2014. — 544 с.
10. Ашарина, И.В. Основы программирования на языках С и С++: Курс лекций для высших учебных заведений / И.В. Ашарина. — М.: Гор. линия-Телеком, 2012. — 208 с.
11. Баженова, И.Ю. Языки программирования: Учебник для студентов учреждений высш. проф. образования / И.Ю. Баженова; Под ред. В.А. Сухомлин. — М.: ИЦ Академия, 2012. — 368 с.
12. Белоусова, С.Н. Основные принципы и концепции программирования на языке VBA в Excel: Учебное пособие / С.Н. Белоусова, И.А. Бессонова. — М.: БИНОМ ЛЗ, 2010. — 200 с.
13. Бьянкуцци, Ф. Пионеры программирования: Диалоги с создателями наиболее популярных языков программирования / Ф. Бьянкуцци, Ш. Уорден; Пер. с англ. С. Маккавеев. — СПб.: Символ-Плюс, 2011. — 608 с.
14. Бьянкуцци, Ф. Пионеры программирования. Диалоги с создателями наиболее популярных языков программирования / Ф. Бьянкуцци, Ш. Уорден. — М.: Символ, 2011. — 608 с.
15. Бьянкуцци, Ф. Пионеры программирования. Диалоги с создателями наиболее популярных языков программирования / Ф. Бьянкуцци, Ш. Уорден. — СПб.: Символ-Плюс, 2011. — 608 с.
16. Гавриков, М.М. Теоретические основы разработки и реализации языков программирования: Учебное пособие / М.М. Гавриков, А.Н Иванченко, Д.В Гринченков. — М.: КноРус, 2010. — 184 с.
17. Гергель, В.П. Современные языки и технологии параллельного программирования: Учебник / В.П. Гергель. — М.: МГУ, 2012. — 408 с.
18. Гергель, В.П. Современные языки и технологии параллельного программирования: Учебник / предисл.: В.А. Садовничий. / В.П. Гергель. — М.: Изд МГУ, 2012. — 408 с.
19. Голицына, О.Л. Языки программирования: Учебное пособие / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. — М.: Форум, НИЦ ИНФРА-М, 2013. — 400 с.
20. Головин, И.Г. Языки и методы программирования: Учебник для студентов учреждений высшего профессионального образования / И.Г. Головин, И.А. Волкова. — М.: ИЦ Академия, 2012. — 304 с.
21. Довек, Ж. Введение в теорию языков программирования / Ж. Довек, Ж.-Ж. Леви. — М.: ДМК, 2016. — 134 с.
22. Дорогов, В.Г. Основы программирования на языке С: Учебное пособие / В.Г. Дорогов, Е.Г. Дорогова; Под общ. ред. проф. Л.Г. Гагарина. — М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. — 224с.

# ПРИЛОЖЕНИЕ А

На рисунке 1 представлена use case диаграмма АИС инвентаризации ПО в MS OC Windows.

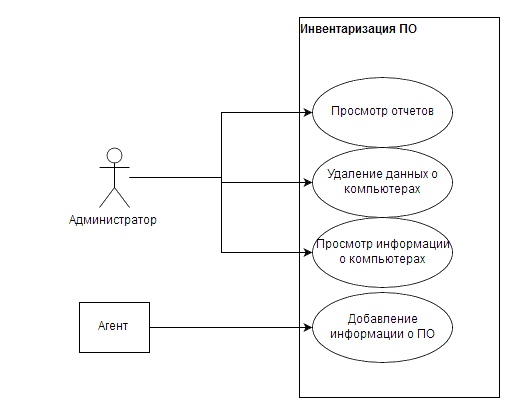
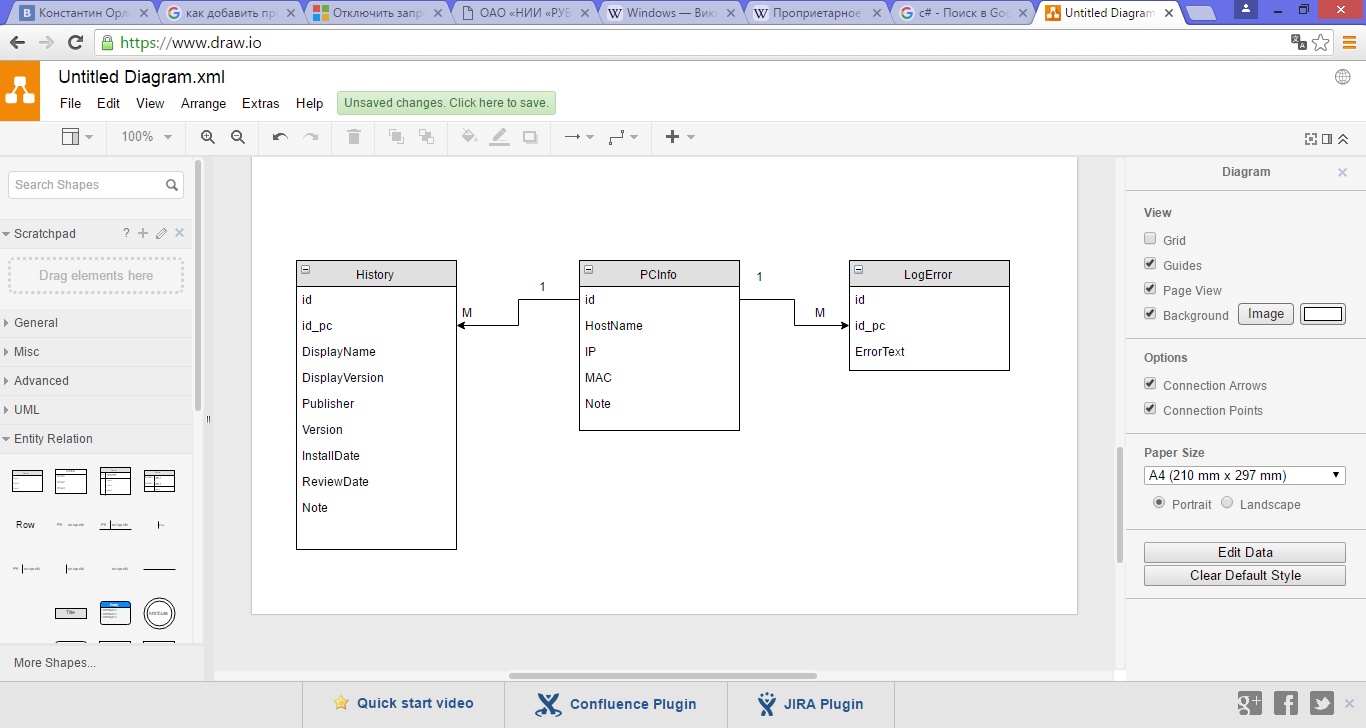


Рисунок 1 – use case диаграмма

# ПРИЛОЖЕНИЕ Б

На рисунке 2 представлена схема созданной базы данных для АИС по инвентаризации ПО в MS OC Windows.

  
Рисунок 2 – Схема БД

# ПРИЛОЖЕНИЕ В

Основные модули для агента для сбора информации о программах представлены на рисунке 1.

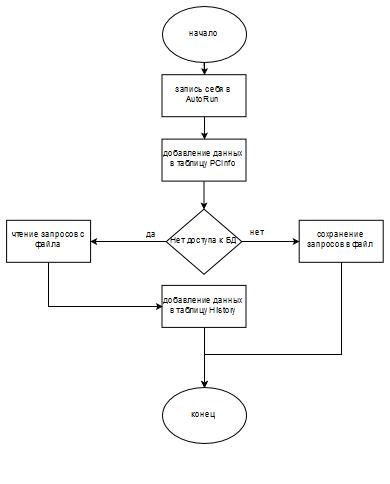
****

Рисунок 1 – Основные модули Агента

# Приложение Г

Программа-агент – это консольное приложение и она не имеет графического интерфейса. Листинг кода программы-агента представлен ниже.

using System;

using Microsoft.Win32;

using System.Data.SqlClient;

using System.Net;

using System.Net.NetworkInformation;

using System.IO;

using MySql.Data.MySqlClient;

using System.Data;

using System.Reflection;

namespace ConsoleApplication2

{

//Что бы вкл консоль надо Проект->Свойства->Приложение->Тип входных данных

class MainClass

{

public static bool error = false; //error = true когда нет доступа к бд

public static int j = 0;

public static string connStr = ""; //параметры подключения

public static void Main(string[] args)

{

StreamReader fs = new StreamReader(@"C:\setting.txt"); //ip server

Console.WriteLine();

string serverName = fs.ReadToEnd();

fs.Close();

string userName = "kostik";

string dbName = "diplom";

string port = "3306";

string password = "kostik\_pass";

connStr = "server=" + serverName +

";user=" + userName +

";database=" + dbName +

";port=" + port +

";password=" + password + ";";

var key = Registry.LocalMachine.OpenSubKey(@"SOFTWARE\Microsoft\Windows\CurrentVersion\Run\", true);

key.SetValue(Path.GetFileName(Assembly.GetExecutingAssembly().Location), Assembly.GetExecutingAssembly().Location); // добавление в автозагрузку

string s;

s = "SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Uninstall";

RegistryKey rk = Registry.LocalMachine.OpenSubKey(@s, true);

PCInfo(); // добавление/проверка информации о пк

AddData(rk); // добавление данных из реестра в бд

}

private static string GetMacAddress() //Получаем МАС адрес

{

string macAddresses = "";

foreach (NetworkInterface nic in NetworkInterface.GetAllNetworkInterfaces())

{

if (nic.OperationalStatus == OperationalStatus.Up)

{

macAddresses += nic.GetPhysicalAddress().ToString();

break;

}

}

return macAddresses;

}

public static void PCInfo()

{

string s1 = "";

try

{

string host = Dns.GetHostName().ToString();

s1 = "SELECT id FROM pcinfo WHERE HostName='"

+ Dns.GetHostName().ToString() + "';";

if (select(s1)) //Есть ли //такой HostName в бд

{

s1 = "INSERT INTO pcinfo (HostName, IP, MAC, Note) VALUES('" + host + "','"

+ Dns.GetHostByName(host).AddressList[0].ToString() + "','" + GetMacAddress() + "','first run pc: " + DateTime.Now.ToString() + "');";

query(s1); //Добавляем информацию о пк

}

}

catch { error = true; }

}

public static void AddData(RegistryKey rk)

{

j++;

string s1 = "";

string Name = Convert.ToString(rk.GetValue("DisplayName"));

if (Name.Length != 0) //добавляем только приложения, у которых не нулевое имя в реестре

{

string s = "SELECT id FROM history WHERE id\_pc=(SELECT id FROM PCInfo WHERE HostName='"

+ Dns.GetHostName().ToString() + "') and DisplayName = '"

+ rk.GetValue("DisplayName") + "' and DisplayVersion= '"

+ rk.GetValue("DisplayVersion") + "' and Publisher ='"

+ rk.GetValue("Publisher") + "' and Version= '"

+ rk.GetValue("Version") + "' and InstallDate= '"

+ rk.GetValue("InstallDate") + "';";

s1 = "INSERT INTO History (id\_pc,DisplayName,DisplayVersion,Publisher,Version,InstallDate,ReviewDate) VALUES ((SELECT id FROM PCInfo WHERE HostName='"

+ Dns.GetHostName().ToString() + "'),'"

+ rk.GetValue("DisplayName") + "','"

+ rk.GetValue("DisplayVersion") + "','"

+ rk.GetValue("Publisher") + "','"

+ rk.GetValue("Version") + "','"

+ rk.GetValue("InstallDate") + "','"

+ DateTime.Now.Date.ToString().Substring(0, 10) + "'); ";

if (error)

{

File.AppendAllText(@"C:\sql.txt", s1 + Environment.NewLine); //если нет //доступа к бд запросы сохряняться в файле до следующего удачного подключения к бд

}

else

{

try

{

StreamReader fs = new StreamReader(@"C:\sql.txt"); //сначала //отправляем запросы из сохраненного файла

string temp = fs.ReadToEnd();

query(temp);

fs.Close();

File.Delete(@"C:\sql.txt"); //во избежании //повторных результатов - удаляем файл после выборки всех запросов

}

catch

{

if (select(s)) //проверяем //совпадения в бд, для сохранения места в ней

query(s1); //добавляем //запись, если такой не существует и/или была изменена в отличии от предыдущих проверок

}

}

}

foreach (string s in rk.GetSubKeyNames()) //передвигаемся //по иерархии в реестре

{

if (j == 2000)

break;

AddData(rk.OpenSubKey(s));

}

}

public static void query(string s) //добавляем записи

{

MySqlConnection conn = new MySqlConnection(connStr);

MySqlCommand sql = new MySqlCommand(s, conn);

conn.Open();

sql.ExecuteNonQuery(); //передаем запрос

conn.Close();

}

public static bool select(string s) //Проверяем совпадения в бд

{

MySqlConnection conn = new MySqlConnection(connStr);

conn.Open();

MySqlCommand sql = new MySqlCommand(s, conn);

DataTable dt = new DataTable();

MySqlDataAdapter da = new MySqlDataAdapter(sql);

da.Fill(dt);

conn.Close();

if (dt.Rows.Count == 0) //при отсутсвии совпадения return true в противном //случае return false

return true;

else

return false;

}

}

}

Листинг интерфейса оболочки представлен ниже.

<Window x:Class="part2.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:part2"

mc:Ignorable="d"

Title="Инвентаризация"

WindowState="Maximized"

MinHeight="500"

MinWidth="1200">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="23"/>

<RowDefinition/>

</Grid.RowDefinitions>

<Menu x:Name="menu">

<MenuItem Name="changes" Header="Информация">

<MenuItem Name="menu\_pc" Header="о компьютерах" Click="menu\_pc\_Click" />

<MenuItem Name="menu\_history" Header="о программах" Click="menu\_history\_Click"/>

</MenuItem>

<MenuItem Name="fiter" Header="Фильтр изменений">

<MenuItem Name="menu\_publisher\_filter" Header="Фильтр по издателю" Click="menu\_publisher\_filter\_Click" />

<MenuItem Name="menu\_version\_filter" Header="Фильтр по версии" Click="menu\_version\_filter\_Click" />

<MenuItem Name="menu\_name\_filter" Header="Фильтр по названию" Click="menu\_name\_filter\_Click"/>

</MenuItem>

<MenuItem Name="menu\_about" Header="О программе">

<MenuItem Name="menu\_author" Header="Автор"/>

<MenuItem Name="menu\_version" Header="Версия"/>

<MenuItem Name="menu\_mail" Header="Обратная связь"/>

</MenuItem>

</Menu>

<Grid x:Name="pc\_grid" Grid.Row="1" Visibility="Hidden">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="227\*"/>

<ColumnDefinition Width="287\*"/>

<ColumnDefinition Width="246\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="437\*"/>

<RowDefinition Height="44\*"/>

</Grid.RowDefinitions>

<DataGrid Name="pc\_datagrid" Grid.ColumnSpan="3" ItemsSource="{Binding}" SelectedItem="{Binding Path=SelectedCustomer, Mode=TwoWay}" IsReadOnly="True"/>

<Button Name="delete\_pc" Content="Удалить" Grid.Row="1" Grid.Column="1" Click="delete\_pc\_Click"/>

</Grid>

<Grid x:Name="history\_grid" Grid.Row="1" Visibility="Hidden">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="379\*"/>

<ColumnDefinition Width="114\*"/>

<ColumnDefinition Width="133\*"/>

<ColumnDefinition Width="134\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="307\*"/>

<RowDefinition Height="30"/>

<RowDefinition Height="48\*"/>

<RowDefinition Height="41\*"/>

<RowDefinition Height="61\*"/>

</Grid.RowDefinitions>

<Button x:Name="button" Content="Показать" Grid.Row="3" Grid.Column="2" Click="button\_Click" />

<DataGrid x:Name="history\_data\_datagrid" Grid.ColumnSpan="4" ItemsSource="{Binding}" IsReadOnly="True"/>

<DataGrid x:Name="history\_pc\_datagrid" Grid.Row="2" Grid.RowSpan="3" ItemsSource="{Binding}" IsReadOnly="True"/>

<DatePicker x:Name="with" Grid.Column="2" Grid.Row="1"/>

<DatePicker x:Name="to" Grid.Column="3" Grid.Row="1"/>

<Label x:Name="label" Content="Интервал ревизии" Grid.Column="1" Grid.Row="1"/>

<Label x:Name="label1" Content="выбирите нужный компьютер" Grid.Row="1"/>

</Grid>

<Grid x:Name="publisher\_grid" Grid.Row="1" Visibility="Hidden">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="379\*"/>

<ColumnDefinition Width="114\*"/>

<ColumnDefinition Width="133\*"/>

<ColumnDefinition Width="134\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="307\*"/>

<RowDefinition Height="30"/>

<RowDefinition Height="48\*"/>

<RowDefinition Height="41\*"/>

<RowDefinition Height="61\*"/>

</Grid.RowDefinitions>

<DataGrid x:Name="publisher\_pc\_datagrid" Grid.Row="2" Grid.RowSpan="3" ItemsSource="{Binding}" IsReadOnly="True"/>

<DataGrid x:Name="publisher\_history\_datagrid" Grid.ColumnSpan="4" ItemsSource="{Binding}" IsReadOnly="True"/>

<Label x:Name="label2" Content="выбирите нужный компьютер" Grid.Row="1" />

<Label x:Name="label3" Content="Введите требуемого автора" Grid.Row="1" Grid.Column="1" Grid.ColumnSpan="3" TextOptions.TextFormattingMode="Display" />

<Button x:Name="publisher\_filter\_button" Content="Показать" Grid.Row="3" Grid.Column="2" Click="publisher\_filter\_button\_Click" />

<TextBox x:Name="publisher\_filter" Grid.Row="2" Grid.Column="1" Grid.ColumnSpan="3"/>

</Grid>

<Grid x:Name="Name\_grid" Grid.Row="1" Visibility="Hidden">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="247\*"/>

<ColumnDefinition Width="133\*"/>

<ColumnDefinition Width="114\*"/>

<ColumnDefinition Width="133\*"/>

<ColumnDefinition Width="134\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="307\*"/>

<RowDefinition Height="30"/>

<RowDefinition Height="48\*"/>

<RowDefinition Height="41\*"/>

<RowDefinition Height="61\*"/>

</Grid.RowDefinitions>

<DataGrid x:Name="name\_pc\_datagrid" Grid.Row="2" Grid.RowSpan="3" ItemsSource="{Binding}" IsReadOnly="True" Margin="0,0,1,0"/>

<DataGrid x:Name="name\_history\_datagrid" Grid.ColumnSpan="5" ItemsSource="{Binding}" IsReadOnly="True"/>

<Label x:Name="label5" Content="Введите название программы" Grid.Row="1" Grid.Column="1" Grid.ColumnSpan="2" TextOptions.TextFormattingMode="Display" Margin="4,0,1,0" />

<Button x:Name="name\_filter\_button" Content="Показать" Grid.Row="3" Grid.Column="1" Click="name\_filter\_button\_Click" Grid.ColumnSpan="2" />

<TextBox x:Name="name\_filter" Grid.Row="2" Grid.Column="1" Grid.ColumnSpan="2" KeyUp="name\_filter\_KeyDown" />

</Grid>

<Grid x:Name="Version\_grid" Grid.Row="1" Visibility="Hidden">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="379\*"/>

<ColumnDefinition Width="114\*"/>

<ColumnDefinition Width="133\*"/>

<ColumnDefinition Width="134\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="307\*"/>

<RowDefinition Height="30"/>

<RowDefinition Height="48\*"/>

<RowDefinition Height="41\*"/>

<RowDefinition Height="61\*"/>

</Grid.RowDefinitions>

<DataGrid x:Name="version\_pc\_datagrid" Grid.Row="2" Grid.RowSpan="3" ItemsSource="{Binding}" IsReadOnly="True"/>

<DataGrid x:Name="version\_history\_datagrid" Grid.ColumnSpan="4" ItemsSource="{Binding}" IsReadOnly="True"/>

<Label x:Name="label6" Content="выбирите нужный компьютер" Grid.Row="1"/>

<Label x:Name="label7" Content="Выберите версию " Grid.Row="1" Grid.Column="1" Grid.ColumnSpan="3" TextOptions.TextFormattingMode="Display" />

<Button x:Name="version\_filter\_button" Content="Показать" Grid.Row="3" Grid.Column="2" Click="version\_filter\_button\_Click" />

<ComboBox x:Name="Version\_box" Grid.Column="2" Grid.Row="1"/>

</Grid>

</Grid>

</Window>

Листинг программного кода оболочки представлен ниже.

using MySql.Data.MySqlClient;

using System;

using System.Data;

using System.Windows;

using System.Windows.Controls;

namespace part2

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

string connStr;

MySqlCommand sc;

MySqlConnection conn;

private void connector(string connStr, string sql)

{

try

{

conn = new MySqlConnection(connStr);

sc = new MySqlCommand(sql, conn);

conn.Open();

sc.ExecuteNonQuery();

}

catch { }

}

private void select(string sql, object sender)

{

try

{

DataGrid dg = (DataGrid)sender;

connector(connStr, sql);

DataTable dt = new DataTable();

MySqlDataAdapter da = new MySqlDataAdapter(sc);

da.Fill(dt);

dg.DataContext = dt;

}

catch { }

}

public MainWindow()

{

string serverName = "localhost"; // Адрес сервера (для локальной базы пишите "localhost")

string userName = "kostik"; // Имя пользователя

string dbName = "diplom"; //Имя базы данных

string port = "3306"; // Порт для подключения

string password = "kostik\_pass"; // Пароль для подключения

connStr = "server=" + serverName +

";user=" + userName +

";database=" + dbName +

";port=" + port +

";password=" + password + ";";

InitializeComponent();

select("select \* from PCInfo;", pc\_datagrid);

}

private void AllHidden()

{

pc\_grid.Visibility = Visibility.Hidden;

history\_grid.Visibility = Visibility.Hidden;

publisher\_grid.Visibility = Visibility.Hidden;

Version\_grid.Visibility = Visibility.Hidden;

Name\_grid.Visibility = Visibility.Hidden;

}

private void menu\_pc\_Click(object sender, RoutedEventArgs e)

{

AllHidden();

pc\_grid.Visibility = Visibility.Visible;

select("select HostName as 'Имя компьютера', IP, MAC from pcinfo;", pc\_datagrid);

}

private void menu\_history\_Click(object sender, RoutedEventArgs e)

{

AllHidden();

history\_grid.Visibility = Visibility.Visible;

select("select \* from pcinfo;", history\_pc\_datagrid);

}

private void menu\_publisher\_filter\_Click(object sender, RoutedEventArgs e)

{

AllHidden();

publisher\_grid.Visibility = Visibility.Visible;

select("select \* from pcinfo;", publisher\_pc\_datagrid);

}

private void menu\_version\_filter\_Click(object sender, RoutedEventArgs e)

{

AllHidden();

Version\_grid.Visibility = Visibility.Visible;

select("select \* from pcinfo;", version\_pc\_datagrid);

sc = new MySqlCommand("select DisplayVersion from history group by DisplayVersion", conn);

MySqlDataReader r = sc.ExecuteReader();

while (r.Read()) {try { string rez = r[0].ToString(); if(rez!="") Version\_box.Items.Add(r[0].ToString()); } catch { } }

}

private void menu\_name\_filter\_Click(object sender, RoutedEventArgs e)

{

AllHidden();

Name\_grid.Visibility = Visibility.Visible;

select("select \* from pcinfo;", name\_pc\_datagrid);

}

private void delete\_pc\_Click(object sender, RoutedEventArgs e)

{

if (pc\_datagrid.SelectedItem != null)

{

var selectedCellSt = pc\_datagrid.SelectedCells[0];

var cellContentSt = selectedCellSt.Column.GetCellContent(selectedCellSt.Item);

int id = Convert.ToInt32((cellContentSt as TextBlock).Text);

string s = "DELETE FROM History WHERE id\_pc="+id+ "; DELETE FROM PCInfo WHERE id=" + id + ";";

connector(connStr, s);

MessageBox.Show(s);

select("select \* from pcinfo;",pc\_datagrid);

}

else

{

MessageBox.Show("Компьютер не выбран!");

}

}

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

{

if (history\_pc\_datagrid.SelectedItem != null)

{ if (with.Text == "") with.Text = "01.01.2000";

if (to.Text == "") to.Text = DateTime.Now.ToString();

String to\_date = to.Text.Substring(6, 4) + to.Text.Substring(3, 2) + to.Text.Substring(0, 2);

String with\_date = with.Text.Substring(6, 4) + with.Text.Substring(3, 2) + with.Text.Substring(0, 2);

var selectedCellSt = history\_pc\_datagrid.SelectedCells[0];

var cellContentSt = selectedCellSt.Column.GetCellContent(selectedCellSt.Item);

int id = Convert.ToInt32((cellContentSt as TextBlock).Text);

string s = string.Format("select \* from history where id\_pc = {0} and ReviewDate >= {1} and ReviewDate <= {2};",id,with\_date,to\_date);

select(s, history\_data\_datagrid);

}

else

{

MessageBox.Show("Компьютер не выбран!");

}

}

private void publisher\_filter\_button\_Click(object sender, RoutedEventArgs e)

{

string s = "";

if (publisher\_pc\_datagrid.SelectedItem != null)

{

var selectedCellSt = publisher\_pc\_datagrid.SelectedCells[0];

var cellContentSt = selectedCellSt.Column.GetCellContent(selectedCellSt.Item);

int id = Convert.ToInt32((cellContentSt as TextBlock).Text);

s = string.Format("select \* from history where id\_pc = {0} and Publisher = '{1}';", id, publisher\_filter.Text);

MessageBox.Show(s);

}

else

{

s = string.Format("select \* from history where Publisher = '{0}';", publisher\_filter.Text);

}

select(s, publisher\_history\_datagrid);

}

private void version\_filter\_button\_Click(object sender, RoutedEventArgs e)

{

string s = string.Format("select \* from history where DisplayVersion = '{0}'",Version\_box.Text);

select(s,version\_history\_datagrid);

}

private void name\_filter\_button\_Click(object sender, RoutedEventArgs e)

{

string s = string.Format("Select \* from history where DisplayName LIKE '{0}%'", name\_filter.Text);

select(s, name\_history\_datagrid);

}

private void name\_filter\_KeyDown(object sender, System.Windows.Input.KeyEventArgs e)

{

string s = string.Format("Select \* from history where DisplayName LIKE '{0}%'", name\_filter.Text);

select(s, name\_history\_datagrid);

}

}

}

# ПРИЛОЖЕНИЕ Д

1 НАЗНАЧЕНИЕ ПРОГРАММЫ

1.1 Функциональное назначение программы

Основной функцией информационной системы является инвентаризация программного обеспечения, отслеживания изменения в ПО, фиксации даты изменения.

1.2 Эксплуатационное назначение программы

Основное назначение программы - использование данного ПО в любой организации.

1.3 Состав функций

1.3.1 Функция вывода информации о компьютерах

Данная функция выводит информацию о всех компьютерах, на которых хоть раз был запущен агент по сбору информации о ПО на компьютерах.

1.3.2 Функция удаления информации о компьютерах и о программах

Данная функция предназначена для удаления информации о компьютерах, а так же попутно удаляет информацию о ПО для соответствующего компьютера.

1.3.3 Функция поиска по названию программы/автору/версии

Данная функция предназначена для поиска по различным полям в БД, а именно по названию программы, по автору продукта, по версии ПО, а так же выбор нужного компьютера, на котором необходимо провести поиск.

2 УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

2.1 Минимальный состав аппаратных средств

Минимальный состав используемых технических (аппаратных) средств:

- процессор Intel или другой совместимый;

- объем свободной оперативной памяти ~500 Кб;

- объем необходимой памяти на жестком диске ~3Мб;

- стандартный VGA-монитор или совместимый;

- стандартная клавиатура;

- манипулятор «мышь».

2.2 Минимальный состав программных средств

Минимальный состав используемых программных средств:

- Microsoft Windows 95 или совместимая;

- язык интерфейса – русский;

- MySQL.

2.3 Требования к персоналу (пользователю)

Конечный пользователь программы (оператор) должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы.

3 ВЫПОЛНЕНИЕ ПРОГРАММЫ

3.1 Загрузка и запуск программы

Программа распространяется на дисках. Запуск программы начинается сразу после вставки диска. Установка программы не требуется.

3.2 Выполнение программы

Сразу после запуска программа готова к использованию. Выполнение функций программы может осуществляться в любом порядке.

3.3 Завершение работы программы

Данная программа может быть завершена посредством нажатия на кнопку «Х» в правом верхнем углу программы. После нажатия работа программы остановится, и окно будет закрыто.

4 СООБЩЕНИЯ ОПЕРАТОРУ

4.1 Сообщение об ошибке

Программа выдает сообщение об ошибке, показанное на рисунке 1.

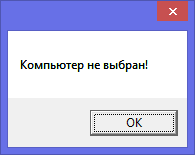


Рисунок 1 – ошибка

ПРИЧИНА: Компьютер, для которого осуществлять поиск, не выбран.

# ПРИЛОЖЕНИЕ Е

1 Введение

Настоящая программа и методика испытаний разработана для определения возможностей информационной системы инвентаризации ПО в OC MS Windows.

2 Объект испытаний

2.1 Наименование объекта испытаний

Полное наименование объекта испытаний: информационной системы инвентаризации ПО в OC MS Windows

2.2 Область применения

Система предназначена для использования в организациях. Система обеспечивает:

- вывод данных о изменениях в программных продуктах;

- выполнение поиска по наименованию программы;

- вывод информации о компьютерах.

3 Цель испытания

Цель проведения испытаний – оценка эксплуатационных характеристик информационной системы, проверка и подтверждение работоспособности информационной системы в условиях, максимально приближенных к условиям реальной эксплуатации и применения в организациях.

В ходе проведения испытаний выполняется:

1) визуальная оценка работы информационной системы в части мониторинга и записи данных в режиме реального времени;

2) проверка работоспособности функции поиска;

4 Общие положения и условия проведения испытаний

4.1 Испытания по настоящей методике проводятся приемочной комиссией, состав которой определяется в установленном порядке.

4.2 Дополнительные условия по проведению испытаний устанавливаются соответствующими разделами настоящей программы и методики испытаний.

4.3 При проведении работ по установке и настройке информационной системы, а также подключении рабочего места оператора должна обеспечиваться безопасность обслуживающего персонала в соответствии с системой стандартов безопасности труда и соблюдением норм пожарной безопасности. Все работы по настройке, эксплуатации и техническому обслуживанию оборудования информационной системы должны выполняться персоналом, прошедшим инструктаж.

5 Порядок испытаний

Испытания по настоящей программе и методике испытаний проводятся в соответствии с таблицей 1.

Таблица 1 – Содержание и порядок испытаний

|  |  |
| --- | --- |
| Порядок испытаний | Содержание испытаний |
| Этап 1 | Проверка удаления компьютеров в базе данных |
| Этап 2 | Проверка поиска по наименованию программы |
| Этап 3 | Проверка поиска данных по автору продукта |
| Этап 4 | Проверка вывода изменений за разные периоды |
| Этап 4 | Проверка правильности выборки данных Агентом из реестра |

6 Обработка, анализ и оценка результатов испытаний

6.1 В ходе испытаний необходимо документально фиксировать содержание и результаты испытаний.

6.2 Результаты проведения испытаний оформляются Актами рабочей комиссии.

6.3 Результаты считаются положительными, если система испытана в полном объеме и последовательности, которые установлены настоящей методикой, и соответствует всем требованиям, по которым проводятся испытания.

6.4 При выявлении недостатков приемочная комиссия принимает решение об устранении выявленных недостатков и определяет сроки их устранения.

6.5 После устранения недостатков, проводятся повторные испытания.

7 Результаты испытаний

При испытании программы по заданным этапам, программа прошла все испытания, и не было выявлено сбоев и ошибок в работе программы.

# ПРИЛОЖЕНИЕ Ж

Спроектированный интерфейс представлен на следующих пяти рисунках.

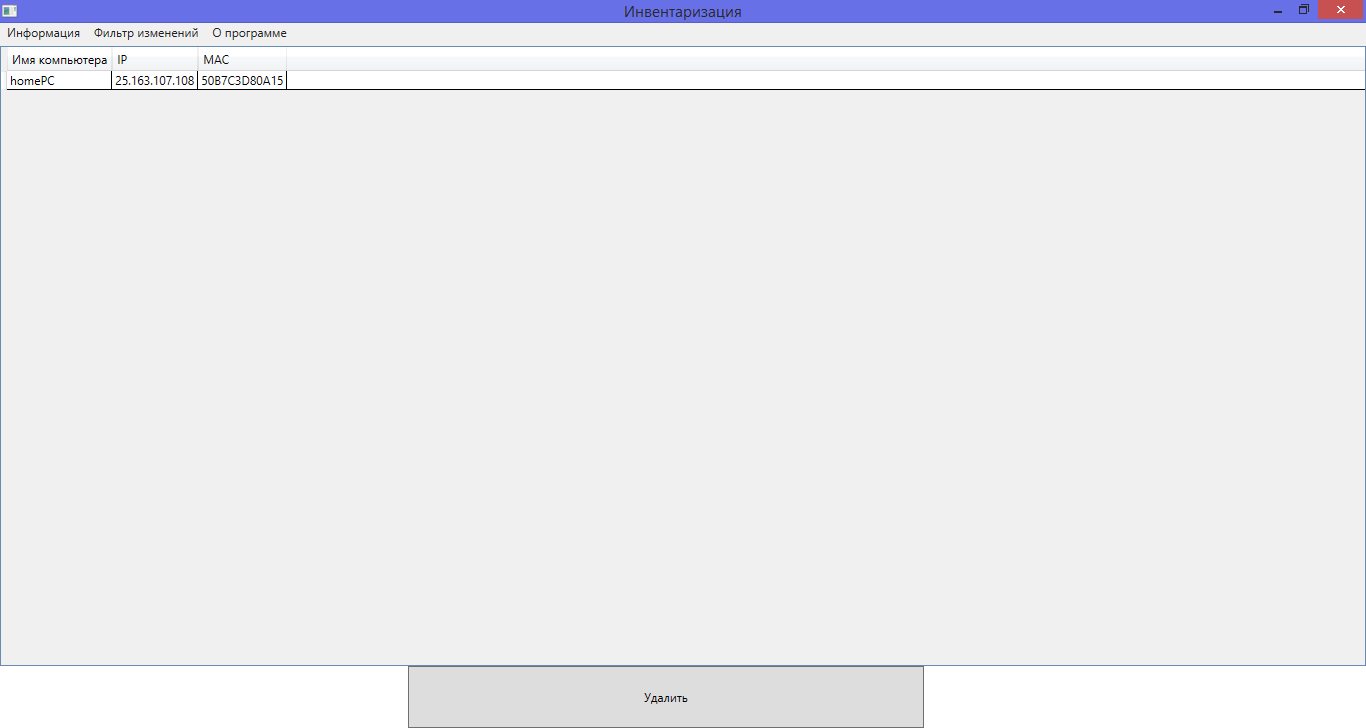


Рисунок 1 - Информация о компьютерах

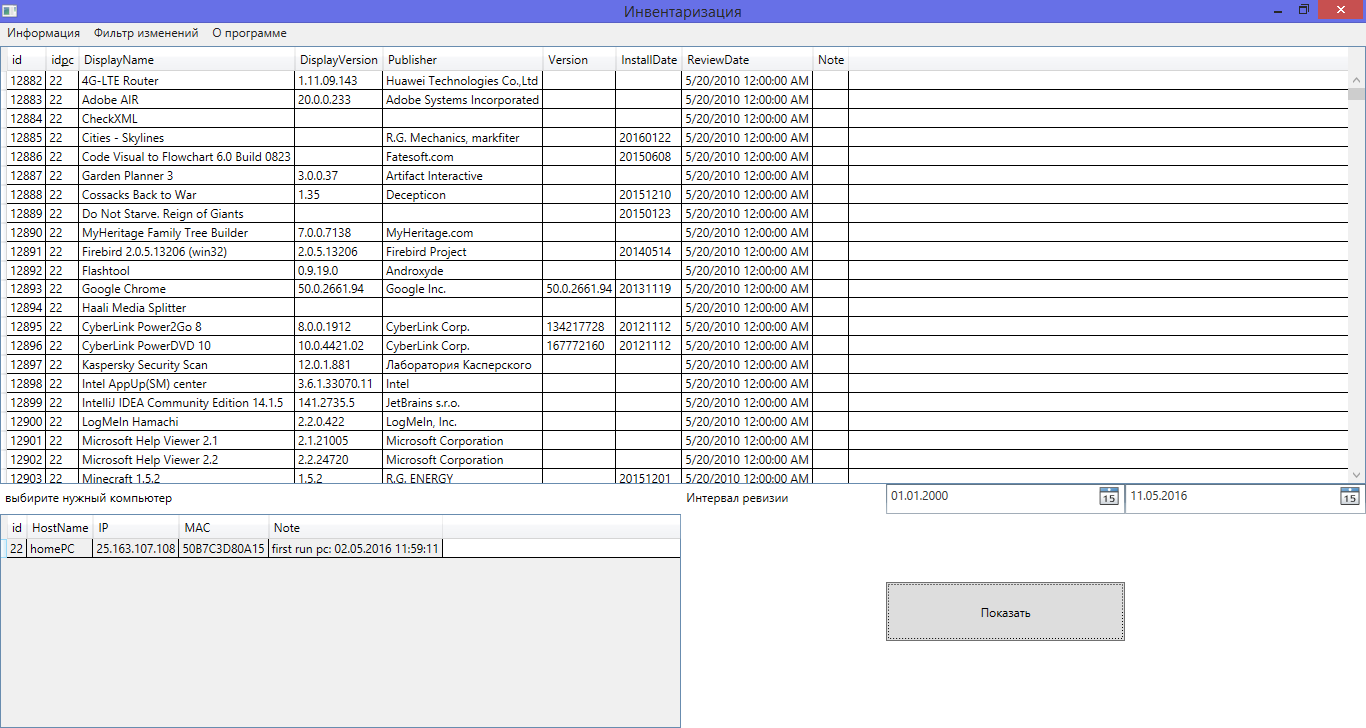


Рисунок 2 - Информация о программах

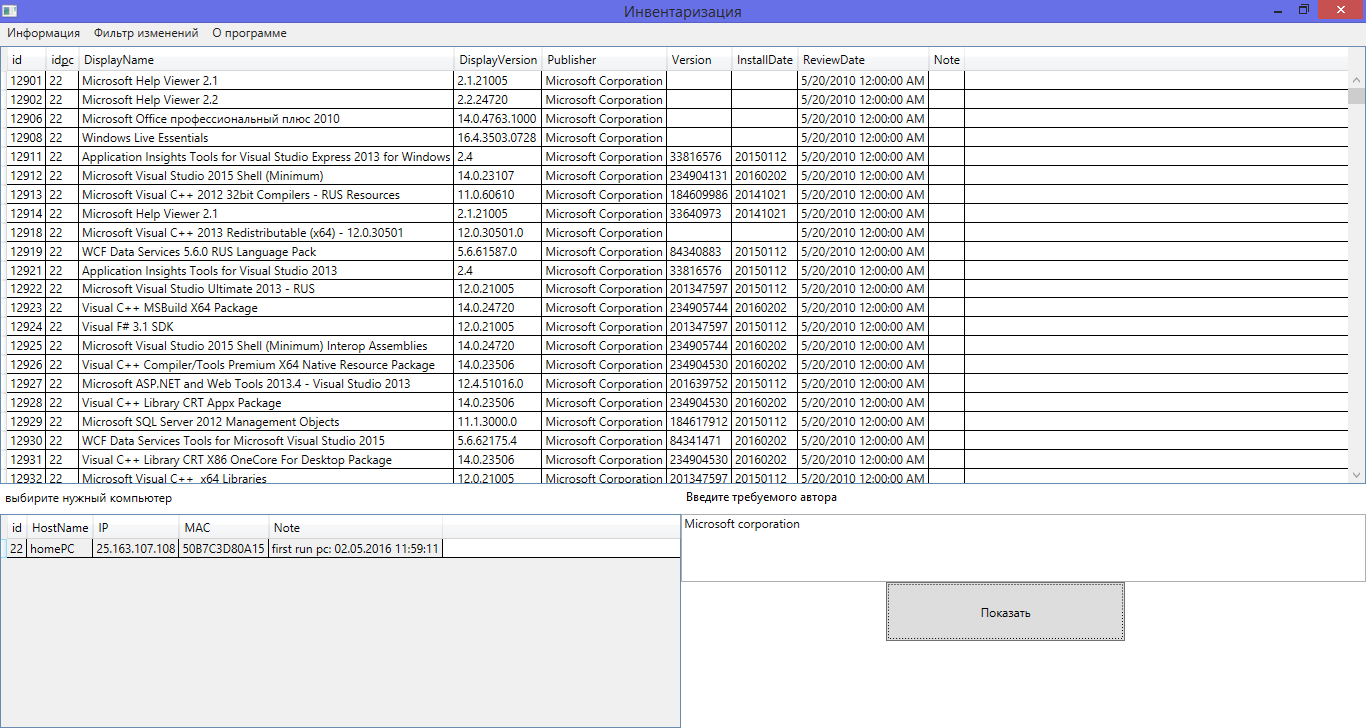


Рисунок 3 - Фильтрация по автору программы

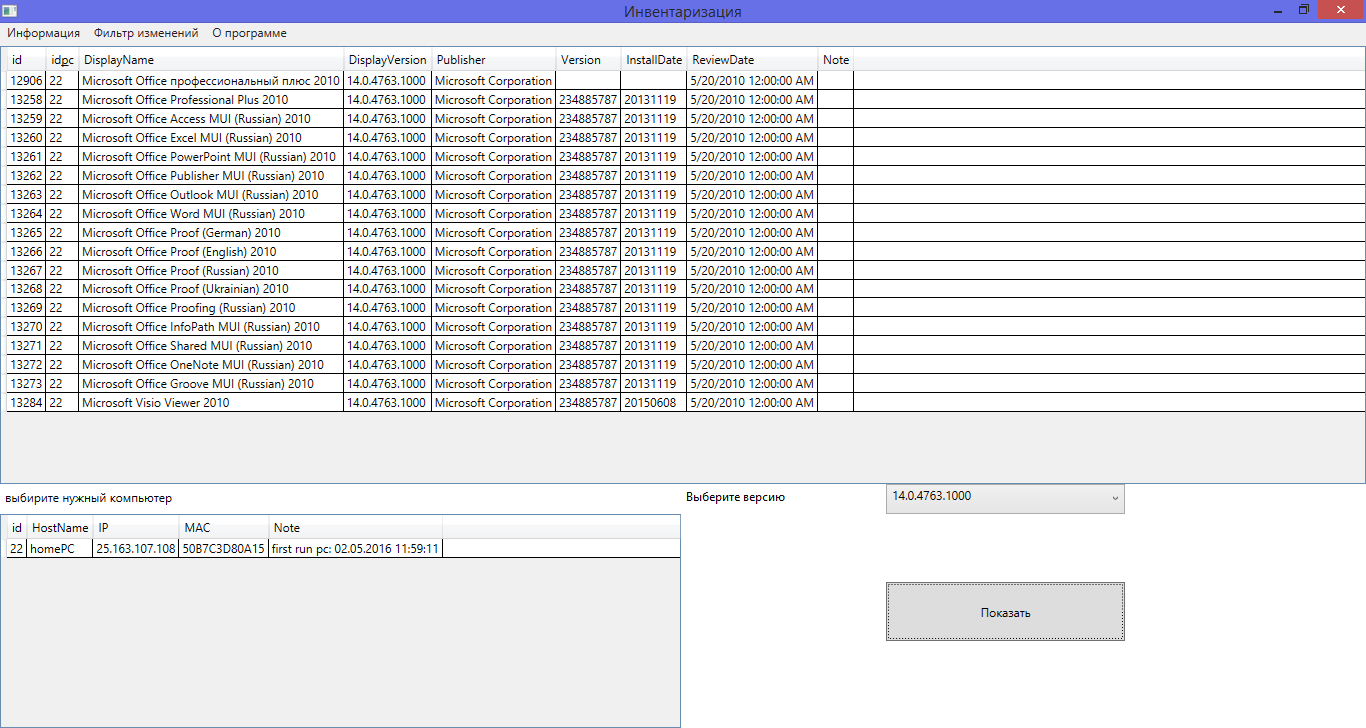


Рисунок 4 - Фильтрация по версии

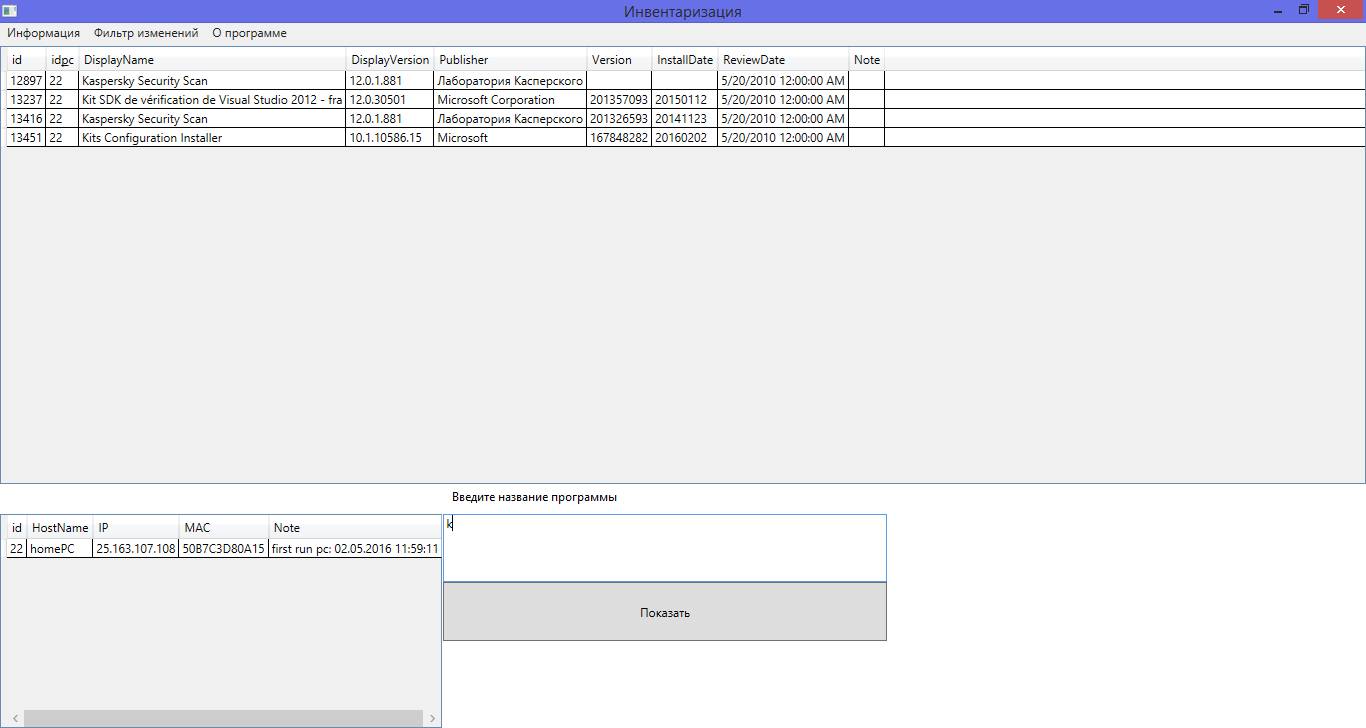


Рисунок 5 - Фильтрация по названию программы