

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет Математики и информатики
Кафедра Информационных и управляющих систем
Направление подготовки 09.03.01 - «Информатика и вычислительная техника»
Направленность (профиль) образовательной программы Автоматизированные
системы обработки информации и управления

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. Кафедрой
_____ А.В. Бушманов
« _____ » _____ 2023г

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка ПО для инвентаризации локальной сети АмГУ

Исполнитель студент группы 953об	_____	А.Л. Дорощенко
	(подпись, дата)	
Руководитель доцент, канд. техн. наук	_____	О.В. Жилиндина
	(подпись, дата)	
Консультант по безопасности и экологичности доцент, канд. техн. наук	_____	А.Б. Булгаков
	(подпись, дата)	
Нормоконтроль инженер кафедры	_____	В.Н. Адаменко
	(подпись, дата)	

Благовещенск 2023

РЕФЕРАТ

Дипломная (бакалаврская) работа содержит 74 страницы, 30 рисунков, 26 таблиц, 1 приложение, 27 источников.

.NET WPF, ДЕСКТОПНОЕ ПРИЛОЖЕНИЕ, АВТОМТИЗИРОВАННАЯ ИНВЕНТАРИЗАЦИЯ, SQLITE, ОБЪЕКТ-RELATIONAL MAPPING, СКАНИРОВАНИЕ

Цель разработки приложения для автоматизированной инвентаризации заключается в создании упрощённой платформы, которая обеспечит удобный и эффективный процесс инвентаризации активов организации. Для достижения этой цели необходимо провести исследование и анализ существующих процессов инвентаризации, а также определить требования и потребности пользователей.

В ходе работы требуется разработать функциональность, позволяющую пользователям управлять процессом инвентаризации, включая создание списка активов, их классификацию, присвоение уникальных идентификаторов и запись основных характеристик. Приложение должно предоставлять удобный интерфейс для ввода и отображения данных.

Результатом данной работы является десктопное приложение для автоматизированной инвентаризации, которое предоставляет пользователям простой и надежный способ управлять процессом инвентаризации активов. Приложение позволяет эффективно отслеживать и контролировать наличие и состояние активов организации, упрощая процесс учета и управления ресурсами.

СОДЕРЖАНИЕ

Введение	8
1 Общая характеристика предметной области	10
1.1 Характеристика исследуемой организации	10
1.1.1 Описание Амурского государственного университета	10
1.1.2 Описание ЦЦТиГО	10
1.2 Характеристика инвентаризации сети	14
1.2.1 Сущность инвентаризации сети	14
1.2.2 Методы инвентаризации сети	16
1.2.3 Существующие решения автоматической инвентаризации	17
2 Разработка программного продукта	26
2.1 Формулировка требований к информационной системе	26
2.2 Выбор инструментов и средств разработки	26
2.2.1 Выбор языка программирования	26
2.2.2 Выбор СУБД	27
2.2.3 Выбор графической подсистемы	28
2.2.4 Выбор интегрированной среды разработки	29
2.3 Модель работы ИС	31
2.3.1 Описание модели	31
3.3.2 Описание работы ИС	31
2.4 Проектирование структуры модулей и классов	34
2.5 Проектирование базы данных	35
2.5.1 Инфологическое проектирование	35
2.5.2 Логическое проектирование	41
2.5.3 Физическое проектирование	43
2.6 Проектирование пользовательского интерфейса	48
2.6.1 Подсистема «Сканирование»	48
2.6.2 Подсистема «Мастер отчётов»	49

2.6.3 Подсистема «Настройки»	50
2.7 Реализация ИС	51
2.7.1 Обеспечение возможности сканирования компьютеров	51
2.7.2 Обеспечение отзывчивой работы программы	52
2.7.3 Описание SQLite	54
2.7.4 Разработка пользовательского интерфейса	55
2.7.5 Обеспечение информационной безопасности	57
2.7.6 Итоги реализации	57
3 Безопасность и экологичность	60
3.1 Безопасность	60
3.1.1 Опасности и вред ПЭВМ на рабочем месте пользовател	60
3.1.2 Организация рабочего места	60
3.1.3 Освещение	62
3.1.4 Шум	64
3.1.5 Микроклимат	64
3.1.6 Требования к графическому интерфейсу	65
3.2 Экологичность	66
3.3 Чрезвычайные ситуации	67
3.3.1 Аварийные ситуации	67
3.3.2 Меры пожарной безопасности на рабочих местах	68
ЗАКЛЮЧЕНИЕ	69
БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ	70
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	71
ПРИЛОЖЕНИЕ А	74

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей бакалаврской работе использованы ссылки на следующие стандарты и нормативные документы:

СП 2.2.3670-20. Санитарно-эпидемиологические требования к условиям труда.

СанПиН 1.2.3685-21. Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания.

ГОСТ Р 50948-2001. Средства отображения информации индивидуального пользования. Общие эргономические требования и требования безопасности.

ГОСТ Р 50949-2001. Средства отображения информации индивидуального пользования. Методы измерений и оценки эргономических параметров и параметров безопасности.

ГОСТ 28406-89. Персональные электронные вычислительные машины. Интерфейсы видеомониторов. Общие требования

ГОСТ Р 51645-2017. Рабочее место для инвалида по зрению типовое специальное компьютерное. Технические требования к оборудованию и производственной среде.

ГОСТ Р ИСО 1503-2014. Эргономика. Требования к пространственной ориентации и направлениям движения органов управления.

ГОСТ 12.1.007-76 Система стандартов безопасности труда (ССБТ). Вредные вещества. Классификация и общие требования безопасности.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

WPF – Windows Presentation Foundation;

SQL – Structured Query Language;

UWP – Universal Windows Platform;

MAUI – Multi-Platform App UI;

UInt16 – unsigned integer 16bit;

XAML – eXtended Application Markup Language;

WMI – Windows Management Instrumentation;

MVVM – Model-View-ViewModel;

ЦЦТиТО – Центр цифровой трансформации и технического обеспечения;

СУБД – система управления базами данных;

ПО – программное обеспечение;

ИС – информационная система;

ВВЕДЕНИЕ

Информационные технологии охватывают широкий спектр инструментов, технологий и процессов, используемых для сбора, обработки, хранения и передачи информации. Они играют ключевую роль в современном бизнесе и организациях, обеспечивая эффективность, автоматизацию и улучшение рабочих процессов.

Одной из важных задач в области информационных технологий является инвентаризация компьютеров и сетевых ресурсов в организации. Инвентаризация позволяет точно определить и отслеживать наличие и состояние аппаратного и программного обеспечения, а также управлять ресурсами и планировать их обновление.

Однако, проблемы инвентаризации могут возникать в связи с быстрым развитием информационных систем и постоянными изменениями в компьютерном парке. Это создает сложности при отслеживании и учете всех компьютеров, их конфигураций и программного обеспечения.

Другая проблема, связанная с инвентаризацией, заключается в обнаружении и учете периферийных устройств, таких как принтеры, сканеры, мониторы и т.д. Корректная инвентаризация этих устройств важна для обеспечения их правильной работы и обслуживания.

Также, сложность может возникать при сетевой инвентаризации, когда необходимо отследить и классифицировать сетевые устройства, их IP-адреса, порты и другие сетевые параметры.

Решение этих проблем требует использования специализированных инструментов и программных решений, таких как программы инвентаризации компьютеров, которые позволяют автоматизировать процесс сбора информации о компьютерах и устройствах в сети, а также проводить регулярное обновление и контроль инвентарных данных.

Для АмГУ характерна проблема инвентаризации. Текущий способ

инвентаризации в данной организации – вручную. Из-за этого вытекают следующие проблемы:

- времязатратность. Ручная инвентаризация требует значительного времени и усилий, особенно в случае больших организаций с большим количеством компьютеров и устройств. Это может замедлить процесс и снизить производительность сотрудников, занимающихся инвентаризацией;

- ограниченная масштабируемость. Ручная инвентаризация становится более сложной и неэффективной при увеличении объема и сложности сетевой инфраструктуры. В больших организациях с множеством подразделений, отделов и удаленных местоположений может быть трудно обеспечить единый и последовательный процесс инвентаризации;

- отсутствие автоматизации. Ручная инвентаризация не предоставляет возможности для автоматического сбора данных о компьютерах и устройствах. Это может привести к упущению некоторых устройств или неправильному обновлению информации при замене или перемещении оборудования.

Данную проблему можно решить путём разработки информационной системы, обеспечивающей автоматизированную инвентаризацию. Следовательно, цель работы – автоматизировать инвентаризацию локальной сети Амурского государственного университета. Для достижения данной цели необходимо выполнить следующие задачи:

- анализ деятельности предприятия;
- анализ проблемы инвентаризации;
- анализа существующих решений;
- определение требований к разрабатываемой ИС;
- проектирование и реализация ИС.

Разработка позволит упростить и ускорить данный процесс в организации.

1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 Характеристика исследуемой организации

1.1.1 Описание Амурского государственного университета

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Амурский государственный университет» занимается услугами предоставления среднего профессионального и высшего образования по программам бакалавриата, магистратуры и специалитета. Также есть общеобразовательный лицей, предоставляющий среднее и основное общее образование.

Локальная сеть АмГУ распространяется на всю территорию, занимаемую учебными корпусами – 1033 метра полезной площади.

Краткие характеристики ИТ-инфраструктуры АмГУ:

- количество интранет-серверов – 66;
- количество локальных сетей в образовательном учреждении – 1;
- количество терминалов, с которых имеется доступ к сети Интернет – 13;
- общее количество компьютеров – 1696;
- количество компьютеров, с которых имеется доступ к электронным библиотечным системам – 1513;
- всего компьютерных классов – 47.

Обеспечением работы ИТ-инфраструктуры занимается специализированный отдел – центр цифровой трансформации и технического обеспечения.

1.1.2 Описание ЦЦТиТО

Отдел «Центр цифровой трансформации и технического обеспечения» (ЦЦТиТО) занимается обслуживанием технического обеспечения АмГУ, а также сопровождением веб-сайта и базы данных, реализованной в 1С: Предприятие в конфигурации Университет ПРОФ.

ЦЦТиТО сотрудничает с множеством подразделений ФГБОУ ВО АмГУ, выполняет различные задачи, связанные с внедрением и поддержкой цифровых технологий в управленческих и образовательных процессах университета (рис. 1).



Рисунок 1 – Организационная структура

Основными задачами ЦЦТиТО являются:

- развитие современных информационных технологий в АмГУ;
- внедрение, развитие и техническая поддержка электронной информационно-образовательной среды;
- обеспечение доступа сотрудников и студентов АмГУ информационным системам АмГУ;
- контроль за техническим состоянием средств вычислительной техники, телекоммуникационного оборудования, телефонной сети;
- установка и настройка аппаратных и программных средств информатизации в структурных подразделениях АмГУ;
- развитие и сопровождение вычислительной системы АмГУ.

Таблица 1 – Характеристики компьютеров, используемых работниками ЦЦТиТО

Наименование	Модель
Оперативная память	DDR4-2666 16 Гб
Дисковый накопитель	2.5" 512GB SSD SATA III
Процессор	Intel® Core™ i3 10-го поколения
Видеокарта	Intel® HD Graphics
Операционная система	Windows 10 Pro 22H2
Материнская плата	DEPO DPH410T ДАЦН.469535.009
Экран	23,8" Full HD IPS (1920 x 1080)

Также компьютеры используются в учебных лабораториях для выполнения студентами работ, в результате которых они закрепляют навыки компьютерной грамотности или работы с прикладным программным обеспечением.

Таблица 2 – Типовые компьютеры учебных лабораторий

Наименование	Модель
Оперативная память	DDR3-1333 8 Гб
Дисковый накопитель	240 Гб SATA III HDD
Процессор	Intel® Core™ i3 2-го поколения
Видеокарта	Intel® HD Graphics
Операционная система	Windows 10 Pro
Материнская плата	DEPO DPH410T ДАЦН.469535.009
Экран	19" SXGA IPS (1280 x 1024)

На все компьютеры устанавливается IP-адрес при помощи DHCP.

DHCP (Dynamic Host Configuration Protocol) – это протокол, который используется в компьютерных сетях для автоматической настройки IP-адресов и других сетевых параметров для устройств, подключенных к сети. Он позволяет сетевым устройствам получать IP-адрес, маску подсети, адрес шлюза по умолчанию и другие настройки, не требуя их ручной конфигурации.

Работа протокола DHCP основана на следующих основных компонентах:

– «DHCP-сервер» – сервер, который управляет и выдает IP-адреса и другие сетевые параметры устройствам в сети. DHCP-сервер хранит пул доступных IP-адресов, а также настройки сети, которые будут предоставлены

клиентам;

- «DHCP-клиент» – устройство, которое запрашивает сетевые настройки у DHCP-сервера. Клиенты отправляют DHCP-запросы на локальной сети для получения доступных IP-адресов и других параметров;

- «DHCP-ретранслятор» – это сетевое устройство (например, маршрутизатор), которое пересылает DHCP-запросы и ответы между DHCP-клиентами и DHCP-сервером в разных сегментах сети.

Процесс работы DHCP выглядит следующим образом:

- «DHCP-клиент» отправляет широковещательный DHCP-запрос (DHCP Discover) на локальной сети для поиска доступных DHCP-серверов;

- «DHCP-серверы», получившие DHCP Discover, отвечают с предложением (DHCP Offer), содержащим доступные IP-адреса и другие настройки;

- «DHCP-клиент» выбирает одно из предложений и отправляет запрос на аренду IP-адреса (DHCP Request) выбранному DHCP-серверу;

- «DHCP-сервер» подтверждает аренду IP-адреса (DHCP Ack) и предоставляет клиенту запрошенные сетевые настройки;

- «DHCP-клиент» использует полученные настройки для настройки своего сетевого соединения и начинает использовать полученный IP-адрес.

Кроме того, DHCP поддерживает механизмы обновления и продления аренды IP-адреса, а также реализует возможность повторного использования освободившихся IP-адресов.

Сеть АмГУ проявляет динамичность и подвержена частым изменениям, которые связаны с появлением новых компьютеров и расширением сетевых сегментов, включая компьютерные лаборатории. Также в рамках процесса наблюдается замещение старых компьютеров на новые модели. Эти изменения являются неотъемлемой частью развития сети, они способствуют постепенному расширению возможностей и обеспечивают совершенствование инфраструктуры АмГУ. Динамический характер сети

позволяет осуществлять постоянное обновление и прогрессирование в соответствии с потребностями и требованиями организации.

Для эффективной организации и управления динамичной сетью АмГУ возникает необходимость в систематическом и аккуратном проведении инвентаризации сетевых ресурсов. Инвентаризация сети является важной задачей, направленной на документирование и анализ имеющегося аппаратного и программного обеспечения, а также сетевых устройств, с целью обеспечения их контроля, обновления и эффективного использования.

Учитывая динамическую природу сети АмГУ, процесс инвентаризации представляет собой непрерывную деятельность, которая должна быть осуществлена с регулярной периодичностью. Новые компьютеры, добавленные в сеть, должны быть идентифицированы и документированы, а устаревшее оборудование должно быть обновлено и заменено. Кроме того, расширение сетевых сегментов и введение новых компьютерных лабораторий требуют точного отслеживания сетевых активов и обеспечения их правильной конфигурации.

Инвентаризация сети позволит организации эффективно управлять своими ресурсами, оптимизировать процессы обновления и модернизации, а также обеспечить безопасность и стабильность сетевой инфраструктуры. Благодаря этому подходу, сеть АмГУ сможет успешно справляться с растущими потребностями и изменениями, обеспечивая бесперебойную работу и удовлетворение требований всех её пользователей.

1.2 Характеристика инвентаризации сети

1.2.1 Сущность инвентаризации сети

Инвентаризация локальной сети – это процесс сбора и документирования информации о всех устройствах и ресурсах, находящихся в локальной сети организации или предприятия. Она включает в себя идентификацию, классификацию и регистрацию всех сетевых компонентов, таких как компьютеры, серверы, маршрутизаторы, коммутаторы, принтеры,

устройства хранения данных и другие сетевые устройства.

Суть инвентаризации локальной сети заключается в создании полного и точного списка всех устройств и ресурсов, которые существуют в сети, а также их основных характеристик, таких как IP-адреса, MAC-адреса, операционные системы, установленное программное обеспечение и другие атрибуты. Эта информация позволяет организации лучше понять свою сетевую инфраструктуру, управлять ресурсами, обеспечивать безопасность и планировать обновления или расширения.

Процесс инвентаризации локальной сети может включать следующие шаги:

- обнаружение устройств. При помощи специальных инструментов или программного обеспечения происходит сканирование сети для обнаружения и идентификации устройств, подключенных к сети;

- сбор информации. Для каждого обнаруженного устройства собирается информация, такая как IP-адрес, MAC-адрес, имя хоста, операционная система, установленное программное обеспечение и другие характеристики;

- классификация и категоризация. Устройства группируются по типам, например, компьютеры, серверы, сетевые устройства и т.д. Это позволяет проводить более удобное администрирование и управление сетью;

- документирование и хранение данных. Информация об устройствах и их характеристиках регистрируется и сохраняется в базе данных или специальных системах управления конфигурацией сети;

- обновление и поддержка. Инвентаризация должна быть процессом, который постоянно обновляется, так как сеть может быть изменена или добавлены новые устройства. Поэтому важно регулярно обновлять информацию и поддерживать актуальность данных.

Инвентаризация локальной сети помогает организациям эффективно управлять своей сетевой инфраструктурой, планировать расходы на оборудование и обновления, а также обеспечивать безопасность и поддержку

сетевых операций.

Целевая аудитория приложений для инвентаризации локальной сети включает следующие группы пользователей:

– аудиторы и консультанты – независимые специалисты или команды, которые занимаются проверкой и анализом информационных систем предприятия. Они могут использовать приложение для сбора данных о сетевых устройствах и анализа информации об инфраструктуре сети;

– системные аналитики и разработчики – специалисты, которые занимаются проектированием, разработкой и оптимизацией информационных систем и сетевых приложений, также сотрудники, которые занимаются установкой, настройкой и обслуживанием компьютеров, серверов и другого сетевого оборудования. Они будут использовать приложение для получения подробной информации о каждом устройстве, включая его параметры, статус и конфигурацию.

Основная цель приложений для инвентаризации локальной сети – обеспечить полную и точную информацию о сетевых устройствах и ресурсах для управления и обслуживания сети.

1.2.2 Методы инвентаризации сети

Проблема инвентаризации известна уже довольно давно. Развитие инвентаризации компьютерных сетей прошло через несколько этапов, каждый из которых представляет собой прогресс в технологиях и методах сбора и управления информацией о компьютерных ресурсах:

– ручная инвентаризация. На ранних этапах компьютерных сетей, когда количество устройств было невелико, инвентаризация проводилась вручную. Сотрудники отвечали за ведение списков и регистрацию компьютеров, периферийных устройств и программного обеспечения. Этот подход был трудоемким и подвержен ошибкам;

– программные инструменты инвентаризации. С развитием компьютерных технологий стали появляться программные инструменты,

предназначенные для автоматизации процесса инвентаризации. Эти инструменты собирали информацию о компьютерных ресурсах, таких как аппаратное обеспечение, операционные системы, установленное программное обеспечение и т.д. Они позволяли проводить более точную и эффективную инвентаризацию;

– сетевая инвентаризация. С развитием сетевых технологий и протоколов, таких как SNMP (Simple Network Management Protocol), стало возможным собирать информацию о компьютерных ресурсах удаленно через сеть. Сетевые инструменты инвентаризации позволяют автоматически обнаруживать и анализировать устройства в сети, собирать информацию о их конфигурации и состоянии;

– централизованная инвентаризация. С появлением централизованных систем управления ресурсами, таких как системы управления конфигурацией (Configuration Management Systems), инвентаризация компьютерных сетей стала более организованной и удобной. Централизованные системы позволяют хранить информацию обо всех ресурсах сети в едином месте, обеспечивая управление и контроль над ними;

– автоматизированная инвентаризация. Современные инструменты инвентаризации компьютерных сетей предлагают автоматизацию процесса сбора, анализа и управления информацией о ресурсах. Они предлагают функции автоматического обнаружения устройств, автоматического обновления информации, оповещения о изменениях и другие возможности для упрощения и оптимизации процесса инвентаризации.

1.2.3 Существующие решения автоматической инвентаризации

Ввиду обширности ИТ-инфраструктуры крупных компаний, её сложности и многообразия типов устройств и сервисов, необходимо разработать комплексный подход, включающий использование специализированных программных и программно-аппаратных решений, для эффективного решения задачи инвентаризации сетевых ресурсов. Такие

комплексы позволяют собирать и анализировать большие объемы данных о сетевой инфраструктуре, автоматизировать процессы инвентаризации, а также обеспечивать централизованное управление и контроль за активами сети. Это позволяет компаниям более точно и оперативно определить состояние своей инфраструктуры, выявить потенциальные уязвимости и проблемы, а также принимать обоснованные решения по обновлению и модернизации системы.

На данный момент самыми популярными на рынке являются следующие программные продукты:

- Spiceworks Network Inventory;
- Lansweeper;
- Open-AudIT;
- OCS Inventory NG;
- Total Network Inventory;
- 10-Страйк Инвентаризация Компьютеров.

Spiceworks Network Inventory, разработанная фирмой: Spiceworks в США – бесплатная программа для инвентаризации компьютеров и сетевых устройств. Она позволяет автоматически обнаруживать устройства в сети, собирать информацию о железе и ПО, а также отслеживать изменения в конфигурации устройств. Spiceworks Network Inventory имеет простой интерфейс и интуитивно понятный процесс установки. Однако, она имеет ограниченные возможности по сравнению с другими платными решениями, и могут возникать проблемы с обновлением информации об устройствах. Программа является свободно распространяемой. Скриншот программы представлен на рисунке 2.

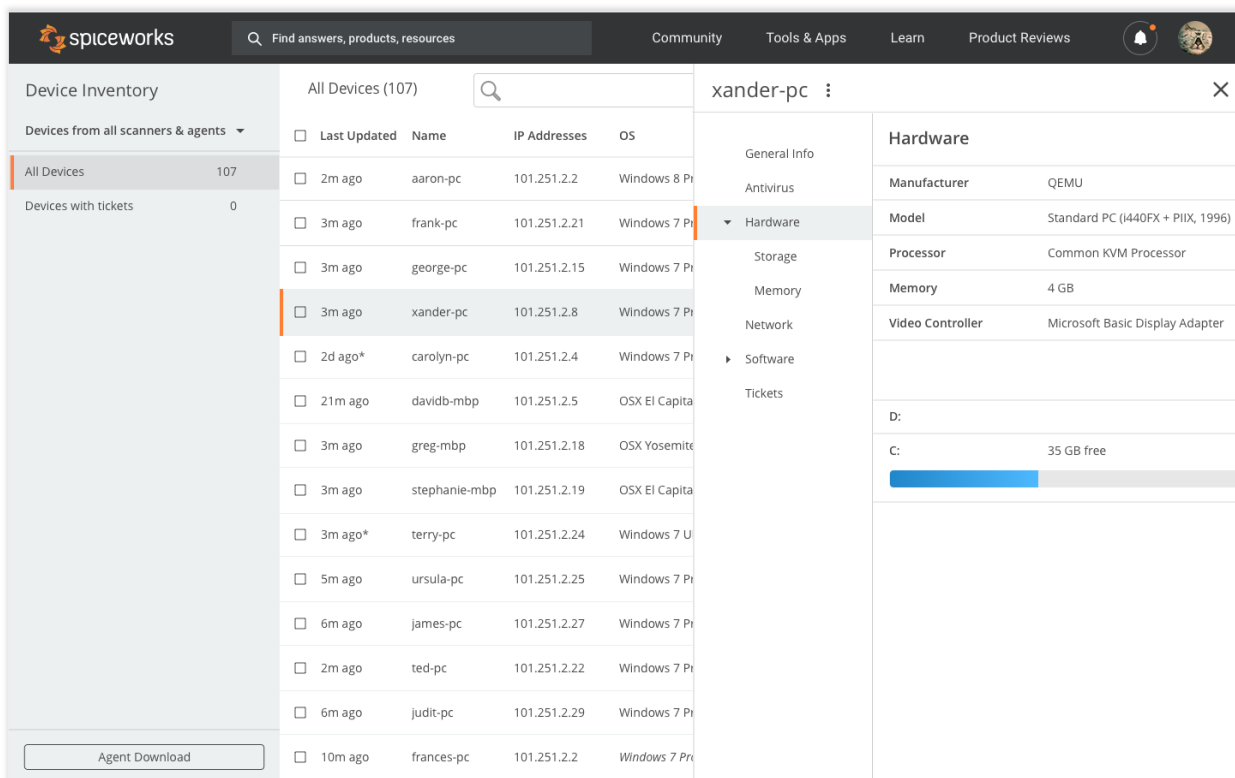


Рисунок 2 – Скриншот программы Spiceworks

Lansweeper, разработанная одноимённой компанией в Бельгии, это программа для инвентаризации сетевых устройств и ПО, которая предоставляет множество функций для автоматического сканирования, отслеживания изменений, анализа данных и создания отчетов. Она поддерживает различные типы устройств и операционных систем, а также интегрируется с другими системами мониторинга и управления. Lansweeper предлагает гибкие настройки и возможности массового редактирования данных, а также предоставляет API для интеграции с другими системами. Программа имеет бесплатную версию с ограничением до 100 устройств. Платная версия начинается от \$2000 в год и от 2000 устройств. Скриншот программы показан на рисунке 3.

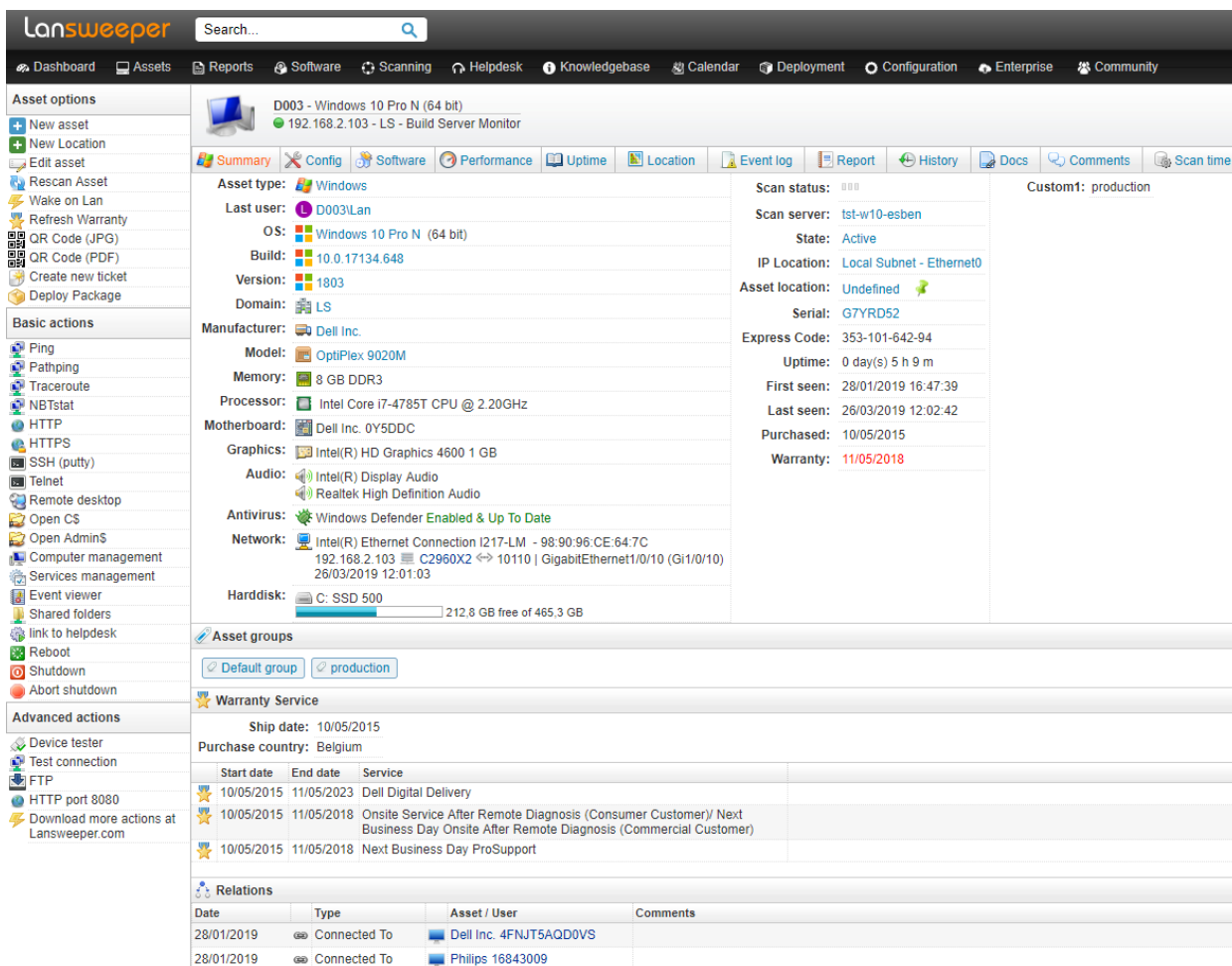


Рисунок 3 – Скриншот программы Lansweeper

Open-AudIT, разрабатываемый Ormantek (Дочерняя компания FirstWave), Австралия, это бесплатная программа для инвентаризации сетевых ресурсов, которая позволяет автоматически сканировать и анализировать все компьютеры, серверы, принтеры, роутеры и другие устройства в сети. Программа обладает широким набором функций, включая детальный анализ железа и софта, создание отчетов, а также возможность отслеживания изменений в конфигурации устройств. Open-AudIT поддерживает различные операционные системы, включая Windows, Linux и macOS. Программа имеет открытый исходный код и доступна для загрузки на официальном сайте проекта. На рисунке 4 показан скриншот программы.

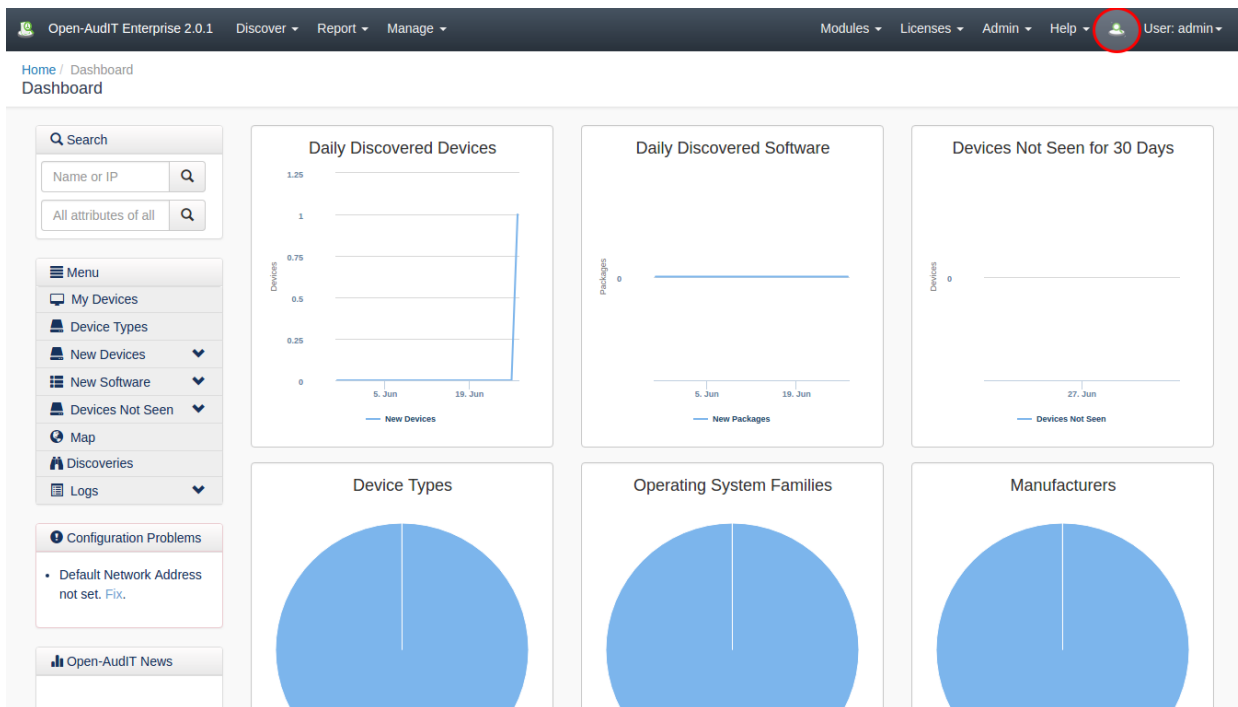


Рисунок 4 – Скриншот программы Open-Audit

OCS Inventory NG (рис. 5), разработчиком которого является FactorFX, Франция, это бесплатное ПО с открытым исходным кодом для инвентаризации сетевых ресурсов и управления ПО. Оно поддерживает большое количество операционных систем и языков, и может работать как на локальных, так и на удаленных компьютерах. OCS Inventory NG предоставляет подробную информацию о аппаратном и программном обеспечении, установленных на компьютерах, а также позволяет проводить удаленное управление и развертывание программного обеспечения. С помощью OCS Inventory NG можно также создавать отчеты и настраивать оповещения для более эффективного управления сетью. Программа является свободно распространяемой. На рисунке 5 показан скриншот программы.

Ver. 2.0RC3

Show: 10

Restrict view: Filter X

Add column: X

19 Result(s) (Download)

Tag Type X	Last inventory X	Computer X	User X	Operating system X	RAM (MB) X	CPU (MHz) X	Delete
PORTABLE	2011-03-16 19:25:10	BZO	Guillaume	Microsoft Windows XP Professional	2000	2393	X
SERVEUR	2011-03-16 07:36:03	nunux	root	Debian GNU/Linux 6.0 (squeeze)	757	2171	X
SMARTPHONE	2011-03-15 20:07:35	KS-20		Windows CE	209		X
PC FIXE	2011-02-28 21:52:27	GUILLAUME	goufa	Microsoft Windows XP Professional	2048	2671	X
SERVEUR	2011-02-22 13:47:01	W16753100IP2	Administrateur	Microsoft(R) Windows(R) Server 2003, Enterprise Edition	512	2662	X
SERVEUR	2011-02-22 13:43:50	LABO-AV-2K-04	Administrateur	Microsoft(R) Windows(R) Server 2003, Standard Edition	1280	995	X
SERVEUR	2011-01-05 18:48:11	thekid	root	OpenBSD AMD Athlon(tm) 64 Processor 3500+	2046	0	X
PC FIXE	2011-01-05 18:47:44	SNOOPY-DIDIER	didier	Microsoft Windows 7 Edition Intégrale	2048	2200	X
VM OPENBSD	2011-01-05 18:47:00	ocs-server	luser	OpenBSD Intel(R) Pentium(R) 4 CPU 2.80GHz (GenuineIntel 686-class)	2039	0	X
SERVEUR	2011-01-05 18:46:15	ocs-opensuse	root	opensUSE 11.3 (i586)	499	1919	X

0.1

Рисунок 5 – Скриншот программы OCS Inventory NG

Total Network Inventory, разрабатываемый компанией Softinventive Lab (рис. 6), США это программа для инвентаризации сети, разработанная компанией Softinventive Lab Inc. Она позволяет сканировать компьютеры и другие устройства в сети, собирать информацию об установленном программном обеспечении, железных компонентах, серийных номерах и других характеристиках оборудования. Программа позволяет создавать пользовательские отчеты, проводить анализ данных и контролировать лицензии на программное обеспечение. Она также поддерживает удаленный доступ к компьютерам и устройствам в сети, а также позволяет проводить диагностику и управлять оборудованием. Total Network Inventory поддерживает различные операционные системы, включая Windows, Linux, macOS и др. Стоимость данной программы начинается от \$90 в год за 25 устройств. Имеет 60-дневную пробную версию.

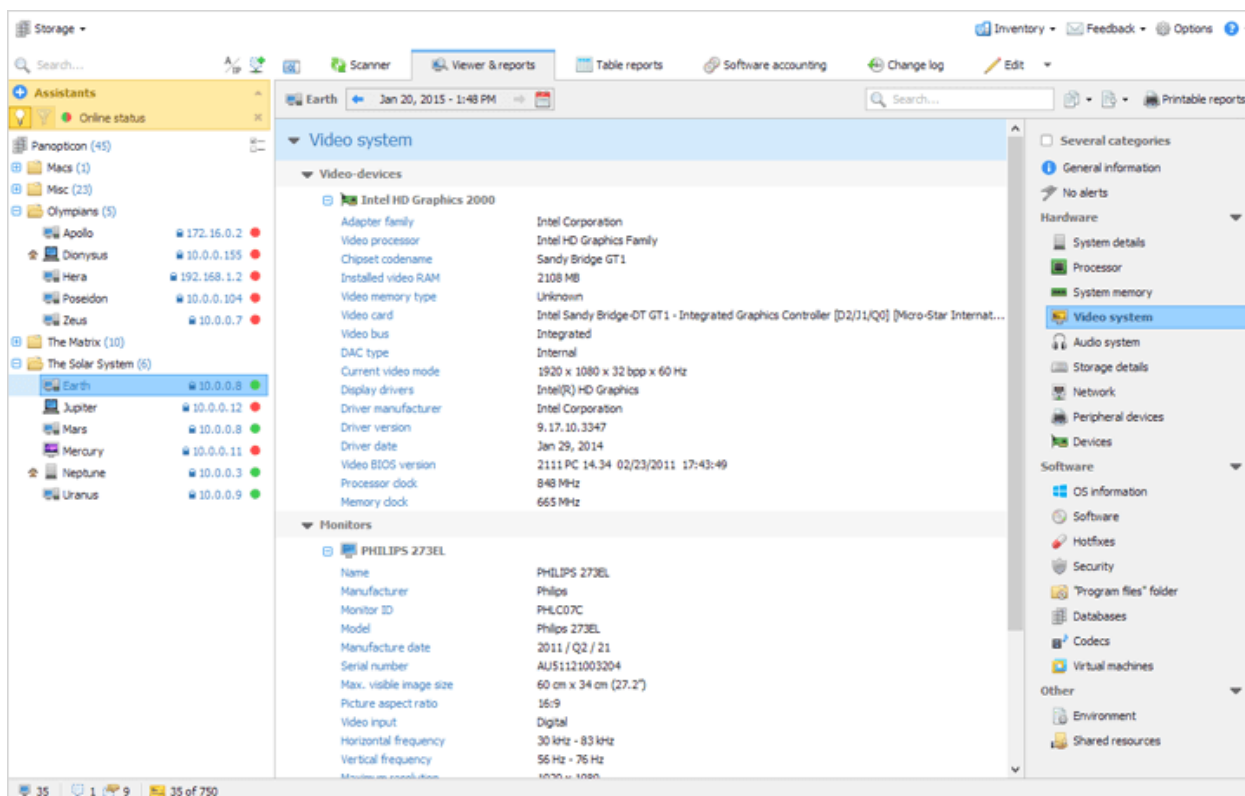


Рисунок 6 – Скриншот программы Total Network Inventory

10-Страйк: Инвентаризация Компьютеров – это отечественная программа для инвентаризации сетевых компьютеров и устройств (рис 7). Она позволяет автоматически сканировать сеть и собирать информацию об установленном программном обеспечении, железе и других параметрах компьютеров и устройств. Программа позволяет проводить инвентаризацию удаленно, в том числе через Интернет. Она также позволяет проводить мониторинг состояния компьютеров и уведомлять администраторов о проблемах. Программа поддерживает автоматическое создание отчетов, а также позволяет экспортировать данные в различных форматах. 10-Strike Inventory имеет простой интерфейс и легко настраивается. Программа имеет стоимость, начинающуюся от 10000 рублей за 50 устройств, и предоставляет пожизненную лицензию на все версии программы, выпущенные в течение года.

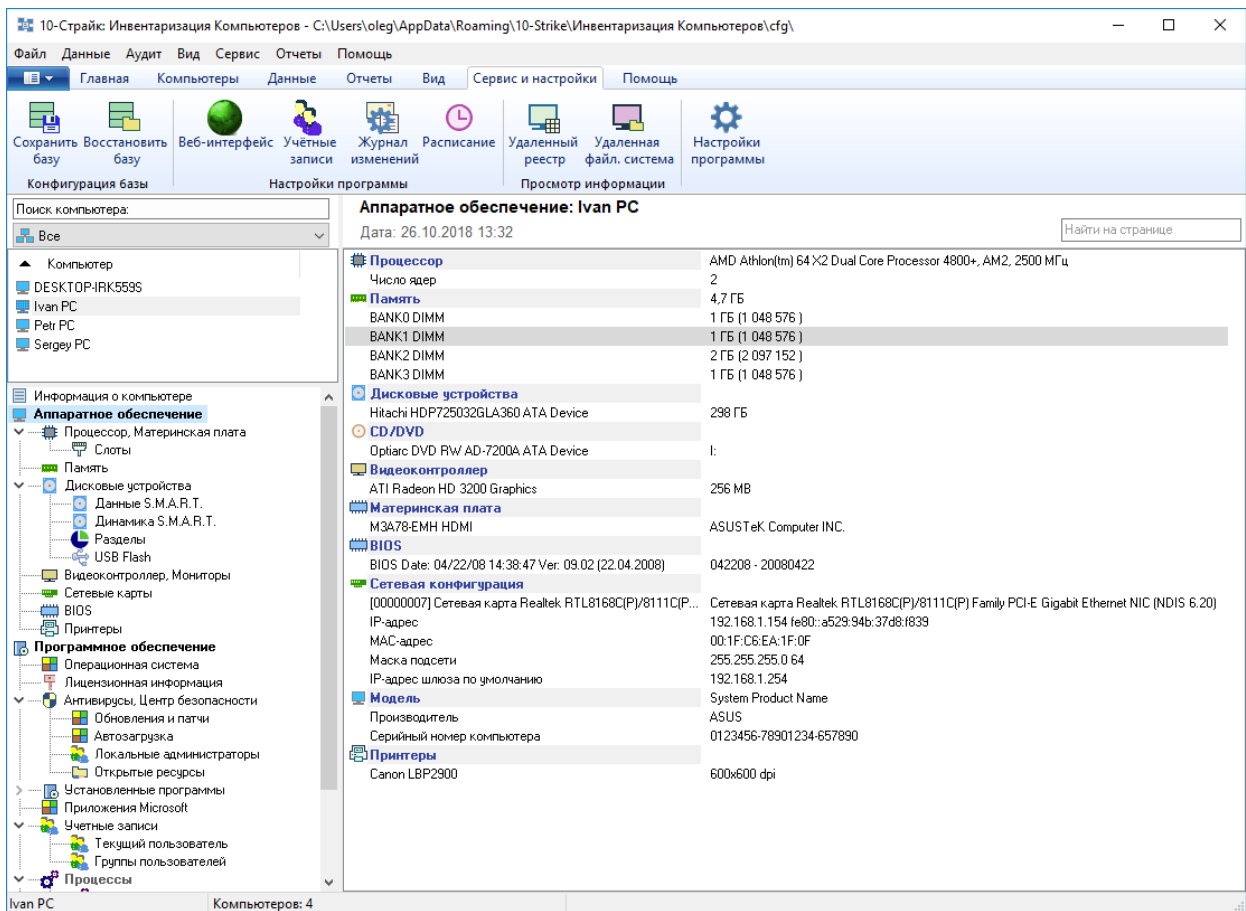


Рисунок 7 – Скриншот программы 10-Страйк: Инвентаризация Компьютеров

Из проведенного анализа следует, что большинство существующих программных решений для инвентаризации сетей имеют широкий набор функций, который может не соответствовать потребностям малых и средних предприятий. Большинство программ предназначены для крупных сетей, имеют платную лицензию и высокую стоимость.

Однако, существует несколько бесплатных программ или программ с бесплатной версией, которые могут быть полезны для малых и средних предприятий. Например, Spiceworks, которая обладает возможностью инвентаризации как локальных, так и удаленных сетей, а также имеет ряд дополнительных функций, включая систему уведомлений и интеграцию с сервисами облачного хранения, но является зарубежным ПО. Но, данное решение разрабатывается и сопровождается зарубежной организацией, что при текущей политической ситуации и курсе на отечественных разработчиков

делает использование данного продукта невостребованным.

В контексте рассмотрения локальной сети АмГУ предлагается разработать собственный программный продукт, специализированный под требования данной организации. Данный подход является целесообразным, поскольку он позволяет эффективно удовлетворить потребности организации и обеспечить ряд преимуществ.

Один из основных преимуществ разработки собственного программного продукта для локальной сети АмГУ заключается в возможности оперативного обновления программы для добавления нового функционала и исправления ошибок. Это позволит поддерживать программу в актуальном состоянии и адаптировать ее под изменяющиеся требования и потребности организации. Кроме того, наличие собственного программного продукта обеспечивает возможность проведения консультаций с персоналом напрямую, что способствует более эффективному взаимодействию и оперативному реагированию на потребности пользователей.

Одним из ключевых аспектов разработки данного программного продукта является ограничение его функционала с целью поддержания небольшого размера и оптимальной производительности. Это позволит снизить потребление ресурсов и обеспечить более быструю работу программы. Кроме того, пользовательский интерфейс будет разработан с учетом требований, направленных на минимизацию времени, необходимого для обучения персонала работе с программой. Это обеспечит более удобное и интуитивно понятное взаимодействие пользователей с программным продуктом, сократив время на обучение и повысив эффективность использования программы.

Таким образом, разработка собственного программного продукта, специализирующегося под требования локальной сети АмГУ, является целесообразным решением.

2 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА

Для создания ИС необходимо выполнить ряд шагов разработки:

- сформулировать требования;
- выбрать инструменты и средства разработки;
- описать модель ИС;
- спроектировать структуру модулей и классов;
- спроектировать базу данных;
- спроектировать пользовательский интерфейс;
- реализовать ИС на основе выбранных компонентов.

2.1 Формулировка требований к информационной системе

К разрабатываемому программному продукту предъявляются следующие требования:

- совместимость с ОС Windows и Linux;
- эффективное выполнение целевых функций программы;
- отслеживание изменений состояния сканируемых устройств;
- возможность хранения и вывода данных;
- простой, интуитивно понятный пользовательский интерфейс.

2.2 Выбор инструментов и средств разработки

2.2.1 Выбор языка программирования

Использование языка C# при создании приложения для инвентаризации компьютеров имеет несколько преимуществ:

– удобство и продуктивность разработки. C# является высокоуровневым языком программирования, который предоставляет разнообразные инструменты и библиотеки для упрощения разработки. Синтаксис языка C# интуитивно понятен и читабелен, что способствует быстрой и эффективной разработке приложений;

- интеграция с платформой .NET: C# является основным языком

программирования платформы .NET, что позволяет использовать богатый набор классов, библиотек и инструментов для разработки приложений. Это обеспечивает широкие возможности в области работы с сетью, базами данных, пользовательским интерфейсом и другими компонентами, необходимыми для создания приложения для инвентаризации;

– поддержка асинхронного программирования: Язык C# обладает встроенной поддержкой асинхронного программирования, что позволяет эффективно работать с асинхронными операциями, такими как получение данных из сети или выполнение запросов к базе данных. Это особенно полезно при разработке приложений для инвентаризации, где могут быть задействованы долгие операции сетевого сканирования или сбора информации о компьютерах;

– масштабируемость и переносимость. Приложения, написанные на C#, могут быть легко масштабированы и перенесены на различные платформы и операционные системы, включая Windows, Linux и macOS. Это обеспечивает гибкость и универсальность при создании приложений для инвентаризации, которые могут быть развернуты в различных средах.

В целом использование языка C# при создании приложения для инвентаризации обеспечивает разработчикам мощные инструменты, простоту синтаксиса и возможность эффективно работать с сетью, базами данных и асинхронными операциями, что способствует разработке надежных и функциональных приложений.

2.2.2 Выбор СУБД

В качестве СУБД была выбрана SQLite. Язык C# и платформа .NET предлагает работу с базами данных в этом решении как с классами, что освобождает от необходимости написания SQL-запросов, вместо них можно воспользоваться LINQ – библиотекой для работы с обобщениями, а также ускоряет работу программы в плане записи/чтения данных, так как SQLite не требует отдельного сервера для своей работы.

SQLite – это компактная встроенная реляционная база данных, которая позволяет сохранять и управлять данными в локальном хранилище. Она является самодостаточной, не требует отдельного сервера баз данных и может быть легко интегрирована в различные приложения. Некоторые особенности и преимущества SQLite:

- простота использования. SQLite предоставляет простой и интуитивно понятный интерфейс для работы с базой данных. Он имеет небольшой размер и минимальные требования к ресурсам, что делает его идеальным выбором для мобильных и встроенных приложений;

- быстрота. SQLite обеспечивает высокую производительность и быстрый доступ к данным. Он предлагает оптимизированные механизмы индексации и выполнения запросов, что позволяет эффективно работать с большими объемами данных;

- надежность. SQLite обеспечивает надежное хранение данных и обеспечивает целостность базы данных. Он поддерживает транзакции и обеспечивает согласованность данных при многопользовательском доступе;

- поддержка множества платформ. SQLite доступен для различных операционных систем, включая Windows, macOS, Linux и мобильные платформы, такие как Android и iOS. Он также может быть интегрирован в различные языки программирования, включая C#, Java, Python и другие;

- обширный набор функций. SQLite предоставляет множество функций для работы с данными, включая поддержку различных типов данных, агрегатных функций, операторов и триггеров. Он также поддерживает SQL-запросы, что делает его мощным инструментом для манипуляции данными.

SQLite широко применяется в различных сферах, таких как мобильные приложения, встроенные системы, настольные программы, веб-сайты и т.д. Он предоставляет удобный способ хранения и управления данными без необходимости установки и настройки отдельного сервера баз данных.

2.2.3 Выбор графической подсистемы

При работе с языком C# и платформой .NET существуют такие графические подсистемы, как Windows Forms, WPF, UWP и MAUI. Два последних варианта не поддерживают работу в Windows 7, что важно для данного программного продукта, так как в организации всё ещё есть компьютеры с данной версией операционной системы, так что эти варианты не предпочтительны.

Между Windows Forms и Windows Presentation Foundation есть такие различия, как:

- графические возможности: WPF предлагает более современные и гибкие возможности визуализации, включая векторную графику, анимацию, привязку данных, стили и шаблоны. Это позволяет создавать более богатые и интерактивные пользовательские интерфейсы. В то время как Windows Forms имеет более простую и ограниченную графическую модель;

- расположение элементов: WPF предлагает более гибкое расположение элементов с помощью панелей (panels) и возможностью использовать сетку (grid) для более точного позиционирования. Windows Forms использует простые контейнеры и систему координат для размещения элементов;

- стили и темы: WPF предоставляет мощные средства для создания стилей и тем, что позволяет легко изменять внешний вид приложения. Windows Forms имеет ограниченную поддержку стилей и тем;

- поддержка мультимедиа: WPF имеет встроенную поддержку мультимедиа, включая воспроизведение аудио и видео. Windows Forms не имеет такой поддержки без предварительной настройки.

Для данного проекта более предпочтительной подсистемой будет WPF, так как требуется интуитивно понятный и оптимальный для продолжительной работы графический интерфейс, а также привязка данных для сокращения объёмов создаваемого программного кода.

2.2.4 Выбор интегрированной среды разработки

Для разработки проекта на .NET существует два наиболее популярных решения: Visual Studio от Microsoft и Rider от JetBrains.

Из преимуществ Visual Studio можно отметить:

– расширенный IntelliSense и IntelliCode. Это интеллектуальная функция, предоставляемая в интегрированных средах разработки, таких как Visual Studio, для языка программирования C#. Она обеспечивает автоматическое предсказание и подсказку кода, что позволяет разработчикам увеличить производительность и снизить количество ошибок. IntelliSense предлагает доступные методы, свойства, переменные и другие элементы кода во время набора, а также предоставляет информацию о типах данных и параметрах функций, что помогает разработчику легче и быстрее писать код, облегчает отладку и повышает качество конечного продукта. Благодаря IntelliSense разработчики получают контекстную информацию о коде на лету, что значительно улучшает их опыт разработки и способствует более эффективному использованию языка C#.

– бесплатность. У Visual Studio есть бесплатная Community-версия, тогда как все версии Rider распространяются платно, однако есть версия для студентов, но для её получения необходимо подтвердить соответствующий статус при помощи почты вуза.

– Blend. Для расширенной работы с проектами .NET, такими как UWP, WPF и MAUI, существует Blend for Visual Studio. Это расширенная версия Visual Studio специально для проектов, графическая подсистема которых разрабатывается при помощи языка разметки XAML. В этой среде разработки, по сравнению с обычной Visual Studio, есть расширенная работа с анимацией, таких как поддержка временных линий, ключевых кадров, настройки трансформаций и переходов, также все элементы интерфейса представлены в виде дерева, что позволяет визуально контролировать их положение в иерархии при вёрстке.

На стороне Rider был отмечен более строгий контроль над именовани

переменных, методов, полей, структур, классов и т.д. При должной компетенции разработчик обязан соблюдать их в любом случае, но вероятность ошибки всё ещё есть.

Исходя из этого, для разработки выбрана Blend for Visual Studio.

2.3 Модель работы ИС

2.3.1 Описание модели

Планируется, что полноценный программный продукт будет установлен на компьютер системного администратора («компьютер А»), также будет облегчённая версия (так называемый агент) для установки на подчинённые компьютерные устройства («компьютер Б») в виде службы, работающей в фоне, для получения данных от главного приложения по запросу, а также обновления информации о компьютере (его идентификатор, задаваемый администратором, и адрес сетевого расположения), на который и установлен агент, через равные промежутки времени. Соответствующая структура показана на рисунке 8. Главное приложение должно иметь прямую связь с базой данных для организации продуктивного рабочего процесса администратора ввиду быстрой записи и не менее быстрого получения данных, обеспечивая возможность сравнения состояния определённого компьютера с его предыдущими состояниями при их наличии.



Рисунок 8 – Модель работы ИС

3.3.2 Описание работы ИС

При сканировании последовательно происходят следующие процессы:

- формирование списка компьютеров;
- циклический обход списка компьютеров, включающий отправку запроса на сканирование и ожидание ответа;
- получение ответа, преобразование в вид структурированных данных, одновременное отображение пользователю и запись в базу данных.

Схематизация данного алгоритма показана на рисунке 9.

Также необходимо реализовать фоновое сканирование, которое будет проходить по идентичному алгоритму, но запускаться через равные промежутки времени.

Важной задачей будет реализация базы данных для хранения информации о компьютерах, что позволит хранить данные о результатах процессов сканирования и сравнивать при необходимости, что соответствует принципам централизованной и автоматизированной инвентаризации.

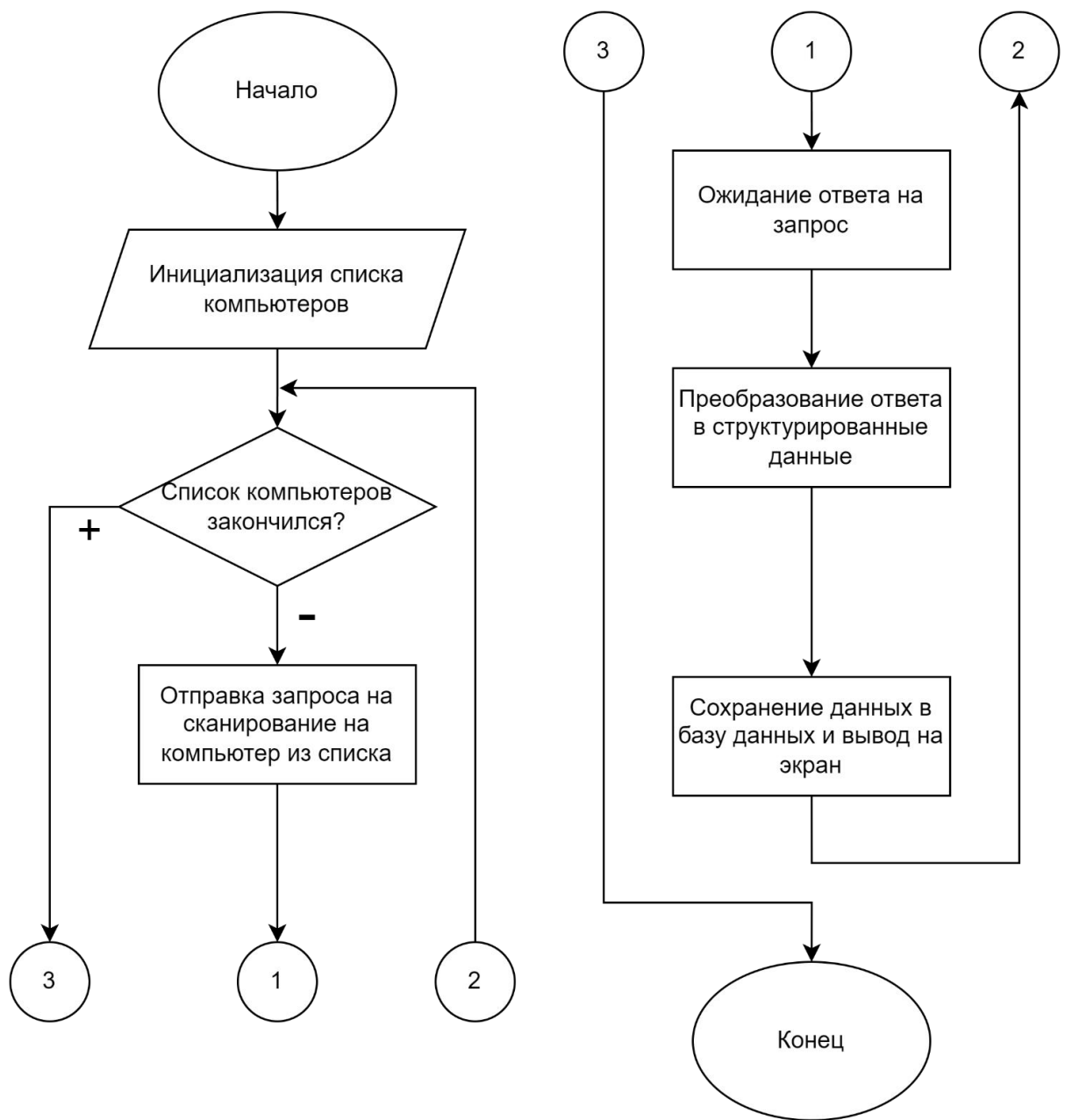


Рисунок 9 –Блок-схемы процесса сканирования

Для записи данных о компьютерах при первичной настройке приложения понадобится так называемое приложение-агент, которое будет установлено на все компьютеры в сети и будет отвечать на запросы главного приложения. Для этого может понадобиться размещение приложения как службы в операционной системе, а также расширенный доступ к ней (права администратора).

2.4 Проектирование структуры модулей и классов

Предполагается поддержка нескольких операционных систем: Windows и Linux. Для правильной организации поддержки необходимо грамотно организовать модульную структуру приложения, а именно выделить так называемое ядро, которое будет одинаково работать на обеих платформах и открывать доступ к различному по реализации, но идентичному по содержанию функционалу. Такая структура позволит упростить написание и сопровождение уже написанного кода. Подходящая структура изображена на рисунке 10. Пользовательский интерфейс и сканирование будут реализованы при использовании разных библиотек, так как единого стандарта сканирования между этими операционными системами не предусмотрено. Похожая ситуация с пользовательским интерфейсом. Существуют кроссплатформенные решения, например, Avalonia, но одинаковая работа на обеих системах не гарантируется. Хоть проект и имеет немалую популярность, следует использовать разные фреймворки для разных платформ.

С базой данных ситуация обстоит проще: SQLite изначально будет работать одинаково как на Windows, так и на Linux, так как операции работы с базой данных описаны на более высоком уровне, который не зависит от особенностей операционных систем, то же можно сказать и про организацию обмена пакетами, но в контексте среды .NET Core.

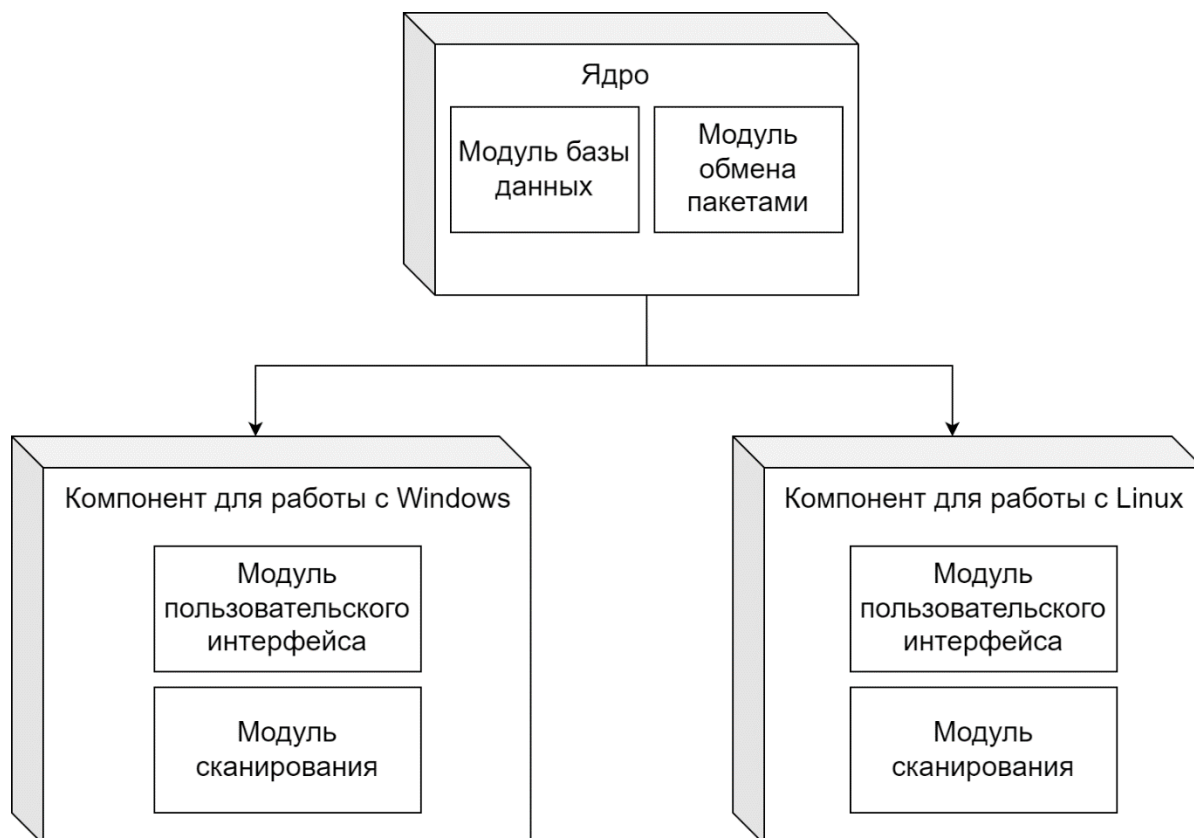


Рисунок 10 – Примерная модульная структура программного продукта

2.5 Проектирование базы данных

2.5.1 Инфологическое проектирование

Определён такой набор сущностей:

- компьютер (хранит информацию об определённом компьютере как о конечной точке сети);
- процессор (хранит информацию о процессорах, установленных в компьютер);
- видеоадаптер (хранит информацию о видеоадаптерах, установленных в компьютеры);
- ОЗУ (хранит информацию о модулях оперативной памяти, установленных в компьютеры);
- дисковый накопитель (хранит информацию о физических дисковых накопителях, установленных в компьютер);
- логический диск (хранит информацию о локальных дисках,

настроенных на компьютерах);

– операционная система (хранит информацию об операционных системах, установленных на компьютерах);

– приложение (хранит информацию о приложениях, установленных на компьютерах);

– событие (хранит информацию о событиях, происходящих в результате инвентаризации, для отображения уведомлений администратору и хранения их истории).

Теперь нужно определить набор атрибутов для каждой сущности.

Сущность «Компьютер» необходима программе для информации для подключения к компьютерам в сети, поэтому лучше всего будет хранить в ней строковый идентификатор компьютера, задаваемый администратором, и адрес подключения к компьютеру. Состав сущности показан на таблице 3.

Таблица 3 – Атрибуты сущности «Компьютер»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Текстовый	–	DESKTOP-1C598KV
Физический адрес	IP-адрес компьютера	Текстовый	–	127.0.0.1

Сущность «Процессор» нужна программе для хранения информации об одноимённом компоненте системной платы. В ней будут храниться идентификатор, назначаемый СУБД, наименование процессора и дополнительную ознакомительную информацию – в этом случае выбраны количество ядер и частота. Структура сущности показана на таблице 4.

Таблица 4 – Атрибуты сущности «Процессор»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	3
Наименование	Наименование процессора	Текстовый	–	AMD Ryzen 5 3550h with Radeon Vega Graphics
Количество ядер	Количество ядер процессора	Числовой	0-65535	4
Частота	Базовая частота работы процессора	Числовой	0-65535	2100

В сущности, под названием «Видеоадаптер» хранится информация о наименовании и о какой-нибудь дополнительной характеристике самого видеоадаптера, например, количество видеопамяти. Атрибуты этой сущности показаны в таблице 5.

Таблица 5 – Атрибуты сущности «Видеоадаптер»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	2
Наименование	Наименование видеоадаптера	Текстовый	–	Radeon RX 560X
Количество видеопамяти	Количество памяти, используемое видеоадаптером	Числовой	0-65535	8192

Сущность «ОЗУ» хранится информация о наименовании, частоте работы и ёмкости модуля оперативной памяти. Структура этой сущности показана в таблице 6.

Таблица 6 – Атрибуты сущности «ОЗУ»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	7
Наименование	Наименование модуля ОЗУ	Текстовый	–	HMA81GS6CJR8N-VK
Частота работы	Частота работы модуля ОЗУ	Числовой	0-65535	2400
Ёмкость	Ёмкость модуля ОЗУ	Числовой	0-65535	8192

Сущность «Физический диск» хранит информацию о дисковых накопителях компьютеров. В её состав включены наименование, общая ёмкость накопителя, а также его статус.

Таблица 7 – Атрибуты сущности «Физический диск»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	6
Наименование	Наименование физического диска	Текстовый	–	ST1000LM035-1RK172
Общая ёмкость	Общая ёмкость физического диска	Числовой	0-65535	120
Статус	Статус физического диска	Текстовый	–	Ok

Сущность «Логический диск», в отличие от сущности «Физический диск», хранит информацию о логических томах, необходимую для администратора, ее состав представлена в таблице 8.

Таблица 8 – Атрибуты сущности «Логический диск»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	11
Буква диска	Буква, присвоенная при создании тома	Текстовый	–	C:
Общая ёмкость	Общая ёмкость логического диска, выраженная в гигабайтах	Числовой	0-65535	120
Свободное пространство	Ёмкость незанятого пространства, выраженная в гигабайтах	Числовой	0-65535	10
Процент свободного пространства	Ёмкость незанятого пространства, выраженная в процентах от общего пространства	Числовой	0-100	10

Таблица 9 – Атрибуты сущности «Операционная система»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	13
Наименование	Наименование операционной системы	Текстовый	–	Майкрософт Windows 11 Pro
Версия	Версия ОС	Текстовый	–	10.0.22623
Сборка	Номер сборки ОС	Текстовый	–	22623
Разрядность	Разрядность ОС	Текстовый	–	64-разрядная

Таблица 10 – Атрибуты сущности «Приложение»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	31
Наименование	Наименование приложения	Текстовый	–	Android Studio
Поставщик	Поставщик приложения	Текстовый	–	Google LLC
Версия	Версия приложения	Текстовый	–	2022.1

Таблица 11 – Атрибуты сущности «Событие»

Название атрибута	Описание	Тип данных	Диапазон значений	Пример атрибута
Идентификатор	Однозначно идентифицирует экземпляр сущности в масштабе таблицы	Числовой	0-65535	40
Текст сообщения	Текст, выводимый пользователю	Текстовый	–	В компьютер 1 установили приложение Android Studio, поставщик: Неизвестен
Дата возникновения	Дата возникновения события	Дата	01.01.2000 00:00 – 31.12.3000 23:59	13.06.2023 05:19

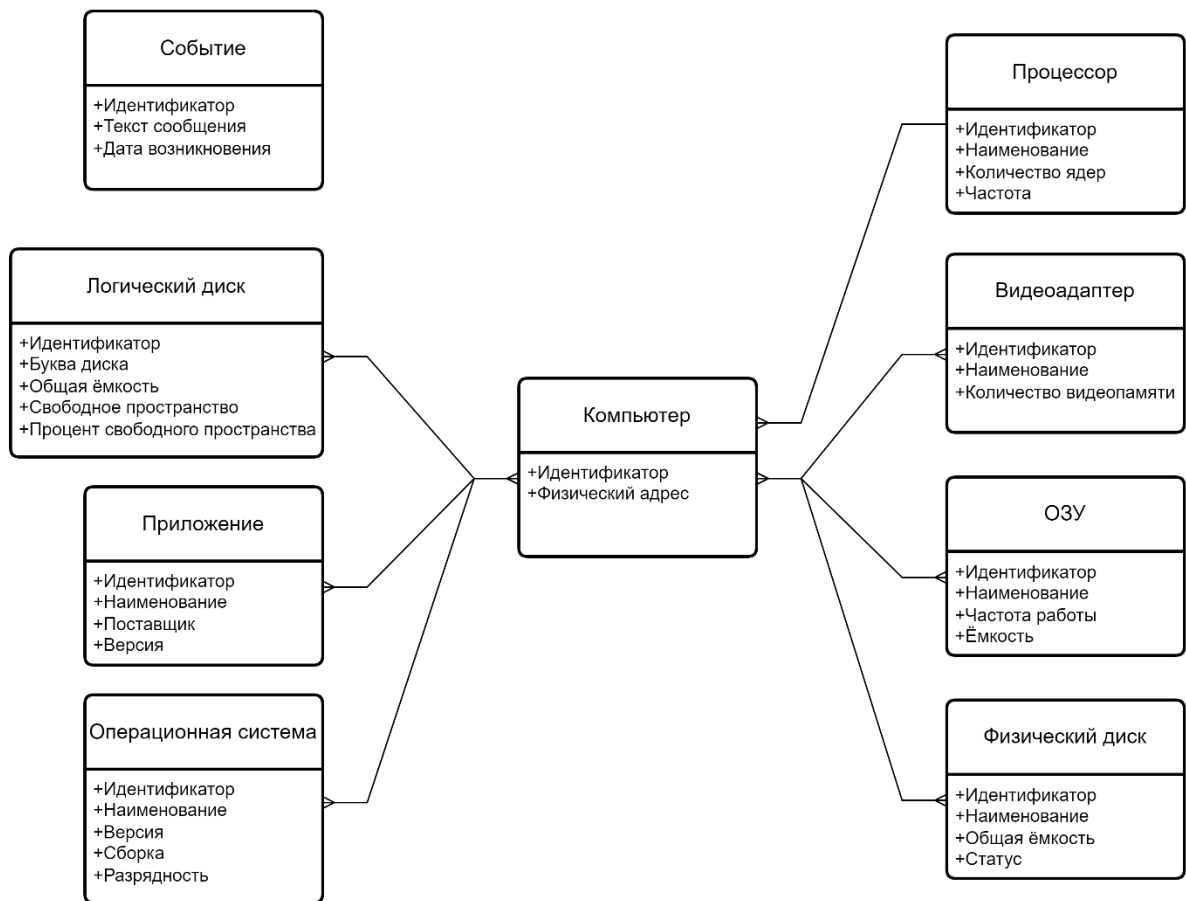


Рисунок 11 – Инфологическая модель

2.5.2 Логическое проектирование

Отношение «Компьютер-Процессор» изображено на рисунке 12. Суть связи этого отношения заключается в том, что компьютеру принадлежит только один процессор, но один и тот же процессор может быть установлен в разные компьютеры сети. Разрешения связи в данном случае не требуется. К сущности «Компьютер» следует добавить атрибут «Идентификатор процессора» в качестве внешнего ключа.



Рисунок 12 – Отношение «Компьютер-Процессор»

Связь в отношении «Компьютер-Видеоадаптер» заключается в том, что на одном компьютере может быть установлено несколько видеоадаптеров, а

один и тот же видеоадаптер может быть установлен в несколько компьютеров. Следует добавить промежуточную сущность, разрешающую связь «многие-ко-многим» (рисунок 14). Остальные отношения разрешаются идентичным образом.



Рисунок 13 – Отношение «Компьютер-Видеоадаптер»

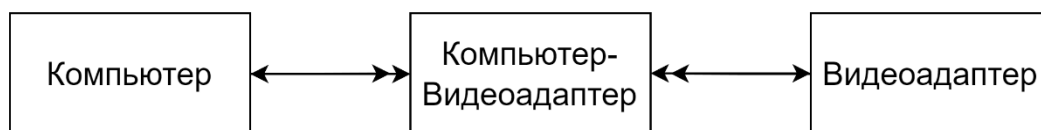


Рисунок 14 – Разрешение связи в отношении «Компьютер-Видеоадаптер»



Рисунок 15 – Отношение «Компьютер-ОЗУ»

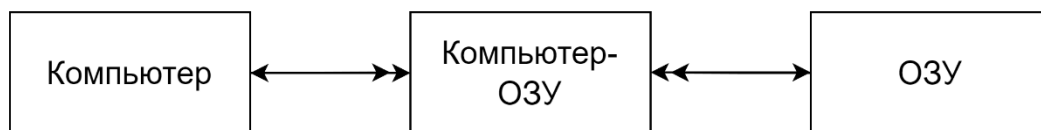


Рисунок 16 – Разрешение связи в отношении «Компьютер-ОЗУ»



Рисунок 17 – Отношение «Компьютер-Физический диск»

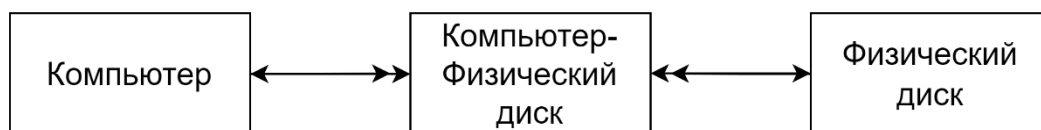


Рисунок 18 – Разрешение связи в отношении «Компьютер-Физический диск»



Рисунок 19 – Отношение «Компьютер-Логический диск»

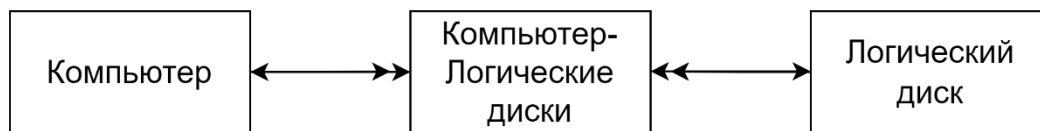


Рисунок 20 – Разрешение связи в отношении «Компьютер-Логический диск»



Рисунок 21 – Отношение «Компьютер-Операционная система»

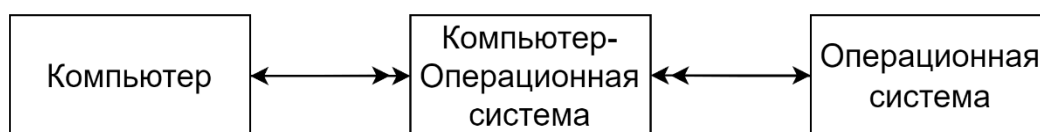


Рисунок 22 – Разрешение связи в отношении «Компьютер-Операционная система»



Рисунок 23 – Отношение «Компьютер-Приложение»

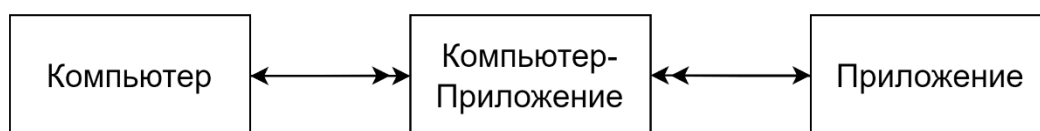


Рисунок 24 – Разрешение связи в отношении «Компьютер-Приложение»

2.5.3 Физическое проектирование

При проектировании физической структуры для обеспечения возможности версионирования во все связующие таблицы был добавлен атрибут «Номер версии».

Таблица 12 – Таблица «Компьютер»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Текстовый	–	System. StringBuilder	Primary key
Физический адрес	Текстовый	–	System. StringBuilder	
Идентификатор процессора	Числовой	–	UInt16	Foreign key

Таблица 13 – Таблица «Процессор»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Наименование	Текстовый	–	System. StringBuilder	
Количество ядер	Числовой	–	UInt16	
Частота	Числовой	–	UInt16	

Таблица 14 – Таблица «Видеоадаптер»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Наименование	Текстовый	–	System. StringBuilder	
Количество видеопамяти	Числовой	–	UInt16	

Таблица 15 – Таблица «ОЗУ»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Наименование	Текстовый	–	System. StringBuilder	
Частота работы	Числовой	–	UInt16	
Ёмкость	Числовой	–	UInt16	

Таблица 16 – Таблица «Физический диск»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Наименование	Текстовый	–	System. StringBuilder	
Общая ёмкость	Числовой	–	UInt16	
Статус	Текстовый	–	System. StringBuilder	

Таблица 17 – Таблица «Логический диск»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Буква диска	Текстовый	–	System. StringBuilder	
Общая ёмкость	Числовой	–	UInt16	
Свободное пространство	Числовой	–	UInt16	
Процент свободного пространства	Числовой	–	UInt16	

Таблица 18 – Таблица «Операционная система»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Наименование	Текстовый	–	System. StringBuilder	
Версия	Текстовый	–	System. StringBuilder	
Сборка	Текстовый	–	System. StringBuilder	
Разрядность	Текстовый	–	System. StringBuilder	

Таблица 19 – Таблица «Приложение»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Наименование	Текстовый	–	System. StringBuilder	
Поставщик	Текстовый	–	System. StringBuilder	
Версия	Текстовый	–	System. StringBuilder	

Таблица 20 – Таблица «Событие»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор	Числовой	–	UInt16	Primary key
Текст сообщения	Текстовый	–	System. StringBuilder	
Дата возникновения	Дата и время	–	DateTime	

Таблица 21 – Таблица «Компьютер-Видеоадаптер»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор компьютера	Текстовый	–	System. StringBuilder	Foreign key
Идентификатор видеоадаптера	Числовой	–	UInt16	Foreign key
Номер версии	Числовой	–	UInt16	

Таблица 22 – Таблица «Компьютер-ОЗУ»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор компьютера	Текстовый	–	System. StringBuilder	Foreign key
Идентификатор ОЗУ	Числовой	–	UInt16	Foreign key
Номер версии	Числовой	–	UInt16	

Таблица 23 – Таблица «Компьютер-Физический диск»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор компьютера	Текстовый	–	System. StringBuilder	Foreign key
Идентификатор физического диска	Числовой	–	UInt16	Foreign key
Номер версии	Числовой	–	UInt16	

Таблица 24 – Таблица «Компьютер-Логический диск»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор компьютера	Текстовый	–	System. StringBuilder	Foreign key
Идентификатор логического диска	Числовой	–	UInt16	Foreign key
Номер версии	Числовой	–	UInt16	

Таблица 25 – Таблица «Компьютер-Операционная система»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор компьютера	Текстовый	–	System. StringBuilder	Foreign key
Идентификатор операционной системы	Числовой	–	UInt16	Foreign key
Номер версии	Числовой	–	UInt16	

Таблица 26 – Таблица «Компьютер-Приложение»

Название атрибута	Тип данных	Условия	Формат данных	Индексация
Идентификатор компьютера	Текстовый	–	System. StringBuilder	Foreign key
Идентификатор приложения	Числовой	–	UInt16	Foreign key
Номер версии	Числовой	–	UInt16	

Такой выбор типов данных обусловлен выбранной СУБД. SQLite поддерживает встроенные в библиотеку .NET типы данных. Тип

«System.StringBuilder» является реализацией изменяемых строк, которые хранятся в стеке программы вместо управляемой кучи.

Физическая модель базы данных представлена в приложении А.

2.6 Проектирование пользовательского интерфейса

Главный интерфейс программы должен представлять собой форму с боковым меню, на котором находятся кнопки функциональных подсистем.

В программе должно быть три функциональные подсистемы:

- сканирование;
- мастер отчётов;
- настройки.

2.6.1 Подсистема «Сканирование»

В подсистеме «Сканирование» должны отображаться такие элементы, как:

- поле со списком компьютеров, обязательно с возможностью множественного выбора;
- поле с результатами сканирования;
- элементы управления, такие как кнопки, одна из которых должна отвечать за сканирование выбранных из списка компьютеров, другая – за сканирование всех компьютеров;
- графическая анимация, отображающая ход сканирования. Это нужно ввиду того, что все операции в программе выполняются асинхронно, не блокируя интерфейс, поэтому пользователь должен знать, что и в какой момент происходит.

Также расположение элементов должно быть интуитивно понятно пользователю программы.

В результате получился вариант интерфейса, изображенный на рисунке 25.

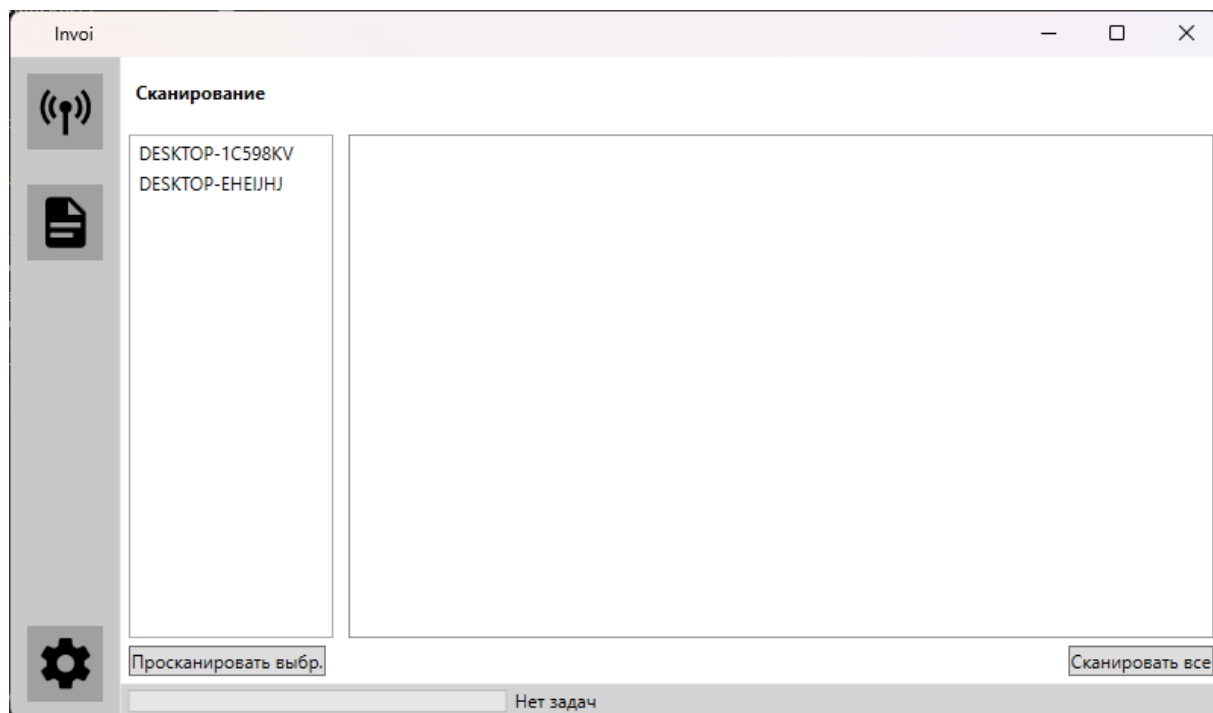


Рисунок 25 – Вариант интерфейса подсистемы «Сканирование»

2.6.2 Подсистема «Мастер отчётов»

В подсистеме «Мастер отчётов» пользователь должен иметь доступ к разным версиям состояния одного и того же компьютера. Возможный состав формы таков:

- список компьютеров;
- поле выбора версии выбранного компьютера;
- поле с содержанием отчёта.

Вариант интерфейса, проиллюстрированный на рисунке 26, демонстрирует возможный итоговый состав формы.

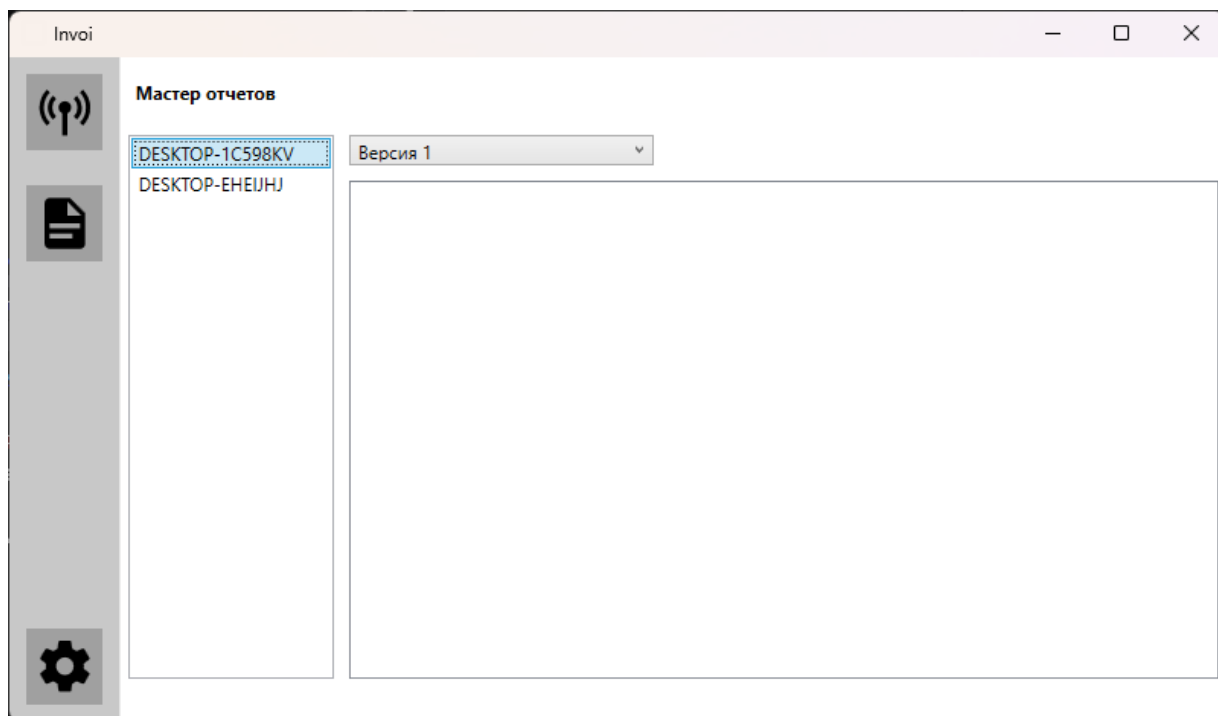


Рисунок 26 – Возможный состав подсистемы «Мастер отчётов»

2.6.3 Подсистема «Настройки»

В подсистеме «Настройки» пользователь должен видеть реализованные настройки программы, они должны быть структурированы и отделены по категориям. Возможный вариант интерфейса подсистемы «Настройки» изображён на рисунке 27.

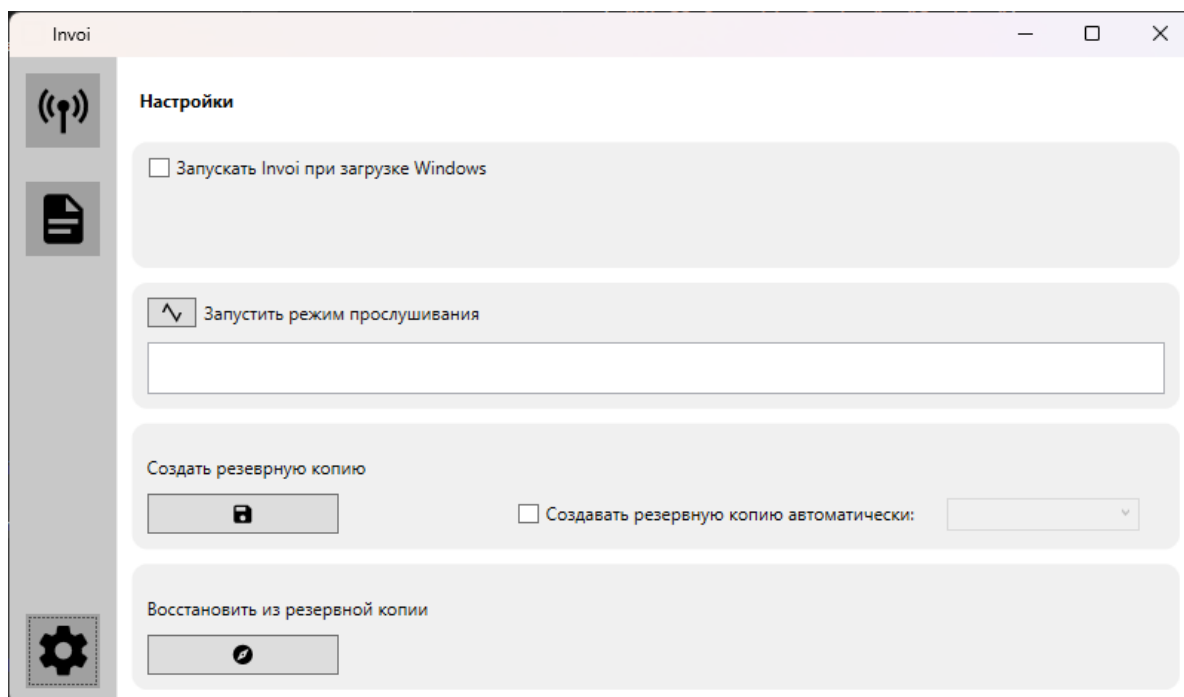


Рисунок 27 – Вариант внешнего вида подсистемы «Настройки»

2.7 Реализация ИС

2.7.1 Обеспечение возможности сканирования компьютеров

Для сканирования аппаратного и программного обеспечения компьютеров на Windows используется библиотека System.Windows.Management. Она является частью .NET Framework и предоставляет доступ к функциям Windows Management Instrumentation (WMI). WMI представляет собой мощную технологию для управления и мониторинга компонентов операционной системы Windows и других системных ресурсов.

Библиотека System.Management предоставляет классы и методы для работы с WMI, позволяя выполнять различные задачи, такие как получение информации о системе, управление процессами, реестром, службами, сетевыми настройками и другими компонентами операционной системы.

С помощью классов и методов из пространства имен System.Management можно выполнять запросы WMI, получать информацию о свойствах и методах объектов WMI, создавать, изменять или удалять объекты WMI, а также подписываться на события WMI.

Библиотека System.Management является важным инструментом при разработке системного программного обеспечения, инструментов администрирования, мониторинга и других приложений, требующих управления и мониторинга системы на уровне операционной системы Windows.

Запросы на WQL (WMI Query Language) не поддерживают ни одного вида соединения, кроме внутреннего, соответственно, оно задаётся оператором JOIN без явного указания типа соединения. С одной стороны, такое решение может запутать разработчиков, хорошо знакомых с SQL, но не имеющих опыта при работе с WQL. С другой стороны, доступ к изменению данных в таблицах WMI имеет только операционная система, что исключает случаи использования других видов соединения ввиду гарантированного

отсутствия дублей данных и записей с неверно заполненными полями.

Для получения запросов на компьютеры используется прослушивание портов. Для этого были использованы классы `TCPListener` и `Socket` – классы в языке программирования `C#`, которые используются для прослушивания портов и установления сетевых соединений.

Класс `TCPListener` предоставляет простой способ создания сервера, который прослушивает определенный порт и принимает входящие соединения от клиентов. `TCPListener` обеспечивает функциональность TCP-сервера и предоставляет методы для прослушивания и принятия подключений, а также для обработки полученных данных. Он может быть использован для создания сетевых приложений, которые взаимодействуют с клиентами по протоколу TCP.

Класс `Socket` является более низкоуровневым и предоставляет более гибкий и мощный способ работы с сетевыми соединениями. `Socket` позволяет установить соединение с удаленным хостом или прослушивать порты на локальной машине. Он предоставляет различные методы и свойства для управления соединениями, отправки и получения данных, установки параметров соединения и обработки ошибок. Класс `Socket` может быть использован для реализации как клиентской, так и серверной части сетевых приложений, и он обеспечивает более низкоуровневый доступ к сетевым возможностям операционной системы.

Использование `TCPListener` и `Socket` позволяет разработчикам создавать сетевые приложения, работающие по протоколу TCP, и управлять соединениями на уровне портов. Они предоставляют удобный интерфейс для прослушивания портов, принятия подключений и обмена данными с клиентами.

2.7.2 Обеспечение отзывчивой работы программы

Для работы с асинхронностью в языке `C#` есть операторы `async/await`, а также класс `Task`.

Класс Task в языке C# представляет асинхронную операцию, которая может выполняться в фоновом потоке или на пуле потоков. Он является частью пространства имен System.Threading.Tasks и предоставляет удобные методы и свойства для работы с асинхронными задачами. Вот некоторые основные аспекты класса Task:

- создание и запуск задачи. Класс Task предоставляет различные статические методы для создания и запуска задач. Например, метод Task.Run() позволяет создать и запустить задачу на выполнение. Можно также создать экземпляр класса Task, используя конструктор, и затем запустить его с помощью метода Start().

- асинхронное выполнение. Задачи позволяют выполнять асинхронные операции, которые не блокируют основной поток выполнения. Методы Task предоставляют возможность выполнять операции асинхронно и ожидать их завершения. Например, метод Task.Delay() позволяет задержать выполнение задачи на определенное время без блокировки потока.

- управление завершением задачи. Класс Task предоставляет методы и свойства для управления жизненным циклом задачи. Методы Wait() и WaitAll() позволяют ожидать завершения задачи или нескольких задач. Свойства IsCompleted, IsCanceled и IsFaulted позволяют проверить состояние задачи.

- обработка результатов. Задачи могут возвращать результаты своего выполнения. Для этого можно использовать свойство Result, которое содержит возвращаемое значение. Можно также использовать методы ContinueWith() или асинхронные ключевые слова async/await для обработки результатов задачи.

В разрабатываемом программном продукте экземпляры класса Task используются повсеместно в задачах, требующих время на выполнение, например, при ожидании пакетов от удалённых компьютеров, при сканировании компьютера, загрузке данных для отчётов.

Для немногочисленной анимации пользовательского интерфейса использовался класс `Thread`, позволяющий запустить выполнение кода в отдельном потоке. Именно его так называемым духовным наследником является класс `Task`.

2.7.3 Описание SQLite

SQLite является легковесной и встраиваемой реляционной базой данных, которая широко используется в различных приложениях, включая WPF (Windows Presentation Foundation) приложения. Некоторая информация об SQLite в контексте приложений WPF:

- возможности SQLite. SQLite предоставляет полноценный SQL-движок, который поддерживает большинство стандартных операций базы данных, таких как создание таблиц, выполнение запросов, индексирование, транзакции и т. д. Он также обеспечивает хранение данных в одном файле, что делает его удобным и компактным для встраивания в WPF-приложения;

- подключение к SQLite в WPF. Для работы с SQLite в приложении WPF, можно использовать ADO.NET-провайдер для SQLite, который предоставляет набор классов и методов для установки соединения с базой данных SQLite, выполнения запросов и получения результатов. Вы можете установить этот провайдер, добавив NuGet-пакет `System.Data.SQLite` в ваш проект;

- работа с базой данных. После подключения к базе данных SQLite, можно создать таблицы и определить структуру данных, используя SQL-запросы или ORM (Object-Relational Mapping) библиотеки, такие как `Entity Framework` или `SQLite-Net-PCL`. Также можно выполнять CRUD-операции (Create-Read-Update-Delete) записей в таблицах с помощью SQL-запросов или ORM-методов;

- управление соединениями и транзакциями. SQLite поддерживает многопоточное использование и множество соединений к базе данных. В приложении WPF вы можете управлять открытием и закрытием соединений с базой данных в соответствии с потребностями вашего приложения. Также

можно использовать транзакции для обеспечения целостности данных и атомарности операций записи/обновления/удаления;

– работа с данными в WPF-интерфейсе. В WPF есть возможность связывать данные из базы данных SQLite с элементами управления, такими как DataGrid, ListView или ComboBox, используя привязку данных (data binding). Это даёт возможность получать данные из базы данных и отображать их в пользовательском интерфейсе, а также обновлять данные и сохранять изменения обратно в базу данных;

– миграция базы данных. Если WPF-приложение требует изменений в структуре базы данных SQLite (например, добавление новых таблиц или полей), можно использовать механизм миграции базы данных, такой как SQLite-net-migrations, чтобы обновить существующую базу данных без потери данных.

В целом SQLite предоставляет удобный и эффективный способ хранения данных в приложениях WPF. Он подходит для малых и средних проектов, где требуется встроенная база данных с легким весом и хорошей производительностью.

2.7.4 Разработка пользовательского интерфейса

Для разработки пользовательского интерфейса был использован XAML.

XAML (Extensible Application Markup Language) является декларативным языком разметки, используемым в Windows Presentation Foundation (WPF) для описания пользовательского интерфейса (UI) в приложениях. Он предоставляет удобный способ создания графического интерфейса, разделения структуры и внешнего вида элементов интерфейса от логики приложения.

XAML позволяет описывать элементы пользовательского интерфейса, их свойства, стили, шаблоны, анимации и другие аспекты в виде иерархической структуры объектов, подобно XML. За счет этого разделение дизайна и логики становится более явным и гибким.

Основная идея XAML состоит в том, что разработчик может создавать и настраивать элементы интерфейса с помощью разметки, без необходимости программирования на языке C# или другом языке. XAML-разметка может быть связана с логикой приложения с помощью привязок данных и событий.

Применение XAML в WPF позволяет разработчикам более эффективно работать с интерфейсом приложения, улучшает читаемость кода и облегчает совместную работу между дизайнерами и разработчиками.

XAML реализует паттерн MVVM – паттерн проектирования, используемый в разработке пользовательского интерфейса, особенно в технологиях, таких как WPF и Xamarin. Он помогает разделить логику приложения от представления данных и обеспечивает хорошую организацию и поддержку кода.

Основные компоненты паттерна MVVM:

- модель (Model). Представляет бизнес-логику и данные приложения. Включает в себя классы и структуры данных, а также операции, связанные с обработкой данных;

- представление (View). Отображает данные модели пользователю. Это пользовательский интерфейс приложения, состоящий из элементов управления, макетов и графических компонентов;

- представитель (ViewModel). Связывает модель и представление. Он предоставляет данные и команды, которые необходимы для отображения и взаимодействия с моделью. Представитель отвечает за преобразование данных из модели в формат, понятный представлению, и обрабатывает команды пользователя, взаимодействуя с моделью.

Основные принципы MVVM:

- разделение ответственностей. Каждый компонент (Модель, Представление, Представитель) имеет свою специфическую роль и отвечает только за свои задачи;

- двусторонняя привязка данных. Представление и представитель

взаимодействуют с помощью двусторонней привязки данных, позволяющей автоматически обновлять данные в обоих направлениях;

– команды. Представитель предоставляет команды, которые могут быть вызваны из представления для выполнения определенных действий. Это позволяет обрабатывать пользовательский ввод и взаимодействовать с моделью.

2.7.5 Обеспечение информационной безопасности

Для защиты сетевого обмена данных используется протокол связи SSH для защищенной передачи пакетов между компьютерами, применяется шифрование данных для обеспечения конфиденциальности и целостности информации.

Для защиты базы данных необходимо обеспечить защиту файловой системы, в которой хранится база данных SQLite, чтобы предотвратить несанкционированный доступ к файлам. Применяется шифрование базы данных SQLite с помощью предоставляемых самой SQLite механизмов шифрования, таких как SQLCipher.

Для доступа к программе необходимо обладать правами администратора, так как используется контроль учётных записей.

Для логирования и мониторинга ведётся журнал событий и ошибок приложения, чтобы обнаружить и реагировать на потенциальные инциденты безопасности. Организован мониторинг приложения и базы данных для обнаружения подозрительной активности или аномалий.

2.7.6 Итоги реализации

На рисунках 28 и 29 показана работа программы в виде снимков экрана.

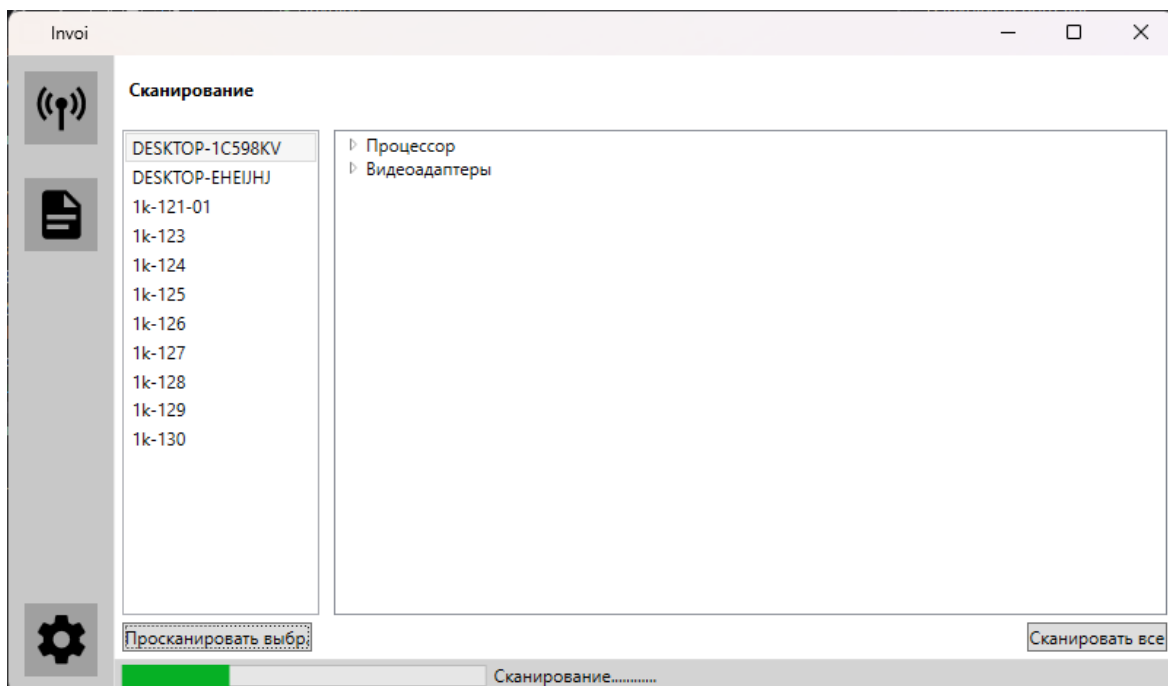


Рисунок 28 – Процесс сканирования выбранного компьютера

В процессе сканирования можно пользоваться другими функциями приложения: всё работает асинхронно и не блокирует интерфейс. Результат сканирования показан на рисунке 29.

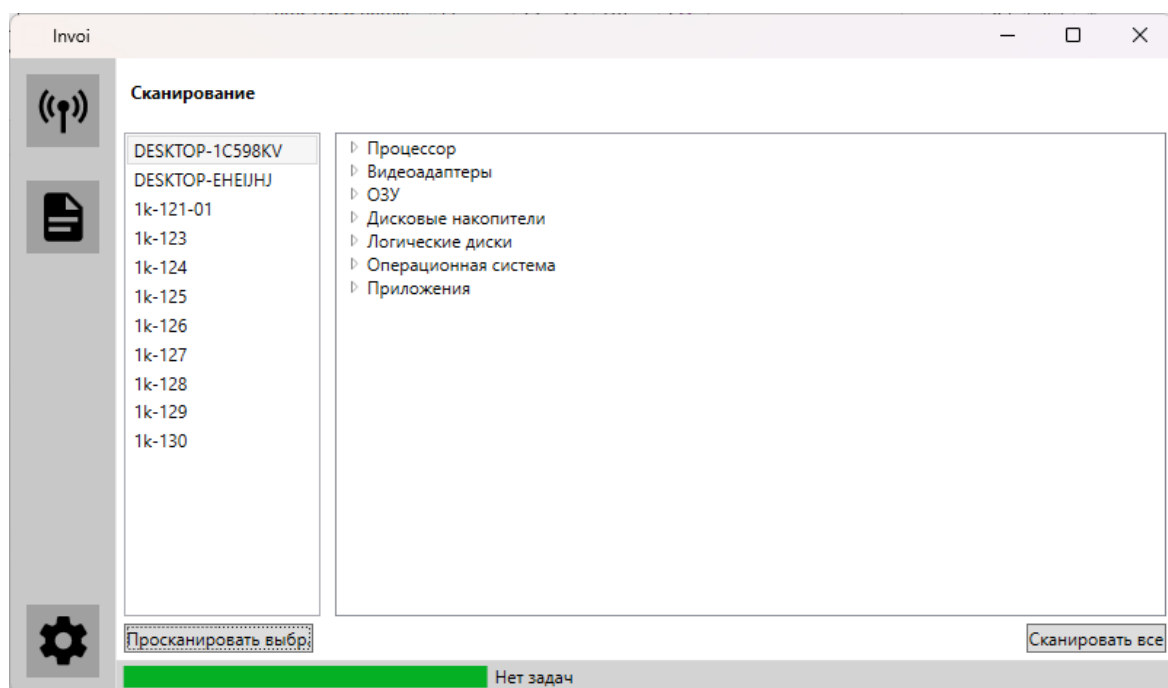


Рисунок 29 – Результат сканирования

В подсистеме «Отчеты» при открытии определённой версии компьютера мгновенно формируется отчёт. Это показано на рисунке 30.

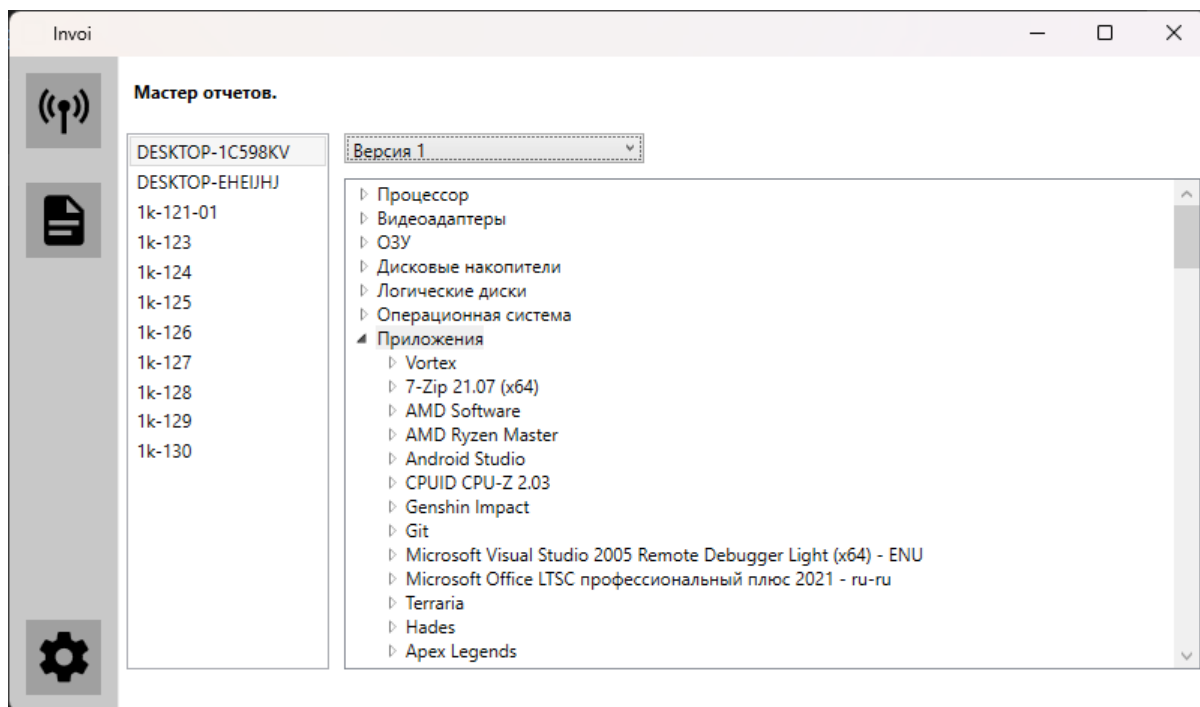


Рисунок 30 – Отчёт об одной из версий состояния компьютера

У получившейся ИС всё ещё есть направления развития. Например, полноценная поддержка Linux, сканирование по расписанию, улучшение пользовательского интерфейса, обеспечение сканирования периферийных устройств, доступных в данный момент.

3 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

Полноценное функционирование оконного приложения и технической поддержки системы подразумевает наличие рабочих мест и специально оборудованного помещения для его реализации. Поэтому необходимо организовать эти места в соответствии с нормативными документами и стандартами, а также обеспечить сохранение здоровья сотрудников при работе с приложением на компьютере.

3.1 Безопасность

3.1.1 Опасности и вред ПЭВМ на рабочем месте пользовател

В соответствии с ГОСТом 12.0.003-2015, работа на персональном компьютере сопряжена с потенциальными опасностями и неблагоприятными факторами для здоровья. Среди таких факторов можно выделить электростатические поля, риск поражения электрическим током, электромагнитное излучение, экстремальные температуры воздуха, монотонность рабочего процесса, высокий уровень шума, нехватка естественного освещения и другие.

Для предотвращения или снижения воздействия этих вредных факторов на пользователей ПЭВМ разработаны специальные требования, которые касаются организации рабочего места, освещения, уровня шума и общей организации помещений. Кроме того, также предоставляются рекомендации для пользователей ПЭВМ, чтобы обеспечить безопасные условия работы и снизить негативное воздействие на здоровье.

3.1.2 Организация рабочего места

Для оборудования рабочего места с компьютером (ПЭВМ) необходимо учесть следующие требования [1]:

– высота рабочей поверхности стола, используемого для работы с ПЭВМ, должна регулироваться в пределах (680–800) мм для взрослых пользователей. Если регулировка невозможна, высота должна составлять 725

мм;

– рабочий стол должен обеспечивать пространство для ног с высотой не менее 600 мм, шириной не менее 500 мм и глубиной не менее 450 мм на уровне колен, а на уровне вытянутых ног – не менее 650 мм;

– поверхность сиденья должна иметь ширину и глубину не менее 400 мм, закругленный передний край, регулироваться по высоте в пределах (400–550) мм, а также иметь возможность наклона вперед до 15 градусов и назад до 5 градусов. Угол наклона спинки в вертикальной плоскости должен быть ± 30 градусов;

– подлокотники должны иметь длину не менее 250 мм и ширину в пределах (50–70) мм, регулироваться по высоте над сиденьем в пределах 230 ± 30 мм и иметь внутреннее расстояние между ними от 350 до 500 мм;

– рабочее место пользователя ПЭВМ должно быть оборудовано подставкой для ног с шириной не менее 300 мм, глубиной не менее 400 мм, регулировкой по высоте в пределах 150 мм и возможностью наклона опорной поверхности до 20 градусов;

– клавиатура должна размещаться на поверхности стола на расстоянии 100 мм от пользователя или на специальной рабочей поверхности, отделенной от основного столешницы на расстоянии 300 мм.

Эти требования обеспечат комфортные и безопасные условия работы пользователей ПЭВМ.

На каждое рабочее место с компьютером (ПЭВМ), где отсутствует периферийное оборудование и установлен ЖК-монитор, требуется площадь 4,5 м². В противном случае, для каждого рабочего места с ПЭВМ необходима площадь 6 м². Кроме того, расстояние между боковыми стенками мониторов не должно быть менее 1,2 метра, а освещение от окон должно падать со стороны левой или правой стороны.

Для самостоятельной работы по обслуживанию сервера допускаются следующие условия:

- работники должны быть не моложе 18 лет;
- перед поступлением на работу необходим предварительный медицинский осмотр, чтобы определить соответствие состояния здоровья работника выполняемым задачам;
- в процессе трудовой деятельности должны проводиться периодические медицинские осмотры для наблюдения за состоянием здоровья работника и своевременного выявления ранних признаков воздействия вредных производственных факторов на здоровье работников;
- перед началом работы требуется вводный инструктаж;
- на рабочем месте проводится первичный инструктаж и стажировка в течение (2-14) смен под руководством назначенного лица;
- работники должны пройти обучение безопасным приемам и методам выполнения работ, а также оказанию первой помощи при несчастных случаях;
- работники должны пройти обучение и проверку знаний по электробезопасности, соответствующие второй группе.

Эти требования обеспечат безопасные условия работы с ПЭВМ и обслуживания сервера.

3.1.3 Освещение

Освещение является важным требованием для помещений с компьютерами (ПЭВМ). Правильное освещение способствует повышению производительности труда, уменьшает нагрузку на глаза. С другой стороны, плохое освещение может привести к быстрой утомляемости, снижению концентрации при работе за компьютером, ослеплению и раздражению от излишней яркости. Виды освещения включают естественное, искусственное, совмещенное и аварийное.

Естественное освещение обязательно должно присутствовать во всех помещениях, где работает персонал. Оно может быть боковым, верхним или комбинированным в зависимости от расположения.

Искусственное освещение применяется преимущественно в темное

время суток. Оно должно обеспечивать равномерное освещение всей рабочей области. Когда расположение источников света учитывает расположение рабочих мест, используется локализованное искусственное освещение.

Совмещенное освещение требуется, когда естественного освещения недостаточно. Этот тип освещения часто применяется в работах, требующих высокой точности.

Аварийное освещение используется при отключении основного освещения. В СанПиН 1.2.3685 - "Гигиенические нормы к требованиям к обеспечению безопасности и (или) безвредности для человека факторов среды обитания" четко определены требования к освещению на рабочих местах с ПЭВМ.

Согласно этим требованиям, равномерность освещенности должна быть не менее 0,6 в основных помещениях (0,7 для учебных кабинетов черчения и рисования, 0,5 на ледовых аренах и 0,7 для спортивных залов в физкультурно-оздоровительных организациях). Во вспомогательных помещениях минимальная равномерность освещенности составляет 0,4. Учебные помещения и читальные залы для обучающихся с нарушениями зрения оборудуются комбинированными системами общего искусственного и местного освещения.

Суммарный уровень освещенности от общего и местного освещения должен быть следующим: 1000 лк для обучающихся с высокой степенью осложненной близорукости и высокой степени дальнозоркости, (1000-1500) лк для обучающихся с поражением сетчатки и зрительного нерва без светобоязни, и не более 500 лк для обучающихся со светобоязнью.

Для обеспечения требуемого освещения рекомендуется использовать люминесцентные лампы с высокой световой отдачей и спектральным составом, близким к естественному. Освещенность на рабочей поверхности стола должна быть в диапазоне от 300 до 500 лк. Внутренняя отделка помещений с ПЭВМ должна использовать материалы с диффузно-

отражающими свойствами и коэффициентом отражения: потолок – (0,7-0,8), стены – (0,5-0,6), пол – (0,3-0,5).

3.1.4 Шум

Не менее важным опасным и вредным фактором при работе за ПЭВМ является повышенный уровень шума. В производственных помещениях при выполнении основных или вспомогательных работ с использованием ПЭВМ уровни шума на рабочих местах не должны превышать предельно допустимых значений, установленных для данных видов работ в соответствии с действующими санитарно-эпидемиологическими нормативами.

Шум на рабочих местах при использовании ПЭВМ, в помещениях жилых, общественных зданий и на территории жилой застройки нормируется. Уровень шума в помещении, в котором человек работает за ПЭВМ, не должен превышать 50 дБ.

3.1.5 Микроклимат

Микроклимат в производственных помещениях охватывает различные факторы, такие как температура, влажность, тепловое излучение и другие, которые влияют на комфорт и работоспособность человека. Сохранение подходящего микроклимата на рабочем месте в пределах норм является важной задачей по охране труда.

ПЭВМ на рабочих местах является значительным источником выделений, повышающих температуру человека и помещения в целом. Это может снижать работоспособность и производительность. Поддержание требуемой температуры в помещении обеспечивает безопасность и комфорт при работе с ПЭВМ.

Для регулирования микроклимата в помещении используются системы вентиляции, которые обеспечивают обмен воздуха и подачу свежего воздуха извне. Естественная вентиляция часто применяется для достижения наиболее комфортных условий, а весной и летом могут использоваться системы кондиционирования для полного контроля микроклиматических параметров и

создания комфортных условий труда.

Системы кондиционирования используются для поддержания постоянной температуры, влажности и очистки от вредных веществ в помещении. Они помогают решить проблему задержки углекислого газа в воздухе.

Система отопления поддерживает постоянную и равномерную температуру воздуха в рабочих помещениях в холодное время года. Она рассчитывается с учетом потерь тепла через стены здания и нагрева воздуха, проникающего в помещение. Существуют различные типы систем отопления, включая водяные, паровые, воздушные и комбинированные.

Системы водяного отопления наиболее эффективны и обеспечивают возможность регулировки температуры в широких пределах. Они широко применяются в помещениях с ПЭВМ. Рекомендуемые температурные диапазоны в холодное время года составляют (22–24) °С, а в теплое время года – (20–25) °С

3.1.6 Требования к графическому интерфейсу

При разработке системы были использованы компоненты, описанные в ГОСТ Р ИСО 9241-161-2016. С графическими элементами интерфейса пользователя можно работать посредством различных способов ввода, использующих:

- ввод данных с клавиатуры;
- указание с помощью компьютерной мыши, ручки, распознавания жестов,
- отслеживания положения глазного яблока;
- речевой ввод с использованием голосовых команд, распознавания голоса.

Если в интерактивной системе используют несколько способов ввода, у пользователя должна быть возможность применения всех таких способов. В данной работе используется классический ввод с использованием клавиатуры

и мышцы.

Оконное приложение разработано по классическим принципам разработки пользовательского интерфейса. Были использованные элементы и принципы взаимодействия с ними, описанные в ГОСТ Р ИСО 9241-161-2016. Все способы взаимодействия с системой соответствуют современным принципам пользовательского интерфейса, в основном используются классические и общедоступные средства ввода.

3.2 Экологичность

ПЭВМ содержит множество компонентов, которые содержат токсичные вещества, представляющие опасность для людей и окружающей среды. Эти вещества включают [2]:

- ртуть, содержащаяся в подсветке ЖК-мониторов, является вредным для мозга и нервной системы;
- щелочи, содержащиеся в щелочных аккумуляторах источников бесперебойного питания, могут прожигать слизистые оболочки и кожу;
- никель и цинк, находящиеся в материнских платах и батареях питания для ноутбуков, могут вызывать дерматит;
- поливинилхлорид, содержащийся в кабелях, используемых для подключения к электронным устройствам, может разрушать нервную систему и вызывать раковые заболевания.

Из-за этого ПЭВМ требует специального комплексного подхода к утилизации. Этот подход включает в себя сортировку металлических и неметаллических компонентов, при этом металлические части подлежат переплавке для последующего использования, а неметаллические компоненты компьютера утилизируются специальным способом.

В настоящее время в различных отраслях промышленности разрабатываются и внедряются малоотходные технологии. Однако полный переход ведущих отраслей промышленности на безотходную технологию требует решения сложных технологических, конструкторских и

организационных задач.

3.3 Чрезвычайные ситуации

3.3.1 Аварийные ситуации

Во время работы могут возникать следующие аварийные ситуации: обрыв проводов питания, проблемы с заземлением, повреждение электрооборудования, повреждение инженерных коммуникаций, а также повреждение конструктивных элементов здания или помещения.

В случае возникновения любой из описанных чрезвычайных ситуаций или при ухудшении самочувствия необходимо принять следующие меры:

- прекратить выполнение работ и, при необходимости, отключить электропитание;
- при наличии пострадавших предоставить первую помощь;
- обеспечить открытие аварийных выходов и эвакуацию персонала;
- доложить о принятых мерах руководителю работ и действовать в соответствии с его указаниями;
- сообщить оперативному дежурному о произошедшем.

Сотрудник, находящийся рядом с местом происшествия или несчастного случая, должен оказать предварительную медицинскую помощь и сообщить об этом оперативному дежурному или начальнику отдела. Если человек подвергся воздействию электрического тока, необходимо немедленно отключить источник электропитания и освободить его от контакта с током. Если невозможно отключить электроустановку, следует предпринять другие меры для освобождения пострадавшего, например, использовать изолирующие средства, такие как канат, палка или другой непроводящий электрический ток предмет. Также можно освободить пострадавшего, потянув за сухую одежду, избегая контакта с металлическими предметами или непокрытыми частями тела. Для защиты рук оказывающего помощь рекомендуется использовать изолирующие перчатки, а при их отсутствии можно обмотать руки сухой одеждой.

3.3.2 Меры пожарной безопасности на рабочих местах

Необходимо обеспечить свободный доступ к путям эвакуации и выходам при установке технологического и другого оборудования. При установке персонального компьютера и монитора необходимо выбрать надежную опору (тумбочку, подставку, кронштейн и т.д.), которая предотвратит их падение. Следует избегать установки ПЭВМ:

- в нишах мебельных "стенок", тумбочках и т.п.;
- менее чем в 1 метре от электронагревательных приборов и горючих предметов, таких как тюли, занавески, гардины, шторы, декоративные украшения, новогодние ёлки и т.д.;
- менее чем в 0,7 метрах от проходов, путей передвижения и эвакуации людей.

Перед началом эксплуатации персонального компьютера необходимо выполнить следующие шаги:

- проверить место установки компьютера и монитора на соответствие требованиям безопасности, описанным выше;
- осмотреть компьютер, монитор, сетевой кабель и вилку на наличие повреждений; при повреждении корпуса, сетевого кабеля, вилки или задней крышки запрещается использовать компьютер;
- удалить горючие предметы (салфетки, накидки, книги, газеты, декоративные украшения и т.д.) и емкости с жидкостью (вазы с живыми цветами) с и около компьютера и монитора;
- убедиться, что вентиляционные отверстия на задней крышке компьютера и монитора не заблокированы предметами;
- убедиться в наличии огнетушителя или противопожарной ткани рядом с компьютером. Соблюдение этих мер безопасности при работе с ПЭВМ поможет снизить риск возникновения пожара.

ЗАКЛЮЧЕНИЕ

В ходе проведения данного исследования и разработки решения для инвентаризации компьютерных сетей были выполнены ряд задач, решены трудности и достигнуты следующие результаты. Была определена проблема отсутствия эффективного механизма инвентаризации, а также разработан программный продукт, который успешно решает данную проблему.

Поставленные задачи по разработке инструмента инвентаризации сети были выполнены в полной мере. Созданный программный продукт позволяет эффективно и точно определять состояние компьютерных ресурсов в сети, предоставляя ценную информацию о них. Это позволяет упростить процесс управления и обслуживания компьютерной инфраструктуры организации.

Внедрение разработанного программного продукта требует проведения соответствующих мероприятий, таких как установка и настройка, обучение персонала и тестирование. Однако, его внедрение обещает значительные преимущества. Он позволит повысить эффективность работы персонала, сократить время и ресурсы, затрачиваемые на инвентаризацию сети, а также обеспечить более надежное и актуальное управление компьютерными ресурсами.

Таким образом, результаты данной работы являются важным вкладом в область управления и обслуживания компьютерных сетей. Разработанный программный продукт успешно решает проблему инвентаризации, предоставляя полезную информацию и улучшая процессы управления компьютерной инфраструктурой.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1 Булгаков А.Б. Безопасность жизнедеятельности [Электронный ресурс]: сб. учеб.-метод. материалов для всех направлений подготовки бакалавров и специалистов / АмГУ, ИФФ; сост. А.Б. Булгаков, В.Н. Аверьянов, М. В. Гриценко. – Благовещенск: Изд-во Амур. гос. ун-та, 2017.
http://irbis.amursu.ru/DigitalLibrary/AmurSU_Edition/9036.pdf

2 Кардаш, Т. А. Эргономика рабочих мест служащих и инженерно-технических работников, оснащенных ПЭВМ [Текст]: учеб. пособие / Т. А. Кардаш; АмГУ, ИФФ. - Благовещенск: Изд-во Амур. гос. ун-та, 2002. - 60 с.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Албахари Дж., Албахари Б. С# 7.0 в подлиннике. – Санкт-Петербург: Питер, 2017. – С 44–55.
- 2 Алексеев С. С. Современное программирование на С# и .NET: Учебное пособие. – Москва: Издательский дом "Питер", 2018. – С 61–66.
- 3 Ашитков В. А., Тимофеев А. В., Мамедов А. Ф. SQLite. Карманный справочник. – Москва: Питер, 2018. – С 123–128
- 4 Булгаков А.Б. Безопасность жизнедеятельности [Электронный ресурс]: сб. учеб.-метод. материалов для всех направлений подготовки бакалавров и специалистов / АмГУ, ИФФ; сост. А.Б. Булгаков, В.Н. Аверьянов, М. В. Гриценко. – Благовещенск: Изд-во Амур. гос. ун-та, 2017. http://irbis.amursu.ru/DigitalLibrary/AmurSU_Edition/9036.pdf
- 5 Безопасность жизнедеятельности в химической промышленности [Электронный ресурс]: учебник / Н. И. Акинин [и др.]. – Санкт-Петербург: Лань, 2019. – Режим доступа: <https://e.lanbook.com/book/116363>.
- 6 Болл Р. SQLite. Разработка приложений для Android и iOS. – Санкт-Петербург: БХВ-Петербург, 2018.
- 7 Дейтел П., Дейтел Х. Язык программирования С#. Технологии .NET и платформы Microsoft. – Москва: ООО "Издательство ДМК Пресс", 2019.
- 8 Инструментарий управления Windows (WMI) // Microsoft.Документация. URL: <https://learn.microsoft.com/ru-ru/windows/win32/wmisdk/wmi-start-page> (дата обращения: 17.02.2023)
- 9 Кардаш, Т. А. Эргономика рабочих мест служащих и инженерно-технических работников, оснащенных ПЭВМ [Текст]: учеб. пособие / Т. А. Кардаш; АмГУ, ИФФ. - Благовещенск: Изд-во Амур. гос. ун-та, 2002. - 60 с.
- 10 Ключников Ю. А. Базы данных SQLite и С#. – Москва: Издательский дом "Питер", 2018.
- 11 Крибс Д. Практика применения WPF и MVVM на С#. – Санкт-

Петербург: БХВ-Петербург, 2018.

12 Обзор Total Network Inventory – утилиты для автоматической инвентаризации парка компьютеров // iXBT.com. [Москва]. URL: <https://www.ixbt.com/soft/total-network-inventory.shtml> (дата обращения: 15.02.2023).

13 Рихтер Джеффри. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. – Москва: ООО "Издательство ДМК Пресс", 2018.

14 Свод правил СП 1.13130.2020 Система противопожарной защиты. Эвакуационные пути и выходы. Введ. 2020-09-19

15 Семенихин А. В. Разработка приложений на платформе .NET с использованием WPF и ADO.NET. – Москва: ООО "Издательство ДМК Пресс", 2018.

16 Скит Дж. C# 7.0 в глубину. – Москва: Издательство "ДМК Пресс", 2019.

17 Троелсен Э., Япикс П. C# 7.0 и платформа .NET Core 2. – Москва: ДМК Пресс, 2018.

18 Фримен Э., Робинс Дж. WPF. Практическое руководство разработчика. – Санкт-Петербург: Питер, 2018.

19 Хортон И. Программирование на языке C# 7 и .NET Core 2 для профессионалов. – Санкт-Петербург: Питер, 2018.

20 Шилдт Г. C# 7.0. Полное руководство. – Санкт-Петербург: Питер, 2018.

21 About OCS inventory // OCS. URL: <https://ocsinventory-ng.org/?lang=en> (дата обращения: 11.01.2023).

22 Inventory your network devices // Spiceworks. URL: <https://www.spiceworks.com/free-pc-network-inventory-software/> (дата обращения: 20.02.2023).

23 IT Heroes Trust Lansweeper for Complete Visibility // Lansweeper. URL:

<https://www.lansweeper.com/> (дата обращения: 10.02.2023).

24 Open-Audit – Documentation // Opmantek. URL: <https://community.opmantek.com/display/OA/Home> (дата обращения: 11.02.2023).

25 What is Network Inventory // Network Inventory Advisor. URL: <https://www.network-inventory-advisor.com/more-about-network-inventory.html> (дата обращения: 13.01.2023).

26 10 лучших программ для инвентаризации сети 2023 // SoftinventiveLab. [Москва]. URL: <https://www.softinventive.ru/best-network-inventory-tools> (дата обращения: 20.02.2023)

27 10-Страйк Инвентаризация Компьютеров – Программа для сканирования компьютеров по сети // 10-Strike Software. [Ульяновск]. URL: <https://www.10-strike.ru/networkinventoryexplorer/> (дата обращения: 10.01.2023).

ПРИЛОЖЕНИЕ А

