МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

Факультет	*Кафедра*
Вычислительной техники	МОиПЭВМ

Направление подготовки 09.03.02 «Информационные системы и технологии»

БАКАЛАВРСКАЯ РАБОТА

на тему

Система подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности

Студент	_____ (подпись, дата)	Краснов Сергей Сергеевич (ФИО полностью)
Руководитель	_____ (подпись, дата)	Попова Н.А. (фамилия, инициалы)
Нормоконтролёр	_____ (подпись, дата)	Такташкин Д.В. (фамилия, инициалы)

Работа допущена к защите (протокол заседания кафедры от _____№_____)

Заведующий кафедрой	_____ (подпись)	Козлов А.Ю. (фамилия, инициалы)

Работа защищена с отметкой _____ (протокол заседания ГЭК от _____№_____)

Секретарь ГЭК	_____ (подпись)	Попова Н.А. (фамилия, инициалы)

Пенза, 2023

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ

Факультет Вычислительной техники

Кафедра МОиПЭВМ

«Утверждаю»
Заведующий кафедрой
_____ А.Ю. Козлов
«___» _____ 2023 г

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
БАКАЛАВРА

1. Студент Краснов Сергей Сергеевич гр. 19ВИ1 факультета ВТ
<Фамилия, имя, отчество полностью>

2. Тема работы Система подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности

Тема утверждена приказом ПГУ №_____ от "_____"_____20_____

3. Руководитель работы к.т.н., доцент кафедры «МО и ПЭВМ»

4. Задание на работу (назначение разработки, исходные данные и т.п.)

Назначение: оптимизация процессов на предприятии, связанных с подбором комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности

Основные функции системы:

1. Ведение нормативно-справочной информации о номенклатуре и сотрудниках;
2. Формирование рабочего места по заданным параметрам;
3. Учет поступлений техники на склад;
4. Учет введенной в работу техники;
5. Учет передачи техники в ремонт и приема из него;
6. Учет списанной техники;
7. Формирование отчетов.

Технология разработки: предметно-ориентированное программирование.
Язык программирования: встроенный язык программирования 1С: Предприятие.
Среда программирования: технологическая платформа 1С: Предприятие 8.3.

5. Перечень подлежащих разработке вопросов:

1. анализ предметной области и постановка задачи;
2. проектирование программных средств;
3. реализация программных средств;
4. тестирование программных средств;
5. оформление ВКР.

6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1	Анализ предметной области и постановка задачи	12%	12.04.2023 – 17.04.2023	
2	Проектирование программных средств	22%	18.04.2023 – 28.04.2023	
3	Реализация программных средств	30%	29.04.2023 – 13.05.2023	
4	Тестирование программных средств	10%	14.05.2023 – 18.05.2023	
5	Оформление ВКР	26%	19.05.2023 – 31.05.2023	

Дата выдачи задания "_____"_____20_____

Руководитель бакалаврской работы _____
(подпись, дата)

Задание к исполнению принял студент _____
(подпись, дата)

Работу к защите допустить _____
Декан факультета (подпись, дата)

Реферат

Пояснительная записка содержит 101 лист, 43 рисунка, 13 таблиц, 3 листинга, 28 использованных источников и 8 приложений.

1С, КОРПОРАТИВНЫЕ СИСТЕМЫ, 1С ПРЕДПРИЯТИЕ 8.3, ОПЕРАТИВНЫЙ УЧЕТ, ИНФОРМАЦИОННЫЕ СИСТЕМЫ, КОМЛЕКТАЦИЯ РАБОЧЕГО МЕСТА, ПК, КОНФИГУРАТОР ПК, УЧЕТ ТЕХНИКИ, ДИАГРАММЫ UML.

Предмет работы: разработка программного обеспечения подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Объект работы: оптимизация подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Цель работы: разработка программного обеспечение подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Технология разработки: платформа 1С: Предприятие 8.3.

Результаты работы: спроектирована и разработана 1С конфигурация подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Область применения: предприятия, использующие на рабочих местах компьютеры.

Содержание

Введение 7
1 Постановка задачи и анализ аналогов разрабатываемой системы 9
1.1 Постановка задачи 9
1.2 Анализ программ аналогов 10
1.3 Анализ функциональных требований 12
1.4 Обоснование выбора средств проектирования и разработки 15
2 Планирование и расчет бюджета разрабатываемой системы 17
3 Проектирование информационной системы 20
3.1 Проектирование с помощью диаграмм UML 20
3.1.1 Диаграмма деятельности 20
3.1.2 Диаграмма кооперации 22
3.1.3 Диаграмма последовательности 23
3.2 Проектирование логической модели базы данных 24
3.3 Проектирование пользовательского интерфейса 27
4 Реализация информационной системы 30
4.1 Реализация структур программного средства 30
4.2 Реализация кода системы 48
5 Тестирование документов конфигурации 51
5.1 Тестирование методом черного ящика 51
5.1.1 Классы эквивалентности 51
5.1.2 Функциональная диаграмма и таблица решений 52
5.2 Тестирование метрики кода 54
5.3 Автоматизированное тестирование 57
Заключение 60
Список использованных источников 61
Приложение А. Диаграмма вариантов использования 64
Приложение Б. Диаграмма деятельности 66
Приложение В. Диаграмма кооперации 68
Приложение Г. Диаграмма последовательности 70

Приложение Д. Логическая модель базы данных 72
Приложение Е. Диаграмма компонентов 74
Приложение Ж. Листинг конфигурации 76
Приложение И. Результаты тестирования 98

Введение

В современном мире персональные компьютеры или же ПК используются повсеместно как для личного пользования, так и для рабочих задач. Компьютеры позволяют облегчить документооборот на предприятиях, что повышает эффективность и скорость обработки документов, уменьшает затраты на бумагу и упрощает процесс поиска и хранения файлов. Помимо этого, использование ПК сокращает количество ошибок при взаимодействии с документами и оптимизирует взаимодействие сотрудников между собой.

Из-за большого числа рабочих мест с компьютерами, могут возникнуть трудности с подбором подходящего оборудования на каждое место. Одному работнику необходим принтер, другому гарнитура для связи, третьему производительный компьютер для работы с тяжелыми задачами. Поиск техники на складе может занимать некоторое время, да и техники как таковой может вообще не быть в наличии.

Поэтому на предприятиях для ускорения и облегчения процессов следует использовать специальное приложение, позволяющее оптимизировать подбор комплектации рабочих мест пользователей ПК.

Объектом выпускной квалификационной работы является оптимизация подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Предметом является программное обеспечение подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Цель данной работы заключается в разработке программного обеспечение подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

Для достижения поставленной цели были выделены следующие задачи:

– анализ предметной области и постановка задачи на разработку системы;

– определение требований к разрабатываемой системе;

– проектирование системы подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности;

– реализация системы;

– тестирование разработанной системы.

В результате выполнения работы была разработана конфигурация, позволяющая оптимизировать подбор комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности.

1 Постановка задачи и анализ аналогов разрабатываемой системы

1.1 Постановка задачи

Приложение для комплектации рабочего места в виде самостоятельного программного средства не является удобным решением. На предприятиях, скорее всего уже используется специализированное приложение, позволяющее осуществлять хозяйственный учет. Хозяйственный учет позволяет изучать, управлять, а также оценивать деятельность организации. Данный вид учета по своему составу делится на три вида:

– оперативный учет – позволяет отслеживать информацию о движении материальных средств на предприятии;

– статистический учет – позволяет собирать статистику о деятельности организации и на основе этого принимать различные решения;

– бухгалтерский учет – отражает финансовую и хозяйственную деятельность предприятия [1].

Комплектация рабочего места относится к оперативному учету, так как отражает перемещение оборудования на рабочих местах. Поэтому наиболее удачным решением является использование системы комплектации места в составе программы ведения оперативного учета. Это позволяет использовать базу данных предприятия, что ещё больше облегчает пользование системой.

Таким образом, разрабатываемая система подбора рабочего места должна являться частью программы для управления бизнес-процессами на предприятии. Необходимо разработать приложение с возможностью выполнения различных операций складского учета. При этом у пользователей должна быть возможность добавления характеристик для номенклатуры. Нужно предусмотреть возможность поступления и списания техники, а также отправки её в ремонт и обратно. Также необходимо реализовать передачу техники между рабочими местами. В дополнение к этому, было бы полезно

сделать различные отчеты для просмотра остатков номенклатуры на складе, в ремонте и т. д.

Помимо этого, необходимо реализовать возможность подбора номенклатуры на рабочее место по заданным параметрам пользователя. При этом необходимо учитывать наличие номенклатуры на складе, чтобы добавлялась только имеющаяся техника. Дополнительно следует создать конфигуратор ПК, где продвинутые пользователи смогут создавать собственные конфигурации компьютеров, собирая их из отдельных комплектующих.

1.2 Анализ программ аналогов

Перед разработкой системы необходимо найти и проанализировать уже существующие аналоги, чтобы учесть их достоинства и недостатки, а также понять, что может быть ещё необходимо пользователю такого продукта.

1С: Управление нашей фирмой. Решение имеет большой функционал, позволяющий заниматься бизнесом в одном приложении. Программа позволяет вести: клиентскую базу, банковские и кассовые операции, учет номенклатуры, покупок и продаж, учет доходов, расходов и многое другое. Таким образом, 1С: Управление нашей фирмой является универсальным решением, предоставляющее все необходимые функции для ведения учета и управления в малом бизнесе. Однако системы подбора оптимальной комплектации рабочего места в ней нет.

Стоимость данного решения начинается от 6000 рублей за базовую версию на одно рабочее место. За наиболее полную версию придётся заплатить 37600 рублей за пять рабочих мест с возможность дополнения. Помимо этого есть и облачная версия программы, которая обойдётся в 1490 рублей в месяц [2].

Другим похожим программным средством является МойСклад. Программа позволяет автоматизировать складской учет, обработку заказов и продаж, оформлять сборку и производство продукции, а также работать с

клиентами и финансами. Одной из особенностей приложения является то, что программа доступна с любого устройства, что дает возможность вести работу с любой точки планеты.

Цена за лицензию МойСклад варьируется от 495 до 6715 рублей в месяц в зависимости от выбранного тарифного плана, а также есть бесплатный тариф, но со значительными ограничениями [3].

Ещё одним вариантом является система БИТ.WMS. Она позволяет оптимизировать процессы на складе и решать проблемы, возникающие при его работе. Программа дает возможность контролировать прием, перемещение, хранение и отгрузку товаров на складе, проводить инвентаризацию, а также строить различные отчеты. Стоимость лицензии на данный продукт довольно высока и составляет 250000 рублей [4].

Рассмотрев схожие по функционалу программы, были выделены критерии для их сравнения:

– управление складом – показатель функционала для ведения складского учета и операций;

– управление персоналом – показатель функционала для работы с персоналом;

– работа с клиентами – показатель функционала для работы с клиентами;

– формирование отчетов – показатель функционала для формирования отчетов;

– стоимость – показатель стоимости продукта (чем меньше, тем дешевле).

Для оценки критериев используются цифры от 1 до 3, где 1 – низкое проявление критерия, а 3 – высокое. Результат оценки приведен в таблице 1.

Таблица 1 – Критериальная оценка программ аналогов

	1С: Управление нашей фирмой	МойСклад	БИТ.WMS
Управление складом	1	2	3
Управление персоналом	3	1	2
Работа с клиентами	2	3	1
Формирование отчетов	3	2	3
Стоимость	2	1	3

По результатам сравнения программ аналогов будут определены функциональные требования для разрабатываемой системы. Помимо этого было выявлено, что ни в одной из аналогичных систем нет возможности комплектации рабочего места сотрудников исходя из их профессиональной деятельности. Поэтому разработка такой системы позволит оптимизировать процессы управления на предприятии.

1.3 Анализ функциональных требований

На основе поставленной задачи и сравнения аналогов были определены основные требования к разрабатываемой программе. Система подбора комплектации рабочего места должна иметь следующие основные функции:

– хранение справочной информации о номенклатуре и её характеристиках, сотрудниках и фирмах ремонта;

– учет поступления и списания номенклатуры;

– введение в эксплуатацию, перемещение между рабочими местами, отправка в ремонт и прием из него номенклатуры;

– комплектация рабочего места по заданным параметрам;

– возможность создания собственной конфигурации ПК;

– контроль остатков номенклатуры;

– формирование отчетов.

Для визуализации всех основных функции системы была разработана диаграмма вариантов использования, представленная в приложении А.

Диаграмма представляет собой множество актеров, которые взаимодействуют с системой с помощью вариантов использования. Актером

является любая сущность, воздействующая на систему. Это может быть как человек, так и какое-либо техническое оборудование. Вариантом использования называется действие, выполняемое актером. Элементы диаграммы связаны друг с другом при помощи отношений. Существуют следующие виды отношений:

– ассоциация – устанавливает связь между актером и вариантом использования;

– расширение – позволяет присоединять дополнительное поведение одного варианта использования к другому;

– включение – указывает, что вариант использования является составным компонентом в поведении другого;

– обобщение – позволяет задать наследование между вариантами использования [5].

Далее описаны элементы, изображенные на диаграмме вариантов использования.

Актером является сотрудник предприятия.

Вариант использования «Получить технику по накладной» заключается в создании документа «Приходная накладная», где указывается список техники, которая будет доставлена на склад. Список представляет собой таблицу с указанием названия техники, её характеристик и количества. При необходимости, сотрудник может добавить новую технику в базу данных, чтобы использовать её впоследствии.

Вариант использования «Создать заявку на поступление техники» заключается в создании документа «Заявки для заказа», где указывается список техники, которая будет доставлена на склад. Заявку можно создать, если при вводе техники в работу необходимой техники не будет в наличии.

Вариант использования «Заполнить основные поля» заключается в выборе техники, её характеристик, рабочего места и ввода количества.

Вариант использования «Ввести технику в работу» заключается в создании документа «Ввод техники в работу», который позволяет назначить

технику на определенное рабочее место. Для создания документа необходимо заполнить его основные поля (данный вариант использования был описан ранее). Также есть возможность подобрать рабочее место или собрать собственную конфигурацию ПК для дальнейшего использования.

Вариант использования «Списать технику» заключается в создании документа «Списание техники», который позволяет снять технику с эксплуатации. Для создания документа необходимо заполнить его основные поля (данный вариант использования был описан ранее).

Вариант использования «Передать технику в ремонт» заключается в создании документа «Передача в ремонт», который позволяет отправить неисправную технику в ремонт. Для создания документа необходимо заполнить его основные поля (данный вариант использования был описан ранее), а также выбрать фирму исполнителя и дату окончания ремонта.

Вариант использования «Получить технику из ремонта» заключается в создании документа «Приход с ремонта», который позволяет получить отремонтированную технику из ремонта. Для создания документа необходимо заполнить его основные поля (данный вариант использования был описан ранее).

Вариант использования «Переместить технику» заключается в создании документа «Передача техники», который позволяет переместить технику из одного рабочего места на другое. Для создания документа необходимо заполнить его основные поля (данный вариант использования был описан ранее), а также указать новое рабочее место.

Вариант использования «Сформировать отчет» заключается в формировании различных отчетов:

– отчет запасов техники, отражающий количество техники на складе на указанную дату;

– отчет с описанием техники, отражающий список характеристик выбранной техники;

– отчет с техникой на местах, отражающий на каком рабочем месте располагается техника;

– сводный отчет, отражающий количество техники на складе, в работе, в ремонте, а также количество списанной техники.

Таким образом, составленная диаграмма вариантов использования отображает весь функционал системы и определяет требования к программе.

1.4 Обоснование выбора средств проектирования и разработки

Для проектирования UML-диаграмм системы подбора комплектации рабочего места будет использоваться такое средство как Visual Paradigm, так оно позволяет строить множество диаграмм UML и предоставляет бесплатный пробный период, которого достаточно для проектирования приложения.

Visual Paradigm – это инструмент проектирования и управления для разработки IT систем. Программное средство позволяет строить различные UML диаграммы, используя каталог ресурсов, благодаря чему повышается эффективность и качество диаграмм. Помимо этого Visual Paradigm предоставляет возможность повторно использовать элементы в разных моделях, проверять синтаксис и устанавливать связи между элементами модели и внешними ресурсами [6].

Перед выбором средства разработки необходимо определиться с методологией разработки. Существует две основных модели разработки программного обеспечения: каскадная и итерационная.

Каскадная модель предполагает последовательное выполнение этапов разработки, благодаря чему возможно точно посчитать сроки и стоимость выполнения проекта. Однако это приводит к невозможности внесения изменений, так как проверить продукт можно будет только на этапе тестирования.

Итерационная модель предполагает разбиение всей разработки на несколько циклов. Сначала выполняется разработка рабочей версии программы, которая включает в себя только базовый функционал. На

следующих этапах происходит добавление новых функций и уточнение требований. В совокупности после выполнения всех циклов и получается готовый продукт.

Помимо этого, существует методология быстрой разработки приложений RAD (Rapid Application Development). Модель предполагает использование инструментальных средств визуального моделирования и разработки. За счет этого в кратчайшие сроки стоится прототип, который обсуждается с заказчиком и после этого вводятся необходимые изменения [7].

Благодаря высокой скорости разработки и её дешевизны именно RAD методология и будет использоваться для разработки программы данной работы. Одной из программ для быстрой разработки является платформа 1С: Предприятие 8.3. Она имеет бесплатную учебную версию и предоставляет большой функционал для разработки конфигураций. Платформа предназначена для автоматизации учета и управления бизнес-процессами на предприятии, позволяет вести учет финансовой деятельности, налоговый учет, управлять складскими запасами и т. п. При этом программа является low-code системой, что позволяет создавать приложения без глубоких знаний в области программирования.

Таким образом, в качестве средства разработки будет использоваться платформа 1С: Предприятие 8.3.

2 Планирование и расчет бюджета разрабатываемой системы

Планирование бюджета заключается в определении необходимых финансовых и временных ресурсов, которые будут потрачены на разработку программного средства. Правильное распределение средств и времени обеспечивает успешное завершение проекта в рамках установленных сроков и бюджета, сокращает риски и позволяет избежать проблем, связанных с нехваткой каких-либо ресурсов. Для планирования процесса разработки и расчета бюджета можно использовать диаграмму Ганта.

Диаграммы Ганта – это инструмент управления проектами, позволяющий визуализировать выполнение работы с привязкой ко времени. Диаграмма отображает основные задачи проекта, а также позволяет разделить их на задачи меньшего размера, что облегчает процесс управления [8].

Для построения диаграммы необходимо выделить задачи, а также сроки их выполнения. Задачи проекта представлены в таблице 2.

Таблица 2 – Задачи проекта

Название задачи	Длительность задачи в днях	Дата начала	Дата окончания
Начало проекта	0	12.04.2023	12.04.2023
Постановка задачи и анализ аналогов	5	12.04.2023	17.04.2023
Постановка задачи	1	12.04.2023	13.04.2023
Анализ программ аналогов	1	14.04.2023	15.04.2023
Выбор технологий проектирования и разработки	1	16.04.2023	17.04.2023
Проектирование конфигурации	10	18.04.2023	28.04.2023
Проектирование логической модели БД	2	18.04.2023	20.04.2023
Проектирование диаграмм UML	4	21.04.2023	25.04.2023
Проектирование пользовательского интерфейса	2	26.04.2023	28.04.2023
Реализация конфигурации	14	29.04.2023	13.05.2023
Кодирование	11	29.04.2023	10.05.2023
Реализация пользовательского интерфейса	2	11.05.2023	13.05.2023
Тестирование ИС	4	14.05.2023	18.05.2023
Тестирование методом черного ящика	1	14.05.2023	15.05.2023
Тестирование метрикой Холстеда	1	15.05.2023	16.05.2023
Автоматизированное тестирование	2	16.05.2023	18.05.2023
Подготовка документации	11	19.05.2023	30.05.2023
Написание документации	6	19.05.2023	25.05.2023

Продолжение таблицы 2

Название задачи	Длительность задачи в днях	Дата начала	Дата окончания
Оформление и печать документации	4	26.05.2023	30.05.2023
Конец проекта	0	31.05.2023	31.05.2023

На основе выделенных задач и сроков была построена диаграмма Ганта, представленная на рисунке 1.

Рисунок 1 – Диаграмма Ганта

Для подсчета затрат на разработку необходимо определить ресурсы проекта. Выделенные ресурсы приведены в таблице 3.

Таблица 3 – Ресурсы проекта

Название ресурса	Тип	Стандартная ставка	Начисление
Аналитик	Трудовой	100,00 ₽	Пропорциональное
Проектировщик	Трудовой	150,00 ₽	Пропорциональное
Программист	Трудовой	200,00 ₽	Пропорциональное
Тестировщик	Трудовой	150,00 ₽	Пропорциональное
Компьютер	Материальный	20 000,00 ₽	В начале
Периферия	Материальный	5 000,00 ₽	В начале
Бумага	Материальный	300,00 ₽	В начале
Принтер	Материальный	3 000,00 ₽	В начале

Для описанных ранее задач, на основе ресурсов, была рассчитана стоимость, необходимая для выполнения каждого этапа разработки. Результат представлен в таблице 4.

Таблица 4 – Задачи и затраты проекта

Название задачи	Длительность задачи в днях	Название ресурса	Затраты
Начало проекта	0	Компьютер, Перифирия, Бумага, Принтер	28 300,00 ₽
Постановка задачи и анализ аналогов	5		**2 400,00 ₽**
Постановка задачи	1	Аналитик	800,00 ₽
Анализ программ аналогов	1	Аналитик	800,00 ₽
Выбор технологий проектирования и разработки	1	Аналитик	800,00 ₽
Проектирование конфигурации	10		**9 600,00 ₽**
Проектирование логической модели БД	2	Проектировщик	2 400,00 ₽
Проектирование диаграмм UML	4	Проектировщик	4 800,00 ₽
Проектирование пользовательского интерфейса	2	Проектировщик	2 400,00 ₽
Реализация конфигурации	14		**20 800,00 ₽**
Кодирование	11	Программист	17 600,00 ₽
Реализация пользовательского интерфейса	2	Программист	3 200,00 ₽
Тестирование конфигурации	4		**4 800,00 ₽**
Тестирование методом черного ящика	1	Тестировщик	1 200,00 ₽
Тестирование метрикой Холстеда	1	Тестировщик	1 200,00 ₽
Автоматизированное тестирование	2	Тестировщик	2 400,00 ₽
Подготовка документации	11		**8 000,00 ₽**
Написание документации	6	Аналитик	4 800,00 ₽
Оформление и печать документации	4	Аналитик	3 200,00 ₽
Конец проекта	0		

Таким образом, суммарное время для разработки составляет 50 дней, а стоимость проекта – 73 900 рублей.

3 Проектирование информационной системы

Проектирование любой системы заключается в ответе на следующие вопросы:

– какая цель у проекта;
– какие функции должны быть;
– какие данные и архитектура будут использоваться [9].

На основе ответов, можно получить примерный план разработки всего приложения. Грамотно спроектировав архитектуру программы, в дальнейшем получится без проблем масштабировать и поддерживать систему, если такая необходимость появится. Более того, имея четкую схему работы каждого элемента и понимание связей между ними, ускоряется процесс разработки, что позволяет сократить время и средства на создание системы.

3.1 Проектирование с помощью диаграмм UML

UML (Unified Modeling Language) – это унифицированный язык моделирования, позволяющий разработчикам описывать архитектуру программных средств или их компонентов используя единый набор графических обозначений и правил. Язык помогает объяснить суть разрабатываемого объекта разработчикам с разным уровнем технической подготовки, полностью спланировать функционал приложения, а также писать документацию к программному обеспечению [10].

3.1.1 Диаграмма деятельности

Для визуализации алгоритма действий при вводе техники в работу была составлена диаграмма деятельности, представленная в приложении Б.

Диаграмма деятельности представляет собой граф деятельности, вершинами которого являются состояния действий, а дугами – переходы от одного состояния действия к другому. Состоянием действия может быть любое действие, записанное на языке программирования, псевдокоде или естественном языке.

Выполнение деятельности начинается с начального состояния и заканчивается конечным. Переход между состояниями действия происходит последовательно, то есть для начала выполнения следующего действия необходимо завершить текущее. Это можно обойти, используя разделение и слияние параллельных потоков. В этом случае действия выполняются параллельно друг другу и заканчиваются по завершении всех действий потока.

Помимо этого имеется механизм ветвления, позволяющий переходить к действию только при выполнении сторожевого условия. Это помогает, когда необходимо разделить алгоритм на альтернативные ветви. Также есть возможность объединения веток от различных состояний действия в одну [11].

Далее описан алгоритм ввода техники в работу, изображенный на диаграмме деятельности.

Для того чтобы создать документ «Ввод техники в работу» необходимо в первую очередь выбрать рабочее место и указать либо его номер, либо материально ответственное лицо в зависимости от типа рабочего места. Далее сотрудник добавляет как минимум одну строку табличной части, где выбирает технику, её характеристики и количество. При необходимости можно добавить ещё одну строку таблицы и аналогично заполнить. Когда вся необходимая техника добавлена и выбрано рабочее место, сотрудник должен нажать кнопку «Провести». Если вся перечисленная в табличной части техника есть на складе, то документ успешно проводится. Иначе выводится сообщение о том, какой именно номенклатуры не хватает и в каком количестве. Документ в этом случае не проводится.

Если сотрудник во время заполнения документа нажмет кнопку «Подбор рабочего места», то откроется соответствующая форма, где необходимо будет выбрать задачи работника, количество мониторов и другие параметры. После нажатия кнопки «Применить» форма закроется, табличная часть очистится и заполнится согласно заданным параметрам. При необходимости можно добавить необходимую технику вручную.

В случае если будет нажата кнопка «Конфигуратор ПК», откроется специальная форма. На ней сотрудник выбирает составные элементы компьютера, такие как процессор, материнская плата, видеокарта и так далее. По завершении сборки и нажатии кнопки «Применить» созданный компьютер будет добавлен в табличную часть документа.

3.1.2 Диаграмма кооперации

Диаграмма кооперации позволяет описать поведение системы на уровне отдельных элементов или вариантов использования. Объекты представляют собой экземпляры классов, которые создаются на этапе реализации модели или выполнения программы.

Объекты взаимодействуют друг с другом при помощи связей. Связи могут быть экземплярами ассоциации или сообщениями [12].

В качестве примера, была разработана диаграмма кооперации для проведения документа «Ввод техники в работу», представленная в приложении В.

Для создания документа «Ввод техники в работу» сотрудник должен открыть соответствующую форму. Далее необходимо выбрать рабочее место. В зависимости от выбора на форме меняется тип и заголовок поля «НомерЛицо». Если рабочее место выбрано «Материально ответственное лицо», тогда заголовок меняется на одноименный текст, а тип – на «СправочникСсылка.Сотрудники», чтобы был выбор человека из справочника сотрудников. В противном случае заголовок меняется на «Номер места», а тип – на числовой.

После выбора места, сотрудник заполняет табличную часть и по завершении нажимает кнопку «Провести и закрыть». Форма добавляет новый документ в базу данных и закрывается, отображая сообщение об успешном выполнении проведения.

3.1.3 Диаграмма последовательности

Диаграмма последовательности предназначена для более детального описания вариантов использования. Она позволяет отобразить взаимодействие между объектами, взаимодействующими между собой с помощью сообщений. Объекты представляют собой экземпляры классов, сущности или пользователей. Сообщения инициируют выполнение какого-либо действия или возвращают его результат. Отдельные элементы диаграммы можно выделить с помощью фрейма. Например, с помощью фрейма можно задать условия для выполнения отдельных фрагментов или же обозначить их цикличное выполнение. Также важной особенностью диаграммы является отображение течения времени. Поэтому важно не только наличие связей между объектами, но и их положения относительно друг друга [13].

В качестве примера была разработана диаграмма последовательности для проведения документа «Списание техники». Диаграмма представлена в приложении Г.

Был выделен актер «Сотрудник» и 6 объектов:

- форма документа «Списание техники»;
- форма выбора справочника «Сотрудники»;
- форма выбора справочника «Номенклатура»;
- формы выбора справочника «Характеристики»;
- регистр накопления «Техника в работе»;
- регистр накопления «Списанная техника».

Для создания документа необходимо открыть форму и выбрать рабочее место. Если рабочее место выбрано «Материально ответственное лицо», то нужно открыть справочник «Сотрудники» и выбрать там человека, которому будет назначена техника. Если рабочее место иное, то необходимо ввести номер места, откуда будет списано оборудование.

Далее сотрудник должен добавить минимум одну строку табличной части и заполнить её, выбрав номенклатуру и характеристику из соответствующих

справочников, а также ввести количество. Сотрудник может добавлять столько строк, сколько потребуется.

Для завершения создания документа, необходимо нажать кнопку «Провести и закрыть». После этого произойдет проверка наличия перечисленной техники в регистре накопления «Техника в работе» и если она будет, то будет создано движение вида «Расход» для регистра накопления «Техника в работе» и движение «Приход» для регистра накопления «Списанная техника». После этого форма закрывается и документ создается. В случае если какой-либо номенклатуры не будет в наличии, будет отображено соответствующее сообщение. Документ при этом не может быть создан.

3.2 Проектирование логической модели базы данных

Для описания структуры данных необходимо спроектировать логическую модель базы данных. Она представляет собой информацию о сущностях и отношениях между ними. Сущностью называется таблица, хранящая данные об объекте. Сущность состоит из атрибутов, которые являются свойствами объекта. Отношения описывают связь сущностей между собой [14]. Связи могут быть трех видов:

– один к одному – запись одной таблицы может иметь только одну связанную запись другой таблицы;

– один ко многим – запись одной таблицы может иметь множество связанных записей в других таблицах;

– многие ко многим – множество записей одной таблицы может иметь множество связанных записей в других таблицах [15].

Составленная логическая модель базы данных представлена в приложении Д. Описание сущностей и их атрибутов приведено в таблице 5.

Таблица 5 – Описание сущностей и их атрибутов

Сущность	Атрибут	Роль атрибута	Описание атрибута
Номенклатура	Код	PK	Уникальный код справочника
	Наименование		Наименование номенклатуры
	Родитель	FK	Ссылка на объект «Номенклатура»

Продолжение таблицы 5

Сущность	Атрибут	Роль атрибута	Описание атрибута
Фирмы ремонта	Код	PK	Уникальный код справочника
	Наименование		Наименование фирмы ремонта
Сотрудники	Код	PK	Уникальный код справочника
	ФИО		Фамилия, имя, отчество сотрудника
Характеристики	Код	PK	Уникальный код справочника
	Наименование		Наименование характеристики
	Владелец	FK	Ссылка на объект «Номенклатура»
Значения характеристик	Код	PK	Уникальный код регистра сведений
	Характеристика	FK	Ссылка на объект «Характеристики»
	Свойство	FK	Ссылка на объект «Свойства номенклатуры»
	Значение		Значение свойства характеристики
Свойства номенклатуры	Код	PK	Уникальный код плана видов характеристик
	Наименование		Наименование свойства номенклатуры
	Тип значения		Тип значения свойства номенклатуры
Значения свойств	Код	PK	Уникальный код справочника
	Наименование		Наименование значения свойства
	Владелец	FK	Ссылка на объект «Свойства номенклатуры»
Приходная накладная	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
Табличная часть документа «Приходная накладная»	Код документа	PK	Уникальный код документа «Приходная накладная»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры
Заявки для заказа	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
Табличная часть документа «Заявки для заказа»	Код документа	PK	Уникальный код документа «Заявки для заказа»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры
Ввод техники в работу	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
	Рабочее место	FK	Ссылка на объект «Рабочие места»
	НомерЛицо	FK	Число либо ссылка на объект «Сотрудники»
Табличная часть документа «Ввод техники в работу»	Код документа	PK	Уникальный код документа «Заявки для заказа»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры
Списание техники	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
	Рабочее место	FK	Ссылка на объект «Рабочие места»
	НомерЛицо	FK	Число либо ссылка на объект «Сотрудники»
Табличная часть документа «Списание техники»	Код документа	PK	Уникальный код документа «Списание техники»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры

Продолжение таблицы 5

Сущность	Атрибут	Роль атрибута	Описание атрибута
Передача техники	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
	Рабочее место	FK	Ссылка на объект «Рабочие места»
	НомерЛицо	FK	Число либо ссылка на объект «Сотрудники»
	Новое рабочее место	FK	Ссылка на объект «Рабочие места»
	Новое номерЛицо	FK	Число либо ссылка на объект «Сотрудники»
Табличная часть документа «Передача техники»	Код документа	PK	Уникальный код документа «Передача техники»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры
Передача в ремонт	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
	Рабочее место	FK	Ссылка на объект «Рабочие места»
	НомерЛицо	FK	Число либо ссылка на объект «Сотрудники»
	Дата окончания ремонта		Дата когда ремонт будет завершен
	Фирма исполнитель		Ссылка на объект «Фирмы ремонта»
Табличная часть документа «Передача в ремонт»	Код документа	PK	Уникальный код документа «Передача в ремонт»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры
Приход с ремонта	Код	PK	Уникальный код документа
	Дата		Дата создания документа
	Табличная часть	FK	Ссылка на объект «Табличная часть»
	Рабочее место	FK	Ссылка на объект «Рабочие места»
	НомерЛицо	FK	Число либо ссылка на объект «Сотрудники»
Табличная часть документа «Приход с ремонта»	Код документа	PK	Уникальный код документа «Приход с ремонта»
	Код номенклатуры	PK	Уникальный код номенклатуры
	Код характеристики	PK	Уникальный код характеристики
	Количество		Количество номенклатуры
Процессоры	Код	PK	Уникальный код справочника
	Наименование		Наименование процессора
	Сокет	FK	Ссылка на объект «Сокеты»
	Тип памяти	FK	Ссылка на объект «Типы памяти»
	Энергопотребление		Количество потребляемой электроэнергии в Вт
	Видеядро		Наличие видеоядра у процессора
	Количество ядер		Количество ядер процессора
Оперативная память	Код	PK	Уникальный код справочника
	Наименование		Наименование процессора
	Тип памяти	FK	Ссылка на объект «Типы памяти»
	Объем		Количество Гб оперативной памяти
	Частота		Частота оперативной памяти
	Количество модулей		Количество модулей оперативной памяти в комплекте
Материнские платы	Код	PK	Уникальный код справочника
	Наименование		Наименование процессора
	Сокет	FK	Ссылка на объект «Сокеты»
	Тип памяти	FK	Ссылка на объект «Типы памяти»

Окончание таблицы 5

Сущность	Атрибут	Роль атрибута	Описание атрибута
Хранение данных	Код	PK	Уникальный код справочника
	Наименование		Наименование устройства хранения данных
	Энергопотребление		Количество потребляемой электроэнергии в Вт
	Объем		Количество Гб устройства хранения данных
	Тип	FK	Ссылка на объект «Тип хранения данных»
Видеокарты	Код	PK	Уникальный код справочника
	Наименование		Наименование видеокарты
	Объем видеопамяти		Количество видеопамяти в Гб
	Энергопотребление		Количество потребляемой электроэнергии в Вт
	Разъем питания		Количество PIN разъема питания
Блоки питания	Код	PK	Уникальный код справочника
	Наименование		Наименование видеокарты
	Мощность		Мощность блока питания в Вт
	Разъемы GPU		Количество PIN для питания видеокарты

Таким образом, логическая модель данных имеет 27 сущностей и позволяет создать конфигурацию согласно поставленным требованиям.

3.3 Проектирование пользовательского интерфейса

При запуске приложения пользователя встречает начальная страница, на которой отображается сводный отчет по запасам техники, а также список документов о номенклатуре, у которой завершился ремонт. Внешний вид страницы представлен на рисунке 2.

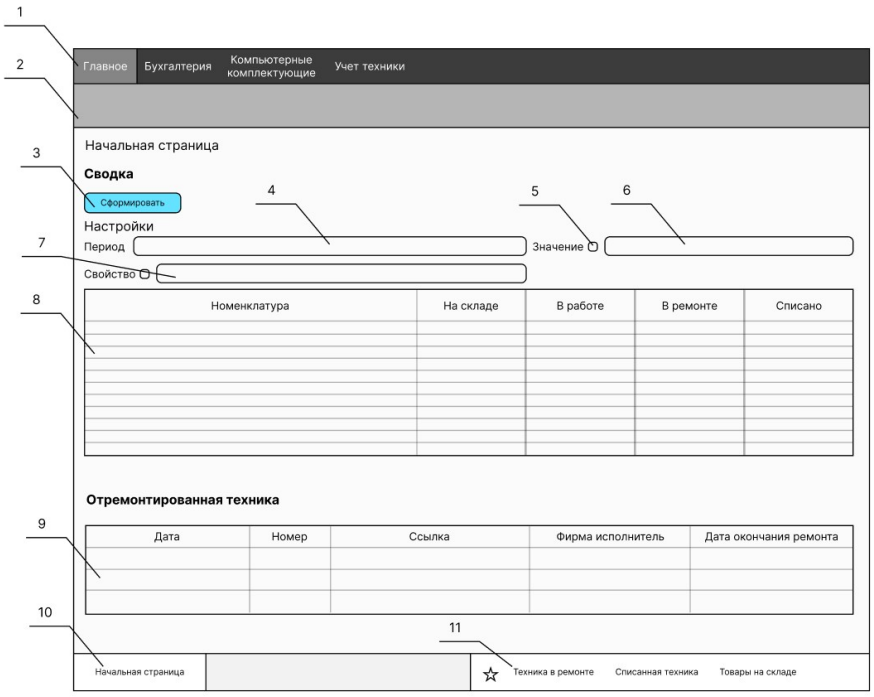


Рисунок 2 – Начальная страница системы

На начальной странице представлены следующие элементы:

1 – панель разделов приложения;

2 – панель функций текущего раздела;

3 – кнопка формирования отчета;

4 – поле выбора даты для отчета;

5 – переключатель для включения фильтра;

6 – поле выбора значения свойства;

7 – поле выбора свойства

8 – табличная часть, где указывается номенклатура и её количество на складе, в работе, в ремонте, а также сколько списано;

9 – табличная часть, где указываются документы с техникой, у которой завершен ремонт;

10 – панель открытых вкладок;

11 – панель избранного.

Система содержит три главных раздела: «Бухгалтерия», «Учет техники» и «Компьютерные комплектующие».

В разделе «Бухгалтерия» содержатся:

- документ «Заявки для заказа»;
- справочник «Сотрудники»;
- справочник «Номенклатура»;
- справочник «Фирмы ремонта».

В разделе «Учет техники» содержатся:

- документ «Приходная накладная»;
- документ «Ввод техники в работу»;
- документ «Передача в ремонт»;
- документ «Приход с ремонта»;
- документ «Списание техники».

В разделе «Компьютерные комплектующие» содержатся:

- справочник «Процессоры»;
- справочник «Материнские платы»;

– справочник «Оперативная память»;
– справочник «Видеокарты»;
– справочник «Хранение данных»;
– справочник «Блоки питания».

В нижней левой части экрана находится панель открытых вкладок. Благодаря этому пользователь может с легкостью ориентироваться в приложении и быстро переключаться между задачами.

В нижней правой стороне располагается панель избранного, с помощью которой пользователь может иметь быстрый доступ к необходимым ему страницам.

4 Реализация информационной системы

4.1 Реализация структур программного средства

Разработка на платформе 1С: Предприятие 8.3 представляет собой создание и настройку различных объектов конфигурации. С помощью них разработчик может с легкостью создать каркасную версию системы, которую в дальнейшем дополнить сложными и специфическими функциями.

В ходе разработки конфигурации были использованы следующие объекты:

– справочники;
– документы;
– перечисления;
– план видов характеристик;
– регистры сведений;
– регистры накопления;
– общие формы;
– отчеты.

Для демонстрации всех объектов конфигурации была составлена диаграмма компонентов.

Диаграмма компонентов предназначена для описания физического представления системы. На диаграмме устанавливаются зависимости между компонентами, тем самым определяя архитектуру приложения. Компонентом является физически существующая часть системы, например коды модулей или командные файлы. Компоненты связаны друг с другом при помощи отношения зависимости, то есть изменения одного объекта влияет на поведение другого [16].

Диаграмма компонентов конфигурации представлена в приложении Е.

Справочники. Они предназначены для хранения в базе данных информации с однотипной структурой [17]. Для разрабатываемой конфигурации были созданы 11 справочников.

1. Справочник «Сотрудники» доступен в подсистеме «Бухгалтерия» и содержит ФИО сотрудника, работающего в организации. Форма справочника представлена на рисунке 3.

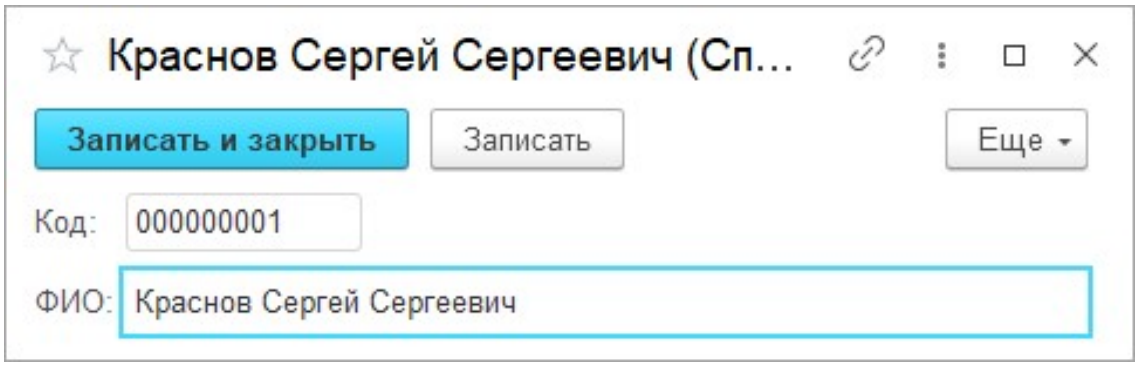


Рисунок 3 – Форма справочника «Сотрудники»

2. Справочник «Номенклатура» доступен в подсистеме «Учет техники» и содержит наименование и категорию техники. Форма справочника представлена на рисунке 4.

Рисунок 4 – Форма справочника «Номенклатура»

3. Справочник «Характеристики» расположен на форме справочника «Номенклатура» и содержит наименование и ссылку на владельца. Форма справочника представлена на рисунке 5.

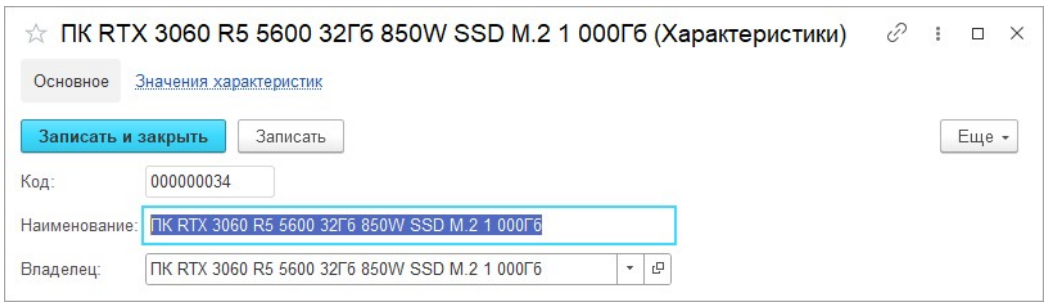


Рисунок 5 – Форма справочника «Характеристики»

4. Справочник «Значения свойств» доступен на форме регистра сведений «Значения характеристик» и содержит наименование и владельца. Форма справочника представлена на рисунке 6.

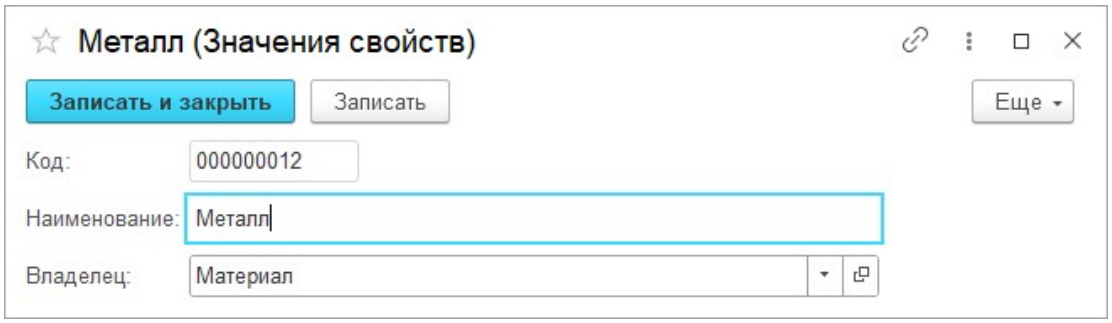


Рисунок 6 – Форма справочника «Значения свойств»

5. Справочник «Фирмы ремонта» доступен в подсистеме «Бухгалтерия» и содержит наименование компании, осуществляющей ремонт техники. Форма справочника представлена на рисунке 7.

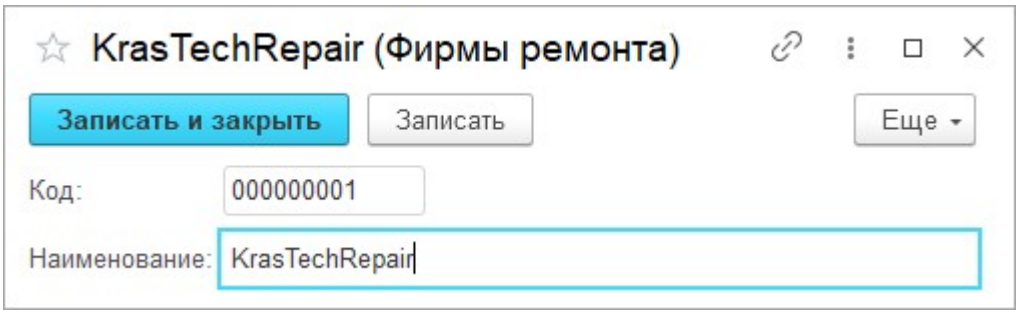


Рисунок 7 – Форма справочника «Фирмы ремонта»

6. Справочник «Процессоры» доступен в подсистеме «Компьютерные комплектующие» и содержит такие характеристики процессора как: наименование, сокет, поддерживаемый тип памяти, энергопотребление,

наличие видеоядра и количество ядер. Форма справочника представлена на рисунке 8.

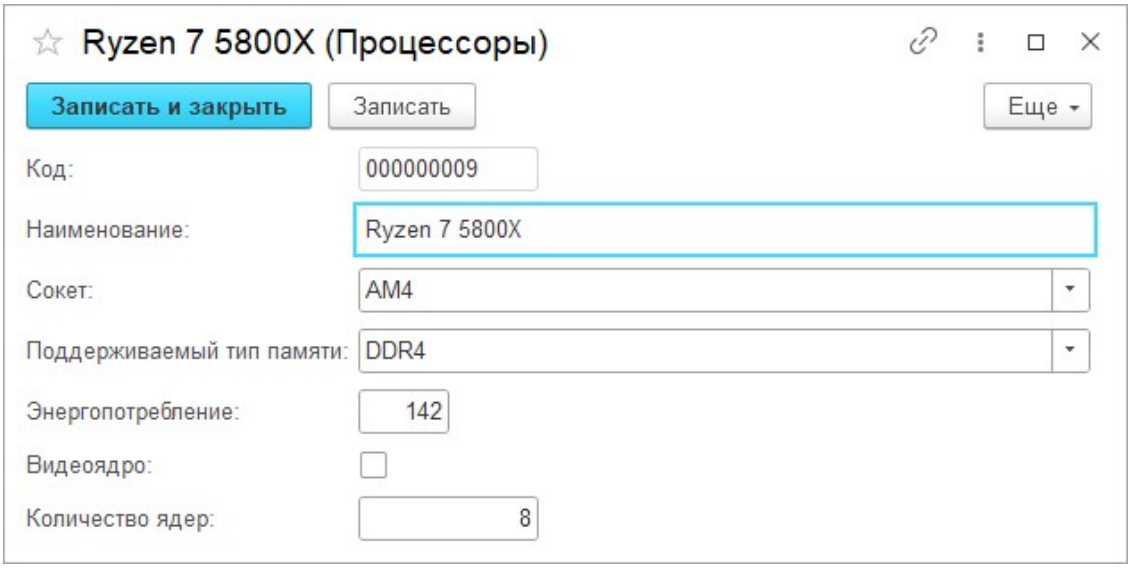


Рисунок 8 – Форма справочника «Процессоры»

7. Справочник «Материнские платы» доступен в подсистеме «Компьютерные комплектующие» и содержит такие характеристики материнские платы как: наименование, сокет и поддерживаемый тип памяти. Форма справочника представлена на рисунке 9.

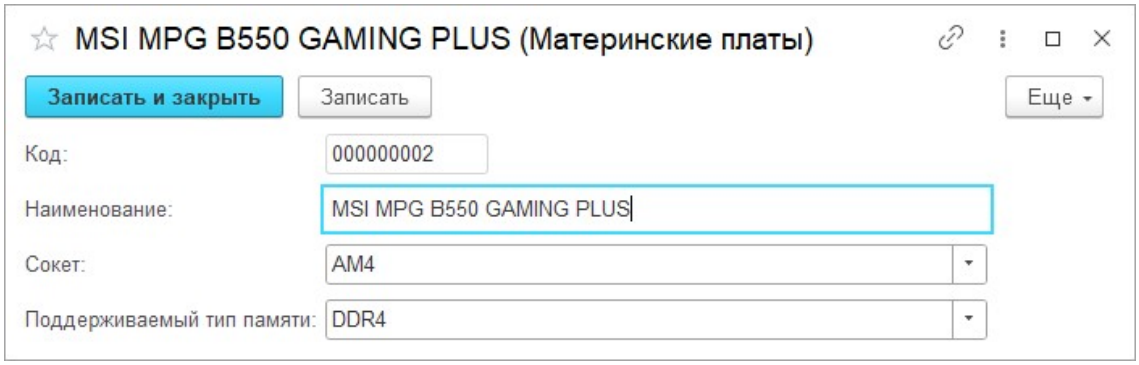


Рисунок 9 – Форма справочника «Материнские платы»

8. Справочник «Видеокарты» доступен в подсистеме «Компьютерные комплектующие» и содержит такие характеристики видеокарты как: наименование, объем видеопамяти, энергопотребление и количество PIN разъема питания. Форма справочника представлена на рисунке 10.

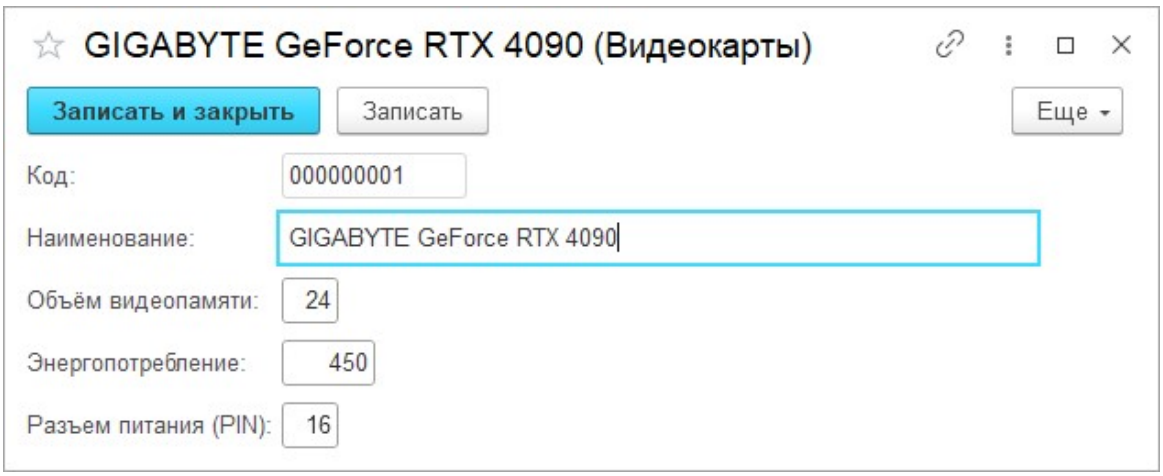


Рисунок 10 – Форма справочника «Видеокарты»

9. Справочник «Оперативная память» доступен в подсистеме «Компьютерные комплектующие» и содержит такие характеристики оперативной памяти как: наименование, тип памяти, объем, частота и количество модулей в комплекте. Форма справочника представлена на рисунке 11.

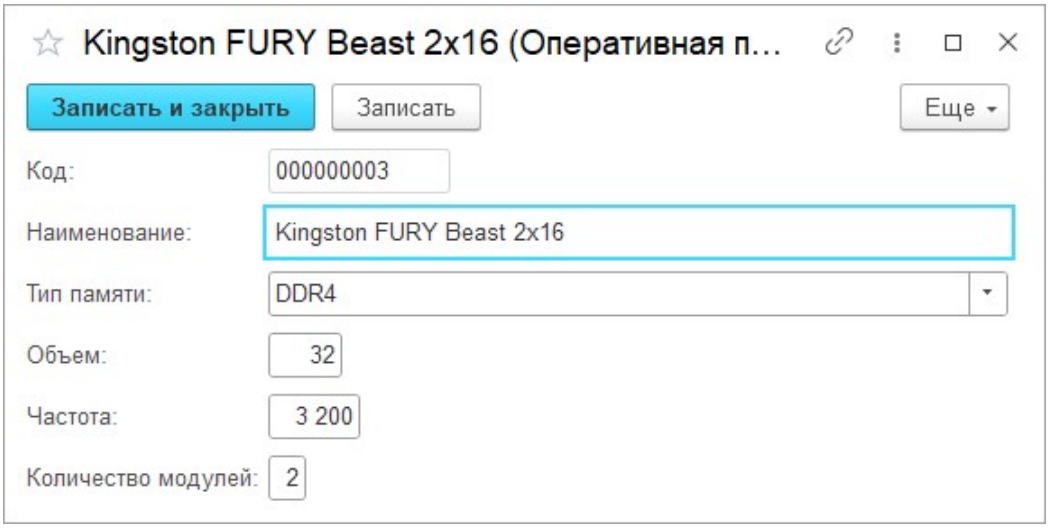


Рисунок 11 – Форма справочника «Оперативная память»

10. Справочник «Хранение данных» доступен в подсистеме «Компьютерные комплектующие» и содержит такие характеристики устройства хранения данных как: наименование, энергопотребление, объем и тип устройства. Форма справочника представлена на рисунке 12.

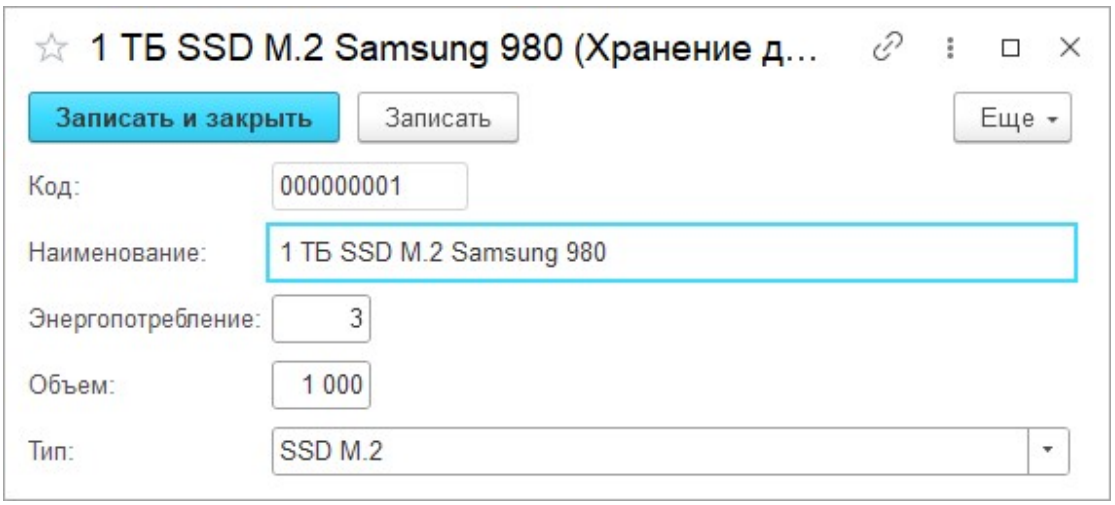


Рисунок 12 – Форма справочника «Хранение данных»

11. Справочник «Блоки питания» доступен в подсистеме «Компьютерные комплектующие» и содержит такие характеристики блока питания как: наименование, мощность и количество PIN для питания видеокарты. Форма справочника представлена на рисунке 13.

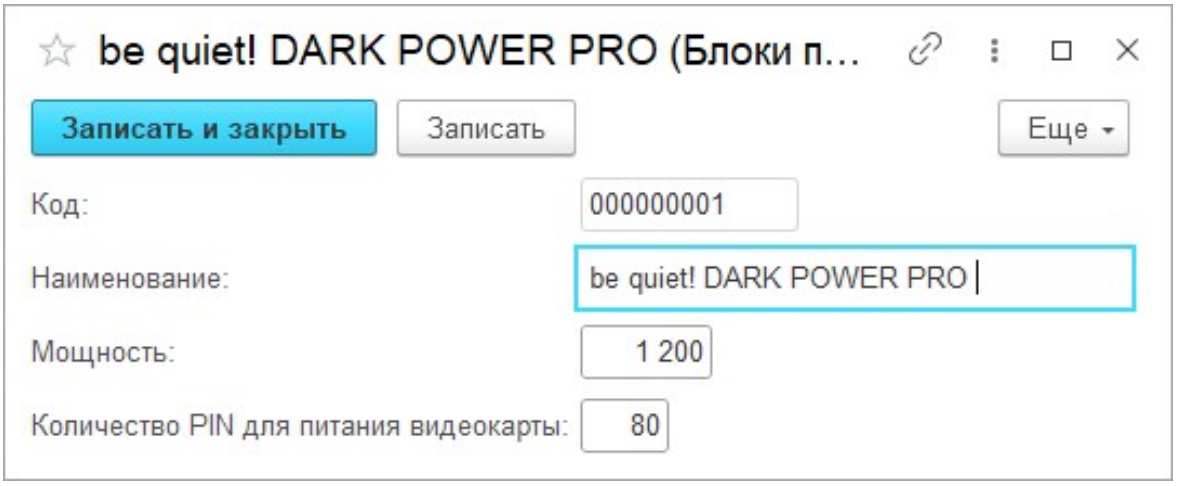


Рисунок 13 – Форма справочника «Хранение данных»

Перечисления. Они предназначены для хранения в базе данных фиксированных, то есть неизменяемых значений [18]. Для разрабатываемой конфигурации были созданы 5 перечислений.

1. Перечисление «Рабочие места». Предназначено для хранения типов рабочих мест. Имеет следующие значения: отдел, офис, территория, материально ответственное лицо.

2. Перечисление «Задачи работника». Предназначено для хранения типов задач работника. Имеет следующие значения: документооборот, программирование, работа с графикой.

3. Перечисление «Сокеты». Предназначено для хранения типов сокетов. Имеет следующие значения: LGA1200, LGA1700, AM4.

4. Перечисление «ТипыПамяти». Предназначено для хранения типов оперативной памяти. Имеет следующие значения: DDR3, DDR4, DDR5.

5. Перечисление «Тип хранения данных». Предназначено для хранения типов устройств для хранения данных. Имеет следующие значения: SSD, HDD, SSDM2.

Документы. Они предназначены для хранения в базе данных информации о произошедших в процессе работы событиях, то есть неизменяемых значений [19]. Для разрабатываемой конфигурации были созданы 7 документов.

1. Документ «Приходная накладная» доступен в подсистеме «Бухгалтерия» и предназначен для фиксации факта поступления номенклатуры на склад. Форма документа представлена на рисунке 14. Печатная форма документа приведена на рисунке 15.

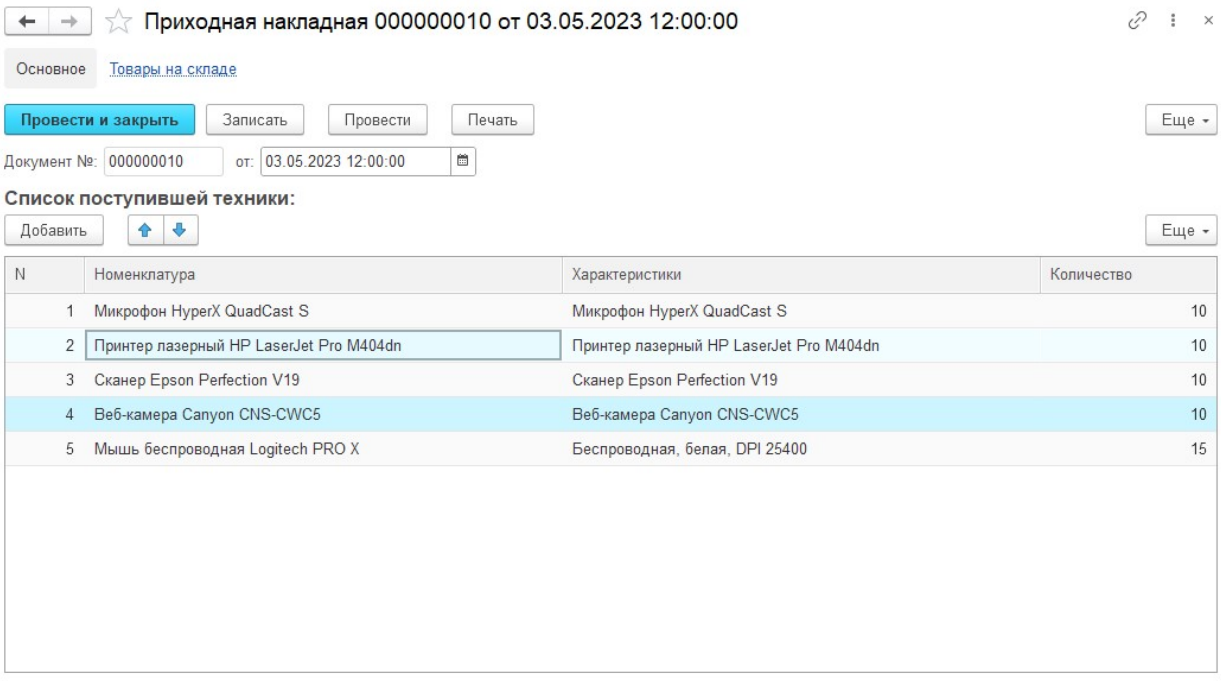


Рисунок 14 – Форма документа «Приходная накладная»

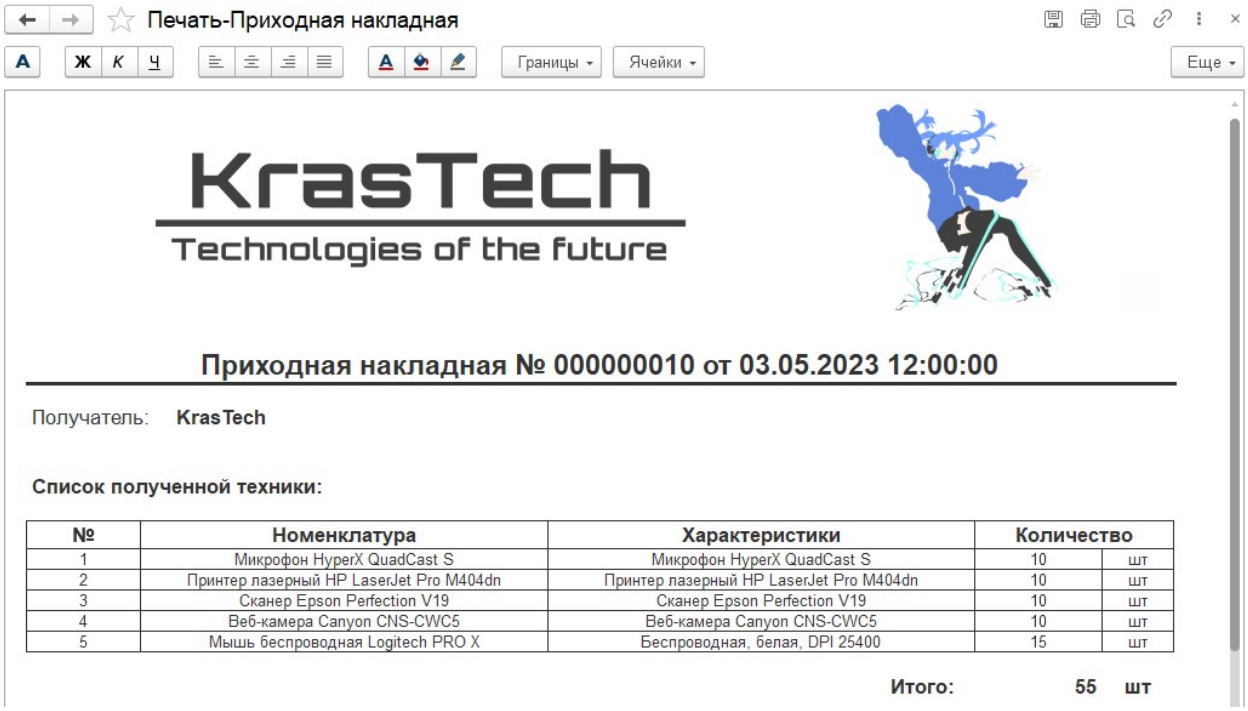


Рисунок 15 – Печатная форма документа «Приходная накладная»

2. Документ «Ввод техники в работу» доступен в подсистеме «Учет техники» и предназначен для фиксации факта ввода техники в работу на рабочее место. Форма документа представлена на рисунке 16. Печатная форма документа приведена на рисунке 17.

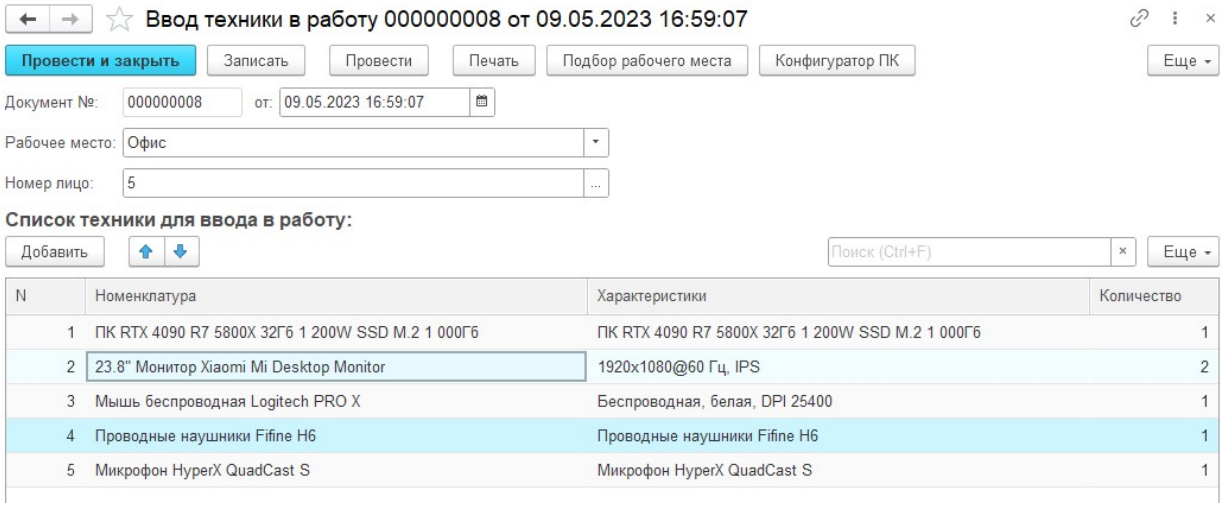


Рисунок 16 – Форма документа «Ввод техники в работу»

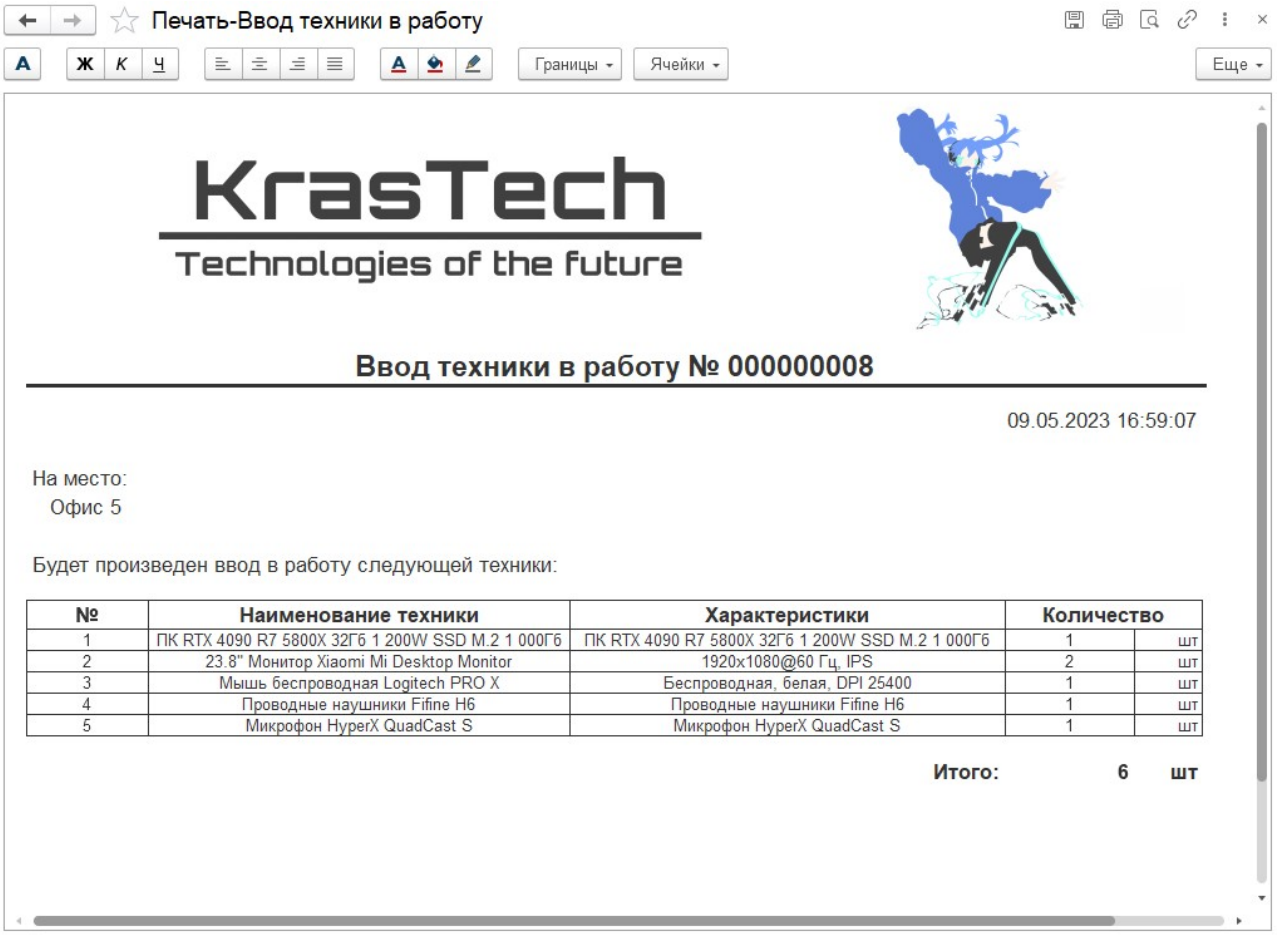


Рисунок 17 – Печатная форма документа «Ввод техники в работу»

3. Документ «Списание техники» доступен в подсистеме «Учет техники» и предназначен для фиксации факта списания техники. Форма документа представлена на рисунке 18. Печатная форма документа приведена на рисунке 19.

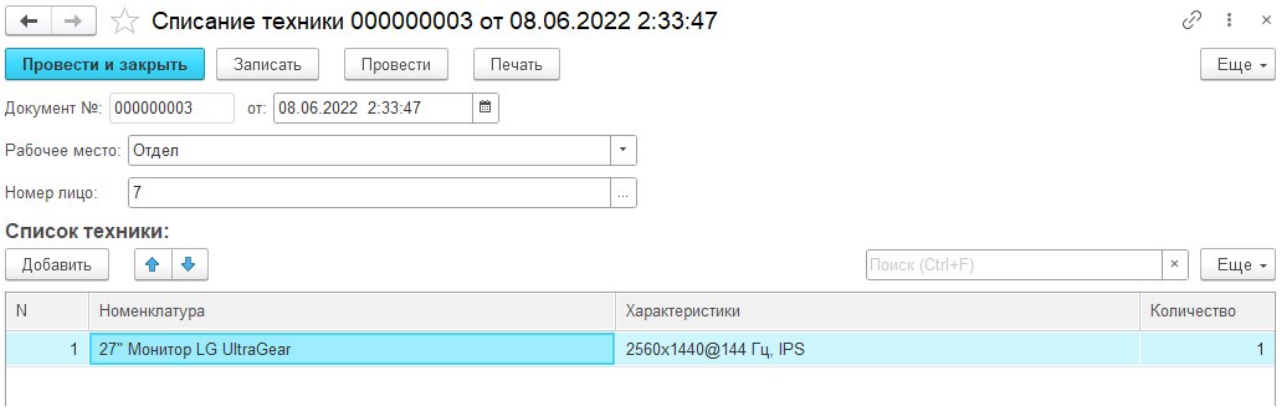


Рисунок 18 – Форма документа «Списание техники»

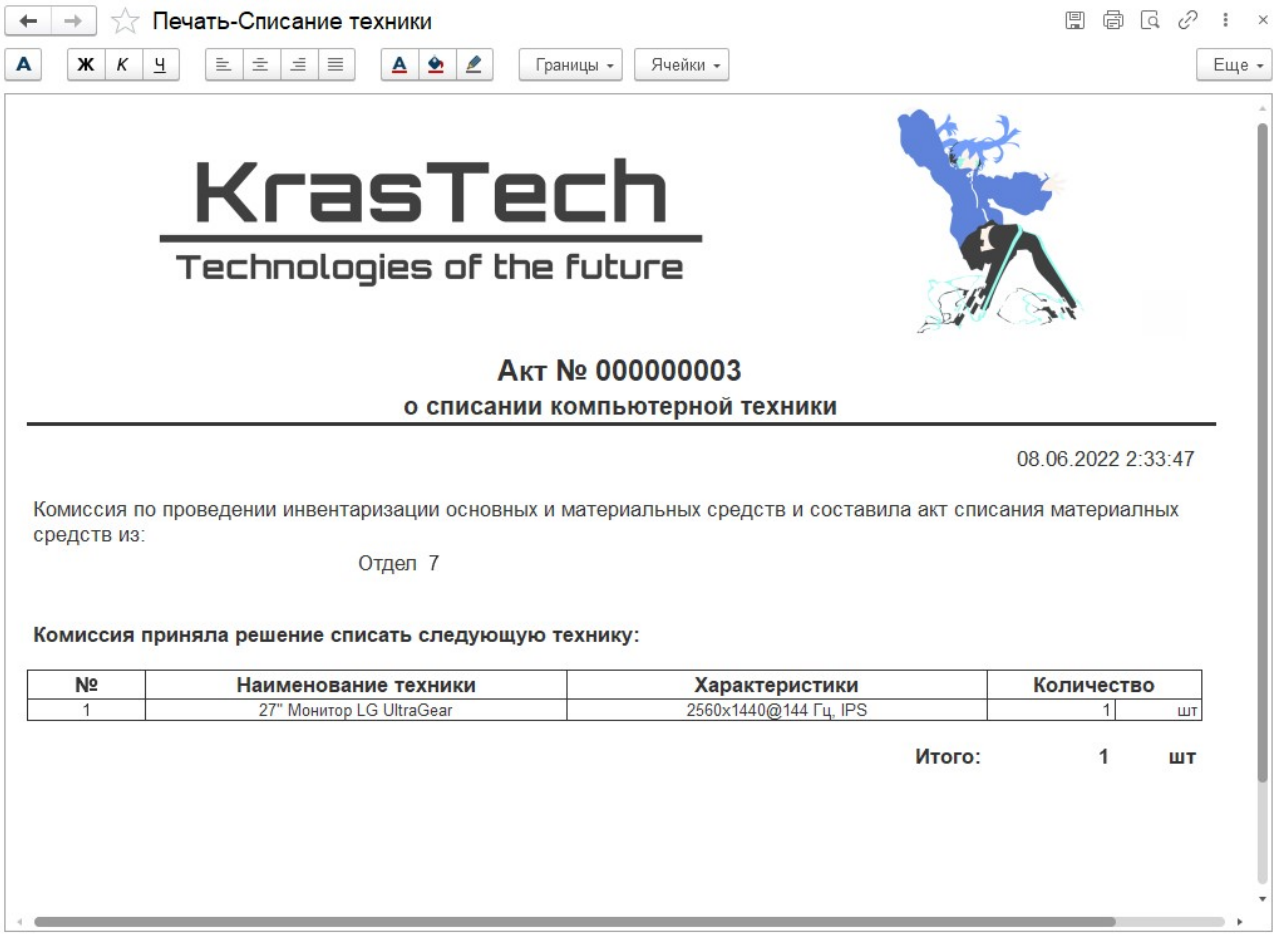


Рисунок 19 – Печатная форма документа «Списание техники»

4. Документ «Передача в ремонт» доступен в подсистеме «Учет техники» и предназначен для фиксации факта передачи техники в ремонт в указанную фирму. Форма документа представлена на рисунке 20. Печатная форма документа приведена на рисунке 21.

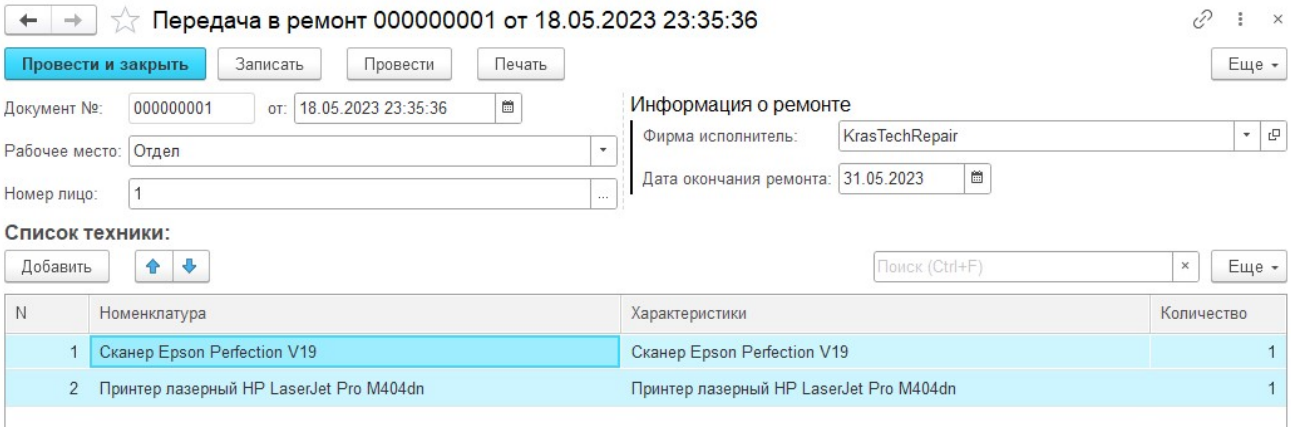


Рисунок 20 – Форма документа «Передача в ремонт»

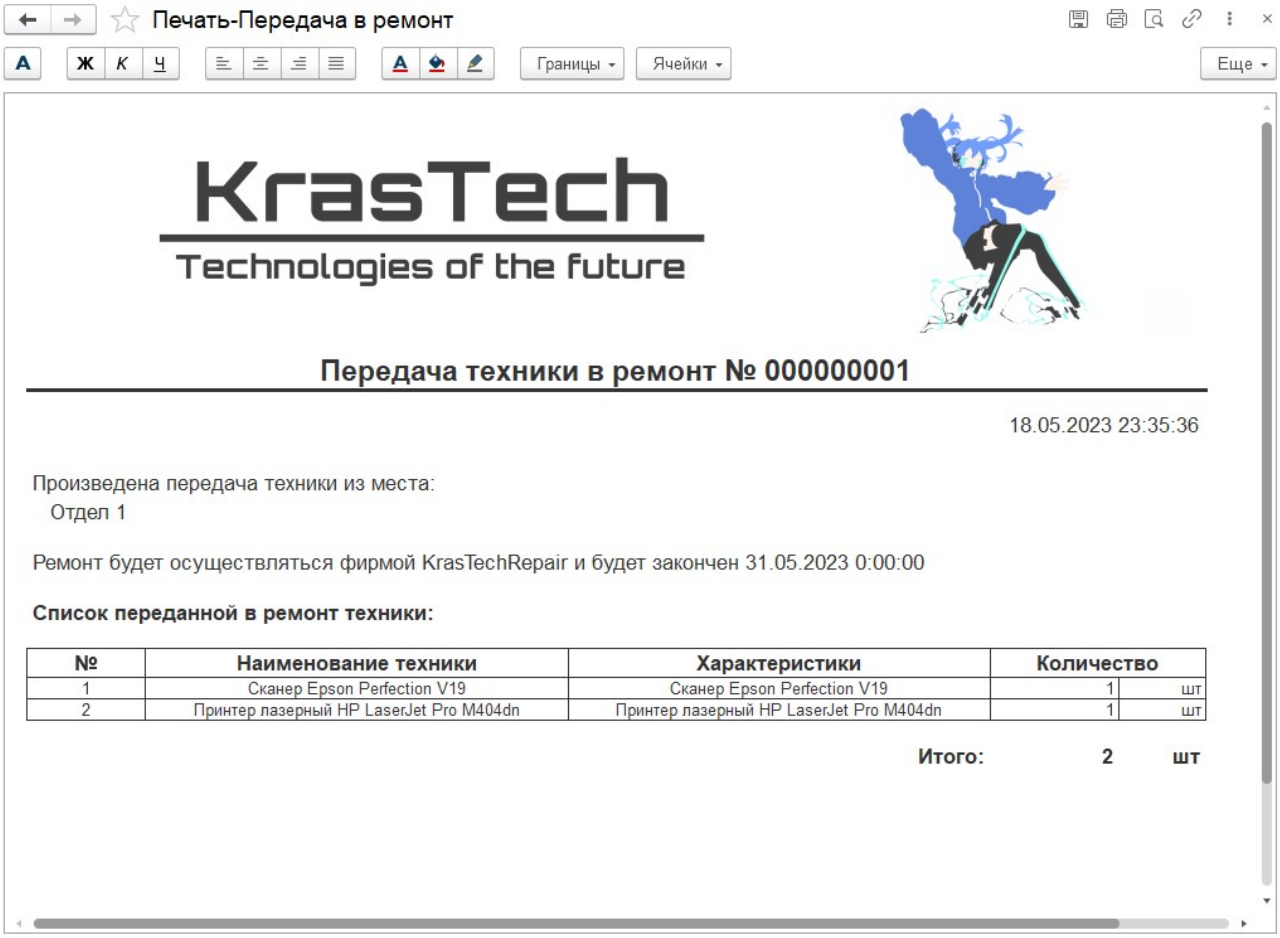


Рисунок 21 – Печатная форма документа «Передача в ремонт»

5. Документ «Приход с ремонта» доступен в подсистеме «Учет техники» и предназначен для фиксации факта приема техники из ремонта. Форма документа представлена на рисунке 22. Печатная форма документа приведена на рисунке 23.

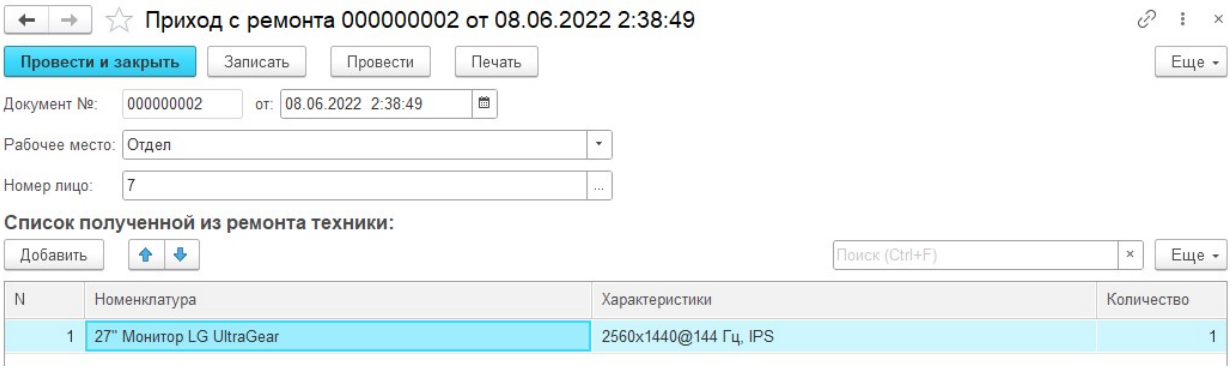


Рисунок 22 – Форма документа «Приход с ремонта»

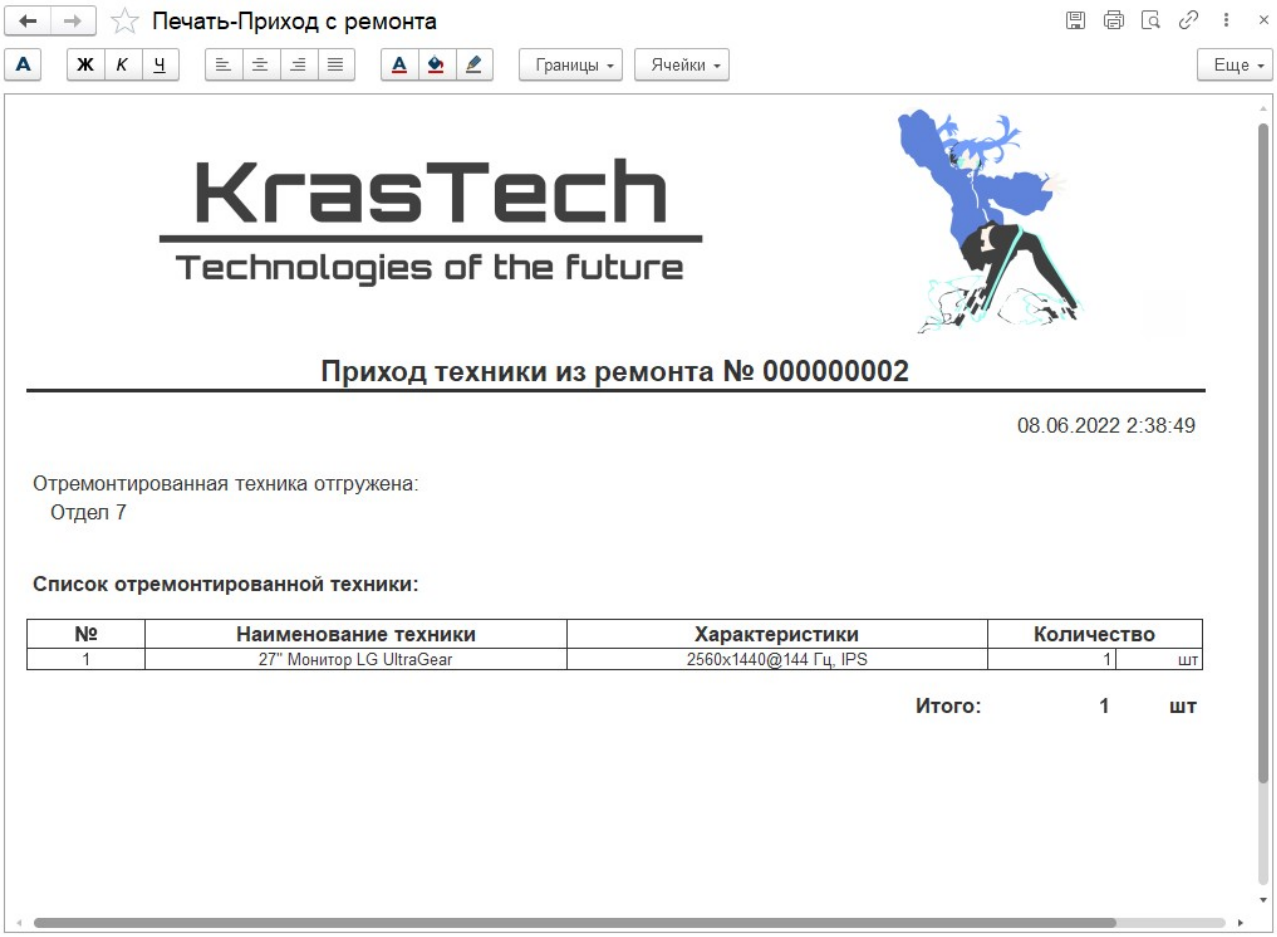


Рисунок 23 – Печатная форма документа «Приход с ремонта»

6. Документ «Передача техники» доступен в подсистеме «Учет техники» и предназначен для фиксации факта перемещения техники с одного рабочего места на другое. Форма документа представлена на рисунке 24. Печатная форма документа приведена на рисунке 25.

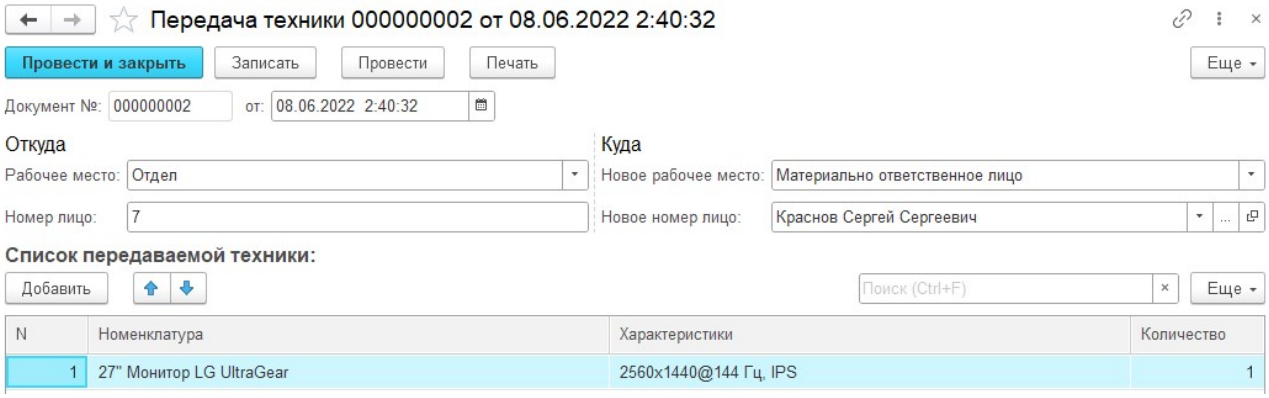


Рисунок 24 – Форма документа «Передача техники»

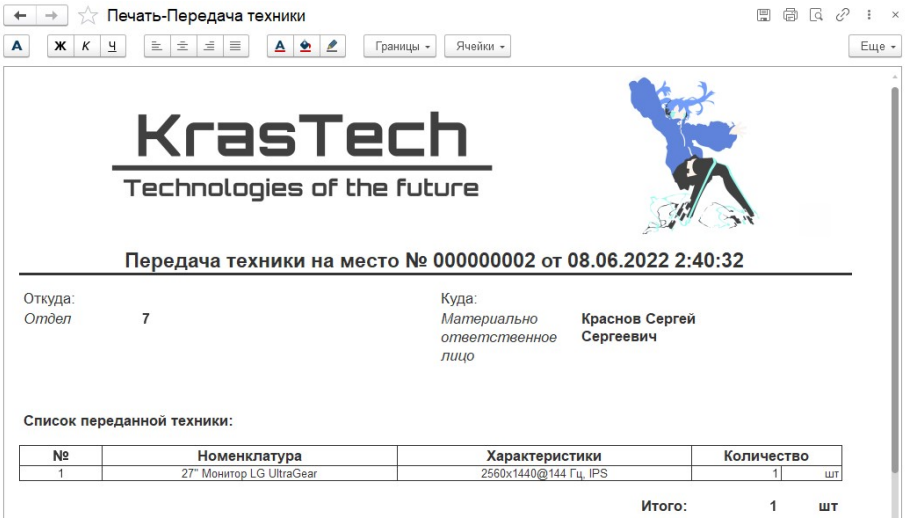


Рисунок 25 – Печатная форма документа «Передача техники»

7. Документ «Заявки для заказа» доступен в подсистеме «Бухгалтерия» и предназначен для фиксации факта создания заявки для заказа техники. Форма документа представлена на рисунке 26. Печатная форма документа приведена на рисунке 27.

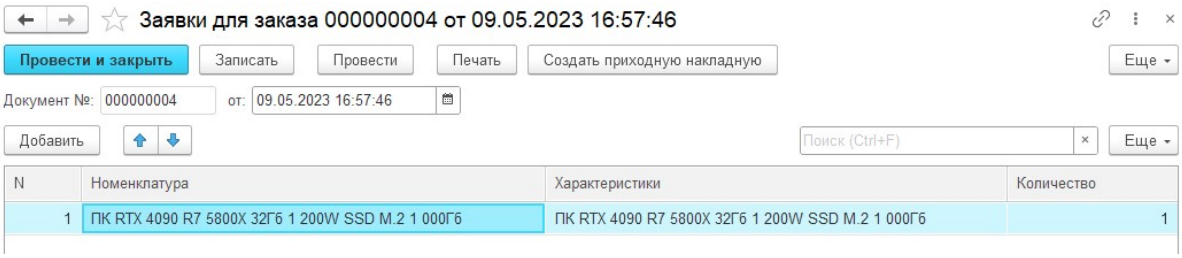


Рисунок 26 – Форма документа «Заявки для заказа»

Рисунок 27 – Печатная форма документа «Заявки для заказа»

Планы видов характеристик. Они предназначены для хранения информации о характеристиках объектов [20]. Для разрабатываемой конфигурации был создан план видов характеристик «Свойства номенклатуры» для хранения информации о характеристиках номенклатуры. План видов характеристик представлен на рисунке 28.

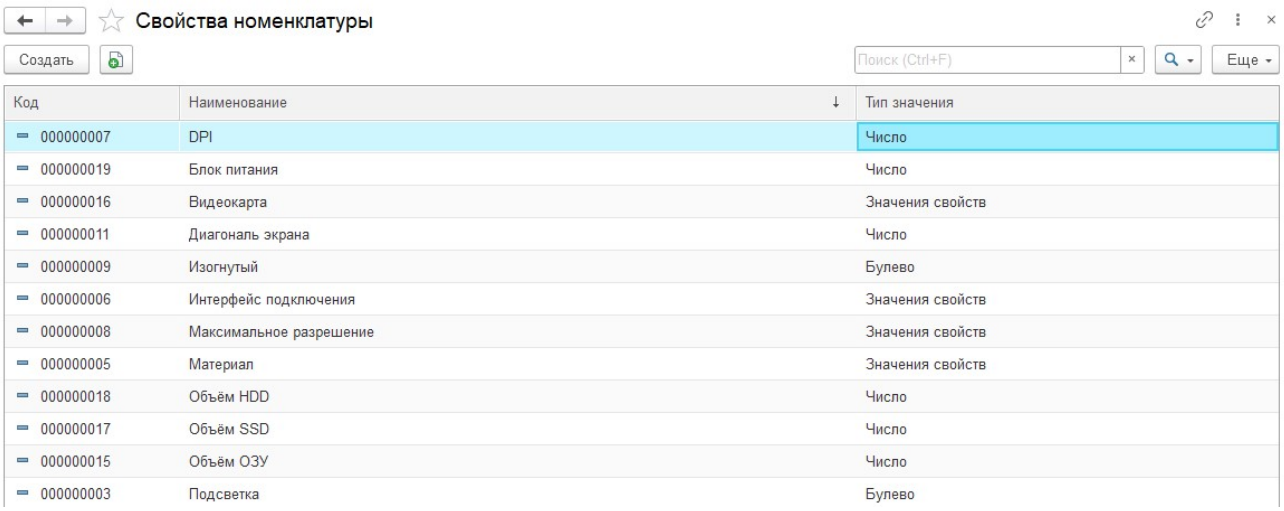


Рисунок 28 – План видов характеристик «Свойства номенклатуры»

Регистры сведений. Они предназначены для хранения информации в разрезе измерений [21]. Для разрабатываемой конфигурации был создан регистр сведений «Значения характеристик» для хранения характеристик и их свойств. Регистр сведений представлен на рисунке 29.

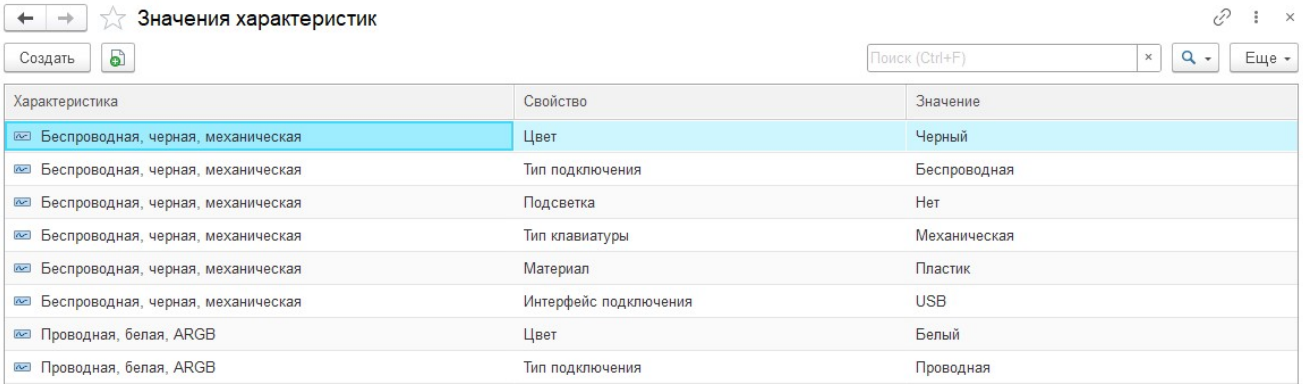
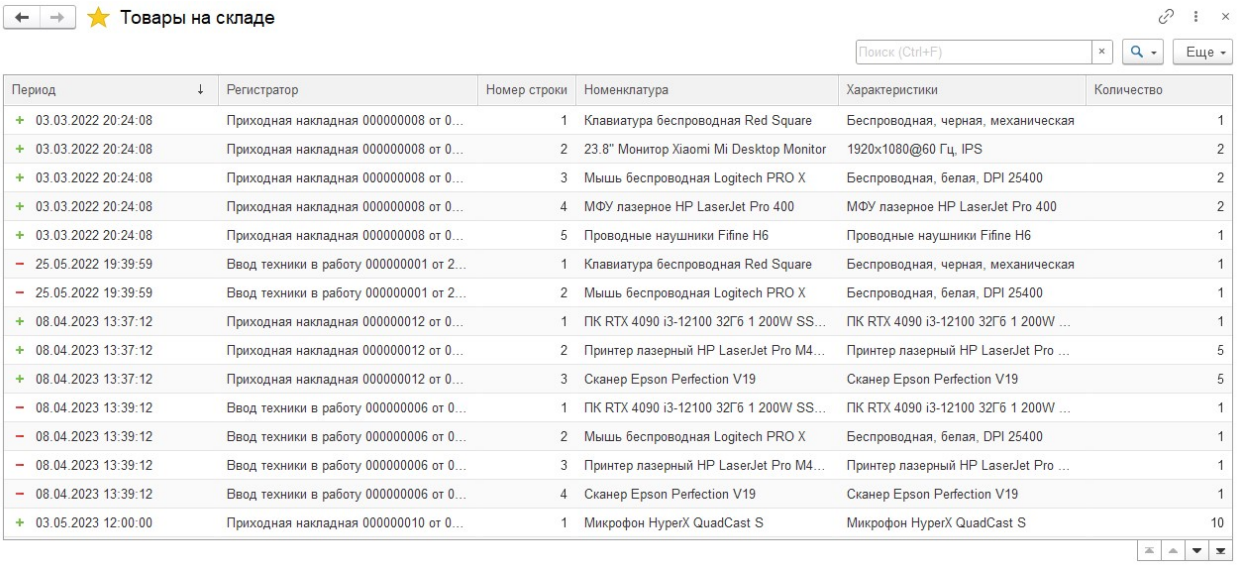


Рисунок 29 – Регистр сведений «Значения характеристик»

Регистры накопления. Они предназначены для хранения информации в разрезе измерений и позволяют накапливать числовые данные [22]. Для разрабатываемой конфигурации были созданы 4 регистра накопления.

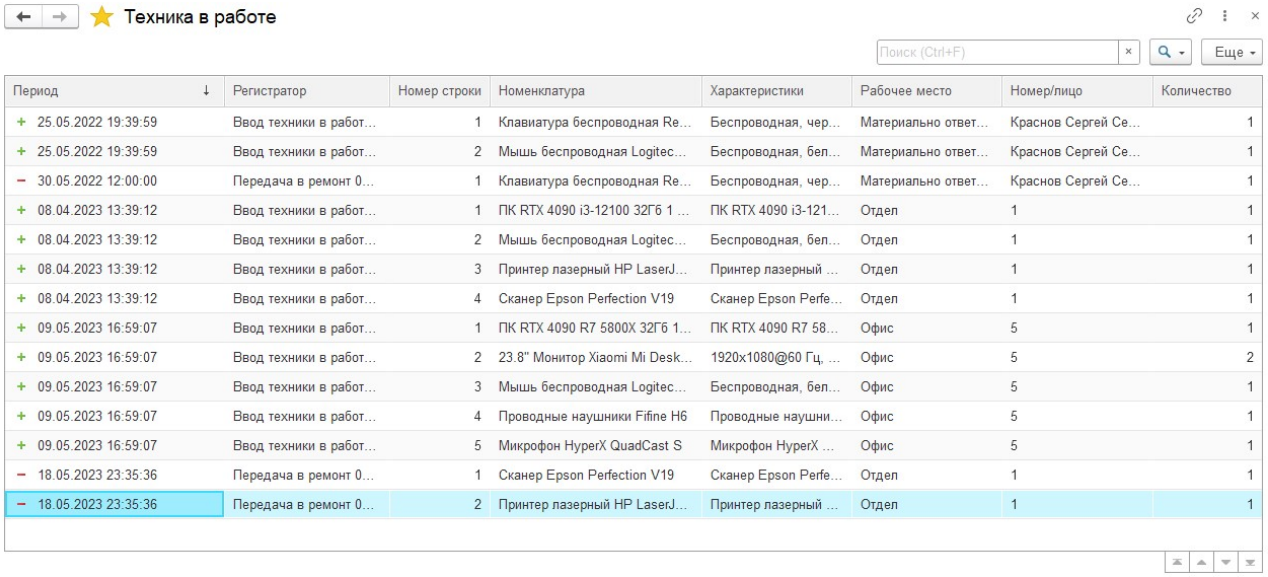
1. Регистр накопления «Товары на складе» предназначен для накопления информации о товарах на складе. Является регистратором для документов «Приходная накладная» и «Ввод техники в работу». Регистр накопления представлен на рисунке 30.



Период	Регистратор	Номер строки	Номенклатура	Характеристики	Количество
+ 03.03.2022 20:24:08	Приходная накладная 000000008 от 0…	1	Клавиатура беспроводная Red Square	Беспроводная, черная, механическая	1
+ 03.03.2022 20:24:08	Приходная накладная 000000008 от 0…	2	23.8" Монитор Xiaomi Mi Desktop Monitor	1920x1080@60 Гц, IPS	2
+ 03.03.2022 20:24:08	Приходная накладная 000000008 от 0…	3	Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400	2
+ 03.03.2022 20:24:08	Приходная накладная 000000008 от 0…	4	МФУ лазерное HP LaserJet Pro 400	МФУ лазерное HP LaserJet Pro 400	2
+ 03.03.2022 20:24:08	Приходная накладная 000000008 от 0…	5	Проводные наушники Fifine H6	Проводные наушники Fifine H6	1
− 25.05.2022 19:39:59	Ввод техники в работу 000000001 от 2…	1	Клавиатура беспроводная Red Square	Беспроводная, черная, механическая	1
− 25.05.2022 19:39:59	Ввод техники в работу 000000001 от 2…	2	Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400	1
+ 08.04.2023 13:37:12	Приходная накладная 000000012 от 0…	1	ПК RTX 4090 i3-12100 32Гб 1 200W SS…	ПК RTX 4090 i3-12100 32Гб 1 200W …	1
+ 08.04.2023 13:37:12	Приходная накладная 000000012 от 0…	2	Принтер лазерный HP LaserJet Pro M4…	Принтер лазерный HP LaserJet Pro …	5
+ 08.04.2023 13:37:12	Приходная накладная 000000012 от 0…	3	Сканер Epson Perfection V19	Сканер Epson Perfection V19	5
− 08.04.2023 13:39:12	Ввод техники в работу 000000006 от 0…	1	ПК RTX 4090 i3-12100 32Гб 1 200W SS…	ПК RTX 4090 i3-12100 32Гб 1 200W …	1
− 08.04.2023 13:39:12	Ввод техники в работу 000000006 от 0…	2	Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400	1
− 08.04.2023 13:39:12	Ввод техники в работу 000000006 от 0…	3	Принтер лазерный HP LaserJet Pro M4…	Принтер лазерный HP LaserJet Pro …	1
− 08.04.2023 13:39:12	Ввод техники в работу 000000006 от 0…	4	Сканер Epson Perfection V19	Сканер Epson Perfection V19	1
+ 03.05.2023 12:00:00	Приходная накладная 000000010 от 0…	1	Микрофон HyperX QuadCast S	Микрофон HyperX QuadCast S	10

Рисунок 30 – Регистр накопления «Товары на складе»

2. Регистр накопления «Техника в работе» предназначен для накопления информации о технике в работе. Является регистратором для всех документов кроме документа «Приходная накладная». Регистр накопления представлен на рисунке 31.



Период	Регистратор	Номер строки	Номенклатура	Характеристики	Рабочее место	Номер/лицо	Количество
+ 25.05.2022 19:39:59	Ввод техники в работ…	1	Клавиатура беспроводная Re…	Беспроводная, чер…	Материально ответ…	Краснов Сергей Се…	1
+ 25.05.2022 19:39:59	Ввод техники в работ…	2	Мышь беспроводная Logitec…	Беспроводная, бел…	Материально ответ…	Краснов Сергей Се…	1
− 30.05.2022 12:00:00	Передача в ремонт 0…	1	Клавиатура беспроводная Re…	Беспроводная, чер…	Материально ответ…	Краснов Сергей Се…	1
+ 08.04.2023 13:39:12	Ввод техники в работ…	1	ПК RTX 4090 i3-12100 32Гб 1 …	ПК RTX 4090 i3-121…	Отдел	1	1
+ 08.04.2023 13:39:12	Ввод техники в работ…	2	Мышь беспроводная Logitec…	Беспроводная, бел…	Отдел	1	1
+ 08.04.2023 13:39:12	Ввод техники в работ…	3	Принтер лазерный HP LaserJ…	Принтер лазерный …	Отдел	1	1
+ 08.04.2023 13:39:12	Ввод техники в работ…	4	Сканер Epson Perfection V19	Сканер Epson Perfe…	Отдел	1	1
+ 09.05.2023 16:59:07	Ввод техники в работ…	1	ПК RTX 4090 R7 5800X 32Гб 1…	ПК RTX 4090 R7 58…	Офис	5	1
+ 09.05.2023 16:59:07	Ввод техники в работ…	2	23.8" Монитор Xiaomi Mi Desk…	1920x1080@60 Гц, …	Офис	5	2
+ 09.05.2023 16:59:07	Ввод техники в работ…	3	Мышь беспроводная Logitec…	Беспроводная, бел…	Офис	5	1
+ 09.05.2023 16:59:07	Ввод техники в работ…	4	Проводные наушники Fifine H6	Проводные наушни…	Офис	5	1
+ 09.05.2023 16:59:07	Ввод техники в работ…	5	Микрофон HyperX QuadCast S	Микрофон HyperX …	Офис	5	1
− 18.05.2023 23:35:36	Передача в ремонт 0…	1	Сканер Epson Perfection V19	Сканер Epson Perfe…	Отдел	1	1
− 18.05.2023 23:35:36	Передача в ремонт 0…	2	Принтер лазерный HP LaserJ…	Принтер лазерный …	Отдел	1	1

Рисунок 31 – Регистр накопления «Техника в работе»

3. Регистр накопления «Списанная техника» предназначен для накопления информации о списанной технике. Является регистратором для документа «Списание техники». Регистр накопления представлен на рисунке 32.

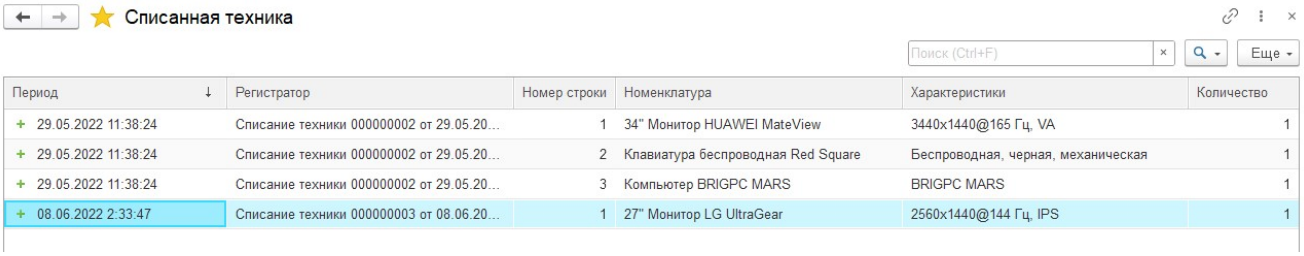


Рисунок 32 – Регистр накопления «Списанная техника»

4. Регистр накопления «Техника в ремонте» предназначен для накопления информации о технике в ремонте. Является регистратором для документов «Передача в ремонт» и «Приход с ремонта». Регистр накопления представлен на рисунке 33.

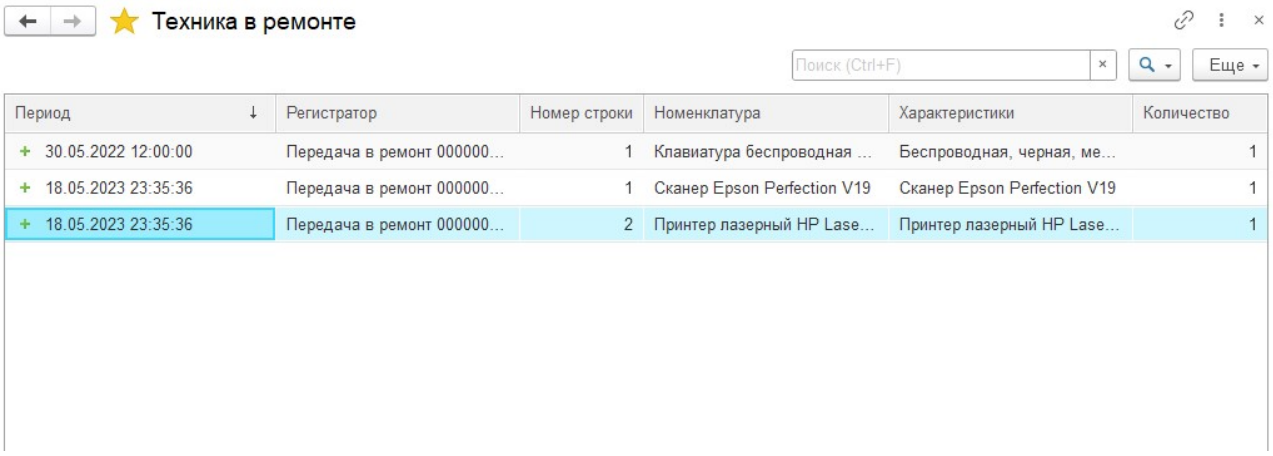


Рисунок 33 – Регистр накопления «Техника в ремонте»

Общие формы. Они не привязаны к конкретному объекту и могут использоваться разными объектами для различных задач [23]. Для разрабатываемой конфигурации были созданы 2 общие формы.

1. Общая форма «Подбор рабочего места» предназначена для задания параметров, по которым будет подбираться рабочее место пользователя. Общая форма представлена на рисунке 34.

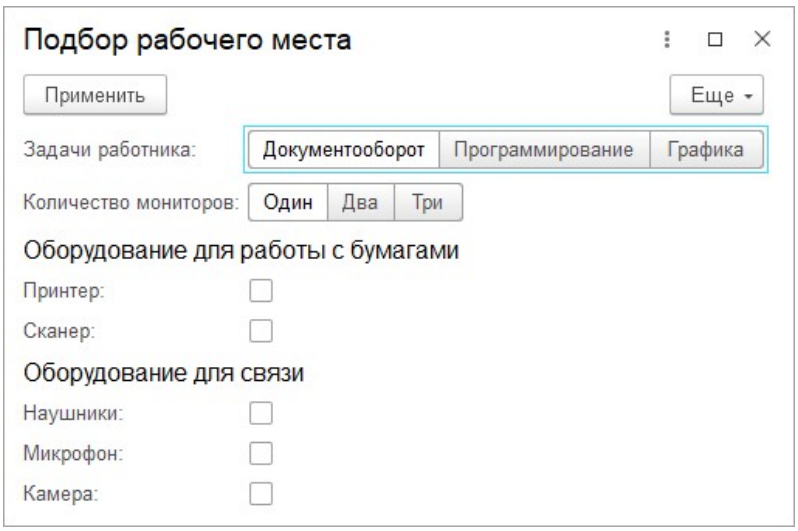


Рисунок 34 – Общая форма «Подбор рабочего места»

2. Общая форма «Конфигуратор ПК» предназначена для ручной сборки конфигурации ПК. Общая форма представлена на рисунке 35.

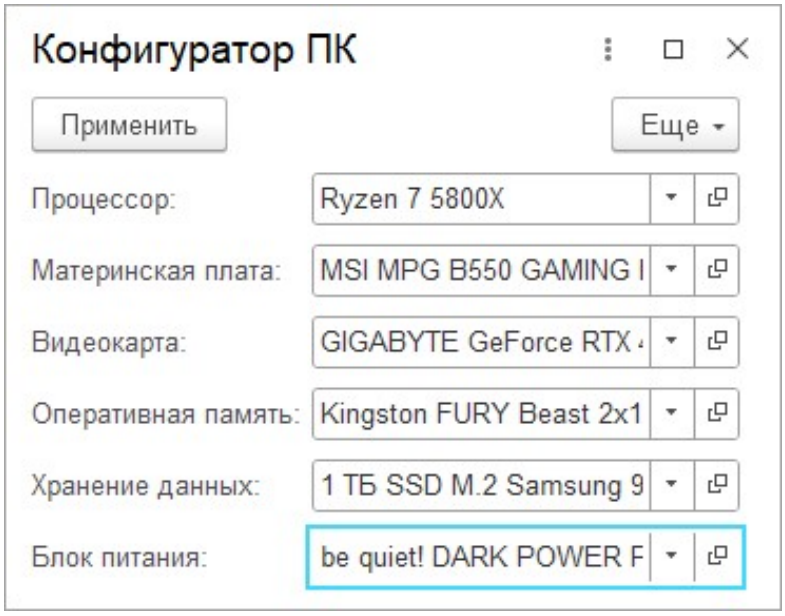
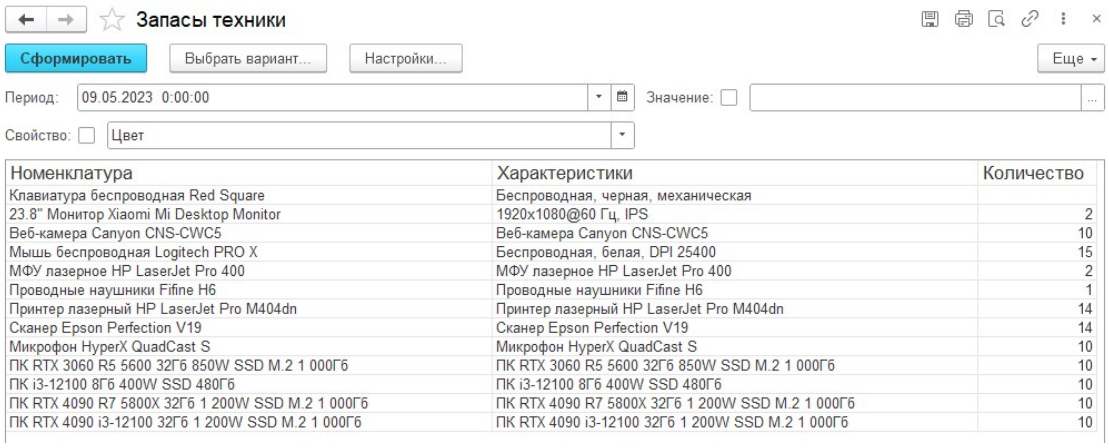


Рисунок 35 – Общая форма «Конфигуратор ПК»

Отчеты. Они предназначены для получения сводных данных на основе накопленной информации [24]. Для разрабатываемой конфигурации были созданы 4 отчета.

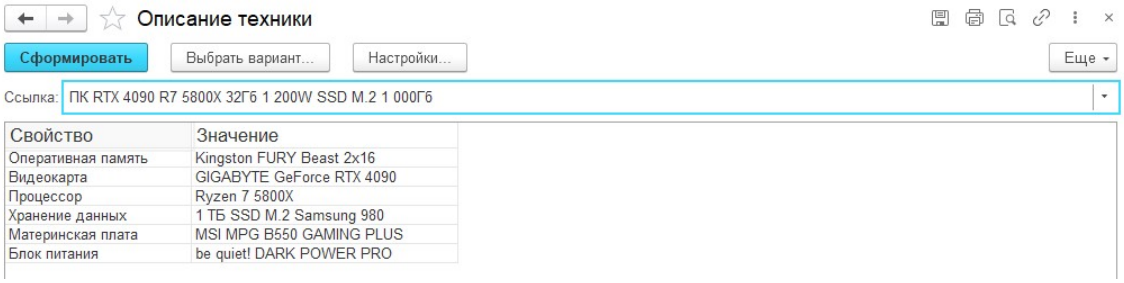
1. Отчет «Запасы техники» отображает всю имеющую технику по заданным параметрам. Отчет представлен на рисунке 36.



Номенклатура	Характеристики	Количество
Клавиатура беспроводная Red Square	Беспроводная, черная, механическая	
23.8" Монитор Xiaomi Mi Desktop Monitor	1920x1080@60 Гц, IPS	2
Веб-камера Canyon CNS-CWC5	Веб-камера Canyon CNS-CWC5	10
Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400	15
МФУ лазерное HP LaserJet Pro 400	МФУ лазерное HP LaserJet Pro 400	2
Проводные наушники Fifine H6	Проводные наушники Fifine H6	1
Принтер лазерный HP LaserJet Pro M404dn	Принтер лазерный HP LaserJet Pro M404dn	14
Сканер Epson Perfection V19	Сканер Epson Perfection V19	14
Микрофон HyperX QuadCast S	Микрофон HyperX QuadCast S	10
ПК RTX 3060 R5 5600 32Гб 850W SSD M.2 1 000Гб	ПК RTX 3060 R5 5600 32Гб 850W SSD M.2 1 000Гб	10
ПК i3-12100 8Гб 400W SSD 480Гб	ПК i3-12100 8Гб 400W SSD 480Гб	10
ПК RTX 4090 R7 5800X 32Гб 1 200W SSD M.2 1 000Гб	ПК RTX 4090 R7 5800X 32Гб 1 200W SSD M.2 1 000Гб	10
ПК RTX 4090 i3-12100 32Гб 1 200W SSD M.2 1 000Гб	ПК RTX 4090 i3-12100 32Гб 1 200W SSD M.2 1 000Гб	10

Рисунок 36 – Отчёт «Запасы техники»

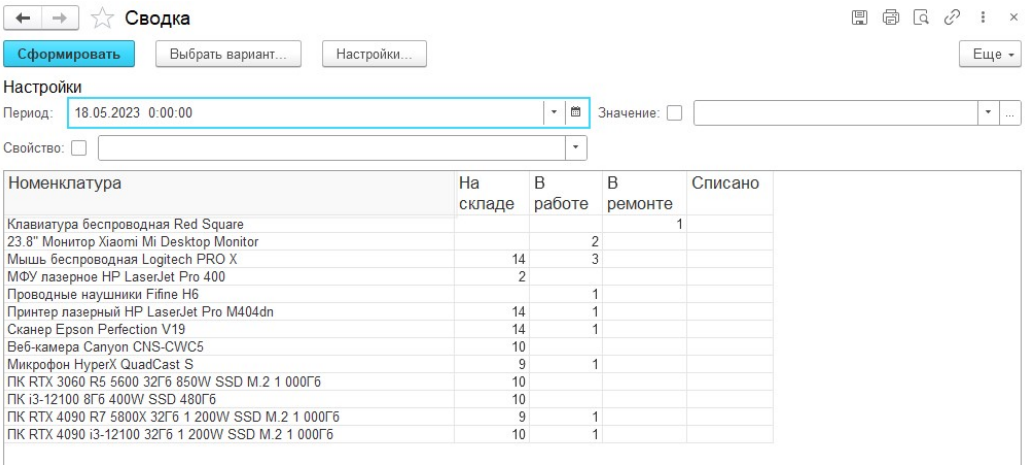
2. Отчет «Описание техники» отображает характеристики выбранной техники. Отчет представлен на рисунке 37.



Свойство	Значение
Оперативная память	Kingston FURY Beast 2x16
Видеокарта	GIGABYTE GeForce RTX 4090
Процессор	Ryzen 7 5800X
Хранение данных	1 ТБ SSD M.2 Samsung 980
Материнская плата	MSI MPG B550 GAMING PLUS
Блок питания	be quiet! DARK POWER PRO

Рисунок 37 – Отчёт «Описание техники»

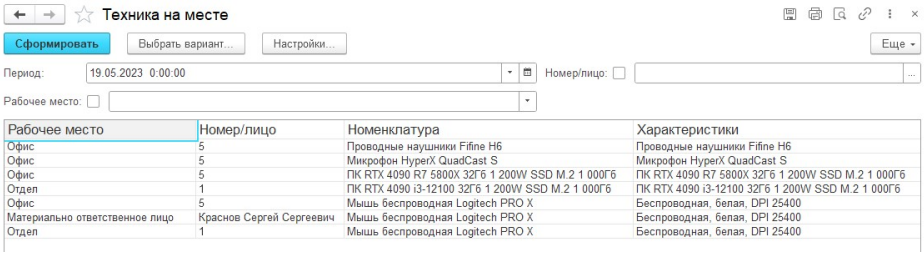
3. Отчет «Сводка» отображает информацию о наличии техники на местах, в ремонте, на складе, а также списанную. Отчет представлен на рисунке 38.



Номенклатура	На складе	В работе	В ремонте	Списано
Клавиатура беспроводная Red Square			1	
23.8" Монитор Xiaomi Mi Desktop Monitor		2		
Мышь беспроводная Logitech PRO X	14	3		
МФУ лазерное HP LaserJet Pro 400	2			
Проводные наушники Fifine H6		1		
Принтер лазерный HP LaserJet Pro M404dn	14	1		
Сканер Epson Perfection V19	14	1		
Веб-камера Canyon CNS-CWC5	10			
Микрофон HyperX QuadCast S	9	1		
ПК RTX 3060 R5 5600 32Гб 850W SSD M.2 1 000Гб	10			
ПК i3-12100 8Гб 400W SSD 480Гб	10			
ПК RTX 4090 R7 5800X 32Гб 1 200W SSD M.2 1 000Гб	9	1		
ПК RTX 4090 i3-12100 32Гб 1 200W SSD M.2 1 000Гб	10	1		

Рисунок 38 – Отчёт «Сводка»

4. Отчет «Техника на месте» отображает информацию техники на выбранном месте. Отчет представлен на рисунке 39.



Рабочее место	Номер/лицо	Номенклатура	Характеристики
Офис	5	Проводные наушники Fifine H6	Проводные наушники Fifine H6
Офис	5	Микрофон HyperX QuadCast S	Микрофон HyperX QuadCast S
Офис	5	ПК RTX 4090 R7 5800X 32Гб 1 200W SSD M.2 1 000Гб	ПК RTX 4090 R7 5800X 32Гб 1 200W SSD M.2 1 000Гб
Отдел	1	ПК RTX 4090 i3-12100 32Гб 1 200W SSD M.2 1 000Гб	ПК RTX 4090 i3-12100 32Гб 1 200W SSD M.2 1 000Гб
Офис	5	Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400
Материально ответственное лицо	Краснов Сергей Сергеевич	Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400
Отдел	1	Мышь беспроводная Logitech PRO X	Беспроводная, белая, DPI 25400

Рисунок 39 – Отчёт «Техника на месте»

Таким образом, разрабатываемая конфигурация состоит из 11 справочников, 7 документов, 5 перечислений, 1 плана видов характеристик, 1 регистра сведений, 4 регистров накопления, 2 общие формы и 4 отчета.

4.2 Реализация кода системы

Разработка приложений на платформе 1С: Предприятие 8.3 несколько отличается от процесса разработки программ на других языках программирования. Здесь разработчик в основном работает с графической средой, где создает объекты и задает им логику работу без написания кода. Платформа позволяет использовать удобные конструкторы для различных задач, будь то обработка проведения документов или формирование отчетов. Однако для реализации каких-либо сложных алгоритмов разработчик должен будет вручную писать процедуры и функции.

Описание процедур и функций системы представлено в таблице 6.

Таблица 6 – Описание процедур и функций системы

Объект	Название функции	Краткое описание функции
Общий модуль «Комплектующие»	Вызвать ошибку(сообщение)	Вызывает исключение с указанным сообщением
	ПолучитьВидеокартаИнфо(имя)	Получает основную информацию указанной видеокарты
	ПолучитьПроцессорИнфо(имя)	Получает основную информацию указанного процессора
	ПолучитьХранениеДанныхИнфо(имя)	Получает основную информацию указанного устройства хранения данных
	ПолучитьМатПлатаИнфо(имя)	Получает основную информацию указанной материнской платы
	СоздатьХарактеристику (характеристика, свойство, значение)	Создает запись характеристики в регистре сведений

Продолжение таблицы 6

Объект	Название функции	Краткое описание функции
Общая форма «Подбор рабочего места»	Применить(Команда)	Закрывает форму с пометкой о выполнении
	ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)	Устанавливает начальные значения реквизитов на форме
Общая форма «Конфигуратор ПК»	Применить(Команда)	Закрывает форму если комплектующие совместимы
	ПриОткрытии(Отказ)	Устанавливает начальные значения реквизитов на форме
	ЗадатьТипПамяти(типПамяти)	Устанавливает отбор для реквизита «Оперативная память»
	ПроцессорПриИзменении()	Устанавливает отбор для реквизитов формы, чтобы в списке были только совместимые комплектующие
	МатеринскаяПлатаПриИзменении()	Устанавливает отбор для реквизитов формы, чтобы в списке были только совместимые комплектующие
	ЗадатьПараметры(параметры)	Создает настройки отбора с указанными параметрами
	ОчиститьПараметры()	Очищает параметры отбора
	СократитьНазваниеПроцессора(название)	Сокращает название процессора
	СократитьНазваниеВидеокарты(название)	Сокращает название видеокарты
	ПроверитьКонфигурациюПК()	Проверяет совместимость каждого компонента ПК
	ПриЗакрытииНаСервере()	Проверяет ПК и создает запись в справочнике «Номенклатура»
Документ «Приходная накладная» (модуль объекта)	ОбработкаПроведения(Отказ, Режим)	Создает записи в регистре накопления «Товары на складе»
Документ «Ввод техники в работу» (модуль объекта)	ОбработкаПроведения(Отказ, Режим)	Проверяет наличие техники на складе и создает записи в регистрах накопления «Товары на складе» и «Приход в работу»
Документ «Ввод техники в работу» (модуль формы)	РабочееМестоПриИзменении(Элемент)	Устанавливает тип и заголовок реквизита «НомерЛицо»
	ПодборРабочегоМеста(Команда)	Открывает форму «Подбор рабочего места»
	ПолучитьПодходящийПК(количествоЯдер)	Получает ПК с указанным количеством ядер
	ПодобратьРабочееМесто(Форма)	Обрабатывает данные с формы «Подбор рабочего места» и добавляет записи в табличную часть
	ПроверитьИДобавить(номенклатура, количество)	Проверяет наличие номенклатуры на складе
	ДобавитьСтроку(табЧасть, номенклатура, характеристики, количество)	Добавляет запись в табличную часть
	ПолучитьНоменклатуру(категория)	Получает номенклатуру указанной категории
	ПередЗаписью(Отказ, ПараметрыЗаписи)	Предлагает создать заявку для заказа, если на складе недостаточно техники
	СоздатьЗаявку(список, дата)	Создает заявку на заказ техники
	ПолучитьОстатокНоменклатуры(Момент, Номенклатура, Характеристики)	Получает количество номенклатуры на складе
	КонфигураторПК(Команда)	Открывает форму «Конфигуратор ПК»

Окончание таблицы 6

Объект	Название функции	Краткое описание функции
Документ «Списание техники» (модуль объекта)	ОбработкаПроведения(Отказ, Режим)	Проверяет наличие техники в работе и создает записи в регистрах накопления «Техника в работе» и «Списанная техника»
Документ «Списание техники» (модуль формы)	РабочееМестоПриИзменении(Элемент)	Устанавливает тип и заголовок реквизита «НомерЛицо»
Документ «Передача в ремонт» (модуль объекта)	ОбработкаПроведения(Отказ, Режим)	Проверяет наличие техники в работе и создает записи в регистрах накопления «Техника в работе» и «Техника в ремонте»
Документ «Передача в ремонт» (модуль формы)	РабочееМестоПриИзменении(Элемент)	Устанавливает тип и заголовок реквизита «НомерЛицо»
Документ «Приход с ремонта» (модуль объекта)	ОбработкаПроведения(Отказ, Режим)	Проверяет наличие техники в ремонте и создает записи в регистрах накопления «Техника в работе» и «Техника в ремонте»
Документ «Приход с ремонта» (модуль формы)	РабочееМестоПриИзменении(Элемент)	Устанавливает тип и заголовок реквизита «НомерЛицо»
Документ «Передача техники» (модуль объекта)	ОбработкаПроведения(Отказ, Режим)	Проверяет наличие техники в работе и создает записи в регистре накопления «Техника в работе»
Документ «Передача техники» (модуль формы)	РабочееМестоПриИзменении(Элемент)	Устанавливает тип и заголовок реквизита «НомерЛицо»

Исходные коды всех модулей конфигурации представлены в приложении Ж.

5 Тестирование документов конфигурации

Для тестирования документов конфигурации были использованы такие виды тестирования как тестирование методом чёрного ящика, тестирование метрик кода, а также автоматизирование тестирование.

5.1 Тестирование методом черного ящика

Тестирование методом чёрного ящика позволяет проверить функциональность программного обеспечения без информации об его внутреннем устройстве. Данный вид тестирования демонстрирует, что приложение выполняет все заявленные функции [25].

Тестирование черного ящика было проведено по методикам разделения на классы эквивалентности, а также построения функциональной диаграммы и таблицы решений.

5.1.1 Классы эквивалентности

У документа «Приходная накладная» имеется табличная часть, где необходимо указать номенклатуру и количество. Последнее поле можно проверить с помощью классов эквивалентности. Они позволяют значительно сократить количество тестов за счет разделения входных данных на группы эквивалентности [26].

В поле «Количество» должно быть введено целое число от 1 до 9999. Для этих требований были составлены классы эквивалентности, отображенные в таблице 7.

Таблица 7 – Классы эквивалентности

	Правильные КЭ	Неправильные КЭ
Допустимый диапазон	1-9999 (1)	Меньше 1(2), больше 9999 (3)
Содержание	Цифры (4)	Не цифры (5)

Графическое представление классов эквивалентности изображено на рисунках 40 и 41.

Рисунок 40 – Классы эквивалентности. Допустимый диапазон

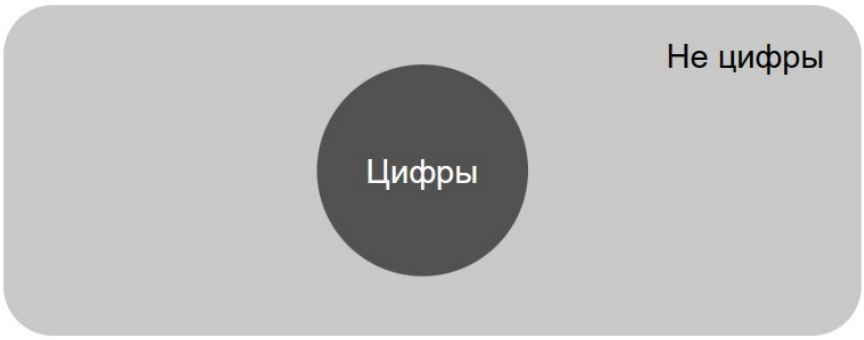


Рисунок 41 – Классы эквивалентности. Содержание

По данным классам эквивалентности были составлены тесты, представленные в таблице 8.

Таблица 8 – Тесты классов эквивалентности

№	КЭ	Входные данные	Выходные данные
1	Все правильные 1,4	666	Успешно.
2	Только один неправильный 3,4	6666666666	Ошибка: число должно не превышать 9999.
3	2,4	0	Ошибка: число должно быть больше 1.
4	1,5	Привет	Ошибка: должно быть число

Таким образом, достаточно проверить лишь одно значение из каждой группы эквивалентности, чтобы удостовериться в правильной работе поля.

5.1.2 Функциональная диаграмма и таблица решений

Для тестирования проведения документа «Приходная накладная» была составлена таблица причин и следствий, представленная в таблице 9.

Таблица 9 – Таблица причин и следствий

П1	Техника выбрана
П2	Количество введено
П3	Количество в диапазоне от 1 до 9999
С1	Сообщение: «Необходимо выбрать технику»
С2	Сообщение: «Необходимо указать количество»
С3	Сообщение: «Количество должно быть в диапазоне от 1 до 9999»
С4	Сообщение: «Документ проведен»

Для успешного проведения документа необходимо выбрать технику из списка номенклатуры и ввести её количество. Визуальное представление причин и следствий изображено на функциональной диаграмме на рисунке 42.

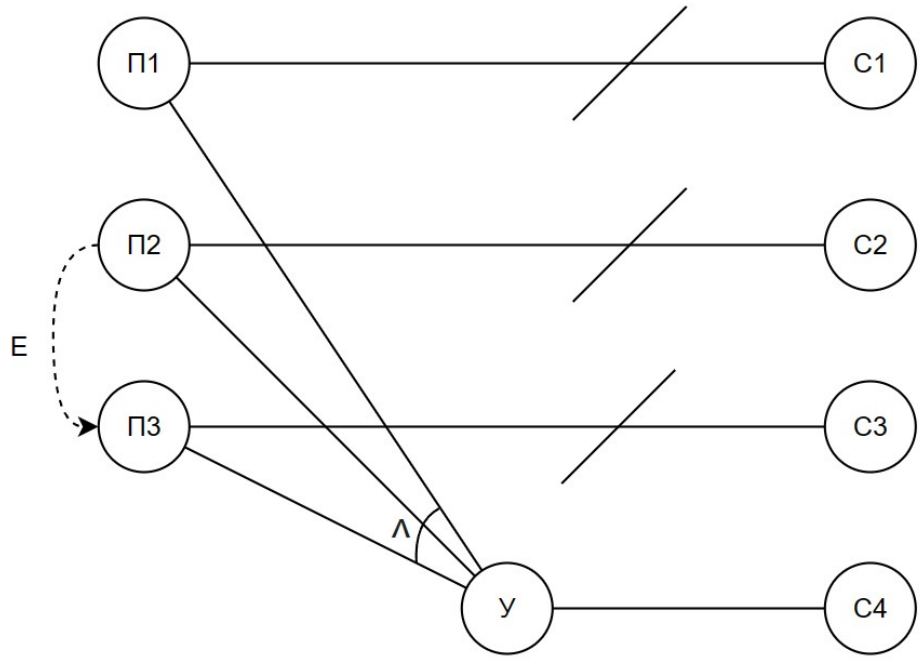


Рисунок 42 – Функциональная диаграмма причин и следствий

Для успешного проведения документа необходимо выполнение всех трех причин. Если хотя бы одна из причин не выполняется, то это приводит к сообщениям об ошибке. Важно учитывать, что третья причина не может быть выполнена без выполнения второй, так как невозможно проверить число, если оно не введено.

На основе таблицы и диаграммы была построена таблица решений (таблица 10), позволяющая составить тесты по всем возможным комбинациям входных данных.

Таблица 10 – Таблица решений

№	П1	П2	П3		С1	С2	С3	С4
1	1	1	1		0	0	0	1
2	1	1	0		0	0	1	0
3	1	0	0		0	1	0	0
4	0	1	0		1	0	1	0
5	0	1	1		1	0	0	0
6	0	0	0		1	1	0	0

По полученным данным были написаны тесты, позволяющие проверить каждую комбинацию причин. Тесты представлены в таблице 11.

Таблица 11 – Тесты таблицы решений

№	Входные данные	Выходные данные
1	Техника выбрана. Количество введено. Количество корректное.	Документ проведен
2	Техника выбрана. Количество введено. Количество некорректное.	Количество должно быть в диапазоне от 1 до 9999 (Рисунок И.1)
3	Техника выбрана. Количество не введено.	Необходимо указать количество (Рисунок И.2)
4	Техника не выбрана. Количество введено. Количество корректное.	Необходимо выбрать технику (Рисунок И.3)
5	Техника не выбрана. Количество введено. Количество некорректное.	Необходимо выбрать технику. Количество должно быть в диапазоне от 1 до 9999 (Рисунок И.4)
6	Техника не выбрана. Количество не введено.	Необходимо выбрать технику. Необходимо указать количество (Рисунок И.5)

Таким образом, можно сказать, что документ «Приходная накладная» работает корректно и при наличии ошибок сообщает пользователю об этом.

5.2 Тестирование метрики кода

Для оценки сложности функции была использованы метрика Холстеда. Данная метрика позволяет рассчитать длину, словарь, а также сложность программы на основе количества общих и уникальных операторов и операндов [27].

В качестве примера была проведена оценка сложности функции «ПроверитьКонфигурациюПК», предназначенная для проверки конфигурации ПК. Код функции представлен в листинге 1.

Листинг 1 – Функция «ПроверитьКонфигурациюПК»

```
Если Процессор.Сокет <> МатеринскаяПлата.Сокет Тогда
      Сообщение = "Материнская плата несовместима с процессором"+Символы.ПС+
      "Сокет процессора:"+Строка(Процессор.Сокет)+Символы.ПС+
      "Сокет материнской платы:"+Строка(МатеринскаяПлата.Сокет);
      Сообщить(Сообщение);
      возврат Истина;
КонецЕсли;

Если МатеринскаяПлата.ТипПамяти <> ОперативнаяПамять.ТипПамяти Тогда
      Сообщение = "Материнская плата не поддерживает тип памяти
"+Строка(ОперативнаяПамять.ТипПамяти);
      Сообщить(Сообщение);
      возврат Истина
КонецЕсли;

Если Процессор.ТипПамяти <> ОперативнаяПамять.ТипПамяти Тогда
      Сообщение = "Процессор не поддерживает тип памяти
"+Строка(ОперативнаяПамять.ТипПамяти);
      Сообщить(Сообщение);
      возврат Истина
КонецЕсли;

Если Видеокарта.Пустая() Тогда
      Если Процессор.Видеоядро = Ложь Тогда
            Сообщение = "В сборке отсутствует видеокарта или процессор с
видеоядром";
            Сообщить(Сообщение);
            возврат Истина
      КонецЕсли;

      tdp_gpu = 0;
      pin_gpu = 0;
Иначе
      tdp_gpu = Видеокарта.Энергопотребление;
      pin_gpu = Видеокарта.РазъемПитания;
КонецЕсли;

Если pin_gpu > БлокПитания.РазъемыGPU  Тогда
      Сообщение = "У блока питания нет подходящих кабелей для питания
Видеокарта"+Символы.ПС+
      "Видеокарте необходимо "+Строка(pin_gpu)+" PIN питание"+Символы.ПС+
      "Блок питания имеет только "+Строка(БлокПитания.РазъемыGPU)+" PIN кабели";
      Сообщить(Сообщение);
      возврат Истина
КонецЕсли;

энергопотреблениеСистемы =
(tdp_gpu+Процессор.Энергопотребление+ХранениеДанных.Энергопотребление)*1.12;
Если энергопотреблениеСистемы > БлокПитания.Мощность Тогда
      Сообщение = "Потребляемая мощность системы:
"+Строка(энергопотреблениеСистемы)+" Вт"+Символы.ПС+
      "Мощность блока питания: "+Строка(БлокПитания.Мощность)+" Вт";
      Сообщить(Сообщение);
      возврат Истина
КонецЕсли;

возврат Ложь;
```

Для оценки сложности процедуры по метрике Холстеда, необходимо посчитать количество общих и уникальных операторов и операндов. Операторы представлены в таблице 12, операнды – в таблице 13.

Таблица 12 – Операторы процедуры

Операторы	Число вхождений
=	12
<>	3
>	2
.	26
Если / КонецЕсли	7
;	32
+	24
*	1
Строка	8
Сообщить	6
возврат	7
Пустая	1
n_1= 12	N_1= 129

Таблица 13 – Операнды процедуры

Операнды	Число вхождений
Процессор	5
МатеринскаяПлата	3
ОперативнаяПамять	4
Видеокарта	3
БлокПитания	4
ХранениеДанных	1
Сокет	4
ТипПамяти	6
Видеоядро	1
Энергопотребление	3
РазъемПитания	1
РазъемыGPU	2
Мощность	2
Символы.ПС	5
Сообщение	12
Текст	15
Истина	6
Ложь	2
tdp_gpu	3
pin_gpu	4
1.12	1
0	2
энергопотреблениеСистемы	3
n_2= 23	N_2= 92

Подсчитав количество операторов и операндов, были рассчитаны длина, словарь, объем и сложность функции.

Длина функции рассчитывается по формуле:

$$N = N_1 + N_2 = 129 + 92 = 221,$$

где N – длина функции;

N_1 – количество использований операторов;

N_2 – количество использований операндов.

Словарь функции рассчитывается по формуле:

$$n = n_1 + n_2 = 12 + 23 = 35,$$

где n – словарь функции;

n_1 – количество операторов;

n_2 – количество операндов.

Объем функции рассчитывается по формуле:

$$V = N \cdot \log_2 n = 221 \cdot 5{,}129 = 1133{,}5,$$

где V – объем функции;

N – длина функции;

n – словарь функции.

Сложность функции рассчитывается по формуле:

$$HDff = (n_1 \div 2) \cdot (N_2 \cdot n_2) = (12 \div 2) \cdot (92 \cdot 23) = 24.$$

где HDiff – сложность функции;

n_1 – количество операторов;

n_2 – количество операндов;

N_2 – количество использований операндов.

Таким образом, сложность функции приблизительно равна 24, поэтому её можно считать простой.

5.3 Автоматизированное тестирование

Автоматизированное тестирование проводилось с помощью системы сценарного тестирования решений на базе 1С: Предприятие – «Тестер». Данная система позволяет программировать и запускать сложные сценарии для

автоматических действий в приложении, имитируя последовательности операций пользователя [28].

Был создан сценарий «ТестВводТехникиВРаботу» и проведена запись действий пользователя для создания, заполнения и проведения документа «Ввод техники в работу». После окончания записи был сгенерирован код для тестирования, представленный в листинге 2.

Листинг 2 – Сгенерированный код тестирования документа

```
Подключить ();

Здесь ( "Сводка" );
Меню ( "Панель разделов / Учет техники" );
Меню ( "Панель функций текущего раздела / Ввод техники в работу" );

Здесь ( "Ввод техники в работу" );
Нажать ( "!ФормаСоздать" );

Здесь ( "Ввод техники в работу (создание)" );
Подобрать ( "!РабочееМесто", "Отдел" );
Установить ( "!НомерЛицо", "1" );
Нажать ( "!ФормаПодборРабочегоМеста" );

Здесь ( "Подбор рабочего места" );
Установить ( "!ЗадачиРаботника", "Графика" );
Установить ( "!КоличествоМониторов", "Три" );
Нажать ( "!Применить" );

Здесь ( "Ввод техники в работу (создание) *" );
Нажать ( "!ФормаПровестиИЗакрыть" );
```

На основе сгенерированного кода, была написана процедура «СоздатьВводТехникиВРаботу». В результате сценарий приобрел вид, представленный в листинге 3.

Листинг 3 – Итоговый код сценария

```
Подключить ();
ЗакрытьВсё ();
СоздатьВводТехникиВРаботу();

Процедура СоздатьВводТехникиВРаботу()
      Здесь ( "Сводка" );
      Меню ( "Панель разделов / Учет техники" );
      Меню ( "Панель функций текущего раздела / Ввод техники в работу" );

      Здесь ( "Ввод техники в работу" );
      Нажать ( "!ФормаСоздать" );

      Здесь ( "Ввод техники в работу (создание)" );
```

```
        Подобрать ( "!РабочееМесто", "Отдел" );
        Установить ( "!НомерЛицо", "1" );
        Нажать ( "!ФормаПодборРабочегоМеста" );

        Здесь ( "Подбор рабочего места" );
        Установить ( "!ЗадачиРаботника", "Графика" );
        Установить ( "!КоличествоМониторов", "Три" );
        Нажать ( "!Применить" );

        Здесь ( "Ввод техники в работу (создание) *" );
        Нажать ( "!ФормаПровестиИЗакрыть" );
КонецПроцедуры
```

При запуске сценария тестер автоматически воспроизвел ранее записанные действия и отобразил результат, изображенный на рисунке 43.

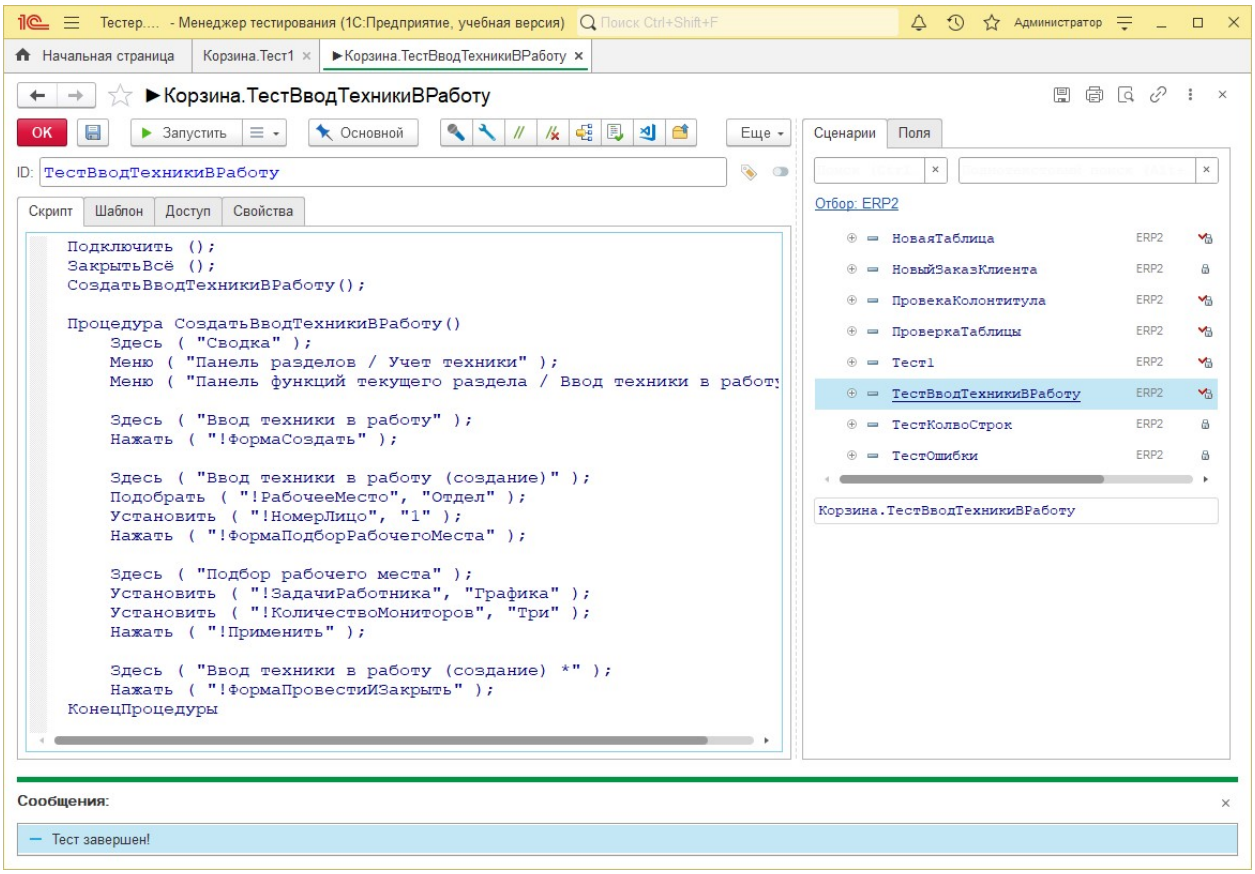


Рисунок 43 – Результат выполнения сценария

Таким образом, автоматизированное тестирование показало, что обработка проведения документа «Ввод техники в работу» работает корректно.

Заключение

В процессе выполнения выпускной квалификационной работы был проведен анализ предметной области, а также поиск программ аналогов, в ходе чего была подтверждена необходимость системы подбора рабочего пользователя ПК. Исходя из этого, были составлены требования и выбраны средства для проектирования и разработки. Также было проведено планирование и расчет бюджета на разработку программного средства.

Было проведено проектирование информационной системы, заключающееся в проектировании диаграмм UML, построении логической модели базы данных и создания макета интерфейса приложения.

Далее была проведена разработка системы, используя средства платформы 1С: Предприятие 8.3. Было проведено тестирование методами черного ящика, метрикой Холстеда и автоматизированное тестирование, показавшее, что программа работает корректно.

Таким образом, в результате выполнения выпускной квалификационной работы была разработана система подбора оптимальной комплектации рабочего места пользователя ПК с учетом его профессиональной деятельности, которая позволяет оптимизировать процесс комплектации рабочего места на предприятии.

Список использованных источников

1. Какие виды бухгалтерского учета бывают? // 1Советник. [Электронный ресурс] URL: https://sovetnik1c.ru/info/articles/bukhgalterskiy-uchet/kakie-vidy-bukhgalterskogo-ucheta-byvayut-/ (дата обращения: 25.04.2023)

2. 1С:Управление нашей фирмой // 1С:Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/small.biz/cena-1s-upravlenie-nashey-firmoy/ (дата обращения: 25.04.2023)

3. Все возможности управления бизнесом // МойСклад. [Электронный ресурс] URL: https://www.moysklad.ru/vozmozhnosti/ (дата обращения: 25.04.2023)

4. Программа для склада БИТ.WMS // первыйБит. [Электронный ресурс] URL: https://penza.1cbit.ru/1csoft/bit-wms/ (дата обращения: 25.04.2023)

5. Visual Paradigm Community Edition – A Free UML Tool // Visual Paradigm. [Электронный ресурс] URL: https://www.visual-paradigm.com/solution/freeumltool/ (дата обращения: 25.04.2023)

6. Балашова, И.Ю. Современные информационные технологии в проектировании программных средств и комплексов / И.Ю. Балашова, Д.В. Такташкин, под ред. П.П. Макарычев – Пенза: Изд-во ПГУ, 2019. – 121 с.

7. Модели жизненного цикла, принципы и методологии разработки программного обеспечения // Evergreens. [Электронный ресурс] URL: https://evergreens.com.ua/ru/articles/software-development-metodologies.html. (дата обращения: 25.04.2023)

8. Что такое диаграммы Ганта? // ATLASSIAN. [Электронный ресурс] URL: https://www.atlassian.com/ru/agile/project-management/gantt-chart (дата обращения: 27.04.2023)

9. Проектирование информационных систем // КОМПЬЮТЕР ПРЕСС. [Электронный ресурс] URL: https://compress.ru/article.aspx?id=11764 (дата обращения: 30.04.2023)

10. Разновидности диаграмм UML // Lucidchart. [Электронный ресурс] URL: https://www.lucidchart.com/blog/ru/types-of-UML-diagrams (дата обращения: 30.04.2023)

11. Диаграмма деятельности и особенности ее построения// [Электронный ресурс] / НОУ ИНТУИТ. URL: https://intuit.ru/studies/courses/32/32/lecture/1020 (дата обращения: 30.04.2023)

12. Диаграмма кооперации // НОУ ИНТУИТ. [Электронный ресурс] URL: https://intuit.ru/studies/courses/32/32/lecture/1012 (дата обращения: 04.05.2023)

13. Буч, Г. Язык UML. Руководство пользователя / Г. Буч, Д. Рамбо, И. Якобсон. – 2-е изд.: Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2006. – 496 с.

14. Data Modeling: An Overview // Toward Data Science. [Электронный ресурс] URL: https://towardsdatascience.com/data-modeling-an-overview-ed6165a27309 (дата обращения: 04.05.2023)

15. Руководство по межтабличным связям // Microsoft Поддержка. [Электронный ресурс] URL: https://support.microsoft.com/ru-ru/office/руководство-по-межтабличным-связям-30446197-4fbe-457b-b992-2f6fb812b58f#bm2 (дата обращения: 04.05.2023)

16. Элементы графической нотации диаграммы компонентов // НОУ ИНТУИТ. [Электронный ресурс] URL: https://intuit.ru/studies/courses/32/32/lecture/1022 (дата обращения: 08.05.2023)

17. Справочники // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/spravochniki/(дата обращения: 08.05.2023)

18. Перечисления // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/perechisleniya/(дата обращения: 08.05.2023)

19. Документы // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/dokumenty/(дата обращения: 08.05.2023)

20. План видов характеристик // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/plan-vidov-harakteristik/ (дата обращения: 09.05.2023)

21. Регистр сведений // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/registr-svedeniy/ (дата обращения: 09.05.2023)

22. Регистр накопления // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/registr-nakopleniya/ (дата обращения: 09.05.2023)

23. Объект конфигурации «Общая форма»// 1С: ИТС. [Электронный ресурс] URL: https://its.1c.ru/db/edtdoc/content/141/hdoc (дата обращения: 10.05.2023)

24. Отчет // 1С: Предприятие 8. [Электронный ресурс] URL: https://v8.1c.ru/platforma/otchet/ (дата обращения: 10.05.2023)

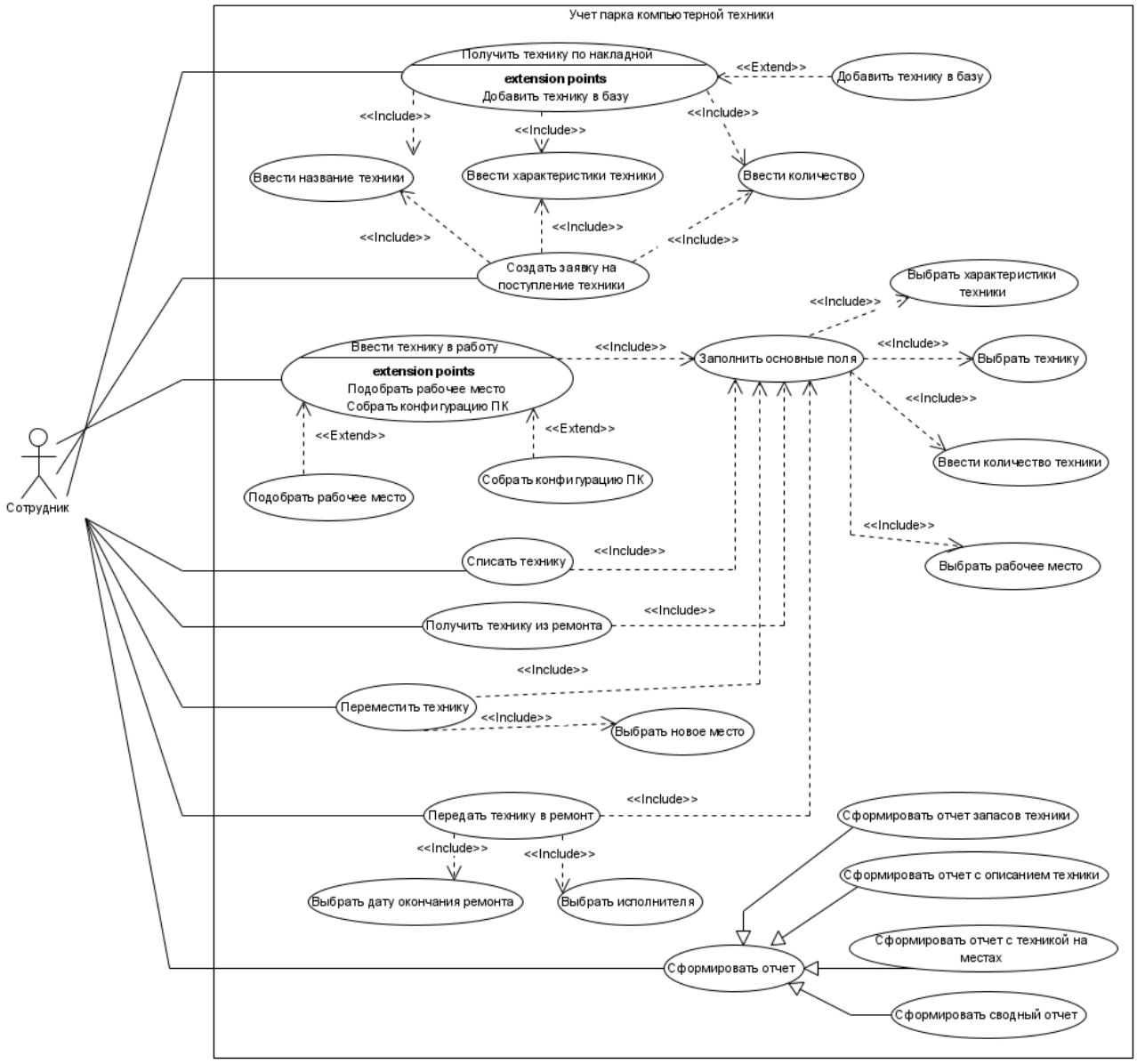
25. Особенности тестирования «черного ящика» // Лаборатория качества. [Электронный ресурс] URL: https://quality-lab.ru/blog/key-principles-of-black-box-testing/ (дата обращения: 15.05.2023)

26. Техники тест дизайна // Хекслет. [Электронный ресурс] URL: https://ru.hexlet.io/courses/qa-engineer-workflow/lessons/test-design-techniques/theory_unit (дата обращения: 15.05.2023)

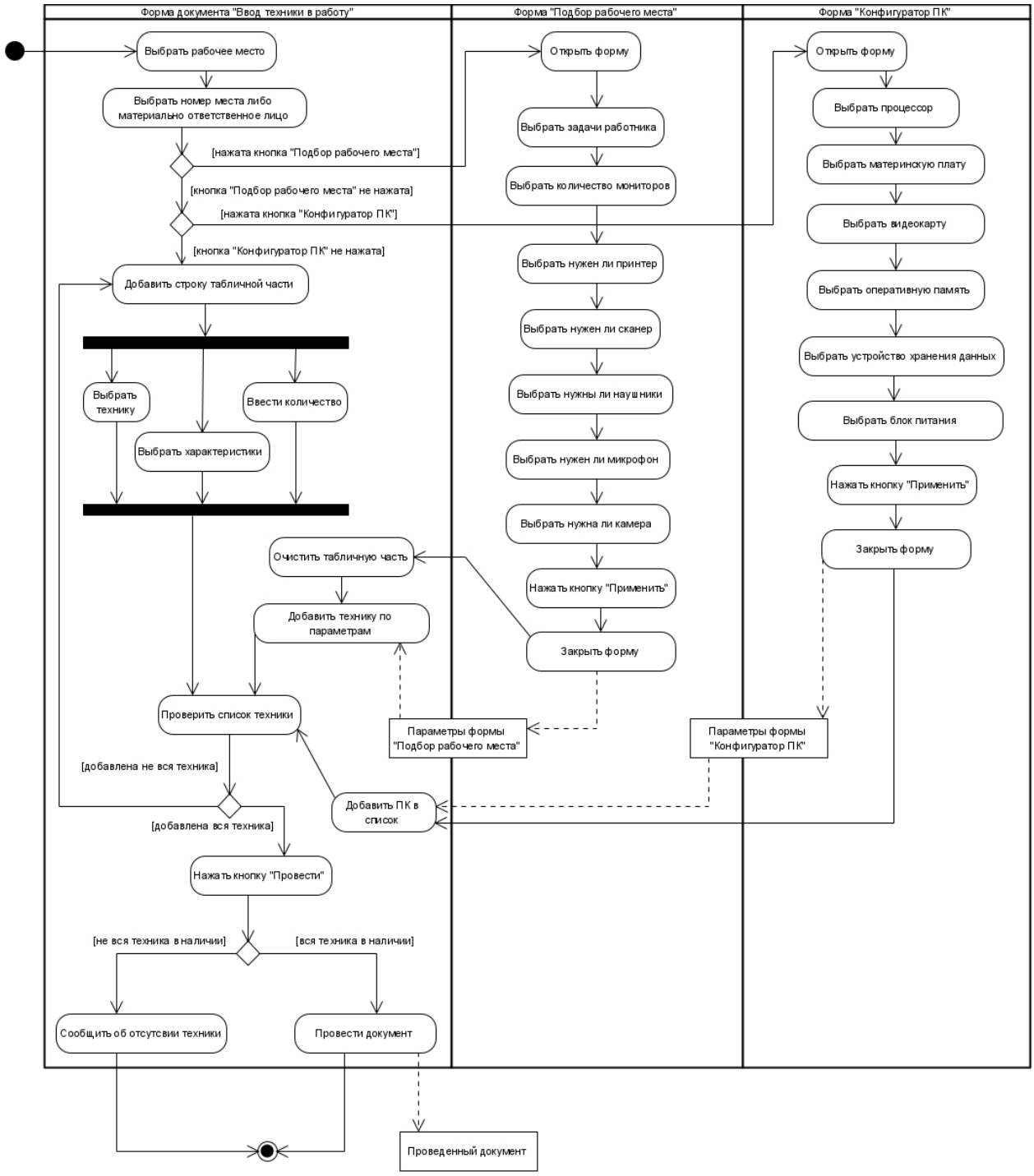
27. Метрики кода и практическая реализация по их сбору и анализу. Часть 1 – метрики // CNews клуб. [Электронный ресурс] URL: https://club.cnews.ru/blogs/entry/metriki_koda_i_prakticheskaya_realizatsiya_po_ih_sboru_i_analizu_chast_1_ (дата обращения: 15.05.2023)

28. Тестер // Github. [Электронный ресурс] URL: https://github.com/grumagargler/tester (дата обращения: 16.05.2023)

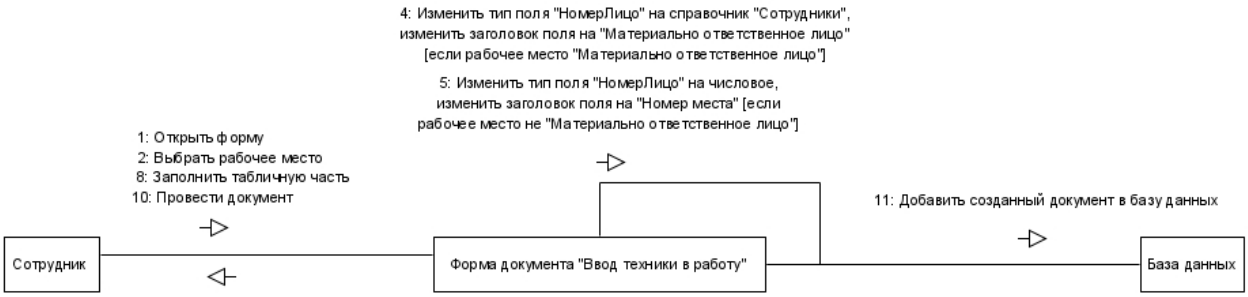
ПРИЛОЖЕНИЕ А
ДИАГРАММА ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ
(рекомендуемое)


Учет парка компьютерной техники
Получить технику по накладной
extension points
Добавить технику в базу
<<Extend>>
Добавить технику в базу
<<Include>>
<<Include>>
<<Include>>
Ввести название техники
Ввести характеристики техники
Ввести количество
<<Include>>
<<Include>>
<<Include>>
Создать заявку на поступление техники
Выбрать характеристики техники
<<Include>>
Ввести технику в работу
extension points
Подобрать рабочее место
Собрать конфигурацию ПК
<<Include>>
Заполнить основные поля
<<Include>>
Выбрать технику
<<Include>>
Ввести количество техники
<<Include>>
Выбрать рабочее место
<<Extend>>
<<Extend>>
Подобрать рабочее место
Собрать конфигурацию ПК
Сотрудник
Списать технику
<<Include>>
Получить технику из ремонта
<<Include>>
<<Include>>
Переместить технику
<<Include>>
Выбрать новое место
Передать технику в ремонт
<<Include>>
<<Include>>
<<Include>>
Выбрать дату окончания ремонта
Выбрать исполнителя
Сформировать отчет запасов техники
Сформировать отчет с описанием техники
Сформировать отчет с техникой на местах
Сформировать отчет
Сформировать сводный отчет

ПРИЛОЖЕНИЕ Б
ДИАГРАММА ДЕЯТЕЛЬНОСТИ
(рекомендуемое)


Форма документа "Ввод техники в работу"
Форма "Подбор рабочего места"
Форма "Конфигуратор ПК"
Выбрать рабочее место
Выбрать номер места либо материально ответственное лицо
[нажата кнопка "Подбор рабочего места"]
[кнопка "Подбор рабочего места" не нажата]
[нажата кнопка "Конфигуратор ПК"]
[кнопка "Конфигуратор ПК" не нажата]
Добавить строку табличной части
Выбрать технику
Ввести количество
Выбрать характеристики
Очистить табличную часть
Добавить технику по параметрам
Проверить список техники
[добавлена не вся техника]
[добавлена вся техника]
Нажать кнопку "Провести"
[не вся техника в наличии]
[вся техника в наличии]
Сообщить об отсутсвии техники
Провести документ
Проведенный документ
Открыть форму
Выбрать задачи работника
Выбрать количество мониторов
Выбрать нужен ли принтер
Выбрать нужен ли сканер
Выбрать нужны ли наушники
Выбрать нужен ли микрофон
Выбрать нужна ли камера
Нажать кнопку "Применить"
Закрыть форму
Параметры формы "Подбор рабочего места"
Открыть форму
Выбрать процессор
Выбрать материнскую плату
Выбрать видеокарту
Выбрать оперативную память
Выбрать устройство хранения данных
Выбрать блок питания
Нажать кнопку "Применить"
Закрыть форму
Параметры формы "Конфигуратор ПК"
Добавить ПК в список

ПРИЛОЖЕНИЕ В
ДИАГРАММА КООПЕРАЦИИ
(рекомендуемое)



3: Отобразить выбранное рабочее место

6: Отобразить заголовок поля "НомеЛицо" как "Материально ответственное лицо"
[если рабочее место "Материально ответственное лицо"]

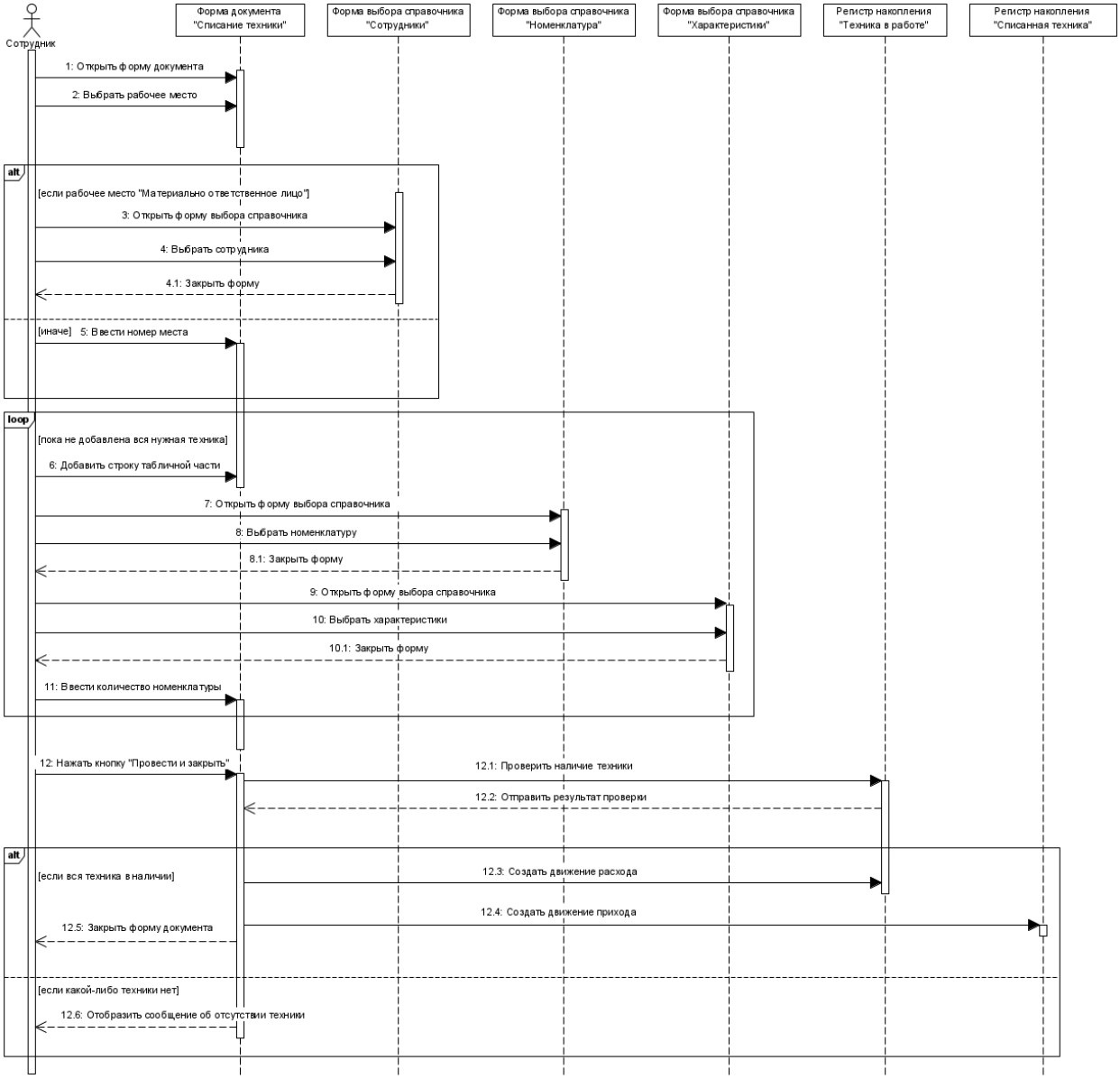
7: Отобразить заголовок поля "НомеЛицо" как "Номер места"
[если рабочее место не "Материально ответственное лицо"]

9: Отобразить заполненную табличную часть

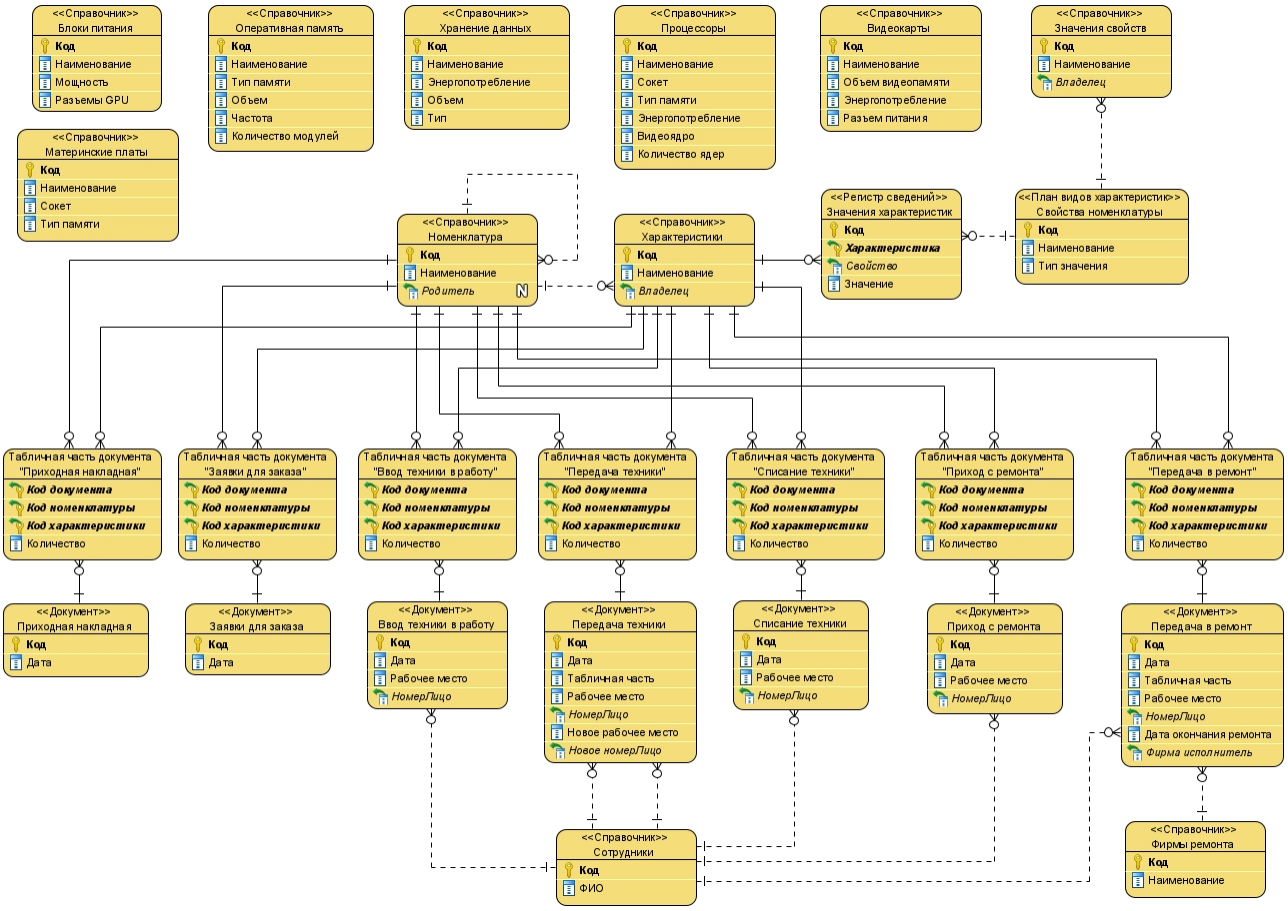
12: Закрыть форму

13: Отобразить сообщение об успешном проведении документа

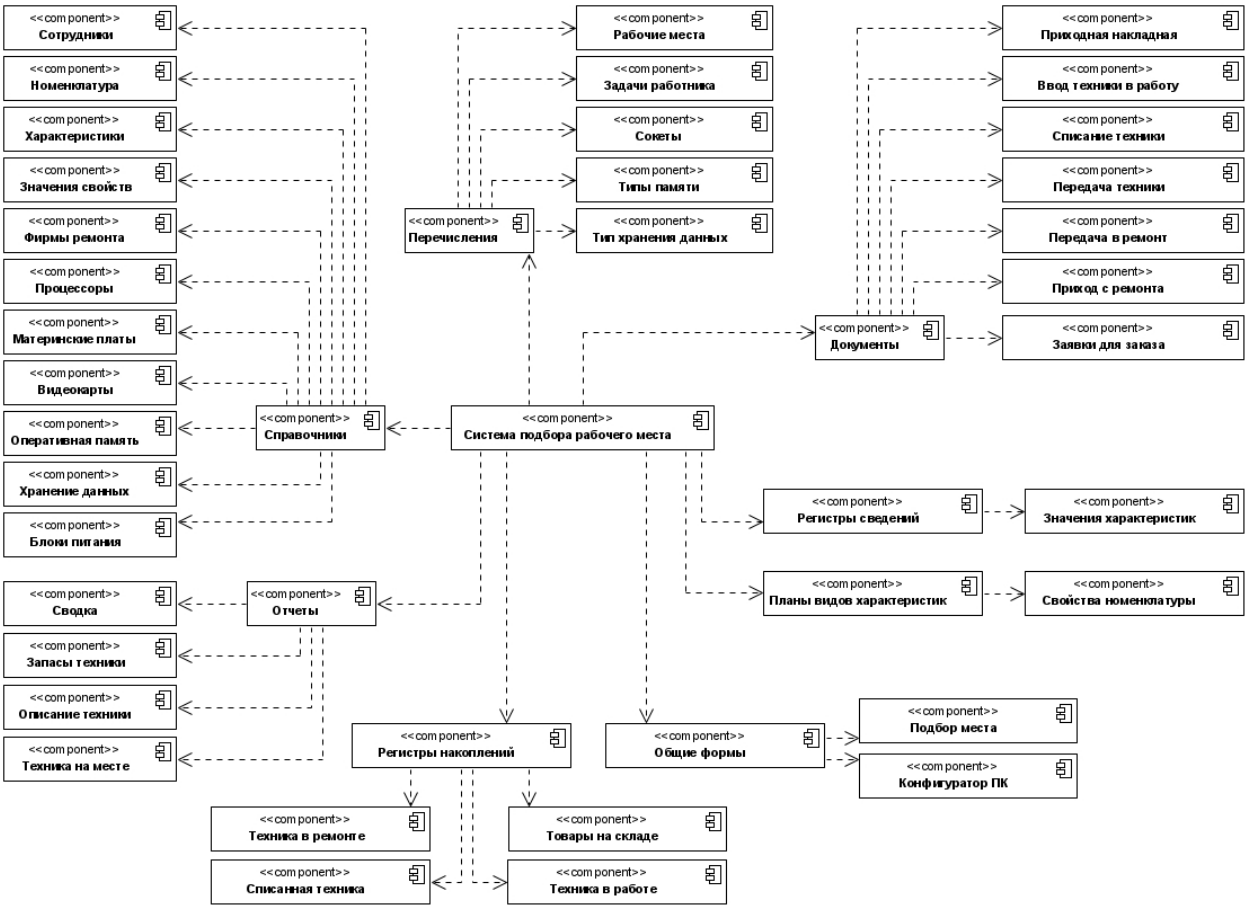
ПРИЛОЖЕНИЕ Г
ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТИ
(рекомендуемое)


Сотрудник
Форма документа "Списание техники"
Форма выбора справочника "Сотрудники"
Форма выбора справочника "Номенклатура"
Форма выбора справочника "Характеристики"
Регистр накопления "Техника в работе"
Регистр накопления "Списанная техника"
1: Открыть форму документа
2: Выбрать рабочее место
alt
[если рабочее место "Материально ответственное лицо"]
3: Открыть форму выбора справочника
4: Выбрать сотрудника
4.1: Закрыть форму
[иначе] 5: Ввести номер места
loop
[пока не добавлена вся нужная техника]
6: Добавить строку табличной части
7: Открыть форму выбора справочника
8: Выбрать номенклатуру
8.1: Закрыть форму
9: Открыть форму выбора справочника
10: Выбрать характеристики
10.1: Закрыть форму
11: Ввести количество номенклатуры
12: Нажать кнопку "Провести и закрыть"
12.1: Проверить наличие техники
12.2: Отправить результат проверки
alt
[если вся техника в наличии]
12.3: Создать движение расхода
12.4: Создать движение прихода
12.5: Закрыть форму документа
[если какой-либо техники нет]
12.6: Отобразить сообщение об отсутствии техники

ПРИЛОЖЕНИЕ Д
ЛОГИЧЕСКАЯ МОДЕЛЬ БАЗЫ ДАННЫХ
(рекомендуемое)


<<Справочник>>
Блоки питания
Код
Наименование
Мощность
Разъемы GPU
<<Справочник>>
Оперативная память
Код
Наименование
Тип памяти
Объем
Частота
Количество модулей
<<Справочник>>
Хранение данных
Код
Наименование
Энергопотребление
Объем
Тип
<<Справочник>>
Процессоры
Код
Наименование
Сокет
Тип памяти
Энергопотребление
Видеоядро
Количество ядер
<<Справочник>>
Видеокарты
Код
Наименование
Объем видеопамяти
Энергопотребление
Разъем питания
<<Справочник>>
Значения свойств
Код
Наименование
Владелец
<<Справочник>>
Материнские платы
Код
Наименование
Сокет
Тип памяти
<<Справочник>>
Номенклатура
Код
Наименование
Родитель
<<Справочник>>
Характеристики
Код
Наименование
Владелец
<<Регистр сведений>>
Значения характеристик
Код
Характеристика
Свойство
Значение
<<План видов характеристик>>
Свойства номенклатуры
Код
Наименование
Тип значения
Табличная часть документа "Приходная накладная"
Код документа
Код номенклатуры
Код характеристики
Количество
Табличная часть документа "Заявки для заказа"
Код документа
Код номенклатуры
Код характеристики
Количество
Табличная часть документа "Ввод техники в работу"
Код документа
Код номенклатуры
Код характеристики
Количество
Табличная часть документа "Передача техники"
Код документа
Код номенклатуры
Код характеристики
Количество
Табличная часть документа "Списание техники"
Код документа
Код номенклатуры
Код характеристики
Количество
Табличная часть документа "Приход с ремонта"
Код документа
Код номенклатуры
Код характеристики
Количество
Табличная часть документа "Передача в ремонт"
Код документа
Код номенклатуры
Код характеристики
Количество
<<Документ>>
Приходная накладная
Код
Дата
<<Документ>>
Заявки для заказа
Код
Дата
<<Документ>>
Ввод техники в работу
Код
Дата
Рабочее место
НомерЛицо
<<Документ>>
Передача техники
Код
Дата
Табличная часть
Рабочее место
НомерЛицо
Новое рабочее место
Новое номерЛицо
<<Документ>>
Списание техники
Код
Дата
Рабочее место
НомерЛицо
<<Документ>>
Приход с ремонта
Код
Дата
Рабочее место
НомерЛицо
<<Документ>>
Передача в ремонт
Код
Дата
Табличная часть
Рабочее место
НомерЛицо
Дата окончания ремонта
Фирма исполнитель
<<Справочник>>
Сотрудники
Код
ФИО
<<Справочник>>
Фирмы ремонта
Код
Наименование

ПРИЛОЖЕНИЕ Е
ДИАГРАММА КОМПОНЕНТОВ
(рекомендуемое)


<<component>>
Сотрудники
<<component>>
Номенклатура
<<component>>
Характеристики
<<component>>
Значения свойств
<<component>>
Фирмы ремонта
<<component>>
Процессоры
<<component>>
Материнские платы
<<component>>
Видеокарты
<<component>>
Оперативная память
<<component>>
Хранение данных
<<component>>
Блоки питания
<<component>>
Рабочие места
<<component>>
Задачи работника
<<component>>
Сокеты
<<component>>
Типы памяти
<<component>>
Перечисления
<<component>>
Тип хранения данных
<<component>>
Приходная накладная
<<component>>
Ввод техники в работу
<<component>>
Списание техники
<<component>>
Передача техники
<<component>>
Передача в ремонт
<<component>>
Приход с ремонта
<<component>>
Документы
<<component>>
Заявки для заказа
<<component>>
Справочники
<<component>>
Система подбора рабочего места
<<component>>
Регистры сведений
<<component>>
Значения характеристик
<<component>>
Планы видов характеристик
<<component>>
Свойства номенклатуры
<<component>>
Сводка
<<component>>
Запасы техники
<<component>>
Описание техники
<<component>>
Техника на месте
<<component>>
Отчеты
<<component>>
Регистры накоплений
<<component>>
Общие формы
<<component>>
Подбор места
<<component>>
Конфигуратор ПК
<<component>>
Техника в ремонте
<<component>>
Списанная техника
<<component>>
Товары на складе
<<component>>
Техника в работе

ПРИЛОЖЕНИЕ Ж
ЛИСТИНГ КОНФИГУРАЦИИ
(рекомендуемое)

Листинг Ж.1 – Модуль объекта «ПриходнаяНакладная»

```
Процедура ОбработкаПроведения(Отказ, Режим)
	Движения.ТоварыНаСкладе.Записывать = Истина;
	Для Каждого ТекСтрокаТовары Из Товары Цикл
		Движение = Движения.ТоварыНаСкладе.Добавить();
		Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
		Движение.Период = Дата;
		Движение.Номенклатура = ТекСтрокаТовары.Номенклатура;
		Движение.Характеристики = ТекСтрокаТовары.Характеристики;
		Движение.Количество = ТекСтрокаТовары.Количество;
	КонецЦикла;
КонецПроцедуры
```

Листинг Ж.2 – Модуль объекта «ВводТехникиВРаботу»

```
Процедура ОбработкаПроведения(Отказ, Режим)

	Запрос = Новый Запрос;
	Запрос.Текст =
		"ВЫБРАТЬ
		|	ВводТехникиВРаботуСписокТехники.Номенклатура КАК Номенклатура,
		|	СУММА(ВводТехникиВРаботуСписокТехники.Количество) КАК
Количество,
		|	ВводТехникиВРаботуСписокТехники.Характеристики КАК
Характеристики
		|ПОМЕСТИТЬ ТабДок
		|ИЗ
		|	Документ.ВводТехникиВРаботу.СписокТехники КАК
ВводТехникиВРаботуСписокТехники
		|		ВНУТРЕННЕЕ СОЕДИНЕНИЕ Документ.ВводТехникиВРаботу КАК
ВводТехникиВРаботу
		|		ПО ВводТехникиВРаботуСписокТехники.Ссылка =
ВводТехникиВРаботу.Ссылка
		|ГДЕ
		|	ВводТехникиВРаботу.Ссылка = &Ссылка
		|
		|СГРУППИРОВАТЬ ПО
		|	ВводТехникиВРаботуСписокТехники.Номенклатура,
		|	ВводТехникиВРаботуСписокТехники.Характеристики
		|;
		|

	|//////////////////////////////////////////////////////////////////////
///////
		|ВЫБРАТЬ
		|	ТабДок.Номенклатура КАК Номенклатура,
		|	ТабДок.Количество КАК Количество,
		|	ТабДок.Характеристики КАК Характеристики,
		|	ЕстьNULL(ТоварыНаСкладеОстатки.КоличествоОстаток,0) КАК
КоличествоОстаток
		|ИЗ
		|	ТабДок КАК ТабДок
		|		ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ТоварыНаСкладе.Остатки(
		|			&Момент,
		|			(Номенклатура, Характеристики) В
		|				(ВЫБРАТЬ
		|					ТабДок.Номенклатура КАК
Номенклатура,
```

```
            |                                        ТабДок.Характеристики КАК
Характеристики
            |                                  ИЗ
            |                                        ТабДок КАК ТабДок)) КАК
ТоварыНаСкладеОстатки
            |           ПО ТабДок.Номенклатура =
ТоварыНаСкладеОстатки.Номенклатура";

      Если Режим = РежимПроведенияДокумента.Оперативный Тогда
            Запрос.УстановитьПараметр("Момент", неопределено);
      Иначе
            Запрос.УстановитьПараметр("Момент", МоментВремени());
      КонецЕсли;
      Запрос.УстановитьПараметр("Ссылка", Ссылка);

      РезультатЗапроса = Запрос.Выполнить();

      Выборка = РезультатЗапроса.Выбрать();

      Движения.ТоварыНаСкладе.Записывать = Истина;
      Движения.ТехникаВРаботе.Записывать = Истина;
      Пока Выборка.Следующий() Цикл
            // Расход со склада
            Движение = Движения.ТоварыНаСкладе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.Количество = Выборка.Количество;

            Если Выборка.КоличествоОстаток < Выборка.Количество Тогда
                  //Отказ = Истина;
                  ВызватьОшибку("Не хватает товара '" + Выборка.Номенклатура +
"' в количестве "+ (Выборка.Количество-Выборка.КоличествоОстаток));
            КонецЕсли;

            // Приход в работу
            Движение = Движения.ТехникаВРаботе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.РабочееМесто = РабочееМесто;
            Движение.НомерЛицо = НомерЛицо;
            Движение.Количество = Выборка.Количество;

      КонецЦикла;

КонецПроцедуры
```

Листинг Ж.3 – Форма документа «ВводТехникиВРаботу»

```
&НаКлиенте
Процедура РабочееМестоПриИзменении(Элемент)
        Если Строка(Объект.РабочееМесто) = "Материально ответственное лицо"
Тогда
        Объект.НомерЛицо = Справочники.Сотрудники.ПустаяСсылка();
            Элементы.НомерЛицо.Заголовок = "Материально ответственное лицо"
    Иначе
        Объект.НомерЛицо = "";
            Элементы.НомерЛицо.Заголовок = "Номер места"
      КонецЕсли;
```

```
	Элементы.НомерЛицо.ВыбиратьТип = Ложь;
КонецПроцедуры


&НаКлиенте
Процедура ПодборРабочегоМеста(Команда)
	Форма = ПолучитьФорму("ОбщаяФорма.ПодборМеста");
	Форма.ОткрытьМодально();

	Если Форма.Выполнено Тогда
		ПодобратьРабочееМесто(Форма);
	КонецЕсли;
КонецПроцедуры

&НаСервере
Функция ПолучитьПодходящийПК(КоличествоЯдер)

	Запрос = Новый Запрос;
	Запрос.Текст =
		"ВЫБРАТЬ
		|	ЗначенияХарактеристик.Значение КАК Проц,
		|	Процессоры.КоличествоЯдер КАК КоличествоЯдер,
		|	ЗначенияХарактеристик.Характеристика.Владелец.Ссылка КАК
ССылкаПК,
		|	ЗначенияХарактеристик.Характеристика КАК Характеристика
		|ИЗ
		|	РегистрНакопления.ТоварыНаСкладе.Остатки КАК
ТоварыНаСкладеОстатки
		|		ЛЕВОЕ СОЕДИНЕНИЕ РегистрСведений.ЗначенияХарактеристик
КАК ЗначенияХарактеристик
		|			ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Процессоры КАК
Процессоры
		|			ПО (Процессоры.Ссылка =
ЗначенияХарактеристик.Значение)
		|		ПО (ЗначенияХарактеристик.Характеристика.Владелец.Ссылка
= ТоварыНаСкладеОстатки.Номенклатура)
		|ГДЕ
		|	ЗначенияХарактеристик.Свойство = &Свойство
		|	И Процессоры.КоличествоЯдер >= &КоличествоЯдер
		|
		|УПОРЯДОЧИТЬ ПО
		|	КоличествоЯдер";

	Запрос.УстановитьПараметр("КоличествоЯдер", КоличествоЯдер);
	Запрос.УстановитьПараметр("Свойство",
ПланыВидовХарактеристик.СвойстваНоменклатуры.Процессор);

	РезультатЗапроса = Запрос.Выполнить();

	ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

	Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
		пк = Новый Структура;
		пк.Вставить("Номенклатура",ВыборкаДетальныеЗаписи.ССылкаПК);
		пк.Вставить("Характеристики",ВыборкаДетальныеЗаписи.Характеристика);
		возврат пк;
	КонецЦикла;

	возврат "";

КонецФункции

&НаКлиенте
Процедура ПодобратьРабочееМесто(Форма)
```

```
	Объект.СписокТехники.Очистить();

	ЗадачиРаботника = Форма.ЗадачиРаботника;

	КоличествоМониторов = Форма.КоличествоМониторов;
	Принтер = Форма.Принтер;
	Сканер = Форма.Сканер;
	Наушники = Форма.Наушники;
	Микрофон = Форма.Микрофон;
	Камера = Форма.Камера;

	Если ЗадачиРаботника =
ПредопределенноеЗначение("Перечисление.ЗадачиРаботника.Документооборот") Тогда
		пк = ПолучитьПодходящийПК(2);
	ИначеЕсли ЗадачиРаботника =
ПредопределенноеЗначение("Перечисление.ЗадачиРаботника.Программирование") Тогда

	  пк = ПолучитьПодходящийПК(4);
	ИначеЕсли ЗадачиРаботника =
ПредопределенноеЗначение("Перечисление.ЗадачиРаботника.РаботаСГрафикой") Тогда

		пк = ПолучитьПодходящийПК(8);
	КонецЕсли;

	Если пк = "" Тогда
		Сообщить("На складе нет подходящего ПК");
	Иначе
		ДобавитьСтроку(Объект.СписокТехники, пк.Номенклатура,
пк.Характеристики, 1);
	КонецЕсли;

	клавиатура = ПолучитьНоменклатуру("Клавиатуры");
	мышь = ПолучитьНоменклатуру("Мыши");
	монитор = ПолучитьНоменклатуру("Мониторы");

	ПроверитьИДобавить(монитор, КоличествоМониторов);
    ПроверитьИДобавить(мышь, 1);
	ПроверитьИДобавить(клавиатура, 1);

	Если Принтер = Истина Тогда
		ПроверитьИДобавить(ПолучитьНоменклатуру("Принтеры"), 1)
	КонецЕсли;

	Если Сканер = Истина Тогда
		ПроверитьИДобавить(ПолучитьНоменклатуру("Сканеры"), 1)
	КонецЕсли;

	Если Наушники = Истина Тогда
		ПроверитьИДобавить(ПолучитьНоменклатуру("Наушники"), 1)
	КонецЕсли;

	Если Микрофон = Истина Тогда
		ПроверитьИДобавить(ПолучитьНоменклатуру("Микрофоны"), 1)
	КонецЕсли;

	Если Камера = Истина Тогда
		ПроверитьИДобавить(ПолучитьНоменклатуру("Камеры"), 1)
	КонецЕсли;

КонецПроцедуры

&НаКлиенте
Функция ПроверитьИДобавить(номенклатура, количество)
	Если ТипЗнч(номенклатура) <> Тип("Структура") Тогда
```

```
            Сообщить("На складе не хватает товара с категорией: "+номенклатура);
            возврат Ложь;
      КонецЕсли;

      ДобавитьСтроку(Объект.СписокТехники, номенклатура.Номенклатура,
номенклатура.Характеристики, количество);
КонецФункции

&НаКлиенте
Процедура ДобавитьСтроку(табЧасть, номенклатура, характеристики, количество)
      НоваяСтрока = табЧасть.Добавить();
      НоваяСтрока.Номенклатура = номенклатура;
      НоваяСтрока.Характеристики = характеристики;
      НоваяСтрока.Количество = количество;
КонецПроцедуры


&НаСервере
Функция ПолучитьНоменклатуру(категория)
      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     ХарактеристикиСписок.Владелец КАК Номенклатура,
            |     ХарактеристикиСписок.Ссылка КАК Характеристики,
            |     ТоварыНаСкладеОстатки.КоличествоОстаток КАК КоличествоОстаток
            |ИЗ
            |     РегистрНакопления.ТоварыНаСкладе.Остатки КАК
ТоварыНаСкладеОстатки
            |           ЛЕВОЕ СОЕДИНЕНИЕ Справочник.Характеристики КАК
ХарактеристикиСписок
            |           ПО ТоварыНаСкладеОстатки.Характеристики =
ХарактеристикиСписок.Ссылка
            |                 И ТоварыНаСкладеОстатки.Номенклатура.Ссылка =
ХарактеристикиСписок.Владелец
            |ГДЕ
            |     ХарактеристикиСписок.Владелец.Родитель = &Родитель";

      Запрос.УстановитьПараметр("Родитель",
Справочники.Номенклатура.НайтиПоНаименованию(категория));

      Выборка = Запрос.Выполнить().Выбрать();


      Пока Выборка.Следующий() Цикл
            номенклатура = Новый Структура;
          номенклатура.Вставить("Номенклатура", Выборка.Номенклатура);
            номенклатура.Вставить("Характеристики", Выборка.Характеристики);
            возврат номенклатура;
      КонецЦикла;

      возврат категория;
КонецФункции



&НаКлиенте
Процедура ПередЗаписью(Отказ, ПараметрыЗаписи)

      //Список = РассчитатьОстатки(Объект.Ссылка, Объект.Дата);

      список = Новый Массив;

      Для каждого стр из Объект.СписокТехники цикл
```

```
            наСкладе = ПолучитьОстатокНоменклатуры(Объект.Дата,
стр.Номенклатура, стр.Характеристики);
            //Сообщить(""+стр.Количество+" "+стр.Номенклатура +" "+наСкладе );

            Если стр.Количество > наСкладе Тогда
                  элемент = Новый Структура;
                  элемент.Вставить("Номенклатура",стр.Номенклатура);
                  элемент.Вставить("Характеристики",стр.Характеристики);
                  элемент.Вставить("Количество",стр.Количество-наСкладе);
                  список.Добавить(элемент);
            КонецЕсли;

      КонецЦикла;

      //Сообщить(список.Количество());

      Если список.Количество() <> 0 Тогда
            Результат = Вопрос("Создать заявку на получение недостающей
номенклатуры?", РежимДиалогаВопрос.ДаНет);
            Если Результат = КодВозвратаДиалога.Да Тогда
                  ССылкаНаДокумент = СоздатьЗаявку(список, Объект.Дата);
                  ОткрытьЗначение(ССылкаНаДокумент);
            КонецЕсли;

      КонецЕсли;

КонецПроцедуры

&НаСервере
Функция СоздатьЗаявку(список, дата)

      НовыйДокумент = Документы.ЗаявкиДляЗаказа.СоздатьДокумент();
      НовыйДокумент.Дата = дата - 60;


      Для Каждого Элемент из список Цикл
            НоваяСтрока = НовыйДокумент.Товары.Добавить();
            НоваяСтрока.Номенклатура = Элемент.Номенклатура;
            НоваяСтрока.Характеристики = Элемент.Характеристики;
            НоваяСтрока.Количество = Элемент.Количество;
      КонецЦикла;

      НовыйДокумент.Записать();

      Возврат НовыйДокумент.Ссылка;

КонецФункции

&НаСервере
Функция ПолучитьОстатокНоменклатуры(Момент, Номенклатура, Характеристики)
            //{{КОНСТРУКТОР_ЗАПРОСА_С_ОБРАБОТКОЙ_РЕЗУЛЬТАТА
      // Данный фрагмент построен конструктором.
      // При повторном использовании конструктора, внесенные вручную изменения
будут утеряны!!!

      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     ТоварыНаСкладеОстатки.Номенклатура КАК Номенклатура,
            |     ТоварыНаСкладеОстатки.Характеристики КАК Характеристики,
            |     ТоварыНаСкладеОстатки.КоличествоОстаток КАК Количество
            |ИЗ
            |     РегистрНакопления.ТоварыНаСкладе.Остатки(&Момент, ) КАК
ТоварыНаСкладеОстатки
```

```
            |ГДЕ
            |     ТоварыНаСкладеОстатки.Номенклатура = &Номенклатура
            |     И ТоварыНаСкладеОстатки.Характеристики = &Характеристики";

      Запрос.УстановитьПараметр("Момент", Момент);
      Запрос.УстановитьПараметр("Номенклатура", Номенклатура);
      Запрос.УстановитьПараметр("Характеристики", Характеристики);

      РезультатЗапроса = Запрос.Выполнить();

      ВыборкаДетальныеЗаписи = РезультатЗапроса.Выбрать();

      Пока ВыборкаДетальныеЗаписи.Следующий() Цикл
            возврат ВыборкаДетальныеЗаписи.Количество;
      КонецЦикла;

      возврат 0;

КонецФункции


&НаКлиенте
Процедура КонфигураторПК(Команда)
      Форма = ПолучитьФорму("ОбщаяФорма.КонфигураторПК");
      Форма.ОткрытьМодально();

      Если Форма.КонфигурацияВерна Тогда
            Сообщить(Форма.ПК);
            ДобавитьСтроку(Объект.СписокТехники, Форма.ПК, Форма.Характеристика,
1);
      КонецЕсли;
КонецПроцедуры
```

Листинг Ж.4 – Модуль объекта «СписаниеТехники»

```
Процедура ОбработкаПроведения(Отказ, Режим)

      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     СписаниеТехникиСписокТехники.Номенклатура КАК Номенклатура,
            |     СУММА(СписаниеТехникиСписокТехники.Количество) КАК Количество,
            |     СписаниеТехникиСписокТехники.Характеристики КАК
Характеристики,
            |     СписаниеТехники.РабочееМесто КАК РабочееМесто,
            |     СписаниеТехники.НомерЛицо КАК НомерЛицо
            |ПОМЕСТИТЬ ТабДок
            |ИЗ
            |     Документ.СписаниеТехники.СписокТехники КАК
СписаниеТехникиСписокТехники
            |           ВНУТРЕННЕЕ СОЕДИНЕНИЕ Документ.СписаниеТехники КАК
СписаниеТехники
            |           ПО СписаниеТехникиСписокТехники.Ссылка =
СписаниеТехники.СписокТехники.Ссылка
            |ГДЕ
            |     СписаниеТехники.СписокТехники.Ссылка = &Ссылка
            |
            |СГРУППИРОВАТЬ ПО
            |     СписаниеТехникиСписокТехники.Номенклатура,
            |     СписаниеТехникиСписокТехники.Характеристики,
            |     СписаниеТехники.РабочееМесто,
            |     СписаниеТехники.НомерЛицо
```

```
            |;
            |

      |////////////////////////////////////////////////////////////////////////
///////
            |ВЫБРАТЬ
            |     ТабДок.Номенклатура КАК Номенклатура,
            |     ТабДок.Количество КАК Количество,
            |     ТабДок.Характеристики КАК Характеристики,
            |     ТабДок.РабочееМесто КАК РабочееМесто,
            |     ТабДок.НомерЛицо КАК НомерЛицо,
            |     ЕСТЬNULL(ТехникаВРаботеОстатки.КоличествоОстаток, 0) КАК
КоличествоОстаток
            |ИЗ
            |     ТабДок КАК ТабДок
            |           ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ТехникаВРаботе.Остатки(
            |                        &Момент,
            |                        (Номенклатура, Характеристики, РабочееМесто,
НомерЛицо) В
            |                              (ВЫБРАТЬ
            |                                    ТабДок.Номенклатура КАК
Номенклатура,
            |                                    ТабДок.Характеристики КАК
Характеристики,
            |                                    ТабДок.РабочееМесто КАК
РабочееМесто,
            |                                    ТабДок.НомерЛицо КАК НомерЛицо
            |                              ИЗ
            |                                    ТабДок КАК ТабДок)) КАК
ТехникаВРаботеОстатки
            |           ПО ТабДок.Номенклатура =
ТехникаВРаботеОстатки.Номенклатура";

      Если Режим = РежимПроведенияДокумента.Оперативный Тогда
            Запрос.УстановитьПараметр("Момент", неопределено);
      Иначе
            Запрос.УстановитьПараметр("Момент", МоментВремени());
      КонецЕсли;
      Запрос.УстановитьПараметр("Ссылка", Ссылка);

      РезультатЗапроса = Запрос.Выполнить();

      Выборка = РезультатЗапроса.Выбрать();

      Движения.ТехникаВРаботе.Записывать = Истина;
      Движения.СписаннаяТехника.Записывать = Истина;
      Пока Выборка.Следующий() Цикл
            // Расход с работы
            Движение = Движения.ТехникаВРаботе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.РабочееМесто = РабочееМесто;
            Движение.НомерЛицо = НомерЛицо;
            Движение.Количество = Выборка.Количество;

            Если Выборка.КоличествоОстаток < Выборка.Количество Тогда
                  //Отказ = Истина;
                  ВызватьОшибку("Не хватает товара '" + Выборка.Номенклатура +
"' в количестве "+ (Выборка.Количество-Выборка.КоличествоОстаток));
            КонецЕсли;
```

```
            // Приход в списание
            Движение = Движения.СписаннаяТехника.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.Количество = Выборка.Количество;

      КонецЦикла;

КонецПроцедуры
```

Листинг Ж.5 – Форма документа «СписаниеТехники»

```
&НаКлиенте
Процедура РабочееМестоПриИзменении(Элемент)
        Если Строка(Объект.РабочееМесто) = "Материально ответственное лицо"
Тогда
        Объект.НомерЛицо = Справочники.Сотрудники.ПустаяСсылка();
            Элементы.НомерЛицо.Заголовок = "Материально ответственное лицо"
    Иначе
        Объект.НомерЛицо = "";
            Элементы.НомерЛицо.Заголовок = "Номер места"
      КонецЕсли;
      Элементы.НомерЛицо.ВыбиратьТип = Ложь;

КонецПроцедуры
```

Листинг Ж.6 – Модуль объекта «ПередачаВРемонт»

```
Процедура ОбработкаПроведения(Отказ, Режим)

      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     ПередачаВРемонтСписокТехники.Номенклатура КАК Номенклатура,
            |     СУММА(ПередачаВРемонтСписокТехники.Количество) КАК Количество,
            |     ПередачаВРемонтСписокТехники.Характеристики КАК
Характеристики,
            |     ПередачаВРемонт.РабочееМесто КАК РабочееМесто,
            |     ПередачаВРемонт.НомерЛицо КАК НомерЛицо,
            |     ПередачаВРемонт.ДатаОкончанияРемонта КАК ДатаОкончанияРемонта
            |ПОМЕСТИТЬ ТабДок
            |ИЗ
            |     Документ.ПередачаВРемонт.СписокТехники КАК
ПередачаВРемонтСписокТехники
            |           ВНУТРЕННЕЕ СОЕДИНЕНИЕ Документ.ПередачаВРемонт КАК
ПередачаВРемонт
            |           ПО ПередачаВРемонтСписокТехники.Ссылка =
ПередачаВРемонт.СписокТехники.Ссылка
            |ГДЕ
            |     ПередачаВРемонт.СписокТехники.Ссылка = &Ссылка
            |
            |СГРУППИРОВАТЬ ПО
            |     ПередачаВРемонтСписокТехники.Номенклатура,
            |     ПередачаВРемонтСписокТехники.Характеристики,
            |     ПередачаВРемонт.РабочееМесто,
            |     ПередачаВРемонт.НомерЛицо,
```

```
            |     ПередачаВРемонт.ДатаОкончанияРемонта
            |;
            |

      |////////////////////////////////////////////////////////////////////
///////
            |ВЫБРАТЬ
            |     ТабДок.Номенклатура КАК Номенклатура,
            |     ТабДок.Количество КАК Количество,
            |     ТабДок.Характеристики КАК Характеристики,
            |     ТабДок.РабочееМесто КАК РабочееМесто,
            |     ТабДок.НомерЛицо КАК НомерЛицо,
            |     ЕСТЬNULL(ТехникаВРаботеОстатки.КоличествоОстаток, 0) КАК
КоличествоОстаток,
            |     ТабДок.ДатаОкончанияРемонта КАК ДатаОкончанияРемонта
            |ИЗ
            |     ТабДок КАК ТабДок
            |           ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ТехникаВРаботе.Остатки(
            |                       &Момент,
            |                       (Номенклатура, Характеристики, РабочееМесто,
НомерЛицо) В
            |                             (ВЫБРАТЬ
            |                                   ТабДок.Номенклатура КАК
Номенклатура,
            |                                   ТабДок.Характеристики КАК
Характеристики,
            |                                   ТабДок.РабочееМесто КАК
РабочееМесто,
            |                                   ТабДок.НомерЛицо КАК НомерЛицо
            |                             ИЗ
            |                                   ТабДок КАК ТабДок)) КАК
ТехникаВРаботеОстатки
            |           ПО ТабДок.Номенклатура =
ТехникаВРаботеОстатки.Номенклатура";

      Если Режим = РежимПроведенияДокумента.Оперативный Тогда
            Запрос.УстановитьПараметр("Момент", неопределено);
      Иначе
            Запрос.УстановитьПараметр("Момент", МоментВремени());
      КонецЕсли;
      Запрос.УстановитьПараметр("Ссылка", Ссылка);

      РезультатЗапроса = Запрос.Выполнить();

      Выборка = РезультатЗапроса.Выбрать();

      Движения.ТехникаВРаботе.Записывать = Истина;
      Движения.ТехникаВРемонте.Записывать = Истина;
      Пока Выборка.Следующий() Цикл
            // Расход с работы
            Движение = Движения.ТехникаВРаботе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.РабочееМесто = РабочееМесто;
            Движение.НомерЛицо = НомерЛицо;
            Движение.Количество = Выборка.Количество;

            Если Выборка.КоличествоОстаток < Выборка.Количество Тогда
                  //Отказ = Истина;
                  ВызватьОшибку("Не хватает товара '" + Выборка.Номенклатура +
"' в количестве "+ (Выборка.Количество-Выборка.КоличествоОстаток));
```

```
            КонецЕсли;

            // Приход в ремонт
            Движение = Движения.ТехникаВРемонте.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.Количество = Выборка.Количество;
      КонецЦикла;

КонецПроцедуры
```

Листинг Ж.7 – Форма документа «ПередачаВРемонт»

```
&НаКлиенте
Процедура РабочееМестоПриИзменении(Элемент)
        Если Строка(Объект.РабочееМесто) = "Материально ответственное лицо"
Тогда
        Объект.НомерЛицо = Справочники.Сотрудники.ПустаяСсылка();
            Элементы.НомерЛицо.Заголовок = "Материально ответственное лицо"
    Иначе
        Объект.НомерЛицо = "";
            Элементы.НомерЛицо.Заголовок = "Номер места"
      КонецЕсли;
      Элементы.НомерЛицо.ВыбиратьТип = Ложь;

КонецПроцедуры
```

Листинг Ж.8 – Модуль объекта «ПриходСРемонта»

```
Процедура ОбработкаПроведения(Отказ, Режим)

      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     ПриходСРемонтаСписокТехники.Номенклатура КАК Номенклатура,
            |     СУММА(ПриходСРемонтаСписокТехники.Количество) КАК Количество,
            |     ПриходСРемонтаСписокТехники.Характеристики КАК Характеристики
            |ПОМЕСТИТЬ ТабДок
            |ИЗ
            |     Документ.ПриходСРемонта.СписокТехники КАК
ПриходСРемонтаСписокТехники
            |           ВНУТРЕННЕЕ СОЕДИНЕНИЕ Документ.ПриходСРемонта КАК
ПриходСРемонта
            |           ПО ПриходСРемонтаСписокТехники.Ссылка =
ПриходСРемонта.СписокТехники.Ссылка
            |ГДЕ
            |     ПриходСРемонта.СписокТехники.Ссылка = &Ссылка
            |
            |СГРУППИРОВАТЬ ПО
            |     ПриходСРемонтаСписокТехники.Номенклатура,
            |     ПриходСРемонтаСписокТехники.Характеристики
            |;
            |

      |////////////////////////////////////////////////////////////////////
///////
            |ВЫБРАТЬ
            |     ТабДок.Номенклатура КАК Номенклатура,
```

```
            |     ТабДок.Количество КАК Количество,
            |     ТабДок.Характеристики КАК Характеристики,
            |     ЕСТЬNULL(ТехникаВРемонтеОстатки.КоличествоОстаток, 0) КАК
КоличествоОстаток
            |ИЗ
            |     ТабДок КАК ТабДок
            |           ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ТехникаВРемонте.Остатки(
            |                        &Момент,
            |                        (Номенклатура, Характеристики) В
            |                              (ВЫБРАТЬ
            |                                    ТабДок.Номенклатура КАК
Номенклатура,
            |                                    ТабДок.Характеристики КАК
Характеристики
            |                              ИЗ
            |                                    ТабДок КАК ТабДок)) КАК
ТехникаВРемонтеОстатки
            |           ПО ТабДок.Номенклатура =
ТехникаВРемонтеОстатки.Номенклатура";

      Если Режим = РежимПроведенияДокумента.Оперативный Тогда
            Запрос.УстановитьПараметр("Момент", неопределено);
      Иначе
            Запрос.УстановитьПараметр("Момент", МоментВремени());
      КонецЕсли;
      Запрос.УстановитьПараметр("Ссылка", Ссылка);

      РезультатЗапроса = Запрос.Выполнить();

      Выборка = РезультатЗапроса.Выбрать();

      Движения.ТехникаВРаботе.Записывать = Истина;
      Движения.ТехникаВРемонте.Записывать = Истина;
      Пока Выборка.Следующий() Цикл
            // Приход в работу
            Движение = Движения.ТехникаВРаботе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.РабочееМесто = РабочееМесто;
            Движение.НомерЛицо = НомерЛицо;
            Движение.Количество = Выборка.Количество;

            Если Выборка.КоличествоОстаток < Выборка.Количество Тогда
                  //Отказ = Истина;
                  ВызватьОшибку("Не хватает товара '" + Выборка.Номенклатура +
"' в количестве "+ (Выборка.Количество-Выборка.КоличествоОстаток));
            КонецЕсли;

            // Расход с ремонта
            Движение = Движения.ТехникаВРемонте.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.Количество = Выборка.Количество;
      КонецЦикла;
КонецПроцедуры
```

Листинг Ж.9 – Форма документа «ПриходСРемонта»

```
&НаКлиенте
Процедура РабочееМестоПриИзменении(Элемент)
        Если Строка(Объект.РабочееМесто) = "Материально ответственное лицо"
Тогда
        Объект.НомерЛицо = Справочники.Сотрудники.ПустаяСсылка();
            Элементы.НомерЛицо.Заголовок = "Материально ответственное лицо"
    Иначе
        Объект.НомерЛицо = "";
            Элементы.НомерЛицо.Заголовок = "Номер места"
      КонецЕсли;
      Элементы.НомерЛицо.ВыбиратьТип = Ложь;

КонецПроцедуры
```

Листинг Ж.10 – Модуль объекта «ПередачаТехники»

```
Процедура ОбработкаПроведения(Отказ, Режим)

      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     ПередачаТехникиСписокТехники.Номенклатура КАК Номенклатура,
            |     СУММА(ПередачаТехникиСписокТехники.Количество) КАК Количество,
            |     ПередачаТехникиСписокТехники.Характеристики КАК
Характеристики,
            |     ПередачаТехники.РабочееМесто КАК РабочееМесто,
            |     ПередачаТехники.НомерЛицо КАК НомерЛицо
            |ПОМЕСТИТЬ ТабДок
            |ИЗ
            |     Документ.ПередачаТехники.СписокТехники КАК
ПередачаТехникиСписокТехники
            |           ВНУТРЕННЕЕ СОЕДИНЕНИЕ Документ.ПередачаТехники КАК
ПередачаТехники
            |           ПО ПередачаТехникиСписокТехники.Ссылка =
ПередачаТехники.СписокТехники.Ссылка
            |ГДЕ
            |     ПередачаТехники.СписокТехники.Ссылка = &Ссылка
            |
            |СГРУППИРОВАТЬ ПО
            |     ПередачаТехникиСписокТехники.Номенклатура,
            |     ПередачаТехникиСписокТехники.Характеристики,
            |     ПередачаТехники.РабочееМесто,
            |     ПередачаТехники.НомерЛицо
            |;
            |

      |////////////////////////////////////////////////////////////////////
///////
            |ВЫБРАТЬ
            |     ТабДок.Номенклатура КАК Номенклатура,
            |     ТабДок.Количество КАК Количество,
            |     ТабДок.Характеристики КАК Характеристики,
            |     ТабДок.РабочееМесто КАК РабочееМесто,
            |     ТабДок.НомерЛицо КАК НомерЛицо,
            |     ЕСТЬNULL(ТехникаВРаботеОстатки.КоличествоОстаток, 0) КАК
КоличествоОстаток
            |ИЗ
            |     ТабДок КАК ТабДок
            |           ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ТехникаВРаботе.Остатки(
            |                       &Момент,
```

```
            |                        (Номенклатура, Характеристики, РабочееМесто,
НомерЛицо) В
            |                              (ВЫБРАТЬ
            |                                    ТабДок.Номенклатура КАК
Номенклатура,
            |                                    ТабДок.Характеристики КАК
Характеристики,
            |                                    ТабДок.РабочееМесто КАК
РабочееМесто,
            |                                    ТабДок.НомерЛицо КАК НомерЛицо
            |                                ИЗ
            |                                    ТабДок КАК ТабДок)) КАК
ТехникаВРаботеОстатки
            |          ПО ТабДок.Номенклатура =
ТехникаВРаботеОстатки.Номенклатура";

      Если Режим = РежимПроведенияДокумента.Оперативный Тогда
            Запрос.УстановитьПараметр("Момент", неопределено);
      Иначе
            Запрос.УстановитьПараметр("Момент", МоментВремени());
      КонецЕсли;
      Запрос.УстановитьПараметр("Ссылка", Ссылка);
      РезультатЗапроса = Запрос.Выполнить();
      Выборка = РезультатЗапроса.Выбрать();
      Движения.ТехникаВРаботе.Записывать = Истина;
      Пока Выборка.Следующий() Цикл
            // Расход с работы
            Движение = Движения.ТехникаВРаботе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Расход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.РабочееМесто = РабочееМесто;
            Движение.НомерЛицо = НомерЛицо;
            Движение.Количество = Выборка.Количество;
            Если Выборка.КоличествоОстаток < Выборка.Количество Тогда
                  //Отказ = Истина;
                  ВызватьОшибку("Не хватает товара '" + Выборка.Номенклатура +
"' в количестве "+ (Выборка.Количество-Выборка.КоличествоОстаток));
            КонецЕсли;

            // Приход в работу
            Движение = Движения.ТехникаВРаботе.Добавить();
            Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
            Движение.Период = Дата;
            Движение.Номенклатура = Выборка.Номенклатура;
            Движение.Характеристики = Выборка.Характеристики;
            Движение.РабочееМесто = НовоеРабочееМесто;
            Движение.НомерЛицо = НовоеНомерЛицо;
            Движение.Количество = Выборка.Количество;
      КонецЦикла;
КонецПроцедуры
```

Листинг Ж.11 – Форма документа «ПередачаТехники»

```
&НаКлиенте
Процедура РабочееМестоПриИзменении(Элемент)
      Если Строка(Объект.РабочееМесто) = "Материально ответственное лицо" Тогда
Объект.НомерЛицо = Справочники.Сотрудники.ПустаяСсылка();
            Элементы.НомерЛицо.Заголовок = "Материально ответственное лицо"
    Иначе
Объект.НомерЛицо = "";
```

```
            Элементы.НомерЛицо.Заголовок = "Номер места"
      КонецЕсли;
      Элементы.НомерЛицо.ВыбиратьТип = Ложь;

      Если Строка(Объект.НовоеРабочееМесто) = "Материально ответственное лицо"
Тогда
Объект.НовоеНомерЛицо = Справочники.Сотрудники.ПустаяСсылка();
            Элементы.НовоеНомерЛицо.Заголовок = "Материально ответственное лицо"
    Иначе
Объект.НовоеНомерЛицо = "";
            Элементы.НовоеНомерЛицо.Заголовок = "Номер места"
      КонецЕсли;
      Элементы.НовоеНомерЛицо.ВыбиратьТип = Ложь;
КонецПроцедуры
```

Листинг Ж.12 – Общая форма «ПодборРабочегоМеста»

```
&НаКлиенте
Процедура Применить(Команда)
      Выполнено = Истина;
      Закрыть();
КонецПроцедуры

&НаСервере
Процедура ПриСозданииНаСервере(Отказ, СтандартнаяОбработка)
      ЗадачиРаботника = Перечисления.ЗадачиРаботника.Документооборот;
      КоличествоМониторов = 1;
      Выполнено = Ложь;
КонецПроцедуры
```

Листинг Ж.13 – Общая форма «КонфигураторПК»

```
&НаКлиенте
Процедура Применить(Команда)
      Если ПриЗакрытииНаСервере() Тогда
            Закрыть();
      КонецЕсли;

КонецПроцедуры

&НаКлиенте
Процедура ПриОткрытии(Отказ)
      Если Не Процессор.Пустая() Тогда
            ПроцессорПриИзменении();
      КонецЕсли;

      Если Не МатеринскаяПлата.Пустая() Тогда
            МатеринскаяПлатаПриИзменении();
      КонецЕсли;
КонецПроцедуры

&НаКлиенте
Процедура ЗадатьТипПамяти(типПамяти)

      памятьИнфо = Новый Структура;
    памятьИнфо.Вставить("ТипПамяти", типПамяти);

      Элементы.ОперативнаяПамять.ПараметрыВыбора = ЗадатьПараметры(памятьИнфо);
```

```
КонецПроцедуры

&НаКлиенте
Процедура ПроцессорПриИзменении()

	Если Процессор.Пустая() Тогда
		Элементы.МатеринскаяПлата.ПараметрыВыбора = ОчиститьПараметры();
	Иначе

		проц = Комплектующие.ПолучитьПроцессорИнфо(Процессор);

		процИнфо = Новый Структура;
	    процИнфо.Вставить("Сокет", проц.Сокет);
	    процИнфо.Вставить("ТипПамяти", проц.ТипПамяти);

		Элементы.МатеринскаяПлата.ПараметрыВыбора =
ЗадатьПараметры(процИнфо);

		ЗадатьТипПамяти(проц.ТипПамяти);

	КонецЕсли;

КонецПроцедуры

&НаКлиенте
Процедура МатеринскаяПлатаПриИзменении()

	Если МатеринскаяПлата.Пустая() Тогда
		Элементы.Процессор.ПараметрыВыбора = ОчиститьПараметры();
	Иначе
		плата = Комплектующие.ПолучитьМатеринкаИнфо(МатеринскаяПлата);

		платаИнфо = Новый Структура;
	    платаИнфо.Вставить("Сокет", плата.Сокет);
	    платаИнфо.Вставить("ТипПамяти", плата.ТипПамяти);

		Элементы.Процессор.ПараметрыВыбора = ЗадатьПараметры(платаИнфо);

		ЗадатьТипПамяти(плата.ТипПамяти);
	КонецЕсли;

КонецПроцедуры

&НаКлиенте
Функция ЗадатьПараметры(параметры)

	НовыйМассив = Новый Массив();

	Для Каждого параметр из параметры Цикл
		отбор = "Отбор."+параметр.Ключ;
		НовыйМассив.Добавить(Новый ПараметрВыбора(отбор,
параметр.Значение));
	КонецЦикла;

	НовыеПараметры = Новый ФиксированныйМассив(НовыйМассив);

	возврат НовыеПараметры;
КонецФункции

&НаКлиенте
Функция ОчиститьПараметры()

	НовыйМассив = Новый Массив();
	НовыеПараметры = Новый ФиксированныйМассив(НовыйМассив);
```

```
	возврат НовыеПараметры;

КонецФункции

Функция СократитьНазваниеПроцессора(полноеНазвание)
	RegExp = Новый COMОбъект("VBScript.RegExp");

	название_проца = полноеНазвание;
	RegExp.Pattern ="Ryzen\s+";
	Если RegExp.Test(полноеНазвание) Тогда
		возврат RegExp.Replace(полноеНазвание,"R")
	КонецЕсли;

	RegExp.Pattern ="Intel Core\s+";
	Если RegExp.Test(полноеНазвание) Тогда
		возврат RegExp.Replace(полноеНазвание,"")
	КонецЕсли;

	возврат название_проца
КонецФункции

Функция СократитьНазваниеВидеокарты(полноеНазвание)
	RegExp = Новый COMОбъект("VBScript.RegExp");

	название_видеокарты = полноеНазвание;

	RegExp.Pattern ="(GTX.*)|(RTX.*)";
	Matches = RegExp.Execute(полноеНазвание);
	Если Matches.Count <> 0 Тогда
		Для i = 0 По Matches.Item(0).SubMatches.Count-1 Цикл
		стр = Matches.Item(0).SubMatches.Item(i);
			Если Не ПустаяСтрока(стр)  Тогда
				возврат стр;
			КонецЕсли;
		КонецЦикла;

	КонецЕсли;

	возврат название_видеокарты;

КонецФункции

Функция ПроверитьКонфигурациюПК()

	Если Процессор.Сокет <> МатеринскаяПлата.Сокет Тогда
		Сообщение = "Материнская плата несовместима с
Процессором"+Символы.ПС+
		"Сокет Процессора:"+Строка(Процессор.Сокет)+Символы.ПС+
		"Сокет материнской платы:"+Строка(МатеринскаяПлата.Сокет);
		Сообщить(Сообщение);
		возврат Истина;
	КонецЕсли;

	Если МатеринскаяПлата.ТипПамяти <> ОперативнаяПамять.ТипПамяти Тогда
		Сообщение = "Материнская плата не поддерживает тип памяти
"+Строка(ОперативнаяПамять.ТипПамяти);
		Сообщить(Сообщение);
		возврат Истина
	КонецЕсли;

	Если Процессор.ТипПамяти <> ОперативнаяПамять.ТипПамяти Тогда
		Сообщение = "Процессор не поддерживает тип памяти
"+Строка(ОперативнаяПамять.ТипПамяти);
		Сообщить(Сообщение);
```

```
            возврат Истина
      КонецЕсли;

      Если Видеокарта.Пустая() Тогда
            Если Процессор.Видеоядро = Ложь Тогда
                  Сообщение = "В сборке отсутствует Видеокартакарта или
Процессор с Видеокартаядром";
                  Сообщить(Сообщение);
                  возврат Истина
            КонецЕсли;

            tdp_gpu = 0;
            pin_gpu = 0;
      Иначе
            tdp_gpu = Видеокарта.Энергопотребление;
            pin_gpu = Видеокарта.РазъемПитания;
      КонецЕсли;

      Если pin_gpu > БлокПитания.РазъемыGPU  Тогда
            Сообщение = "У блока питания нет подходящих кабелей для питания
Видеокартакарты"+Символы.ПС+
            "Видеокарте необходимо "+Строка(pin_gpu)+" PIN питание"+Символы.ПС+
            "Блок питания имеет только "+Строка(БлокПитания.РазъемыGPU)+" PIN
кабели";
            Сообщить(Сообщение);
            возврат Истина
      КонецЕсли;

      энергопотреблениеСистемы =
(tdp_gpu+Процессор.Энергопотребление+ХранениеДанных.Энергопотребление)*1.12;
      Если энергопотреблениеСистемы > БлокПитания.Мощность Тогда
            Сообщение = "Потребляемая мощность системы:
"+Строка(энергопотреблениеСистемы)+" Вт"+Символы.ПС+
            "Мощность блока питания: "+Строка(БлокПитания.Мощность)+" Вт";
            Сообщить(Сообщение);
            возврат Истина
      КонецЕсли;

      возврат Ложь;
КонецФункции

&НаСервере
Функция ПриЗакрытииНаСервере()

      КонфигурацияВерна = Ложь;

      Если  Процессор.Пустая() ИЛИ
                  МатеринскаяПлата.Пустая() ИЛИ
                  ОперативнаяПамять.Пустая() ИЛИ
                  БлокПитания.Пустая() ИЛИ
                  ХранениеДанных.Пустая()
      Тогда
            Сообщить("Необходимо заполнить поля");
            возврат Ложь;
      КонецЕсли;

      процессор = Процессор;
      матПлата = МатеринскаяПлата;
      озу = ОперативнаяПамять;
      видео = Видеокарта;
      блокПитания = БлокПитания;
      хранениеДанных = ХранениеДанных;

      Если ПроверитьКонфигурациюПК() = Истина Тогда
```

```
            КонфигурацияВерна = Ложь;
            возврат Ложь;
      КонецЕсли;

      название_проца = СтрЗаменить(Процессор.Наименование, "Intel Core ", "");
      название_проца = СтрЗаменить(название_проца, "Ryzen ", "R");


      Если Видеокарта.Пустая() Тогда
            название = СократитьНазваниеПроцессора(Процессор.Наименование)+"
"+ОперативнаяПамять.Объем+"Гб";
      Иначе
            название = СократитьНазваниеВидеокарты(Видеокарта.Наименование)+"
"+СократитьНазваниеПроцессора(Процессор.Наименование)+"
"+ОперативнаяПамять.Объем+"Гб";
      КонецЕсли;

      название = "ПК "+название+" "+блокПитания.Мощность+"W
"+хранениеДанных.Тип+" "+хранениеДанных.Объем+"Гб";

      Запрос = Новый Запрос;
      Запрос.Текст =
            "ВЫБРАТЬ
            |     ЗначенияХарактеристик.Характеристика.Владелец.Ссылка КАК
ССылкаПК,
            |     ЗначенияХарактеристик.Характеристика КАК Характеристика
            |ИЗ
            |     РегистрСведений.ЗначенияХарактеристик КАК
ЗначенияХарактеристик
            |ГДЕ
            |     ЗначенияХарактеристик.Характеристика.Владелец.Наименование =
&Наименование
            |
            |СГРУППИРОВАТЬ ПО
            |     ЗначенияХарактеристик.Характеристика.Владелец.Ссылка,
            |     ЗначенияХарактеристик.Характеристика";

      Запрос.УстановитьПараметр("Наименование", название);

      РезультатЗапроса = Запрос.Выполнить().Выбрать();

      количествоТакихЖеКомпов = РезультатЗапроса.Количество();
      Если количествоТакихЖеКомпов > 0 Тогда
            Сообщить("Компьютер уже есть");
            Если РезультатЗапроса.Следующий() Тогда
                  ПК = РезультатЗапроса.ССылкаПК;
                  Характеристика = РезультатЗапроса.Характеристика;
            КонецЕсли;
      Иначе

            НовЭл = Справочники.Номенклатура.СоздатьЭлемент();

            НовЭл.Наименование = название;
            НовЭл.Родитель =
Справочники.Номенклатура.НайтиПоНаименованию("Компьютеры");
            НовЭл.Записать();

            ХарактеристикиПК = Справочники.Характеристики.СоздатьЭлемент();
            ХарактеристикиПК.Наименование = название;
            ХарактеристикиПК.Владелец = НовЭл.Ссылка;
            ХарактеристикиПК.Записать();

            Комплектующие.СоздатьПК(ХарактеристикиПК.Ссылка,
                  ПланыВидовХарактеристик.СвойстваНоменклатуры.Процессор,
```

```
                  процессор);

            Комплектующие.СоздатьПК(ХарактеристикиПК.Ссылка,
                  ПланыВидовХарактеристик.СвойстваНоменклатуры.МатеринскаяПлата,
                  матПлата);

            Комплектующие.СоздатьПК(ХарактеристикиПК.Ссылка,
                  ПланыВидовХарактеристик.СвойстваНоменклатуры.Видеокарта,
                  видео);

            Комплектующие.СоздатьПК(ХарактеристикиПК.Ссылка,

      ПланыВидовХарактеристик.СвойстваНоменклатуры.ОперативнаяПамять,
                  озу);

            Комплектующие.СоздатьПК(ХарактеристикиПК.Ссылка,
                  ПланыВидовХарактеристик.СвойстваНоменклатуры.ХранениеДанных,
                  хранениеДанных);

            Комплектующие.СоздатьПК(ХарактеристикиПК.Ссылка,
                  ПланыВидовХарактеристик.СвойстваНоменклатуры.БлокПитания,
                  блокПитания);


            Сообщить(название+" добавлен");

            ПК = НовЭл.Ссылка;
            Характеристика = ХарактеристикиПК.Ссылка;
      КонецЕсли;
      КонфигурацияВерна = Истина;
      возврат Истина;
КонецФункции
```

Листинг Ж.14 – Общий модуль «Комплектующие»

```
Процедура ВызватьОшибку(сообщение) Экспорт
      ВызватьИсключение сообщение;
КонецПроцедуры

Функция ПолучитьВидеокартаИнфо(имя) Экспорт
      элемент = Справочники.Видеокарты.НайтиПоНаименованию(имя);
      инфо = Новый Структура;
    инфо.Вставить("Энергопотребление", элемент.Энергопотребление);
    инфо.Вставить("РазъемПитания", элемент.РазъемПитания);

      возврат инфо;
КонецФункции

Функция ПолучитьПроцессорИнфо(имя) Экспорт
      элемент = Справочники.Процессоры.НайтиПоНаименованию(имя);
      инфо = Новый Структура;
    инфо.Вставить("Сокет", элемент.Сокет);
    инфо.Вставить("ТипПамяти", элемент.ТипПамяти);
      инфо.Вставить("Энергопотребление", элемент.Энергопотребление);

      возврат инфо;
КонецФункции

Функция ПолучитьХранениеДанныхИнфо(имя) Экспорт
      элемент = Справочники.ХранениеДанных.НайтиПоНаименованию(имя);
      инфо = Новый Структура;
```

```
      инфо.Вставить("Энергопотребление", элемент.Энергопотребление);

      возврат инфо;
КонецФункции

Функция ПолучитьМатПлатаИнфо(имя) Экспорт
      плата = Справочники.МатеринскиеПлаты.НайтиПоНаименованию(имя);
      платаИнфо = Новый Структура;
    платаИнфо.Вставить("Сокет", плата.Сокет);
    платаИнфо.Вставить("ТипПамяти", плата.ТипПамяти);

      возврат платаИнфо;
КонецФункции

Процедура СоздатьХарактеристику(характеристика, свойство, значение) Экспорт
      Свойства = РегистрыСведений.ЗначенияХарактеристик.СоздатьМенеджерЗаписи();
      Свойства.Характеристика = характеристика;
      Свойства.Свойство = свойство;
      Свойства.Значение = значение;
      Свойства.Записать();
КонецПроцедуры
```

ПРИЛОЖЕНИЕ И
РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ
(рекомендуемое)

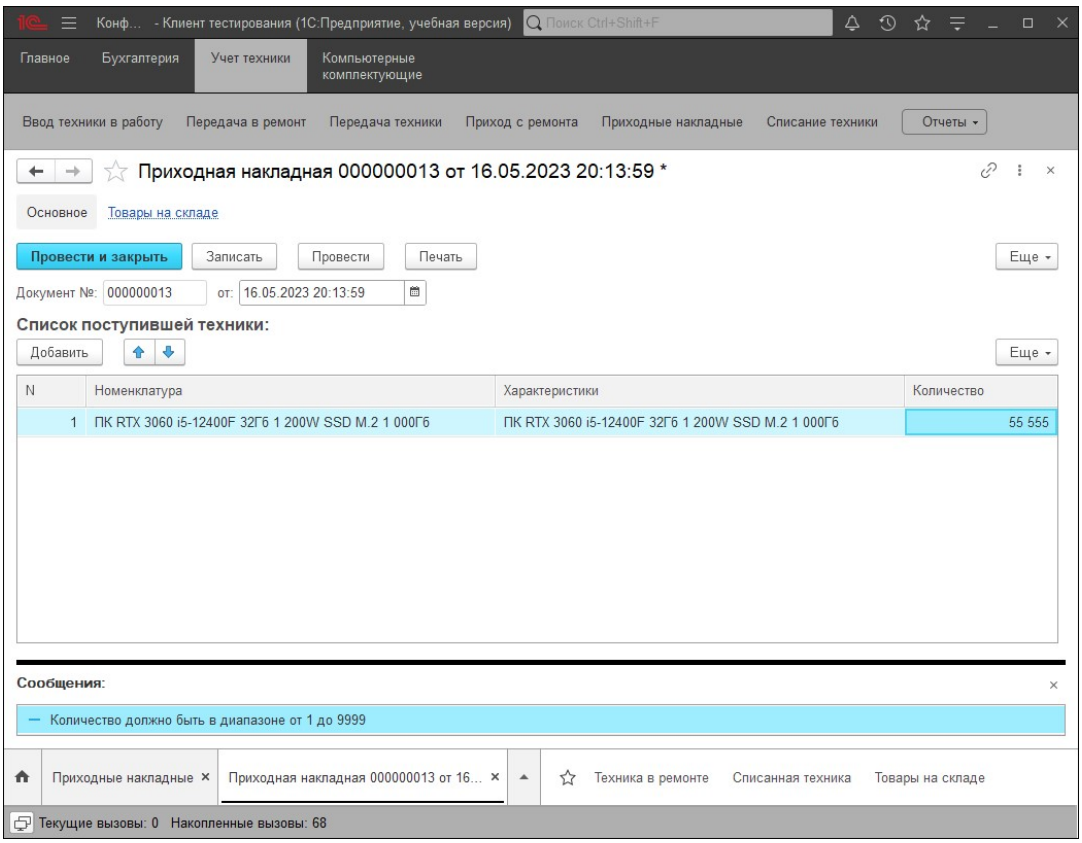


Рисунок И.1 – Результат теста №2 таблицы решений

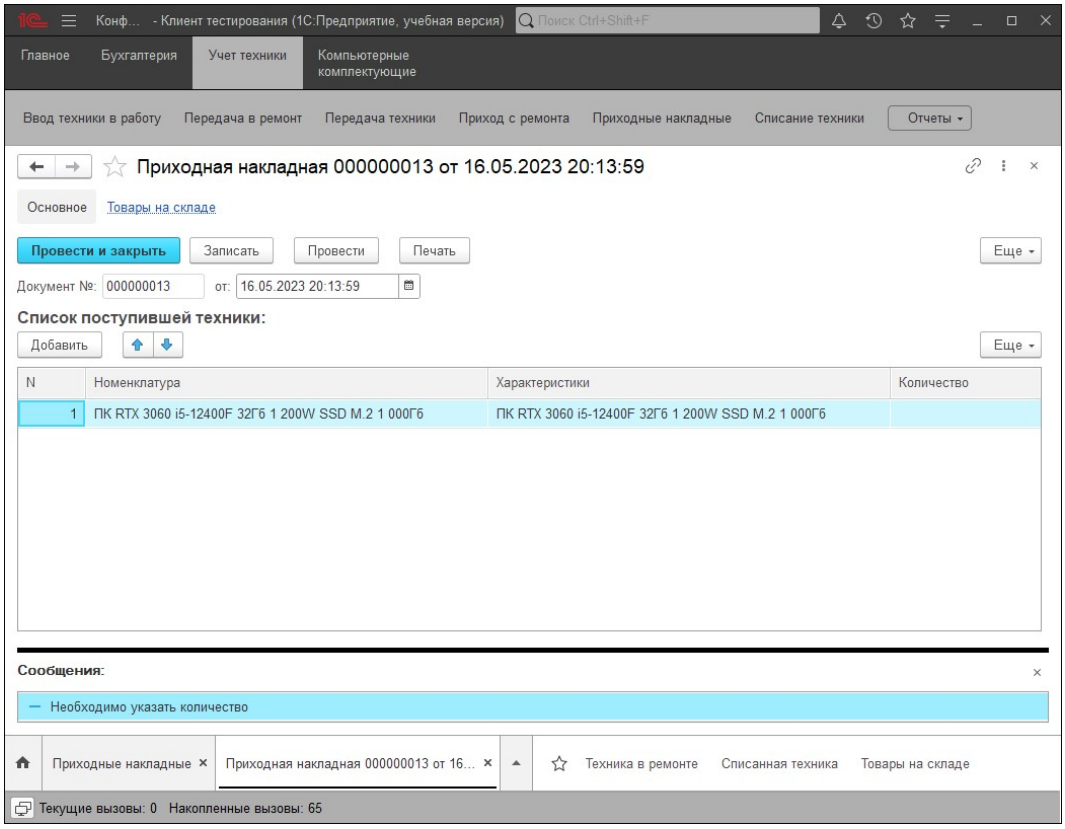


Рисунок И.2 – Результат теста №3 таблицы решений

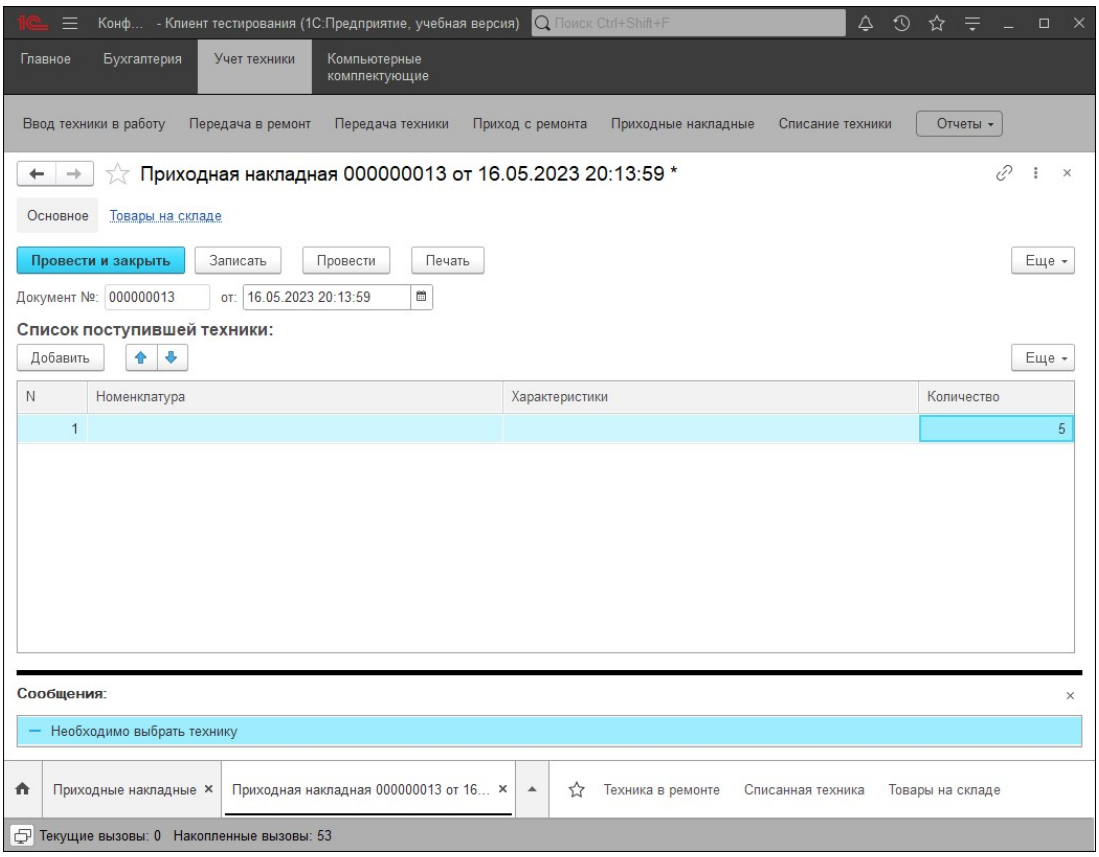


Рисунок И.3 – Результат теста №4 таблицы решений

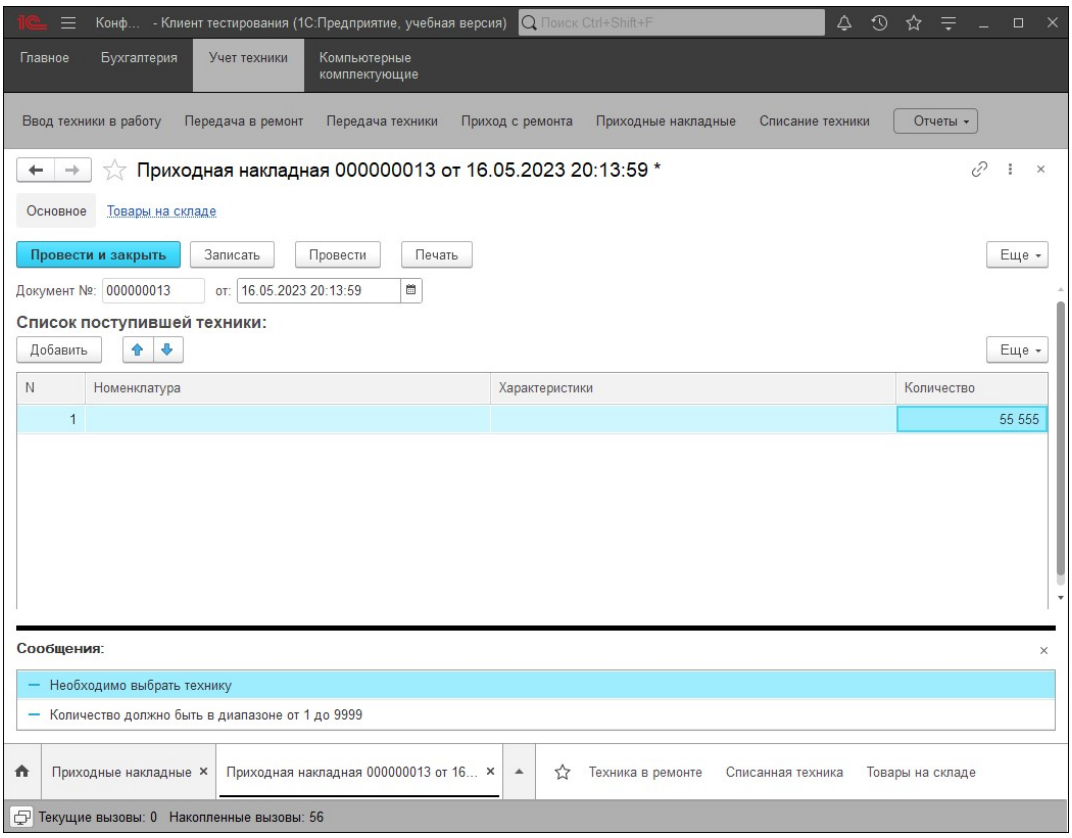


Рисунок И.4 – Результат теста №5 таблицы решений

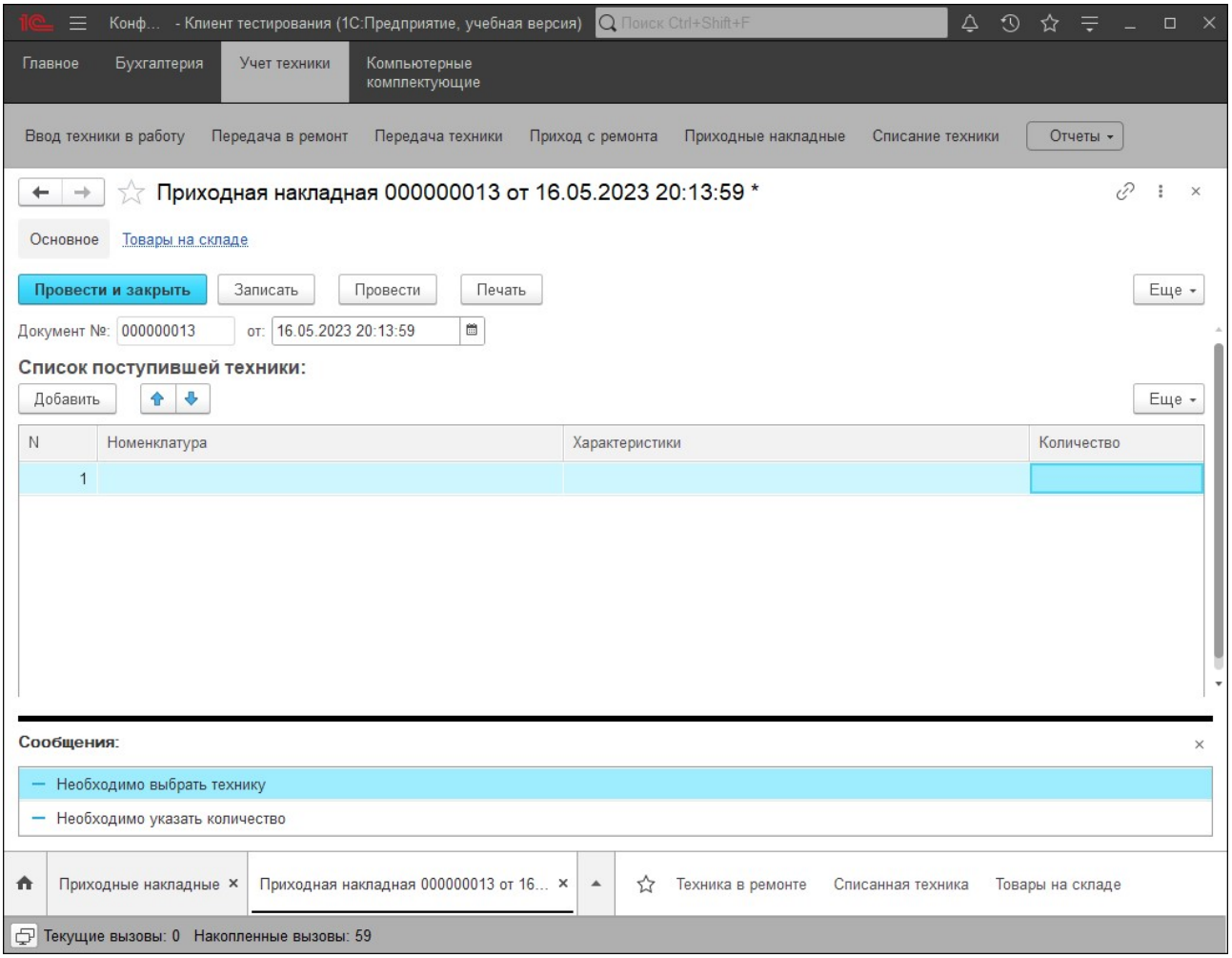


Рисунок И.5 – Результат теста №6 таблицы решений