МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет вычислительной техники

Кафедра «Вычислительная техника»

Направление подготовки: 09.03.01 «Информатика и вычислительная техника»

Профиль подготовки: «Вычислительные машины, комплексы, системы и сети»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

на тему

Распределенное клиент-серверное приложение с удаленной базой данных

Студент 03.06.2020 (подпись, дата) Потапова Регина Евгеньевна (ФИО полностью)

Руководитель 03.06.20 (подпись, дата) Карамышева Н.С. (фамилия, инициалы)

Нормоконтролёр (подпись, дата) Кучин А.В. (фамилия, инициалы)

Работа допущена к защите (протокол заседания кафедры от 03.06.2020 № 12)

Заведующий кафедрой (подпись) Митрохин М.А. (фамилия, инициалы)

Работа защищена с отметкой хорошо (протокол заседания ГЭК от 17.06.2020 № 12)

Секретарь ГЭК (подпись) Гурин Е.И. (фамилия, инициалы)

Пенза, 2020

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет
вычислительной техники

Кафедра
«Вычислительная техника»

«Утверждаю»
Заведующий кафедрой
_____ М.А. Митрохин
«30» апреля 2020 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

1. Студент Потапова Регина Евгеньевна гр. 16ВВ1 факультета ВТ

2. Тема работы Распределенное клиент-серверное приложение с удаленной базой данных

Тема утверждена приказом ПГУ №0335а/о от "30" апреля 2020 г.

3. Руководитель бакалаврской работы доцент кафедры «Вычислительная техника» Карамышева Н.С.

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

1) Назначение: распределенное клиент-серверное приложение предназначено для автоматизации работы сотрудников, участвующих в процессе учета ТМЦ на предприятии и в процессе инвентаризации.

2) Среда выполнения: операционная система Windows, начиная с версии 7.

Средства разработки: языки программирования Java Script, PHP, SQL, язык разметки HTML и язык стилей CSS; инструментальные средства «Notepad++», «Oracle».

Технологии разработки: RUP, JSON, Ext JS.

3) Основные функции распределенного клиент-серверного приложения:

а) Ведение учета остатков ТМЦ, партий ТМЦ, приказов инвентаризации, результатов корректировки остатков ТМЦ, бухгалтерских проводок.

б) Модификация записей.

в) Просмотр данных об остатках ТМЦ, приказах инвентаризации, результатах

корректировки остатков ТМЦ, бухгалтерских проводках.

д) Обновление данных.

е) Аутентификация и авторизация в системе.

ж) Поиск записей.

5. Перечень подлежащих разработке вопросов

1) Анализ предметной области

2) Анализ требований

3) Проектирование

4) Реализация

5) Тестирование

6) План разработки и оценка бюджета

7) Безопасность жизнедеятельности

6. Календарный график выполнения работы

№ п/п	Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя
1	Анализ предметной области и требований	20	30.04.20-4.05.20	
2	Проектирование	14	5.05.20-10.05.20	
3	Реализация	5	11.05.20-27.05.20	
4	Контроль качества	5	27.05.20-30.05.20	
5	План разработки бюджета	8	30.05.20-31.05.20	
6	Безопасность жизнедеятельности	6	01.06.20-2.06.20	
7	Оформление пояснительной записки	10	03.06.20-5.06.20	

Дата выдачи задания "30" апреля 2020 г.

Руководитель бакалаврской работы _____ 30.04.20
(подпись, дата)

Задание к исполнению принял студент _____ 30.04.2020
(подпись, дата)

Работу к защите допустить

Декан факультета _____
(подпись, дата)

Реферат

Пояснительная записка содержит 106 листов, 37 рисунков, 6 таблиц, 28 использованных источника, 4 приложения.

RUP, JSON, EXT JS, JAVASCRIPT, PHP, БАЗА ДАННЫХ, УЧЕТ МАТЕРИАЛЬНО-ПРОИЗВОДСТВЕННЫХ ЗАПАСОВ ПРЕДПРИЯТИЯ, ФОРМИРОВАНИЕ ИНВЕНТАРИЗАЦИОННЫХ ОПИСЕЙ, ИНВЕНТАРИЗАЦИЯ, IDEF1X, UML, SQL, ORACLE, СУБД, WEB-ПРИЛОЖЕНИЕ, ИНФОРМАЦИОННАЯ СИСТЕМА

Объектом разработки является распределенное-клиент серверное приложение с удаленной базой данных для учета материально-производственных запасов предприятия и формирования инвентаризационных описей.

Цель выпускной квалификационной работы – разработать распределенное-клиент серверное приложение с удаленной базой данных, позволяющее автоматизировать работу сотрудников бухгалтерии и материально-ответственных лиц.

Технология разработки – RUP, JSON, Ext JS.

Результат работы – распределенное-клиент серверное приложение с удаленной базой данных, отвечающее всем требованиям технического задания.

Для разработки программного продукта применялись языки JavaScript, PHP, HTML, CSS, SQL, текстовый редактор «Notepad++», система управления базами данных Oracle и операционная система Windows 7.

Содержание

Введение 6
1 Анализ предметной области и требований 8
1.1 Анализ предметной области и постановка задачи 8
1.1.1 Анализ программ аналогов 9
1.1.1.1 Анализ «1С:Бухгалтерия» 9
1.1.1.2 Анализ «Инвентаризация» 10
1.1.1.3 Анализ «Складской учет» 11
1.1.1.4 Вывод 12
1.1.2 Постановка задачи 13
1.1.2.1 Назначение 13
1.1.2.2 Предполагаемая аудитория 14
1.1.2.3 Общий взгляд на продукт 14
1.1.2.4 Предположения и зависимости 14
1.1.3 Выбор технологий разработки 14
1.1.3.1 Методология RUP 14
1.1.3.2 Технология JSON 15
1.1.3.3 Framework ExtJS 16
1.1.4 Выбор инструментальных средств 17
1.2 Анализ требований 18
1.2.1 Функциональные требования 18
1.2.2 Нефункциональные требования 23
1.2.2.1 Интерфейсы пользователей 23
1.2.2.2 Требования к производительности 23
1.2.2.3 Требования к безопасности 23
2 Проектирование 24
2.1 Базовая архитектура системы 24
2.2 Логическая модель базы данных 25
2.2.1 Первая нормальная форма (1NF) 25

2.2.2 Вторая нормальная форма (2NF) 26
2.2.3 Третья нормальная форма (3NF) 26
2.3 Пользовательский интерфейс клиентского приложения 27
2.4 Планирование разработки и оценка бюджета 30
2.4.1 Планирование проекта 31
3 Реализация 38
3.1 Физическая модель базы данных 38
3.2 Серверная часть приложения 38
3.2.1 Реализация представлений 39
3.3 Модель реализации клиентского приложения 40
4 Контроль качества программного обеспечения 43
4.1 Функциональное тестирование 43
5 Безопасность жизнедеятельности 48
5.1 Характеристика условий труда программиста 49
5.2 Требования к производственным помещениям 49
5.2.1 Освещение 49
5.2.2 Параметры микроклимата 51
5.2.3 Шум и вибрация 52
5.2.4 Электромагнитное и ионизирующее излучения 52
5.3 Эргономические требования к рабочему месту 53
Заключение 54
Список использованных источников 55
Приложение А. Глоссарий 57
Приложение Б. Листинг клиентского приложения 60
Приложение В. Результаты тестирования 85
Приложение Г. Листинг скрипта DDL 92
Приложение Д. Презентация 96

Введение

Интенсивное и полномасштабное развитие информационных технологий в России является одним из наиболее важных и приоритетных направлений государственной политики. В частности, принята и реализуется Государственной программа Российской Федерации «Информационное общество» с периодом действия на 2011-2020 годы [1].

Цель программы – повысить конкурентоспособность экономики страны путём снижения издержек и улучшения качества продукции, повышение значимости государственного управления, обеспечение роста качества жизни населения.

Развитие информационных технологий коснулось и такой сферы как ведение и учет ТМЦ. Автоматизированные информационные системы, которые используются в данной сфере помогают контролировать материально-производственные запасы (МПЗ), а также их движение по территории предприятия. Такие системы учета строятся на основе учета материала, который использует конкретное предприятие.

В настоящее время на территории Российской Федерации существует огромное количество предприятий, которые используют партионный учет. Таким предприятиям не выгодно демонстрировать, как именно они организовали структуру бухгалтерского учета, структуру партионного учета и какой программный комплекс они разработали для учета материала и взаимодействия отделов (подразделений) предприятия [2].

На сегодняшний день на рынке программного обеспечения имеется лишь несколько автоматизированных информационных систем складского учета с инвентаризацией по приходу (по партиям). К таким информационным системам относятся, например, «1С:Бухгалтерия», «Инвентаризация», «Складской учет».

Поэтому актуальной является задача разработки собственной автоматизированной информационной системы, которая поможет автоматизировать процесс инвентаризации на предприятии.

Таким образом, на основе вышеизложенного можно сделать вывод, что тема выпускной квалификационной работы является актуальной.

Целью данной работы является разработка распределенного клиент-серверного приложения с удаленной базой данных, где будут автоматизированы учет материально-производственных запасов предприятия и формирование инвентаризационных описей.

1 Анализ предметной области и требований

1.1 Анализ предметной области и постановка задачи

Инвентаризация – один из элементов метода бухгалтерского учета, обеспечивающий достоверность учетных данных путем сверки фактических остатков ценностей и расчетов с данными учета и осуществления контроля за сохранностью собственности. Инвентаризация имеет очень важное контрольное значение и выступает в качестве необходимого дополнения к документации хозяйственных операций. Она служит средством не только вскрытия и выявления недостач и злоупотреблений, но и предупреждения их в дальнейшем [3].

Процесс инвентаризации сводится к выявлению излишков и недостач материально-ответственным лицом.

Инвентаризация может быть плановая (один раз в год) и внеочередная. Опишем процесс плановой инвентаризации по пунктам.

1) Глава организации издает подписанный приказ о проведении инвентаризации, где обозначает сроки сверки ТМЦ и сроки сдачи документов в главную бухгалтерию.

2) Для проведения инвентаризации в организации создается постоянно действующая инвентаризационная комиссия. Инвентаризационная комиссия обеспечивает полноту и точность внесения в описи данных о фактических остатках основных средств, запасов, товаров, денежных средств, другого имущества и финансовых обязательств, правильность и своевременность оформления материалов инвентаризации [4].

3) В период действия инвентаризации материально-ответственное лицо прекращает любые действия, связанные с оформлением прихода и расхода ТМЦ, так как это может отразится на соотнесении бухгалтерских остатков с фактическим наличием.

4) Материально-ответственное лицо производит сверку фактических остатков ценностей с данными учета. При выявлении излишков или недостач

ТМЦ материально-ответственное лицо отражает это в программном обеспечении путем фиксации изменений конкретной позиции.

5) После окончания сверки материально-ответственным лицом, формируется печать инвентаризационной описи по унифицированной форме. Инвентаризационные описи подписываются материально-ответственным лицом и инвентаризационной комиссией, затем сдаются в главную бухгалтерию для дальнейшей проверки.

6) Бухгалтер проверяет распечатанные инвентаризационные описи с данными в программном обеспечении и делает соответствующие изменения в балансе предприятия путем формирования проводок по дебету или кредиту на разницу, сформированную в учете материально-ответственным лицом.

7) После проверки бухгалтером и отражением изменений в балансе предприятия формируется печать сличительной ведомости по унифицированной форме N ИНВ-19, подписывается бухгалтером и директором предприятия, затем сдается в архив.

1.1.1 Анализ программ аналогов

Системы по учету ТМЦ и формированию инвентаризационных описей, такие как «1С:Бухгалтерия», «Инвентаризация» и «Складской учет» представляют собой клиент-серверное приложение.

1.1.1.1 Анализ «1С:Бухгалтерия»

«1С:Бухгалтерия» – это профессиональный инструмент бухгалтера, с помощью которого можно вести бухгалтерский и налоговый учет, готовить и сдавать обязательную отчетность [5].

Отраслевое решение «1С: Бухгалтерия» предназначено для автоматизации огромного количества процессов в бухгалтерском учете, в том числе и процесс инвентаризации.

На рисунке 1 представлено окно инвентаризации товаров в системе «1С:Бухгалтерия».

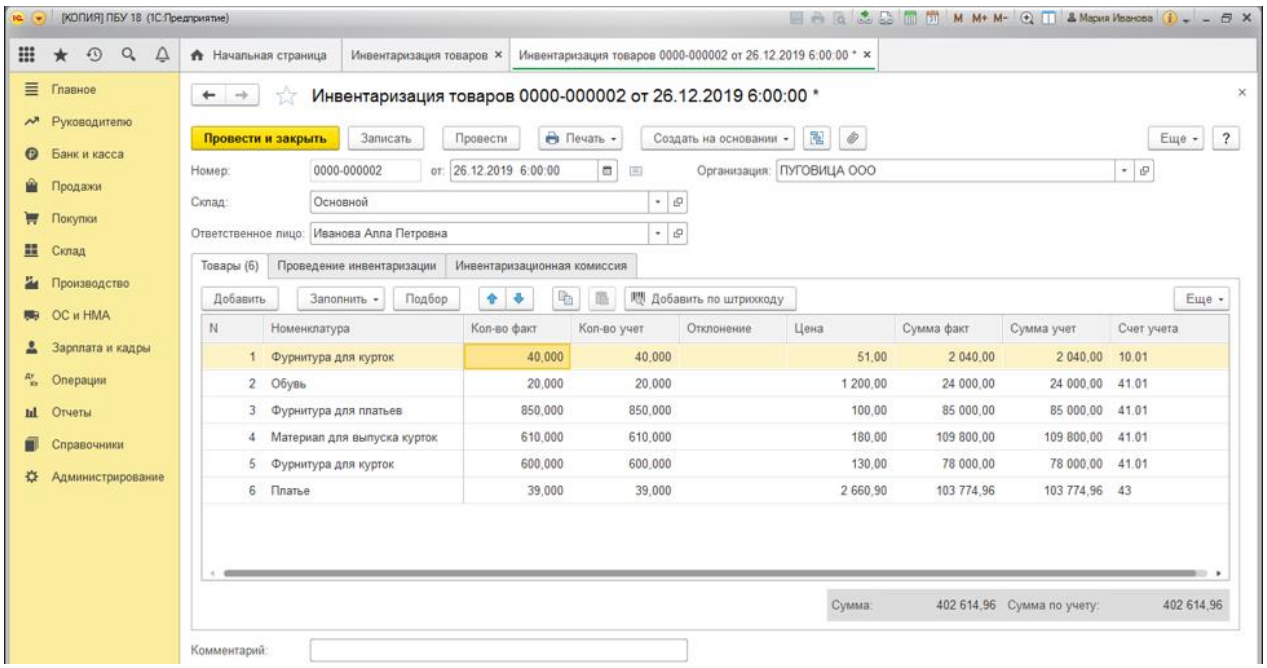


Рисунок 1 – Окно инвентаризации товаров в системе «1С:Бухгалтерия»

Программный продукт отличается большим функционалом, что наделяет ее высокой практической значимостью.

У «1С:Бухгалтерия», рассматривая конкретно процесс самой инвентаризации, имеется много достоинств, такие, как высокая функциональность, удобство и лаконичность интерфейса, набор дополнительных модулей программы для анализа. Отрицательными чертами данной программы являются недостаточность информации о партии для заведения новой позиции (например, «содержание драгоценных металлов в партии»), отсутствие печати в унифицированной форме N ИНВ-5, платный доступ, что влечет за собой возникновение больших убытков компании на закупку данного ПО под большой штат сотрудников, а также отсутствие функционала для материально-ответственных лиц – работать с данной программой могут только сотрудники бухгалтерии.

1.1.1.2 Анализ «Инвентаризация»

«Инвентаризация» – бесплатная программа для простого проведения инвентаризации. Ведет минимальный учет, реализует движение, списание, печать итогового отчета. Поддерживает различные фильтры для быстрого поиска [6].

Окно для заведения нового оборудования в системе «Инвентаризация» представлено на рисунке 2.

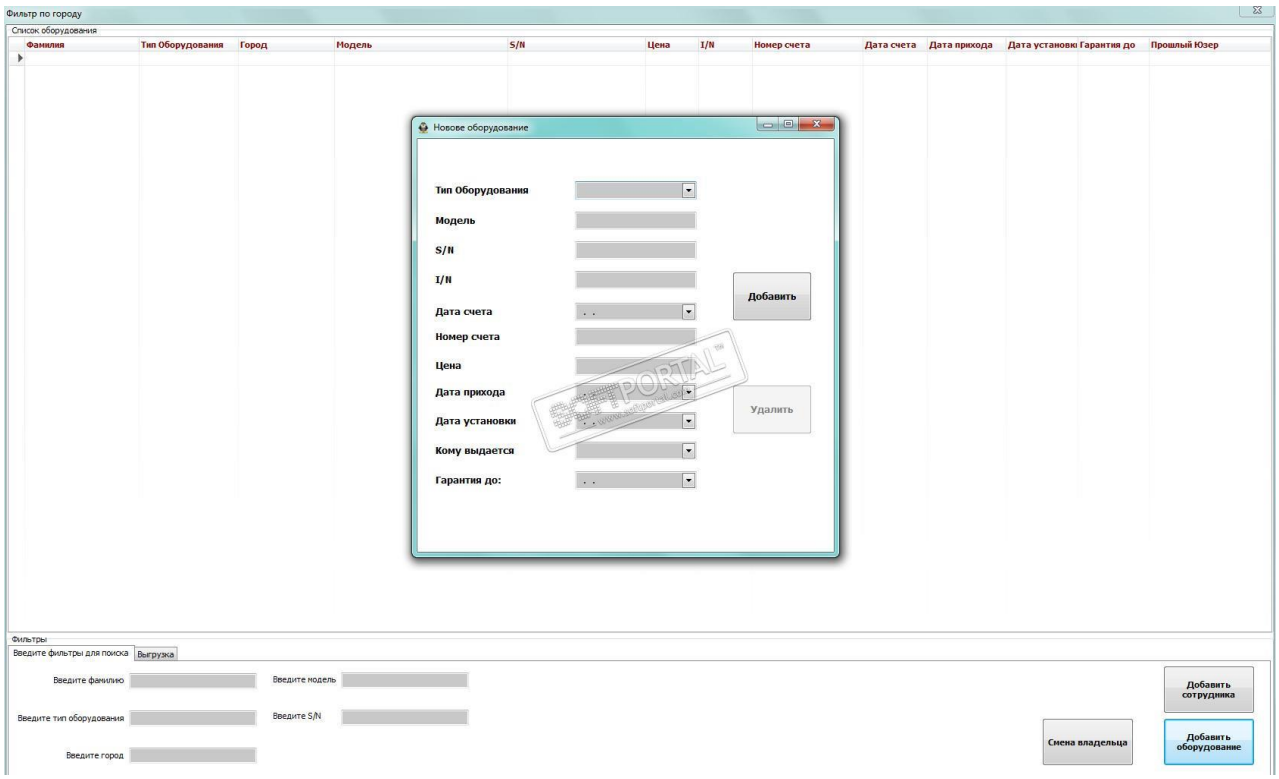


Рисунок 2 – Окно для заведения нового оборудования в системе «Инвентаризация»

В целом «Инвентаризация» – это быстрая и простая система для учета ТМЦ и ведения инвентаризации. Но данный программный продукт также не обделен недостатками, а именно:

– неудобный и не интуитивно понятный интерфейс программы;

– недостаточный функционал для работы с партиями ТМЦ;

– отсутствует функционал печати и выгрузки инвентаризационных описей.

1.1.1.3 Анализ «Складской учет»

Программа «Складской учет» предназначена для ведения аналитического учета товарно-материальных ценностей (ТМЦ). Аналитика ведется в разрезе складов (материально-ответственных лиц), субсчетов бухгалтерского учета и наименований ТМЦ. При вводе информации по приходному документу карточки складского учета автоматически формируются в картотеке.

Используется модель учета по фактической себестоимости единицы товарно-материальных ценностей [7].

Окно заведения для заведения новой карточки ТМЦ представлено на рисунке 3.

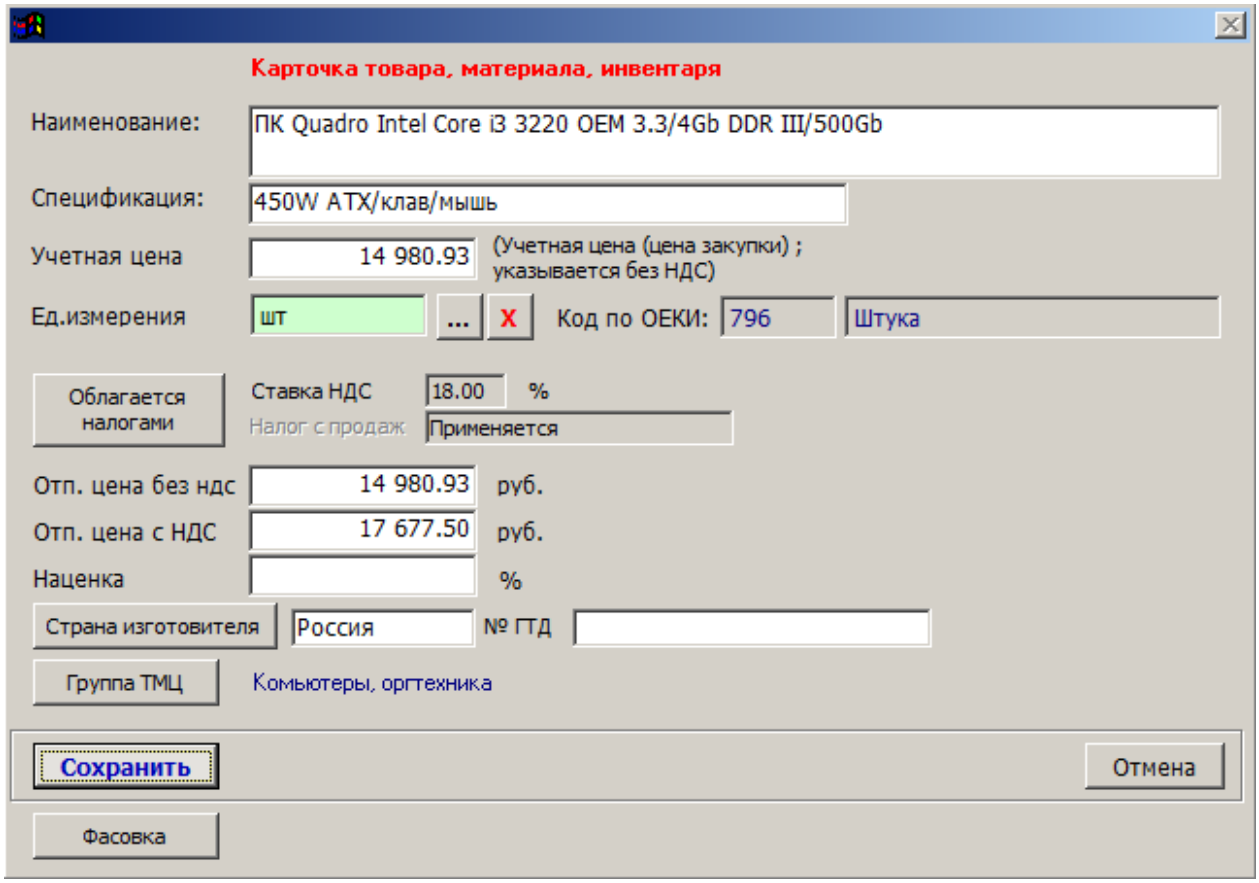


Рисунок 3 – Окно для заведения новой карточки ТМЦ в системе «Складской отчет»

Достоинства данного программного обеспечения следующие: удобный учет прихода, подсчет НДС. Данный программный продукт имеет следующие недостатки:

– устаревший интерфейс;
– платный доступ;
– отсутствие выгрузки данных в формат электронных таблиц.

1.1.1.4 Вывод

Таким образом, принято решение в ходе данной работы реализовать собственную автоматизированную информационную систему учета матери-

ально-производственных запасов предприятия и формирования инвентаризационных описей, лишенную вышеперечисленных недостатков, а именно: дороговизна, недостаточность информации о партии для заведения новой позиции, отсутствие функционала печати и выгрузки инвентаризационных описей по унифицированным формам, отсутствие функционала для материально-ответственных лиц в рамках работы с инвентаризацией, неудобство интерфейса, отсутствие возможности выгрузки в формат электронных таблиц. Также следует отметить, что программы-аналоги, которые существуют на рынке программного обеспечения, в основном предназначены для небольших предприятий, где учет ТМЦ ведется несколькими единицами штата, а разрабатываемое программное обеспечение – напротив, нацелено на большие предприятия, где имеют место быть десятки и сотни складов и широкий штат сотрудников.

Разрабатываемое распределенное клиент-серверное приложение с удаленной базой данных для автоматизации учета материально-производственных запасов предприятия и формирования инвентаризационных описей актуально для предприятий, которые хотят получить необходимый функционал, способствующий повышению качества и скорости работы с минимальными затратами.

1.1.2 Постановка задачи

1.1.2.1 Назначение

Данное клиент-серверное приложение предназначено для автоматизации работы сотрудников, участвующих в процессе учета ТМЦ на предприятии и в процессе инвентаризации. Система должна хранить и обрабатывать информацию об остатках ТМЦ, партиях, приказах инвентаризации, результатах корректировки остатков ТМЦ и о бухгалтерских проводках баланса. Система ориентирована на конечного пользователя, не обладающего высокой квалификацией в области вычислительной техники. Поэтому распределенное клиент-серверное приложение должно обладать простым, удобным и интуи-

тивно понятным интерфейсом. Также должна быть разработана база данных, которая будет обеспечивать надежное и безопасное хранение и доступ к информации.

1.1.2.2 Предполагаемая аудитория

Данное клиент-серверное приложение предназначено для бухгалтеров и материально-ответственных лиц, которые участвуют в процессе учета ТМЦ и в процессе инвентаризации.

1.1.2.3 Общий взгляд на продукт

Разрабатываемое программное обеспечение состоит из трех частей. Первая часть – это серверная база данных для хранения информации об остатках ТМЦ, доступе, партиях, приказах инвентаризации, результатах корректировки остатков ТМЦ и о бухгалтерских проводках баланса. Вторая часть – серверное web-приложение, позволяющее автоматизировать многие функции, связанные с учетом. Третья часть – клиентское web-приложение, управляющее интерфейсом пользователя.

1.1.2.4 Предположения и зависимости

Стабильность распределенного клиент-серверного приложения зависит от наличия подключения по локальной сети, а также от стабильности сервера, на котором находится база данных.

1.1.3 Выбор технологий разработки

1.1.3.1 Методология RUP

RUP (Rational Unified Process) – это методология разработки программного обеспечения. RUP обеспечивает строгий подход к назначению задач и ответственности в пределах группы разработки. Его цель состоит в том, чтобы гарантировать высокое качество программного продукта, отвечающего потребностям конечных пользователей, в пределах предсказуемого

временного графика и бюджета. Создавать современные сложные программные системы последовательно, т.е. сначала определять все проблемы, затем принимать все проективные решения, формировать программное обеспечение и, наконец, проверять изделие, невозможно. Итерационный подход позволяет улучшить понимание проблемы через последовательные усовершенствования и с приращением конкретизировать эффективные решения. Этот подход обеспечивает большую гибкость при учете новых требований или тактических изменений в деловых целях, и позволяет проекту заранее идентифицировать и разрешать риски [8].

1.1.3.2 Технология JSON

JSON (JavaScript Object Notation) – текстовый формат обмена данными, основанный на JavaScript. Как и многие другие текстовые форматы, JSON легко читается людьми. Несмотря на происхождение от JavaScript, формат считается независимым от языка и может использоваться практически с любым языком программирования. Для многих языков существует готовый код для создания и обработки данных в формате JSON. За счёт своей лаконичности по сравнению с XML, формат JSON может быть более подходящим для сериализации сложных структур. Если говорить о веб-приложениях, в таком ключе он уместен в задачах обмена данными как между браузером и сервером (AJAX), так и между самими серверами (программные HTTP-сопряжения). Поскольку формат JSON является подмножеством синтаксиса языка JavaScript, то он может быть быстро десериализован встроенной функцией eval(). Кроме того, возможна вставка вполне работоспособных JavaScript-функций. В языке PHP, начиная с версии 5.2.0, поддержка JSON включена в ядро в виде функций json_decode() и json_encode(), которые сами преобразуют типы данных JSON в соответствующие типы PHP и наоборот [9].

1.1.3.3 Framework ExtJS

В качестве фрейморка для разработки клиентского приложения был выбран Ext JS. Он представляет собой библиотеку JavaScript для разработки веб-приложений и пользовательских интерфейсов. И прежде всего Ext JS предназначен для создания RIA-приложений, то есть таких интернет-приложений, которые отличаются богатым и насыщенным интерфейсом. Ext JS помогает организовывать и управлять элементами на веб-странице и проводить более точную и эффективную их настройку. Причем данный фреймворк позволяет не только эффективно работать с элементами управления, но и с некоторыми графическими моментами, например, с диаграммами [10]. Каждый компонент в ExtJS позволяет себя конфигурировать путем настройки разнообразных опций, генерирует множество событий в ответ почти на любое изменение своего состояния, а также гибко настраивается через задание необходимых свойств оформления в CSS-стилях. Кстати о них. В архитектуре библиотеки есть и такое понятие как тема оформления – это обычный CSS-файл, изменяя который можно мгновенно изменить внешний вид всех элементов интерфейса. Уже создано множество различных пользовательских тем, в разной цветовой гамме и стиле, хотя, на мой взгляд, именно стандартная тема наиболее приятная и привычная для большинства пользователей. Так как любое веб-приложение основано на каких-то данных, то этот слой в библиотеке не менее насыщенный различным функционалом, чем визуальные компоненты. Обмен данными с сервером может вестись в двух форматах – JSON, как самый простой и родной для JavaScript, и XML, что позволяет реализовать более широкий функционал, но ценой ресурсоемкости всего приложения. Вся работа с данными построена на следующей компонентной схеме. Основной компонент Store – с ним и работают другие объекты, желающие получить или отправить данные. Reader отвечает за структуру данных и их верную интерпретацию, вне зависимости от формата. Proxy обеспечивает прозрачный доступ к серверу через промежуточную абстракцию Connection так, что приложению не важно, да и не надо даже знать,

как в реальности данные пришли от сервера. Объект (или набор объектов) Record хранит коллекцию записей из данных (согласно заданной структуре). Исходя из нужного вам функционала, вы можете использовать компоненты, которые работают с JSON-данными или XML, или даже и те, и другие вместе [11].

1.1.4 Выбор инструментальных средств

Для хранения информации об остатках ТМЦ, партиях, приказах инвентаризации, результатах корректировки остатков ТМЦ и о бухгалтерских проводках баланса было принято решение использовать реляционную базу данных. В качестве СУБД был выбран продукт от компании Oracle, который использует язык запросов SQL. Oracle – это система управления базами данных (СУБД) с открытым кодом. Это высокопроизводительная и масштабируемая СУБД с множеством программных интерфейсов. Она обладает огромными функциональными возможностями и подходит для решения самых разных задач. СУБД Oracle предлагает разработчикам практически неограниченные возможности выбора [12].

Для разработки программного обеспечения был выбран текстовый редактор «Notepad++». «Notepad++» – свободный текстовый редактор с открытым исходным кодом для Windows с подсветкой синтаксиса большого количества языков программирования и разметки, а также языков описания аппаратуры VHDL и Verilog. Поддерживает открытие более 100 форматов. Базируется на компоненте Scintilla, написан на C++ с использованием STL, а также Windows API и распространяется под лицензией GNU General Public License. Базовая функциональность программы может быть расширена как за счёт плагинов, так и сторонних модулей, таких как компиляторы и препроцессоры [13].

В качестве языков программирования были выбраны PHP и JavaScript. Данные языки программирования позволяют использовать современные концепции, технологии и библиотеки [14].

В качестве языков разметки были выбраны HTML и XSLT. XSLT – язык преобразования XML-документов. Спецификация XSLT входит в состав XSL и является рекомендацией W3C. При применении таблицы стилей XSLT, состоящей из набора шаблонов, к XML-документу образуется конечное дерево, которое может быть сериализовано в виде XML-документа, XHTML-документа, HTML-документа или простого текстового файла [15].

В качестве языка моделирования был выбран UML. UML – это универсальный язык визуального моделирования систем. UML объединил лучшие современные технические приемы моделирования и разработки программного обеспечения. UML-диаграммы легко воспринимаются и при этом без труда генерируются компьютерами [16].

Для проектирования UML-диаграмм использовался продукт Microsoft Office Visio 2013.

Логическое и физическое проектирование базы данных клиент-серверного приложения выполнялось при помощи case-средства ERWin Data Modeler.

Для планирования и распределения ресурсов по задачам был выбран продукт Microsoft Project 2013.

1.2 Анализ требований

Требование можно определить, как «подробное описание того, что должно быть реализовано». Существует два основных типа требований:

– функциональные требования – какое поведение должна предлагать система;

– нефункциональные требования – особое свойство или ограничение, накладываемое на систему.

1.2.1 Функциональные требования

Диаграммы вариантов использования используются для упрощения взаимодействия с будущими пользователями системы, и особенно пригодят-

ся для определения необходимых характеристик системы. Другими словами, диаграммы вариантов использования рассказывают о том, что система должна делать, не указывая применяемые методы.

В соответствии с поставленной задачей были построены две диаграммы вариантов использования.

Диаграммы вариантов использования представлены на рисунке 4 и 5.

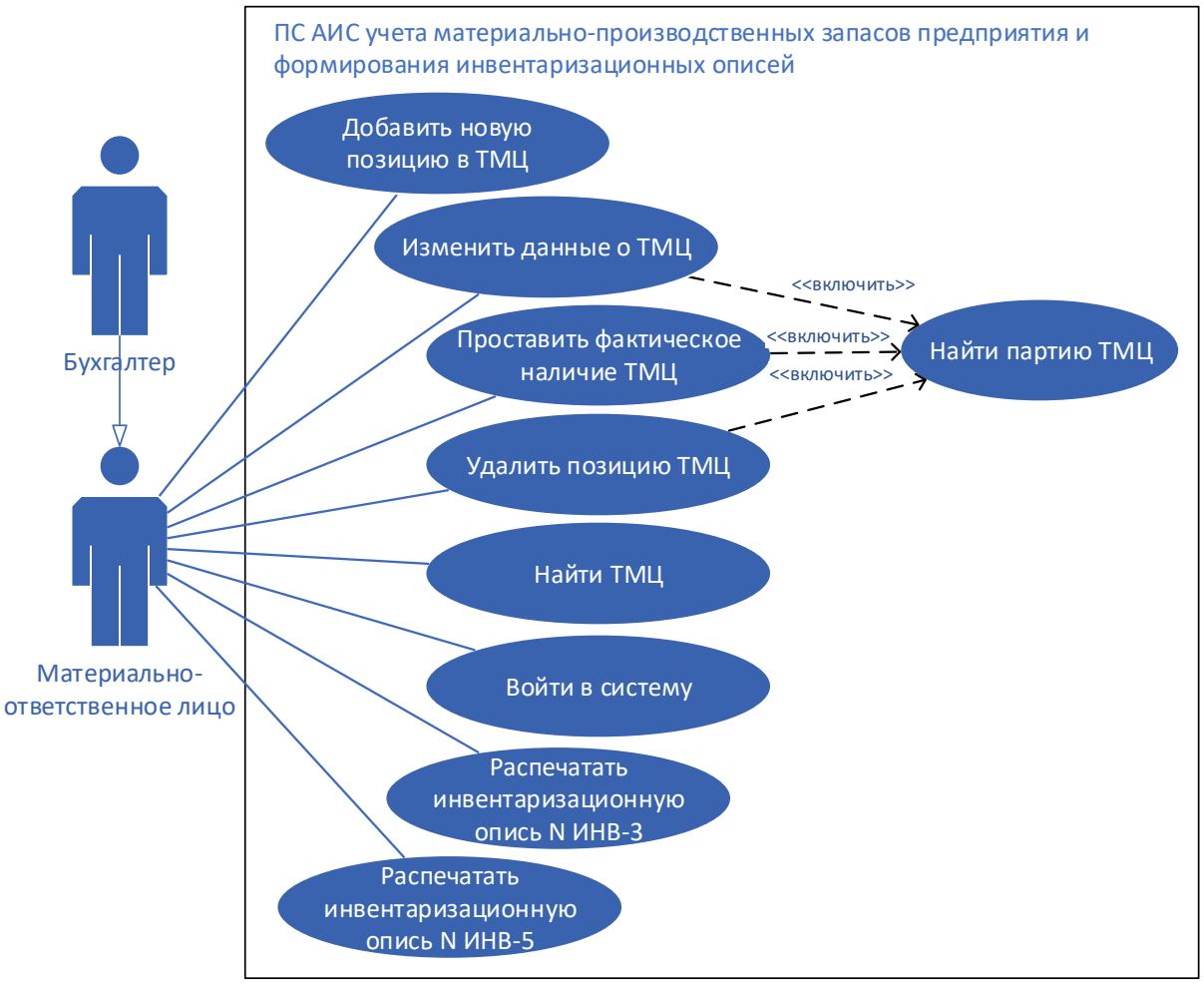


Рисунок 4 – Диаграмма вариантов использования (функции материально-ответственного лица)

Варианты использования «Изменить данные о ТМЦ», «Проставить фактическое наличие ТМЦ», «Удалить позицию ТМЦ» включают в себя вариант: «Найти партию ТМЦ».

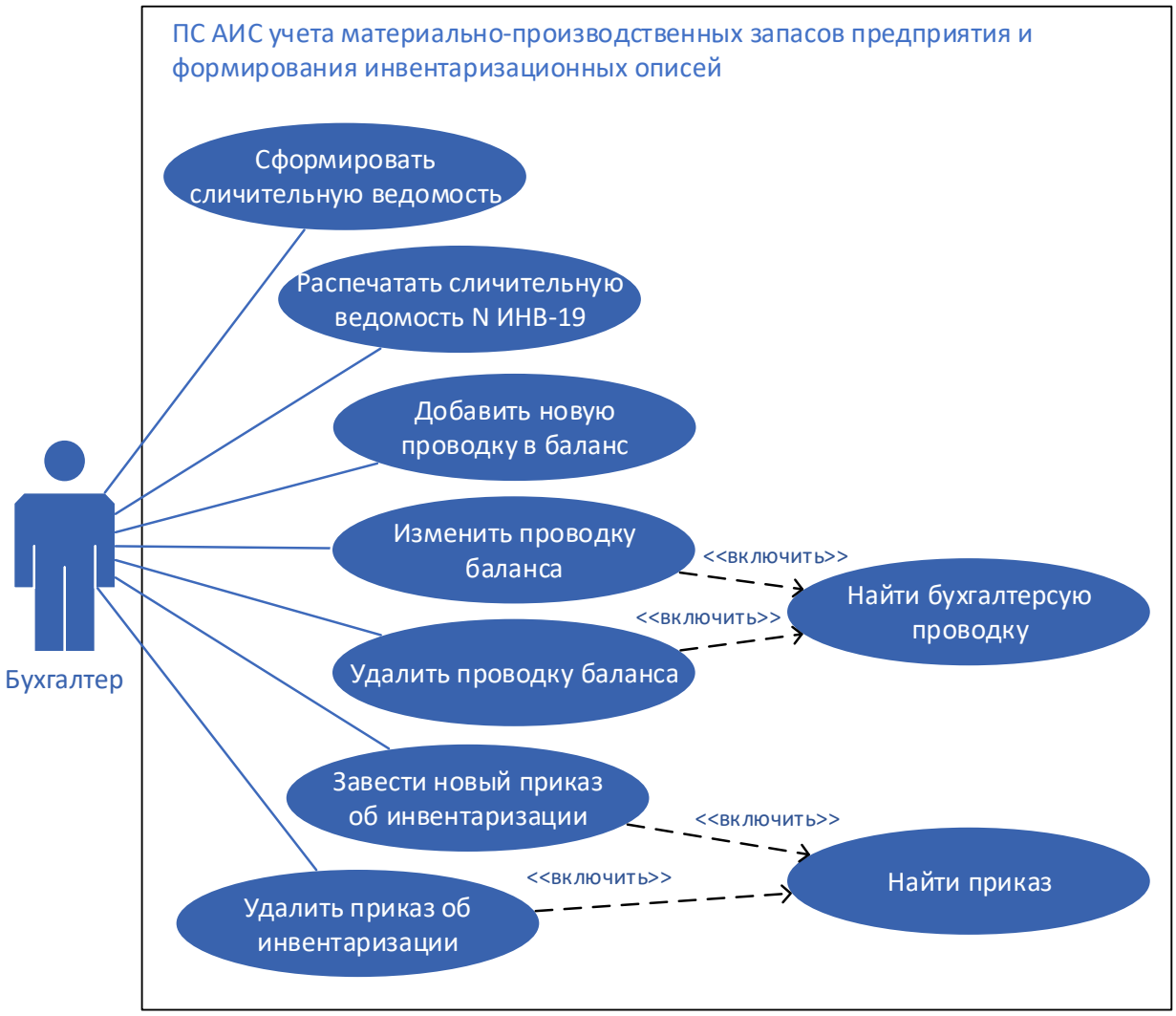


Рисунок 5 – Диаграмма вариантов использования (функции бухгалтера)

Варианты использования «Изменить проводку баланса» и «Удалить проводку баланса» включают в себя вариант «Найти бухгалтерскую проводку». Варианты использования «Завести новый приказ об инвентаризации» и «Удалить приказ об инвентаризации» включают в себя вариант «Найти приказ».

Для наглядного представления поведения вариантов использования используется диаграмма деятельности [17].

Диаграмма деятельности для варианта «Добавить новую позицию ТМЦ» представлена на рисунке 6.

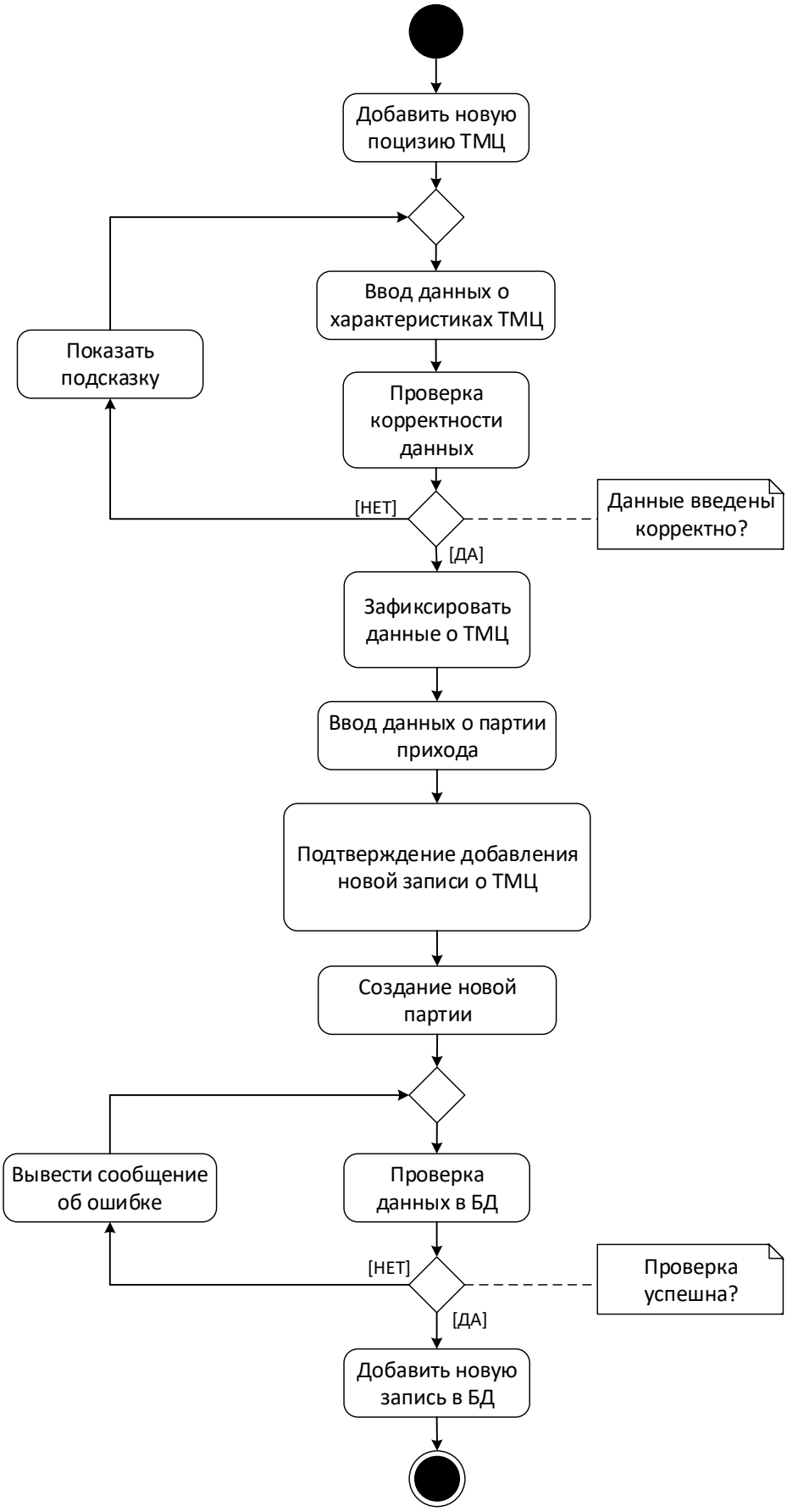


Рисунок 6 – Диаграмма деятельности (добавить новую позицию ТМЦ)

Диаграммы последовательности представляют взаимодействия между линиями жизни как упорядоченную последовательность событий. Это самая богатая и гибкая форма диаграммы воздействий [18].

Для рассмотрения взаимодействия между компонентами системы была спроектирована диаграмма последовательности для варианта «Добавить новую позицию ТМЦ» (рисунок 7).

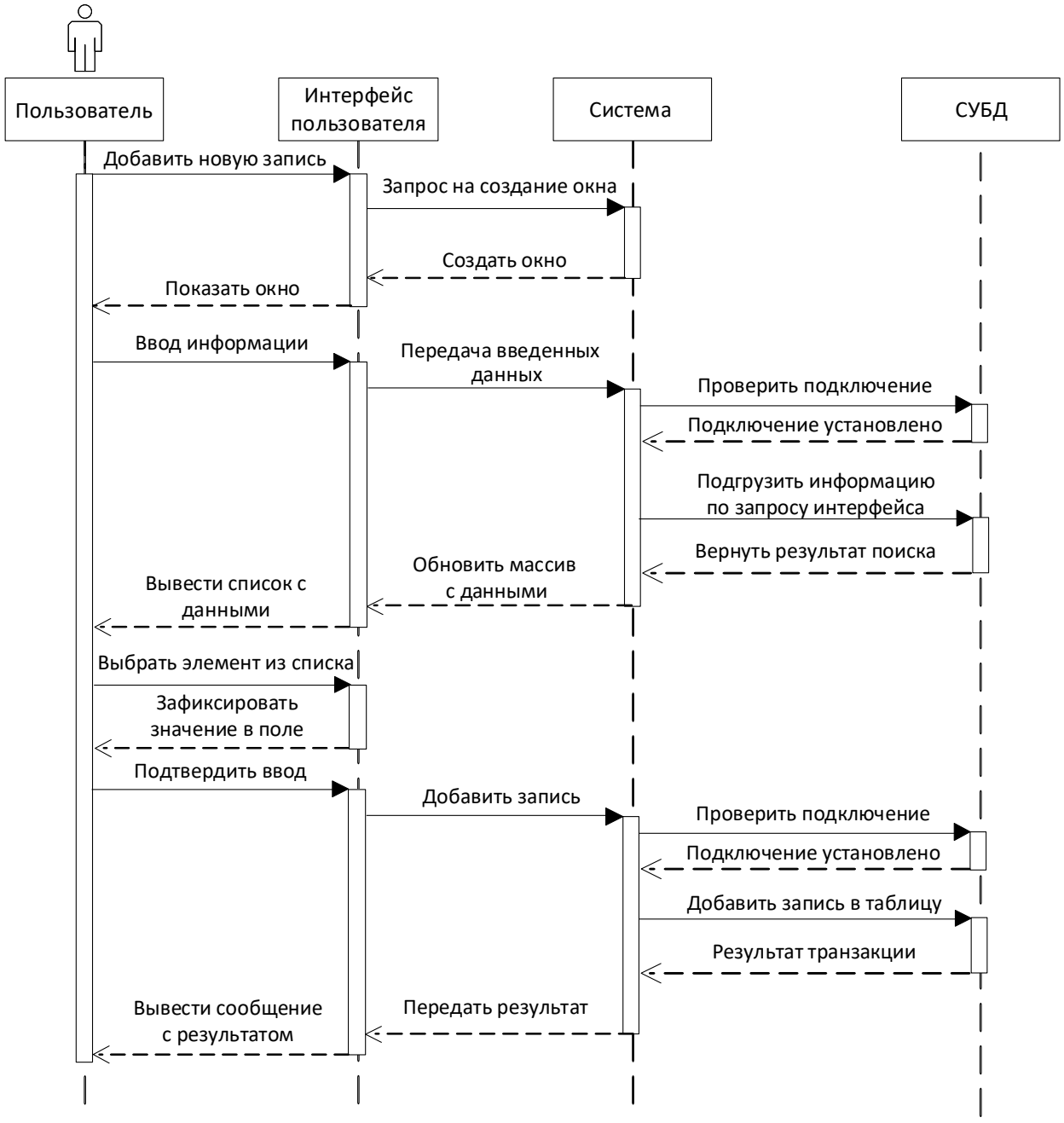


Рисунок 7 – Диаграмма последовательности для варианта «Добавить новую позицию ТМЦ»

На данной диаграмме последовательности (рисунок 7) изображены 4 компонента системы: пользователь, интерфейс пользователя, система и СУБД. Следует отметить, что под интерфейсом пользователя понимается оболочка, созданная при помощи библиотеки ExtJS.

1.2.2 Нефункциональные требования

Нефункциональные требования описывают как должна работать система или программный продукт, и какими свойствами или характеристиками она должна обладать.

1.2.2.1 Интерфейсы пользователей

Разрабатываемое распределенное клиент-серверное приложение для учета материально-производственных запасов предприятия и формирования инвентаризационных описей должно обладать удобным, простым и интуитивно понятным графическим интерфейсом пользователя.

1.2.2.2 Требования к производительности

Клиент-серверное приложение должно обеспечивать стабильную работу при наличии установленного стабильного соединения с сервером. Время ожидания данных от сервера не должно превышать двадцати секунд.

1.2.2.3 Требования к безопасности

Доступ к клиентской части приложения должен осуществляться при помощи аутентификации. А доступ аутентифицированного пользователя к определенным функциям системы должен осуществляться при помощи авторизации.

2 Проектирование

2.1 Базовая архитектура системы

Рассматриваемая автоматизированная информационная система имеет трехуровневую архитектуру клиент-сервер-сервер.

Архитектура автоматизированной информационной системы представлена на рисунке 8.

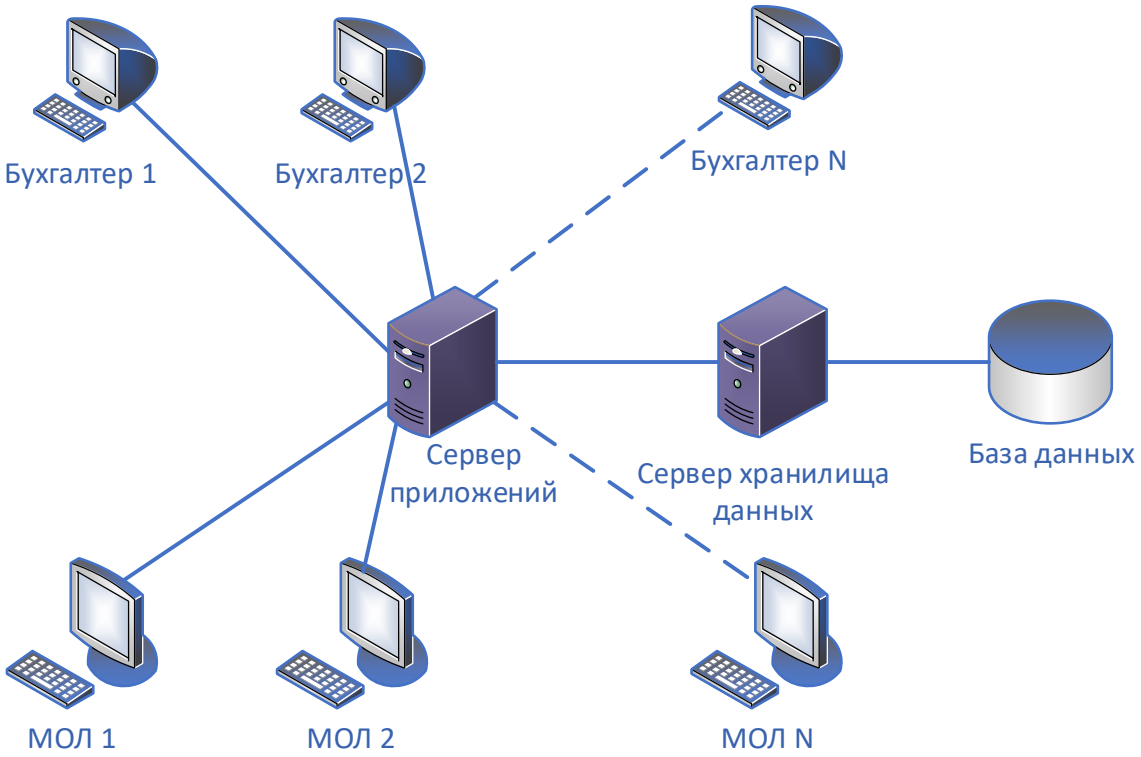


Рисунок 8 – Архитектура автоматизированной информационной системы

В данной архитектуре присутствуют три основных компонента системы. Первый компонент представляет собой станцию клиента, где запускаются некоторые из частей web-приложения. Второй компонент – web-приложение, располагающееся на сервере приложений, к которому удаленно со своих рабочих станций обращаются пользователи системы (бухгалтера и МОЛ). Третий компонент представляет собой север хранилища данных (сервер базы данных), который располагается на отдельной станции и хранит базу данных. Web-приложение, располагающееся на сервере приложений, по-

стоянно обращается к серверу хранилища данных для получения, добавления, корректировки и удаления данных из базы данных системы.

2.2 Логическая модель базы данных

Логическая модель строится на основе концептуальной модели с учетом выбранной модели данных, но не учитывая особенности целевой СУБД. Устранение избыточности модели производится, как правило, за счет декомпозиции отношений таким образом, чтобы в каждом отношении хранились только первичные факты. Процесс преобразования базы данных к виду, отвечающему нормальным формам, называется нормализацией.

Нормализация предназначена для приведения структуры базы данных к виду, обеспечивающему минимальную избыточность. Конечной целью нормализации является уменьшение потенциальной противоречивости хранимой в БД информации [19].

Нормальная форма – свойство отношения в реляционной модели данных, характеризующее его с точки зрения избыточности, которая потенциально может привести к логически ошибочным результатам выборки или изменения данных.

2.2.1 Первая нормальная форма (1NF)

Переменная отношения находится в первой нормальной форме (1НФ) тогда и только тогда**,** когда в любом допустимом значении отношения каждый его кортеж содержит только одно значение для каждого из атрибутов.

В реляционной модели отношение всегда находится в первой нормальной форме по определению понятия отношение. Что же касается различных таблиц, то они могут не быть правильными представлениями отношений и, соответственно, могут не находиться в 1НФ.

2.2.2 Вторая нормальная форма (2NF)

Переменная отношения находится во второй нормальной форме тогда и только тогда, когда она находится в первой нормальной форме, и каждый не ключевой атрибут зависит от ее потенциального ключа.

2.2.3 Третья нормальная форма (3NF)

Переменная отношения находится в третьей нормальной форме тогда и только тогда, когда она находится во второй нормальной форме, и отсутствуют транзитивные функциональные зависимости не ключевых атрибутов от ключевых.

Построим логическую модель базы данных в нотации IDEF1X при помощи case-средства ERwin Data Modeler. Логическая модель представлена на рисунке 9.

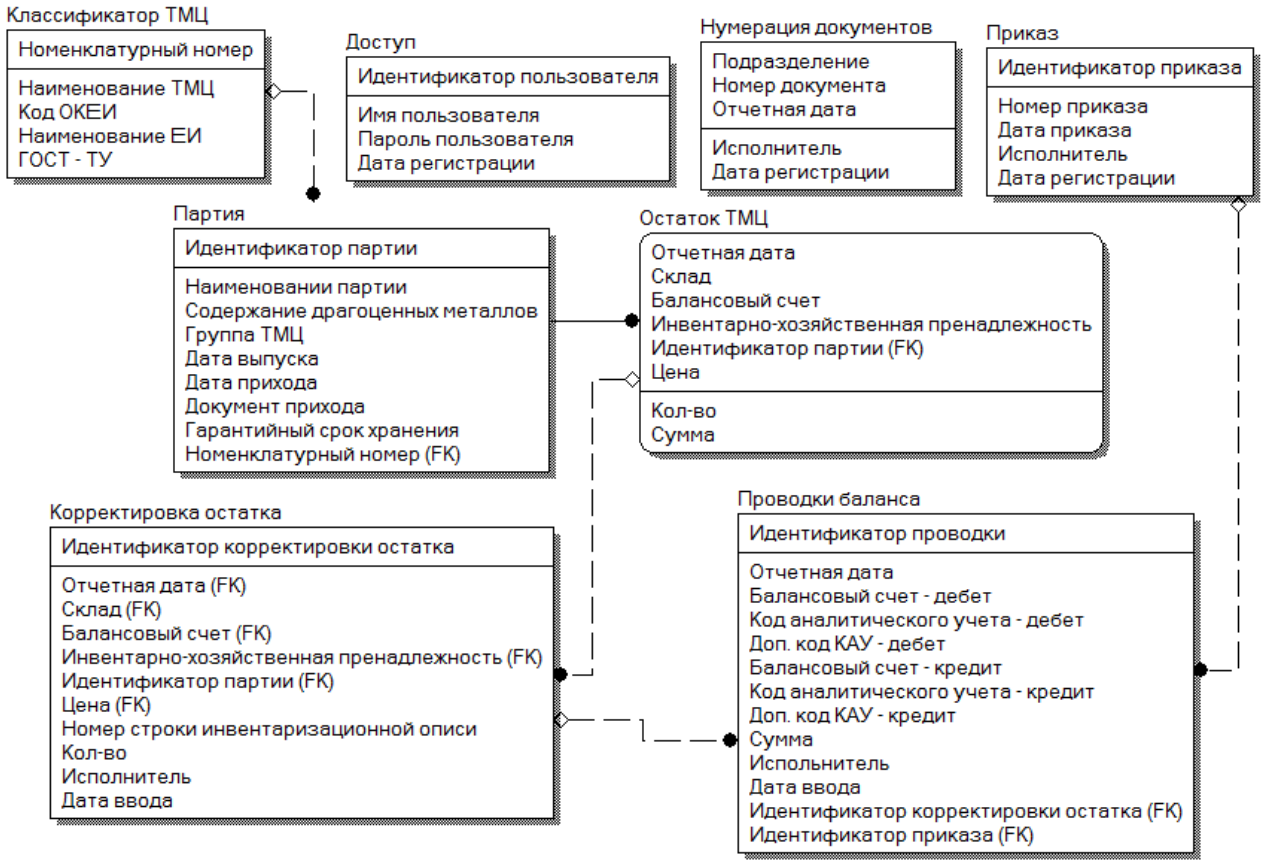


Рисунок 9 – Логическая модель базы данных

Разработанная логическая модель базы данных не содержит транзитивных зависимостей. Таким образом можно сделать вывод, что модель соответствует третьей нормальной форме.

2.3 Пользовательский интерфейс клиентского приложения

Исходя из диаграммы вариантов использования и этапа анализа требований был спроектирован простой и интуитивно понятный графический пользовательский интерфейс.

Графический пользовательский интерфейс был разработан с помощью графической библиотеки ExtJS 3.2.2. На рисунке 10 представлен графический интерфейс главного окна программы с функционалом для материально-ответственного лица.

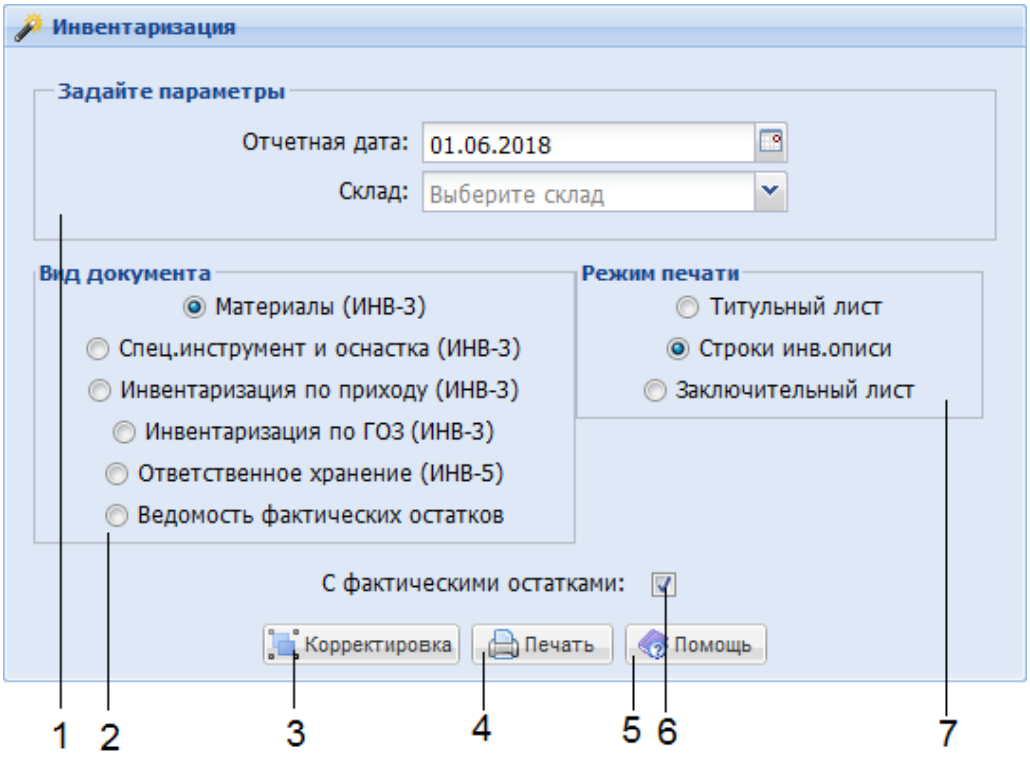


Рисунок 10 – Графический интерфейс главного окна программы (функции для материально-ответственного лица)

1 – область для ввода основных параметров для просмотра и корректировки;

2 – область радио-кнопок для указания вида документа;

3 – кнопка для просмотра и корректировки остатка в ИНВ-описи;

4 – кнопка для печати инвентаризационной описи;

5 – кнопка для просмотра инструкции пользователя;

6 – флаг для обозначения режима печати ИНВ-описи (с остатками или без);

7 – область радио-кнопок для указания части печати ИНВ-описи.

Данное окно представляет собой меню с функциями для работы с инвентаризационной описью.

Графический интерфейс окна для работы с инвентаризационной описью представлен на рисунке 11.

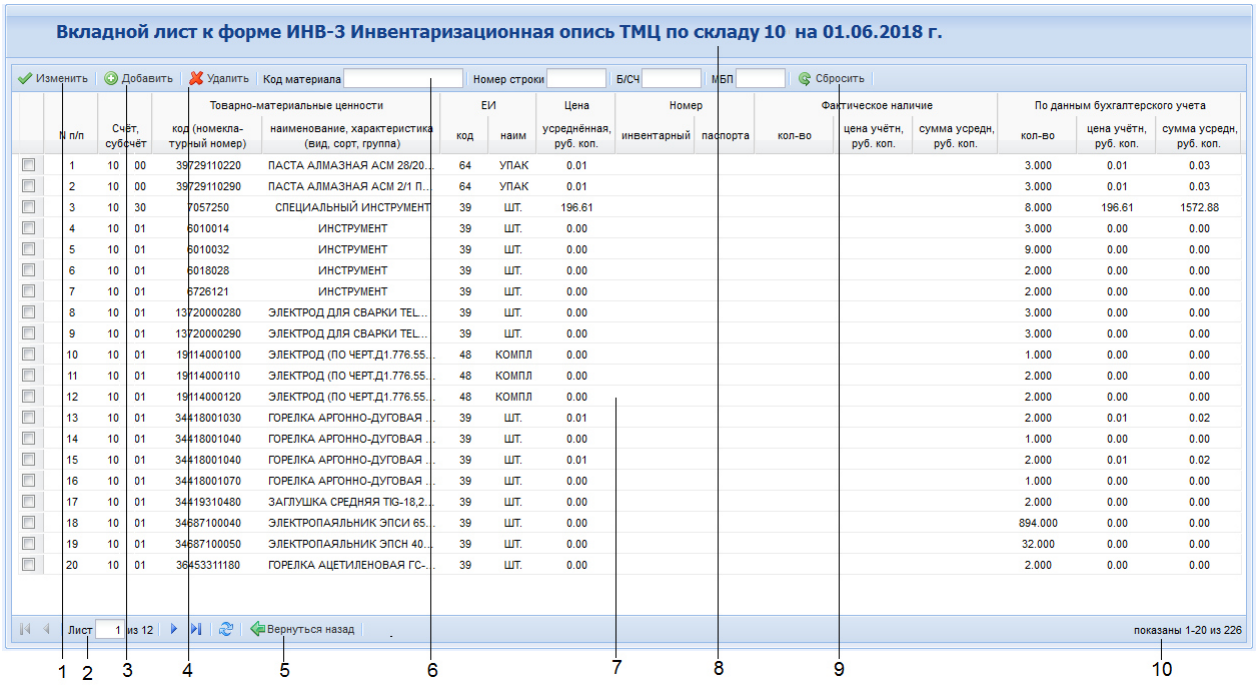


Рисунок 11 – Графический интерфейс окна для работы с инвентаризационной описью

1 – кнопка для корректировки строки инвентаризационной описи;

2 – панель управления навигацией по листам инвентаризационной описи;

3 – кнопка для добавления новой строки инвентаризационной описи (добавление нового остатка);

4 – кнопка для удаления ранее заведенной новой строки инвентаризационной описи;

5 – кнопка для возвращения к главному окну приложения;

6 – поля-фильтры для поиска определенных позиций;

7 – таблица для вывода строк инвентаризационной описи на экран пользователю;

8 – шапка окна программы с общей информацией по инвентаризационной описи;

9 – кнопка для удаления информации из полей-фильтров;

10 – информационное поле с данными о листах инвентаризационной описи.

Графический интерфейс окна корректировки строки инвентаризационной описи представлен на рисунке 12.

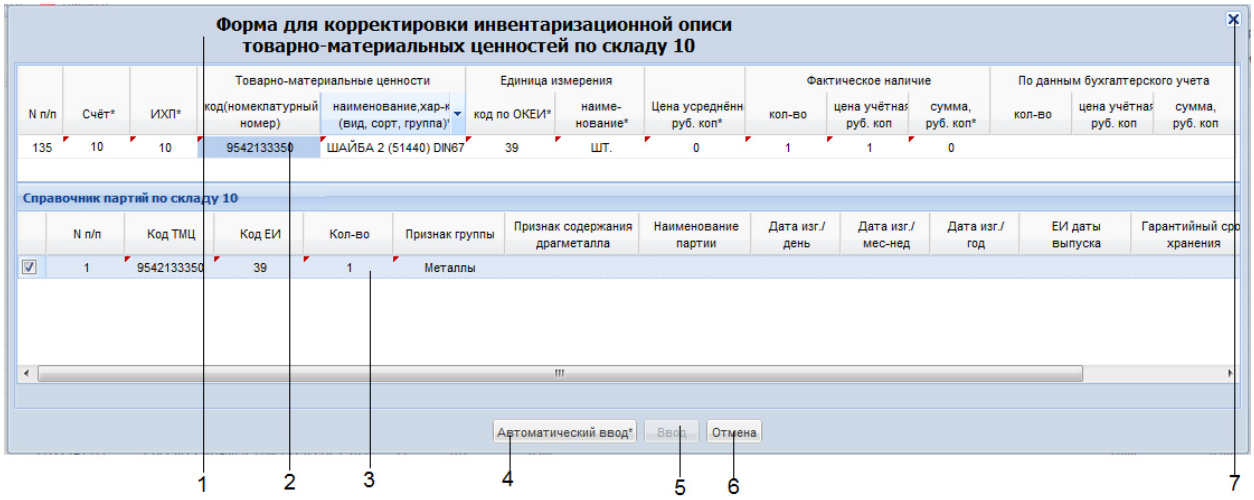


Рисунок 12 – Графический интерфейс окна корректировки строки инвентаризационной описи

1 – шапка для вывода общей информации об инвентаризационной описи;

2 – таблица для указания характеристик ТМЦ;

3 – таблица для работы с партиями ТМЦ;

4 – кнопка для автоматической подгрузки информации «по-умолчанию» в таблицу характеристик ТМЦ;

5 – кнопка для проверки и сохранения введенной информации в базу данных;

6 – кнопка для отмены операции;

7 – кнопка для закрытия данного окна программы.

Графический интерфейс главного окна программы с функционалом для бухгалтера представлен на рисунке 13.

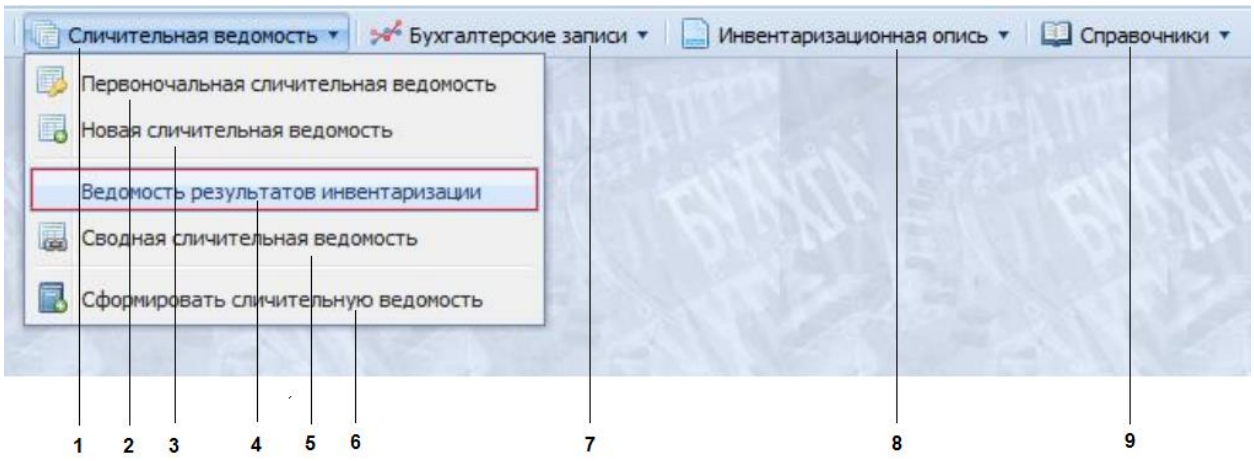


Рисунок 13 – Графический интерфейс главного окна программы (функции для бухгалтера)

1 – подпункт меню с функциями для работы со сличительной ведомостью;

2 – кнопка для просмотра первоначально сформированной ведомости;

3 – кнопка для просмотра сличительной ведомости после корректировки бухгалтером;

4 – кнопка для просмотра ведомости результатов инвентаризации;

5 – кнопка для просмотра сводного отчета по сличительной ведомости;

6 – кнопка для завершения инвентаризации материально-ответственного лица и формирования первоначальной сличительной ведомости бухгалтера;

7 – подпункт меню с функциями формирования и просмотра бухгалтерских проводок в баланс;

8 – подпункт меню с функциями для работы с печатными формами инвентаризационной описи;

9 – подпункт меню с функциями для работы со справочниками.

Оставшиеся окна приложения имеют схожий графический интерфейс.

Спроектированный графический пользовательский интерфейс является простым, удобным и интуитивно понятным.

2.4 Планирование разработки и оценка бюджета

Управление проектами – совокупность процессов по планированию, координации и контролю работ для реализации целей проектов с учетом

ограничений на ресурсы, бюджет и требований качества. В более широком смысле под управлением проектами понимается применение знаний, практического опыта, инструментальных средств и методов для удовлетворения потребностей заинтересованных лиц проекта [20].

2.4.1 Планирование проекта

В ходе планирования проекта было выделено 12 задач. Список задач проекта, их виды, длительность и предшественники приведены в таблице 1.

Таблица 1 – Задачи проекта

№	Название	Вид Задачи	Предшественники	Длительность
1	Начало проекта	Веха	-	-
2	Анализ предметной области	Фаза	-	-
3	Просмотр рынка информационных систем	Задача	-	4
4	Сбор информации о существующих информационных системах	Задача	3	6
5	Анализ требований	Фаза	-	-
6	Анализ функциональных требований	Задача	4	12
7	Анализ используемых технологий	Задача	6;7	8
8	Проектирование	Фаза	-	-
9	Проектирование базовой архитектуры	Задача	6;7	4
10	Проектирование логической модели базы данных	Задача	6	2,5
11	Проектирование интерфейса пользователя	Задача	6;7;9	14
12	Реализация	Фаза	-	-
13	Реализация физической модели базы данных	Задача	10	4
14	Реализация серверной части приложения	Задача	13	8
15	Реализация клиентской части приложения	Задача	9;13;11	22
16	Тестирование	Фаза	-	-
17	Функциональное тестирование	Задача	16;17	10
18	Завершение проекта	Фаза	-	-
19	Оформление документации	Задача	17	10
20	Конец проекта	Веха	-	-

В MS Project 2013 создадим проект с начальной датой 01.12.19. Занесем выше перечисленные задачи в созданный проект.

Выделим необходимые ресурсы для выполнения каждой из задач данного проекта. Ресурсы и затраты представлены в таблице 2.

Таблица 2 – Ресурсы проекта

Название ресурса	Тип	Затраты			Затраты на исп.
		Таблица норм	Станд. Ставка	Ставка сверхур.	
Программист 1	Т	А	22000 р./мес	300 р./ч	-
Программист 2	Т	А	18000 р./мес	280 р./ч	-
Аналитик	Т	А	15000 р./мес	250 р./ч	-
Тестировщик	Т	А	15000 р./мес	250 р./ч	-
Компьютер	М	А	15000 р.		-
Принтер	М	А	3000 р.		-
Бумага	М	А	200 р.		-
Интернет	З				-

Распределение ресурсов по задачам представлено в таблице 3.

Таблица 3 – Распределение ресурсов по задачам

Название	Ресурс	Единицы (затраты)	Таблица норм затрат
Начало проекта	Компьютер	4	А
Просмотр рынка информационных систем	Аналитик, Интернет	100	А 250
Сбор информации о существующих информационных системах	Аналитик, Интернет	100	А 300
Анализ функциональных требований	Аналитик, Программист 1	100 100	А А
Анализ используемых технологий	Программист 1, Аналитик	100 30	А А
Проектирование базовой архитектуры	Программист 1	100	А
Проектирование логической модели базы данных	Программист 2	100	А
Проектирование интерфейса пользователя	Программист 1	100	А
Реализация физической модели базы данных	Программист 2	100	А
Реализация серверной части приложения	Программист 2	100	А
Реализация клиентской части приложения	Программист 1, Программист 2	100 100	А А
Функциональное тестирование	Тестировщик, Программист 2	50 100	А А
Оформление документации	Программист 1, Тестировщик, Принтер, Бумага	100 100 1 1	А А А А

Занесем выше перечисленные ресурсы в проект. Результаты представлены на рисунке 14.

	Название ресурса	Тип	Единицы измерения материалов	Краткое название	Макс. единиц	Стандартная ставка	Ставка сверхурочных	Начисление	Базовый календар	Добавить новый столб
1	Программист1	Трудовой		Прог1	100%	22 000,00р./мес	300,00р./ч	Пропорциональное	Стандартный	
2	Программист2	Трудовой		Прог2	100%	18 000,00р./мес	280,00р./ч	Пропорциональное	Стандартный	
3	Аналитик	Трудовой		Аналит	100%	15 000,00р./мес	250,00р./ч	Пропорциональное	Стандартный	
4	Тестировщик	Трудовой		Тестир	100%	15 000,00р./мес	250,00р./ч	Пропорциональное	Стандартный	
5	Компьютер	Материальный	Штука	Комп		15 000,00р.		Пропорциональное		
6	Принтер	Материальный	Штука	Принт		3 000,00р.		Пропорциональное		
7	Бумага	Материальный	Пачка	Бум		200,00р.		Пропорциональное		
8	Интернет	Затраты		Инет				Пропорциональное		

Рисунок 14 – Ресурсы проекта

Далее закрепим за задачами соответствующие ресурсы. После, MS Project 2013 предложит наглядную иллюстрацию календарного плана в виде диаграммы Ганта. Полученная диаграмма Ганта приведена на рисунке 15.

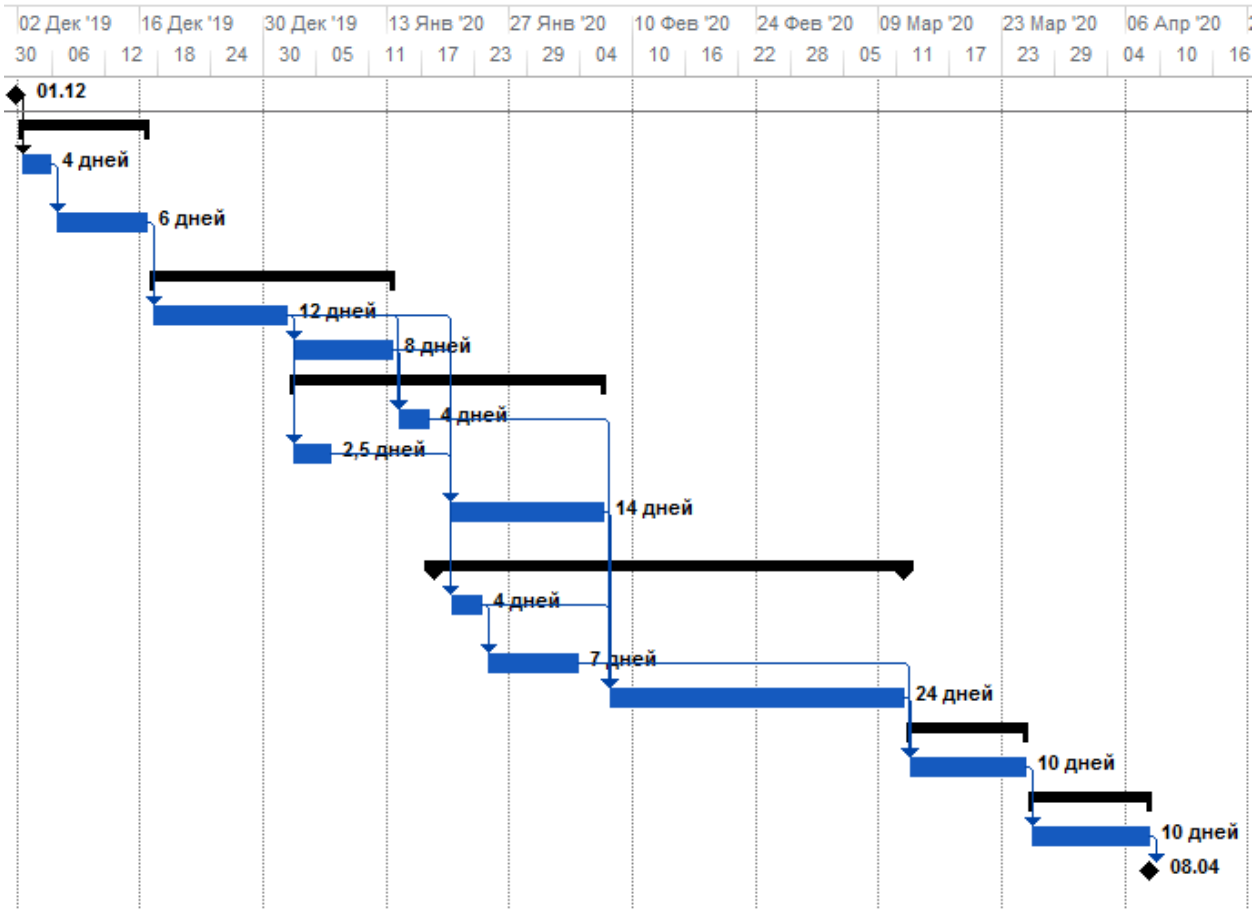


Рисунок 15 – Диаграмма Ганта

Сетевой график данного проекта представлен на рисунке 16. Красным цветом на рисунке 16 показан критический путь.

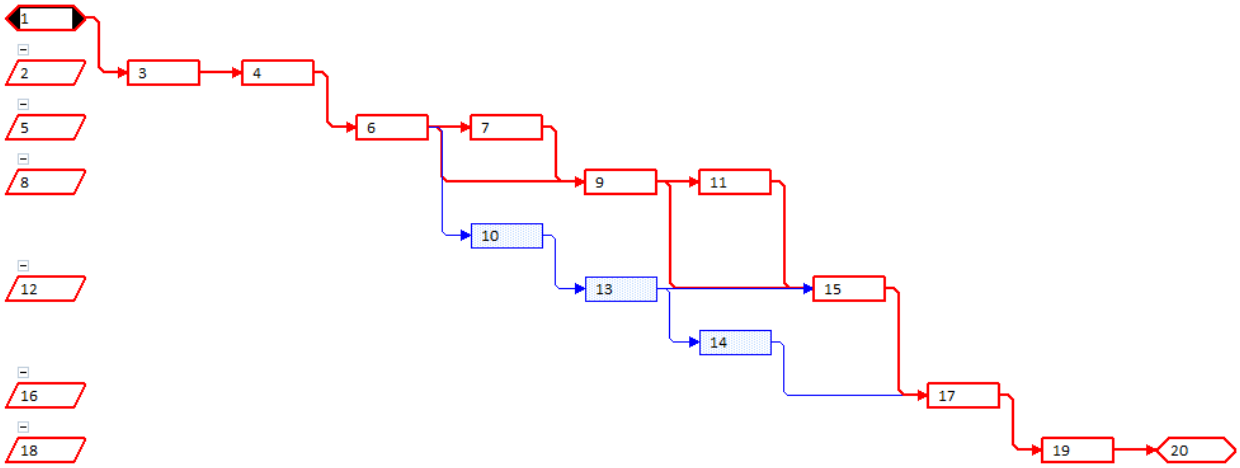


Рисунок 16 – Сетевой график

Графики загрузки трудовых ресурсов программиста 1, программиста 2, аналитика и тестировщика представлены на рисунках 17-20.

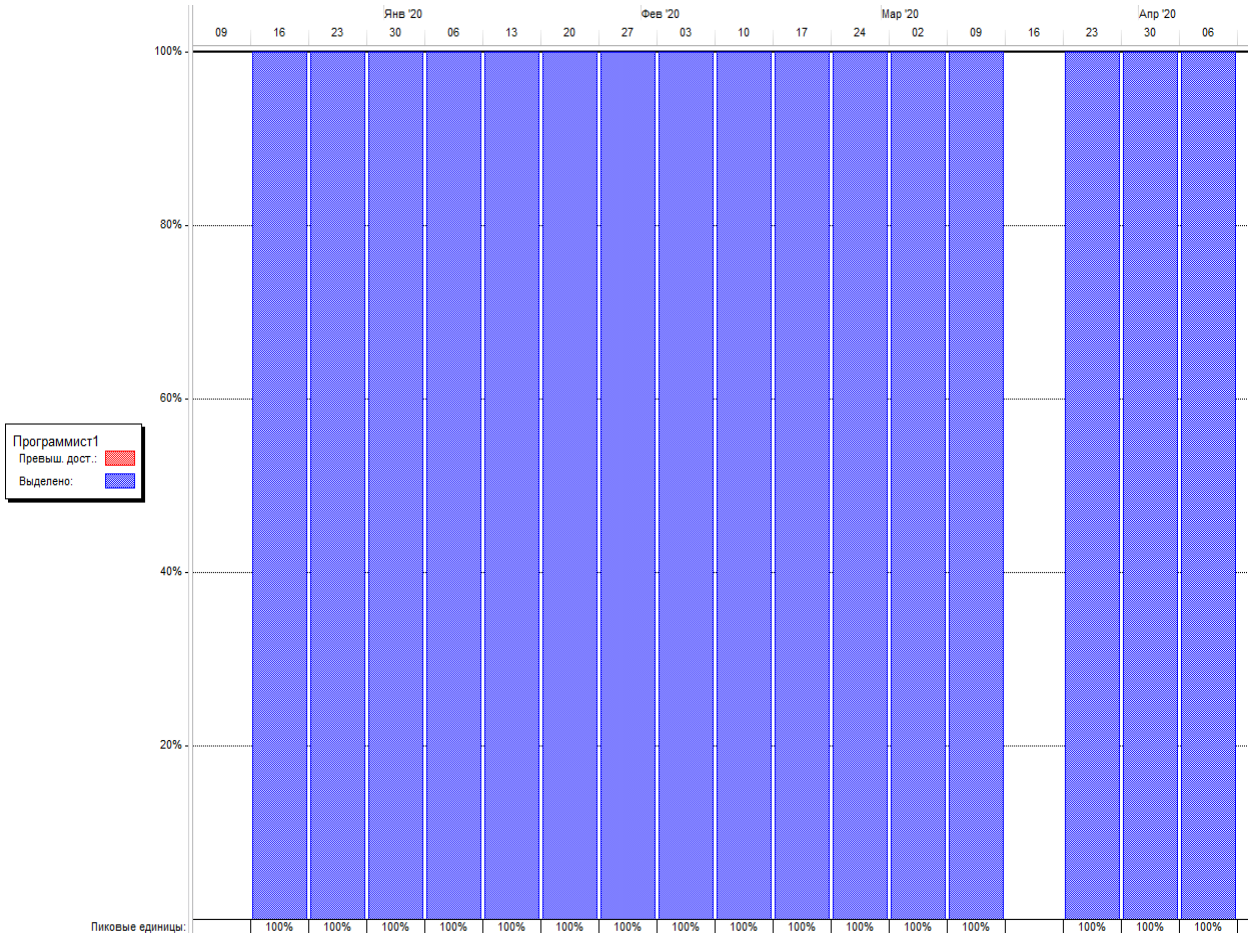


Рисунок 17 – График загрузки программиста 1

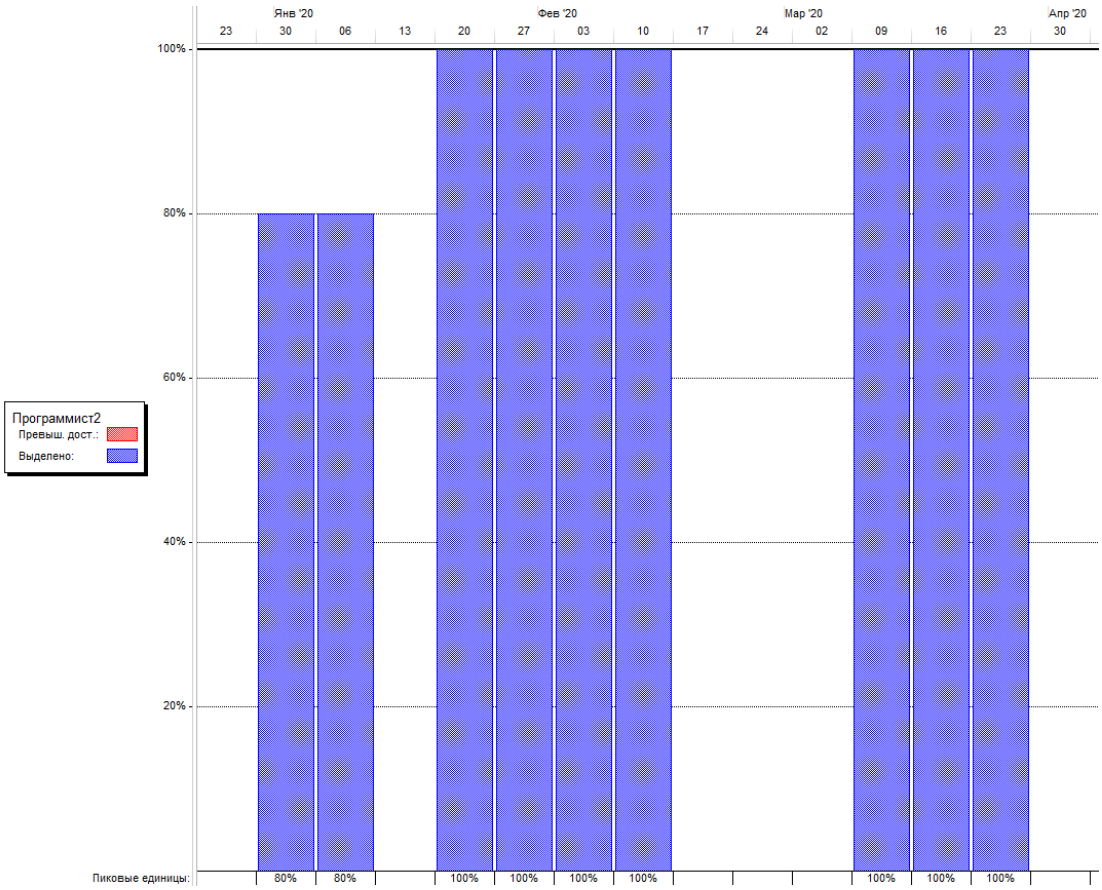


Рисунок 18 – График загрузки программиста 2

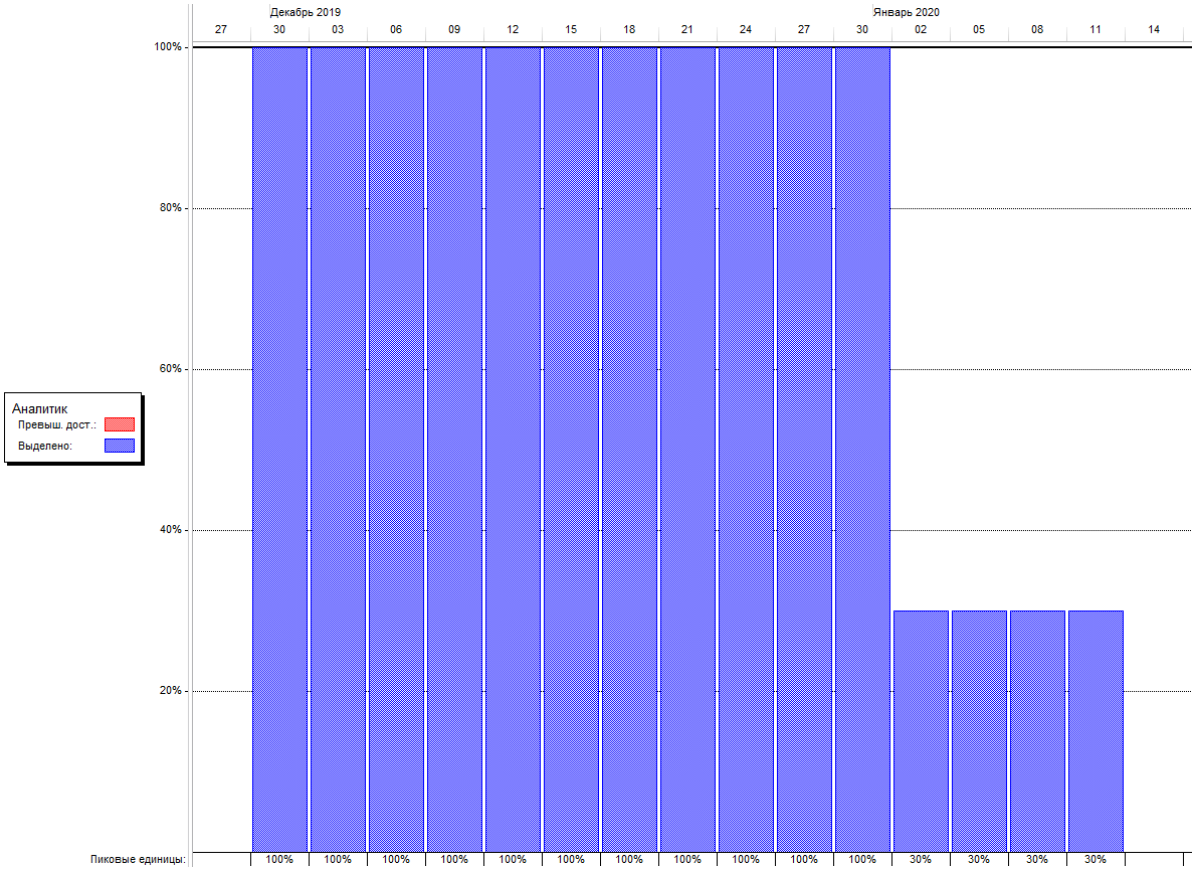


Рисунок 19 – График загрузки аналитика

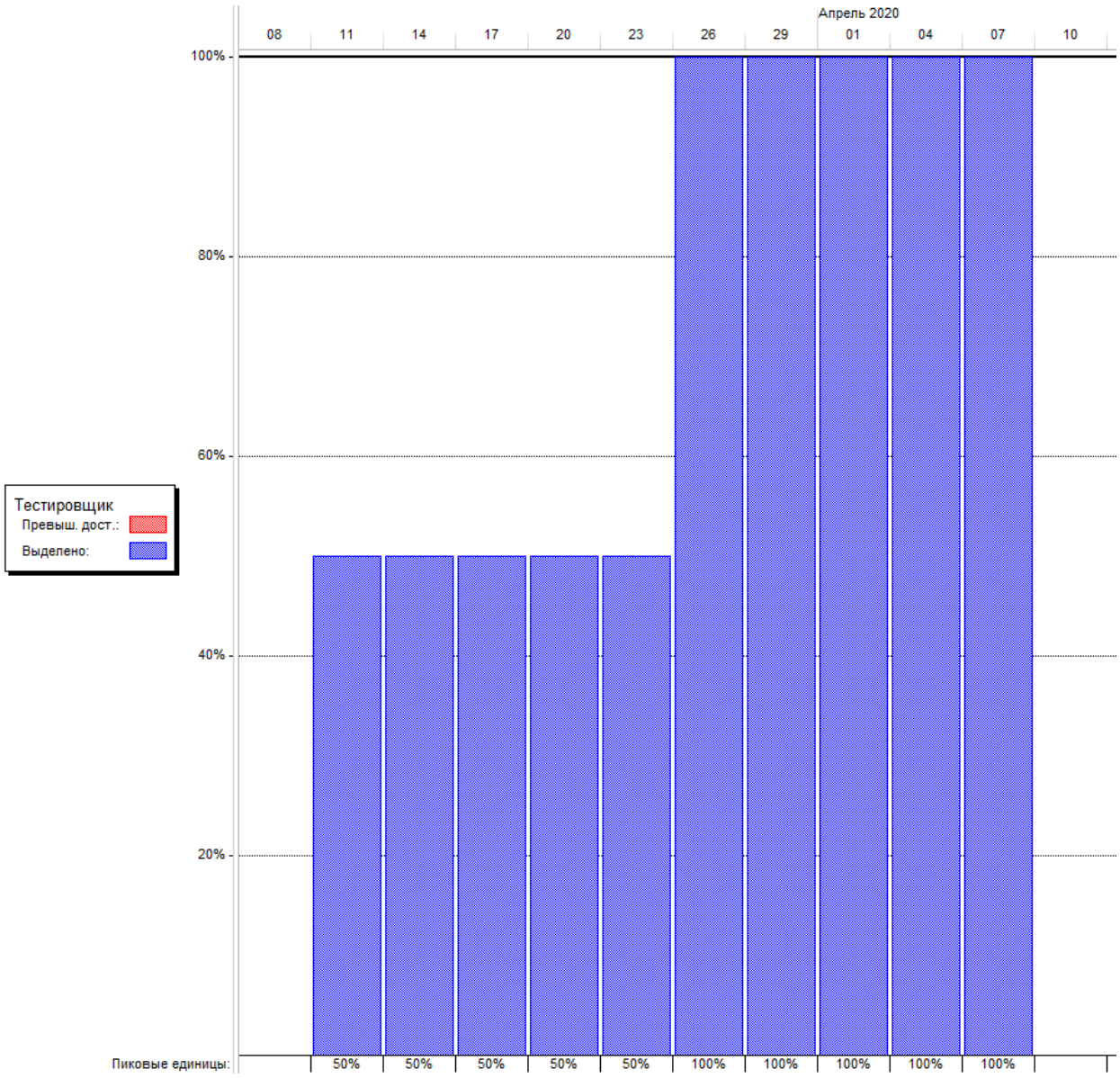


Рисунок 20 – График загрузки тестировщика

Из предоставленных графиков видно, что первый программист загружен полностью, второй программист большую часть времени загружен на 100%, аналитик также большую часть времени имеет загрузку 100%, тестировщик в среднем загружен на 75%.

Нужно отметить, что основную часть работы по анализу, разработке и кодированию программного обеспечения выполняет первый программист. Второй программист выступает в роли помощника главного (первого) программиста.

Первый программист занимается клиентской частью приложения. Второй, в свою очередь, проектирует и кодирует серверную часть приложения.

Сводка по проекту представлена на рисунке 21. На сводке указаны длительность в днях, общие затраты, а также график затрат по фазам проекта.

Рисунок 21 – Сводка по проекту

На рисунке 21 видно, что проект разделен на 5 фаз:

- анализ предметной области;
- анализ требований;
- проектирование;
- реализация;
- тестирование.

Также на рисунке 21 представлены две вехи: «Начало проекта» и «Завершение проекта».

Длительность выполнения проекта составила 93 дня, а его бюджет составил 138 443 рубля.

3 Реализация

3.1 Физическая модель базы данных

Физическая модель строится на основе логической модели данных и учитывает особенности целевой СУБД. В качестве СУБД был выбран продукт Oracle.

При помощи case-средства ERWin Data Modeler была разработана физическая модель, представленная на рисунке 22.

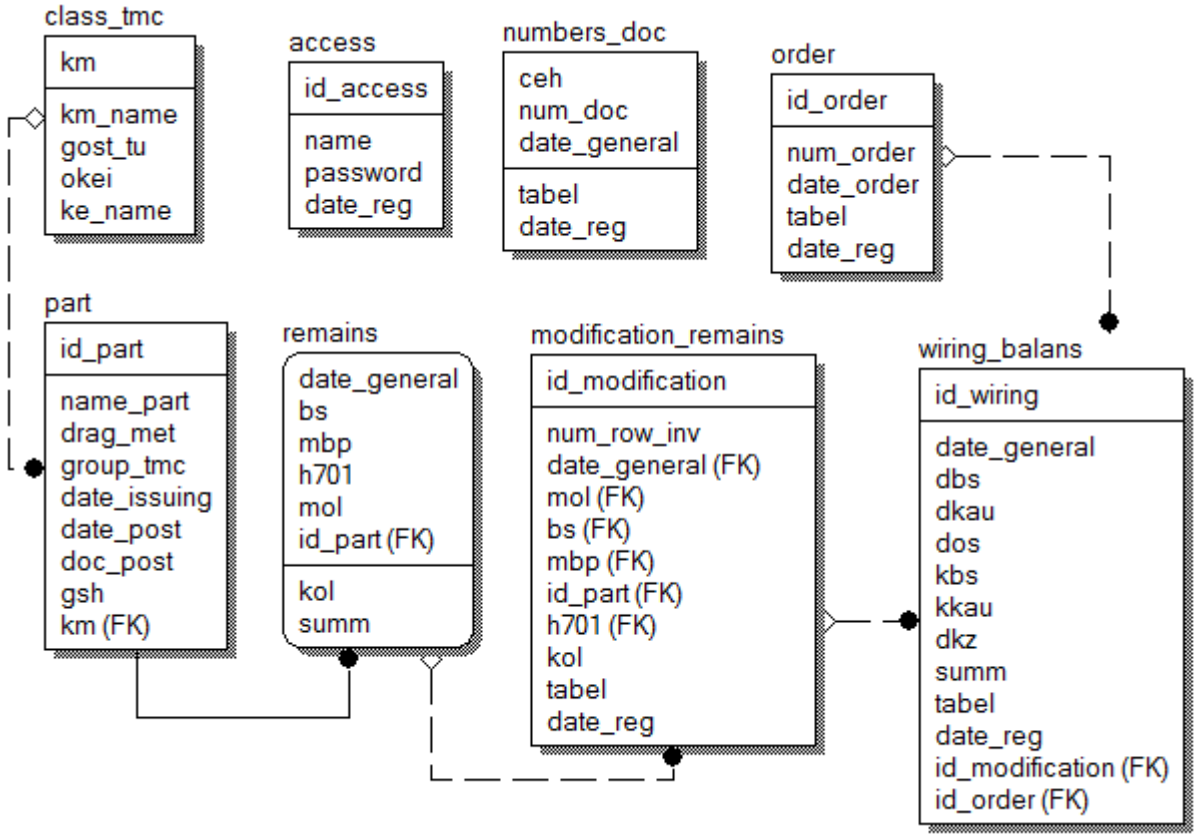


Рисунок 22 – Физическая модель базы данных

Сгенерированный скрипт DDL в ERWin Data Modeler (приложение Г) был выполнен в СУБД Oracle.

3.2 Серверная часть приложения

В ходе разработки распределенного клиент-серверного приложения для учета материально-производственных запасов предприятия и формирования инвентаризационных описей было принято решение описать логику прило-

жения для работы с базой данных в web-приложении серверной части. Серверная часть приложения представляет собой вторую часть приложения на языке программирования PHP и базу данных, которая включает в себя таблицы и представление.

3.2.1 Реализация представлений

Исходный код представления для отображения списка строк инвентаризационной описи на языке PL/SQL представлен в листинге 1.

Листинг 1 – Представление для отображения списка строк ИНВ-описи

```
SELECT date_general,bs,mbp,h701,mol,km,name_part,km_name,okei,ke_name,
     SUM(kol) kol,SUM(summ) summ
FROM(
     SELECT to_char(date_general,'dd.mm.yyyy') date_general,bs,mbp,
          h701,mol,b.km,name_part,km_name,okei,ke_name,kol,summ
     FROM main.remains a JOIN main.part b ON a.id_part=b.id_part
     JOIN main.class_tmc c ON b.km=c.km
     UNION ALL
     SELECT to_char(date_general,'dd.mm.yyyy') date_general,bs,mbp,h701,
     mol,b.km,name_part,km_name,okei,ke_name,kol,summ
     FROM main.modification_remains a JOIN main.part b ON a.id_part=b.id_part
     JOIN main.class_tmc c ON b.km=c.km)
GROUP BY date_general,bs,mbp,h701,mol,km,name_part,km_name,ke_name
HAVING SUM(kol)!=0 or SUM(summ)!=0
```

Исходный код представления для отображения списка строк сличительной ведомости бухгалтера на языке PL/SQL представлен в листинге 2.

Листинг 2 – Представление для отображения списка строк сличительной ведомости

```
SELECT num_row_inv,bs,substr('00'||mbp,-2) mbp,to_char(MAX(km)) km,MAX(ke)
ke,MAX(h701) h701,SUM(nvl(k1,0)) k1,SUM(nvl(sum1,0)) sum1,SUM(k2)
k2,SUM(sum2) sum2,
      CASE WHEN (SUM(nvl(k1,0))-SUM(k2))<0 and (SUM(nvl(sum1,0))-SUM(sum2))<0
THEN 0 ELSE SUM(nvl(k1,0))-SUM(k2) END k3,
      CASE WHEN (SUM(nvl(k1,0))-SUM(k2))<0 and (SUM(nvl(sum1,0))-SUM(sum2))<0
THEN 0 ELSE SUM(nvl(sum1,0))-SUM(sum2) END sum3,
      CASE WHEN (SUM(nvl(k1,0))-SUM(k2))<0 and (SUM(nvl(sum1,0))-SUM(sum2))<0
THEN SUM(nvl(k1,0))-SUM(k2) ELSE 0 END k4,
```

```
      CASE WHEN (SUM(nvl(k1,0))-SUM(k2))<0 and (SUM(nvl(sum1,0))-SUM(sum2))<0
THEN SUM(nvl(sum1,0))-SUM(sum2) ELSE 0 END sum4,0 sum5,0 sum6,0 pz
FROM(
      SELECT num_row_inv,b.bs,b.mbp,km,ke,b.h701,SUM(nvl(c.kol,0))
k1,SUM(nvl(c.kol,0)) sum1,SUM(b.kol) k2,SUM(b.h701*b.kol) sum2
      FROM main.part a, main.modification_remains b
      LEFT OUTER JOIN main.remains c
      ON to_char(b.date_general,'yyyymm')=to_char(c.date_general,'yyyymm')
and b.mol=c.mol and b.bs=c.bs and b.mbp=c.mbp and b.h701=c.h701 and
b.id_part=c.id_part
      WHERE a.id_part=b.id_part
      GROUP BY num_row_inv,b.bs,b.mbp,km,ke,b.h701

      union all

      SELECT num_row_inv,bs,mbp,km,ke,h701,0 k1,0 sum1,kol k2,summ sum2
      FROM main.modification_remains a, main.part b WHERE a.id_part=b.id_part

      union all

      SELECT num_row_inv,CASE WHEN substr('0000'||dbs,-4,2) in (07,10) THEN
dbs ELSE kbs END bs,
      CASE WHEN substr('0000'||dbs,-4,2) in (07,10) THEN
to_number(substr('00000'||dkau,-5,2)) ELSE to_number(substr('00000'||kkau,-
5,2)) END mbp,null km,null ke,null h701,0 k1,CASE WHEN substr('0000'||kbs,-
4,2) in (07,10) THEN (-1)*summ ELSE summ END sum1,0 k2,0 sum2
      FROM main.wiring_balans)
      GROUP BY num_row_inv,bs,mbp
      HAVING (SUM(nvl(sum1,0))-SUM(nvl(sum2,0)))!=0
```

Результат обращения к представлениям соответствует ожидаемым результатам, вследствие чего можно сделать вывод, что представления работают корректно.

3.3 Модель реализации клиентского приложения

Модель реализации описывает, как реализуются в виде компонентов элементы модели проектирования.

Для отображения реализации используется диаграмма компонентов. Диаграмма компонентов представлена на рисунке 23.

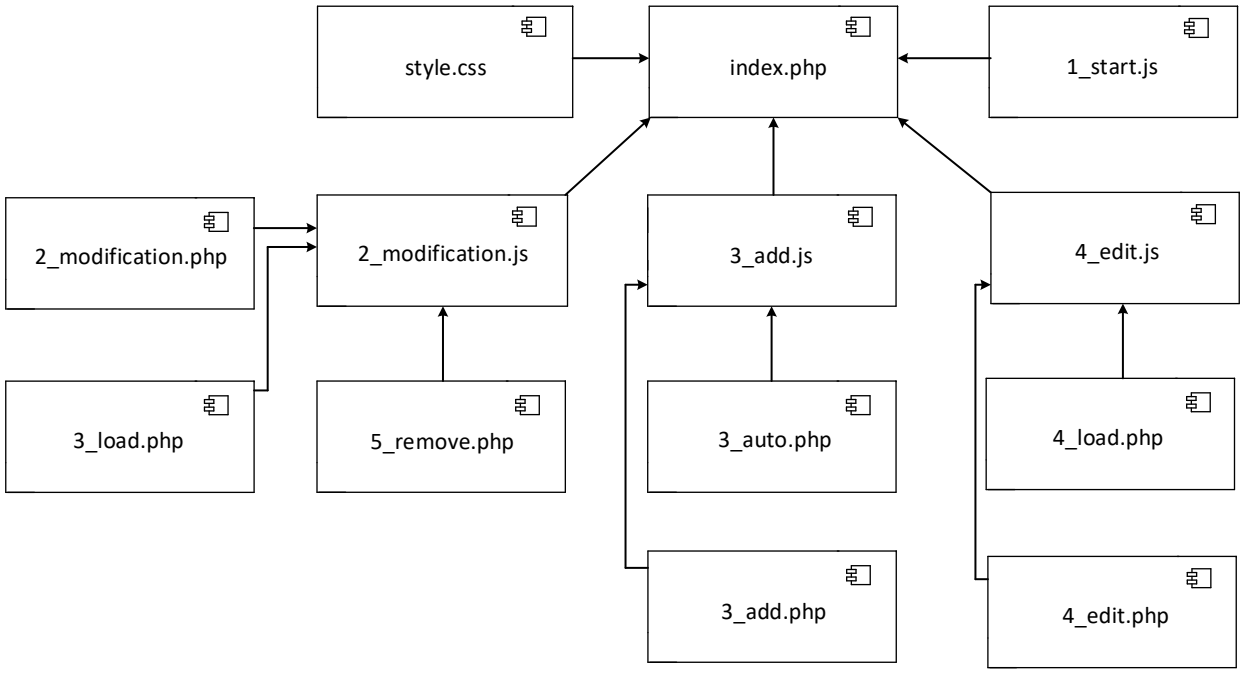


Рисунок 23 – Диаграмма компонентов

Следует отметить, что на диаграмме компонентов главным является файл «index.php», так как именно с него начинается запуск всей программы. Точка входа в программу располагается в файле «1_start.js», а таблица стилей в файле «style.css».

Исходный код клиентского приложения на языках программирования JavaScript и PHP приведен в приложении Б.

Одним из способов моделирования статического вида системы с точки зрения развертывания является диаграмма развертывания.

Диаграмма развертывания предназначена для визуализации компонентов программы, существующих лишь на этапе ее выполнения. При этом представляются только компоненты программы, являющиеся исполнимыми файлами или динамическими библиотеками. Компоненты, которые не используются на этапе выполнения, на диаграмме развертывания не показываются.

Диаграмма развертывания содержит графические изображения процессоров, устройств, процессов и связей между ними. В отличие от диаграмм логического представления, диаграмма развертывания является единой для системы в целом, поскольку должна всецело отражать особенности ее реали-

зации. Эта диаграмма, по сути, завершает процесс для конкретной программной системы и ее разработка, как правило, является последним этапом спецификации модели.

Диаграмма развертывания представлена на рисунке 24.

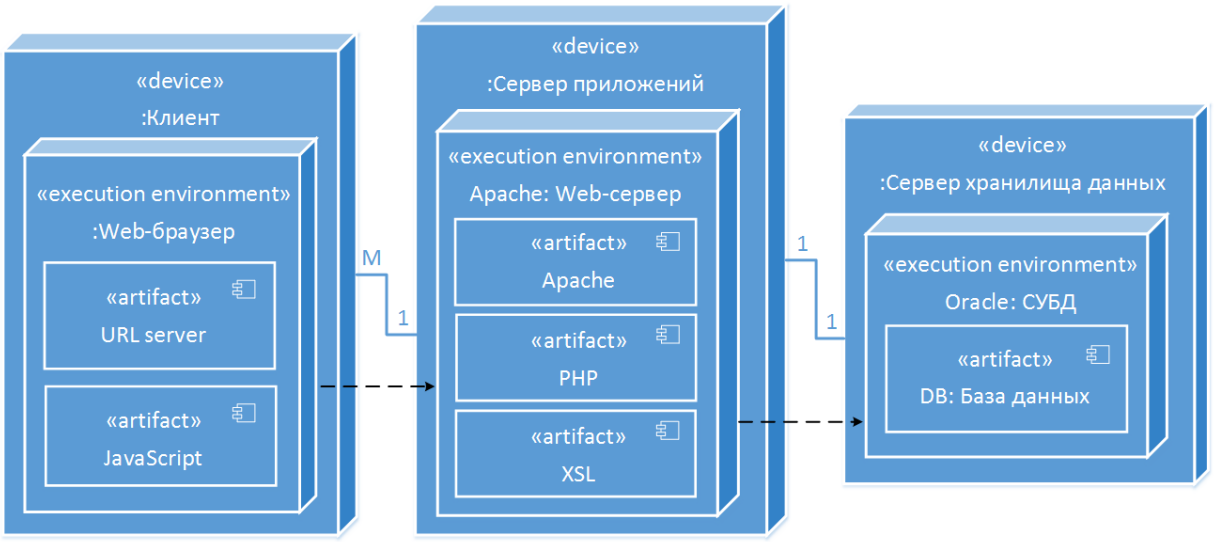


Рисунок 24 – Диаграмма развертывания

На рисунке 24 показано, что на станции клиента должен присутствовать web-браузер с установленным модулем Adobe Flash для запуска JavaScript. После ввода URL-адреса сервера, web-браузер клиента обратится к web-серверу, который располагается на сервере приложений, с целью запуска клиентского web-приложения. После определенных действий пользователя клиентское web-приложение (JavaScript) обратится серверному web-приложению (PHP) для обмена информацией. Далее серверное web-приложение обратится к СУБД, которая располагается на сервере хранилища данных, с целью выполнения некоторых запросов к данным.

4 Контроль качества программного обеспечения

Тестирование – способ обеспечения качества программного продукта.

Качество программного продукта характеризуется набором свойств, определяющих, насколько продукт «хорош» с точки зрения заинтересованных сторон. Тестирование является одним из наиболее устоявшихся способов обеспечения качества разработки программного обеспечения и входит в набор эффективных средств современной системы обеспечения качества программного продукта.

С технической точки зрения тестирование заключается в выполнении приложения на некотором множестве исходных данных и сверке получаемых результатов с заранее известными (эталонными) с целью установить соответствие различных свойств и характеристик приложения заказанным свойствам. Как одна из основных фаз процесса разработки программного продукта, тестирование характеризуется достаточно большим вкладом в суммарную трудоемкость разработки продукта [21].

В данной выпускной квалификационной работе была выбрана методология функционального тестирования.

4.1 Функциональное тестирование

Функциональное тестирование относится к категории тестирования «черного ящика». Каждая функция программы тестируется ввода её исходных данных и анализа выходных. При этом внутренняя структура программы учитывается очень редко [22].

Функциональное тестирование предполагает составление плана тестирования, который строится на основе вариантов использования и включает в себя описание тестов и их результаты.

План тестирования приведен в таблице 4.

Таблица 4 – План тестирования

Вариант использования	Тест	Результат
Войти в систему	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль.	Тест выполнен. Вход в систему разрешен.
Войти в систему	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и неверный пароль.	Тест выполнен. Вход в систему запрещен (рисунок В.1).
Добавить новую позицию ТМЦ	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, нажимаем кнопку «Корректировка». В открывшемся окне нажимаем кнопку «Добавить». В появившемся окне вводим информацию о новой ТМЦ. Нажимаем кнопку «Ввод».	Тест выполнен. Новая ТМЦ была успешно добавлена в инвентаризационную опись (рисунок В.2).
Изменить данные о ТМЦ	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, нажимаем кнопку «Корректировка». В открывшемся окне выбираем строку для корректировки и нажимаем кнопку «Изменить». В появившемся окне корректируем нужную нам информацию о ТМЦ. Нажимаем кнопку «Ввод».	Тест выполнен. Информация о ТМЦ была успешно изменена (рисунок В.3).
Проставить фактическое наличие ТМЦ	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, нажимаем кнопку «Корректировка». В открывшемся окне выбираем строку для корректировки и нажимаем кнопку «Изменить». В появившемся окне указываем реальное количество ТМЦ на складе. Нажимаем кнопку «Ввод».	Тест выполнен. Новое фактическое количество сохранено в системе (рисунок В.4).

Продолжение таблицы 4

Вариант использования	Тест	Результат
Удалить позицию ТМЦ	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, нажимаем кнопку «Корректировка». В открывшемся окне выбираем строку инвентаризационной описи, которую следует удалить, затем нажимаем кнопку «Удалить». Система запрашивает подтверждение удаления. Нажимаем кнопку «Ок».	Тест выполнен. Данные о ранее заведенной строке ТМЦ были успешно удалены.
Найти ТМЦ	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, нажимаем кнопку «Корректировка». В открывшемся окне указываем в поля-фильтры данные о ТМЦ. Нажимаем клавишу «Enter».	Тест выполнен. Система вывела запрашиваемую информацию на экран.
Распечатать инвентаризационную опись N ИНВ-3	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, а также выбираем режим печати «Титульный лист», затем нажимаем кнопку «Печать». Далее повторяем действия для режимов печати «Строки инв.описи» и «Заключительный лист».	Тест выполнен. Выбранная унифицированная форма инвентаризационной описи была сформирована в формат PDF для печати (рисунок В.5, рисунок В.6, рисунок В.7).
Распечатать инвентаризационную опись N ИНВ-5	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем вид документа «Материалы ИНВ-3», заводим отчетную дату и указываем склад, а также выбираем режим печати «Титульный лист», затем нажимаем кнопку «Печать». Далее повторяем действия для режимов печати «Строки инв.описи» и «Заключительный лист».	Тест выполнен. Выбранная унифицированная форма инвентаризационной описи была сформирована в формат PDF для печати.

Продолжение таблицы 4

Вариант использования	Тест	Результат
Сформировать сличительную ведомость	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем в меню подпункт «Сличительная ведомость», затем нажимаем кнопку «Сформировать сличительную ведомость».	Тест выполнен. Новая сличительная ведомость успешно сформирована (рисунок В.8).
Распечатать сличительную ведомость N ИНВ-19	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем в меню подпункт «Инвентаризационная опись», затем нажимаем кнопку «Выгрузка инвентаризационной описи». В открывшемся окне следует указать отчетную дату и склад, выбрать вид документа «Сличительная ведомость (ИНВ-19)», затем нажать кнопку «Выгрузить».	Тест выполнен. Система сформировала электронный документ в формате XLSX для дальнейшей печати (рисунок В.9).
Добавить новую проводку в баланс	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем в меню подпункт «Бухгалтерские записи», затем нажимаем кнопку «Корректирующие проводки в баланс». В открывшемся окне нажимаем кнопку «Добавить». Указываем реквизиты бухгалтерской проводки, а также прикрепляем строку-основание инвентаризационной описи. Нажимаем кнопку «Добавить».	Тест выполнен. Бухгалтерская проводка была успешно заведена в систему (рисунок В.10).
Удалить проводку из баланса	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем в меню подпункт «Бухгалтерские записи», затем нажимаем кнопку «Корректирующие проводки в баланс». В открывшемся окне выбираем из списка бухгалтерскую проводку, которую следует удалить, затем нажимаем кнопку «Удалить». Система запрашивает подтверждение удаления. Нажимаем кнопку «Ок».	Тест выполнен. Выбранная бухгалтерская проводка успешно удалена из системы (рисунок В.11).

Окончание таблицы 4

Вариант использования	Тест	Результат
Завести новый приказ об инвентаризации	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем в меню подпункт «Справочники». Нажимаем на кнопку «Приказ об инвентаризации». В открывшемся окне нажимаем кнопку «Завести новый приказ». Заводим необходимые данные о приказе. Нажимаем кнопку «Сохранить».	Тест выполнен. Новый приказ успешно заведен в систему (рисунок В.12).
Удалить приказ об инвентаризации	Заходим в web-приложение путем ввода URL-адреса в браузере. В появившемся окне вводим верное имя пользователя и пароль. Далее выбираем в меню подпункт «Справочники». Нажимаем на кнопку «Приказ об инвентаризации». В открывшемся окне выбираем нужный приказ и нажимаем кнопку «Удалить приказ». Система запрашивает подтверждение удаления. Нажимаем кнопку «Ок».	Тест выполнен. Выбранный приказ удален из системы (рисунок В.13).

На основе анализа результатов тестирования можно сделать вывод, что разработанное распределенное клиент-серверное приложение для учета материально-производственных запасов предприятия и формирования инвентаризационных описей работает корректно.

5 Безопасность жизнедеятельности

С развитием научно-технического прогресса немаловажную роль играет возможность безопасного исполнения людьми своих трудовых обязанностей. В связи с этим была создана и развивается наука о безопасности труда и жизнедеятельности человека.

Безопасность жизнедеятельности (БЖД) – это комплекс мероприятий, направленных на обеспечение безопасности человека в среде обитания, сохранение его здоровья, разработку методов и средств защиты путем снижения влияния вредных и опасных факторов до допустимых значений, выработку мер по ограничению ущерба в ликвидации последствий чрезвычайных ситуаций мирного и военного времени [23].

Цель и содержание БЖД:

– обнаружение и изучение факторов окружающей среды, отрицательно влияющих на здоровье человека;

– ослабление действия этих факторов до безопасных пределов или исключение их если это возможно;

– ликвидация последствий катастроф и стихийных бедствий.

Охрана здоровья трудящихся, обеспечение безопасности условий труда, ликвидация профессиональных заболеваний и производственного травматизма составляет одну из главных забот человеческого общества. Обращается внимание на необходимость широкого применения прогрессивных форм научной организации труда, сведения к минимуму ручного, малоквалифицированного труда, создания обстановки, исключающей профессиональные заболевания и производственный травматизм [24].

На рабочем месте должны быть предусмотрены меры защиты от возможного воздействия опасных и вредных факторов производства. Уровни этих факторов не должны превышать предельных значений, оговоренных правовыми, техническими и санитарно-техническими нормами. Эти нормативные документы обязывают к созданию на рабочем месте условий труда,

при которых влияние опасных и вредных факторов на работающих либо устранено совсем, либо находится в допустимых пределах.

5.1 Характеристика условий труда программиста

Научно-технический прогресс внес серьезные изменения в условия производственной деятельности работников умственного труда. Их труд стал более интенсивным, напряженным, требующим значительных затрат умственной, эмоциональной и физической энергии. Это потребовало комплексного решения проблем эргономики, гигиены и организации труда, регламентации режимов труда и отдыха.

В настоящее время компьютерная техника широко применяется во всех областях деятельности человека. При работе с компьютером человек подвергается воздействию ряда опасных и вредных производственных факторов: электромагнитных полей (диапазон радиочастот: ВЧ, УВЧ и СВЧ), инфракрасного и ионизирующего излучений, шума и вибрации, статического электричества и др. [25].

Работа с компьютером характеризуется значительным умственным напряжением и нервно-эмоциональной нагрузкой операторов, высокой напряженностью зрительной работы и достаточно большой нагрузкой на мышцы рук при работе с клавиатурой ЭВМ. Большое значение имеет рациональная конструкция и расположение элементов рабочего места, что важно для поддержания оптимальной рабочей позы человека-оператора.

5.2 Требования к производственным помещениям

5.2.1 Освещение

Правильно спроектированное и выполненное производственное освещение улучшает условия зрительной работы, снижает утомляемость, способствует повышению производительности труда, благотворно влияет на производственную среду, оказывая положительное психологическое воздействие на работающего, повышает безопасность труда и снижает травматизм.

Недостаточность освещения приводит к напряжению зрения, ослабляет внимание, приводит к наступлению преждевременной утомленности. Чрезмерно яркое освещение вызывает ослепление, раздражение и резь в глазах. Неправильное направление света на рабочем месте может создавать резкие тени, блики, дезориентировать работающего. Все эти причины могут привести к несчастному случаю или профзаболеваниям, поэтому столь важен правильный расчет освещенности.

Существует три вида освещения – естественное, искусственное и совмещенное (естественное и искусственное вместе) [26].

Естественное освещение – освещение помещений дневным светом, проникающим через световые проемы в наружных ограждающих конструкциях помещений. Естественное освещение характеризуется тем, что меняется в широких пределах в зависимости от времени дня, времени года, характера области и ряда других факторов.

Искусственное освещение применяется при работе в темное время суток и днем, когда не удается обеспечить нормированные значения коэффициента естественного освещения (пасмурная погода, короткий световой день). Освещение, при котором недостаточное по нормам естественное освещение дополняется искусственным, называется совмещенным освещением.

При выполнении работ категории высокой зрительной точности (наименьший размер объекта различения 0,3…0,5 мм) величина коэффициента естественного освещения (КЕО) должна быть не ниже 1,5%, а при зрительной работе средней точности (наименьший размер объекта различения 0,5…1,0 мм) КЕО должен быть не ниже 1,0%. В качестве источников искусственного освещения обычно используются люминесцентные лампы типа ЛБ или ДРЛ, которые попарно объединяются в светильники, которые должны располагаться над рабочими поверхностями равномерно [26].

Требования к освещенности в помещениях, где установлены компьютеры, следующие: при выполнении зрительных работ высокой точности об-

щая освещенность должна составлять 300 лк, а комбинированная – 750 лк; аналогичные требования при выполнении работ средней точности – 200 и 300 лк соответственно.

5.2.2 Параметры микроклимата

Вычислительная техника является источником существенных тепловыделений, что может привести к повышению температуры и снижению относительной влажности в помещении. В помещениях, где установлены компьютеры, должны соблюдаться определенные параметры микроклимата. В санитарных нормах СН-245-71 установлены величины параметров микроклимата, создающие комфортные условия. Эти нормы устанавливаются в зависимости от времени года, характера трудового процесса и характера производственного помещения (таблица 5) [25].

Объем помещений, в которых размещены работники вычислительных центров, не должен быть меньше 19,5 м3/человека с учетом максимального числа одновременно работающих в смену. Нормы подачи свежего воздуха в помещения, где расположены компьютеры, приведены в таблице 6.

Таблица 5 – Параметры микроклимата для помещений, где установлены компьютеры

Период года	Параметр микроклимата	Величина
Холодный	Температура воздуха в помещении Относительная влажность Скорость движения воздуха	22…24°С 40…60% до 0,1м/с
Теплый	Температура воздуха в помещении Относительная влажность Скорость движения воздуха	23…25°С 40…60% 0,1…0,2м/с

Таблица 6 – Нормы подачи свежего воздуха в помещения с компьютерами

Характеристика помещения	Объемный расход подаваемого в помещение свежего воздуха, $м^3$ /на одного человека в час
Объем до 20$м^3$ на человека 20…40$м^3$ на человека Более 40$м^3$ на человека	Не менее 30 Не менее 20 Естественная вентиляция

5.2.3 Шум и вибрация

Шум ухудшает условия труда оказывая вредное действие на организм человека. Работающие в условиях длительного шумового воздействия испытывают раздражительность, головные боли, головокружение, снижение памяти, повышенную утомляемость, понижение аппетита, боли в ушах и т. д. Такие нарушения в работе ряда органов и систем организма человека могут вызвать негативные изменения в эмоциональном состоянии человека вплоть до стрессовых. Под воздействием шума снижается концентрация внимания, нарушаются физиологические функции, появляется усталость в связи с повышенными энергетическими затратами и нервно-психическим напряжением, ухудшается речевая коммутация. Все это снижает работоспособность человека и его производительность, качество и безопасность труда. Длительное воздействие интенсивного шума (выше 80 дБ(А)) на слух человека приводит к его частичной или полной потере [27].

Уровень шума на рабочем месте математиков-программистов и операторов видеоматериалов не должен превышать 50 дБА, а в залах обработки информации на вычислительных машинах – 65 дБА. Для снижения уровня шума стены и потолок помещений, где установлены компьютеры, могут быть облицованы звукопоглощающими материалами. Уровень вибрации в помещениях вычислительных центров может быть снижен путем установки оборудования на специальные виброизоляторы.

5.2.4 Электромагнитное и ионизирующее излучения

Большинство ученых считают, что как кратковременное, так и длительное воздействие всех видов излучения от экрана монитора не опасно для здоровья персонала, обслуживающего компьютеры. Однако исчерпывающих данных относительно опасности воздействия излучения от мониторов на работающих с компьютерами не существует и исследования в этом направлении продолжаются [25].

Максимальный уровень рентгеновского излучения на рабочем месте оператора компьютера обычно не превышает 10 мкбэр/ч, а интенсивность ультрафиолетового и инфракрасного излучений от экрана монитора лежит в пределах 10…100 мВт/м2.

Для снижения воздействия этих видов излучения рекомендуется применять мониторы с пониженным уровнем излучения (MPR-II, TCO-92, TCO-99), устанавливать защитные экраны, а также соблюдать регламентированные режимы труда и отдыха.

5.3 Эргономические требования к рабочему месту

Проектирование рабочих мест, снабженных видеотерминалами, относится к числу важнных проблем эргономического проектирования в области вычислительной техники.

Рабочее место и взаимное расположение всех его элементов должно соответствовать антропометрическим, физическим и психологическим требованиям. Большое значение имеет также характер работы. В частности, при организации рабочего места программиста должны быть соблюдены следующие основные условия: оптимальное размещение оборудования, входящего в состав рабочего места и достаточное рабочее пространство, позволяющее осуществлять все необходимые движения и перемещения.

Эргономическими аспектами проектирования видеотерминальных рабочих мест, в частности, являются: высота рабочей поверхности, размеры пространства для ног, требования к расположению документов на рабочем месте (наличие и размеры подставки для документов, возможность различного размещения документов, расстояние от глаз пользователя до экрана, документа, клавиатуры и т.д.), характеристики рабочего кресла, требования к поверхности рабочего стола, регулируемость элементов рабочего места [26].

Заключение

При выполнении выпускной квалификационной работы было разработано распределенное клиент-серверное приложение для учета материально-производственных запасов предприятия и формирования инвентаризационных описей, состоящее из клиентского приложения и серверной базы данных.

Было проведено логическое и физическое проектирование реляционной базы данных, проектирование графического интерфейса.

Клиентская часть приложения реализована при помощи языка JavaScript, серверная часть приложения реализована при помощи языка программирования PHP в среде разработки «Notepad++». В СУБД с помощью языка SQL была реализована база данных, которая включает в себя таблицы и представление. База данных была реализована в СУБД «Oracle Databese 11g Release 2».

На этапе контроля качества программного обеспечения было проведено тестирование. На основе анализа результатов тестирования можно сделать вывод, что разработанное программное обеспечение работает корректно.

В ходе выполнения работы также было выполнено планирование проекта. Длительность выполнения проекта составила 93 дня, а бюджет равен 138 443 рубля. Оценка сроков и бюджета говорит о том, что данная разработка экономически выгодна.

В рамках данной выпускной квалификационной работы было уделено внимание вопросам безопасности жизнедеятельности.

Таким образом, в результате выполнения выпускной квалификационной работы было разработано программное обеспечение, отвечающее всем требованиям технического задания.

Список использованных источников

1. Распоряжение Правительства Российской Федерации от 20 октября 2010 г. N 1815-р г. Москва «О государственной программе Российской Федерации Информационное общество (2011-2020 годы)».

2. Овчинников М.А. Партионный учет материала и его организация в базе данных // Новые информационные технологии и системы (НИТиС): сборник научных статей XIII Международной научно-технической конференции. – Пенза: Изд-во ПГУ, 2016. – С. 7–9.

3. Бородина В.В. Инвентаризация в бухгалтерском учете: Учебно-практическое пособие / В.В. Бородина. – М: ОАО «Городец», 2005. – 320 с.

4. Приказ Минфина РФ от 13.06.1995 N 49 (ред. от 08.11.2010) «Об утверждении Методических указаний по инвентаризации имущества и финансовых обязательств».

5. «1С:Бухгалтерия 8»: описание программного продукта // ООО «1С»: сайт. 2007. [Электронный ресурс]. URL: https://v8.1c.ru/buhv8/ (дата обращения: 01.05.2020).

6. «Инвентаризация 3.0»: описание программного продукта // Softportal: сайт. 2002. [Электронный ресурс]. URL: https:// www.softportal.com/software-25689-inventarizatsiya.html (дата обращения: 01.05.2020).

7. «Складской учет»: описание программного продукта // Программа НМА: сайт. 2012. [Электронный ресурс]. URL: http://nm-a.ru/home/sklad_doc (дата обращения: 01.05.2020).

8. Крачтен Ф. Введение в Rational Unified Process / Ф. Крачтен. – М.: Вильямс, 2002. – 240 с.

9. Флэнаган Д. JavaScript. Подробное руководство, 6-е издание. / Д. Флэнаган. – Пер. с англ. – СПб: Символ-Плюс, 2012. – 1080 с.

10. Представляем Sencha Ext JS: статья // Хабрахабр: сайт. 2006. [Электронный ресурс]. URL: https://habrahabr.ru/post/225169/ (дата обращения: 23.02.2018).

11. Общий взгляд на AJAX фреймворк ExtJS: статья // Интерфейс: сайт. 2010. [Электронный ресурс]. URL: http://www.interface.ru/home.asp?artid=22109 (дата обращения: 01.03.2019).

12. Кайт Т. Oracle для профессионалов: архитектура и методики программирования, 3-е изд.: Пер. с англ. / Т. Кайт, Д. Кун. – М.: ООО «И.Д. Вильямс», 2016. – 960 с.

13. Notepad++: статья // Википедия: сайт. 2019. [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Notepad%2B%2B (дата обращения: 01.03.2019).

14. Мишель Е. Дэвис. Изучаем PHP и MySQL / Е. Мишель Дэвис. – Пер. с англ. – СПб.: Символ-Плюс, 2008. – 448 с.

15. Валиков А. Технология XSLT / А. Валиков. – СПб.: БХВ-Петербург, 2002. – 79 с.

16. Арлоу Д. UML 2 и Унифицированный процесс / Д. Арлоу, А. Нейштадт. – СПб.: Символ-Плюс, 2007. – 624 с.

17. Пайлон Д. UML 2 для программистов / Д. Пайлон, Н.Питмен. – СПб.: Питер, 2012. – 240 с.

18. Коналлен Дж. Разработка Web-приложений с использованием UML: пер. с англ. / Дж. Коналлен. – М.: Издательский дом «Вильямс», 2001. – 288 с.

19. Конноли Т. Базы данных: Проектирование, реализация и сопровождение. Теория и практика. Второе издание. / Т. Конноли, К. Бегг, А. Страчан. – М.: Вильямс, 2000. – 1139 с.

20. Новиков Д.А. Управление проектами: Организационные механизмы / Д.А. Новиков. – М.: ПМСОФТ, 2007. – 140 с.

21. Котляров В.П. Основы тестирования программного обеспечения / В.П. Котляров, Т.В. Коликова. – М.: Бином. Лаборатория знаний, 2006. – 285 с.

22. Канер С. Тестирование программного обеспечения / С. Канер, Д. Фолк, Е.К. Нгуен. – К.: ДиаСофт, 2001. – 544 с.

23. Дубовцев В.А. Безопасность жизнедеятельности. Расчет и проектирование средств защиты: учебное пособие / В.А. Дубовцев. – Киров.: Изд-во «ВятГУ», 2009. – 103 с.

24. Медведев В.Т. Охрана труда и промышленная экология. / В.Т. Медведев, С.Г. Новиков, А.В. Каралюнец. – М.: Издательский центр «Академия», 2012. – 416 с.

25. Белов С.В. Безопасность жизнедеятельности: Учебник для вузов. / С.В. Белов, А.В. Ильницкая, А.Ф. Козьяков. – М.: Высш.шк., 2007. – 616 с.

26. Кнорринг Г.М. Справочник для проектирования электрического освещения. / Г.М. Кнорринг. – М.: Книга по Требованию, 2012. – 381 с.

27. Юдин Е.Я. Борьба с шумом на производстве: Справочник / Е.Я. Юдин, Л.А. Борисов. – М.: Машиностроение, 1985. – 400с.

28. Мунипов В.М. Эргономика: человекоориентированное проектирование техники, программных средств и среды: Учебник. / В.М. Мунипов, В.П. Зинченко. – М.: Логос, 2001. – 356 с.

ГЛОССАРИЙ

Приложение А

(справочное)

ТМЦ (товарно-материальные ценности) – это активы предприятия, которые используются в производстве товаров и услуг, а также для управленческих нужд.

МПЗ (материально-производственные запасы) – это оборотные средства организации, характерные особенностью которых является то, что они полностью переносят свою стоимость на продукт труда в одном производственном цикле.

ИНВ (инвентаризация) – это проверка наличия, состояния и оценки имущества и обязательств организации.

ИНВ-3, ИНВ-5, ИНВ-19 – номера унифицированных форм отчетности по инвентаризации.

НДС (налог на добавленную стоимость) – косвенный налог, форма изъятия в бюджет государства части стоимости товара, работы или услуги, которая создаётся на всех стадиях процесса производства товаров.

XML (Extensible Markup Language) – это расширяемый язык разметки, разработанный специально для размещения информации в World Wide Web.

AJAX (Asynchronous JavaScript and XML) – подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером.

Фреймворк (Framework) – программная платформа, определяющая структуру программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта.

RIA-приложение – это веб-приложение, загружаемое пользователем через интернет, предназначенное для выполнения функций традиционных настольных приложений и работающее на устройстве пользователя.

URL-адрес – это специальный и индивидуальный адрес различных интернет ресурсов.

CSS (Cascading Style Sheets) – это каскадные таблицы стилей, используются для описания внешнего вида документа, написанного языком разметки.

VHDL (Very High Speed Integrated Circuits) – язык описания аппаратуры интегральных схем.

Verilog – это язык описания аппаратуры, используемый для описания и моделирования электронных систем.

STL (Standard Template Library) – набор согласованных обобщенных алгоритмов, контейнеров, средств доступа к их содержимому и различных вспомогательных функций.

Win32 API – общее наименование набора базовых функций интерфейсов программирования приложений операционных систем семейств Microsoft Windows.

Web-браузер – это программа, с помощью которой отрываются и выводятся на экран web-страницы в удобном для восприятия виде.

DDL (Data Definition Language) – язык описания данных, это семейство компьютерных языков, используемых в компьютерных программах для описания структуры баз данных.

1NF (first normal form) – первая нормальная форма (1НФ), базовая нормальная форма отношения в реляционной модели данных.

2NF (second normal form) – вторая нормальная форма (2НФ), одна из возможных нормальных форм отношения в реляционной базе данных.

3NF (third normal form) – третья нормальная форма (3НФ), одна из возможных нормальных форм отношения в реляционной базе данных.

ЛИСТИНГ КЛИЕНТСКОГО ПРИЛОЖЕНИЯ

Приложение Б

(рекомендуемое)

Листинг Б.1 – Исходный код файла index.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"
"http://www.w3.org/TR/REC-html40/loose.dtd">
<?
session_name('IO');
session_start();

$mes = Date(m);
$god = Date(Y);
if ($mes==1) {$mes = 12; $god = $god-1;}
else {$mes = $mes-1;}

echo "<script type=\"text/javascript\" language=\"javascript\">
                  sys_year = $god;
                  sys_mes  = $mes;
       </script>";
?>
<html>
<head>
    <title>Инвентаризация</title>

    <link rel="stylesheet" type="text/css" href="/ext3/resources/css/ext-
all.css" />
      <link rel="stylesheet" type="text/css" href="css/style.css" />

      <script type="text/javascript" src='/ext3/adapter/ext/ext-
base.js'></script>
      <script type="text/javascript" src='/ext3/ext-all.js'></script>
      <script type='text/javascript' src='/ext3/src/locale/ext-lang-
ru.js'></script>
      <script type='text/javascript'
src='js/plugins/GroupHeaderPlugin.js'></script>
      <script type='text/javascript'
src='js/plugins/Ext.ux.MaskFormattedTextField.js'></script>

      <script type="text/javascript" src="js/1_start.js"></script>
      <script type="text/javascript" src="js/2_modification.js"></script>
      <script type="text/javascript" src="js/3_add.js"></script>
      <script type="text/javascript" src="js/4_edit.js"></script>
      <script type="text/javascript" src="js/functions.js"></script>
</head>
<body>
      <center><font color=blue>Графа 'Фактическое наличие' заполняется только
в том случае, если фактические остатки отличаются от
бухгалтерских</font><center>
      <div id="main"/>
</body>
</html>
```

Листинг Б.2 – Исходный код файла 1_start.js

```
/** ==================== СТАРТОВОЕ ОКНО ================ **/


Ext.Ajax.timeout = 360000;

function msg(a){
      if (a.length <= 0) {a = 'Ошибка!';}
      Ext.Msg.show({
         title:'Сообщение',
         msg:a,
         buttons:Ext.Msg.OK,
         modal:true,
         icon:Ext.MessageBox.INFO
      });
}

function start(imol){

    var date_screen = new Ext.form.DateField({
        fieldLabel:'Отчетная дата',
        format:'d.m.Y',
            anchor:'99%',
            editable:false,
            minValue: new Date(2016,8,1),
            maxValue: new Date(),
            value: new Date(sys_year,sys_mes,0)
    });

      var store_mol = new Ext.data.Store({
            proxy: new Ext.data.HttpProxy({url:'phps/find_mol.php'}),
            autoLoad:true,
            reader:   new   Ext.data.JsonReader({root:'items',   totalProper-
ty:'totalCount'},[{name:'mol', mapping:'mol', type:'int'}])
    });

    var mol = new Ext.form.ComboBox({
            cls:'marg_2',
            xtype:'combo',
            fieldLabel:'Склад',
            emptyText:'Выберите склад',
            maskRe:/^[0-9]/,
            id:'v_mol',
            anchor:'99%',
            hiddenName:'mol',
            store: store_mol,
            displayField:'mol',
            valueField:'mol',
            maxLength:3,
            typeAhead:true,
            mode:'local',
            triggerAction:'all',
            editable:false,
            tabIndex:1
    });

    var store_bs = new Ext.data.SimpleStore({
        fields:['bs','name','id_bs'],
        data:[['002','Ответ.   хранение','297'],['003','Давальческий   матери-
ал','298']]
    });
```

```
    var bsTPL = new Ext.XTemplate(
        '<tpl for="."><div class="search-item">',
        '<font  style="font-size:11pt;  font-family:  Eurostile  Bold;  font-
weight:bold">{bs}</font> - {name}<hr/>',
        '</div></tpl>'
    );

    var bs = new Ext.form.ComboBox({
            fieldLabel:'ЗБ/Счёт',
            store:store_bs,
            displayField:'bs',
            hiddenName:'bs',
            valueField:'bs',
        anchor:'99%',
        disabled:true,
        typeAhead:true,
            listWidth:250,
            mode:'local',
            emptyText:'(только при ИНВ-5)',
            triggerAction:'all',
            editable:false,
            resizable:true ,
            tpl:bsTPL,
            itemSelector:'div.search-item',
            onSelect: function(record){
                  bs.setValue(record.data.bs);
                  bs.collapse();
                  mbp.setValue();
                  store_mbp.load({params:{id_bs:record.data.id_bs}});
            }
    });

    var store_mbp = new Ext.data.Store({
        proxy: new Ext.data.HttpProxy({url:'phps/find_mbp.php'}),
        reader:     new     Ext.data.JsonReader({root:'items',     totalProper-
ty:'total'},[{name:'mbp'},{name:'name', type:'string'}])
    });

    var mbpTPL = new Ext.XTemplate(
        '<tpl for="."><div class="search-item">',
        '<font  style="font-size:11pt;  font-family:  Eurostile  Bold;  font-
weight:bold">{mbp}</font> - {name}<hr/>',
        '</div></tpl>'
    );

    var mbp = new Ext.form.ComboBox({
            fieldLabel:'МБП',
            store:store_mbp,
            displayField:'mbp',
            hiddenName:'mbp',
            valueField:'mbp',
            anchor:'99%',
        typeAhead:true,
        disabled:true,
            listWidth:250,
            mode:'local',
            emptyText:'(только при ИНВ-5)',
            resizable:true ,
        triggerAction:'all',
        tpl:mbpTPL,
        itemSelector:'div.search-item',
        editable:false
      });
```

```
    var check_fact = new Ext.form.Checkbox({
        fieldLabel:'С фактическими остатками',
            anchor:'99%',
        name:'check_fact',
            checked:false
    });

    var check_space_title = new Ext.form.Checkbox({
        fieldLabel:'Пустой титульный лист',
            anchor:'99%',
        name:'check_space_title',
            checked:false,
            disabled:true
    });

      var Radio_doc = new Ext.form.RadioGroup({
            columns:1,
            hideLabel:true,
            items:[{
                  checked:true,
                  boxLabel:'Материалы (ИНВ-3)',
                  name:'Radio_doc',
                  inputValue:'1'
            },{
                  boxLabel:'Спец.инструмент и оснастка (ИНВ-3)',
                  name:'Radio_doc',
                  inputValue:'2'
            },{
                  boxLabel:'Инвентаризация по приходу (ИНВ-3)',
                  name:'Radio_doc',
                  inputValue:'3'
            },{
                  boxLabel:'Инвентаризация по расходу (ИНВ-3)',
                  name:'Radio_doc',
                  inputValue:'4'
            },{
                  boxLabel:'Инвентаризация по ГОЗ (ИНВ-3)',
                  name:'Radio_doc',
                  inputValue:'5'
            },{
                  boxLabel:'Забалансовый учет (ИНВ-5)',
                  name:'Radio_doc',
                  inputValue:'6'
            },{
                  boxLabel:'Ведомость фактических остатков',
                  name:'Radio_doc',
                  inputValue:'7'
            }],
            listeners:{'change': function(){
                  if         (Radio_doc.getValue().inputValue==1)         {but-
ton_view.enable();} else {button_view.disable();}
                  if                        (Radio_doc.getValue().inputValue==2)
{date_screen.disable();} else {date_screen.enable();}
                  if       (Radio_doc.getValue().inputValue==2       ||       Ra-
dio_doc.getValue().inputValue==3 || Radio_doc.getValue().inputValue==4 || Ra-
dio_doc.getValue().inputValue==5     ||     Radio_doc.getValue().inputValue==7)
{check_fact.disable();            check_space_title.disable();}            else
{check_fact.enable();}
            if         (Radio_doc.getValue().inputValue==3         ||         Ra-
dio_doc.getValue().inputValue==4 || Radio_doc.getValue().inputValue==5 || Ra-
dio_doc.getValue().inputValue==7)     {Radio_mode.disable();}     else     {Ra-
dio_mode.enable();}
```

```
                  if     (Radio_doc.getValue().inputValue==6)     {bs.enable();
mbp.enable();}     else     {bs.setValue();     bs.disable();     mbp.setValue();
mbp.disable(); store_mbp.removeAll();}
            }}
    });

      var Radio_mode = new Ext.form.RadioGroup({
            columns:1,
            title:'Режим печати',
            hideLabel:true,
            items:[{
                  boxLabel:'Титульный лист',
                  name:'Radio_mode',
                  inputValue:'1'
            },{
                  checked:true,
                  boxLabel:'Строки инв.описи',
                  name:'Radio_mode',
                  inputValue:'2'
            },{
                  boxLabel:'Заключительный лист',
                  name:'Radio_mode',
                  inputValue:'3'
            }],
            listeners:{'change': function(){
                  if      (Radio_doc.getValue().inputValue==2      ||      Ra-
dio_mode.getValue().inputValue==1    ||    Radio_mode.getValue().inputValue==3)
{check_fact.disable();} else {check_fact.enable();}
                  if      ((Radio_doc.getValue().inputValue==1      ||      Ra-
dio_doc.getValue().inputValue==6)    &&    Radio_mode.getValue().inputValue==1)
{check_space_title.enable();} else {check_space_title.disable();}
            }}
    });

      var button_view = new Ext.Button({
            text:'Корректировка',
            iconCls:'look',
            handler: function(){
                  if (Ext.isEmpty(mol.getValue())) {msg('Пожалуйста, выберите
склад из списка!'); return;}

                  if                        (Radio_doc.getValue().inputValue==1)
{XM531G01(date_screen.getValue(),mol.getValue()); return;}
                  if  (Radio_doc.getValue().inputValue==2)  {msg('Данный  режим
находиться в разработке'); return;}

                  msg('У данного документа нет возможности корректировки!');
return;
            }
    });

      var button_print = new Ext.Button({
            text:'Печать',
            iconCls:'print',
            handler: function(){
                  if (Ext.isEmpty(mol.getValue())) {msg('Пожалуйста, выберите
склад из списка!'); return;}

                  if (Radio_doc.getValue().inputValue==1){
                        if                                                (Ra-
dio_mode.getValue().inputValue==1){window.open("phps/inv_titul.php?mol="+mol.
getValue()+"&space="+check_space_title.getValue());}
                        if                                                (Ra-
dio_mode.getValue().inputValue==2){window.open("phps/inv.php?mes="+date_scree
```

```
n.getValue().format('m')+"&year="+date_screen.getValue().format('Y')+"&mol="+
mol.getValue()+"&fact="+check_fact.getValue());}
                        if                                                  (Ra-
dio_mode.getValue().inputValue==3){window.open("phps/inv_konec.php");}
                    }
                    if (Radio_doc.getValue().inputValue==2){
                        if                                                  (Ra-
dio_mode.getValue().inputValue==1){window.open("phps/excel_osnastca_title.php
?mol="+mol.getValue());}
                        if                                                  (Ra-
dio_mode.getValue().inputValue==2){window.open("phps/excel_osnastca.php?mol="
+mol.getValue());}
                        if                                                  (Ra-
dio_mode.getValue().inputValue==3){window.open("phps/inv_konec.php");}

                    }
                    if                                                      (Ra-
dio_doc.getValue().inputValue==3){window.open("phps/vigruzka.php?&date_genera
lm="+date_screen.getValue().format('m')+"&date_generalg="+date_screen.getValu
e().format('Y')+"&v_mol="+mol.getValue());}
                    if                                                      (Ra-
dio_doc.getValue().inputValue==4){window.open("phps/vigruzka.php?&date_genera
lm="+date_screen.getValue().format('m')+"&date_generalg="+date_screen.getValu
e().format('Y')+"&v_mol="+mol.getValue());}
                    if                                                      (Ra-
dio_doc.getValue().inputValue==5){window.open("phps/vigruzka.php?&date_genera
lm="+date_screen.getValue().format('m')+"&date_generalg="+date_screen.getValu
e().format('Y')+"&v_mol="+mol.getValue());}
                    if (Radio_doc.getValue().inputValue==6){
                  if   (Ext.isEmpty(bs.getValue()))   {msg('Пожалуйста,   выберите
забалансовый счет из списка!'); return;}

                        if                                                  (Ra-
dio_mode.getValue().inputValue==2){window.open('phps/vigruzka_phpexcel.php?mo
l='+mol.getValue()+'&year='+date_screen.getValue().format('Y')+'&mes='+date_s
creen.getValue().format('m')+'&fact='+check_fact.getValue()+'&bs='+bs.getValu
e()+'&mbp='+mbp.getValue());}
                        else                                               {win-
dow.open("phps/title.php?&mol="+mol.getValue()+'&bs='+bs.getValue()+'&mbp='+m
bp.getValue()+"&space="+check_space_title.getValue());}
                    }
                    if                                                      (Ra-
dio_doc.getValue().inputValue==7){window.open("phps/ved.php?mes="+date_screen
.getValue().format('m')+"&year="+date_screen.getValue().format('Y')+"&mol="+m
ol.getValue()+'&fact='+check_fact.getValue());}
                }
    });

      var button_help = new Ext.Button({
            text:'Помощь',
            iconCls:'help',
            tooltip:'Инструкция пользователя',
            handler: function(){w = window.open("instruction.pdf");}
      });

    Start_form = new Ext.FormPanel({
            width:580,
            style:'padding:5% 20% 0',
            frame:true,
            iconCls:'wand',
            labelAlign:'right',
            title:'<p align=left>Инвентаризация</p>',
            bodyStyle:'padding:10px 10px 0',
            items:[{
```

```
                cls:'marg_3',
                xtype:'fieldset',
                title:'Задайте параметры',
                items:[{
                      layout:'form',
                      labelWidth:90,
                      anchor:'60%',
                      items:[date_screen,mol,bs,mbp]
                }]
          },{
                layout:'column',
                items:[{
                      cls:'marg_3',
                      xtype:'fieldset',
                      width:'56%',
                      title:'Вид документа',
                      autoHeight:true,
                      items:[Radio_doc]
                },{
                      cls:'marg_3',
                      xtype:'fieldset',
                      width:'42%',
                      title:'Режим печати',
                      autoHeight:true,
                      items:[Radio_mode]
                }]
          },{
                layout:'form',
                labelWidth:160,
                anchor:'40%',
                style:'padding-top:10px',
                items:[check_fact,check_space_title]
          }],
        buttons:[button_view,button_print,button_help],
        buttonAlign:'center',
            renderTo:'main'
    });

      if (!Ext.isEmpty(imol)) {mol.setValue(imol);}
}

Ext.onReady(function(){start();});
```

Листинг Б.3 – Исходный код файла style.css

```
body {background-color:#E9F1FE;}

#main {overflow: auto;}

.back       {background-image:url(../images/back.gif)       !important;}
.change     {background-image:url(../images/saved.png)            !important;}
.add        {background-image:url(../images/drop-add.gif)   !important;}
.del        {background-image:url(../images/delete.gif)     !important;}
.print            {background-image:url(../images/printer.png)    !important;}
.del_part   {background-image:url(../images/delete_2.gif)   !important;}
.dob_part   {background-image:url(../images/drop-yes.gif)   !important;}
.bmenu            {background-image:url(../images/menu-show.gif)  0  no-repeat
!important;}
.help       {background-image:url(../images/help.gif)             !important;}
.look             {background-image:url(../images/layer.png)           !im-
portant;}
```

```
.arrow            {background-image:url(../images/arrow.png)           !im-
portant;}
.wand             {background-image:url(../images/wand.png)            !im-
portant;}
```

Листинг Б.4 – Исходный код файла 1_add.js

```
// Окно добавления нового остатка
function win_add(ss){
      Ext.Ajax.timeout = 360000;

      Ext.grid.dummyData = ss.st_cor;

      var reader = new Ext.data.ArrayReader({},[
            {name:'num_row_int'},
            {name:'bs',  type:'string'},
            {name:'mbp'},
            {name:'km'},
            {name:'km_name'},
            {name:'ke'},
            {name:'ke_name',  type:'string'},
            {name:'h701'},
            {name:'kol_fact'},
            {name:'c_fact'},
            {name:'sum_fact'},
            {name:'kol_buh'},
            {name:'c_buh'},
            {name:'sum_buh'}
      ]);

      Ext.grid.dummyData1 = ss.st_cor1;

      var reader1 = new Ext.data.ArrayReader({}, [
            {name:'num_row_int'},
            {name:'km'},
            {name:'ke'},
            {name:'kol'},
            {name:'pg'},
            {name:'drug'},
            {name:'np'},
            {name:'datv'},
            {name:'date_issuing_day'},
            {name:'date_issuing_mw'},
            {name:'date_issuing_year'},
            {name:'ei_datv'},
            {name:'gss'},
            {name:'ei_gss'},
            {name:'ind', type: 'int'},
      ]);

      function mm21Renderer(val,p,record){
            if (record.data.sum_fact>0) {return '<span
style="color:black;">'+val+' *</span>';}
            else {return val;}
      }
```

```
      var bs = new Ext.form.NumberField({
            allowBlank:true,
            allowDecimals:false,
            maxLength:4,
            maxLengthText:'Максимальное количество символов - 4'

      });

      var mbp = new Ext.form.NumberField({
            allowBlank:true,
            allowDecimals:false,
            maxLength:2,
            maxLengthText:'Максимальное количество символов - 2'

      });

      var km = new Ext.form.NumberField({
            allowBlank:true,
            maxLength:15,
            allowDecimals:false,
            maxLengthText:'Максимальное количество символов - 15'

      });

      km.on('change', function(obj)
{GR2.store.data.items[0].set('km',km.getValue())});

      var km_name = new Ext.form.TextField({
            allowBlank:true,
            maskRe:/^[A-яA-z0-9.,-/:()!#;+*% ]/
      });

      var ke = new Ext.form.NumberField({
            allowDecimals:false,
            allowBlank:true,
            maxLength:2,
            maxLengthText:'Максимальное количество символов - 2'

      });

      ke.on('change', function(obj)
{GR2.store.data.items[0].set('ke',ke.getValue());});

      var ke_name = new Ext.form.TextField({
            allowBlank:true,
            maxLength:10,
            maskRe:/^[A-яA-z0-9.,-/:()!#;+*% ]/,
            maxLengthText:'Максимальное количество символов - 10'

      });

      var h701 = new Ext.form.NumberField({
            allowNegative:false,
            decimalPrecision:2,
            decimalSeparator:',',
            maxValue:9999999.99
```

```
	});

	var kol_fact = new Ext.form.NumberField({
		maskRe:/^[0-9]+$/,
		readOnly:true
	});

	kol_fact.on('change', function(obj)
{GR1.store.data.items[0].set('sum_fact',h701.getValue()*kol_fact.getValue());
});

	var c_fact = new Ext.form.NumberField({
		allowNegative:false,
		decimalPrecision:2,
		decimalSeparator:',',
		maxLength:13,
		minValue:0.00,
		maxValue:999999999.99,
		allowBlank:true,
		maxLengthText:'Максимальное количество символов - 13'

	});

	var sum_fact = new Ext.form.NumberField({
		allowNegative:false,
		decimalPrecision:2,
		decimalSeparator:',',
		maxLength:13,
		allowBlank:true,
		maxLengthText:'Максимальное количество символов - 13'

	});

	var cm = new Ext.grid.ColumnModel({
		columns:[
			{header:"<center>МОЛ</center>",
			align:"center", width:40,  dataIndex:'mol',    hidden:true},
			{header:"<center>N п/п</center>",
		align:"center", width:40,  dataIndex:'num_row_int',  render-
er:mm21Renderer},
			{header:"<center>Счёт*</center>",
		align:"center", width:45,  dataIndex:'bs',           editor:bs},
			{header:"<center>ИХП*</center>",
		align:"center", width:40,  dataIndex:'mbp',    editor:mbp},
			{header:"<center>код (номекла-<br>турный номер)</center>",
	align:"center", width:105, dataIndex:'km',           editor:km},
			{header:"<center>наименование, характеристика<br>(вид, сорт,
группа)*</center>",align:"center", width:232, dataIndex:'km_name', edi-
tor:km_name},
			{header:"<center>код<br>ЕИ*</center>",
		align:"center", width:45,  dataIndex:'ke',           editor:ke},
			{header:"<center>наиме-<br>нование*</center>",
	align:"center", width:60,  dataIndex:'ke_name', editor:ke_name},
			{header:"<center>усреднённая,<br>руб. коп*</center>",
	align:"center", width:85,  dataIndex:'h701',   editor:h701},
```

```
                  {header:"<center>кол-во</center>",
            align:"center", width:73,  dataIndex:'kol_fact',editor:kol_fact},
                  {header:"<center>цена учётн,<br>руб. коп</center>",
      align:"center", width:73,  dataIndex:'c_fact', editor:c_fact},
                  {header:"<center>сумма,<br>руб. коп*</center>",
      align:"center", width:73,  dataIndex:'sum_fact',editor:sum_fact},
                  {header:"<center>кол-во</center>",
            align:"center", width:73,  dataIndex:'kol_buh'},
                  {header:"<center>цена учётн,<br>руб. коп</center>",
      align:"center", width:73,  dataIndex:'c_buh'},
                  {header:"<center>сумма,<br>руб. коп</center>",
      align:"center", width:73,  dataIndex:'sum_buh'}
            ],
            rows:[[{},{},{},{},
                  {header:'Товарно-материальные ценности', colspan:2,
align:'center'},
                  {header:'Единица измерения', colspan:2, align: 'center'},
                  {header:'Цена', align:'center'},
                  {header:'Фактическое наличие', colspan:3, align:'center'},
                  {header:'По данным бухгалтерского учета', colspan:3,
align:'center'},
            ]]
      });

      GR1 = new Ext.grid.EditorGridPanel({
            title:getShifr('Добавление строки в инвентаризационную ведомость
ТМЦ','XM531G02',1),
            height:155,
            width:1095,
            store: new Ext.data.Store({reader:reader, da-
ta:Ext.grid.dummyData}),
            colModel:cm,
            enableColumnMove:false,
            loadMask:true,
            clicksToEdit:1,
            enableHdMenu:true,
            plugins:[new Ext.ux.plugins.GroupHeaderGrid()]
      });

      var csm = new Ext.grid.CheckboxSelectionModel({singleSelect:true,
width:25, header:'<div style="background:#F0F1F3; height:90%;></div>'});

      var number = new Ext.grid.RowNumberer({header:'N п/п',width:35});

      Ext.util.Format.comboRenderer = function(combo){
            return function(value){
                  var record = combo.findRecord(combo.valueField, value);
                  return record ? record.get(combo.displayField) : com-
bo.valueNotFoundText;
            }
      }

      var kol = new Ext.form.NumberField({
            allowNegative:false,
            decimalPrecision:3,
            decimalSeparator:',',
```

```
            maxLength:10,
            allowBlank:true,
            allowDecimals:true,
            maxLengthText:'Максимальное количество символов - 10'

      });

      kol.on('change', function(obj){
            GR1.store.data.items[0].set('kol_fact',kol.getValue());
            if (!Ext.isEmpty(kol.getValue()) &&
!Ext.isEmpty(GR1.store.getModifiedRecords()[0].data['h701']))
{GR1.store.data.items[0].set('sum_fact',(eval(parseFloat(kol.getValue())+'*'+
parseFloat(GR1.store.getModifiedRecords()[0].data['h701'])).toFixed(2)));}
      });

      var pg = new Ext.form.ComboBox({
            labelSeparator:'',
            hiddenName:'name',
            store: new Ext.data.SimpleStore({fields:['id','name'], da-
ta:[['1','ЭРИ'],['2','Металлы'],['3','Неметаллические материалы'],['4','Общие
материалы']]}),
            anchor:'76%',
            displayField:'name',
            valueField:'id',
            typeAhead:true,
            mode:'local',
            triggerAction:'all',
            editable:false,
            resizable:true
      });

      var drug = new Ext.form.ComboBox({
            labelSeparator:'',
            hiddenName:'name',
            store: new Ext.data.SimpleStore({fields:['id','name'], da-
ta:[['0','без содерж.'],['1','с содерж.']]}),
            anchor:'76%',
            displayField:'name',
            valueField:'id',
            typeAhead: true,
            mode: 'local',
            triggerAction: 'all',
            editable:false,
            resizable: true
      });

      var np = new Ext.form.TextField({
            allowBlank:true,
            maskRe:/^[A-яA-z0-9.,-/:()!#;+*% ]/
      });

      var date_issuing_day = new Ext.form.NumberField({
            anchor:'86%',
            minValue:1,
            maxValue:31,
            allowDecimals:false
```

```
	});

	var date_issuing_mw = new Ext.form.NumberField({
		anchor:'86%',
		minValue:1,
		allowBlank:true,
		maxValue:53,
		allowDecimals:false
	});

	var date_issuing_year = new Ext.form.NumberField({
		allowDecimals:false,
		anchor:'86%',
		allowBlank:true,
		minValue:1950,
		maxValue:2100
	});

	var ei_datv = new Ext.form.ComboBox({
		labelSeparator:'',
		hiddenName:'name',
		store: new Ext.data.SimpleStore({fields:['id','name'], da-
ta:[['0','Без даты'],['1','Дата'],['2','Недели']]}),
		anchor:'76%',
		displayField:'name',
		valueField:'id',
		typeAhead: true,
		mode:'local',
		triggerAction:'all',
		editable:false,
		resizable:true
	});

	var gss = new Ext.form.NumberField({
		allowDecimals:false,
		maxLength:4,
		allowBlank:true,
		maxLengthText:'Максимальное количество символов - 4'

	});

	var ei_gss = new Ext.form.ComboBox({
		labelSeparator:'',
		hiddenName:'name',
		store: new Ext.data.SimpleStore({fields:['id','name'], da-
ta:[['1','Год'],['2','Месяц'],['3','Неделя']]}),
		anchor:'76%',
		displayField:'name',
		valueField:'id',
		typeAhead:true,
		mode:'local',
		triggerAction:'all',
		editable:false,
		resizable:true
	});
```

```
      var cm2 = new Ext.grid.ColumnModel({
            columns:[csm,
                  {header:"<center>N п/п</center>",
                  width:40,  align:'center', dataIndex:'num_row_int'},
                  {header:"<center>Код ТМЦ</center>",
                  width:105, align:'center', dataIndex:'km'},
                  {header:"<center>Код ЕИ</center>",
                  width:50,  align:'center', dataIndex:'ke'},
                  {header:"<center>Количество</center>",
                  width:80,  align:'center', dataIndex:'kol',
      editor:kol},
                  {header:"<center>Признак группы</center>",
                  width:160, align:'center', dataIndex:'pg',
      editor:pg,              renderer:Ext.util.Format.comboRenderer(pg)},
                  {header:"<center>Признак
содержания<br>драгметалла</center>",     width:120, align:'center', dataIn-
dex:'drug',             editor:drug,           render-
er:Ext.util.Format.comboRenderer(drug)},
                  {header:"<center>Наименование<br>партии</center>",
            width:85,  align:'center', dataIndex:'np',                 edi-
tor:np},
                  {header: "<center>Дата изг./<br>день</center>",
            width:70,  align:'center', dataIndex:'date_issuing_day',   edi-
tor:date_issuing_day},
                  {header: "<center>Дата изг./<br>мес-нед</center>",
            width:70,  align:'center', dataIndex:'date_issuing_mw',    edi-
tor:date_issuing_mw},
                  {header: "<center>Дата изг./<br>год</center>",
            width:70,  align:'center', dataIndex:'date_issuing_year',  edi-
tor:date_issuing_year},
                  {header: "<center>ЕИ даты<br>выпуска</center>",
            width:65,  align:'center', dataIndex:'ei_datv',      edi-
tor:ei_datv,            renderer:Ext.util.Format.comboRenderer(ei_datv)},
                  {header: "<center>Гарантийный<br>срок хранения</center>",
      width:85,  align:'center', dataIndex:'gss',               editor:gss},

                  {header: "<center>ЕИ срока<br>хранения</center>",
            width:65,  align:'center', dataIndex:'ei_gss',       edi-
tor:ei_gss,       renderer:Ext.util.Format.comboRenderer(ei_gss)}
            ]
      });
      cm.defaultSortable = false;

      GR2 = new Ext.grid.EditorGridPanel({
            title:'Партии в строке инветаризационной описи',
            height:180,
            width:1095,
            store: new Ext.data.Store({reader:reader1, da-
ta:Ext.grid.dummyData1}),
            colModel:cm2,
            sm:csm,
            enableColumnMove:false,
            loadMask:true,
            clicksToEdit:1,
            enableHdMenu:true
      });
```

```
	var button_enter = new Ext.Button({
		text:'Ввод',
		minWidth:50,
		handler: function(){vvod_stroki();}
	});

	var button_auto = new Ext.Button({
		text:'Автоматический ввод*',
		minWidth:50,
		handler: function(){auto_enter();}
	});

	var button_cancel = new Ext.Button({
		text:'Закрыть',
		minWidth:50,
		tabIndex:15,
		handler: function(){
			window.close();
			store.baseParams =
{num_row_int:find_num_row_int.getValue(),km:find_km.getValue(),bs:find_bs.get
Value(),mbp:find_mbp.getValue()};
			store.load({params:{start:0,limit:20},
callback:function(opts,suss,resp){
				if (store.getCount()<2) {button_add.disable(); but-
ton_edit.disable(); button_remove.disable();}
				else {button_add.enable(); button_edit.enable(); but-
ton_remove.enable();}
			}});
		}
	});
	/*
	function dob_stroki() {
		nomer=GR2.store.getCount();//общее число
		nomer=nomer+1;
		var u = new GR2.store.recordType({
		num_row_int:nomer,
		km:'',//GR2.store.data.items[0].data['km'],
		ke:'',//GR2.store.data.items[0].data['ke'],
		kol:'',
		pg:'',
		drug:'',
		np:'',
		datv:'',
		ei_datv:'',
		gss:'',
		ei_gss:'',
		ind: 1});
		GR2.stopEditing();
		GR2.store.insert(GR2.store.getCount(), u);
		GR2.startEditing(0, 5);
	}*/

	function auto_enter(){
		if (Ext.isEmpty(GR1.store.data.items[0].data['km']) ||
Ext.isEmpty(GR1.store.data.items[0].data['km'])){
```

```
                msg('Для автоматического ввода необходимо заполнить по-
ля:<br>-товарно-материальные ценности / код (номенклатурный номер),<br>-
фактическое наличие / цена учетная,<br>-количество в справочнике партий');
                return;
          }

          if (Ext.isEmpty(GR1.store.getModifiedRecords()[0].data['km']) ||
Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['kol']) ||
Ext.isEmpty(GR1.store.getModifiedRecords()[0].data['c_fact'])){
                msg('Для автоматического ввода необходимо заполнить по-
ля:<br>-товарно-материальные ценности / код (номенклатурный номер),<br>-
фактическое наличие / цена учетная,<br>-количество в справочнике партий');
                return;
          }

          Ext.Ajax.request({
                url:'phps/3_auto.php',
                method:'post',
                params:{
                      mol:  ss.mol,
                      mes:  ss.mes,
                      year: ss.year,
                      //строка первого Grid'a
                      bs:   GR1.store.getModifiedRecords()[0].data['bs'],
                      mbp:  GR1.store.getModifiedRecords()[0].data['mbp'],
                      km:   GR1.store.getModifiedRecords()[0].data['km'],
                      ke:   GR1.store.getModifiedRecords()[0].data['ke'],
                      h701: GR1.store.getModifiedRecords()[0].data['c_fact']
                },
                callback: function(opts,suss,resp){
                      var ff = Ext.decode(resp.responseText);
                      if (ff.success==1){
                            GR1.store.data.items[0].set('bs',ff.bs);
                            GR1.store.data.items[0].set('mbp',ff.mbp);

    GR1.store.data.items[0].set('km_name',ff.km_name);
                            GR1.store.data.items[0].set('ke',ff.ke);
                            GR2.store.data.items[0].set('ke',ff.ke);

    GR1.store.data.items[0].set('ke_name',ff.ke_name);
                            GR1.store.data.items[0].set('h701',ff.csr);

    GR1.store.data.items[0].set('sum_fact',(eval(parseFloat(kol.getValue())+
'*'+parseFloat(ff.csr)).toFixed(2)));
                      }
                      else {msg(ff.msg);}
                }
          });
    }

    //Функция,по описанию функциии для кнопки "Ввод"
    function vvod_stroki() {
          GR2.store.data.items[0].set('ind',0);

          if (GR1.store.getModifiedRecords()[0].data['km']=='' ||
GR1.store.getModifiedRecords()[0].data['kol_fact']=='' ||
```

```
                GR1.store.getModifiedRecords()[0].data['bs']=='' ||
(GR1.store.getModifiedRecords()[0].data['mbp']=='' &&
GR1.store.getModifiedRecords()[0].data['mbp']!=0) ||
                GR1.store.getModifiedRecords()[0].data['ke']=='' ||
GR1.store.getModifiedRecords()[0].data['c_fact']=='' ||
                GR2.store.getModifiedRecords()[0].data['pg']==''){
                      msg('Для добавления строки должны быть заполнены по-
ля:<br>  - Счёт<br>  - ИХП<br>  - код (номенклатурный номер)<br>  - код
ЕИ<br>  - Фактическое наличие / цена учётн, руб. коп.<br><br>  - Партии в
строке / Количество<br>  - Партии в строке / Признак группы');
                      return;
            }

            if (GR1.store.getModifiedRecords()[0].data['kol_fact']<0.001)
{msg('Кол-во должно быть больше нуля!'); return;}

            if (GR2.store.getModifiedRecords()[0].data['ei_datv']==1 &&
(Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['date_issuing_day']) ||
Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['date_issuing_mw']) ||
Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['date_issuing_year']))){
                msg('Вы ввели недостаточно данных для сохранения даты выпус-
ка!'); return;
            }

            if (GR2.store.getModifiedRecords()[0].data['ei_datv']==2 &&
Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['date_issuing_mw'])){
                 msg('Вы ввели недостаточно данных для сохранения даты вы-
пуска!'); return;
            }

            if (GR2.store.getModifiedRecords()[0].data['ei_datv']==1 && (

      (GR2.store.getModifiedRecords()[0].data['date_issuing_mw']==2 &&
GR2.store.getModifiedRecords()[0].data['date_issuing_day']>28) ||
               ((GR2.store.getModifiedRecords()[0].data['date_issuing_mw']==4
|| GR2.store.getModifiedRecords()[0].data['date_issuing_mw']==6 ||
GR2.store.getModifiedRecords()[0].data['date_issuing_mw']==9 ||
GR2.store.getModifiedRecords()[0].data['date_issuing_mw']==11) &&
GR2.store.getModifiedRecords()[0].data['date_issuing_day']>30) ||
GR2.store.getModifiedRecords()[0].data['date_issuing_mw']>12)){
                      msg('Вы указали неверные данные о дате изготовления
материала!'); return;
            }

            if ((Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['gss']) &&
!Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['ei_gss'])) ||
                 (!Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['gss'])
&& Ext.isEmpty(GR2.store.getModifiedRecords()[0].data['ei_gss']))){
                      msg('Введите достаточно данных для полей Гарантийный
срок хранения и ЕИ срока хранения!'); return;
            }

            Ext.Ajax.request({
                  url:'phps/3_add.php',
                  method:'post',
                  params:{
```

```
                        mol: ss.mol,
                        mes: ss.mes,
                        year:ss.year,
                        //строки верхнего Grid'a
                        num_row_int:
GR1.store.getModifiedRecords()[0].data['num_row_int'],
                        bs:
GR1.store.getModifiedRecords()[0].data['bs'],
                        mbp:
GR1.store.getModifiedRecords()[0].data['mbp'],
                        km:
GR1.store.getModifiedRecords()[0].data['km'],
                        ke:
GR1.store.getModifiedRecords()[0].data['ke'],
                        kol_fact:
GR1.store.getModifiedRecords()[0].data['kol_fact'],
                        c_fact:
GR1.store.getModifiedRecords()[0].data['c_fact'],
                        //строки нижнего Grid'a
                        pg:
      GR2.store.getModifiedRecords()[0].data['pg'],
                        drug:
      GR2.store.getModifiedRecords()[0].data['drug'],
                        np:
      GR2.store.getModifiedRecords()[0].data['np'],
                        date_issuing_day:
      GR2.store.getModifiedRecords()[0].data['date_issuing_day'],
                        date_issuing_mw:
      GR2.store.getModifiedRecords()[0].data['date_issuing_mw'],
                        date_issuing_year:
      GR2.store.getModifiedRecords()[0].data['date_issuing_year'],
                        ei_datv:
      GR2.store.getModifiedRecords()[0].data['ei_datv'],
                        gss:
      GR2.store.getModifiedRecords()[0].data['gss'],
                        ei_gss:
      GR2.store.getModifiedRecords()[0].data['ei_gss']
                  },
                  callback: function(opts,suss,resp){
                        var ff = Ext.decode(resp.responseText);
                        if (ff.success==1){
                              msg('В инвентаризационную ведомость успешно до-
бавлена новая строка!');
                              bs.disable(); mbp.disable(); km.disable();
km_name.disable(); ke.disable(); ke_name.disable(); h701.disable();
c_fact.disable();
                              kol.disable(); kol_fact.disable();
sum_fact.disable(); pg.disable(); drug.disable(); np.disable();
date_issuing_day.disable(); date_issuing_mw.disable();
                              date_issuing_year.disable(); ei_datv.disable();
gss.disable(); ei_gss.disable(); button_enter.disable(); but-
ton_auto.disable();

                              store.load({params:{start:0,limit:20},
callback:function(opts,suss,resp){
```

```
                                    if (store.getCount()<2) {but-
ton_add.disable(); button_edit.disable(); button_remove.disable();}
                                    else {button_add.enable(); but-
ton_edit.enable(); button_remove.enable();}
                                }});
                            }
                            else {msg(ff.msg);}
                        }
                });
        }

        var window = new Ext.Window({
                width: 1110,
                height:400,
                layout:'form',
                modal:true,
                resizable:false,
                plain:true,
                buttonAlign:'center',
                items:[{
                        layout:'form',
                        border:false,
                        items:[GR1]
                },{
                        layout:'form',
                        border:false,
                        items:[GR2]
                }],
                buttons:[button_auto,button_enter,button_cancel]
        });
        window.show();
}


Ext.grid.CheckColumn = function(config){
    Ext.apply(this, config);
    if(!this.id){this.id = Ext.id();}
    this.renderer = this.renderer.createDelegate(this);
}

Ext.grid.CheckColumn.prototype = {
    init : function(grid){
        this.grid = grid;
        this.grid.on('render', function(){
            var view = this.grid.getView();
            view.mainBody.on('mousedown', this.onMouseDown, this);
        }, this);
    },

    onMouseDown: function(e,t){
        if(t.className && t.className.indexOf('x-grid3-cc-'+this.id) != -1){
            e.stopEvent();
            var index = this.grid.getView().findRowIndex(t);
            var record = this.grid.store.getAt(index);
```

```
                if (record.get('ind')==1) {record.set(this.dataIndex, !rec-
ord.data[this.dataIndex]);}
        }
    },
    renderer: function(v,p,record){
        p.css += ' x-grid3-check-col-td';
        return '<div class="x-grid3-check-col'+(v?'-on':'')+' x-grid3-cc-
'+this.id+'"> </div>';
    }
}
```

Листинг Б.5 – Исходный код файла 4_load.php

```
<?PHP
REQUIRE('F_IS_DB.PHP');
REQUIRE_ONCE('../JSON/JSON.PHP');

$SH = 'XM531';
$ITEMS = ARRAY();
$JSON  = NEW SERVICES_JSON();

$MES  = $_POST['MES'];
$YEAR = $_POST['YEAR'];
$MOL  = $_POST['MOL'];
$BS     = SUBSTR($_POST['BS'],0,4);
$MBP  = SUBSTR($_POST['BS'],-2,2);
$KM     = $_POST['KM'];
$KE     = $_POST['KE'];
$H701 = NUMBER_FORMAT($_POST['H701'],2,'.','');
$NUM_ROW_INT = $_POST['NUM_ROW_INT'];
$ID_PART = $_POST['ID_PART'];
$PR     = $_POST['PR'];

IF ($PR==0) {DIE($JSON->ENCODE(ARRAY('TOTALCOUNT'=>0, 'ITEMS'=>ARRAY())));}
IF (!EMPTY($ID_PART) && $PR==2) {$FF = " WHERE ID_PART=$ID_PART";}

$DATE_GENERAL = $YEAR.$MES;

$SQL = "SELECT
ID_PART,GROUP_TMC,DRAG_MET,NAME_PART,DATE_ISSUING,GSH,GSH1,DATE_ISSUING_DAY,D
ATE_ISSUING_MW,DATE_ISSUING_YEAR,CE27,NUM_ROW_INT,BS,MBP,KM,$SH.KM_NAME(KM)
KM_NAME,KE,$SH.KE_NAME(KE) KE_NAME,CSR,H701,
                KOL_FACT,SUM_FACT,KOL_BUH,SUM_BUH
            FROM(
                SELECT
CC.ID_PART,CC.GROUP_TMC,CC.DRAG_MET,CC.NAME_PART,CC.DATE_ISSUING,CC.GSH,CC.GS
H1,DD.CE27,AA.MOL,AA.NUM_ROW_INT,AA.BS,AA.MBP,AA.KM,CC.KE,
                        AA.CSR,AA.H701,DD.KOL KOL_FACT,ROUND(DD.KOL*CSR,2)
SUM_FACT,BB.KOL KOL_BUH,BB.KOL*AA.CSR SUM_BUH,
                        CASE CC.DATE_ISSUING WHEN 1 THEN
TO_CHAR(CC.DATE_ISSUING,'DD') ELSE '' END DATE_ISSUING_DAY,
                        CASE CC.DATE_ISSUING WHEN 1 THEN
TO_CHAR(CC.DATE_ISSUING,'MM') WHEN 2 THEN
TO_CHAR(TO_NUMBER(TO_CHAR(CC.DATE_ISSUING,'DDD'))/7) ELSE '' END
DATE_ISSUING_MW,
```

```
                        CASE CC.DATE_ISSUING WHEN 1 THEN
TO_CHAR(CC.DATE_ISSUING,'YYYY') WHEN 2 THEN TO_CHAR(CC.DATE_ISSUING,'YYYY')
ELSE '' END DATE_ISSUING_YEAR
                  FROM(
                        SELECT A.MOL,ROWNUM NUM_ROW_INT,A.BS,A.MBP,A.KM,A.KE,
                              CASE
$SH.F_CSR_OBOR($YEAR,$MES,0,$MOL,BS,MBP,KM,H701,5,11) WHEN 0.00 THEN H701
                              ELSE
$SH.F_CSR_OBOR($YEAR,$MES,0,$MOL,BS,MBP,KM,H701,5,11) END CSR,H701
                        FROM(
                              SELECT
C.MOL,C.BS,C.MBP,B.KM,B.KE,C.H701,SUM(KOL) FROM $SH.PART B, $SH.REMAINS C
                              WHERE C.BS!=1000 AND
TO_CHAR(C.DATE_GENERAL,'YYYYMM')=$DATE_GENERAL AND C.MOL=$MOL AND
C.ID_PART=B.ID_PART
                     GROUP BY C.MOL,C.BS,C.MBP,B.KM,B.KE,C.H701
                              ORDER BY BS,MBP,KM,H701) A
                  ) AA JOIN $SH.REMAINS BB ON AA.MOL=BB.MOL AND AA.BS=BB.BS
AND AA.MBP=BB.MBP AND AA.H701=BB.H701
                 JOIN $SH.PART CC ON BB.ID_PART=CC.ID_PART AND AA.KM=CC.KM
                 LEFT JOIN $SH.MODIFICATION_REMAINS DD ON AA.MOL=DD.MOL AND
AA.NUM_ROW_INT=DD.NUM_ROW_INT AND BB.ID_PART=DD.ID_PART AND
TO_NUMBER(TO_CHAR(DD.DATE_GENERAL,'YYYYMM'))=$DATE_GENERAL
                  WHERE BB.BS!=1000 AND
TO_NUMBER(TO_CHAR(BB.DATE_GENERAL,'YYYYMM'))=$DATE_GENERAL AND AA.BS=$BS AND
AA.MBP=$MBP AND AA.KM=$KM AND AA.KE=$KE AND AA.H701=$H701
                  UNION ALL
                  SELECT
CC.ID_PART,CC.GROUP_TMC,CC.DRAG_MET,CC.NAME_PART,CC.DATE_ISSUING,CC.GSH,CC.GS
H1,BB.CE27,BB.MOL,BB.NUM_ROW_INT,BB.BS,BB.MBP,CC.KM,CC.KE,
               CASE
$SH.F_CSR_OBOR($YEAR,$MES,0,$MOL,BB.BS,BB.MBP,CC.KM,BB.H701,5,11) WHEN 0.00
THEN BB.H701
               ELSE
$SH.F_CSR_OBOR($YEAR,$MES,0,$MOL,BB.BS,BB.MBP,CC.KM,BB.H701,5,11) END CSR,
BB.H701,BB.KOL KOL_FACT,ROUND(
               CASE
$SH.F_CSR_OBOR($YEAR,$MES,0,$MOL,BB.BS,BB.MBP,CC.KM,BB.H701,5,11) WHEN 0.00
THEN BB.H701
               ELSE
$SH.F_CSR_OBOR($YEAR,$MES,0,$MOL,BB.BS,BB.MBP,CC.KM,BB.H701,5,11)
END*BB.KOL,2) SUM_FACT,0 KOL_BUH,0 SUM_BUH,
               CASE CC.DATE_ISSUING WHEN 1 THEN TO_CHAR(CC.DATE_ISSUING,'DD')
ELSE '' END DATE_ISSUING_DAY,
               CASE CC.DATE_ISSUING WHEN 1 THEN TO_CHAR(CC.DATE_ISSUING,'MM')
WHEN 2 THEN TO_CHAR(TO_NUMBER(TO_CHAR(CC.DATE_ISSUING,'DDD'))/7) ELSE '' END
DATE_ISSUING_MW,
               CASE CC.DATE_ISSUING WHEN 1 THEN
TO_CHAR(CC.DATE_ISSUING,'YYYY') WHEN 2 THEN TO_CHAR(CC.DATE_ISSUING,'YYYY')
ELSE '' END DATE_ISSUING_YEAR
                  FROM $SH.MODIFICATION_REMAINS BB, $SH.PART CC
                  WHERE BB.ID_PART=CC.ID_PART AND
TO_CHAR(BB.DATE_GENERAL,'YYYYMM')=$DATE_GENERAL AND MOL=$MOL AND BS=$BS AND
MBP=$MBP AND KM=$KM AND KE=$KE AND H701=$H701 AND
                        NOT EXISTS (SELECT * FROM $SH.PART B, $SH.REMAINS C
```

```
                        WHERE C.BS!=1000 AND
TO_CHAR(C.DATE_GENERAL,'YYYYMM')=$DATE_GENERAL AND C.MOL=$MOL AND
BB.MOL=C.MOL AND BB.BS=C.BS AND BB.MBP=C.MBP AND BB.ID_PART=C.ID_PART AND
BB.H701=C.H701)) $FF";
$CURS = OCIPARSE($CONN,$SQL);
IF (!OCIEXECUTE($CURS)) {DIE($SQL);}
$NROWS = OCIFETCHSTATEMENT($CURS,$ROWS);

IF ($PR==1){
      FOR ($K=0; $K<$NROWS; $K++){
            IF ($ROWS['KOL_FACT'][$K]!='') {$ROWS['KOL_FACT'][$K] = NUM-
BER_FORMAT($ROWS['KOL_FACT'][$K],3,'.',''); $ROWS['SUM_FACT'][$K] = NUM-
BER_FORMAT($ROWS['SUM_FACT'][$K],2,'.','');}
            ARRAY_PUSH($ITEMS,ARRAY('ID_PART'        => $ROWS['ID_PART'][$K],
                                                'NUM_ROW_INT'     =>
$NUM_ROW_INT,
                                                'KM'              =>
$ROWS['KM'][$K],
                                                'BS'              =>
$ROWS['BS'][$K],
                                                'MBP'       =>
$ROWS['MBP'][$K],
                                                'KM_NAME'   => ICONV('KOI8-
R','UTF-8',$ROWS['KM_NAME'][$K]),
                                                'KE'        =>
$ROWS['KE'][$K],
                                                'KE_NAME'   => ICONV('KOI8-
R','UTF-8',$ROWS['KE_NAME'][$K]),
                                                'CSR'       => NUM-
BER_FORMAT($ROWS['CSR'][$K],2,'.',''),
                                                'KOL_FACT'  =>
$ROWS['KOL_FACT'][$K],
                                                'C_FACT'    => NUM-
BER_FORMAT($ROWS['H701'][$K],2,'.',''),
                                                'SUM_FACT'  =>
$ROWS['SUM_FACT'][$K],
                                                'KOL_BUH'   => NUM-
BER_FORMAT($ROWS['KOL_BUH'][$K],3,'.',''),
                                                'C_BUH'           => NUM-
BER_FORMAT($ROWS['H701'][$K],2,'.',''),
                                                'SUM_BUH'   => NUM-
BER_FORMAT($ROWS['SUM_BUH'][$K],2,'.','')));
      }
}
ELSEIF ($NROWS>1) {DIE('OOPS! DUBLICATE PARTS!');}

ELSE{
      IF ($ROWS['KOL_FACT'][0]!='') {$ROWS['KOL_FACT'][0] = NUM-
BER_FORMAT($ROWS['KOL_FACT'][0],3,'.','');}
      IF ($ROWS['GSH'][0]==0) {$ROWS['GSH'][0] = '';}
      ARRAY_PUSH($ITEMS,ARRAY('ID_PART'        => $ROWS['ID_PART'][0],
                                          'KM'              => $ROWS['KM'][0],
                                          'KE'              => $ROWS['KE'][0],
                                          'KOL_FACT'  => $ROWS['KOL_FACT'][0],
                                          'KOL_BUH'   => NUM-
BER_FORMAT($ROWS['KOL_BUH'][0],3,'.',''),
```

```
                                        'PG'        =>
$ROWS['GROUP_TMC'][0],
                                        'DRUG'      => $ROWS['DRAG_MET'][0],
                                        'NP'        => ICONV('KOI8-R','UTF-
8',$ROWS['NAME_PART'][0]),
                                        'DATE_ISSUING_DAY'      =>
$ROWS['DATE_ISSUING_DAY'][0],
                                        'DATE_ISSUING_MW'   =>
$ROWS['DATE_ISSUING_MW'][0],
                                        'DATE_ISSUING_YEAR' =>
$ROWS['DATE_ISSUING_YEAR'][0],
                                        'EI_DATV'   =>
$ROWS['DATE_ISSUING'][0],
                                        'GSS'       => $ROWS['GSH'][0],
                                        'EI_GSS'    => $ROWS['GSH1'][0]));
}

$ARR = ARRAY('TOTALCOUNT'=>$NROWS, 'ITEMS'=>$ITEMS);
OCILOGOFF($CONN);
$OUTPUT = $JSON->ENCODE($ARR);
ECHO($OUTPUT);
?>
```

Листинг Б.6 – Исходный код файла 4_edit.php

```
<?PHP
REQUIRE('F_IS_DB.PHP');
REQUIRE_ONCE('../JSON/JSON.PHP');

$SH = 'XM531';

$MOL  = $_POST['MOL'];
$NUM_ROW_INT      = $_POST['NUM_ROW_INT'];
$BS         = $_POST['BS'];
$MBP  = $_POST['MBP'];
$C_FACT     = $_POST['C_FACT'];
$KOL  = $_POST['KOL'];
$ID_PART    = $_POST['ID_PART'];
$MES  = $_POST['MES'];
$YEAR = $_POST['YEAR'];

$DATE_GENERAL ="01.".$MES.".".$YEAR;

$SQL = "SELECT MAX(NVL(CE27,0)) CE27 FROM $SH.MODIFICATION_REMAINS WHERE
TO_CHAR(DATE_GENERAL,'YYYYMM')=$YEAR||SUBSTR('00'||$MES,-2)";
$RES = @OCIPARSE($CONN,$SQL);
IF (!@OCIEXECUTE($RES)) {OUTPUT(0,'ОШИБКА В РЕЗУЛЬТАТЕ ВЫПОЛНЕНИЯ
ЗАПРОСА!',$RES);}
$NROWS = OCIFETCHSTATEMENT($RES,$ROWS);
IF ($ROWS['CE27'][0]==1) {OUTPUT(2,'ИНВЕНТАРИЗАЦИЯ УЖЕ ЗАКОНЧИЛАСЬ!.');}

$SQL = "SELECT * FROM $SH.MODIFICATION_REMAINS WHERE
TO_CHAR(DATE_GENERAL,'YYYYMM')=$YEAR||SUBSTR('00'||$MES,-2) AND
NUM_ROW_INT=$NUM_ROW_INT AND MOL=$MOL AND BS=$BS AND MBP=$MBP AND
ID_PART=$ID_PART";
```

```
$RES = @OCIPARSE($CONN,$SQL);
IF (!@OCIEXECUTE($RES)) {OUTPUT(0,'ОШИБКА В РЕЗУЛЬТАТЕ ВЫПОЛНЕНИЯ
ЗАПРОСА!',$RES);}
$NROWS = OCIFETCHSTATEMENT($RES,$ROWS);

IF ($NROWS>0){
      $SQL = "UPDATE $SH.MODIFICATION_REMAINS SET KOL=$KOL, H701=$C_FACT,
X367=TRUNC(SYSDATE), TABEL=$SH.GET_USER()
                  WHERE
TO_CHAR(DATE_GENERAL,'YYYYMM')=$YEAR||SUBSTR('00'||$MES,-2) AND MOL=$MOL AND
NUM_ROW_INT=$NUM_ROW_INT AND BS=$BS AND MBP=$MBP AND ID_PART=$ID_PART";
}
ELSE{
      $SQL = "INSERT INTO
$SH.MODIFICATION_REMAINS(ID_PART,DATE_GENERAL,MOL,PAK,NUM_ROW_INT,KVO,BS,MBP,
KOL,H701,TABEL,X367,CE27)
                  VAL-
UES($ID_PART,TO_DATE('$DATE_GENERAL','DD.MM.YYYY'),$MOL,1,$NUM_ROW_INT,94,$BS
,$MBP,$KOL,$C_FACT,$SH.GET_USER(),TRUNC(SYSDATE),0)";
}
$RES = @OCIPARSE($CONN,$SQL);
IF (!@OCIEXECUTE($RES)) {OUTPUT(0,'ОШИБКА В РЕЗУЛЬТАТЕ ВЫПОЛНЕНИЯ
ЗАПРОСА!',$RES);}

$JSON = NEW SERVICES_JSON();
$OUTPUT = $JSON->ENCODE(ARRAY('SUCCESS'=>1));
ECHO($OUTPUT);
?>
```

РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ

Приложение В

(рекомендуемое)

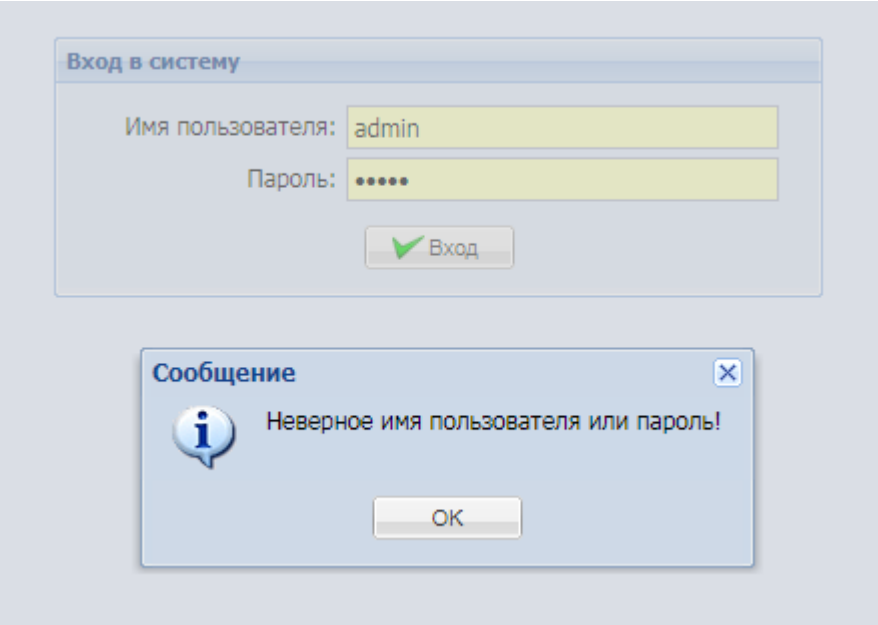


Рисунок В.1 – Результат выполнения теста №2

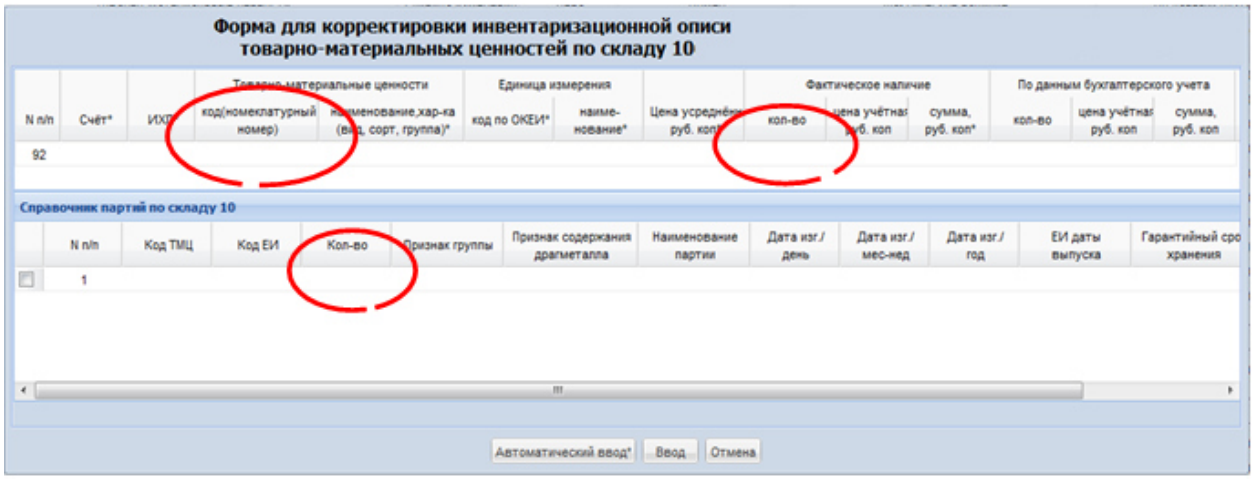


Рисунок В.2 – Результат выполнения теста №3

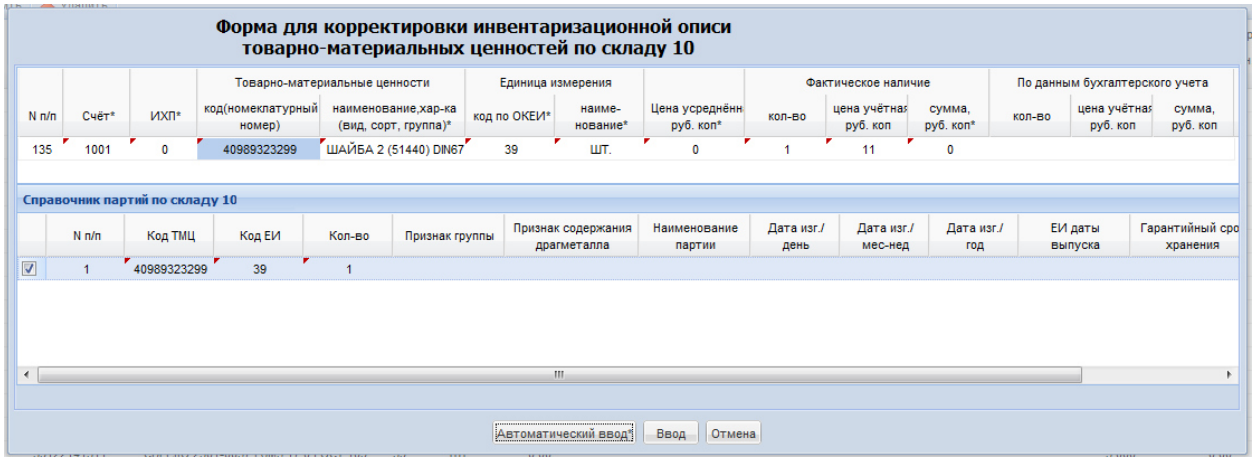
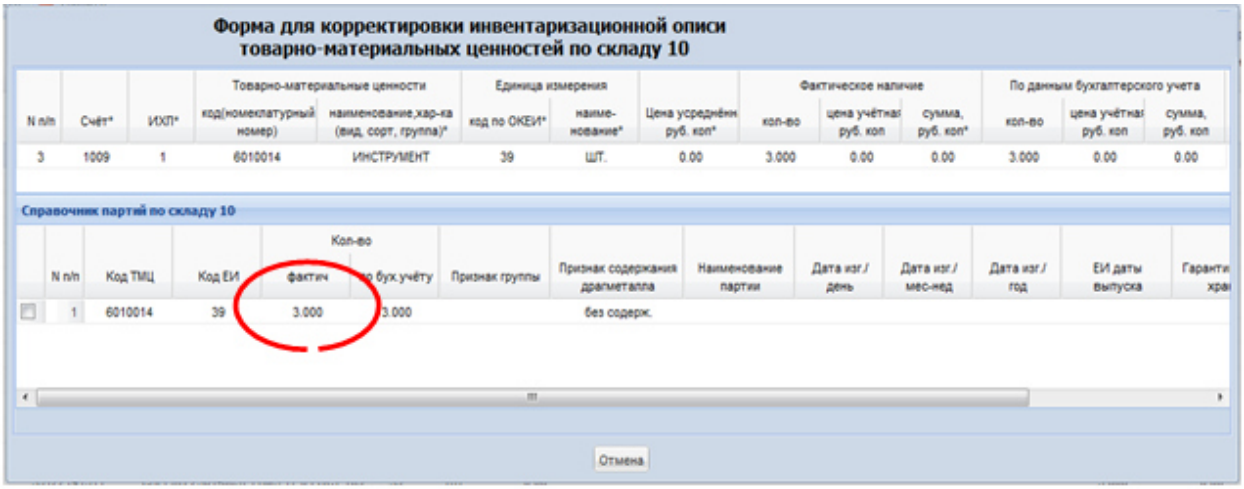


Рисунок В.3 – Результат выполнения теста №4



Форма для корректировки инвентаризационной описи товарно-материальных ценностей по складу 10

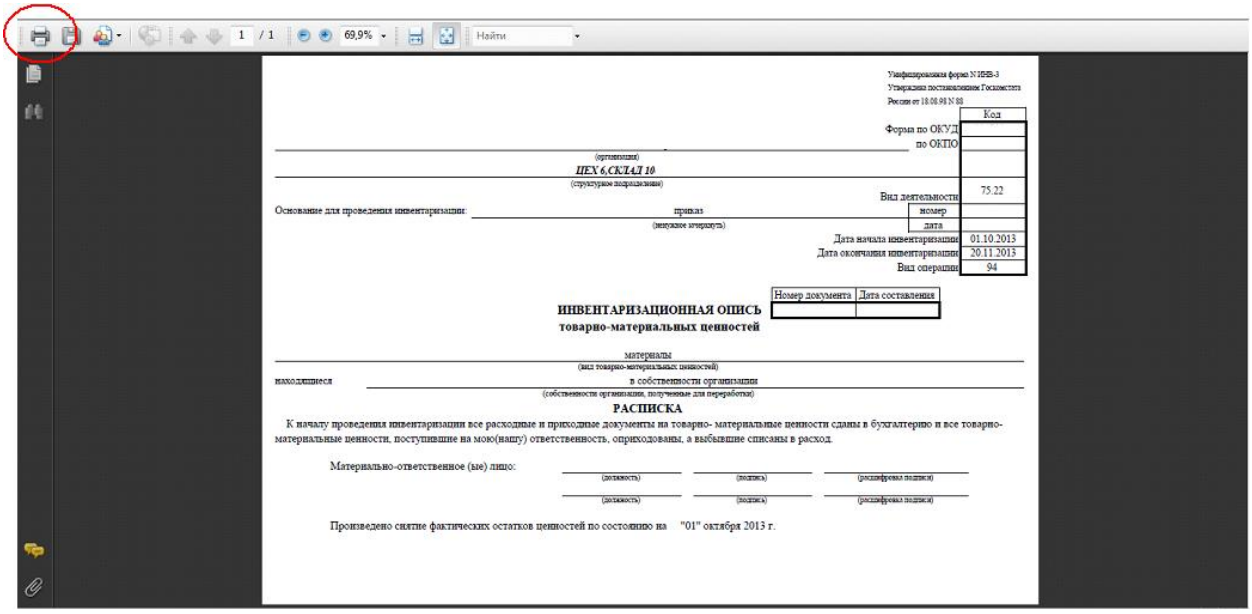
N п/п	Счёт*	ИХП*	Товарно-материальные ценности: код(номенклатурный номер)	Товарно-материальные ценности: наименование,хар-ка (вид, сорт, группа)*	Единица измерения: код по ОКЕИ*	Единица измерения: наиме-нование*	Цена усреднённ руб. коп*	Фактическое наличие: кол-во	Фактическое наличие: цена учётная руб. коп	Фактическое наличие: сумма, руб. коп*	По данным бухгалтерского учета: кол-во	По данным бухгалтерского учета: цена учётная руб. коп	По данным бухгалтерского учета: сумма, руб. коп
3	1009	1	6010014	ИНСТРУМЕНТ	39	ШТ.	0.00	3.000	0.00	0.00	3.000	0.00	0.00

Справочник партий по складу 10

	N п/п	Код ТМЦ	Код ЕИ	Кол-во: фактич	Кол-во: по бух.учёту	Признак группы	Признак содержания драгметалла	Наименование партии	Дата изг./ день	Дата изг./ мес-нед	Дата изг./ год	ЕИ даты выпуска	Гаранти хран
☐	1	6010014	39	3.000	3.000		без содерж.						

Отмена

Рисунок В.4 – Результат выполнения теста №5



Унифицированная форма N ИНВ-3
Утверждена постановлением Госкомстата России от 18.08.98 N 88

Форма по ОКУД
по ОКПО

ЦЕХ 6,СКЛАД 10

Вид деятельности: 75.22

Основание для проведения инвентаризации: приказ номер, дата

Дата начала инвентаризации: 01.10.2013
Дата окончания инвентаризации: 20.11.2013
Вид операции: 94

ИНВЕНТАРИЗАЦИОННАЯ ОПИСЬ товарно-материальных ценностей

Номер документа | Дата составления

материалы (вид товарно-материальных ценностей)
находящиеся в собственности организации (собственности организации, полученные для переработки)

РАСПИСКА

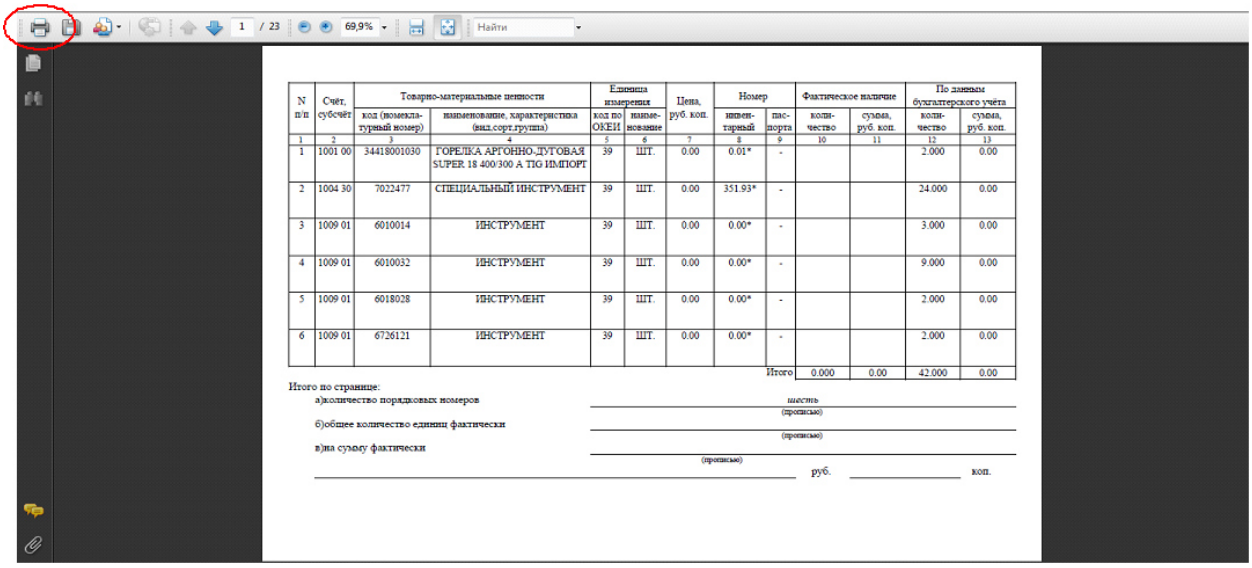
К началу проведения инвентаризации все расходные и приходные документы на товарно- материальные ценности сданы в бухгалтерию и все товарно-материальные ценности, поступившие на мою(нашу) ответственность, оприходованы, а выбывшие списаны в расход.

Материально-ответственное (ые) лицо: (должность) (подпись) (расшифровка подписи)

(должность) (подпись) (расшифровка подписи)

Произведено снятие фактических остатков ценностей по состоянию на "01" октября 2013 г.

Рисунок В.5 – Результат выполнения теста №8



N п/п	Счёт, субсчёт	Товарно-материальные ценности: код (номекла-турный номер)	Товарно-материальные ценности: наименование, характеристика (вид,сорт,группа)	Единица измерения: код по ОКЕИ	Единица измерения: наиме-нование	Цена, руб. коп.	Номер: инвен-тарный	Номер: пас-порта	Фактическое наличие: коли-чество	Фактическое наличие: сумма, руб. коп.	По данным бухгалтерского учёта: коли-чество	По данным бухгалтерского учёта: сумма, руб. коп.
1	2	3	4	5	6	7	8	9	10	11	12	13
1	1001 00	34418001030	ГОРЕЛКА АРГОННО-ДУГОВАЯ SUPER 18 400/300 A TIG ИМПОРТ	39	ШТ.	0.00	0.01*	-			2.000	0.00
2	1004 30	7022477	СПЕЦИАЛЬНЫЙ ИНСТРУМЕНТ	39	ШТ.	0.00	351.93*	-			24.000	0.00
3	1009 01	6010014	ИНСТРУМЕНТ	39	ШТ.	0.00	0.00*	-			3.000	0.00
4	1009 01	6010032	ИНСТРУМЕНТ	39	ШТ.	0.00	0.00*	-			9.000	0.00
5	1009 01	6018028	ИНСТРУМЕНТ	39	ШТ.	0.00	0.00*	-			2.000	0.00
6	1009 01	6726121	ИНСТРУМЕНТ	39	ШТ.	0.00	0.00*	-			2.000	0.00
								Итого	0.000	0.00	42.000	0.00

Итого по странице:
а)количество порядковых номеров: шесть (прописью)
б)общее количество единиц фактически (прописью)
в)на сумму фактически (прописью)
руб. коп.

Рисунок В.6 – Результат выполнения теста №8

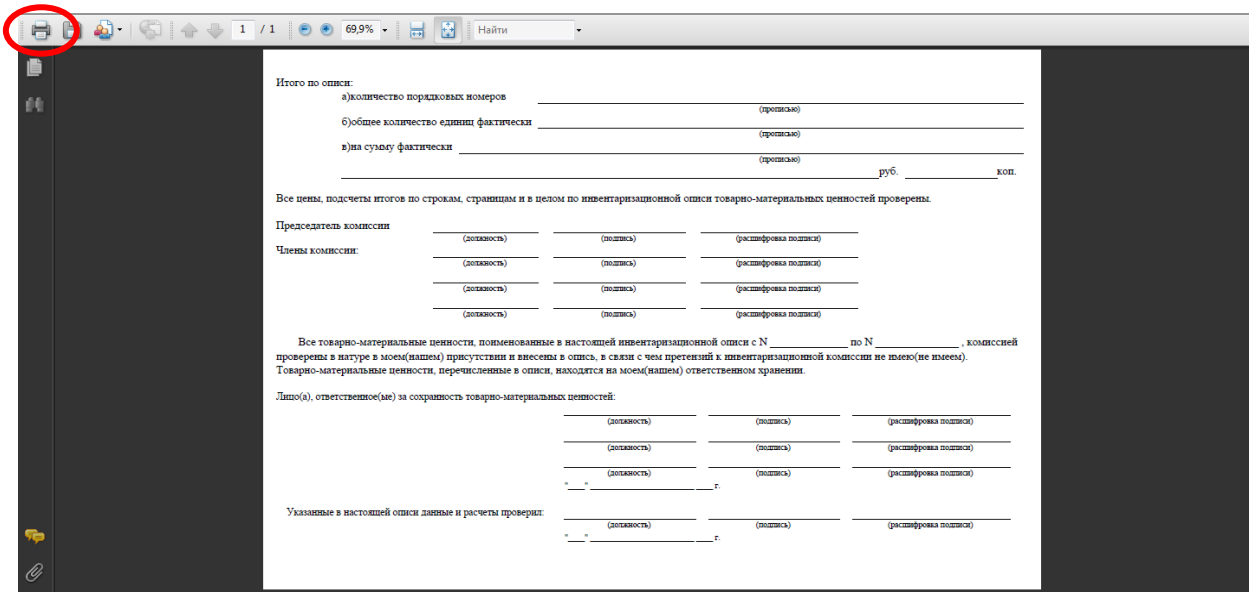


Рисунок В.7 – Результат выполнения теста №8

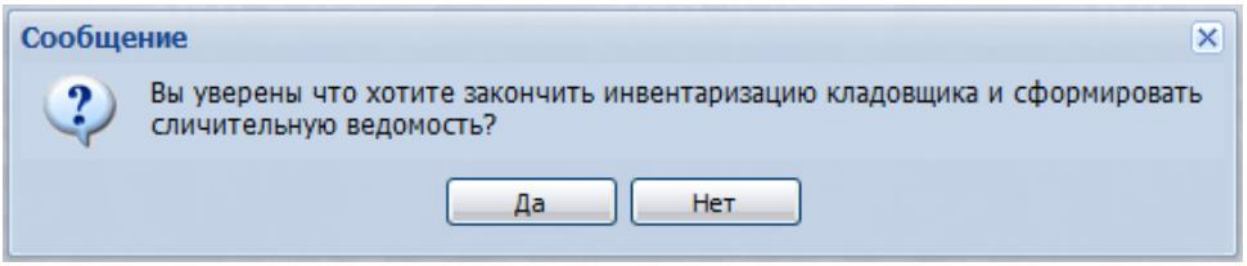
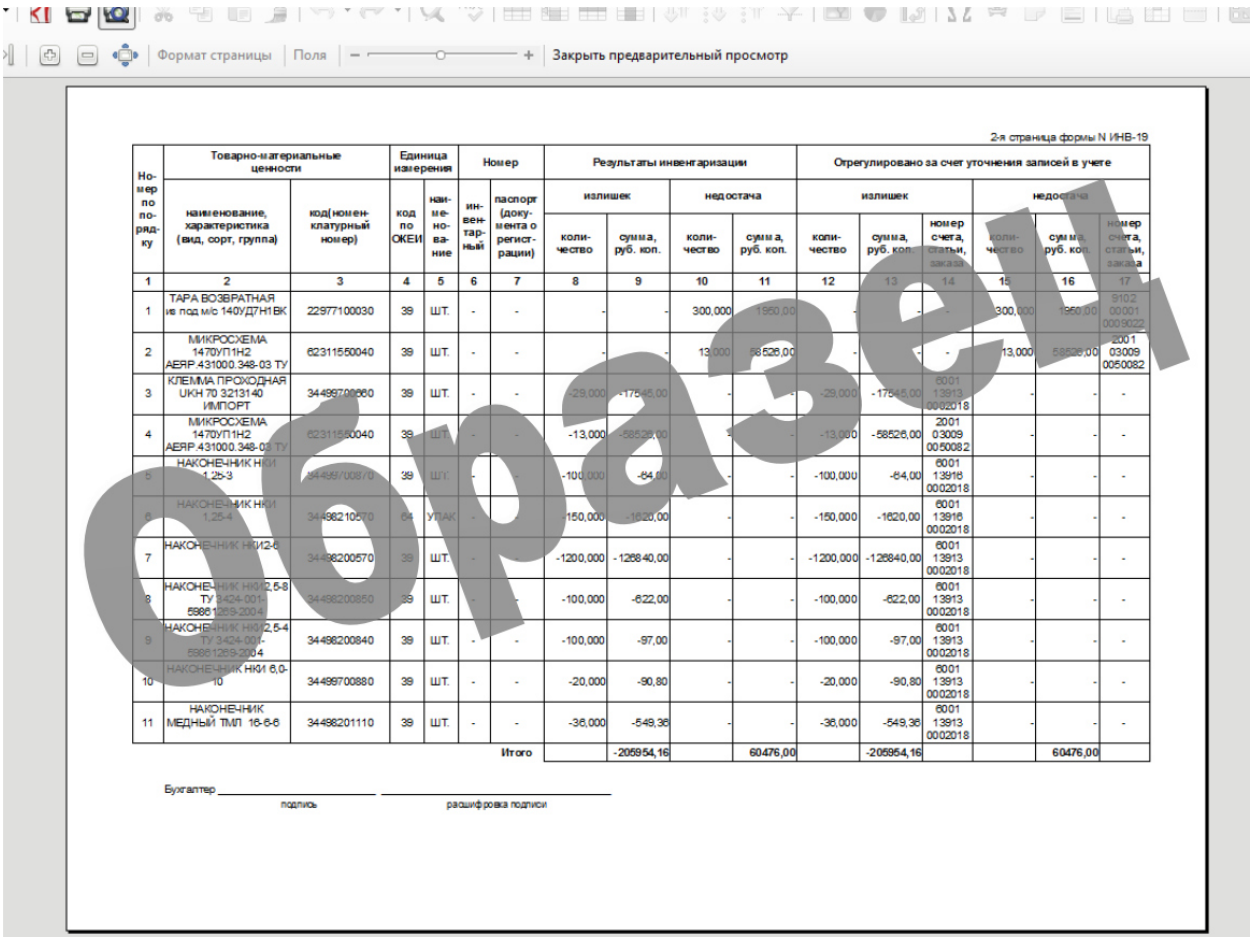


Рисунок В.8 – Результат выполнения теста №10



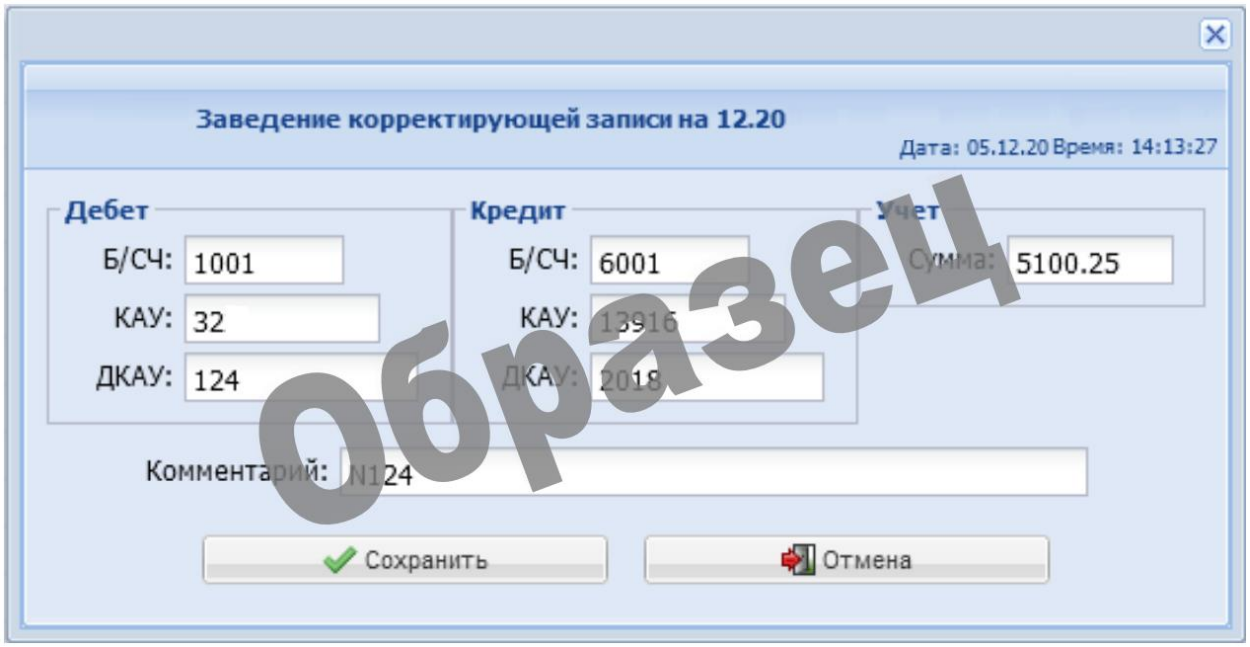
2-я страница формы N ИНВ-19

Номер по порядку	Товарно-материальные ценности: наименование, характеристика (вид, сорт, группа)	Товарно-материальные ценности: код (номенклатурный номер)	Единица измерения: код по ОКЕИ	Единица измерения: наименование	Номер: инвентарный	Номер: паспорт (документа о регистрации)	Результаты инвентаризации: излишек: количество	Результаты инвентаризации: излишек: сумма, руб. коп.	Результаты инвентаризации: недостача: количество	Результаты инвентаризации: недостача: сумма, руб. коп.	Отрегулировано за счет уточнения записей в учете: излишек: количество	Отрегулировано за счет уточнения записей в учете: излишек: сумма, руб. коп.	Отрегулировано за счет уточнения записей в учете: излишек: номер счета, статьи, заказа	Отрегулировано за счет уточнения записей в учете: недостача: количество	Отрегулировано за счет уточнения записей в учете: недостача: сумма, руб. коп.	Отрегулировано за счет уточнения записей в учете: недостача: номер счета, статьи, заказа
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	ТАРА ВОЗВРАТНАЯ на под м/с 140УД7Н1ВК	22977100030	39	ШТ.	-	-	-	-	300,000	1950,00	-	-	-	300,000	1950,00	9102 00001 0009022
2	МИКРОСХЕМА 1470УП1Н2 АЕЯР.431000.348-03 ТУ	62311550040	39	ШТ.	-	-	-	-	13,000	58526,00	-	-	-	13,000	58526,00	2001 03009 0050082
3	КЛЕММА ПРОХОДНАЯ UKH 70 3213140 ИМПОРТ	34499700660	39	ШТ.	-	-	29,000	17545,00	-	-	29,000	17545,00	6001 13913 0002018	-	-	-
4	МИКРОСХЕМА 1470УП1Н2 АЕЯР.431000.348-03 ТУ	62311550040	39	ШТ.	-	-	13,000	58526,00	-	-	13,000	58526,00	2001 03009 0050082	-	-	-
5	НАКОНЕЧНИК НКИ 1,2Б-3	34499700870	39	ШТ.	-	-	100,000	64,00	-	-	100,000	64,00	6001 13916 0002018	-	-	-
6	НАКОНЕЧНИК НКИ 1,25-4	34498210570	64	УПАК	-	-	150,000	1620,00	-	-	150,000	1620,00	6001 13916 0002018	-	-	-
7	НАКОНЕЧНИК НКИ2-6	34498200570	39	ШТ.	-	-	1200,000	126840,00	-	-	1200,000	126840,00	6001 13913 0002018	-	-	-
8	НАКОНЕЧНИК НКИ2,5-8 ТУ 3424-001-55661289-2004	34498200850	39	ШТ.	-	-	100,000	622,00	-	-	100,000	622,00	6001 13913 0002018	-	-	-
9	НАКОНЕЧНИК НКИ2,5-4 ТУ 3424-001-55661289-2004	34498200840	39	ШТ.	-	-	100,000	97,00	-	-	100,000	97,00	6001 13913 0002018	-	-	-
10	НАКОНЕЧНИК НКИ 6,0-10	34499700880	39	ШТ.	-	-	20,000	90,80	-	-	20,000	90,80	6001 13913 0002018	-	-	-
11	НАКОНЕЧНИК МЕДНЫЙ ТМП 16-6-6	34498201110	39	ШТ.	-	-	36,000	549,36	-	-	36,000	549,36	6001 13913 0002018	-	-	-
						Итого		205954,16		60476,00		205954,16			60476,00	

Бухгалтер _____ _____
подпись расшифровка подписи

Образец

Рисунок В.9 – Результат выполнения теста №11



Образец

Рисунок В.10 – Результат выполнения теста №12

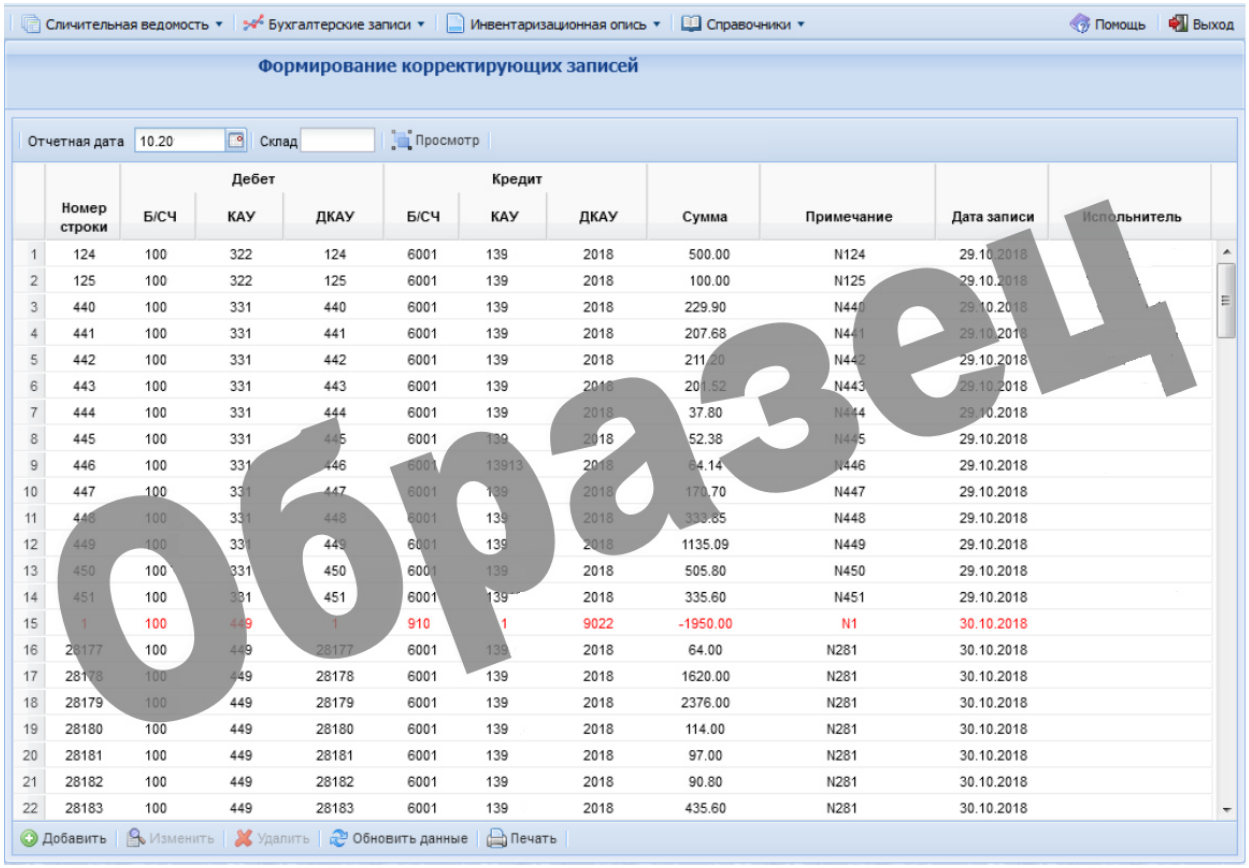


Рисунок В.11 – Результат выполнения теста №13

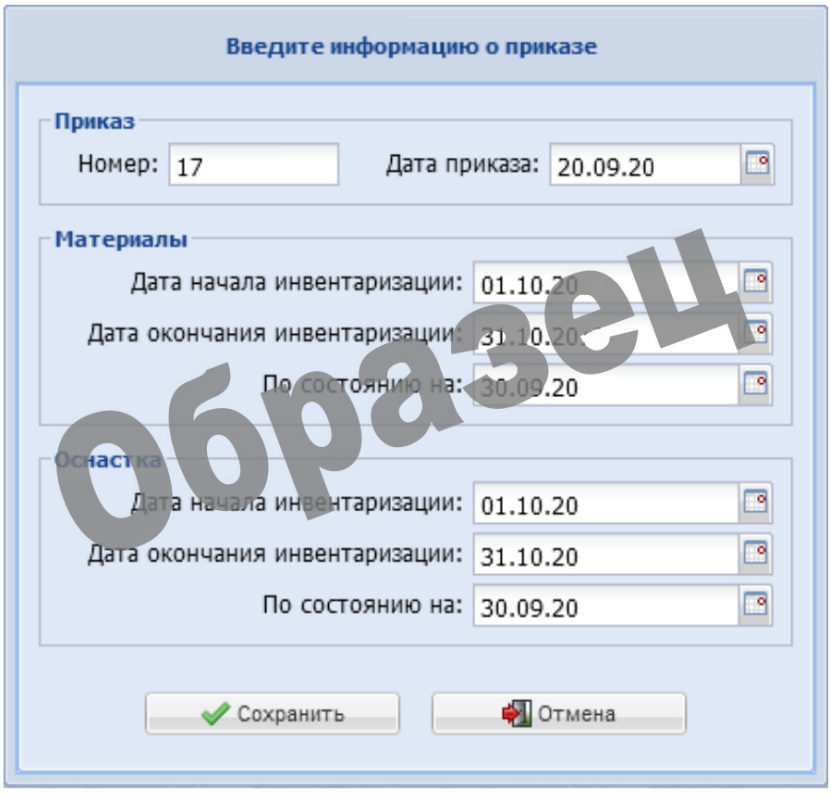


Рисунок В.12 – Результат выполнения теста №14

Образец

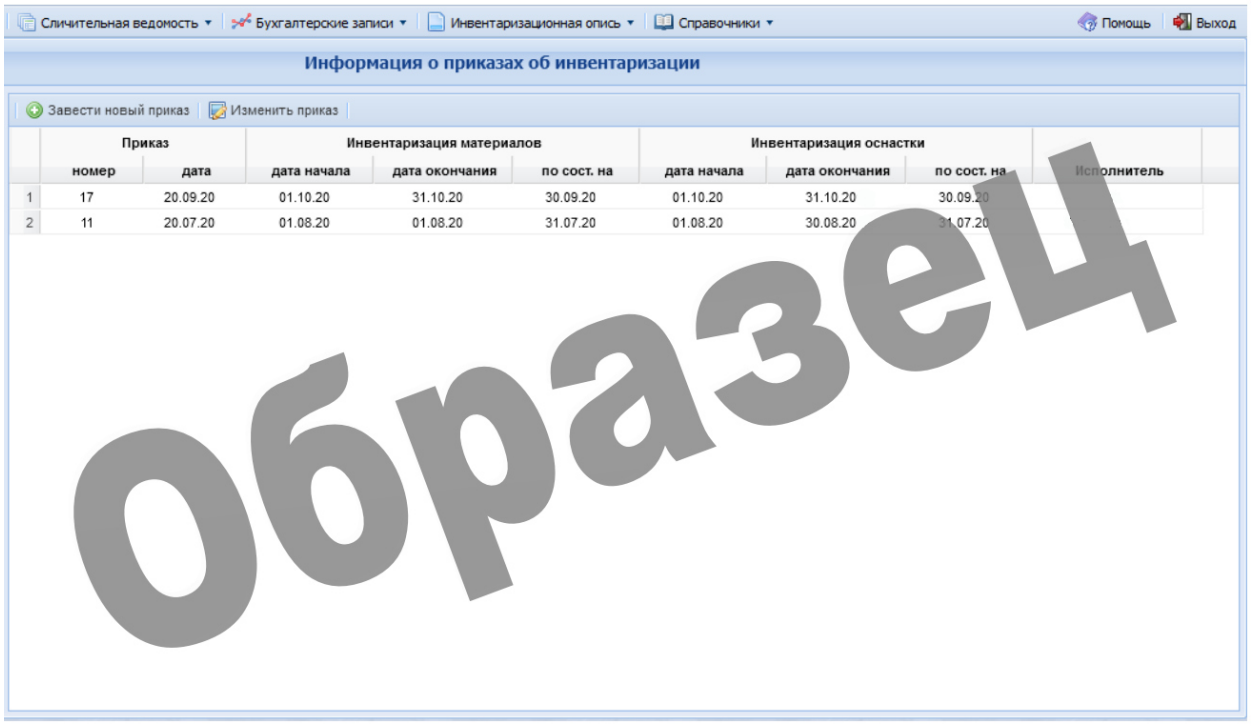


Рисунок В.13 – Результат выполнения теста №15

ЛИСТИНГ СКРИПТА DDL

Приложение Г

(рекомендуемое)

Листинг Г.1 – Скрипт создания таблиц базы данных на языке SQL

```
CREATE TABLE Access
(
      id_access            INTEGER AUTO_INCREMENT,
      login                VARCHAR(150) NULL,
      password             VARCHAR(150) NULL
);

ALTER TABLE Access
ADD PRIMARY KEY (id_access);


CREATE TABLE Category
(
      id_category          INTEGER AUTO_INCREMENT,
      iss                  INTEGER NULL,
      name                 VARCHAR(150) NULL
);

ALTER TABLE Category
ADD PRIMARY KEY (id_category);

CREATE TABLE Client
(
      id_client            INTEGER AUTO_INCREMENT,
      family               VARCHAR(150) NULL,
      first_name           VARCHAR(150) NULL,
      last_name            VARCHAR(150) NULL,
      date_born            DATE NULL,
      gender               VARCHAR(20) NULL,
      phone                INTEGER NULL,
      address              VARCHAR(150) NULL
);

ALTER TABLE Client
ADD PRIMARY KEY (id_client);

CREATE TABLE Passport
(
      id_passport          INTEGER AUTO_INCREMENT,
      number_passport      INTEGER NULL,
      date_issuing         DATE NULL,
      department           VARCHAR(150) NULL,
      id_trainer           INTEGER NULL
);

ALTER TABLE Passport
ADD PRIMARY KEY (id_passport);

CREATE TABLE Record
(
      id_record            INTEGER AUTO_INCREMENT,
      date_start           DATE NULL,
      date_end             DATE NULL,
```

```
      id_client            INTEGER NULL,
      id_trainer           INTEGER NULL,
      id_ticket            INTEGER NULL
);

ALTER TABLE Record
ADD PRIMARY KEY (id_record);

CREATE TABLE Specialization
(
      id_specialization    INTEGER AUTO_INCREMENT,
      name                 VARCHAR(150) NULL
);

ALTER TABLE Specialization
ADD PRIMARY KEY (id_specialization);

CREATE TABLE Ticket
(
      id_ticket            INTEGER AUTO_INCREMENT,
      name                 VARCHAR(150) NULL,
      price                NUMERIC(10,2) NULL
);

ALTER TABLE Ticket
ADD PRIMARY KEY (id_ticket);

CREATE TABLE Trainer
(
      id_trainer           INTEGER AUTO_INCREMENT,
      family               VARCHAR(150) NULL,
      first_name           VARCHAR(150) NULL,
      last_name            VARCHAR(150) NULL,
      date_born            DATE NULL,
      gender               VARCHAR(20) NULL,
      phone                INTEGER NULL,
      address              VARCHAR(150) NULL,
      total_experience     INTEGER NULL,
      experience           INTEGER NULL,
      id_specialization    INTEGER NULL,
      id_category          INTEGER NULL
);

ALTER TABLE Trainer
ADD PRIMARY KEY (id_trainer);

ALTER TABLE Passport
ADD FOREIGN KEY R_11 (id_trainer) REFERENCES Trainer (id_trainer);

ALTER TABLE Record
ADD FOREIGN KEY R_8 (id_client) REFERENCES Client (id_client);

ALTER TABLE Record
ADD FOREIGN KEY R_9 (id_trainer) REFERENCES Trainer (id_trainer);

ALTER TABLE Record
```

```
ADD FOREIGN KEY R_10 (id_ticket) REFERENCES Ticket (id_ticket);

ALTER TABLE Trainer
ADD FOREIGN KEY R_1 (id_specialization) REFERENCES Specialization
(id_specialization);

ALTER TABLE Trainer
ADD FOREIGN KEY R_2 (id_category) REFERENCES Category (id_category);
```

ПРЕЗЕНТАЦИЯ

Приложение Д

(рекомендуемое)

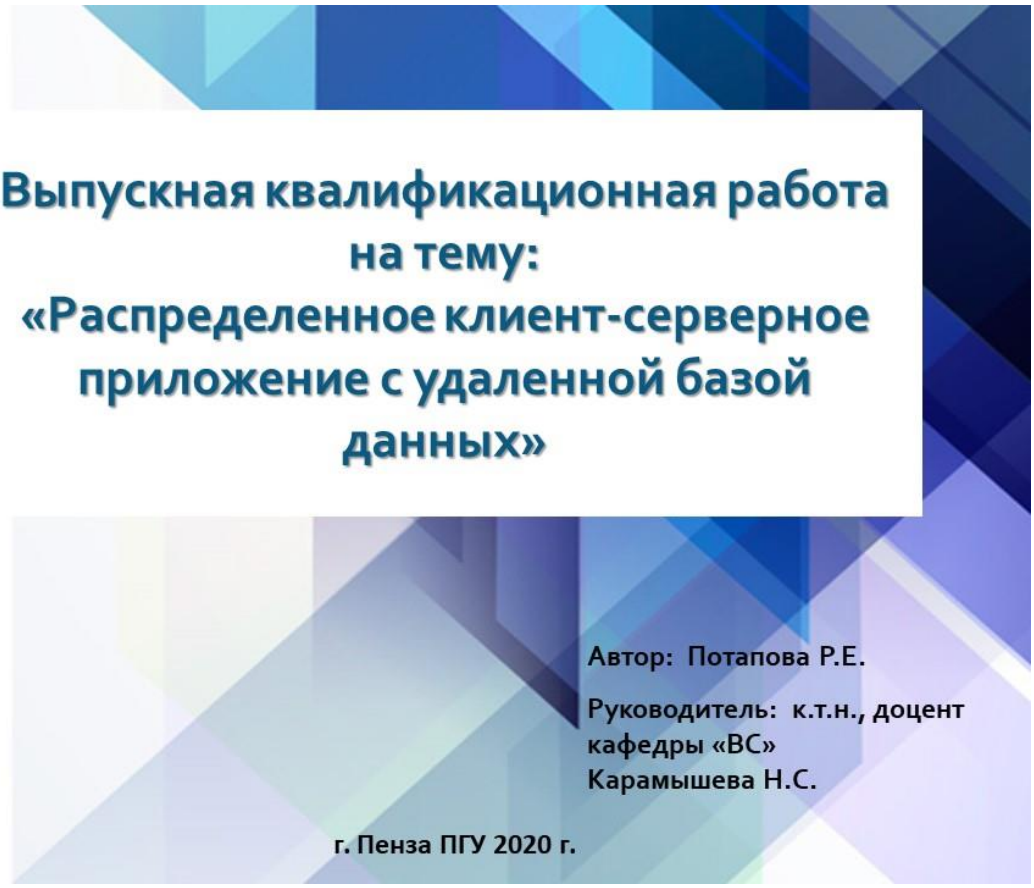


Рисунок Д.1 – Слайд презентации №1

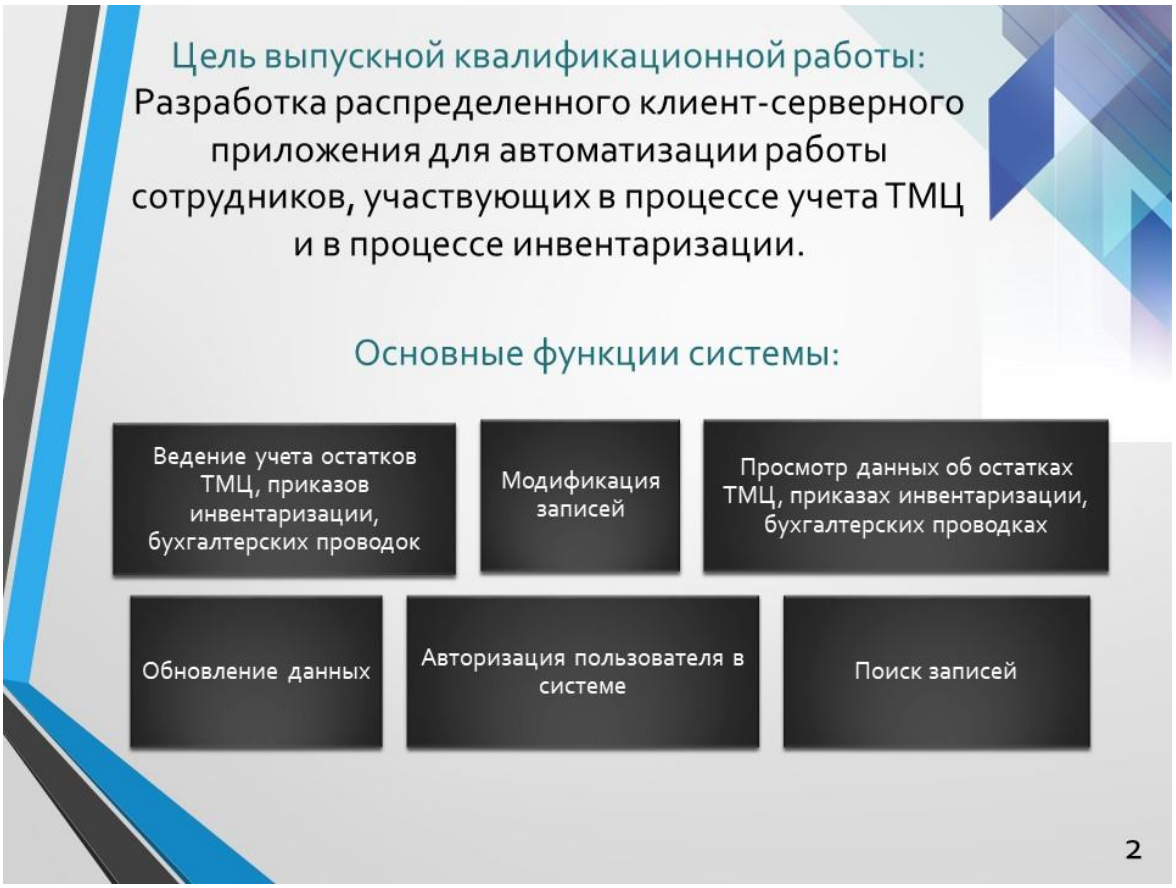


Рисунок Д.2 – Слайд презентации №2

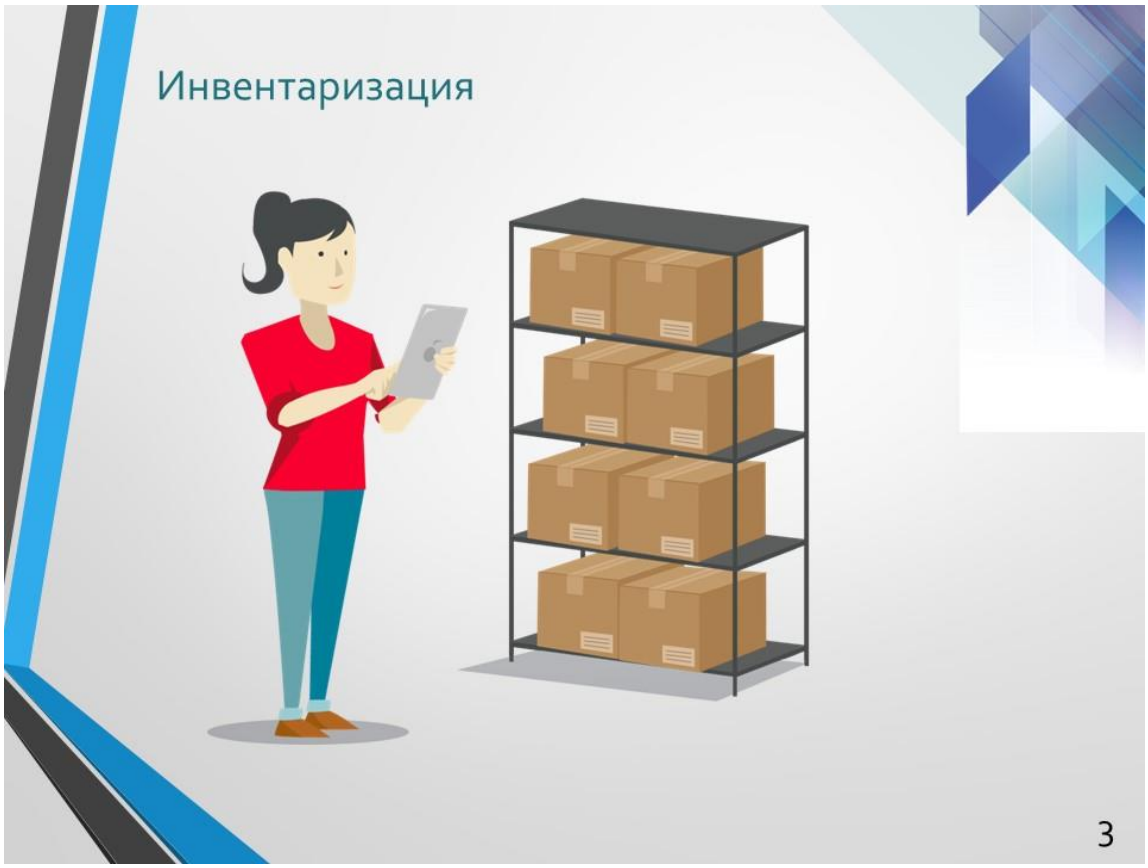


Рисунок Д.3 – Слайд презентации №3

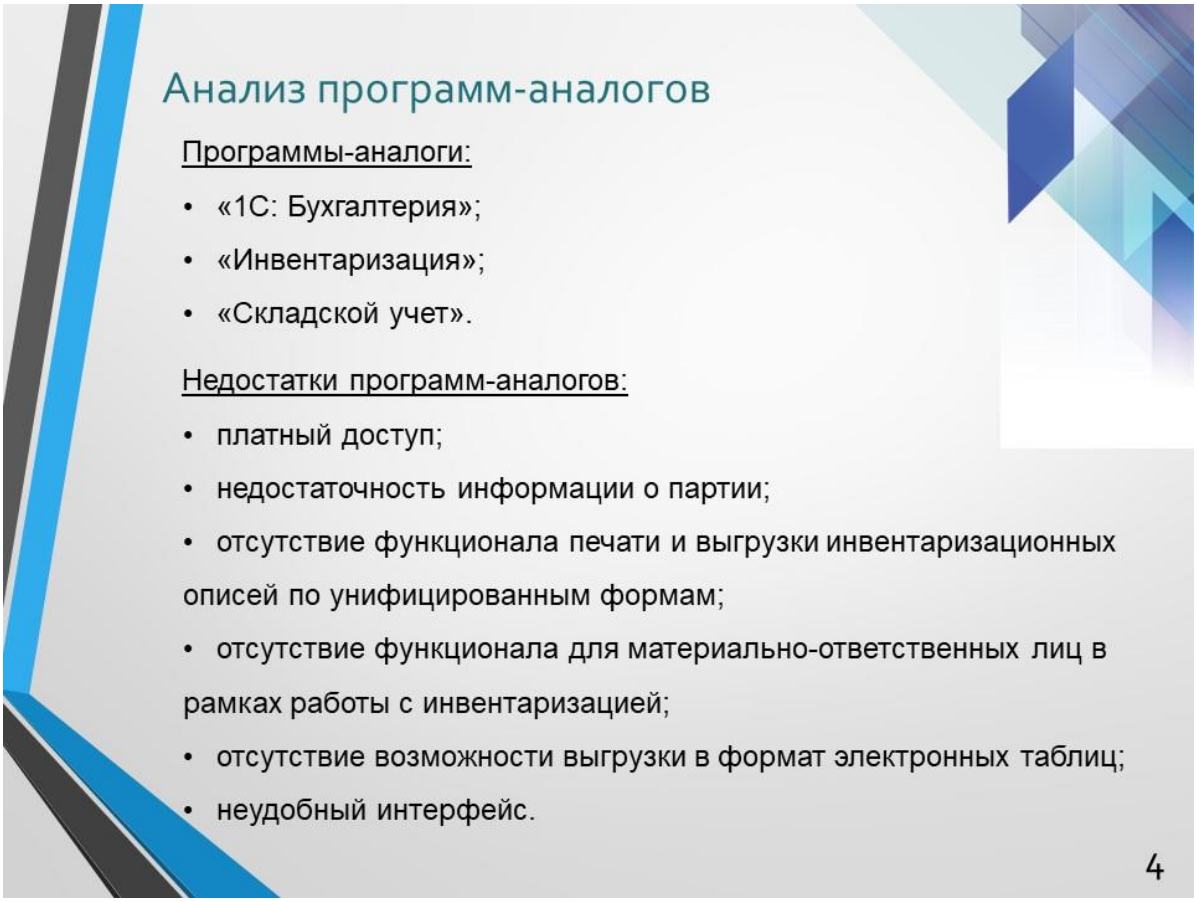


Рисунок Д.4 – Слайд презентации №4

Рисунок Д.5 – Слайд презентации №5

Рисунок Д.6 – Слайд презентации №6

Рисунок Д.7 – Слайд презентации №7

Рисунок Д.8 – Слайд презентации №8

Рисунок Д.9 – Слайд презентации №9

Рисунок Д.10 – Слайд презентации №10

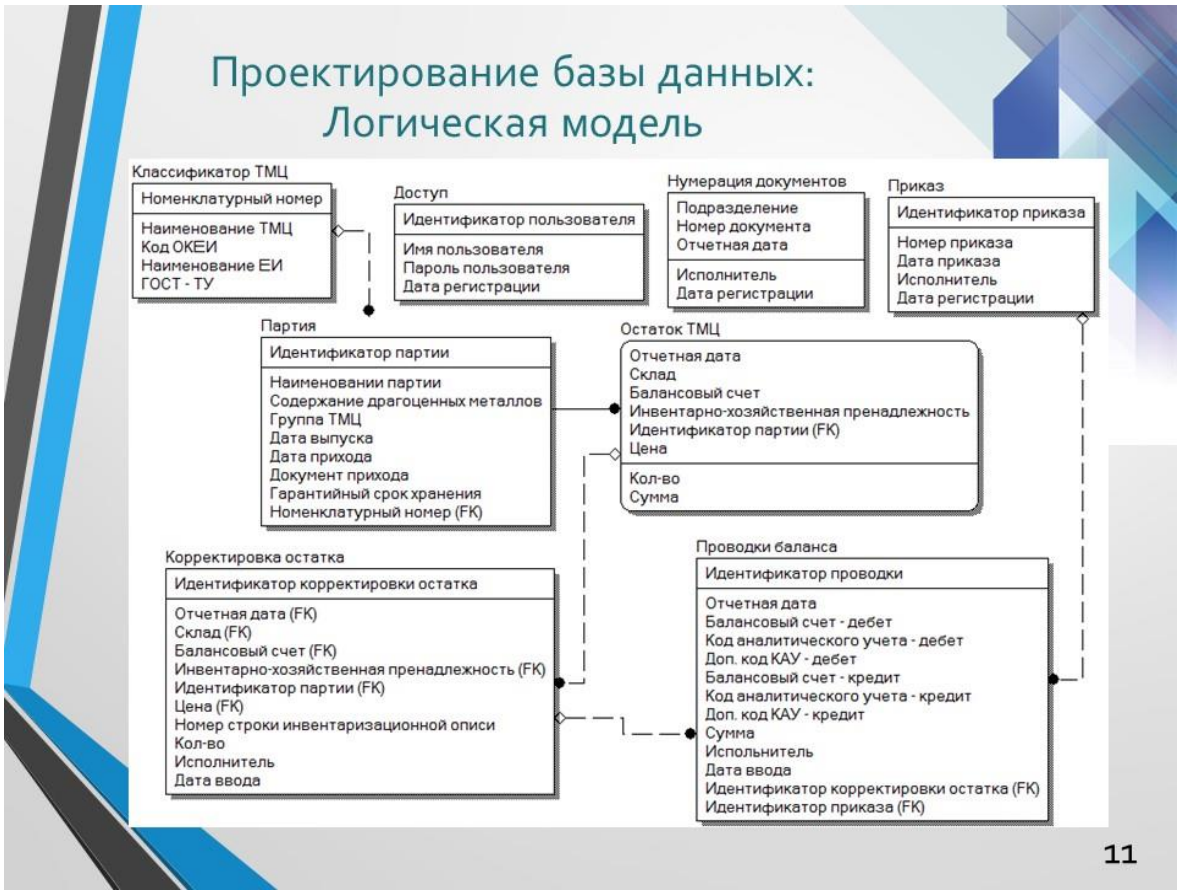


Рисунок Д.11 – Слайд презентации №11

Рисунок Д.12 – Слайд презентации №12

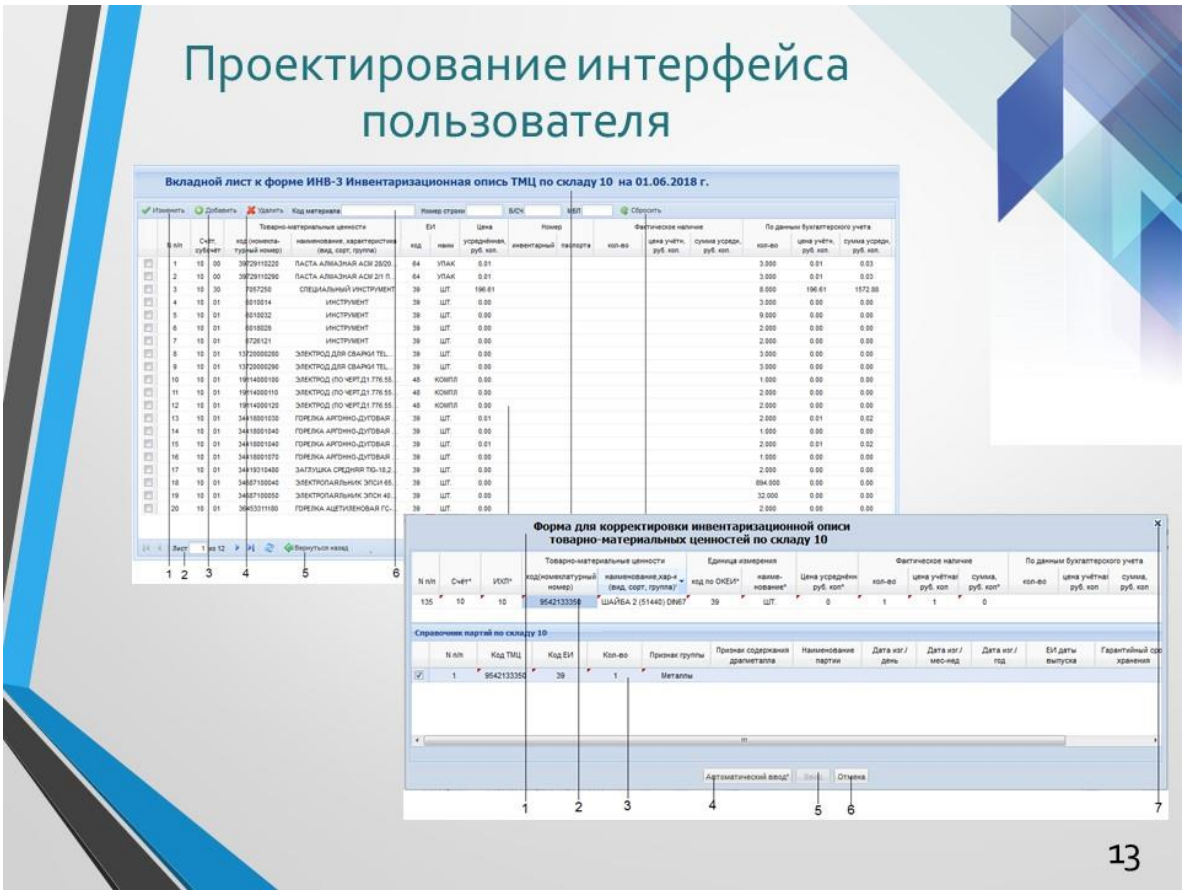


Рисунок Д.13 – Слайд презентации №13

Рисунок Д.14 – Слайд презентации №14

Рисунок Д.15 – Слайд презентации №15

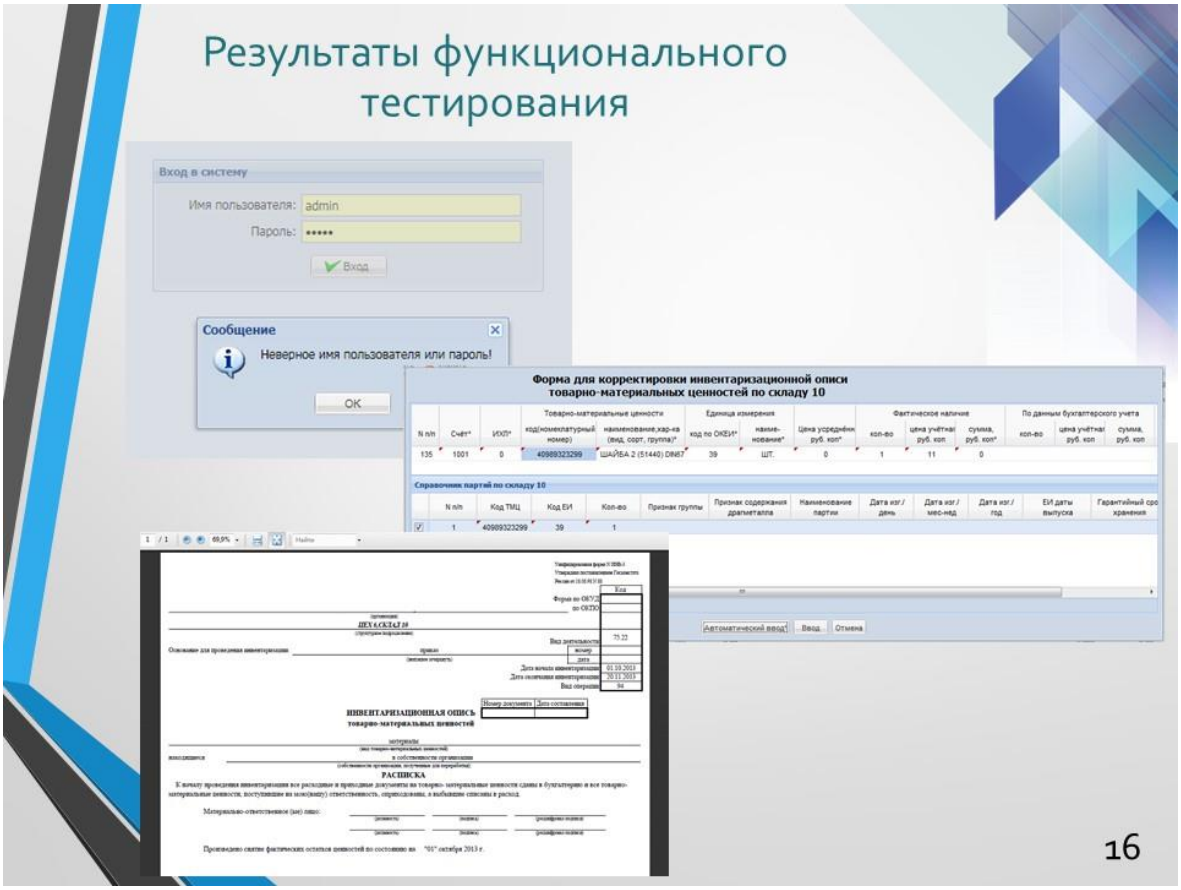


Рисунок Д.16 – Слайд презентации №16

Рисунок Д.17 – Слайд презентации №17

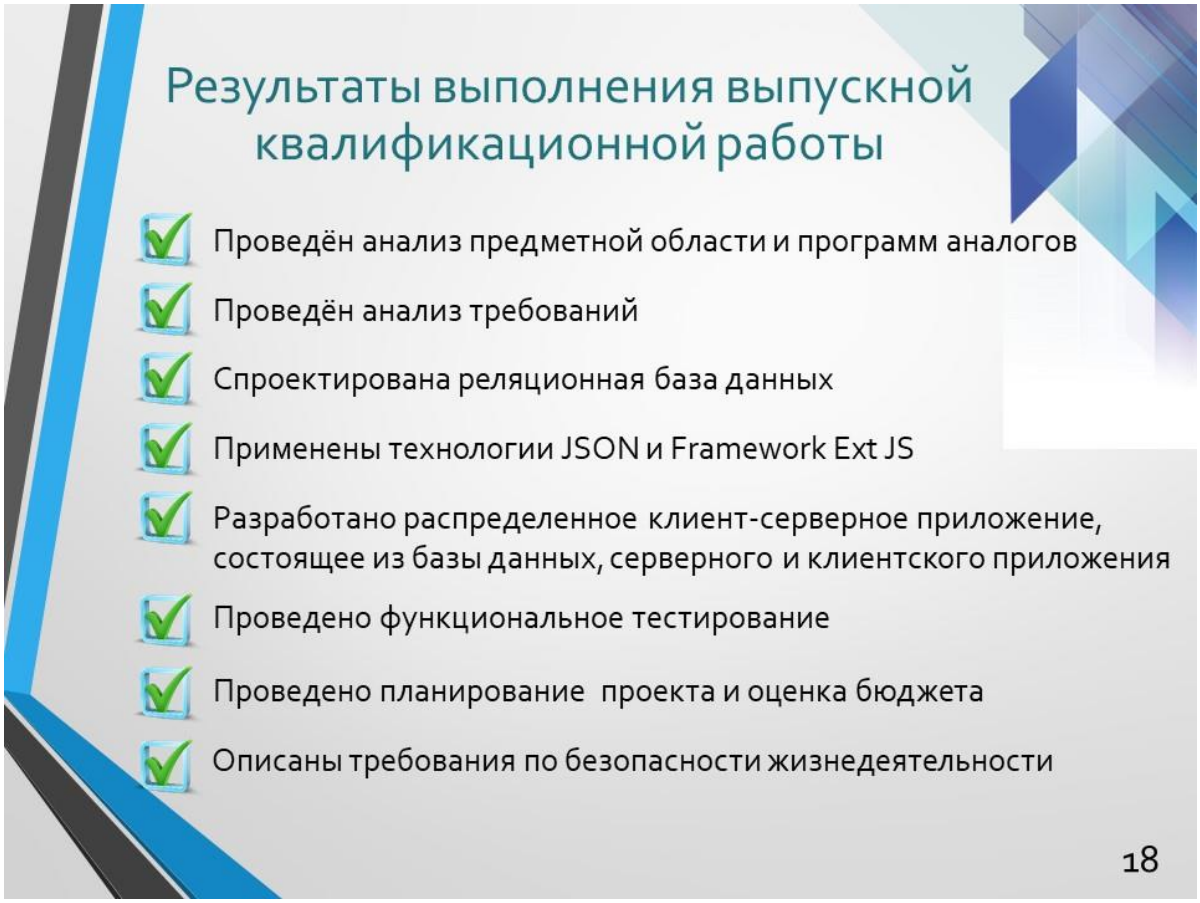


Рисунок Д.18 – Слайд презентации №18

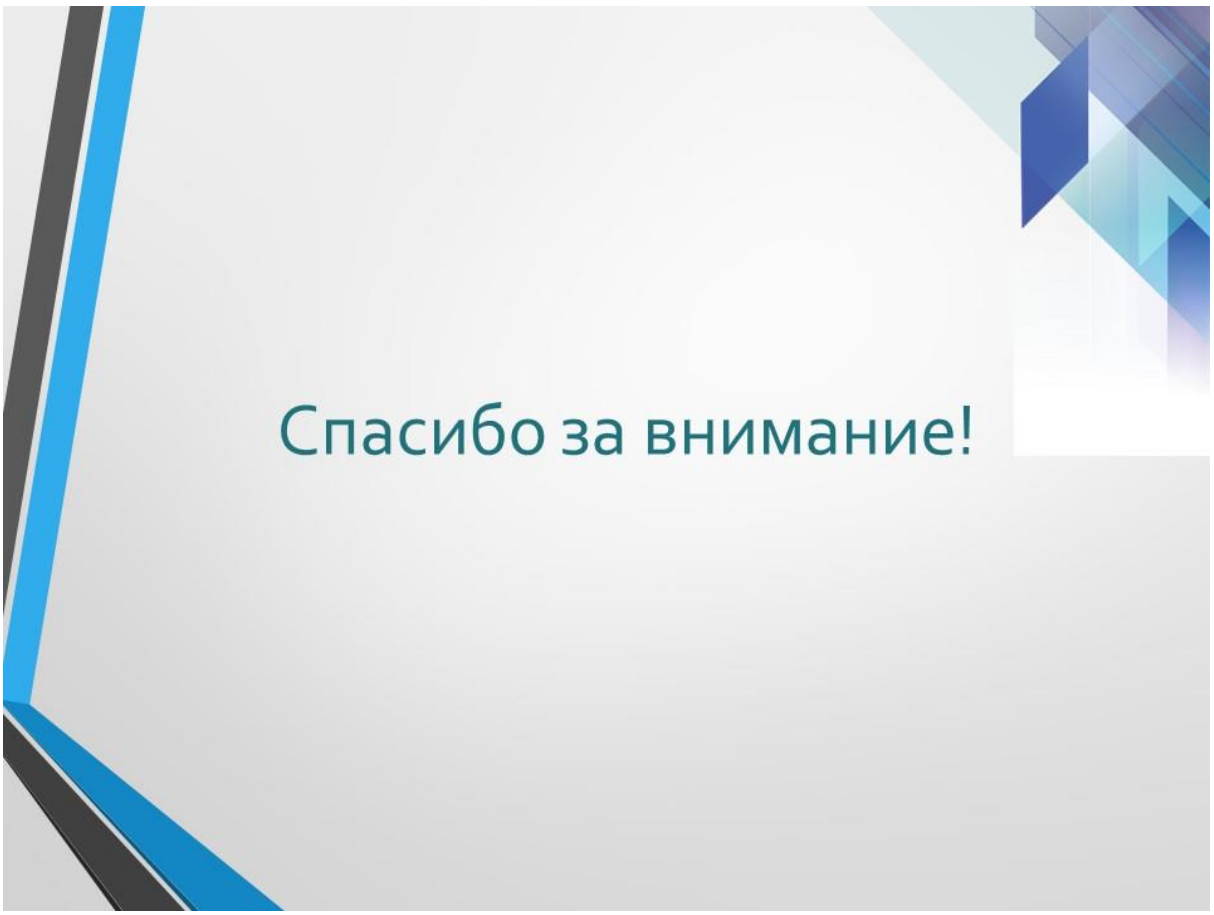


Рисунок Д.19 – Слайд презентации №19