МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
СЕВЕРО-ВОСТОЧНЫЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ЕСТЕСТВЕННЫХ НАУК И МАТЕМАТИКИ

Кафедра информатики

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ТАБАЧНЫЙ СКЛАД»

Допустить к защите:

И.о. зав. кафедрой информатики
к.ф.-м.н.,
Старикова Ольга Александровна

Выпускная квалификационная
работа студента группы ПИз-41
Прокопюка Сергея Валерьевича

Научный руководитель:
КТН, доцент
Сироткин Андрей Вячеславович

Магадан
2018

Оглавление

ВВЕДЕНИЕ....3
ГЛАВА 1. АНАЛИЗ ПРЕДПРИЯТИЯ....5
1.1. Описание предметной области....5
1.2. Описание системы на основе методологии IDEF0....7
1.3. Разработка логической модели базы данных АСУД....12
1.4. Матрица ответственности....19
1.5. Моделирование бизнес-вариантов использования автоматизированной информационной системы «Табачный склад»....19
Глава 2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ....20
2.1. Построение концептуальной модели....20
2.2. Реализация функциональной архитектуры....24
2.3. Диаграммы классов....56
2.4. Выбор СУБД....64
2.5. Построение физической модели базы данных....66
2.6. Архитектура доступа к данным....74
2.7. Интерфейсы....76
ГЛАВА 3. РЕАЛИЗАЦИЯ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ТАБАЧНЫЙ СКЛАД»....77
3.1. Выбор среды разработки....77
3.2. Описание программы....78
3.3. Внедрение разработанной автоматизированной информационной системы «Табачный склад»....94
3.4. Расчет показателей экономической эффективности проекта....96
ЗАКЛЮЧЕНИЕ....108
СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ....109
ПРИЛОЖЕНИЯ....111

ВВЕДЕНИЕ

Современные информационные технологии применяются уже практически во всех сферах деятельности человека. Можно сказать, что в настоящий момент, эффективное развитие любого общества немыслимо без систем управления, построенных на комплексных системах обработки информации, на современных информационных технологиях.

Наибольшая актуальность и практическая потребность в разработке методов применения электронной информации в экономике, наблюдается на уровне различных фирм, так как предприятия выступают как основной субъект хозяйственной деятельности в экономике. Этот факт подтверждается постоянно растущим интересом со стороны менеджеров крупных и средних компаний к внедрению автоматизированных компьютерных систем учёта и поддержки принятия решений.

Внедрённая информационная система позволяет упростить и ускорить большинство процессов за счёт их автоматизации, компьютеризировать вычисления и тем самым избежать некачественной обработки информации, наладить взаимодействие отделов предприятия с помощью разработанных протоколов.

Интерес к автоматизированным системам учёта вызван качественно новыми технологическими возможностями обработки управленческой информации, которые имеют прямое приложение для задач повышения экономической эффективности работы, повышение конкурентоспособности и рентабельности рынка, за счёт использования информационных методов, позволяющих рационализировать управление ресурсами.

Темой данной выпускной квалификационной работы является разработка автоматизированной системы табачного склада. Автоматизация системы данных позволит сэкономить время по внесению данных, их обработке, а также позволит быстро составлять отчеты, получать своевременно необходимую информацию.

Целью данной выпускной квалификационной работы является разработка программного продукта табачного склада, позволяющего автоматизировать обработку и хранения данных по складу.

Объектом исследования деятельность табачного склада.

Предметом исследования является автоматизация деятельности табачного склада АО «ТК Мегаполис».

С учетом изложенного, задачами работы являются:

- анализ и изучение предметной области;
- обработка информации по данной теме;
- моделирование;

- разработка и тестирование программного продукта.

Методы исследования, использованные в выпускной квалификационной работе:

1) Теоретические методы: анализ технической литературы по теме выпускной квалификационной работы; анализ нормативно-правовой базы и др;

2) Общенаучные методы исследования: систематизация, моделирование и др.;

3) Методы статистической обработки результатов, в частности, показателей экономической эффективности проекта.

Практическое значение выпускной квалификационной работы заключается в успешном использовании разработанной автоматизированной информационной системы «Табачный склад» в работе сотрудников отдела склада АО «ТК «Мегаполис».

Последовательное решение сформулированных выше задач определяет структуру выпускной квалификационной работы. Данная работа состоит из введения, трех глав, заключения, списка источников и литературы, и приложений.

Во введении обосновываются: актуальность темы, практическая значимость, цель выпускной квалификационной работы, задачи, объект, предмет, методы исследования.

В первой главе «Анализ предметной области» дается общая характеристика деятельности предприятия АО «ТК «Мегаполис», проводится анализ работы отдела склада, строится логическая модель БД, матрица ответственности, диаграмма вариантов использования.

Во второй главе «Проектирование автоматизированной информационной системы «Табачный склад» рассматривается процесс проектирования АИС, подробно рассматриваются сценарии и последовательности действий вариантов использования. Строятся диаграммы классов и информационная модель базы данных.

В третьей главе «Реализация автоматизированной информационной системы «Табачный склад» приводится обоснование выбора платформы для реализации АИС, описывается процесс программной реализации приложения средствами Microsoft Visual Studio 2015. Рассматривается созданное, в результате выпускной квалификационной работы, программное обеспечение АИС «Табачный склад».

В заключении приводятся итоги исследования, формулируются основные выводы и результаты работы.

Приложения выпускной квалификационной работы содержат: техническое задание на выполнение разработки АИС «Табачный склад».

К ВКР прилагается диск с программным продуктом, акт о внедрении.

ГЛАВА 1. АНАЛИЗ ПРЕДПРИЯТИЯ

1.1. Описание предметной области

Торговая компания «Мегаполис» является международной компанией и основным дистрибьютором табачной продукции. Основной деятельностью компании является реализация табачной продукции.

В данной отрасли более чем актуален вопрос внедрения и применения информационных систем. В компании работает большое количество IT сотрудников, которые разрабатывают и внедряют улучшение программы в ежедневную работу сотрудников.

Высшим органом управления АО «ТК «Мегаполис» является Общее собрание акционеров, в функции которого входит формирование Совета директоров, избрание Ревизионной комиссии, утверждение аудитора, утверждение годового отчета и годовой бухгалтерской отчетности, распределение прибыли, в том числе выплата (объявление) дивидендов (Рис.1.).

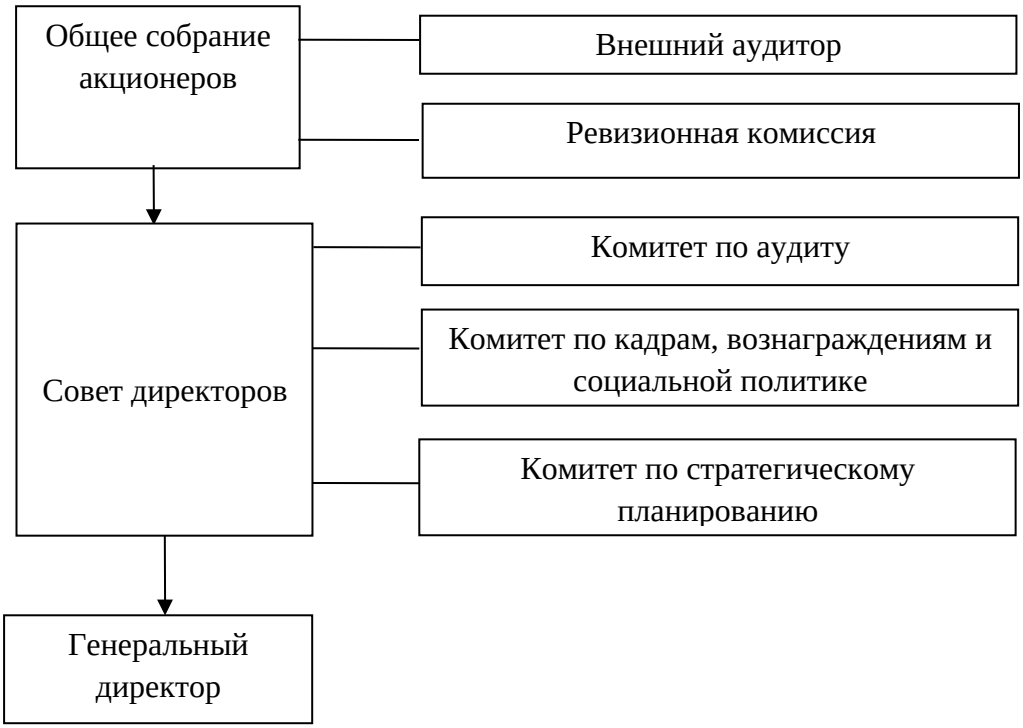


Рис. 1. Общее собрание акционеров

Главным рабочим элементом компании являются несколько подразделений: отдел кадров, отдел логистики, отдел продаж, отдел информационных технологий, юридический отдел, бухгалтерия. Все отделы прочно связаны между собой и постоянно взаимодействуют.

1. Отдел кадров - ведет учет сотрудников, производит подбор персонала, отвечает за оформление документов, согласно Трудовому Кодексу.
2. Отдел логистики – является одним из ключевых звеньев компании. Обеспечивает хранение, доставку товара.
3. Отдел продаж – осуществляет поиск возможных контрагентов для дальнейшего сотрудничества в реализации товара, производит сбор заявок на табачную продукцию.
4. Отдел информационных технологий – обеспечивает бесперебойную работу программ, разрабатывает и внедряет новинки для улучшения информационных систем.
5. Юридический отдел – отвечает за законность рабочих процессов
6. Бухгалтерия – осуществляет формирование бухгалтерских документов, контролирует движение денежных средств компании, отвечает за предоставление отчетов.

1.2. Описание системы на основе методологии IDEF0

Рассматриваемая, в данной выпускной квалификационной работе, компания является дистрибьютором табачного товара. Очень большое внимание уделяется работе отдела склада.

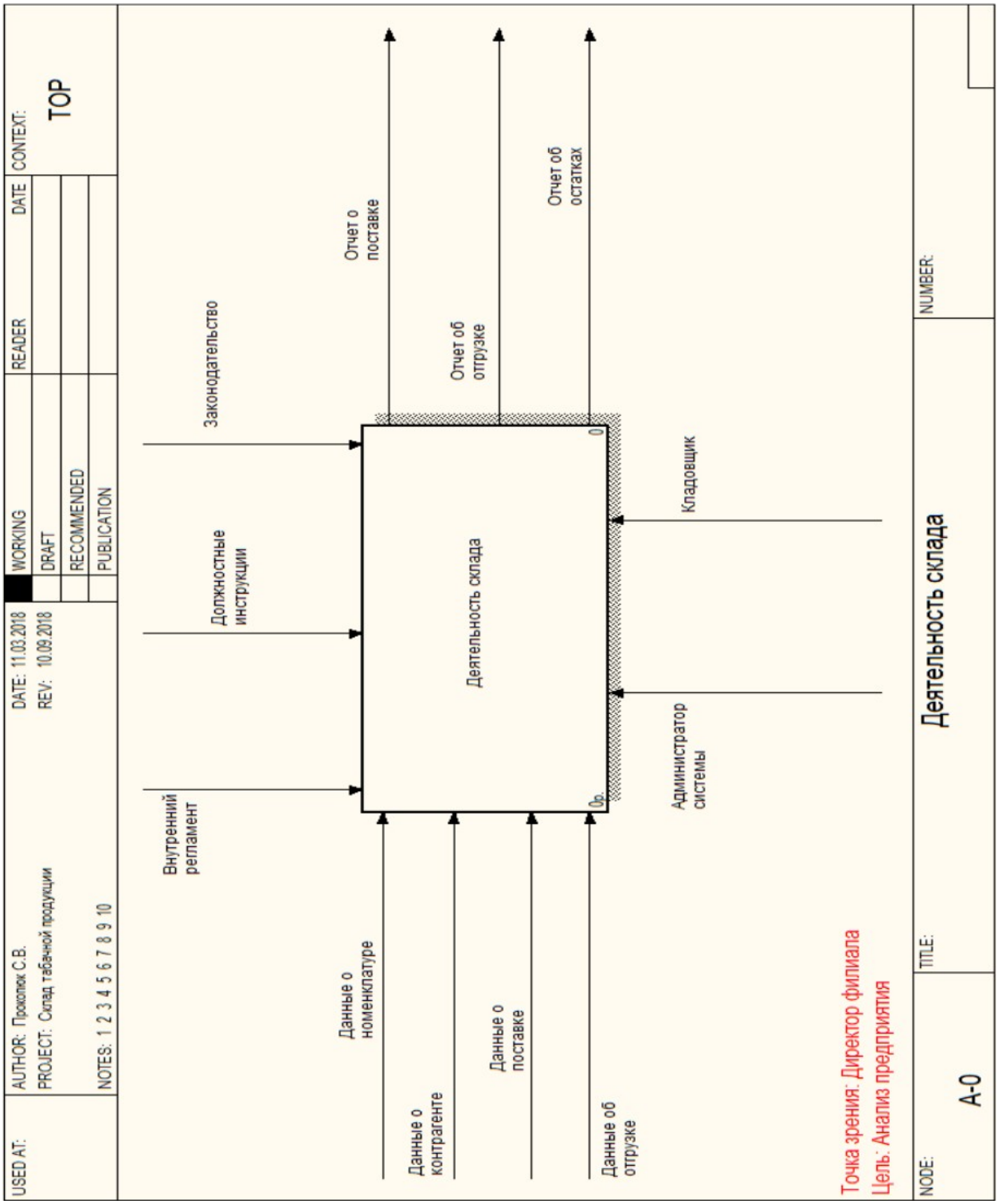


Рис. 2. Деятельность склада

Правильная организация работы склада подразумевает под собой: оформление документов, прием товара, хранение товара и отпуск товара.

Работа склада может складываться в следующих этапах:

При поступлении товара проверяются документы поставки, проверяется товар, данные заносятся в журнал и товар размещается на складе. При отгрузке товара операция схожая только в качестве исходных документов выступают внутренние документы, предоставленные отделом логистики. Основной проблемой в этой системе это производить текущий учет прихода и расхода товара, и последующее получение этих данных другими службами организации, особенно при возникновении спорных ситуаций.

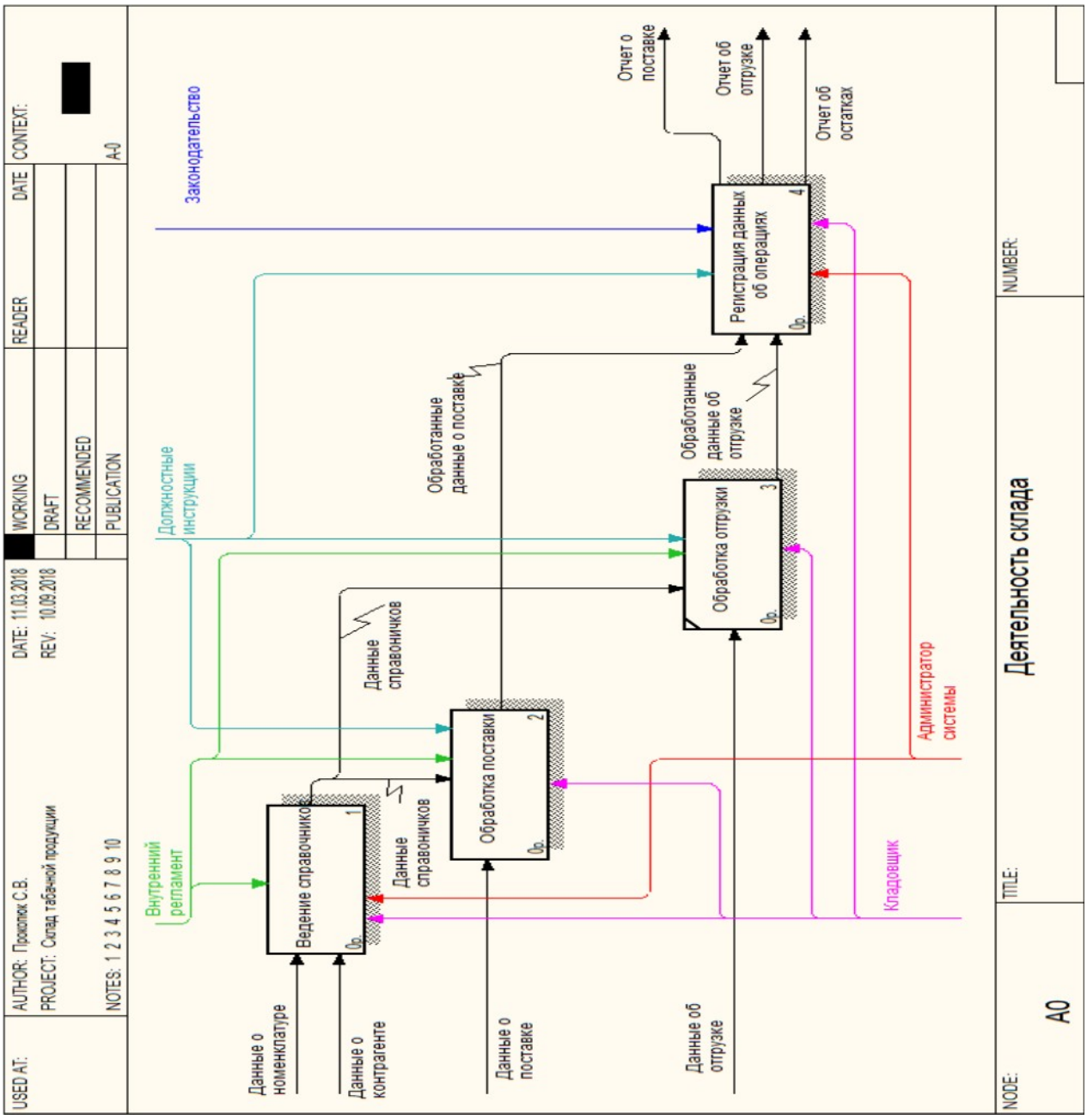


Рис. 3. Декомпозиция учета на складе

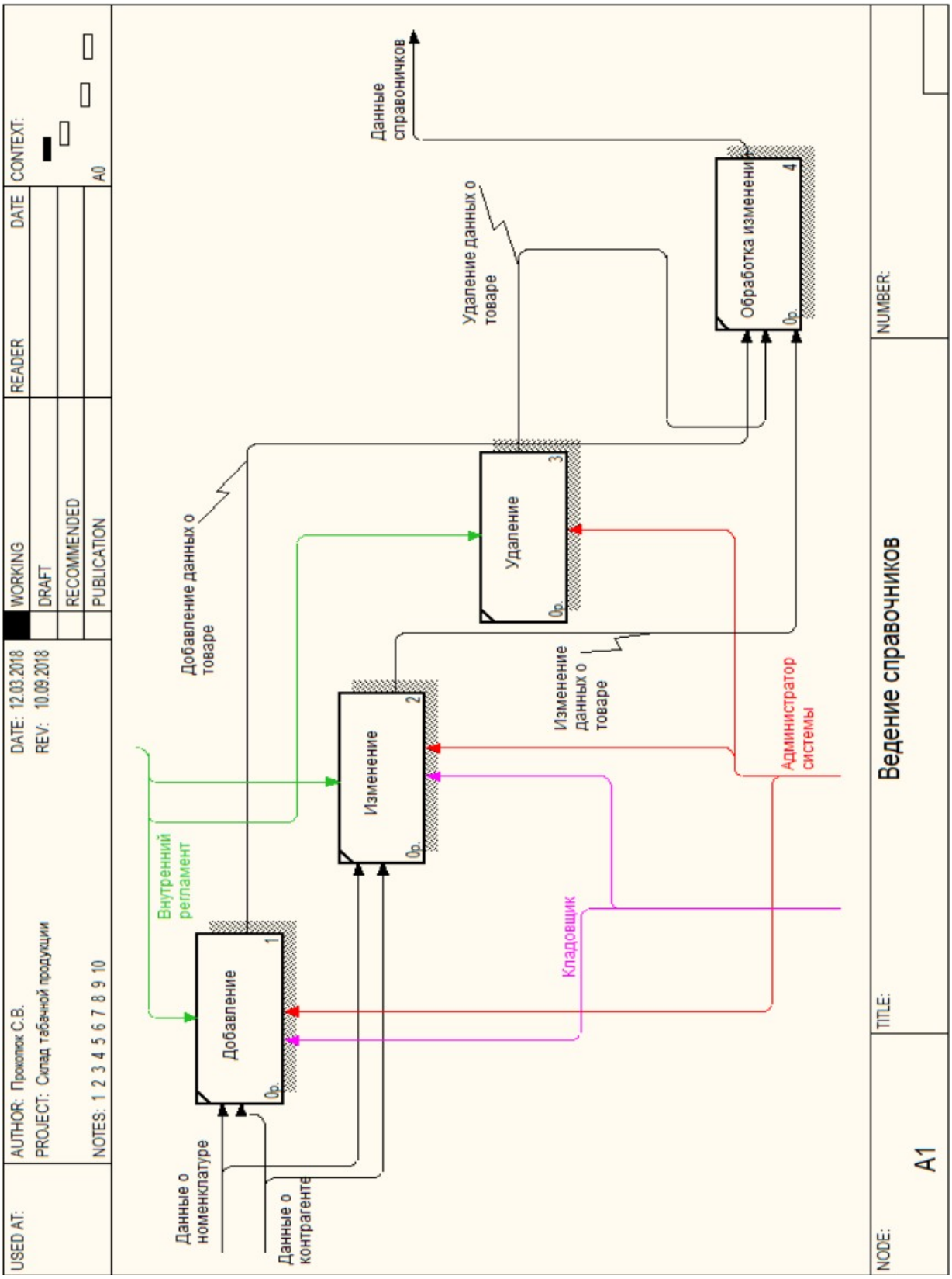


Рис. 4. Декомпозиция «Ведение справочников»

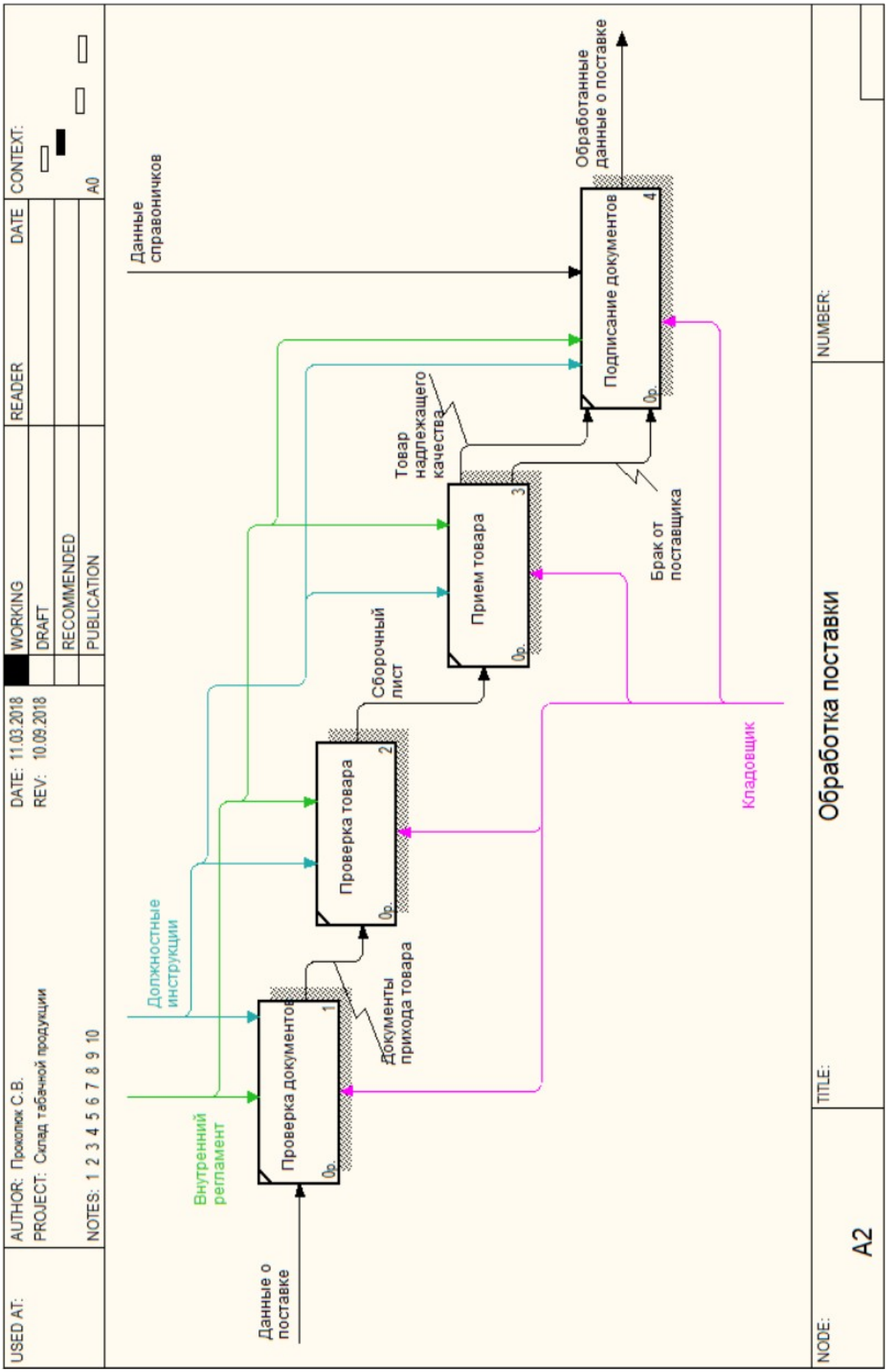


Рис. 5. Декомпозиция «Прием товара»

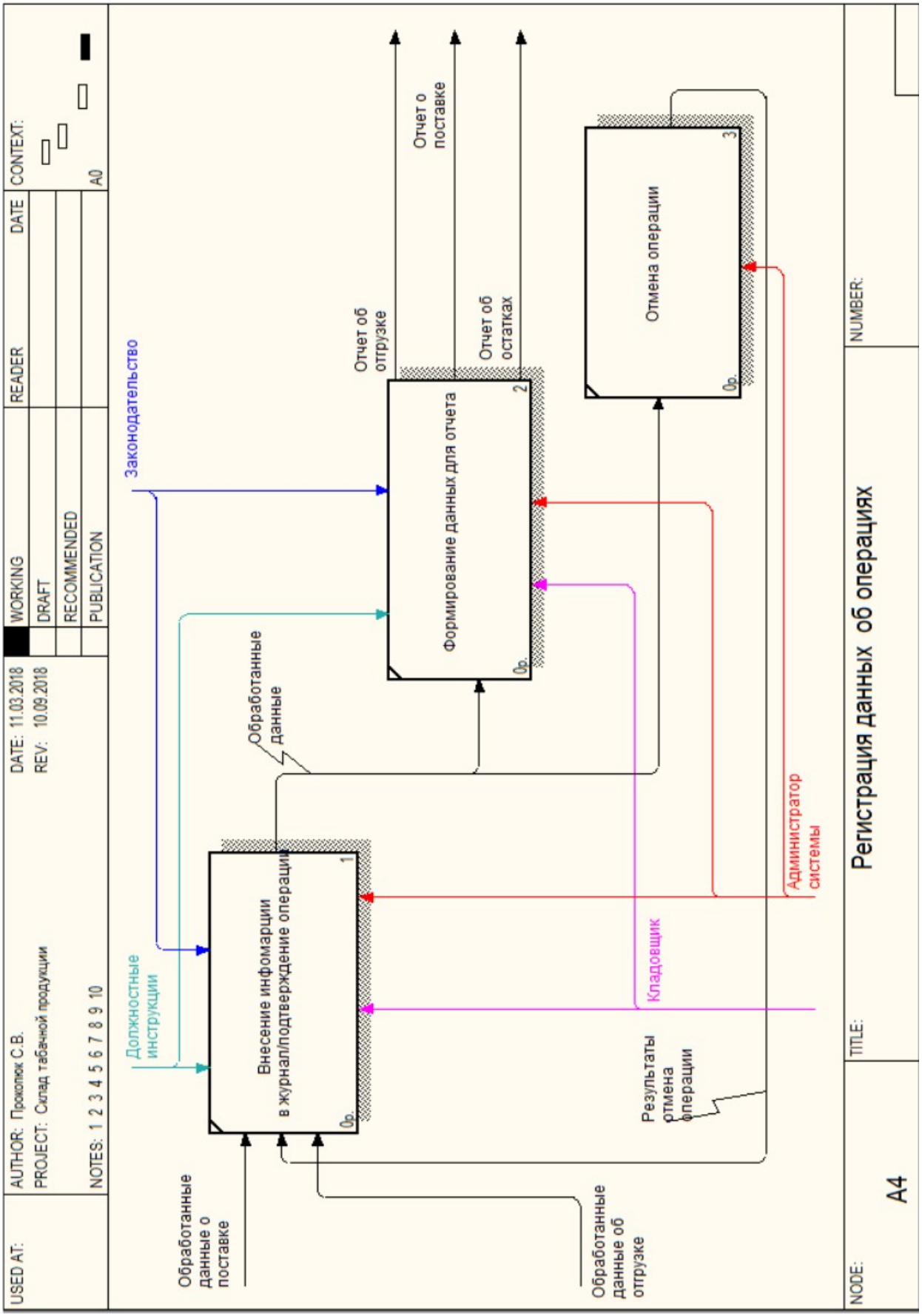


Рис. 6. Декомпозиция «Обработка поставки»

1.3. Разработка логической модели базы данных АСУД

Логическая модель данных – это описание, выполненное с использованием естественного языка, математических формул, таблиц, графиков и других средств. Основными конструктивными элементами логических моделей являются сущности, связи между ними и их свойства (атрибуты).

Методология IDEF1 является профессиональным инструментом проектирования баз данных. Система предназначена для визуального создания и модификации диаграмм «сущность-связь» (ERD – Entity Relationship Diagram). С их помощью определяются важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи).

Сущность – реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой предметной области, информация о котором подлежит хранению.

Связь – поименованная ассоциация между двумя сущностями, значимая для рассматриваемой предметной области.

Атрибут – любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности. Атрибут представляет тип характеристик или свойств, ассоциированных с множеством реальных или абстрактных объектов.

Анализируя предметную область, можно выделить следующие сущности:

- Сотрудники/пользователи;
- Склады;
- Контрагенты;
- Производители товара;
- Категории товара;
- Группы товара;
- Наименования товара;
- Номенклатура товара;
- Роли пользователей;
- Статусы документов;
- Приходные документы;
- Список товаров по приходным документам;
- Расходные документы;
- Список товаров по расходным документам;

- Остатки на складах;
- История остатков на складах;
- Таблица для временных данных приемки;
- Таблица для временных данных отгрузки.

Необходимо определить атрибуты сущности и задать связи между атрибутами.

Для сущности «Сотрудники/пользователи» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код пользователя	Целое число
Код роли	Целое число
Логин	Строка
Пароль	Строка
Фамилия	Строка
Отчество	Строка
Имя	Строка

Для сущности «Склады» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код склада	Целое число
Наименование	Строка
Адрес	Строка

Для сущности «Контрагенты» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код контрагента	Целое число
Полное наименование	Строка
Краткое наименование	Строка
Должность руководителя	Строка
ФИО руководителя	Строка
Основание	Строка
№ документа на право	Строка
Лицензия	Строка
Юридический адрес	Строка
Почтовый адрес	Строка
Адрес отгрузки товара	Строка
Телефон факс	Строка
Email	Строка
ОГРН	Строка
ИНН	Строка
КПП	Строка
Расчетный счет	Строка
ОКВЕД	Строка
ОКОНХ	Строка
ОКАТО	Строка
Признак своей компании	Логический тип

Для сущности «Производители товара» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код справочника	Целое число
Наименование	Строка

Для сущности «Категории товара» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код справочника	Целое число
Наименование	Строка

Для сущности «Группы товара» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код справочника	Целое число
Наименование	Строка

Для сущности «Наименования товара» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код справочника	Целое число
Наименование	Строка

Для сущности «Роли пользователей» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код справочника	Целое число
Наименование	Строка

Для сущности «Статусы документов» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код справочника	Целое число
Наименование	Строка

Для сущности «Номенклатура товара» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код номенклатуры	Целое число
Код товара	Целое число
Код категории	Целое число
Код группы	Целое число
Код производителя	Целое число
Цена покупки	Дробное число
Цена продажи	Дробное число

Для сущности «Список товаров по приходным документам» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код строки в приемке	Целое число
Код документа приемки	Целое число
Код номенклатуры	Целое число
Цена	Дробное число
Цена с НДС	Дробное число
Кол-во	Дробное число
Сумма	Дробное число
Сумма с НДС	Дробное число
Величина НДС	Дробное число

Для сущности «Приходные документы» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код документа приемки	Целое число
Номер документа	Строка
Дата	Дата
Код контрагента	Целое число
Код склада	Целое число
Строк в приемке	Целое число
Кол-во в приемке	Дробное число
Общая сумма	Дробное число
Общая сумма с НДС	Дробное число
Основание	Строка
Код статуса документа	Целое число
Код сотрудника	Целое число

Для сущности «Список товаров по расходным документам» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код строки в отгрузке	Целое число
Код документа отгрузки	Целое число
Код номенклатуры	Целое число
Цена	Дробное число
Цена с НДС	Дробное число
Кол-во	Дробное число
Сумма	Дробное число
Сумма с НДС	Дробное число
Величина НДС	Дробное число

Для сущности «Расходные документы» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код документа отгрузки	Целое число
Номер документа	Строка
Дата	Дата
Код контрагента	Целое число
Код склада	Целое число
Строк в приемке	Целое число
Кол-во в приемке	Дробное число
Общая сумма	Дробное число
Общая сумма с НДС	Дробное число
Основание	Строка
Код статуса документа	Целое число
Код сотрудника	Целое число

Для сущности «Остатки на складах» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код строки остатка	Целое число
Код номенклатуры	Целое число
Код склада	Целое число
Количество	Дробное число

Для сущности «История остатков на складах» выбраны следующие атрибуты:

Имя Атрибута	Тип данных
Код истории	Целое число
Код строки остатка	Целое число
Код номенклатуры	Целое число
Код склада	Целое число
Количество	Дробное число
Дата	Дата
Пользователь	Целое число
Действие	Строка

Логическая модель была построена в ERWIN и представлена на рисунке 7.

На рис. 8. представлена взаимосвязь потоков данных с сущностями разрабатываемой модели базы данных.

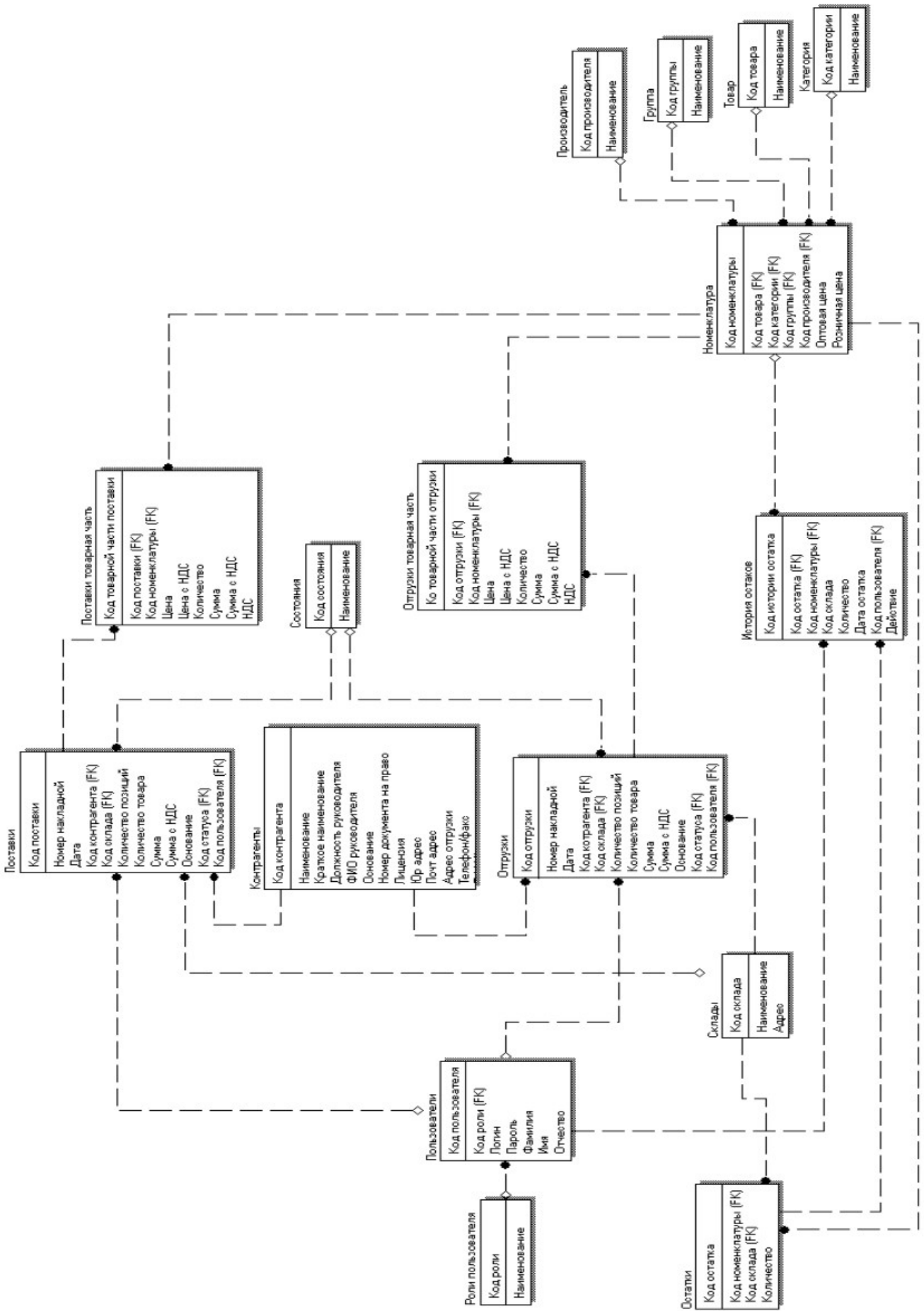


Рис. 7. Логическая модель данных

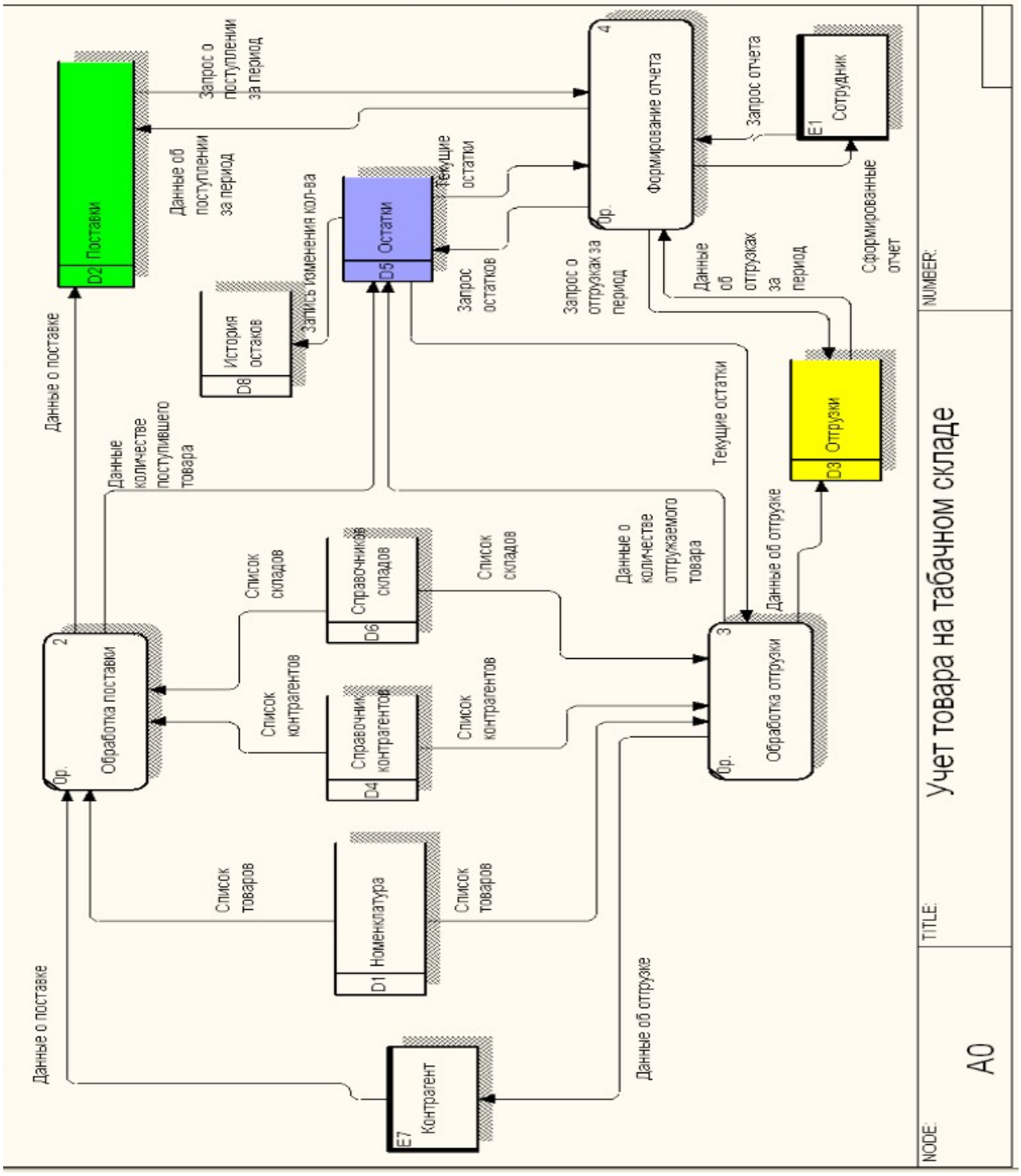


Рис. 8. Диаграмма потоков данных с учетом модели базы данных

1.4. Матрица ответственности

В данной компании, полную ответственность за работу склада и сотрудников логистики несет руководитель отдела логистики и склада. Всю работу, связанную с документооборотом, выполняет специалист отдела логистики, в то время как, сотрудник склада, отвечает за товарооборот.

Сотрудник / Процесс	Пользователь	Администратор
Формирование документов	ио	
Прием товара	ио	
Выдача товара	ио	
Авторизация пользователя	и	о
Ведение справочников		ио
Формирование отчетов	ио	

Степени ответственности.

1. Ответственный (О) – полностью отвечает за выполнение задачи и вправе принимать решения по способу ее реализации.

2. Исполнитель (И) – исполняет задачу, но в общем случае, не несет ответственности за способ ее решения.

1.5. Моделирование бизнес-вариантов использования автоматизированной информационной системы «Табачный склад»

На основе матрицы ответственности строится диаграмма бизнес-вариантов использования АИС «Табачный склад» для отдела склада предприятия АО «ТК «Мегаполис» (Рис.9.).



Рис.9. Диаграмма Business Use Case

ГЛАВА 2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ

2.1. Построение концептуальной модели

Для того чтобы графически представить поведение АИС, была применена методология визуального моделирования. Унифицированный язык моделирования UML – это нотация, которая позволяет детально описать информационную систему, а также отметить особенности реализации системы [11].

На рисунке 10 представлена диаграмма вариантов использования, отражающая отношения между актёрами и прецедентами.

Диаграмма вариантов использования описывает функциональность АИС. Каждая функциональность изображается в виде прецедентов использования (use case) или просто прецедентов [8].

Прецедент обозначается на диаграмме овалом, связанным с пользователями, которых принято называть действующими лицами (актерами, actors). Список всех прецедентов фактически определяет функциональные требования к АИС, которые лежат в основе разработки технического задания на создание системы [19].

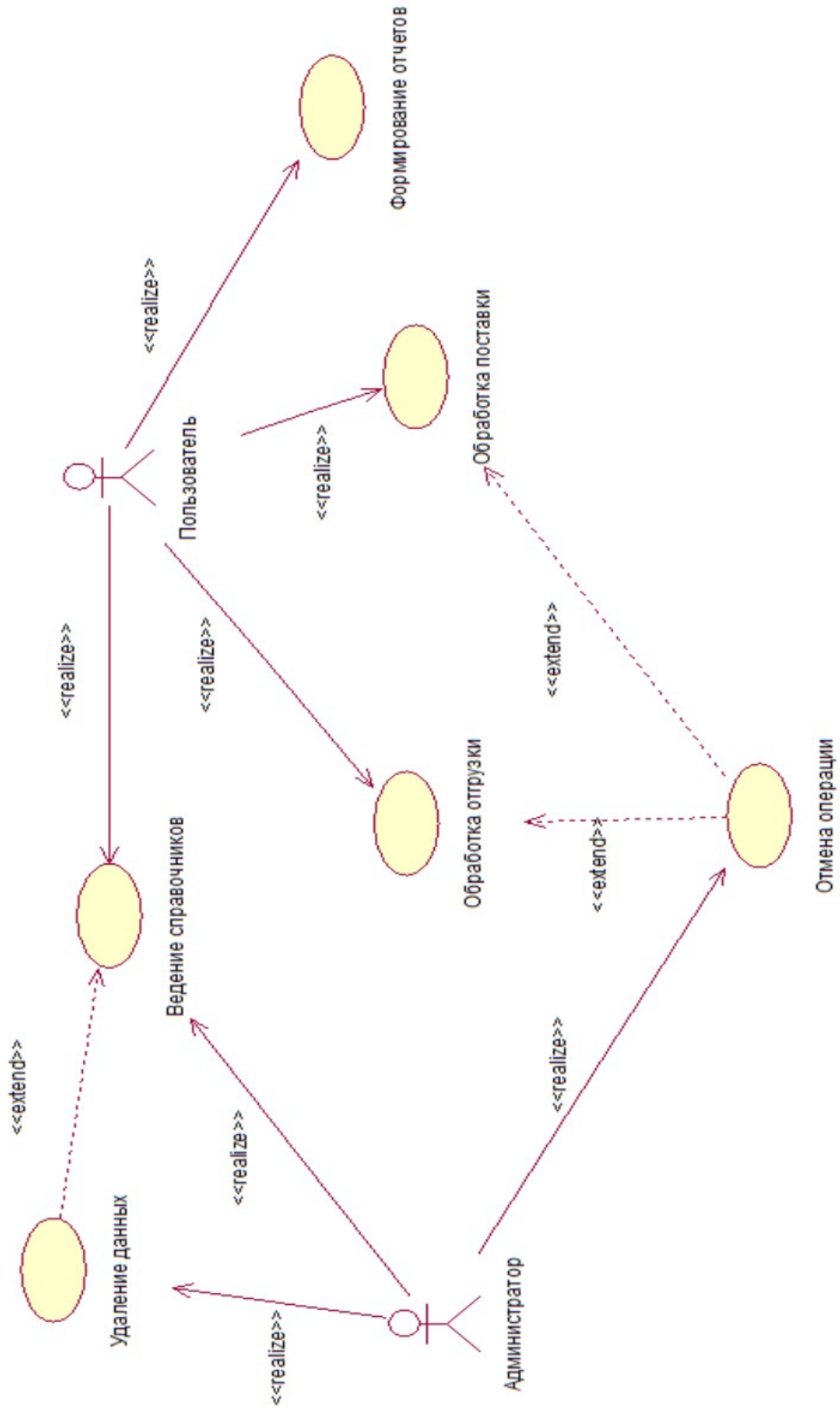


Рис. 10. Диаграмма вариантов использования АИС «Табачный склад»

На диаграмме видно, что система реализует основные действия необходимые для решения задач, поставленных на этапе технического задания:

- Обработка поставки;
- Обработка отгрузки;
- Ведение вспомогательных справочников;
- Формирование отчетных документов по обороту товаров.

Трассировка – один из видов зависимости, указывающий на связь между элементами, которые представляют собой одну и туже концепцию, находящуюся на разных уровнях значимости [11].

Стрелка «include» (включение) – показывает отношение между базовым и включаемым элементами. С его помощью определяется, каким образом поведение, определённое для включаемого элемента, можно вставить в поведение базового элемента. Базовый элемент видит включение и может зависеть от его результатов, однако базовый элемент не имеет доступа к атрибутам включаемого элемента [7].

Стрелка «realize» (реализовывать) – показывает отношение между спецификацией и ее программной реализацией. Описывает объекты, которые реализуют поведение в программном коде и то, каким способом они для этого взаимодействуют между собой [7].

Стрелка «extend» (расширение) – показывает отношения между расширяющим и базовым элементами. Описывает, каким образом поведение, определённое для расширяющего элемента, может быть внесено в поведение, определенное для базового элемента [8].

Диаграмма трассировки, отражающей разбиение задач на более мелкие подзадачи, для автоматизированной информационной системы «Табачный склад» приведена на рисунке 11.

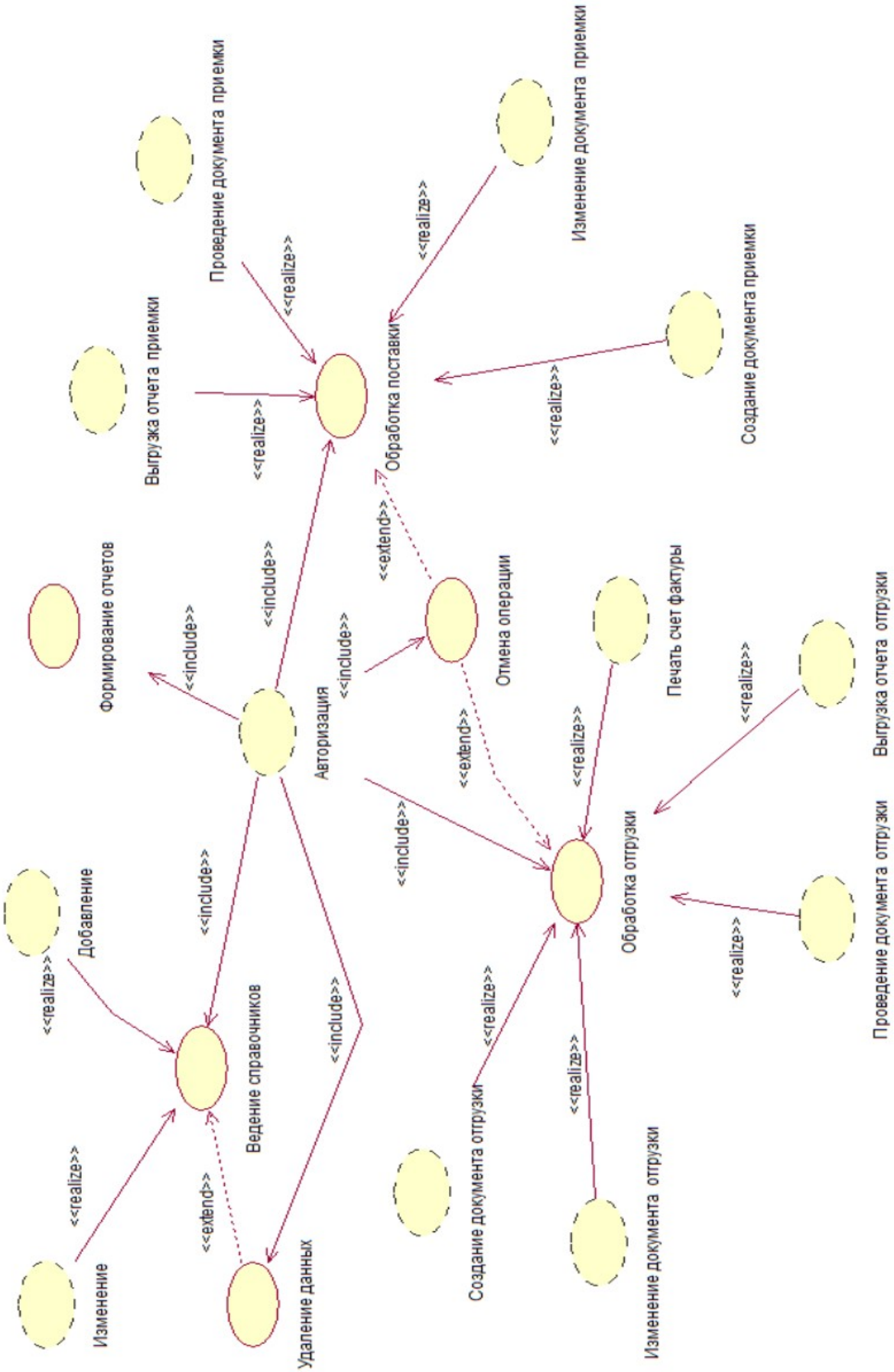


Рис. 11. Диаграмма трассировки АИС «Табачный склад»

2.2. Реализация функциональной архитектуры

Кроме общей информации о варианте использования, описан его основной сценарий и расширение к нему. Основной сценарий представляет собой последовательность действий, при успешном выполнении которых достигается цель варианта использования. Расширения основного сценария описывают действия при возникновении исключительных ситуаций.

Далее рассмотрены сценарии вариантов использования, каждый из которых представлен графическими диаграммами (рисунки 12 – 36).

Для составления диаграмм были использованы:

- Диаграммы последовательности;
- Диаграммы взаимодействия.

Диаграмма последовательности действий отображает взаимодействие объектов, упорядоченное во времени. На ней показаны объекты и классы, используемые в сценарии, и последовательность сообщений, которыми обмениваются объекты для выполнения сценария. Диаграммы последовательности действий обычно соответствуют реализациям вариантов использования в текстовом представлении.

Диаграмма кооперации – это альтернативный способ отображения сценариев. Такой тип диаграммы показывает взаимодействие объектов, организованное вокруг них, и их связи друг с другом. Подобно диаграммам последовательности, кооперативные диаграммы отображают поток событий через конкретный сценарий варианта использования. Диаграммы последовательности упорядочены по времени, а кооперативные диаграммы заостряют внимание на связях между объектами [8].

Актеры в системе учет на табачном складе.

1. Пользователь – вводит/изменяет данные;
2. Администратор – в дополнение к возможностям пользователя может удалить данные, отменять операцию по проводке документов и добавлять новых сотрудников в систему.

Диаграммы последовательности и взаимодействия.

Вариант использования «Авторизация»:

Главное действующее лицо: пользователь системы.

Контекст использования: пользователь системы проходит авторизацию в системе.

Результат успешного завершения: сотрудник вошел в систему.

Сценарий:

1. Система проверяет подключение к базе данных (БД);
2. Систем запрашивает настройку подключения;

3. Сотрудник настраивает подключение к БД;
4. Система запрашивает имя пользователя и пароль;
5. Сотрудник вводит имя пользователя и пароль;
6. Система подтверждает имя и пароль, после чего открывается доступ в систему.

Диаграмма последовательности, иллюстрирующая описанный сценарий «Авторизация», изображена на рисунке 12, и диаграмма кооперации на рисунке 13.

В верхней части диаграммы показаны все действующие лица и объекты, требуемые в системе для выполнения этого варианта использования: пользователь, форма «заставка», форма «авторизации», форма «настройка подключения», форма «главная форма», база данных (БД).

Стрелки соответствуют сообщениям, передаваемым между действующим лицом и объектом или между объектами для выполнения требуемых функций.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится, что подключение настроено неверно – системы выводит сообщение об ошибке и пользователь может вернуться к началу ввода данных. Если обнаружится, что пользователь ввел неправильный пароль, система выводит сообщение об ошибке. Пользователь может вернуться к началу основного сценария или отказаться от входа в систему, при этом выполнение варианта использования завершается.

Если вариант использования выполнен успешно, пользователь входит в систему. Отображается главная форма системы. В противном случае состояние системы не изменяется.

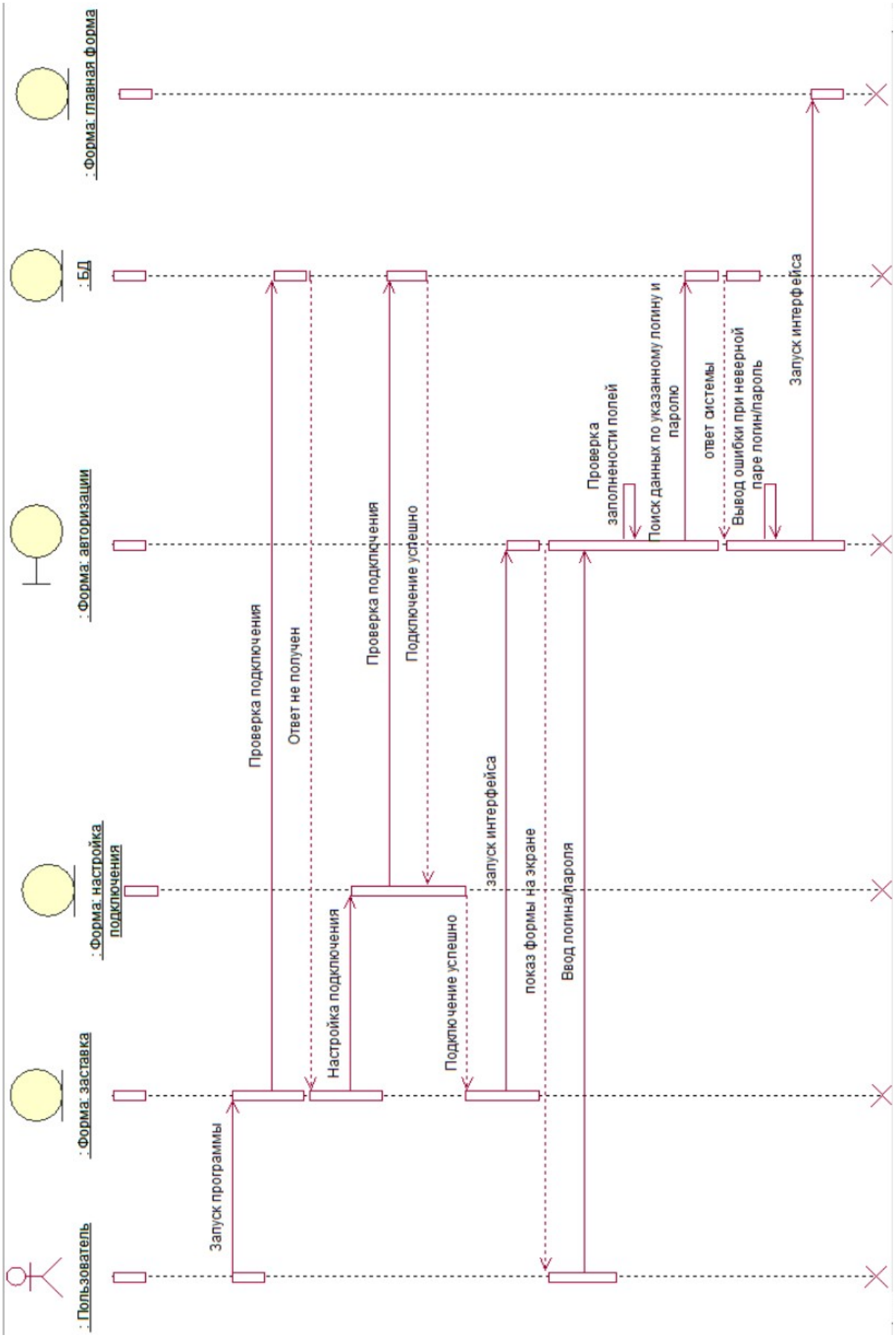


Рис. 12. Диаграмма последовательности действий варианта использования «Авторизация»

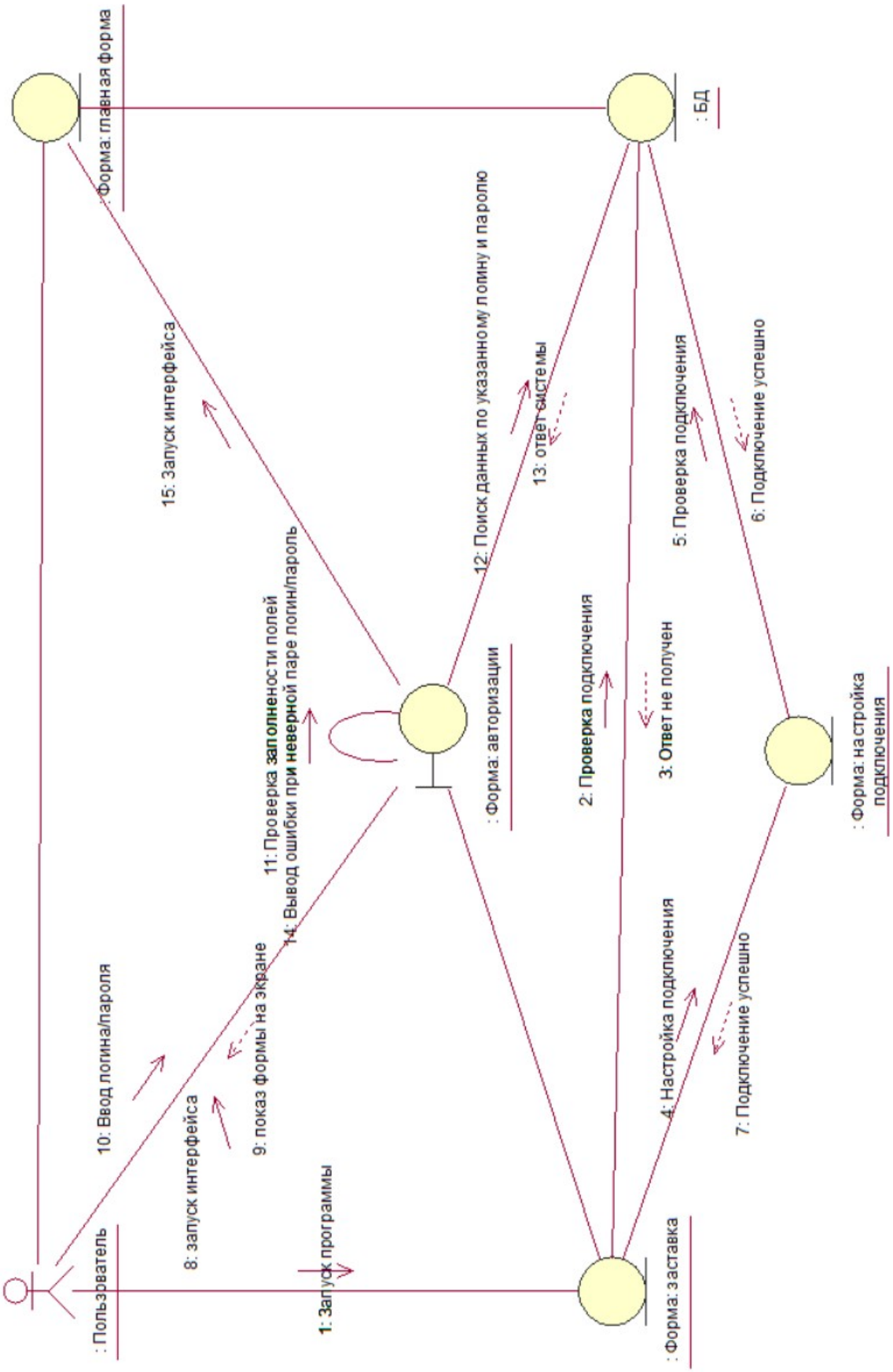


Рис. 13. Диаграмма кооперации действий варианта использования «Авторизация»

Вариант использования «Добавление элемента справочника»:

Главное действующее лицо: пользователь системы (кладовщик/администратор).

Контекст использования: пользователь добавляет новую запись в выбранный справочник.

Результат успешного завершения: в системе добавляется запись с новым элементом справочника.

Сценарий:

1. Пользователь запрашивает справочник;
2. Система выбирает данные справочника из базы и показывает их на форме;
3. Пользователь заполняет поля элемента справочника и нажимает кнопку «добавить»;
4. Система проверяет правильность занесения данных/подтверждение изменения;
5. Сохраняет изменения в базу;
6. Форма справочника запрашивает обновленные данные для отображения;
7. Отображение обновлённых данных.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится, поля заполнены некорректно, система выводит сообщение об ошибке и пользователь может вернуться к началу ввода данных. Пользователь может вернуться к началу основного сценария или отказаться выполнения сценария, при этом выполнение варианта использования завершается.

Если вариант использования выполнен успешно, справочник обновляется. В противном случае состояние системы не изменяется.

При вводе данных проверяется заполненность полей согласно регламенту.

Диаграмма последовательности, иллюстрирующая описанный сценарий, изображена на рисунках 14 и 15.

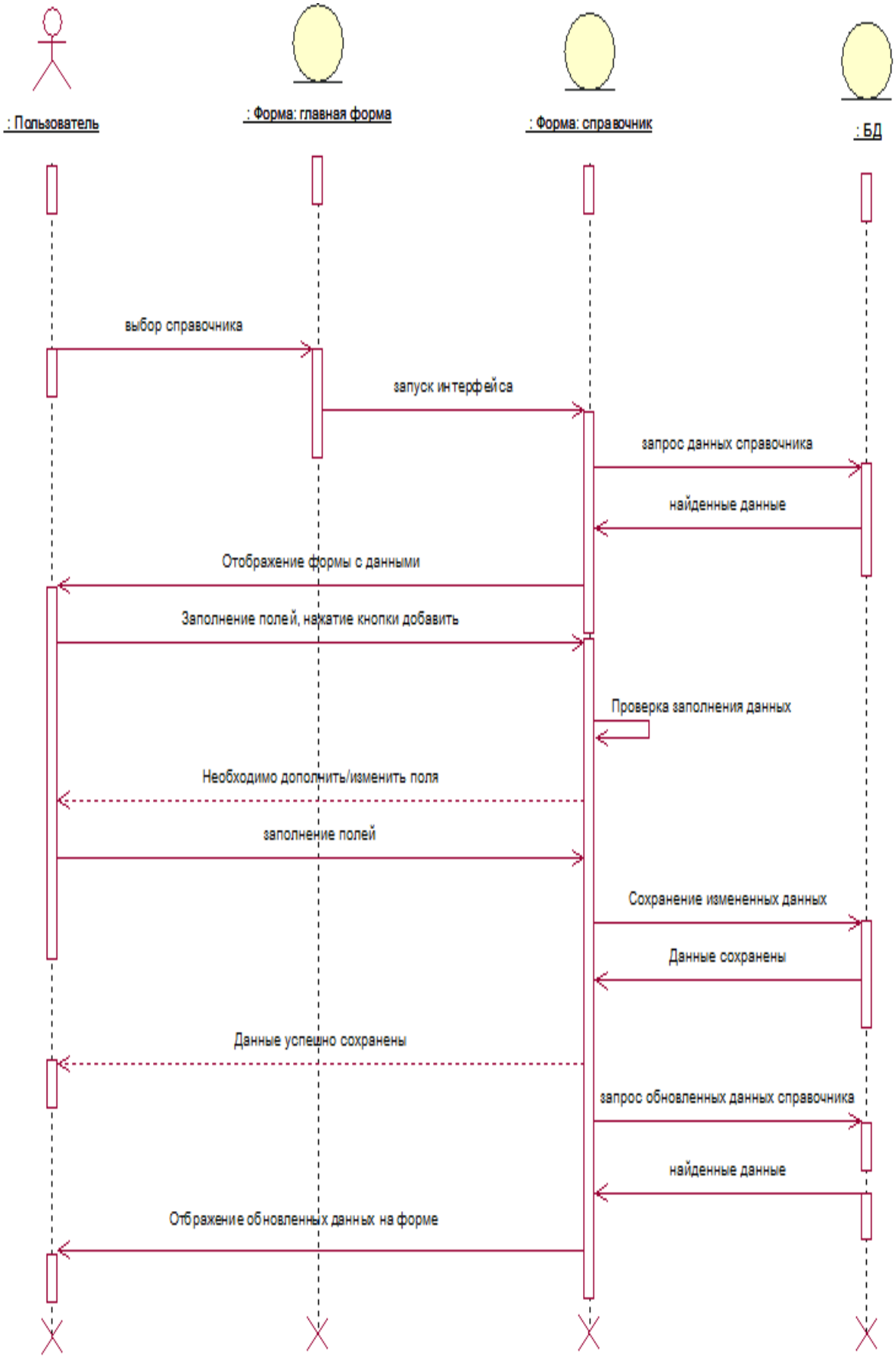


Рис. 14. Диаграмма последовательности действий варианта использования «Добавление элемента справочника»

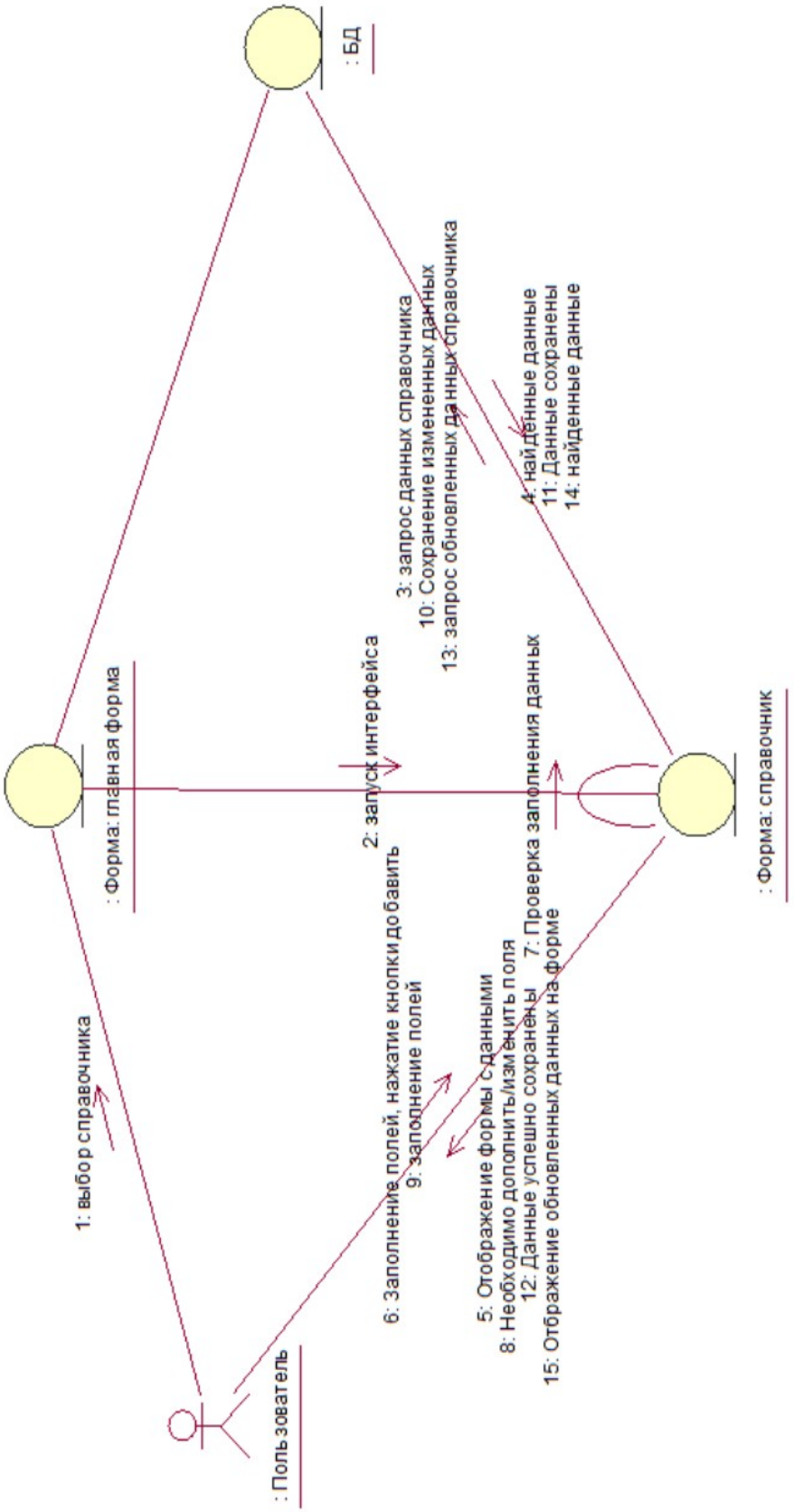


Рис. 15. Диаграмма кооперации действий варианта использования «Добавление элемента справочника»

Вариант использования «Изменение элемента справочника»:

Главное действующее лицо: пользователь системы (кладовщик/администратор).

Контекст использования: пользователь изменяет выбранную запись в справочнике.

Результат успешного завершения: в системе обновляется запись в справочнике.

Сценарий:

1. Пользователь запрашивает справочник;
2. Система выбирает данные справочника из базы и показывает их на форме;
3. Пользователь выбирает элемент справочника;
4. Пользователь изменяет/дополняет поля элемента справочника и нажимает кнопку «изменить»;
5. Система проверяет правильность занесения данных/подтверждение изменения;
6. Сохраняет изменения в базу;
7. Форма справочника запрашивает обновленные данные для отображения;
8. Отображение обновлённых данных.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится, поля заполнены некорректно, система выводит сообщение об ошибке и пользователь может вернуться к началу ввода данных. Пользователь может вернуться к началу основного сценария или отказаться выполнения сценария, при этом выполнение варианта использования завершается.

Если вариант использования выполнен успешно, справочник обновляется. В противном случае состояние системы не изменяется.

При вводе данных проверяется заполненность полей согласно регламенту.

Диаграмма последовательности, иллюстрирующая описанный сценарий, изображена на рисунках 16 и 17.

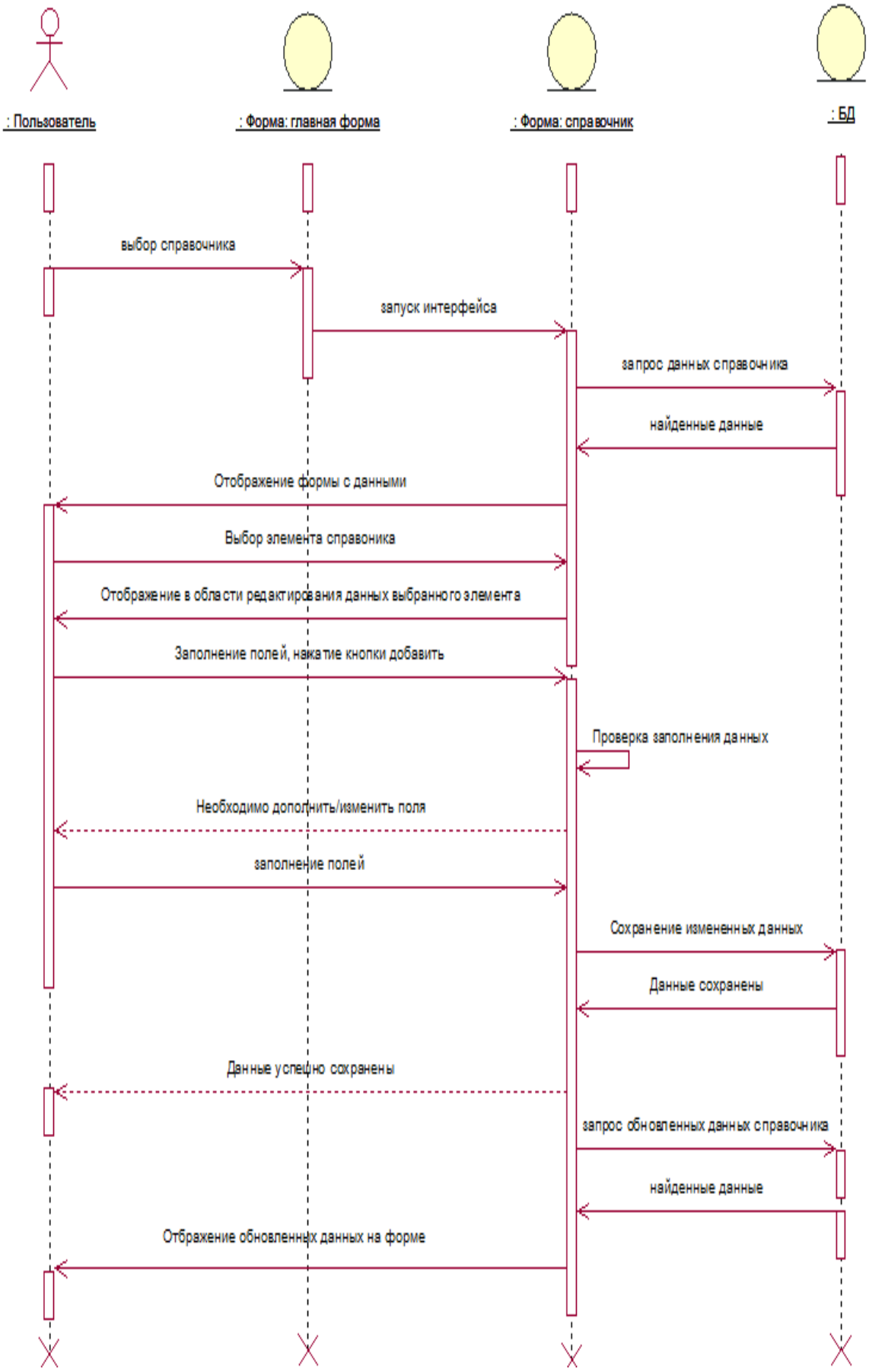


Рис. 16. Диаграмма последовательности действий варианта использования «Изменение элемента справочника»

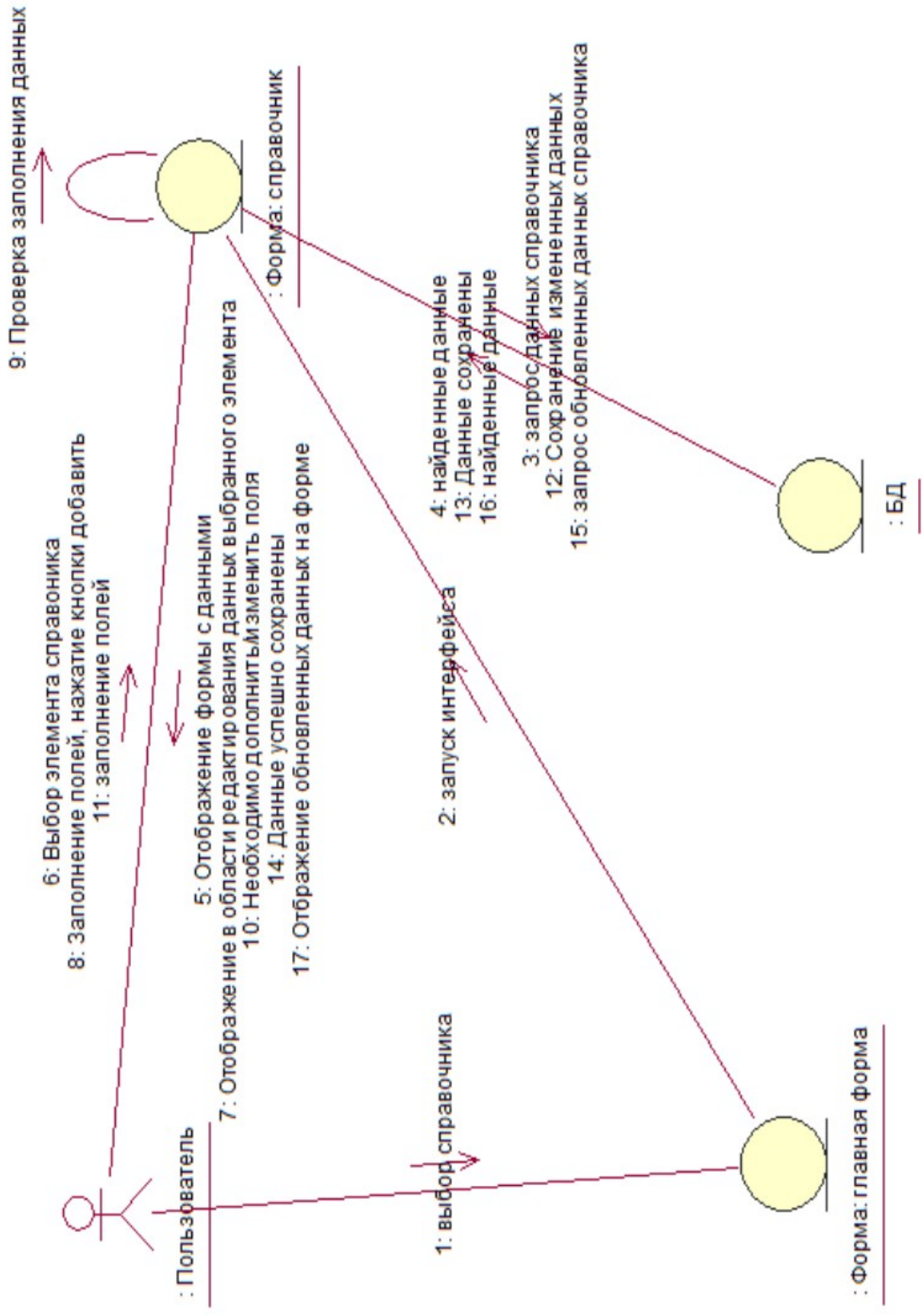


Рис. 17. Диаграмма кооперации действий варианта использования «Изменение элемента справочника»

Вариант использования «Удаление элемента справочника»:

Главное действующее лицо: администратор.

Контекст использования: администратор удаляет выбранную запись в справочнике.

Результат успешного завершения: в системе обновляется запись в справочнике.

Сценарий:

1. Пользователь запрашивает справочник;
2. Система выбирает данные справочника из базы и показывает их на форме;
3. Пользователь выбирает элемент справочника;
4. Пользователь нажимает кнопку удалить элемент справочника и нажимает кнопку «Удалить»;
5. Система запрашивает подтверждения удаления;
6. Пользователь подтверждает удаление;
7. Форма справочника сохраняет изменения в базу;
8. Форма справочника запрашивает обновленные данные для отображения;
9. Отображение обновлённых данных.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, справочник обновляется. В противном случае состояние системы не изменяется.

При вводе данных проверяется заполненность полей согласно регламенту.

Диаграмма последовательности, иллюстрирующая описанный сценарий, изображена на рисунке 18 и диаграмма кооперации на рисунке 19.

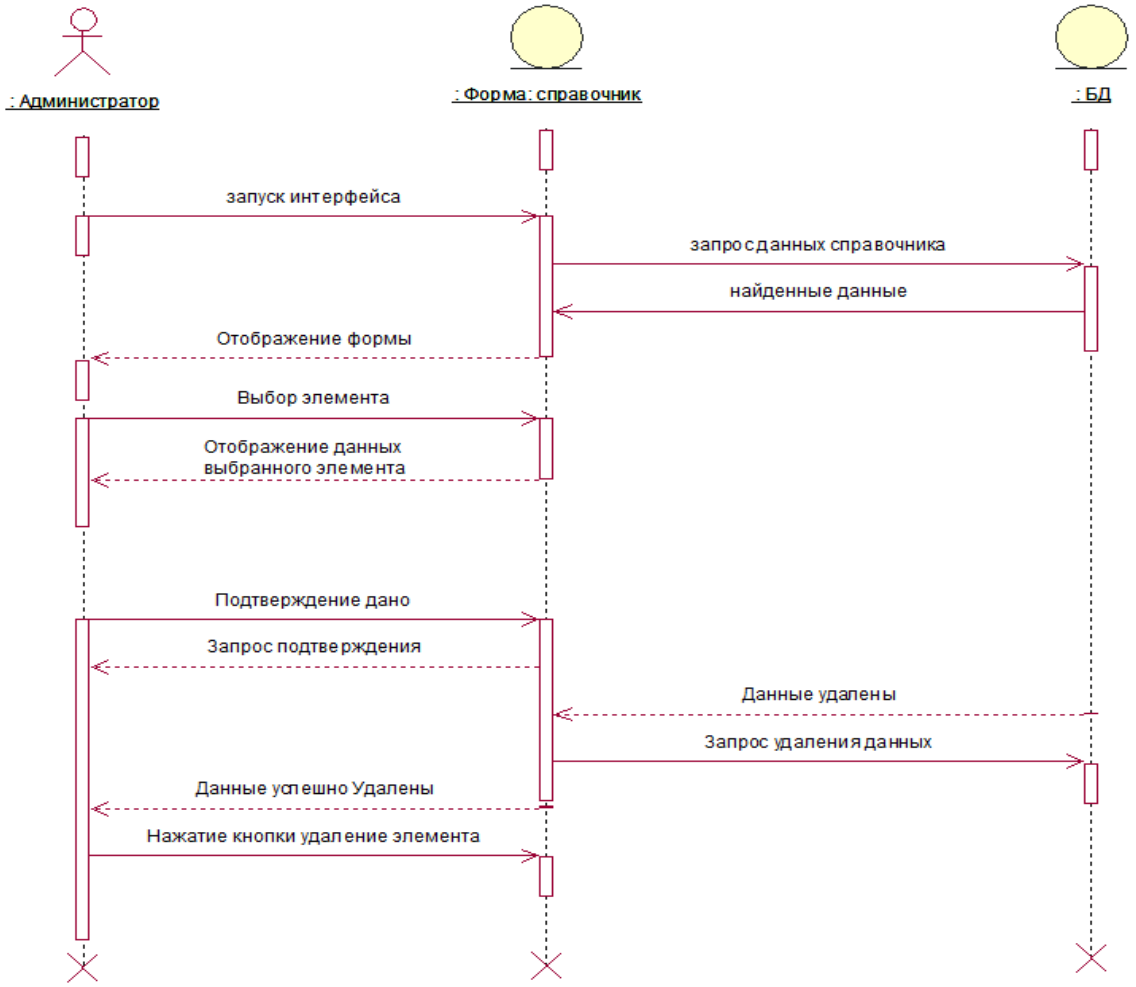


Рис. 18. Диаграмма последовательности действий варианта использования «Удаление элемента справочника»

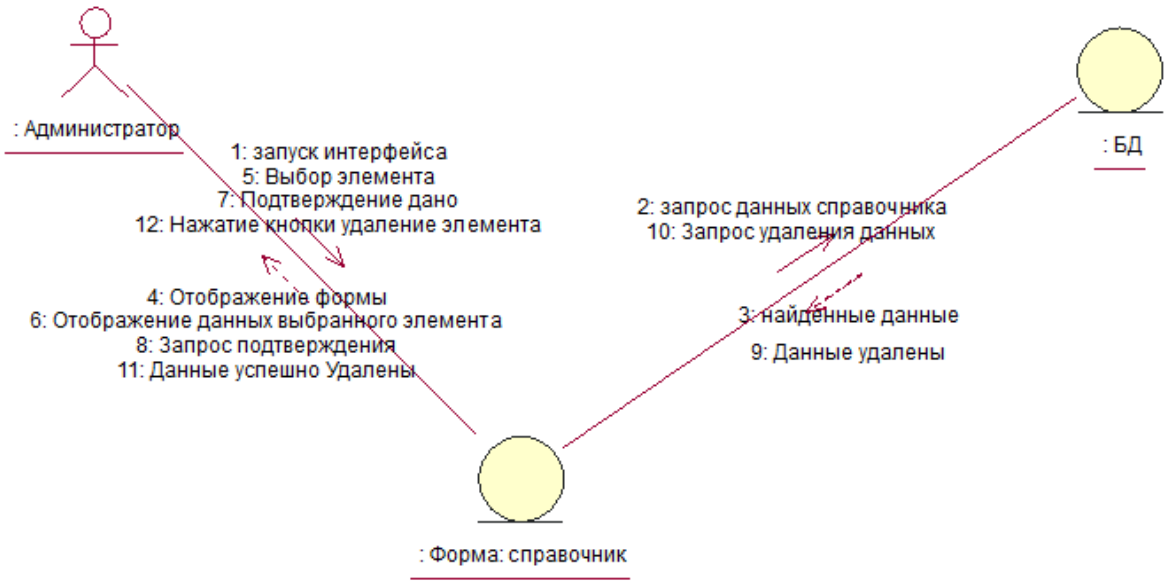


Рис. 19. Диаграмма кооперации действий варианта использования «Удаление элемента справочника»

Вариант использования «Добавление приемки»:

Главное действующее лицо: пользователь.

Контекст использования: кладовщик добавлять данные о приемке в систему.

Результат успешного завершения: в системе добавляется запись о приемке в статусе «черновик».

Сценарий:

1. Выбор меню приемки товара на главной форме;
2. Запрос документов о приемке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных о приемке на форме;
5. Переход в режим приемки нового товара, путем нажатия на кнопку «Добавить приемку»;
6. Запуск интерфейса формы приемки;
7. Отображение формы на экране компьютера пользователя;
8. Пользователь вводит данные по приемке;
9. Система проверяет заполненность всех необходимых полей;
10. Пользователь нажимает сохранить приемку;
11. Данные введённые пользователем направляется через запрос в БД;
12. База возвращает информацию об успешной записи данных;
13. Отображение нового списка документов по приемке, с новой строкой в статусе черновик.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 20 и 21.

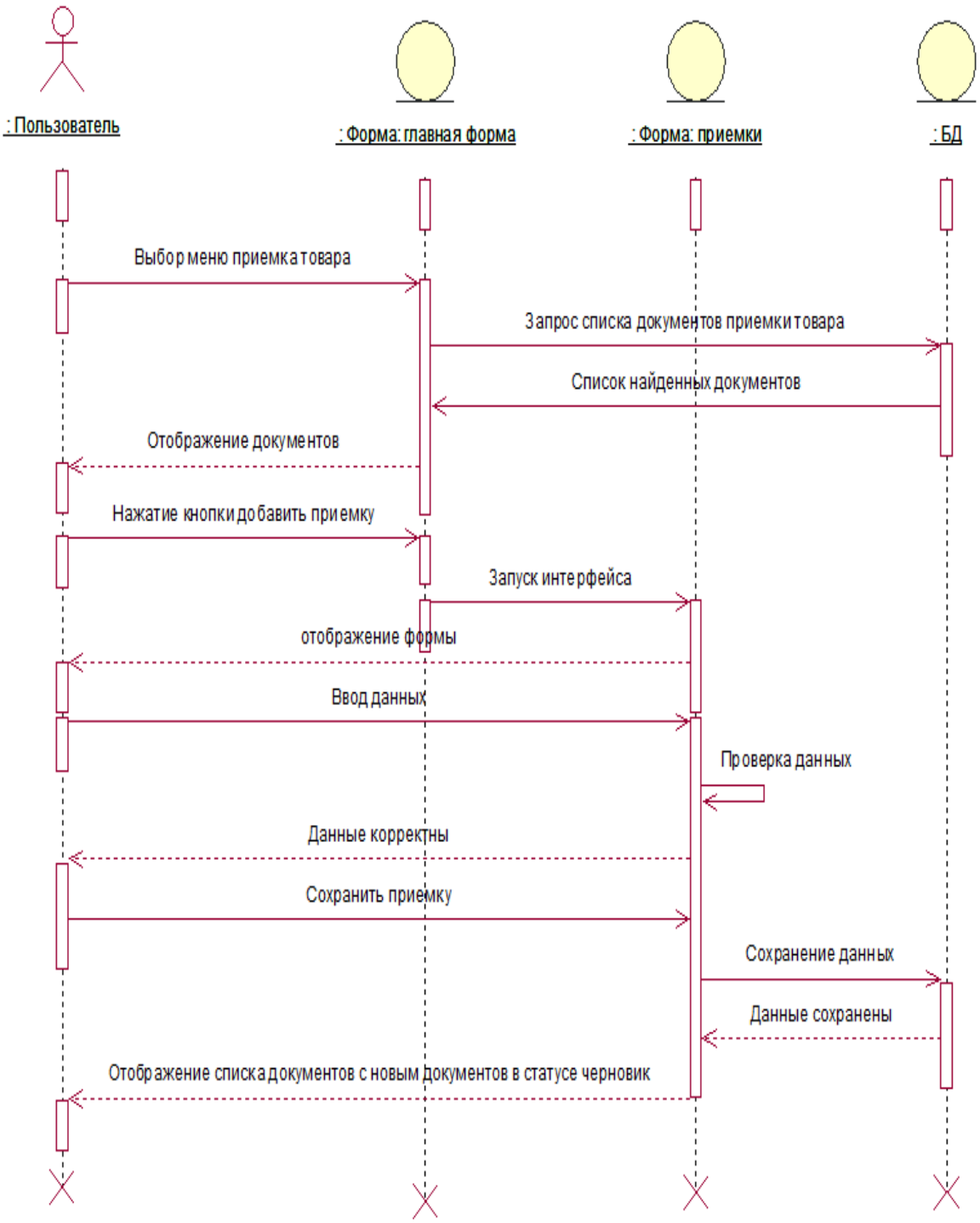


Рис. 20. Диаграмма последовательности действий варианта использования «Добавление приемки»

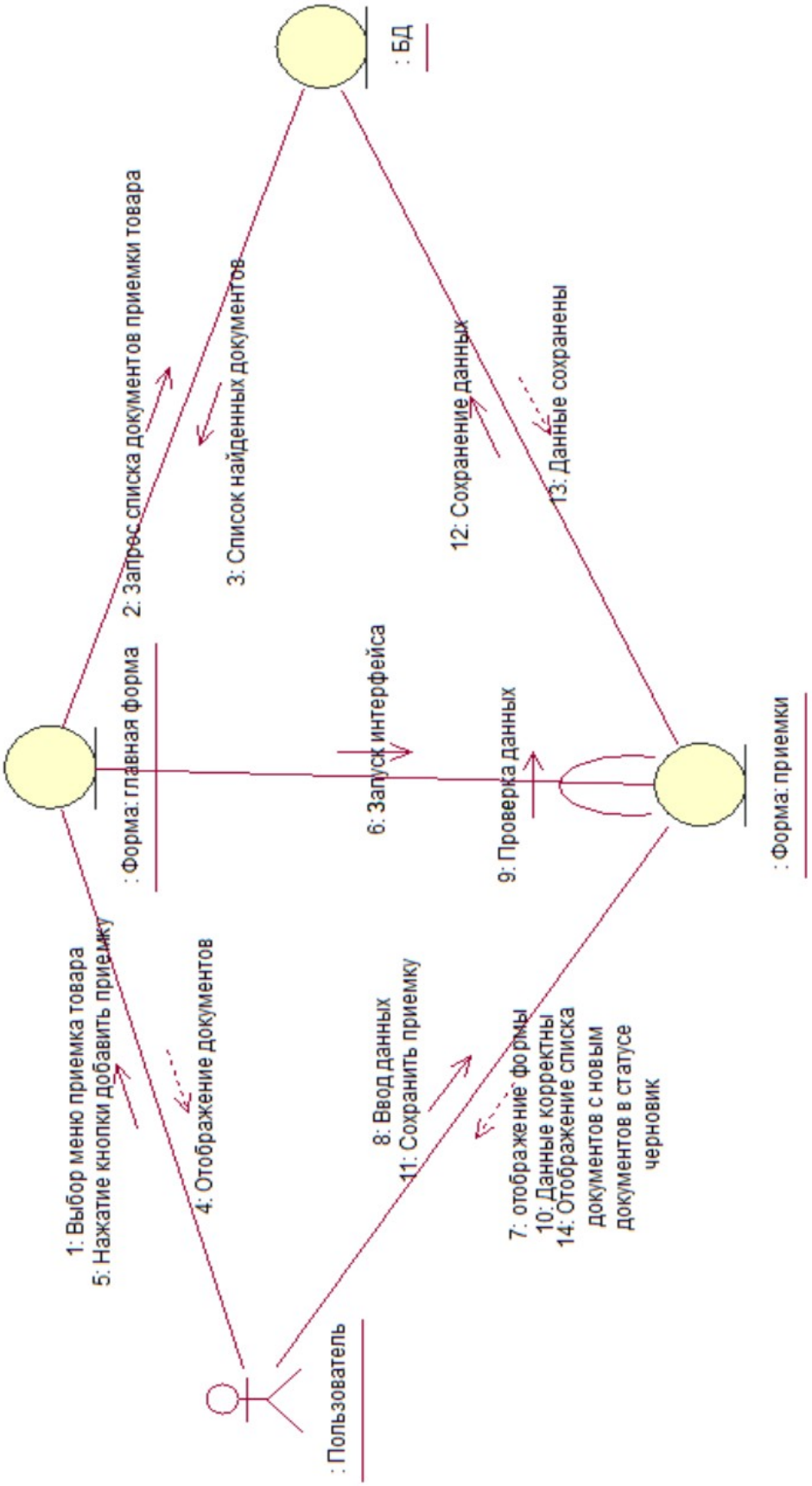


Рис. 21. Диаграмма кооперации действий варианта использования «Добавление приемки»

Вариант использования «Изменение приемки»:

Главное действующее лицо: пользователь.

Контекст использования: кладовщик изменяет данные о приемке в системе.

Результат успешного завершения: в системе изменяет запись о приемке в статусе «черновик».

Сценарий:

1. Выбор меню приемки товара на главной форме;
2. Запрос документов о приемке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных о приемке на форме;
5. Выбор документа для изменения;
6. Переход в режим изменение приемки, путем нажатия на кнопку «Изменить приемку»;
7. Запуск интерфейса формы приемки;
8. Отображение формы на экране компьютера пользователя с данными по приемке;
9. Пользователь изменяет данные по приемке;
10. Система проверяет заполненность всех необходимых полей;
11. Пользователь нажимает сохранить приемку;
12. Данные введённые пользователем направляется через запрос в БД;
13. База возвращает информацию об успешной записи данных;
14. Отображение списка документов по приемке, с измененной строкой в статусе черновик.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 22 и 23.

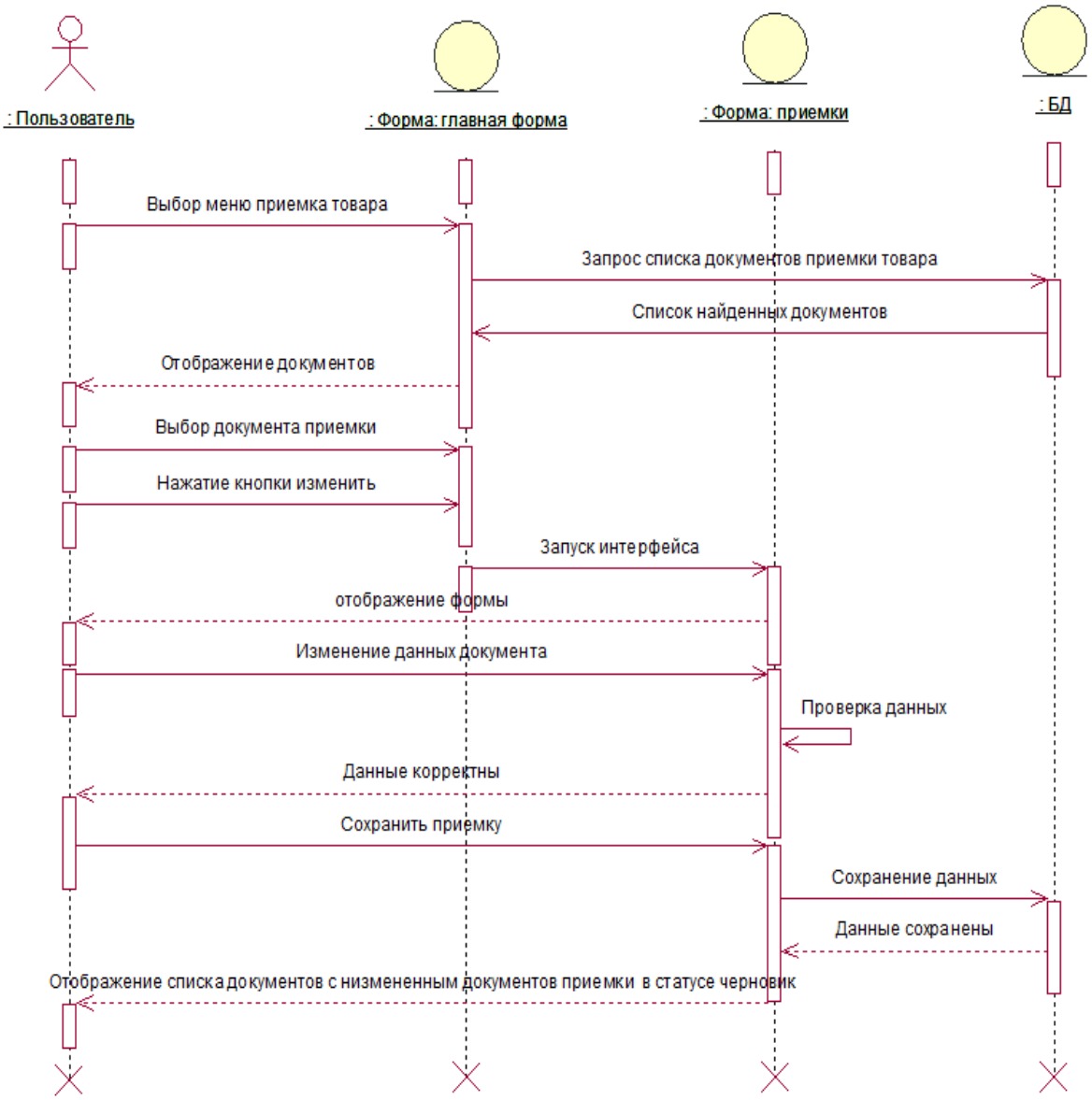


Рис. 22. Диаграмма последовательности действий варианта использования «Изменение приемки»

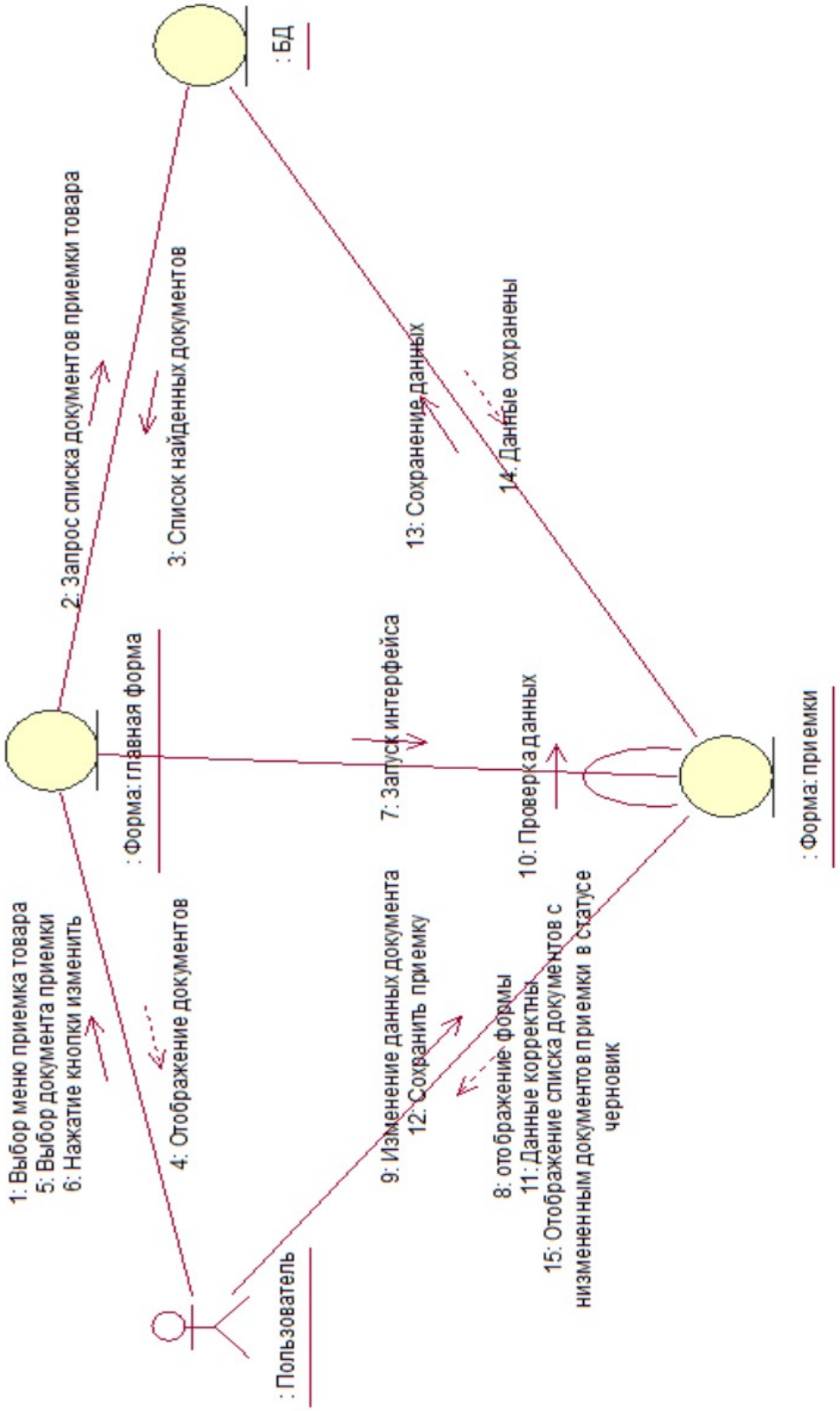


Рис. 23. Диаграмма кооперации действий варианта использования «Изменение приемки»

Вариант использования «Удаление приемки»:

Главное действующее лицо: администратор.

Контекст использования: администратор удаляет данные о приемке в системе.

Результат успешного завершения: в системе удаляет запись о приемке в статусе «черновик».

Сценарий:

1. Выбор меню приемки товара на главной форме;
2. Запрос документов о приемке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных о приемке на форме;
5. Выбор документа для удаления;
6. Проверка статуса документа (в статусе «черновик»);
7. Пользователь нажимает удалить приемку;
8. Запрос подтверждения удаления;
9. Пользователь подтверждает удаление;
10. Направляется запрос в БД на удаление данных;
11. База возвращает информацию об успешном удалении данных;
12. Отображение списка документов по приемке, без удаленной записи.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 24 и 25.

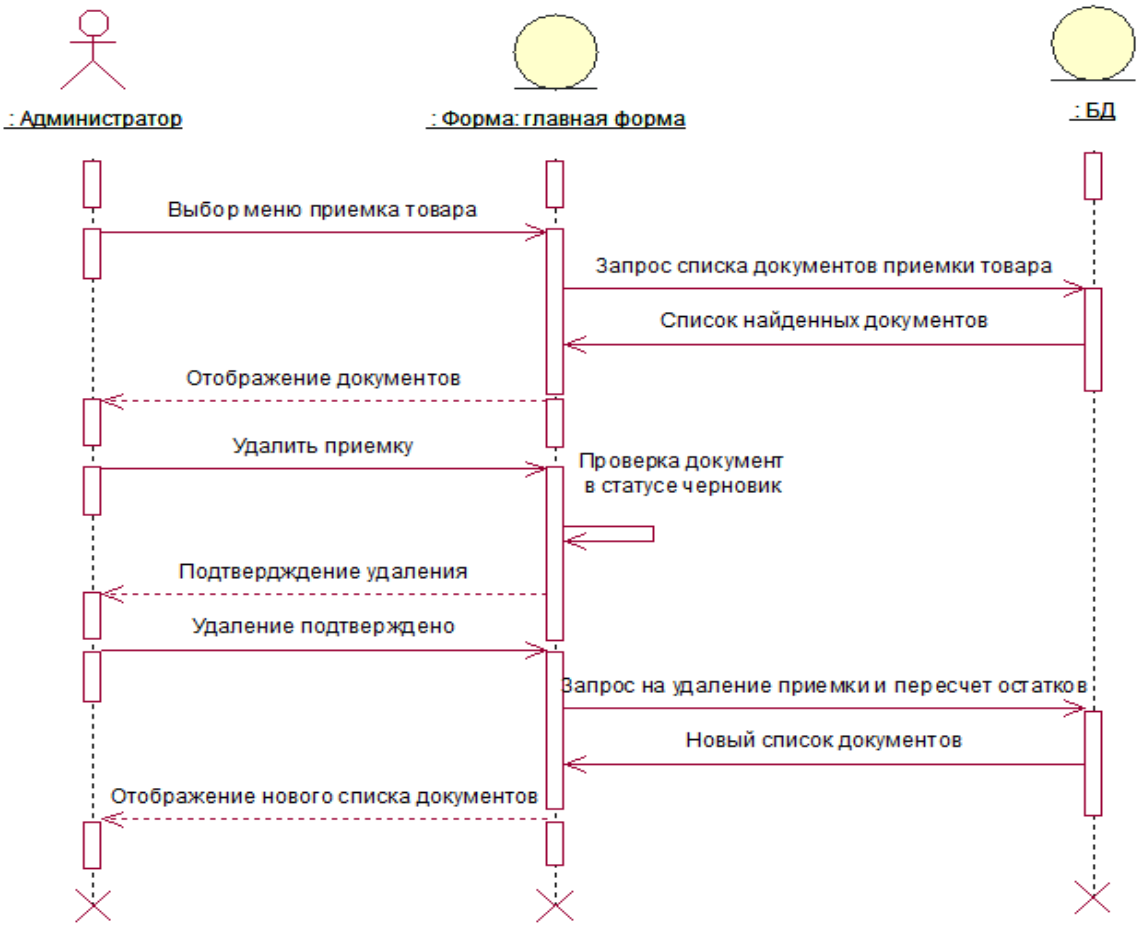


Рис. 24. Диаграмма последовательности действий варианта использования «Удаление приемки»

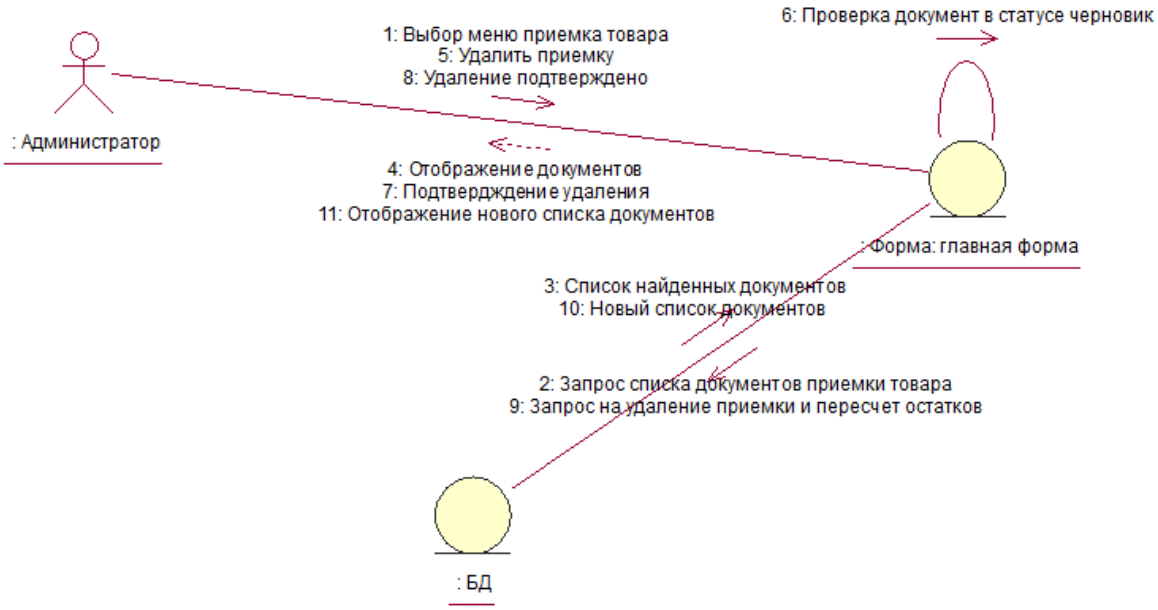


Рис. 25. Диаграмма кооперации действий варианта использования «Удаление приемки»

Вариант использования «Проведение документа»:

Главное действующее лицо: пользователь.

Контекст использования: пользователь проводит выбранный документ.

Результат успешного завершения: система меняет статус документа на проведен и пересчитывает остатки на складах.

Сценарий:

1. Выбор меню приемки/отгрузки товара на главной форме;
2. Запрос документов о приемке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных о приемке/ отгрузки на форме;
5. Выбор документа для проведения (документ в статусе «черновик»);
6. Проверка статуса документа (в статусе «черновик»);
7. Пользователь нажимает провести документ;
8. Направляется запрос в БД на проведение документа и пересчет остатков;
9. База возвращает информацию об успешной смене статуса;
10. Отображение обновленного списка документов.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 26 и 27.

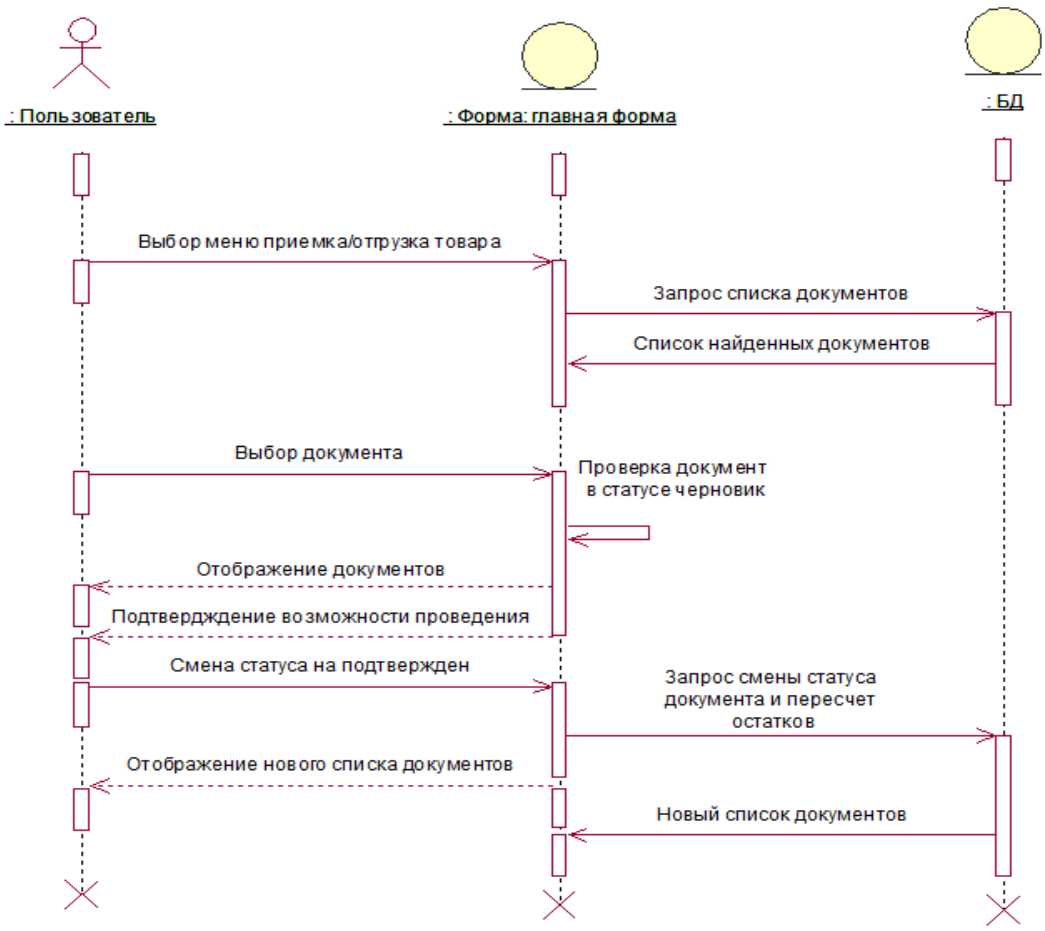


Рис. 26. Диаграмма последовательности действий варианта использования «Проведение документа»

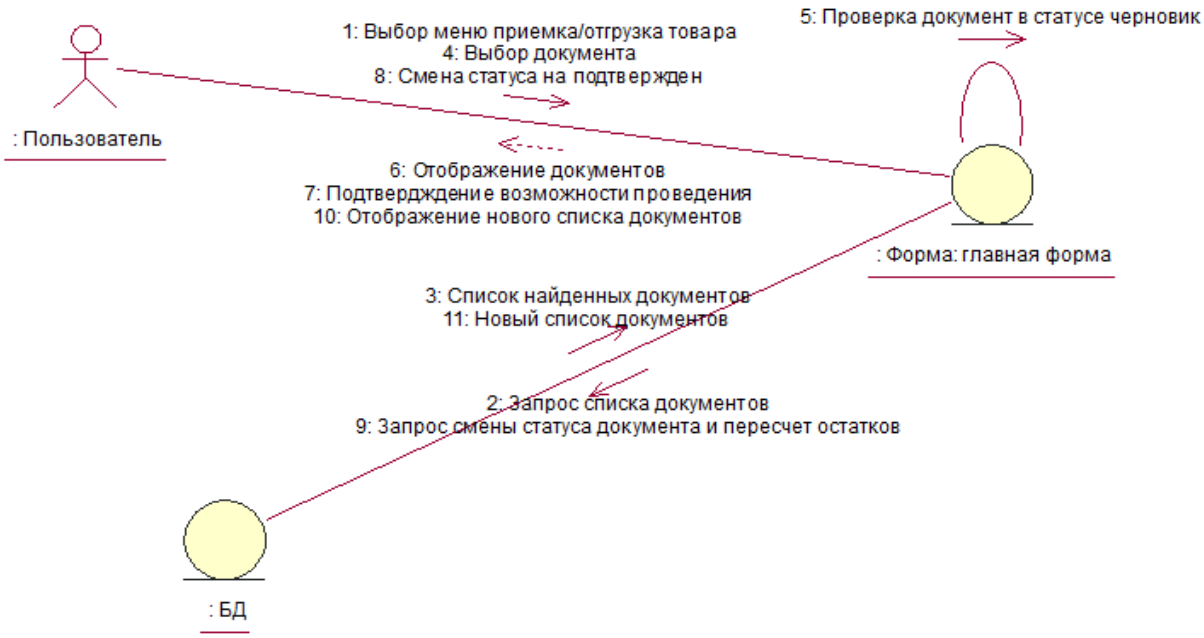


Рис. 27. Диаграмма кооперации действий варианта использования «Проведение документа»

Вариант использования «Отмена проведения документа»:

Главное действующее лицо: администратор.

Контекст использования: администратор отменяет проводку документа.

Результат успешного завершения: система меняет статус документа на «черновик» и пересчитывает остатки на складах.

Сценарий:

1. Выбор меню приемки/отгрузки товара на главной форме;
2. Запрос документов о приемке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных о приемке/ отгрузки на форме;
5. Выбор документа для отмены проведения (документ в статусе «подтверждён»);
6. Проверка статуса документа (в статусе «подтверждён»);
7. Пользователь нажимает отменить проводку документа;
8. Направляется запрос в БД на отмену проведения документа и пересчет остатков;
9. База возвращает информацию об успешной смене статуса;
10. Отображение обновленного списка документов.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 28 и 29.

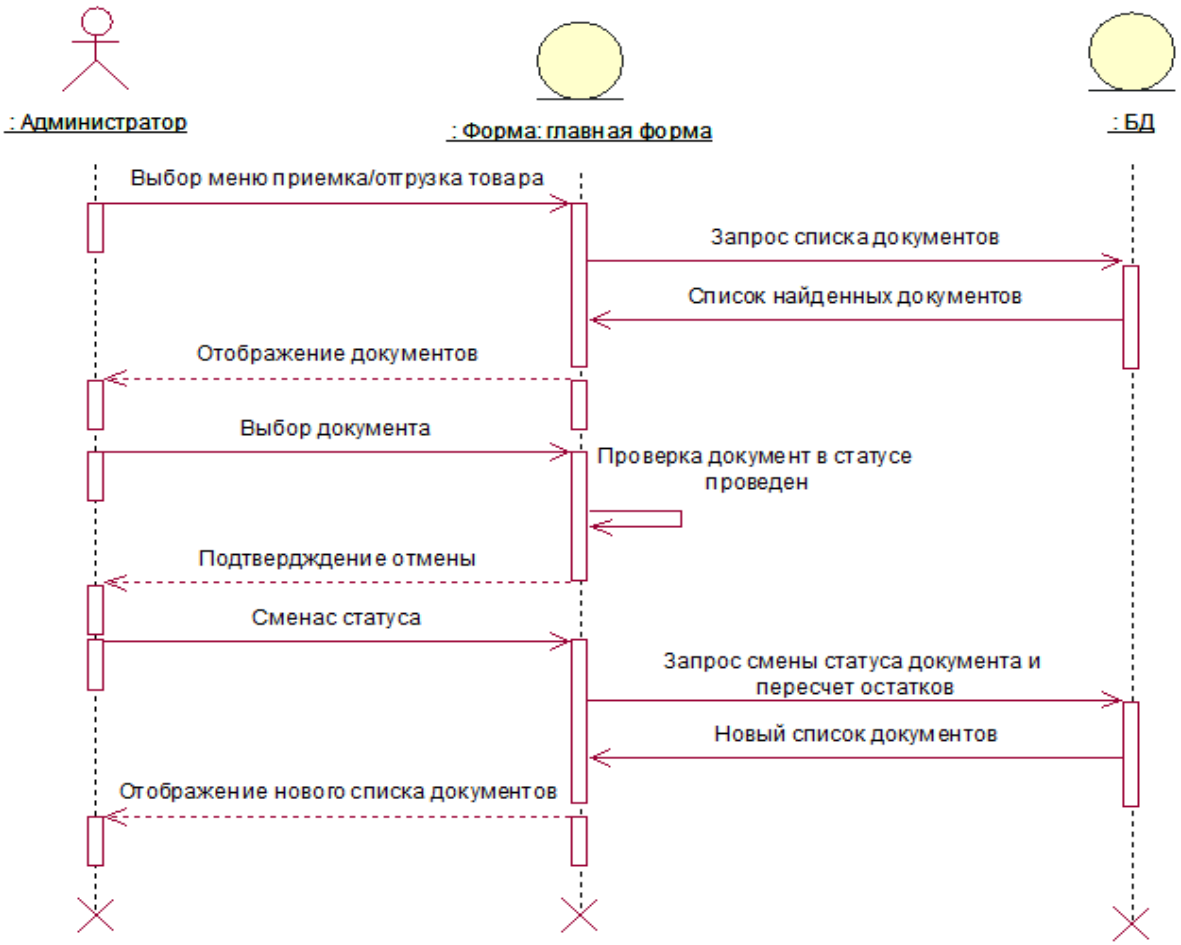


Рис. 28. Диаграмма последовательности действий варианта использования «Отмена проведения документа»

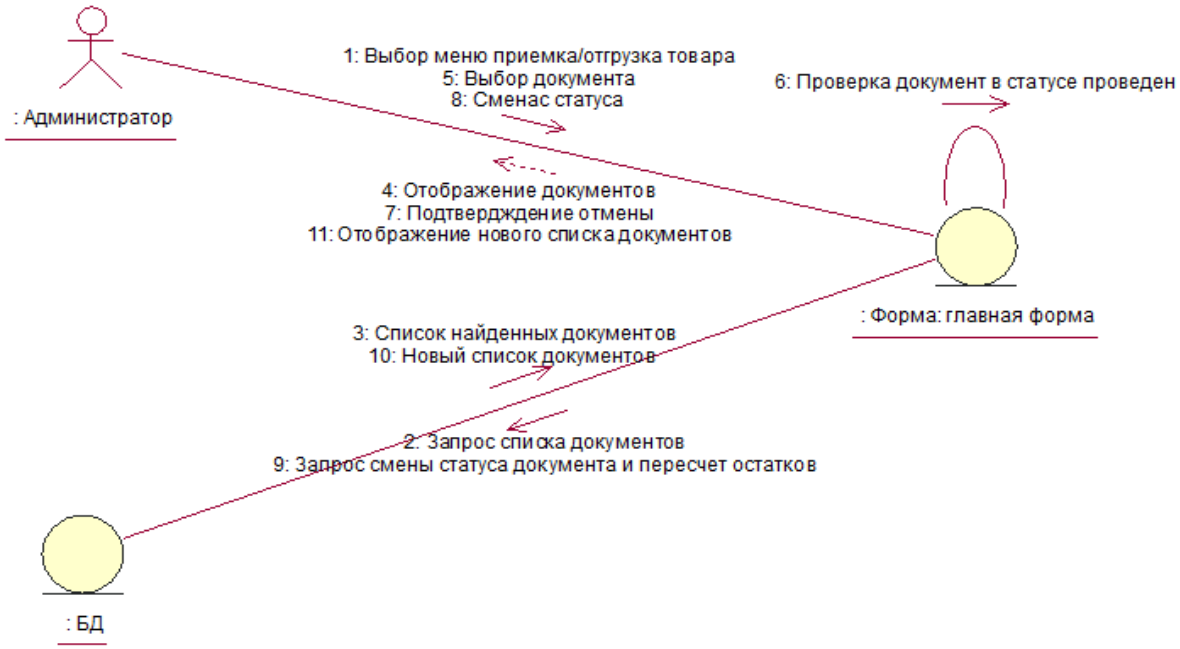


Рис. 29. Диаграмма кооперации действий варианта использования «Отмена проведения документа»

Вариант использования «Формирование отчетов»:

Главное действующее лицо: пользователь.

Контекст использования: пользователь выбирает вид отчета, вводит данные для фильтрации, нажимает сформировать отчет.

Результат успешного завершения: система выгружает данные из БД в Excel формате с форматированием и оформлением.

Сценарий:

1. Выбор меню отчеты – вид отчета;
2. Запуск интерфейса формы с набором фильтров для отчета;
3. Отображение формы на экране пользователя;
4. Ввод данных для фильтрации;
5. Запрос на получение данных из БД;
6. Возврат данных из БД для формирования отчета;
7. Формирование отчета в формате Excel;
8. Отображение файла сформированного отчета.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 30 и 31.

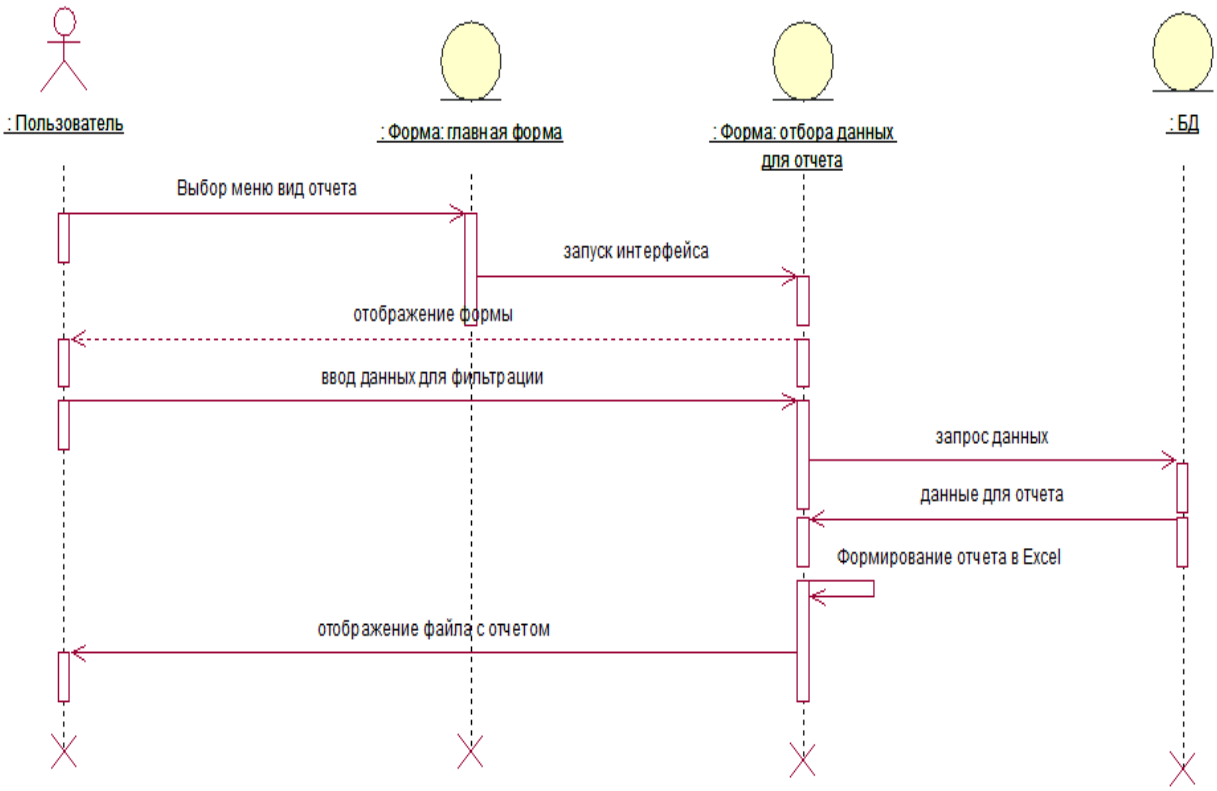


Рис. 30. Диаграмма последовательности действий варианта использования «Формирование отчетов»

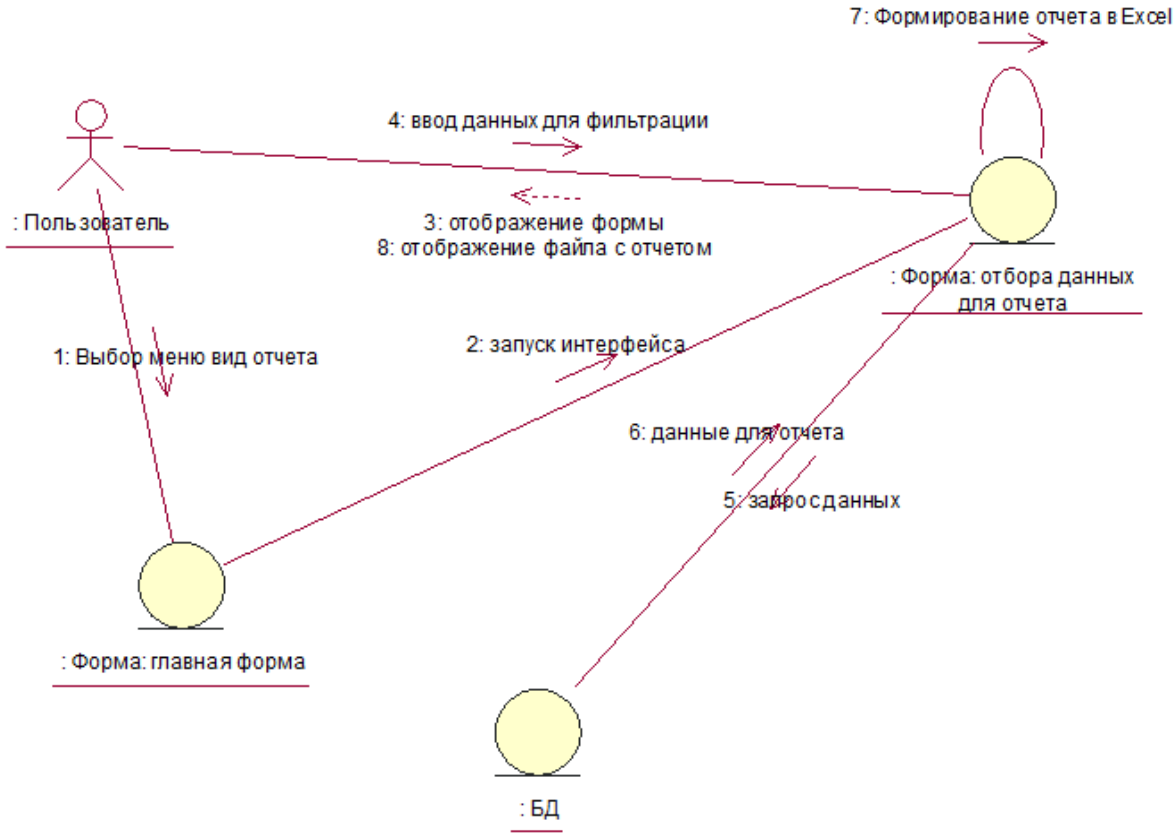


Рис. 31. Диаграмма кооперации действий варианта использования «Формирование отчетов»

Вариант использования «Добавление отгрузки»:

Главное действующее лицо: пользователь.

Контекст использования: пользователь добавлять данные об отгрузке в систему.

Результат успешного завершения: в системе добавляется запись об отгрузке в статусе «черновик».

Сценарий:

1. Выбор меню приемки товара на главной форме;
2. Запрос документов об отгрузке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных об отгрузке на форме;
5. Переход в режим отгрузке товара, путем нажатия на кнопку «Добавить отгрузку»;
6. Запуск интерфейса формы отгрузки;
7. Отображение формы на экране компьютера пользователя;
8. Пользователь вводит данные по отгрузке;
9. Система проверяет заполненность всех необходимых полей;
10. Пользователь нажимает сохранить отгрузку;
11. Данные введённые пользователем направляется через запрос в БД;
12. База возвращает информацию об успешной записи данных;
13. Отображение нового списка документов по отгрузке, с новой строкой в статусе «черновик».

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 32 и 33.

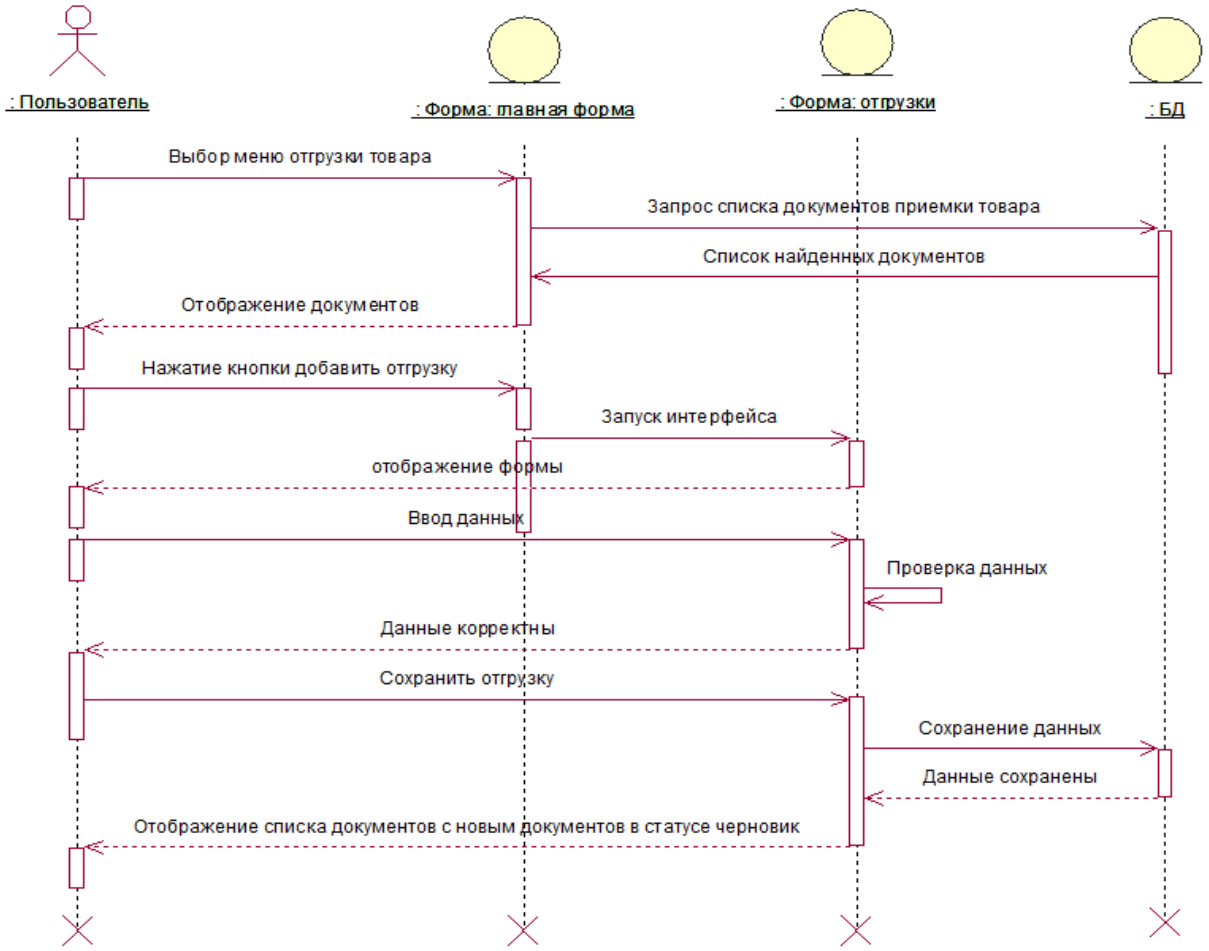


Рис. 32. Диаграмма последовательности действий варианта использования «Добавление отгрузки»

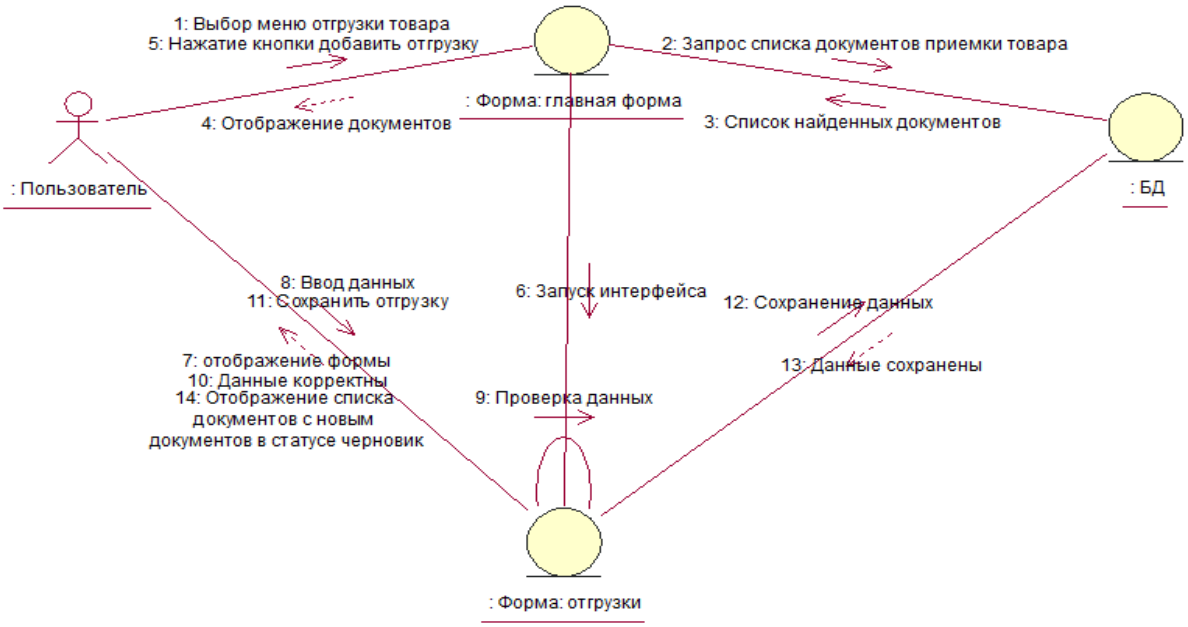


Рис. 33. Диаграмма кооперации действий варианта использования «Добавление отгрузки»

Вариант использования «Изменение отгрузки»:

Главное действующее лицо: пользователь.

Контекст использования: пользователь изменяет данные об отгрузке в системе.

Результат успешного завершения: в системе изменяет запись о приемке в статусе «черновик».

Сценарий:

1. Выбор меню приемки товара на главной форме;
2. Запрос документов об отгрузке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных об отгрузке на форме;
5. Выбор документа для изменения;
6. Переход в режим изменение отгрузки, путем нажатия на кнопку «Изменить отгрузку»;
7. Запуск интерфейса формы отгрузки;
8. Отображение формы на экране компьютера пользователя с данными по отгрузке;
9. Пользователь изменяет данные по приемке;
10. Система проверяет заполненность всех необходимых полей;
11. Пользователь нажимает сохранить отгрузку;
12. Данные введённые пользователем направляется через запрос в БД;
13. База возвращает информацию об успешной записи данных;
14. Отображение списка документов по отгрузке, с измененной строкой в статусе «черновик».

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 34 и 35.

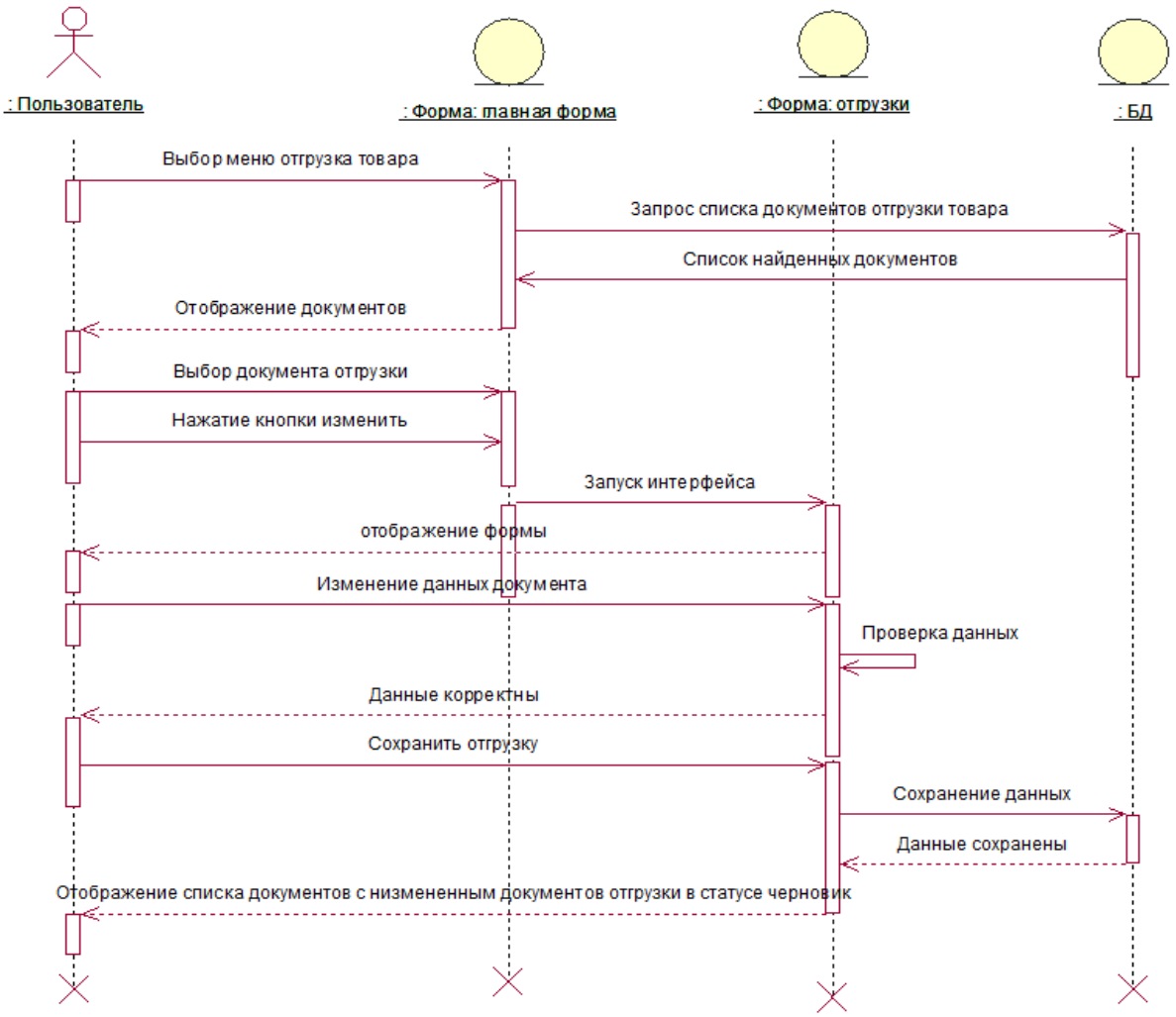


Рис. 34. Диаграмма последовательности действий варианта использования «Изменение отгрузки»

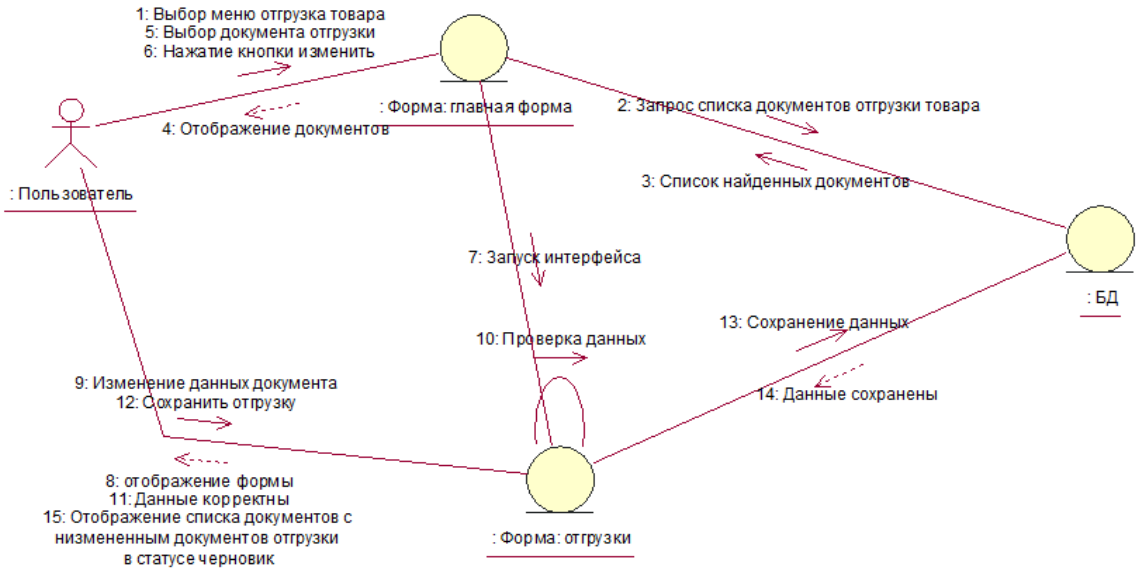


Рис. 35. Диаграмма кооперации действий варианта использования «Изменение отгрузки»

Вариант использования «Удаление отгрузки»:

Главное действующее лицо: администратор.

Контекст использования: администратор удаляет данные об отгрузке в системе.

Результат успешного завершения: в системе удаляет запись об отгрузке в статусе «черновик».

Сценарий:

1. Выбор меню приемки товара на главной форме;
2. Запрос документов об отгрузке товара в БД с указанием склада;
3. Получение списка из БД;
4. Отображение данных об отгрузке на форме;
5. Выбор документа для удаления;
6. Проверка статуса документа (в статусе «черновик»);
7. Пользователь нажимает удалить приемку;
8. Запрос подтверждения удаления;
9. Пользователь подтверждает удаление;
10. Направляется запрос в БД на удаление данных;
11. База возвращает информацию об успешном удалении данных;
12. Отображение списка документов по приемке, без удаленной записи.

Альтернативные сценарии:

Если во время выполнения основного сценария обнаружится ошибка, система выводит сообщение об ошибке и пользователь может вернуться к началу сценария или закончить его выполнении.

Если вариант использования выполнен успешно, список документов обновляется. В противном случае состояние системы не изменяется. Сценарий отображен на рисунках 36 и 37.

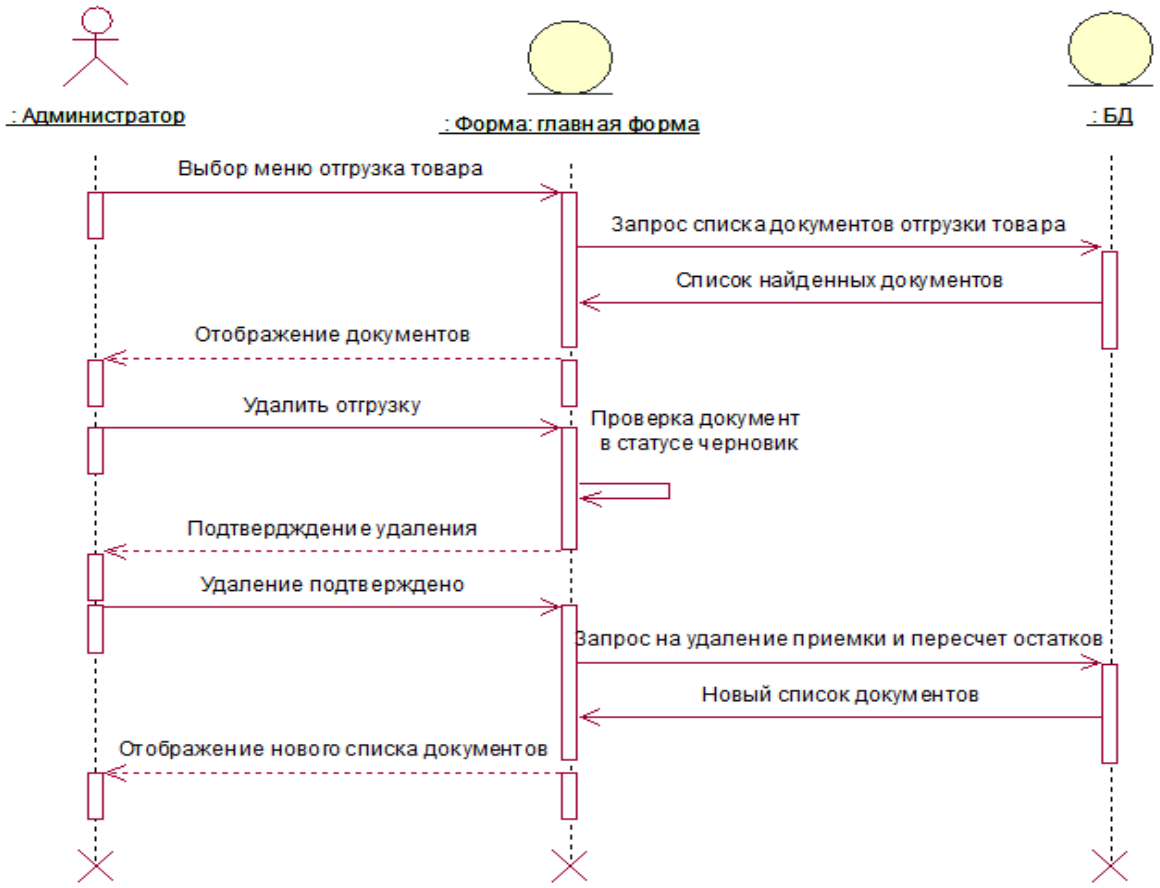


Рис. 36. Диаграмма последовательности действий варианта использования «Удаление отгрузки»

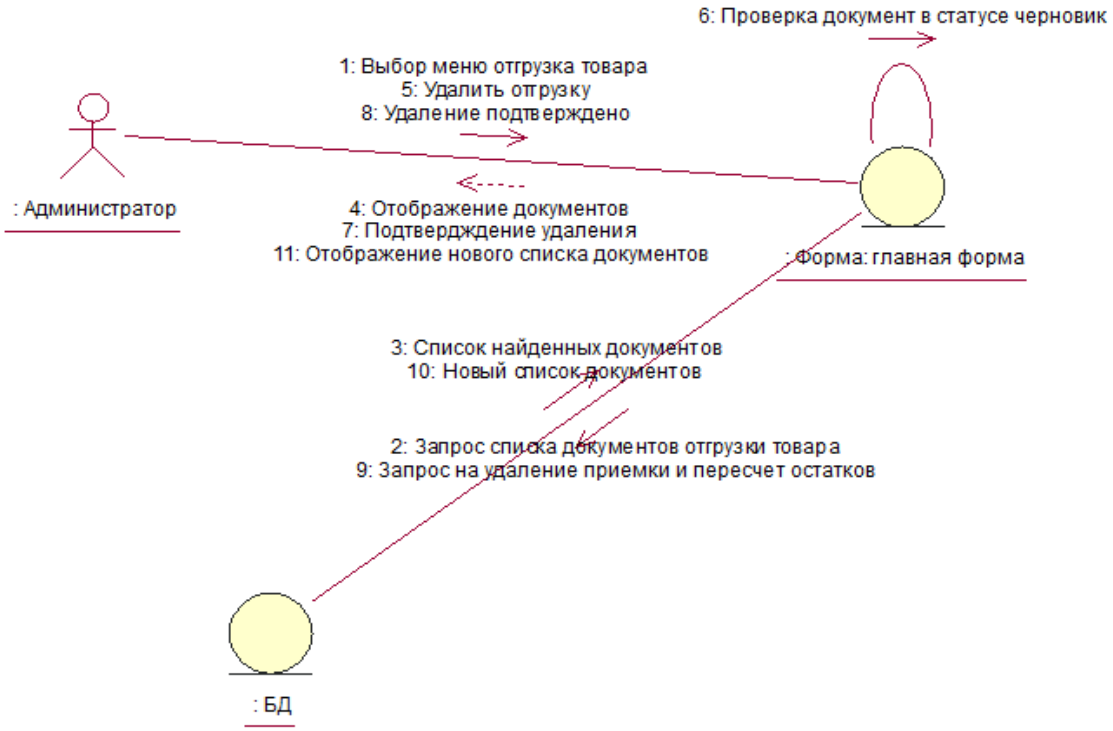


Рис. 37. Диаграмма кооперации действий варианта использования «Удаление отгрузки»

2.3. Диаграммы классов

Диаграммы классов отражают взаимодействие между классами системы. Классы можно рассматривать как типы объектов. На диаграммах классов изображаются также атрибуты классов, операции классов и ограничения, которые накладываются на связи между классами [17].

В потоках событий варианта использования выявляются классы трех типов. Граничные классы (boundary) – служат посредниками при взаимодействии внешних объектов с системой. Как правило, для каждой пары «действующее лицо – вариант использования» определяется один граничный класс. Типы граничных классов: пользовательский интерфейс (обмен информацией с пользователем без деталей интерфейса – кнопок, списков, окон), системный интерфейс и аппаратный интерфейс (используемые протоколы, без деталей их реализации).

Классы – сущности (entity) – представляют собой ключевые абстракции (понятия) разрабатываемой системы. Источники выявления классов – сущностей: ключевые абстракции, созданные в процессе архитектурного анализа, глоссарий, описание потоков событий вариантов использования [7].

Управляющие классы (control) – обеспечивают координацию поведения объектов в системе. Могут отсутствовать в некоторых вариантах использования, ограничивающихся простыми манипуляциями с хранимыми данными. Как правило, для каждого варианта использования определяется один управляющий класс. Примеры управляющих классов: менеджер транзакций, координатор ресурсов, обработчик ошибок [11].

Связь представляет собой семантическую взаимосвязь между классами. Она дает возможность узнавать об атрибутах, операциях и связях другого класса. Иными словами, чтобы один класс мог послать сообщение другому на диаграмме последовательности или взаимодействия, между ними должна существовать связь. На представленной диаграмме классов используется ассоциативная двунаправленная связь. Ассоциация – это семантическая связь между классами. На диаграмме классов ассоциации изображаются в виде линии. Двунаправленность связи отображает прием и передачу сообщений всеми классами реализации [20].

Так, диаграмма классов для варианта использования «Авторизация» выглядит следующим образом (рисунок 38).

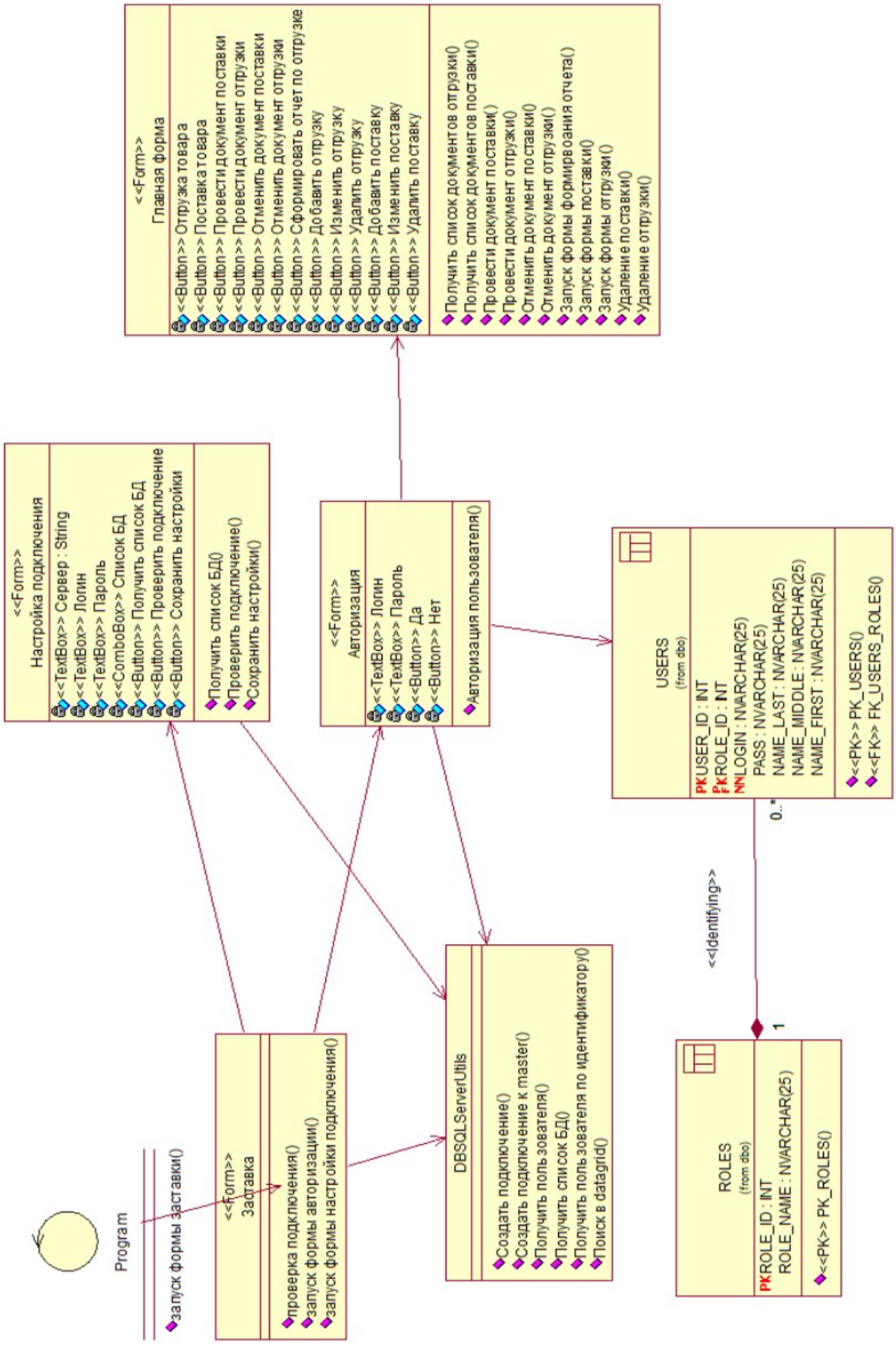


Рис. 38. Диаграмма классов «Авторизация»

Диаграмма классов для диаграммы вариантов использования «Ведение справочников» содержит классы, представленные на рисунке 39.



Рис. 39. Диаграмма классов «Ведение справочников»

Диаграмма классов для диаграммы вариантов использования «Обработка поставки» содержит классы, представленные на рисунке 40.

Граничащие классы: главная форма, поставка, выбор товара.

Классы – сущности: остатки товара на складе, временная таблица для поставки, список поставок, список номенклатуры поставок, список номенклатуры.

Классы-контейнеры (процедуры): процедура обновления сохраненных данных, процедура удаления, пересчета остатков.

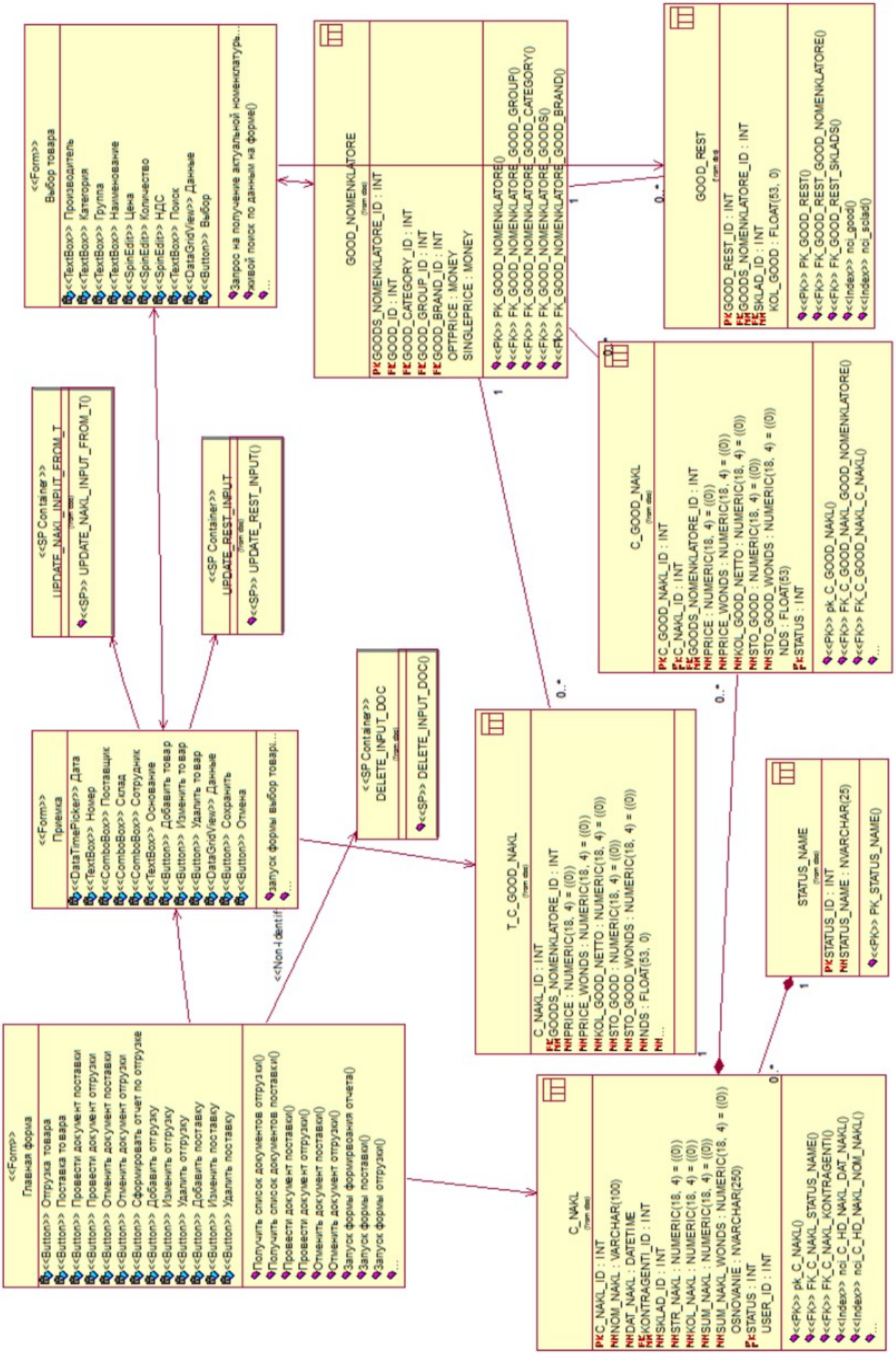


Рис. 40. Диаграмма классов «Обработка поставки»

Диаграмма классов для диаграммы вариантов использования «Обработка отгрузки» содержит классы, представленные на рисунке 41.

Граничащие классы: главная форма, поставка, выбор товара.

Классы – сущности: остатки товара на складе, временная таблица для отгрузки, список отгрузок, список номенклатуры, список номенклатуры.

Классы-контейнеры (процедуры): процедура обновления сохраненных данных, процедура удаления, пересчета остатков.

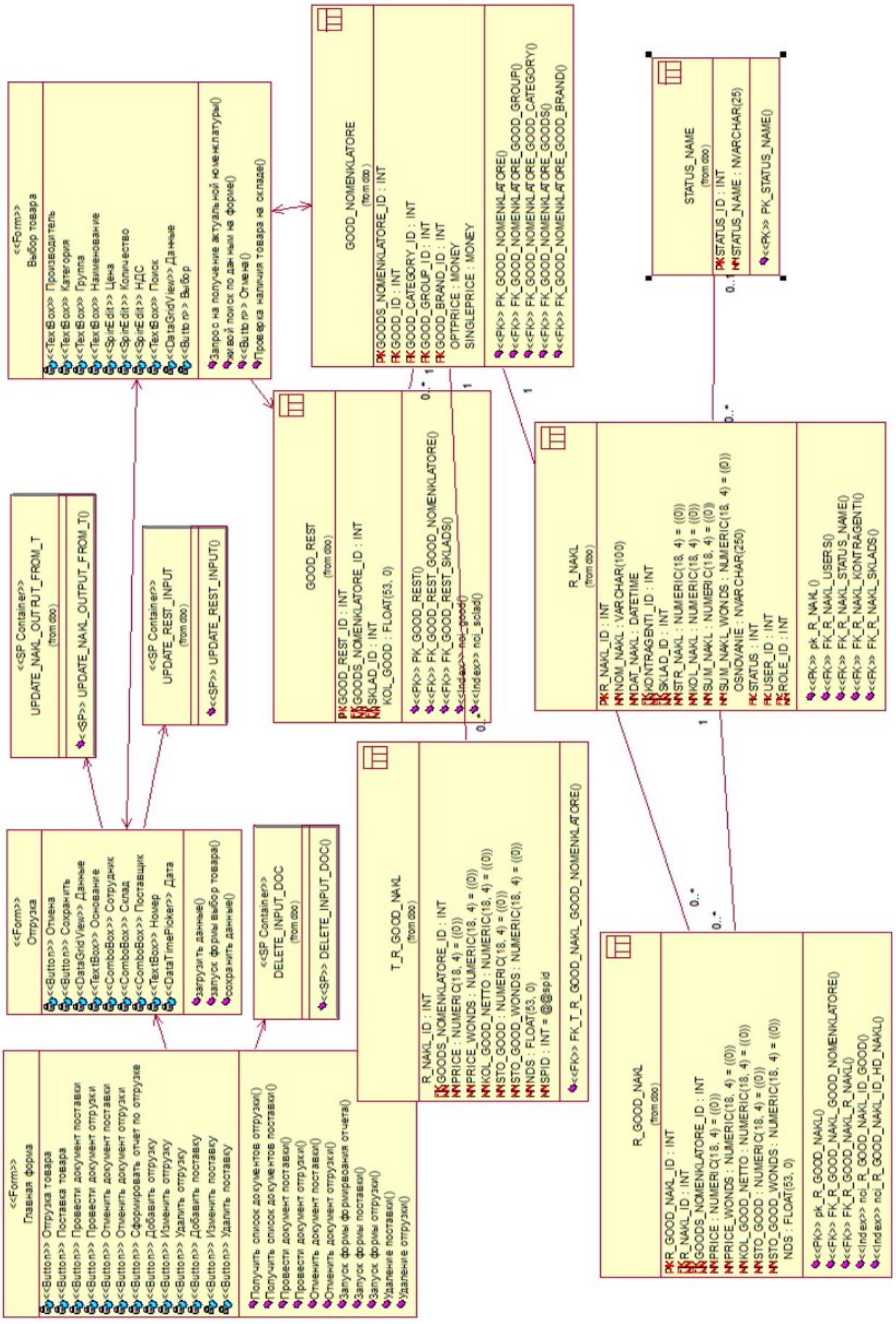


Рис. 41. Диаграмма классов «Обработка отгрузки»

Диаграмма классов для диаграммы вариантов использования «Формирование отчетов» содержит классы, представленные на рисунке 42, так как принцип идентичен для всех видов отчетов, то рассмотрим только процесс.

Граничащие классы: главная форма, отчет об отгрузке.

Классы – сущности: список отгрузок, список товаров в отгрузках, список номенклатуры, список складов, список сотрудников, список контрагентов.

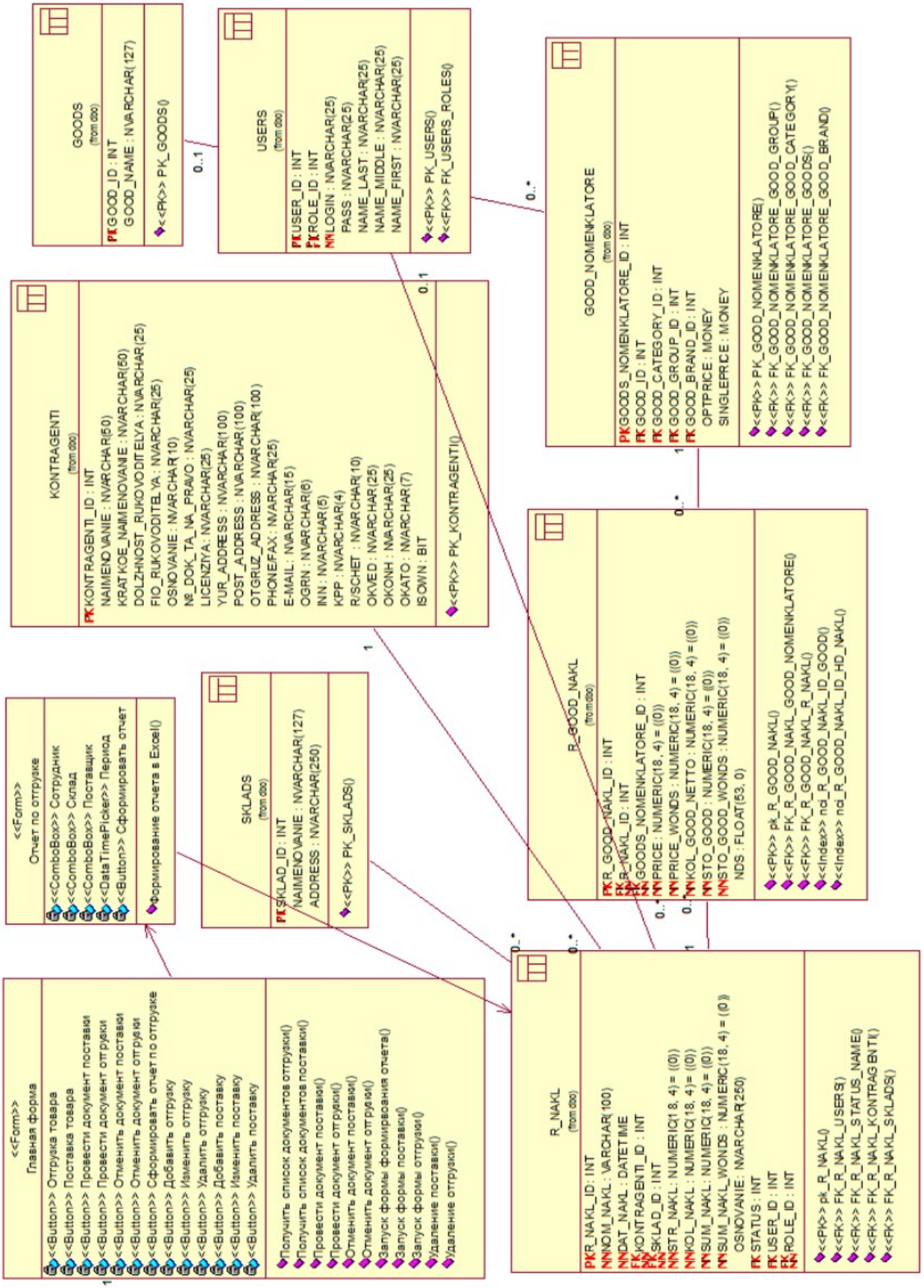


Рис. 42. Диаграмма классов «Формирование отчетов»

2.4. Выбор СУБД

Программное обеспечение включает совокупность программ, реализующих функции и задачи автоматизированных информационных технологий и обеспечивающих устойчивую работу комплексов технических средств. В состав программного обеспечения входят общесистемные и специальные программы, а также инструктивно-методические материалы по применению средств программного обеспечения.

К общесистемному программному обеспечению относятся программы, рассчитанные на широкий круг пользователей и предназначенные для организации вычислительного процесса и решений часто встречающихся задач обработки информации. Они позволяют расширить функциональные возможности ЭВМ, автоматизировать планирование очередности вычислительных работ, осуществлять контроль и управление процессом обработки данных, а также автоматизировать работу программистов. К данному классу ПО можно отнести операционные системы, интегрированные средства управления задачами.

При создании системы помимо программной среды разработки необходимо средство управления базой данных. В настоящее время на рынке программного обеспечения существует большое число программных продуктов, позволяющих создавать эффективные системы управления базами данных.

На сегодняшний день известно более двух десятков форматов данных настольных СУБД, однако наиболее популярными следует признать dBase, Paradox и MS Access, MS SQL Server. Описывать историю возникновения думаю не нужно достаточно привести сравнение их возможностей.

В настоящее время dBase имеет следующие возможности:

- Средства манипуляции данными dBase и FoxPro всех версий;
- Ядро доступа к данным Advantage Database Server фирмы Extended Systems и ODBC-драйвер для доступа к данным этой СУБД;
- Средства визуального построения запросов.

Принцип хранения данных в Paradox сходен с принципами хранения данных в dBase — каждая таблица хранится в своем файле (расширение *.db), MEMO- и BLOB-поля хранятся в отдельном файле (расширение *.md), как и индексы (расширение *.px).

Однако, в отличие от dBase, формат данных Paradox не является открытым, поэтому для доступа к данным этого формата требуются специальные библиотеки.

В состав MS Access входят:

- Средства манипуляции данными Access и данными, доступными через ODBC (последние могут быть «присоединены» к базе данных Access).

- Средства создания форм, отчетов и приложений; при этом отчеты могут быть экспортированы в формат Microsoft Word или Microsoft Excel, а для создания приложений используется Visual Basic for Applications, общий для всех составных частей Microsoft Office.
- Средства доступа к данным серверных СУБД через OLE DB.
- Средства создания клиентских приложений для Microsoft SQL Server.
- Средства администрирования Microsoft SQL Server.

Access – это, прежде всего, система управления базами данных (СУБД). Как и другие продукты этой категории, она предназначена для хранения и поиска данных, представления информации в удобном виде и автоматизации часто повторяющихся операций (таких, как ведение счетов, учет, планирование и т.п.).

В Access в полной мере реализовано управление реляционными базами данных. Система поддерживает первичные и внешние ключи и обеспечивает целостность данных на уровне ядра (что предотвращает несовместимые операции обновления или удаления данных). Кроме того, таблицы в Access снабжены средствами проверки допустимости данных, предотвращающими некорректный ввод вне зависимости от того, как он осуществляется, а каждое поле таблицы имеет свой формат и стандартные описания, что существенно облегчает ввод данных. Access поддерживает все необходимые типы полей, в том числе текстовый, числовой, счетчик, денежный, дата/время, MEMO, логический, гиперссылка и поля объектов OLE. Если в процессе специальной обработки в полях не оказывается никаких значений, система обеспечивает полную поддержку пустых значений.

Access специально спроектирован для создания многопользовательских приложений, где файлы базы данных являются разделяемыми ресурсами в сети.

Система Access поддерживает обработку транзакций с гарантией их целостности. Кроме того, предусмотрена защита на уровне пользователя, что позволяет контролировать доступ к данным отдельных пользователей и целых групп.

Однако система Access обладает важным недостатком в части администрирования и разделения доступа, а также контролем и масштабируемостью.

Microsoft SQL Server – популярная СУБД. Предназначена для создания различных по размеру баз данных, включая крупные базы данных масштаба предприятия. Используется в различных целях и на различных уровнях и помогает управлять приложениями быстро и эффективно. MS SQL Server создан для управления терабайтами данных, способный обрабатывать такие компоненты, как видео, фотографии, бинарные данные, текст, и многое другое. Система обладает всеми преимуществами MS Access и

при этом лишена ее недостатков.

В качестве средства реализации БД была выбрана система MS SQL Server Express.

Microsoft SQL Server Express – это мощная и надежная система управления данными, обеспечивающая множество функций, защиту данных и высокую производительность для внедренных приложений-клиентов, «легких» веб-приложений и локальных хранилищ данных. SQL Server Express предназначен для упрощенного развертывания и быстрого создания прототипов; его можно получить бесплатно и свободно распространять вместе с приложениями.

SQL Server упрощает развёртывание, управление и оптимизацию данных предприятия и аналитических приложений. Как платформа управления данными предприятия, он предоставляет единую консоль управления, которая позволяет администраторам данных, находящимся в любом месте вашей организации, отслеживать, управлять и настраивать все базы данных и связанные службы по всему предприятию. Он предоставляет расширяемую инфраструктуру управления, которая может быть легко запрограммирована при помощи SQL Management Objects (SMO), позволяя пользователям переделывать и расширять их среду управления и независимым поставщикам программных продуктов (ISV) создавать дополнительные инструменты и функциональность для дальнейшего расширения возможностей, поставляемых по умолчанию.

2.5. Построение физической модели базы данных

Методология IDEF1 является профессиональным инструментом проектирования баз данных. Система предназначена для визуального создания и модификации диаграмм «сущность-связь» (ERD – Entity Relationship Diagram). С их помощью определяются важные для предметной области объекты (сущности), их свойства (атрибуты) и отношения друг с другом (связи).

Сущность – реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой предметной области, информация о котором подлежит хранению.

Связь – поименованная ассоциация между двумя сущностями, значимая для рассматриваемой предметной области.

Атрибут – любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения состояния сущности. Атрибут представляет тип характеристик или свойств, ассоциированных с множеством реальных или абстрактных объектов.

Анализируя предметную область, можно выделить следующие сущности:

- Сотрудники/пользователи (USERS);
- Склады (SKLADS);
- Контрагенты (KONTRAGENTI);
- Производители товара (GOOD_BRAND);
- Категории товара (GOOD_CATEGORY);
- Группы товара(GOOD_GROUP);
- Наименования товара (GOODS);
- Номенклатура товара (GOOD_NOMENKLATORE);
- Роли пользователей (ROLES);
- Статусы документов (STATUS_NAME);
- Приходные документы (C_NAKL);
- Список товаров по приходным документам (C_GOOD_NAKL);
- Расходные документы (R_NAKL);
- Список товаров по расходным документам (R_GOOD_NAKL);
- Остатки на складах (GOOD_REST);
- История остатков на складах (GOOD_REST_HISTORY);
- Таблица для временных данных приемки (T_C_GOOD_NAKL);
- Таблица для временных данных отгрузки (T_R_GOOD_NAKL).

Необходимо определить атрибуты сущности и задать связи между атрибутами.

Для сущности «Сотрудники/пользователи» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
USER_ID	Код пользователя	ЦЕЛОЕ ЧИСЛО
ROLE_ID	Код роли	ЦЕЛОЕ ЧИСЛО
LOGIN	Логин	[NVARCHAR](50)
PASS	Пароль	[NVARCHAR](50)
NAME_LAST	Фамилия	[NVARCHAR](50)
NAME_MIDDLE	Отчество	[NVARCHAR](50)
NAME_FIRST	Имя	[NVARCHAR](50)

Для сущности «Склады» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
SKLAD_ID	Код склада	ЦЕЛОЕ ЧИСЛО
NAIMENOVANIE	Наименование	[NVARCHAR](255)
ADDRESS	Адрес	[NVARCHAR](500)

Для сущности «Контрагенты» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
KONTRAGENTI_ID	Код контрагента	[ЦЕЛОЕ ЧИСЛО]
NAIMENOVANIE	Полное наименование	[NVARCHAR](100)
KRATKOE_NAIMENOVANIE	Краткое наименование	[NVARCHAR](100)
DOLZHNOST_RUKOVODITELYA	Должность руководителя	[NVARCHAR](50)
FIO_RUKOVODITELYA	ФИО руководителя	[NVARCHAR](50)
OSNOVANIE	Основание	[NVARCHAR](20)
№_DOK_TA_NA_PRAVO	№ документа на право	[NVARCHAR](50)
LICENZIYA	Лицензия	[NVARCHAR](50)
YUR_ADDRESS	Юридический адрес	[NVARCHAR](200)
POST_ADDRESS	Почтовый адрес	[NVARCHAR](200)
OTGRUZ_ADDRESS	Адрес отгрузки товара	[NVARCHAR](200)
PHONE/FAX	Телефон факс	[NVARCHAR](50)
E-MAIL	Email	[NVARCHAR](30)
OGRN	ОГРН	[NVARCHAR](13)
INN	ИНН	[NVARCHAR](10)
KPP	КПП	[NVARCHAR](9)
R/SCHET	Расчетный счет	[NVARCHAR](20)
OKVED	ОКВЕД	[NVARCHAR](50)
OKONH	ОКОНХ	[NVARCHAR](50)
OKATO	ОКАТО	[NVARCHAR](15)
ISOWN	Признак своей компании	[BIT]

Для сущности «Производители товара» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOOD_BRAND_ID	Код справочника	ЦЕЛОЕ ЧИСЛО
BRAND_NAME	Наименование	[NVARCHAR](255)

Для сущности «Категории товара» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOOD_CATEGORY_ID	Код справочника	ЦЕЛОЕ ЧИСЛО
CATEGORY_NAME	Наименование	[NVARCHAR](255)

Для сущности «Группы товара» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOOD_GROUP_ID	Код справочника	ЦЕЛОЕ ЧИСЛО
GROUP_NAME	Наименование	[NVARCHAR](255)

Для сущности «Наименования товара» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOOD_ID	Код справочника	ЦЕЛОЕ ЧИСЛО
GOOD_NAME	Наименование	[NVARCHAR](255)

Для сущности «Роли пользователей» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
ROLE_ID	Код справочника	ЦЕЛОЕ ЧИСЛО
ROLE_NAME	Наименование	[NVARCHAR](255)

Для сущности «Статусы документов» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
STATUS_ID	Код справочника	ЦЕЛОЕ ЧИСЛО
STATUS_NAME	Наименование	[NVARCHAR](255)

Для сущности «Номенклатура товара» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOODS_NOMENKLATORE_ID	Код номенклатуры	ЦЕЛОЕ ЧИСЛО
GOOD_ID	Код товара	ЦЕЛОЕ ЧИСЛО
GOOD_CATEGORY_ID	Код категории	ЦЕЛОЕ ЧИСЛО
GOOD_GROUP_ID	Код группы	ЦЕЛОЕ ЧИСЛО
GOOD_BRAND_ID	Код производителя	ЦЕЛОЕ ЧИСЛО
OPTPRICE	Цена покупки	MONEY
SINGLEPRICE	Цена продажи	MONEY

Для сущности «Список товаров по приходным документам» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
C_GOOD_NAKL_ID	Код строки в приемке	ЦЕЛОЕ ЧИСЛО
C_NAKL_ID	Код документа приемки	ЦЕЛОЕ ЧИСЛО
GOODS_NOMENKLATORE_ID	Код номенклатуры	ЦЕЛОЕ ЧИСЛО
PRICE	Цена	MONEY
PRICE_WONDS	Цена с НДС	MONEY
KOL_GOOD_NETTO	Кол-во	NUMERIC(18, 4)
STO_GOOD	Сумма	MONEY
STO_GOOD_WONDS	Сумма с НДС	MONEY
NDS	Величина НДС	NUMERIC(18, 4)

Для сущности «Приходные документы» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
C_NAKL_ID	Код документа приемки	ЦЕЛОЕ ЧИСЛО
NOM_NAKL	Номер документа	VARCHAR(100)
DAT_NAKL	Дата	DATETIME
KONTRAGENTI_ID	Код контрагента	ЦЕЛОЕ ЧИСЛО
SKLAD_ID	Код склада	ЦЕЛОЕ ЧИСЛО
STR_NAKL	Строк в приемке	ЦЕЛОЕ ЧИСЛО
KOL_NAKL	Кол-во в приемке	NUMERIC(18, 4)
SUM_NAKL	Общая сумма	MONEY
SUM_NAKL_WONDS	Общая сумма с НДС	MONEY
OSNOVANIE	Основание	NVARCHAR(500)
STATUS	Код статуса документа	ЦЕЛОЕ ЧИСЛО
USER_ID	Код сотрудника	ЦЕЛОЕ ЧИСЛО

Для сущности «Список товаров по расходным документам» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
R_GOOD_NAKL_ID	Код строки в отгрузке	ЦЕЛОЕ ЧИСЛО
R_NAKL_ID	Код документа отгрузки	ЦЕЛОЕ ЧИСЛО
GOODS_NOMENKLATORE_ID	Код номенклатуры	ЦЕЛОЕ ЧИСЛО
PRICE	Цена	MONEY
PRICE_WONDS	Цена с НДС	MONEY
KOL_GOOD_NETTO	Кол-во	NUMERIC(18, 4)
STO_GOOD	Сумма	MONEY
STO_GOOD_WONDS	Сумма с НДС	MONEY
NDS	Величина НДС	NUMERIC(18, 4)

Для сущности «Расходные документы» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
R_NAKL_ID	Код документа отгрузки	ЦЕЛОЕ ЧИСЛО
NOM_NAKL	Номер документа	VARCHAR(100)
DAT_NAKL	Дата	DATETIME
KONTRAGENTI_ID	Код контрагента	ЦЕЛОЕ ЧИСЛО
SKLAD_ID	Код склада	ЦЕЛОЕ ЧИСЛО
STR_NAKL	Строк в приемке	ЦЕЛОЕ ЧИСЛО
KOL_NAKL	Кол-во в приемке	NUMERIC(18, 4)
SUM_NAKL	Общая сумма	MONEY
SUM_NAKL_WONDS	Общая сумма с НДС	MONEY
OSNOVANIE	Основание	NVARCHAR(500)
STATUS	Код статуса документа	ЦЕЛОЕ ЧИСЛО
USER_ID	Код сотрудника	ЦЕЛОЕ ЧИСЛО

Для сущности «Остатки на складах» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOOD_REST_ID	Код строки остатка	ЦЕЛОЕ ЧИСЛО
GOODS_NOMENKLATORE_ID	Код номенклатуры	ЦЕЛОЕ ЧИСЛО
SKLAD_ID	Код склада	ЦЕЛОЕ ЧИСЛО
KOL_GOOD	Количество	[NUMERIC(18, 4)

Для сущности «История остатков на складах» выбраны следующие атрибуты:

Имя Атрибута	Описание	Тип данных
GOOD_REST_HISTORY_ID	Код истории	ЦЕЛОЕ ЧИСЛО
GOOD_REST_ID	Код строки остатка	ЦЕЛОЕ ЧИСЛО
GOODS_NOMENKLATORE_ID	Код номенклатуры	ЦЕЛОЕ ЧИСЛО
SKLAD_ID	Код склада	ЦЕЛОЕ ЧИСЛО
KOL_GOOD	Количество	[NUMERIC(18, 4)
DATE	Дата	DATETIME
USER_ID	Пользователь	ЦЕЛОЕ ЧИСЛО
ACTION	Действие	VARCHAR(20)

Отобразить связь между таблицами можно с помощью диаграммы IDEF1 (рис.43. и рис. 44.).

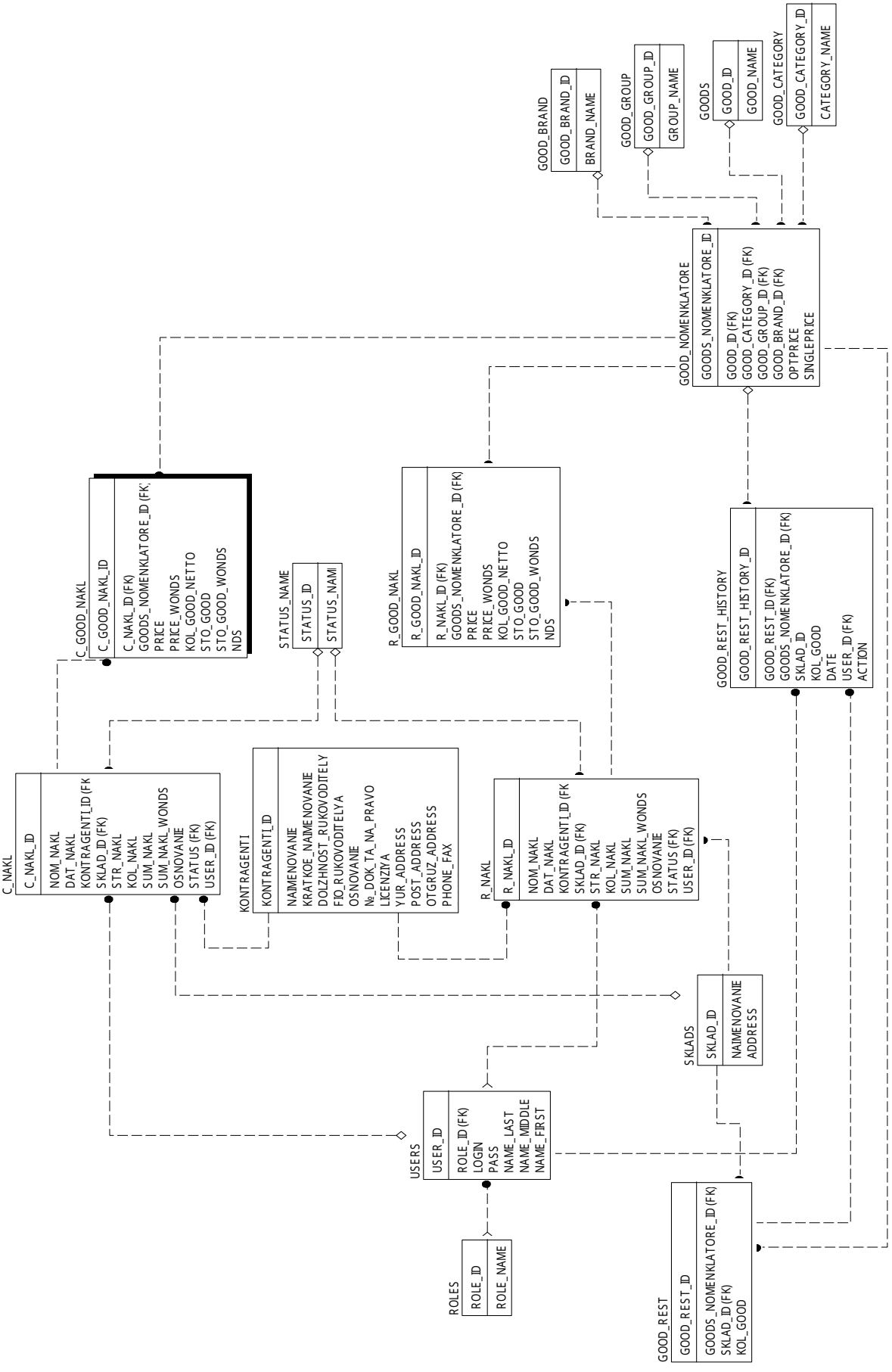


Рис. 43. Физическая модель данных

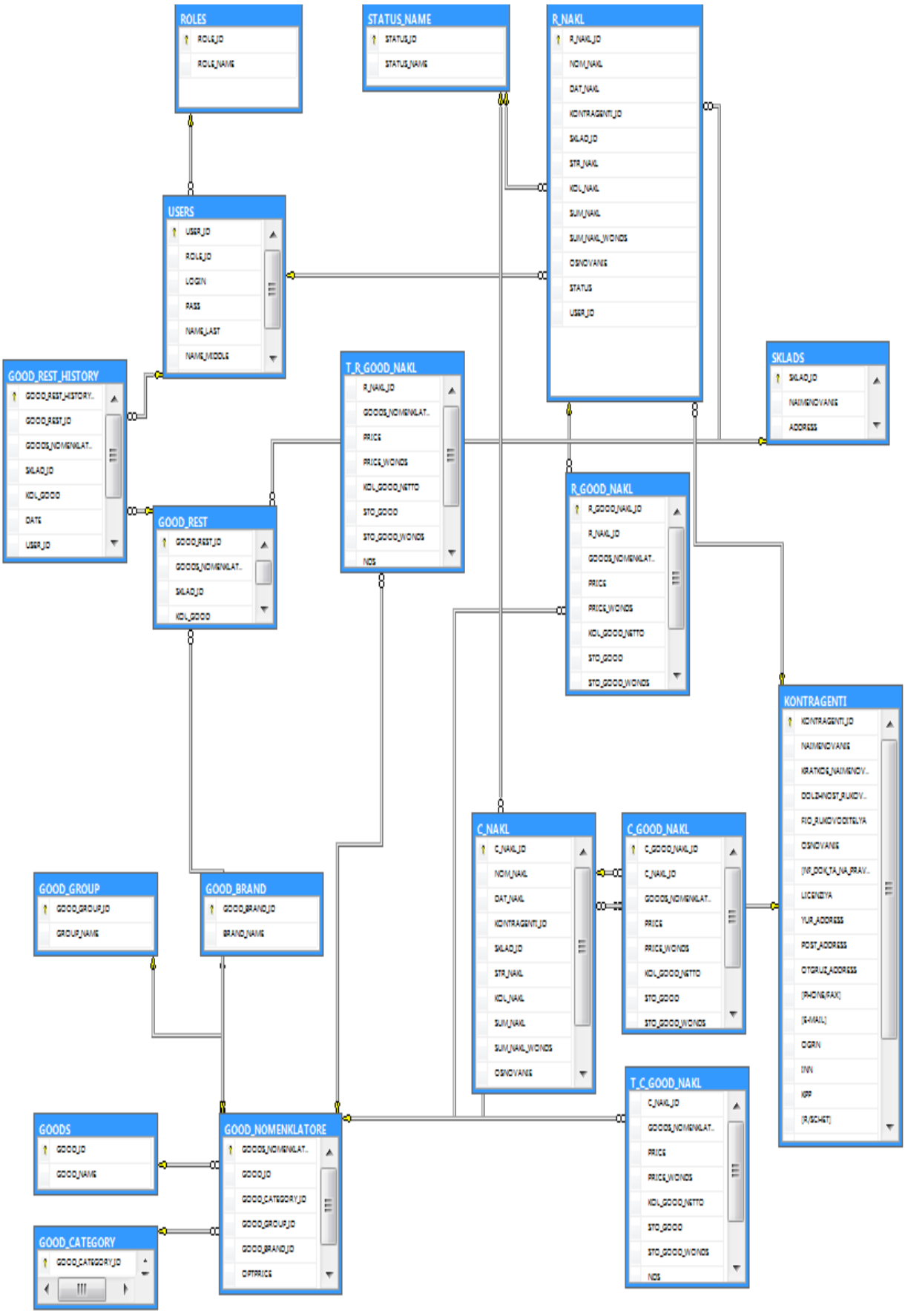


Рис. 44. Схема базы данных в MS SQL Server

2.6. Архитектура доступа к данным

На рисунке 45 представлена архитектура доступа к данным разработанной системой. Как вина на рисунке доступ осуществляется с помощью технологии ADO.NET

ADO.NET относится к промежуточному программному обеспечению, которое внутри каждого приложения ответственно за доступ к базе данных и основано на .NET. Следовательно, использование ADO.NET играет решающую роль в создании эффективных, высокопроизводительных приложений для базы данных.

При создании приложения, которое работает с данными в базе данных, необходимо выполнить такие основные задачи, как определение строк подключения, вставка данных и выполнение хранимых процедур.

Первоначальным компонентом для доступа к данным является SQLConnection оно как раз и отвечает за подключение и пересылку команд от приложения к серверу базы данных.

Для формирования SQL команд используется компонент SQLCommand. После установки подключения мы можем выполнить к базе данных какие-либо команды, например, добавить в базу данных объект, удалить, изменить его или просто извлечь. Команды представлены объектом интерфейса System.Data.IDbCommand. Провайдер для MS SQL предоставляет его реализацию в виде класса SqlCommand. Этот класс инкапсулирует sql-выражение, которое должно быть выполнено. С помощью свойства CommandText устанавливается SQL-выражение, которое будет выполняться. А с помощью свойства Connection можно установить объект подключения SqlConnection.

SqlDataAdapter, Используется в качестве моста между DataSet и SQL Server для извлечения и сохранения данных.

Для связи данных с полученным набором используется и последующим отображением в виде грида использоваться DataTable в связке с DataGridView.

Со стороны сервера базы данных разработана схема базы данных состоящие из сущностей (таблиц) для хранения информации справочной и оперативной а также набора хранимых процедур для обновления нескольких таблиц в процессе работы системы позволяя пользователя вводить данные сразу в несколько сущностей и прерваться на любом этапе.

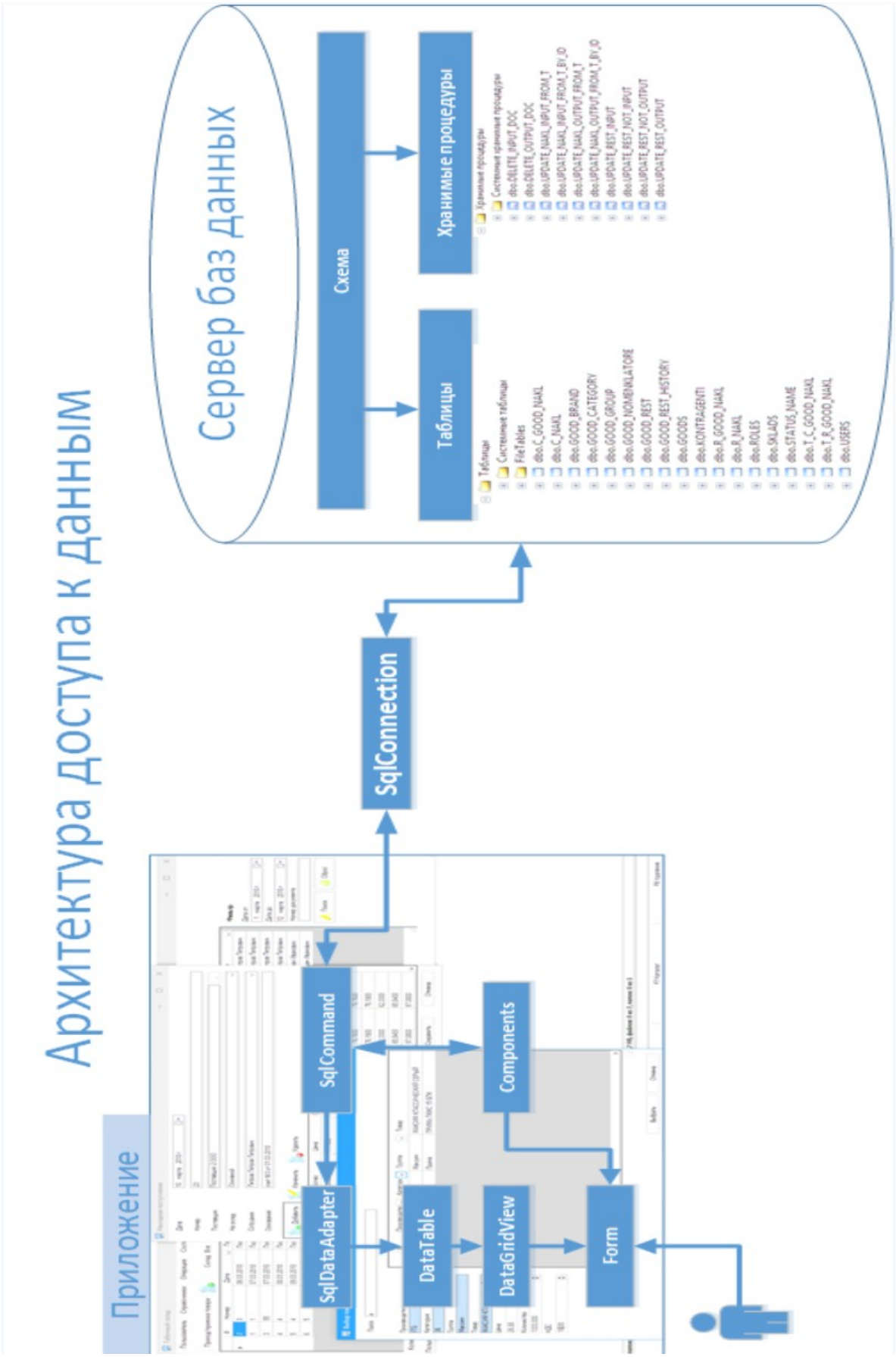


Рис. 44. Архитектура доступа к данным

2.7. Интерфейсы

Интерфейсы обеспечивают взаимодействие между окружающей средой и внутренними элементами системы.

Для обнаружения интерфейсов изучают пары актер/сценарий.

Основные интерфейсы системы:

- Интерфейс пользователя. Включает в себя следующие формы:
  - Форма создания/изменения справочников;
  - Форма создания/изменения приемки;
  - Форма создания/изменения отгрузки;
  - Форма изменения пользователя системы;
  - Форма изменения пароля пользователя;
  - Форма формирования отчета по приемке;
  - Форма формирования отчета по отгрузке;
  - Форма формирования отчета по остаткам;
  - Форма о программе;
  - Форма авторизации;
  - Форма настроек подключения к базе данных;
  - Форма заставка;
  - Главная форма приложения.
- Интерфейс администратора. Включает в себя интерфейс пользователя и дополнительно следующие формы:
  - Форма создания/изменения сотрудников;
  - Активация возможности удаления элементов на форме.

ГЛАВА 3. РЕАЛИЗАЦИЯ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ «ТАБАЧНЫЙ СКЛАД»

3.1. Выбор среды разработки

Для реализации поставленной цели в данной работе использовался язык программирования C#. C# содержит ряд важных новшеств, поднимающих искусство программирования на новую ступень. Например, в состав элементов языка C# включены такие понятия, как делегаты (представители), индексаторы, добавлен синтаксис, поддерживающий атрибуты; упрощено создание компонентов за счёт исключения проблем, связанных с COM; язык C# предлагает средства динамического обнаружения ошибок, обеспечения безопасности и управляемого выполнения программ. Таким образом, C# сочетает первозданную мощь C++ с типовой безопасностью Java, которая обеспечивается наличием механизма контроля типов и корректным использованием шаблонных классов.

В качестве источника данных выбраны базы данных Microsoft SQL Server 2016. MS

SQL Server 2016 Express Edition выбрана в качестве СУБД для разрабатываемого программного обеспечения по ряду причин:

- Предварительный опыт работы с СУБД MS SQL Server;
- Широта распространения;
- Возможности, предоставляемые СУБД разработчику;
- Простота организации взаимодействия с ней в используемой среде разработки MS Visual Studio 2017.

3.2. Описание программы

При запуске программы система проверяет доступ к базе данных. Если таковой отсутствует, то программа предложит, произведи настройку подключения (рис. 45.).

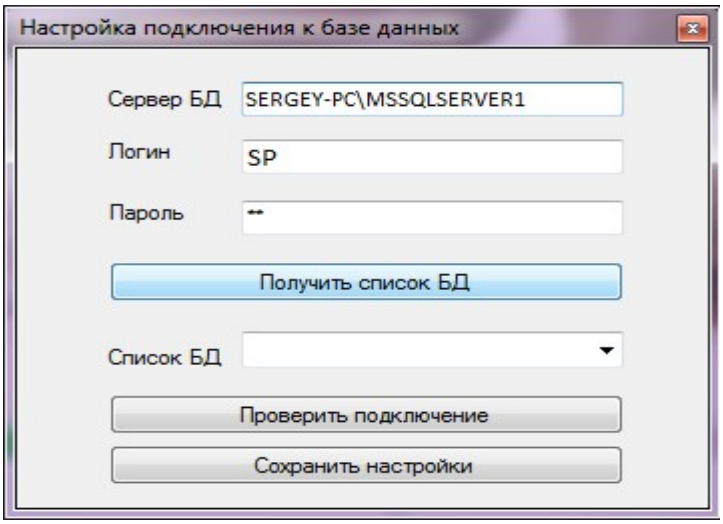


Рис. 45. Настройка подключения

Если подключение проходит успешно то программа предлагает авторизацию пользователю. где нужно ввести логин и пароль (рис.46.).

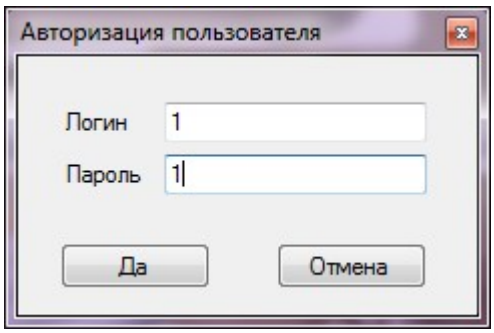


Рис. 46. Авторизация пользователя

Если пользователь нажмет отмену или закрое окно то программа предложит ему подтвердить свое действие (рис. 47.).

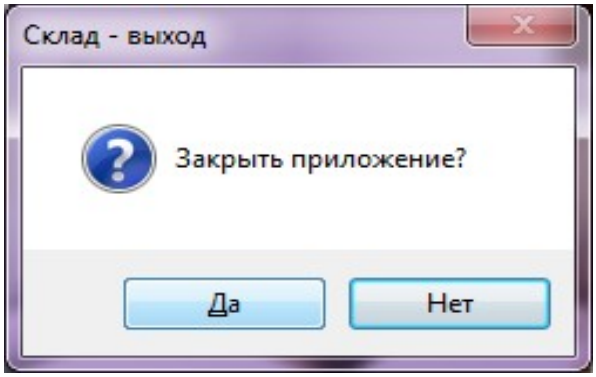


Рис. 47. Закрытие приложения

При неверно вводе паре логина и пароля – программа выдаст предупреждение:

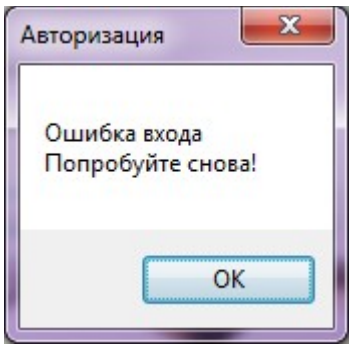
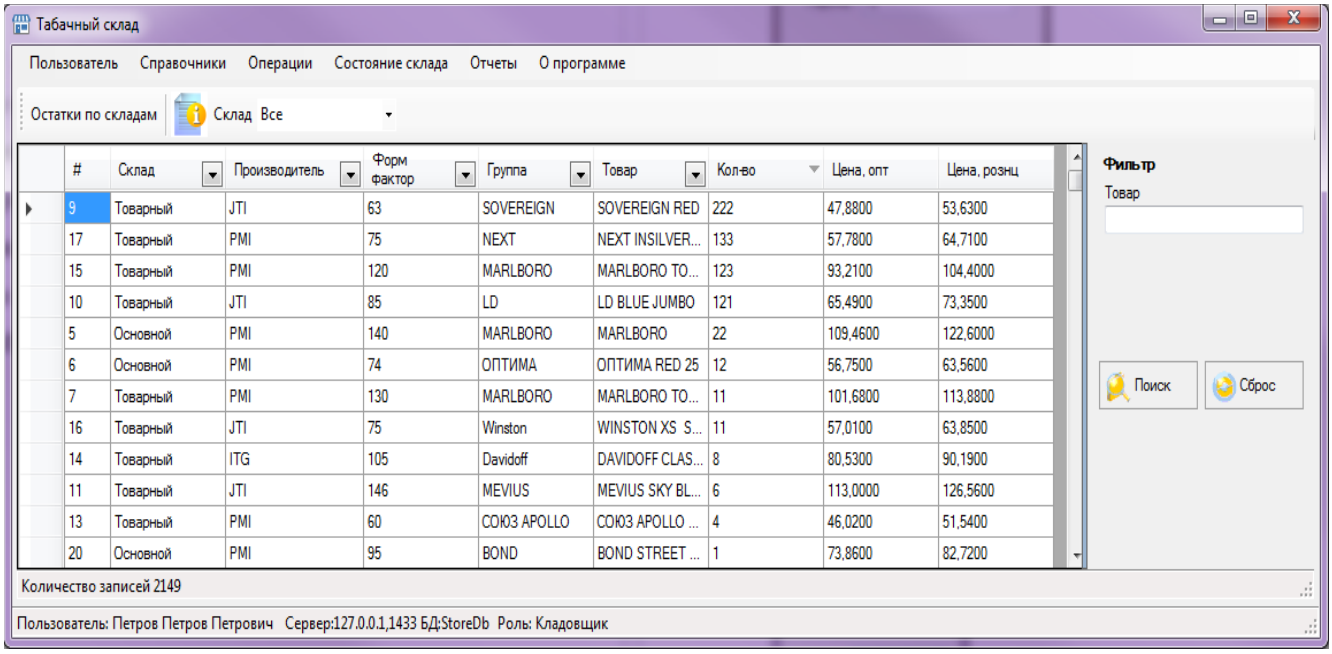


Рис. 48. Неверные логин/пароль

После авторизации отображается главное окно программы, где пользователю сразу представлены текущие остатки и доступ главному меню программы (рис. 49.).



#	Склад	Производитель	Форм фактор	Группа	Товар	Кол-во	Цена, опт	Цена, розн
9	Товарный	JTI	63	SOVEREIGN	SOVEREIGN RED	222	47,8800	53,6300
17	Товарный	PMI	75	NEXT	NEXT INSILVER...	133	57,7800	64,7100
15	Товарный	PMI	120	MARLBORO	MARLBORO TO...	123	93,2100	104,4000
10	Товарный	JTI	85	LD	LD BLUE JUMBO	121	65,4900	73,3500
5	Основной	PMI	140	MARLBORO	MARLBORO	22	109,4600	122,6000
6	Основной	PMI	74	ОПТИМА	ОПТИМА RED 25	12	56,7500	63,5600
7	Товарный	PMI	130	MARLBORO	MARLBORO TO...	11	101,6800	113,8800
16	Товарный	JTI	75	Winston	WINSTON XS S...	11	57,0100	63,8500
14	Товарный	ITG	105	Davidoff	DAVIDOFF CLAS...	8	80,5300	90,1900
11	Товарный	JTI	146	MEVIUS	MEVIUS SKY BL...	6	113,0000	126,5600
13	Товарный	PMI	60	СОЮЗ APOLLO	СОЮЗ APOLLO ...	4	46,0200	51,5400
20	Основной	PMI	95	BOND	BOND STREET ...	1	73,8600	82,7200

Рис. 49. Главное окно программы

Меню содержит следующие пункты:

- Пользователь:
  - Сменить пользователя;
  - Сменить пароль;
  - Выход из программы.
- Справочники:
  - Товары;
  - Товарные группы;
  - Товарные категории;

- Производители;
- Номенклатура товаров;
- Контрагенты;
- Склады;
- Сотрудники/пользователи.

- Операции:
- Приемка товара;
- Выдача товара (отгрузка).

- Состояние склада:
- Остатки.

- Отчеты:
- Отчеты о приходе;
- Отчеты о расходе;
- Отчеты об остатках.

Выбрав пункт меню, смена пароля пользователь может изменить свой текущий пароль на любой другой:

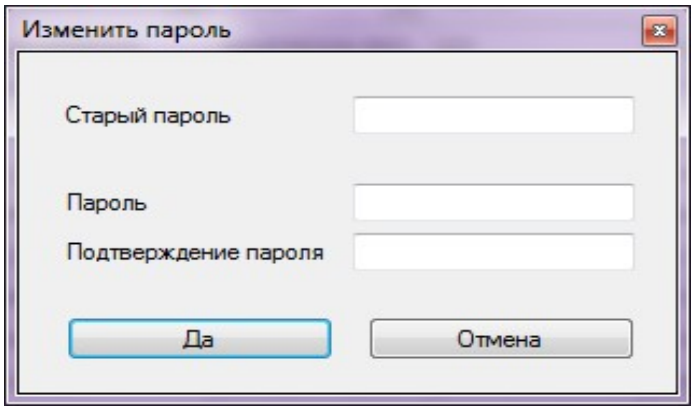


Рис. 50. Форма изменения пароля

Для смены текущего пользователя на другого существует меню сменить пользователя:

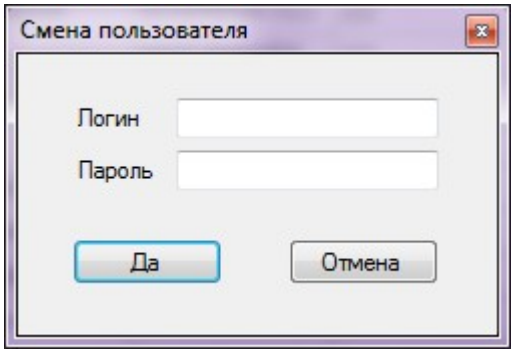


Рис. 51. Форма изменения пользователя

При работе со справочниками выбирается пункты меню и заполняются необходимыми данными пользователю доступно только добавление и изменение, администратору доступно удаление, также только администратору доступны справочники «Наша организация», «Сотрудники» (см рис. 64-70). Все формы походи имеют поля для ввода и список уже веденных данных. При выборе элемента из таблицы поля формы автоматически заполнятся выбранным элементом – что упрощает процесс изменения записи или добавления новой на основе выбранной.

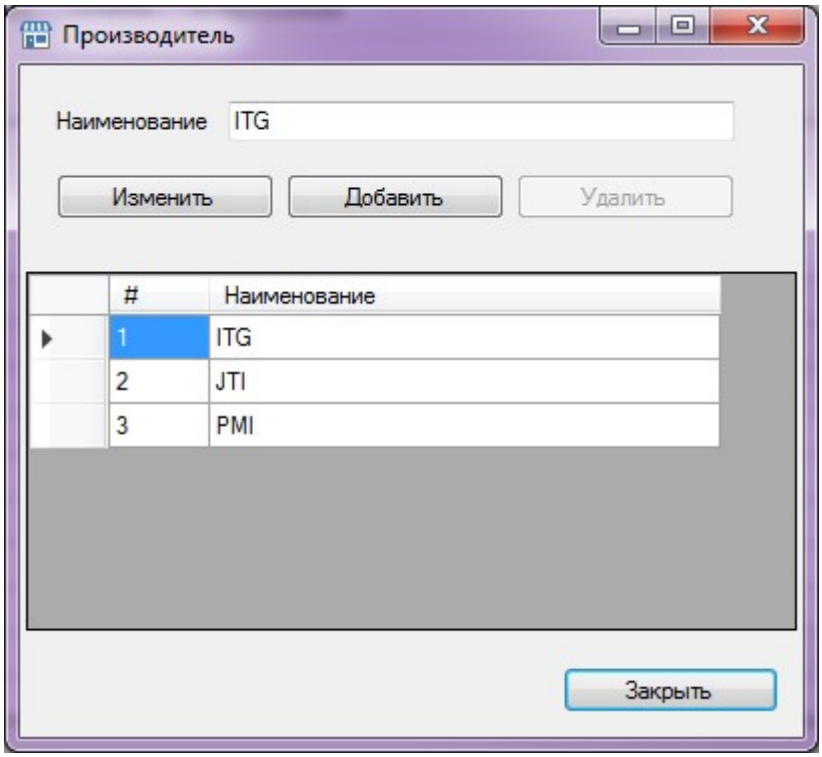


Рис. 52. Форма справочника производитель

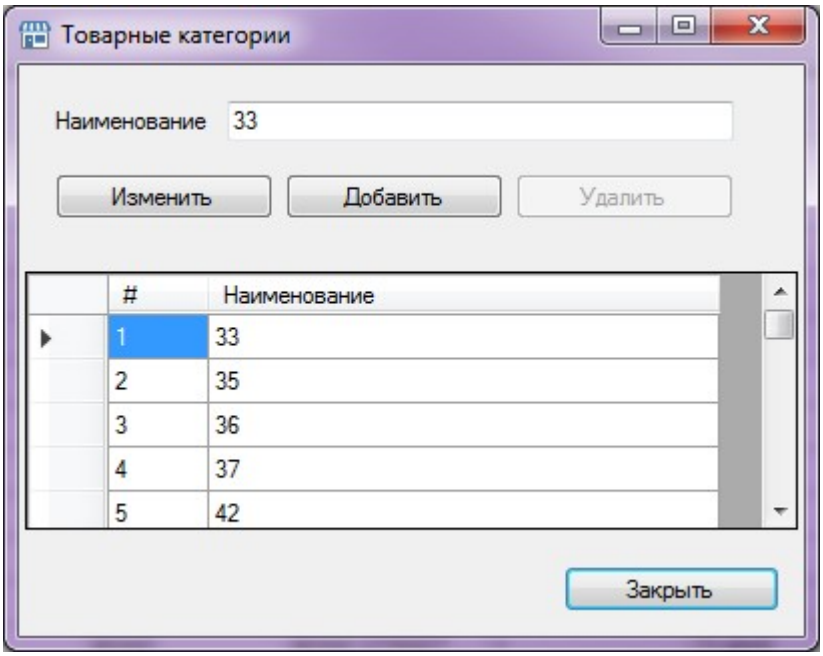


Рис. 53. Форма справочника «Товарные категории»

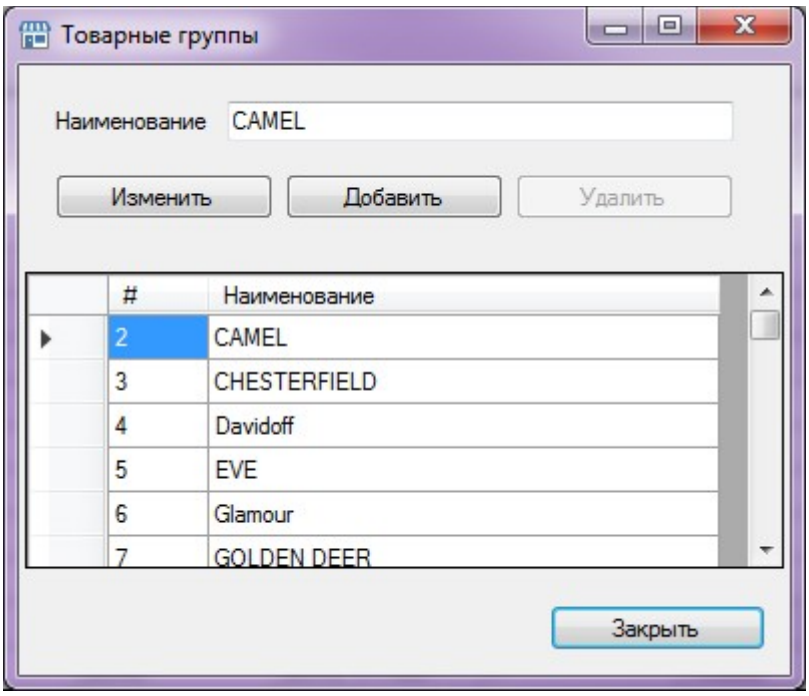


Рис. 54. Форма справочника «Товарные группы»

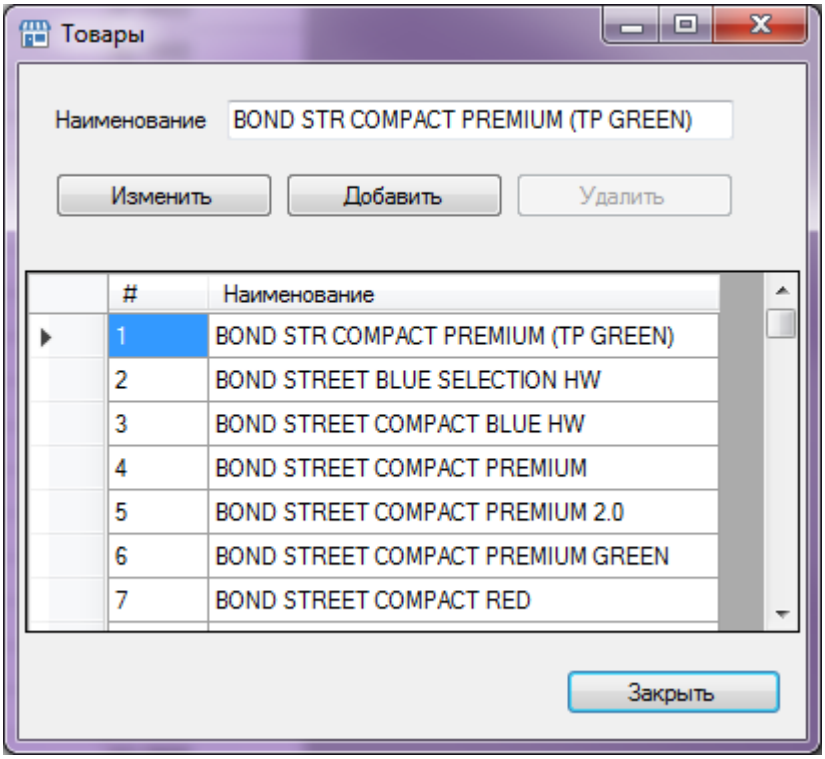


Рис. 55. Форма справочника «Товары»

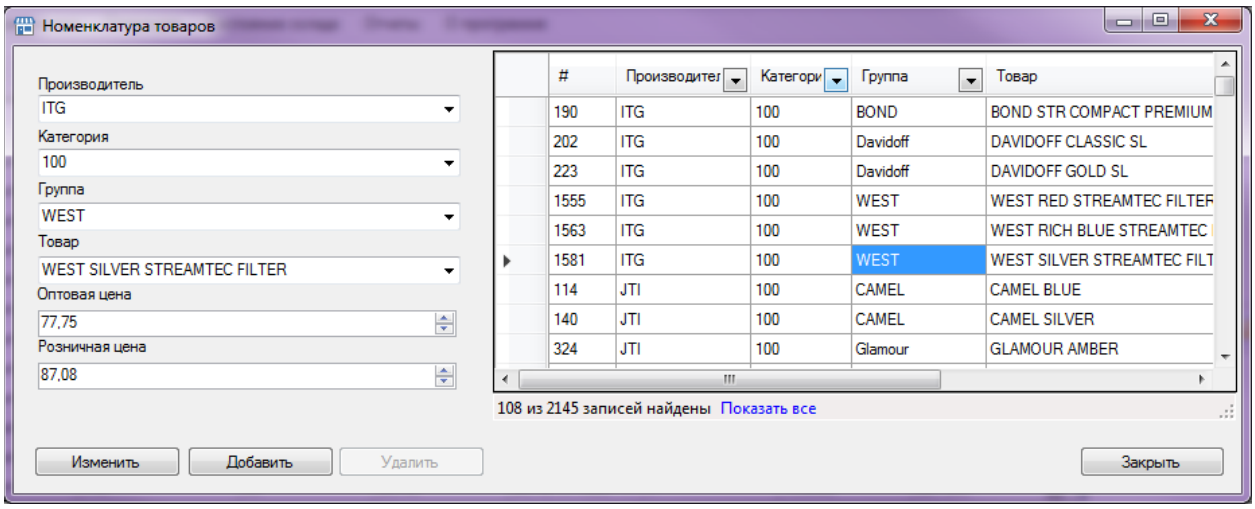


Рис. 56. Форма справочника «Номенклатура товаров»

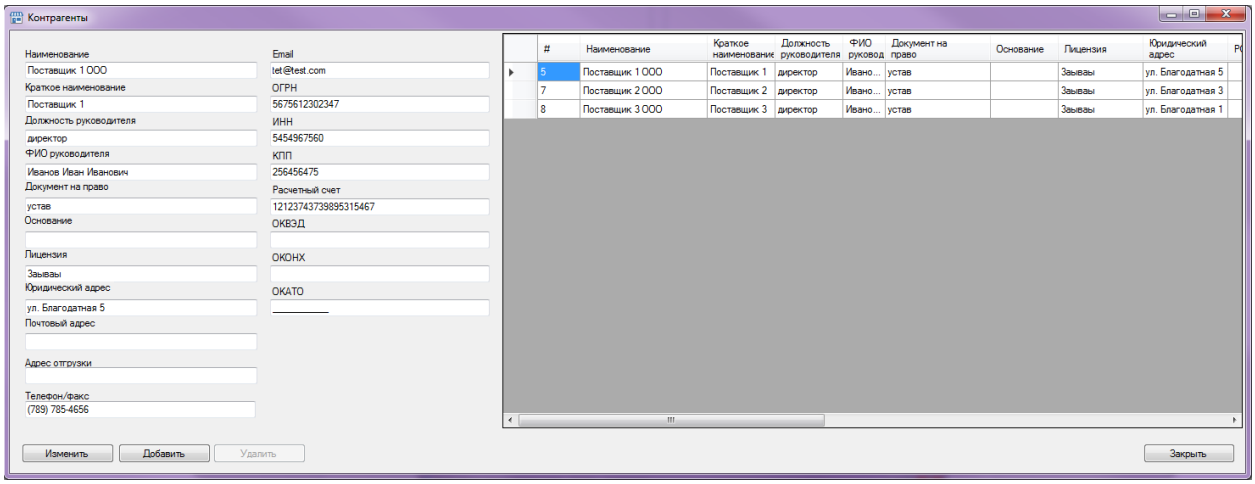


Рис. 57. Форма справочника «Контрагенты»

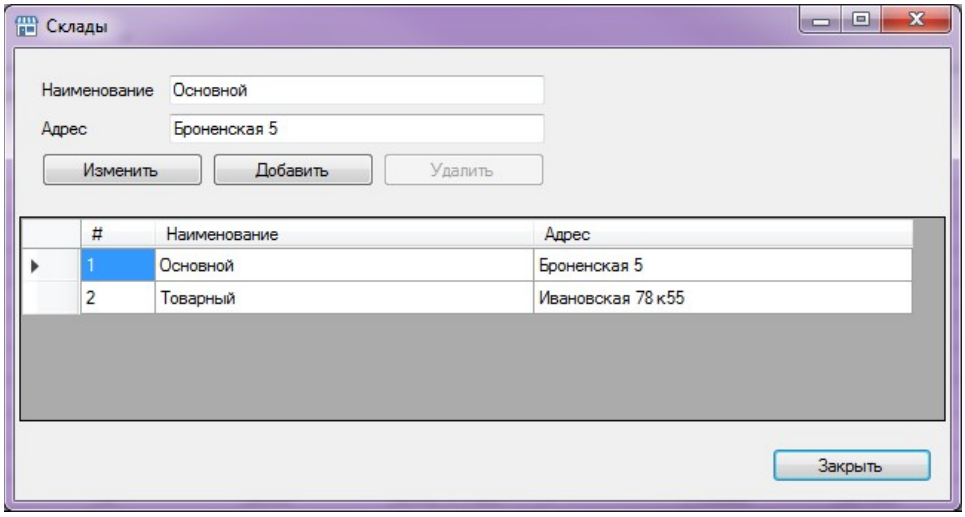


Рис. 58. Форма справочника «Склады»

При приемке товара пользователь может добавлять приемку или менять существующею, если она в статусе «Черновик» - это документ, который создан, но еще не проведен в базе и не учитывается в остатках. На форме предусмотрен фильтр по дате и номере документа, а также автофильтр на табличной форме для удобства навигации (Рис. 59).

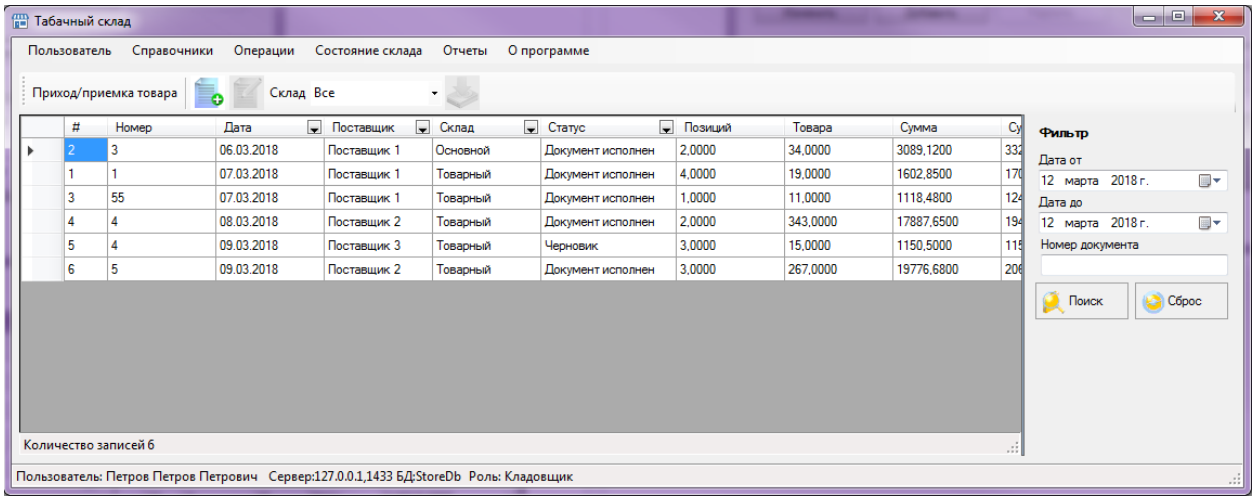


Рис. 59. Приемка товара

При вводе новой приемке или изменения отображается форма ввода приемки:

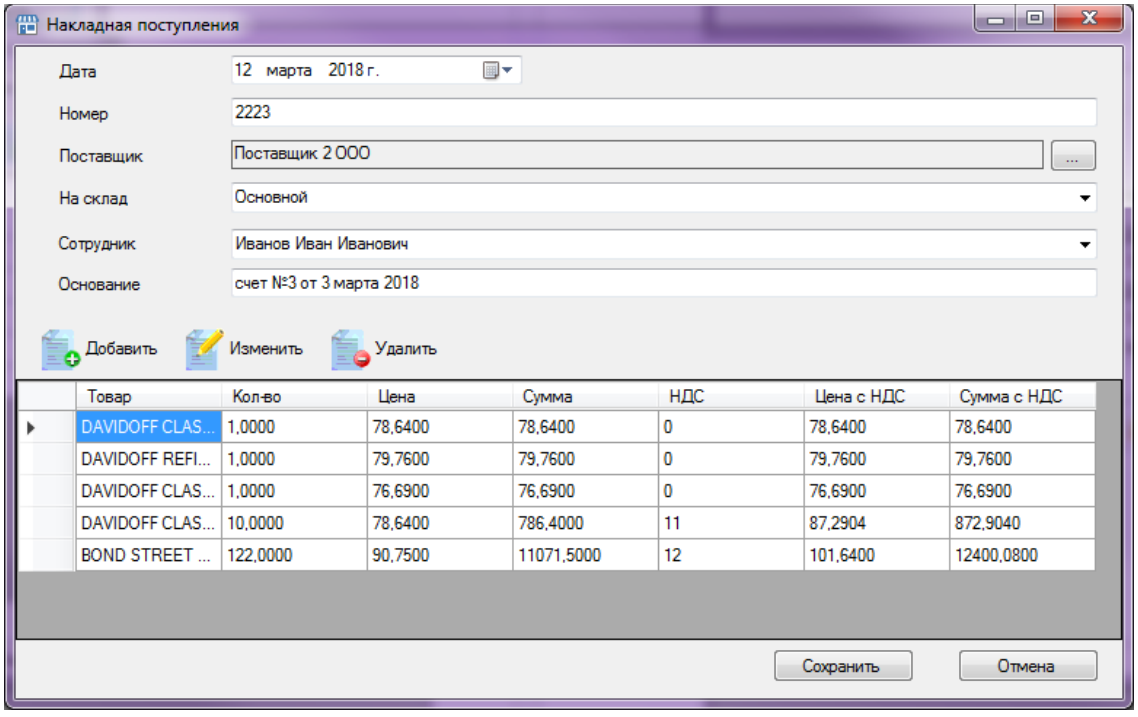


Рис. 60. Ввод приемки товара

На форме имеется поля для ввода заголовочной части документа и табличной части для ввода данных о товарах. При вводе имеется ограничение – нельзя добавить товара/контрагента, которого нет в справочнике. При указании товара максимальная цена приемки ограничена ценой в справочнике.

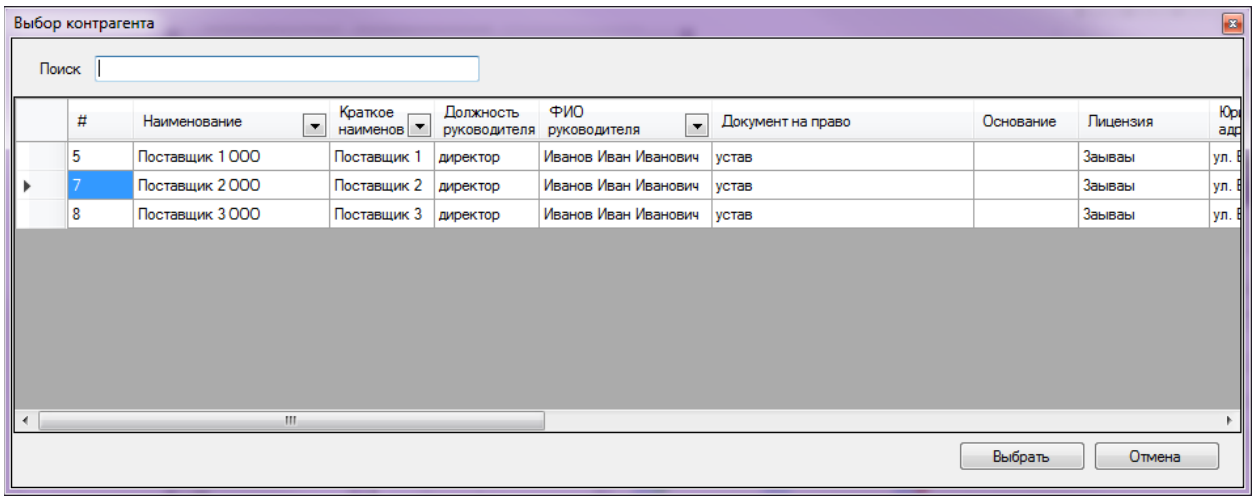


Рис. 61. Форма выбора контрагента

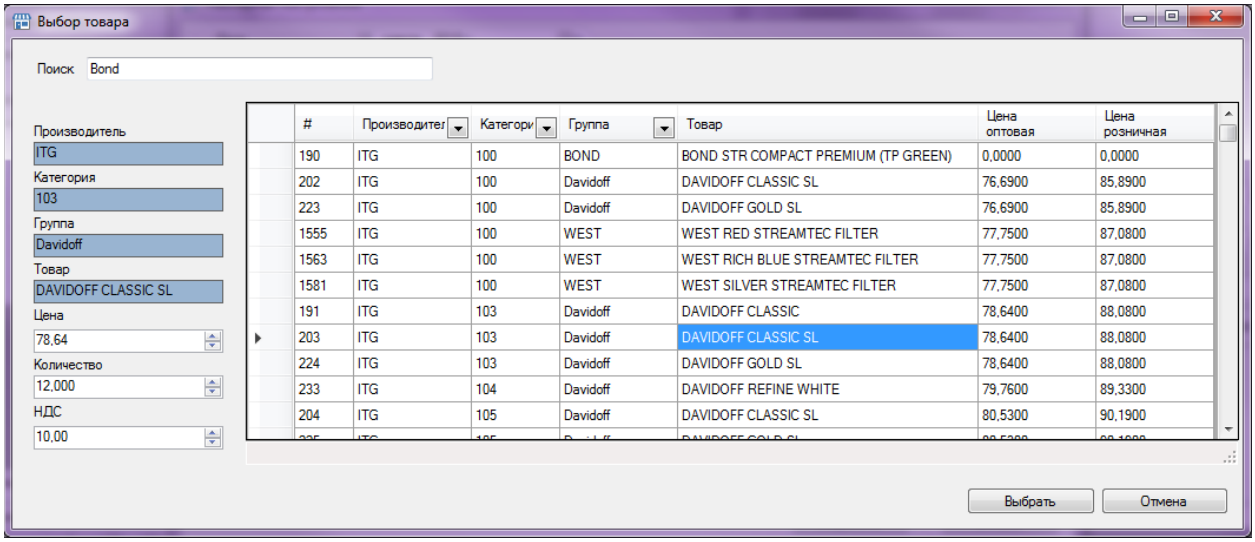


Рис. 62. Форма выбора товара

После приемки накладную необходимо перевести в статус, документ исполнен, для этого существует кнопка с желтой стрелкой вниз на главной форме – при нажатии на нее документ проводится и делается учет поставки.

Отгрузка товара происходит абсолютно также только ограничения на водимый товар – цена не моет быть ниже цены продажи по справочнику. При указании товара в накладную делается проверка по остатку на выбранном складе. Если остаток меньше запрашиваемого – будет показано соответствующее предупреждение и операция будет прервана. Если остаток снизился ниже 20000 единиц – предупреждение будет информирующим и позволит продолжить работу. Для подтверждения документа в режиме отгрузки на главной форме нужно нажать кнопку с желтой стрелкой вверх. На всех кнопках на главной форме существуют подсказки, которые появляются при наведении на них курсора.

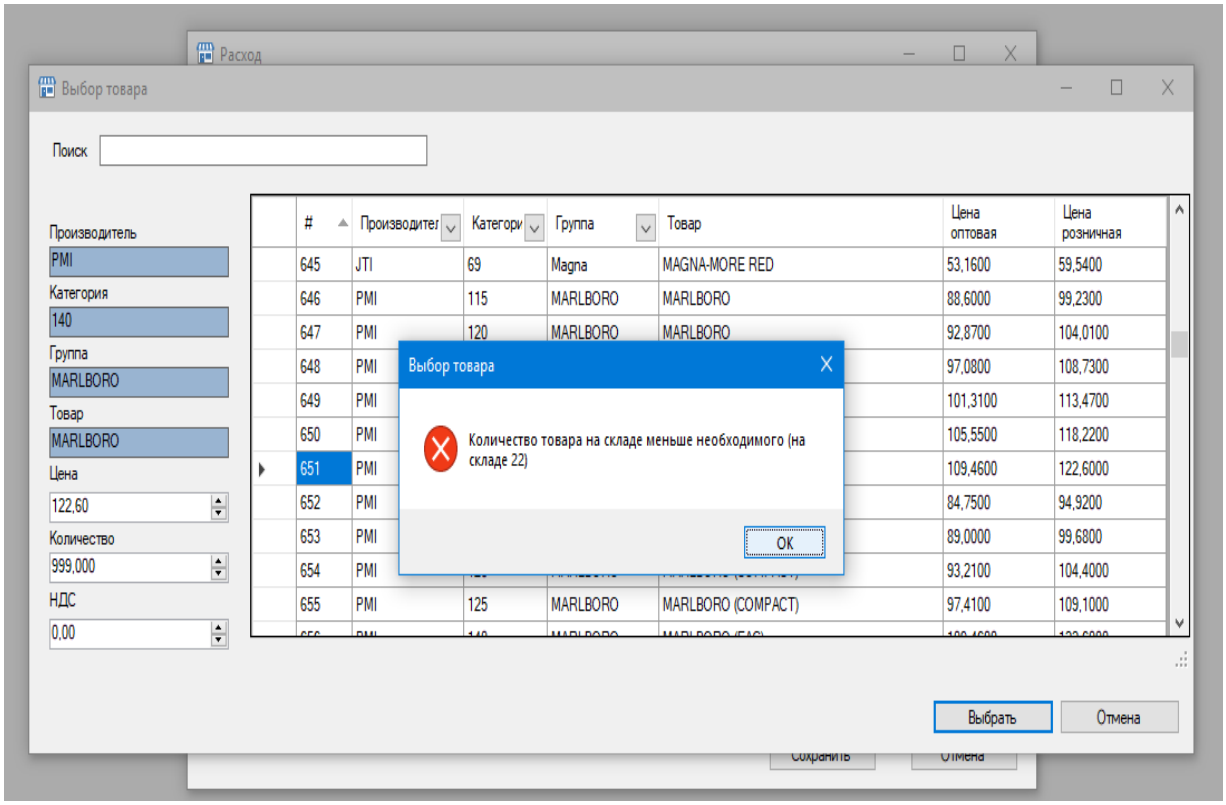


Рис. 63. Предупреждение при превышении остатка

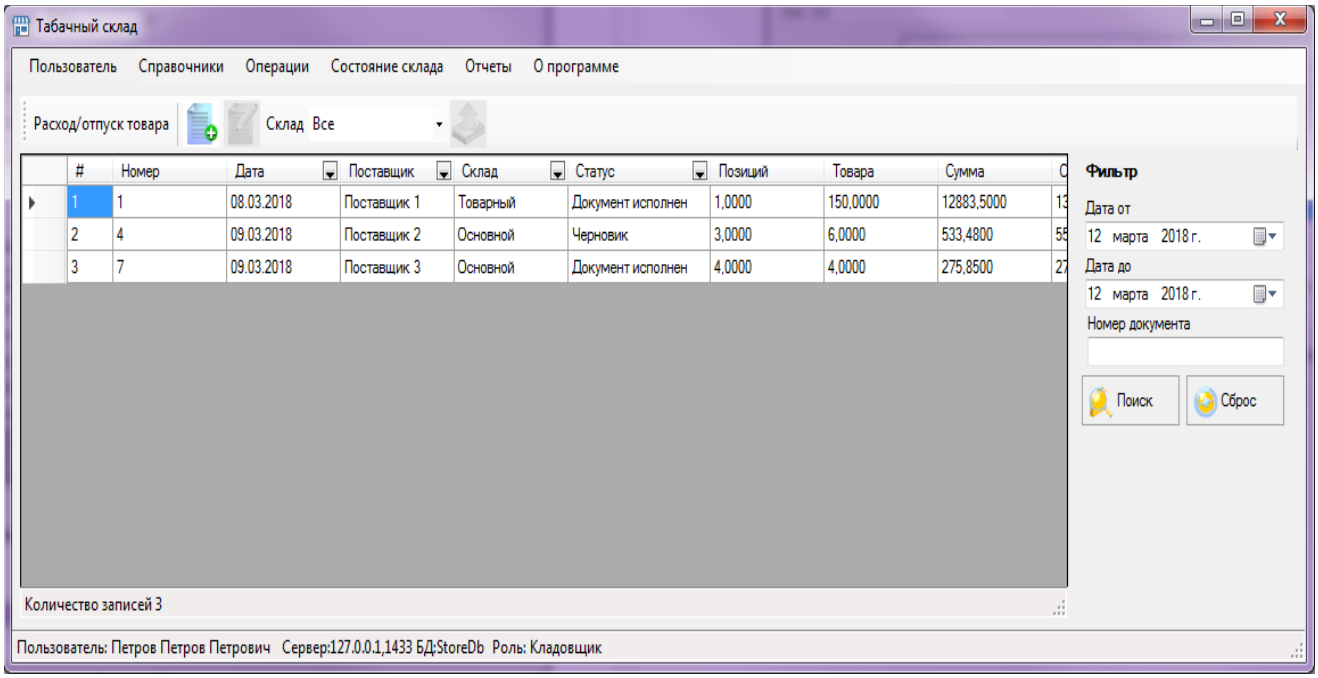


Рис. 64. Отгрузка товара

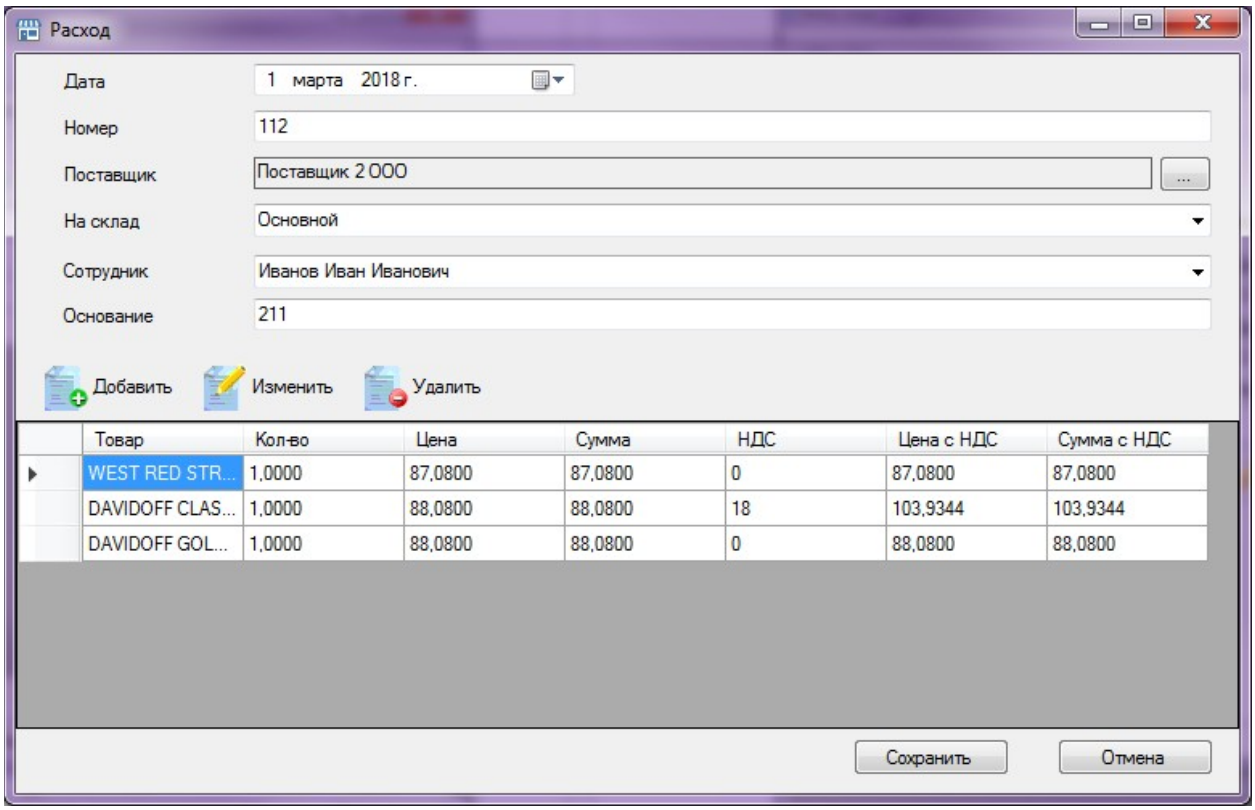


Рис. 65. Ввод отгрузки товара

После создания документа отгрузки к нему можно сформировать счет-фактуру (Рис. 66).

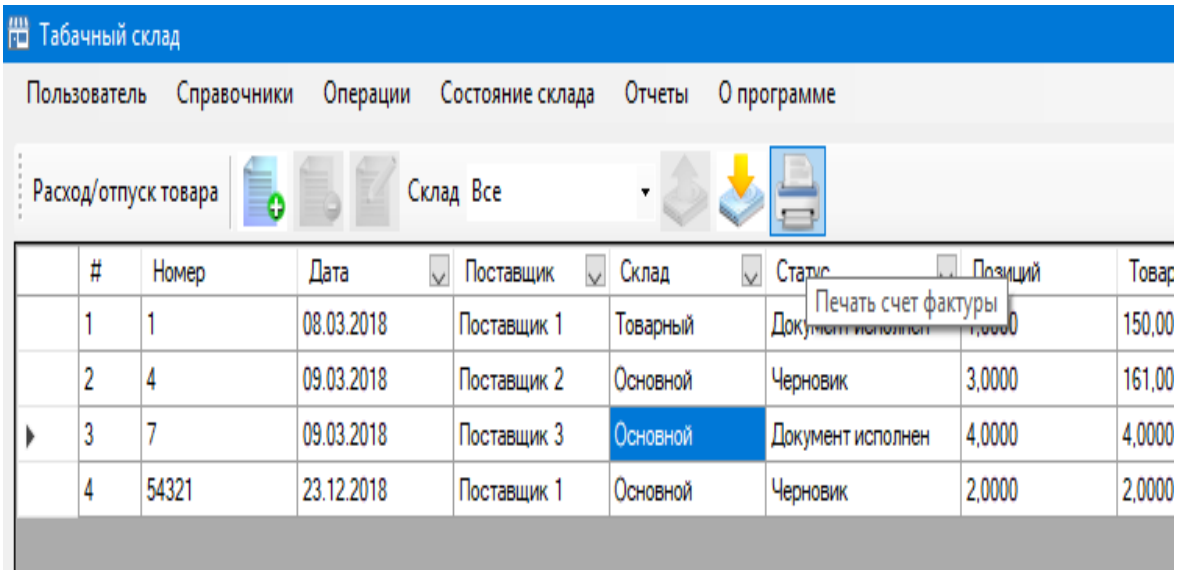
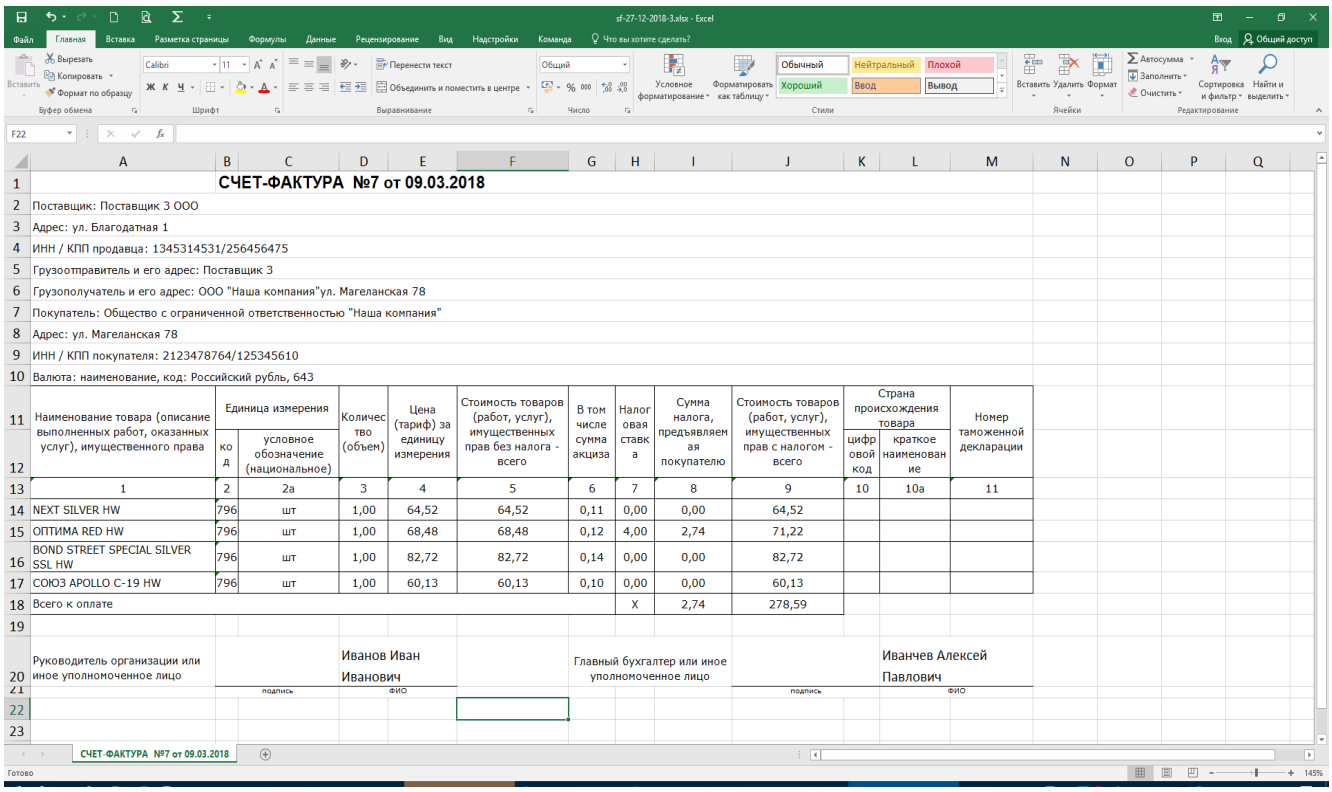


Рис. 66. Формирование счет-фактуры



СЧЕТ-ФАКТУРА №7 от 09.03.2018

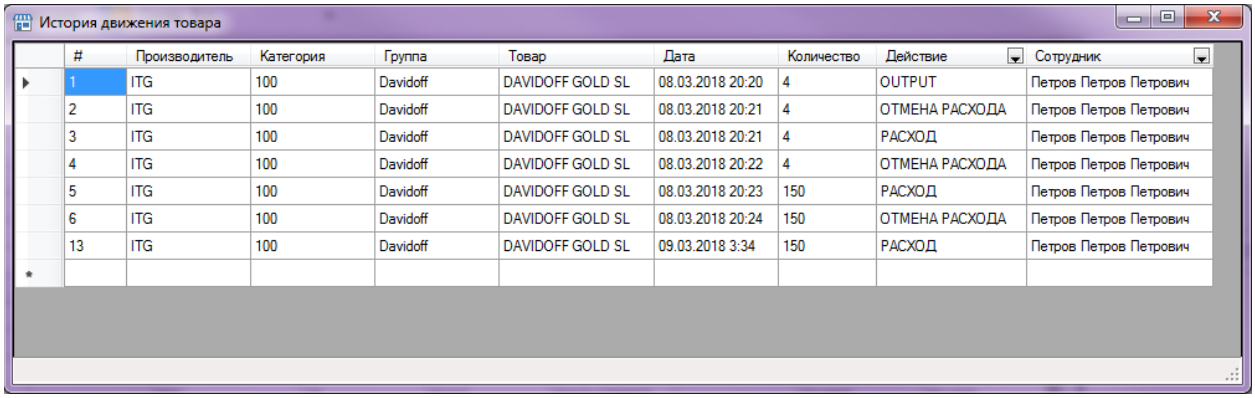
Поставщик: Поставщик 3 ООО
Адрес: ул. Благодатная 1
ИНН / КПП продавца: 1345314531/256456475
Грузоотправитель и его адрес: Поставщик 3
Грузополучатель и его адрес: ООО "Наша компания"ул. Магеланская 78
Покупатель: Общество с ограниченной ответственностью "Наша компания"
Адрес: ул. Магеланская 78
ИНН / КПП покупателя: 2123478764/125345610
Валюта: наименование, код: Российский рубль, 643

Наименование товара (описание выполненных работ, оказанных услуг), имущественного права	Единица измерения		Количество (объем)	Цена (тариф) за единицу измерения	Стоимость товаров (работ, услуг), имущественных прав без налога - всего	В том числе сумма акциза	Налоговая ставка	Сумма налога, предъявляемая покупателю	Стоимость товаров (работ, услуг), имущественных прав с налогом - всего	Страна происхождения товара		Номер таможенной декларации
	код	условное обозначение (национальное)								цифровой код	краткое наименование	
1	2	2а	3	4	5	6	7	8	9	10	10а	11
NEXT SILVER HW	796	шт	1,00	64,52	64,52	0,11	0,00	0,00	64,52			
ОПТИМА RED HW	796	шт	1,00	68,48	68,48	0,12	4,00	2,74	71,22			
BOND STREET SPECIAL SILVER SSL HW	796	шт	1,00	82,72	82,72	0,14	0,00	0,00	82,72			
СОЮЗ APOLLO C-19 HW	796	шт	1,00	60,13	60,13	0,10	0,00	0,00	60,13			
Всего к оплате							X	2,74	278,59			

Руководитель организации или иное уполномоченное лицо — подпись — Иванов Иван Иванович (ФИО)
Главный бухгалтер или иное уполномоченное лицо — подпись — Иванчев Алексей Павлович (ФИО)

Рис. 67. Пример сформированной счет-фактуры

В режиме простора остатков можно посмотреть историю движения любого товара по выбранному складу – для этого надо нажать либо 2 раза на ячейке интересующего товара или соответствующую кнопку над табличной формой.



#	Производитель	Категория	Группа	Товар	Дата	Количество	Действие	Сотрудник
1	ITG	100	Davidoff	DAVIDOFF GOLD SL	08.03.2018 20:20	4	OUTPUT	Петров Петров Петрович
2	ITG	100	Davidoff	DAVIDOFF GOLD SL	08.03.2018 20:21	4	ОТМЕНА РАСХОДА	Петров Петров Петрович
3	ITG	100	Davidoff	DAVIDOFF GOLD SL	08.03.2018 20:21	4	РАСХОД	Петров Петров Петрович
4	ITG	100	Davidoff	DAVIDOFF GOLD SL	08.03.2018 20:22	4	ОТМЕНА РАСХОДА	Петров Петров Петрович
5	ITG	100	Davidoff	DAVIDOFF GOLD SL	08.03.2018 20:23	150	РАСХОД	Петров Петров Петрович
6	ITG	100	Davidoff	DAVIDOFF GOLD SL	08.03.2018 20:24	150	ОТМЕНА РАСХОДА	Петров Петров Петрович
13	ITG	100	Davidoff	DAVIDOFF GOLD SL	09.03.2018 3:34	150	РАСХОД	Петров Петров Петрович

Рис. 68. Форма истории движения товара (режим остатки)

Для формирования отчетов надо выбрать советующий пункт меню и выбрать интересующие критерии отбора данных (рис. 69-74). При формировании отчета они будут учтены и сформированы файлы в каталоге с программой с последующим отображение пользователю (при повторном формировании отчета необходимо закрыть предыдущую версию если она открыта на экране – чтобы получить монопольный доступ к файлу).

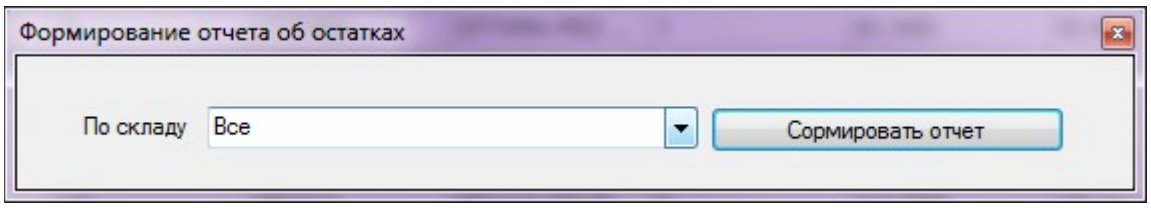
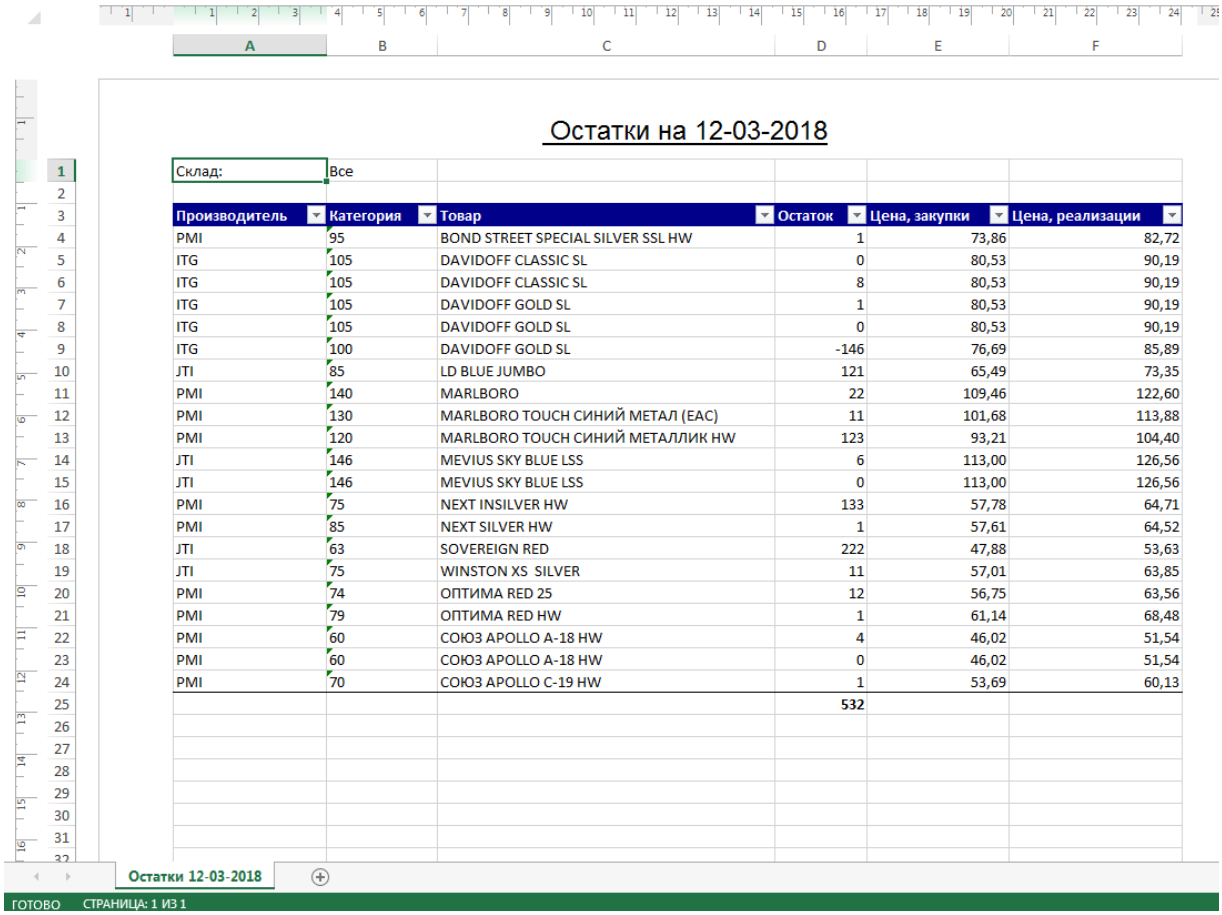


Рис. 69. Форма выбора критериев отчета остатки



Остатки на 12-03-2018

Склад:	Все				
Производитель	Категория	Товар	Остаток	Цена, закупки	Цена, реализации
PMI	95	BOND STREET SPECIAL SILVER SSL HW	1	73,86	82,72
ITG	105	DAVIDOFF CLASSIC SL	0	80,53	90,19
ITG	105	DAVIDOFF CLASSIC SL	8	80,53	90,19
ITG	105	DAVIDOFF GOLD SL	1	80,53	90,19
ITG	105	DAVIDOFF GOLD SL	0	80,53	90,19
ITG	100	DAVIDOFF GOLD SL	-146	76,69	85,89
JTI	85	LD BLUE JUMBO	121	65,49	73,35
PMI	140	MARLBORO	22	109,46	122,60
PMI	130	MARLBORO TOUCH СИНИЙ МЕТАЛ (EAC)	11	101,68	113,88
PMI	120	MARLBORO TOUCH СИНИЙ МЕТАЛЛИК HW	123	93,21	104,40
JTI	146	MEVIUS SKY BLUE LSS	6	113,00	126,56
JTI	146	MEVIUS SKY BLUE LSS	0	113,00	126,56
PMI	75	NEXT INSILVER HW	133	57,78	64,71
PMI	85	NEXT SILVER HW	1	57,61	64,52
JTI	63	SOVEREIGN RED	222	47,88	53,63
JTI	75	WINSTON XS SILVER	11	57,01	63,85
PMI	74	ОПТИМА RED 25	12	56,75	63,56
PMI	79	ОПТИМА RED HW	1	61,14	68,48
PMI	60	СОЮЗ APOLLO A-18 HW	4	46,02	51,54
PMI	60	СОЮЗ APOLLO A-18 HW	0	46,02	51,54
PMI	70	СОЮЗ APOLLO C-19 HW	1	53,69	60,13
			532		

Рис. 70. Вид отчета «Остатки»

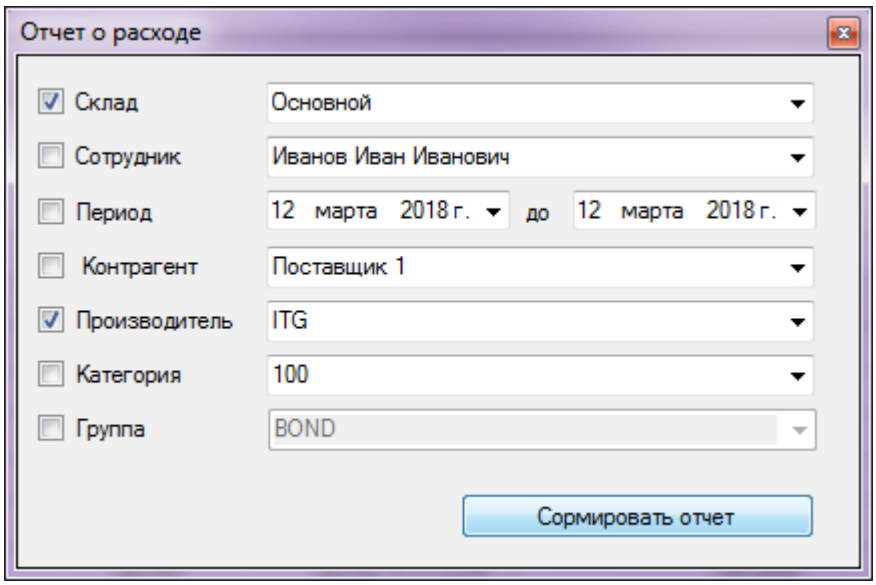


Рис. 71. Форма выбора критериев отчета отгрузки

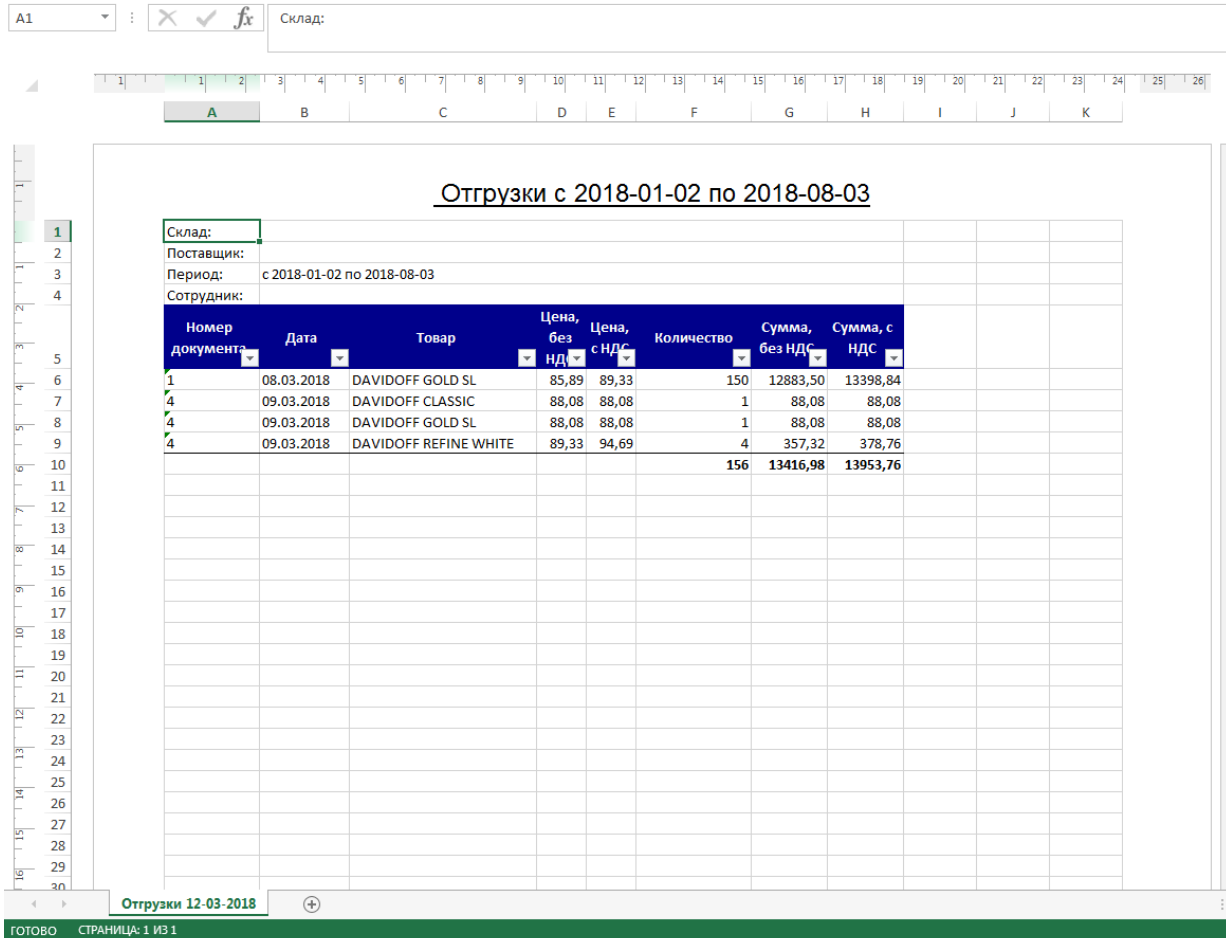


Рис. 72. Вид отчета «Отгрузки»

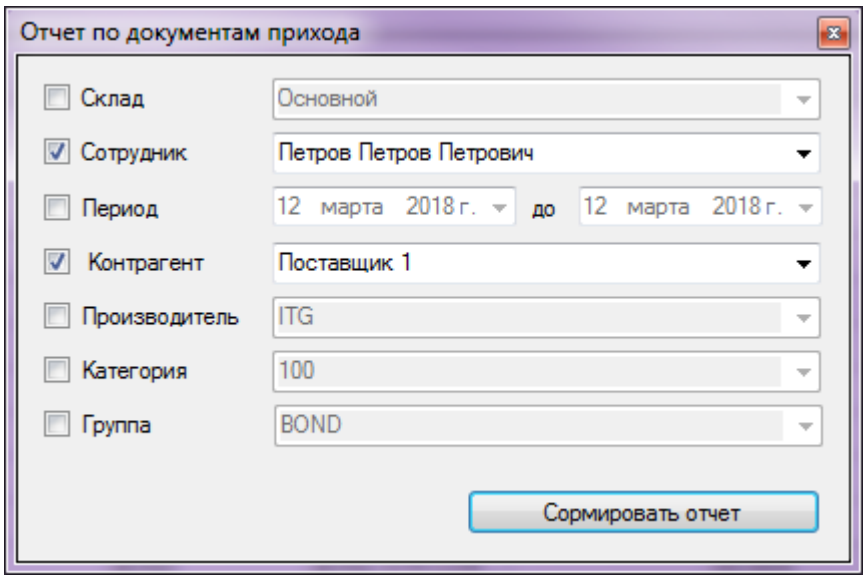
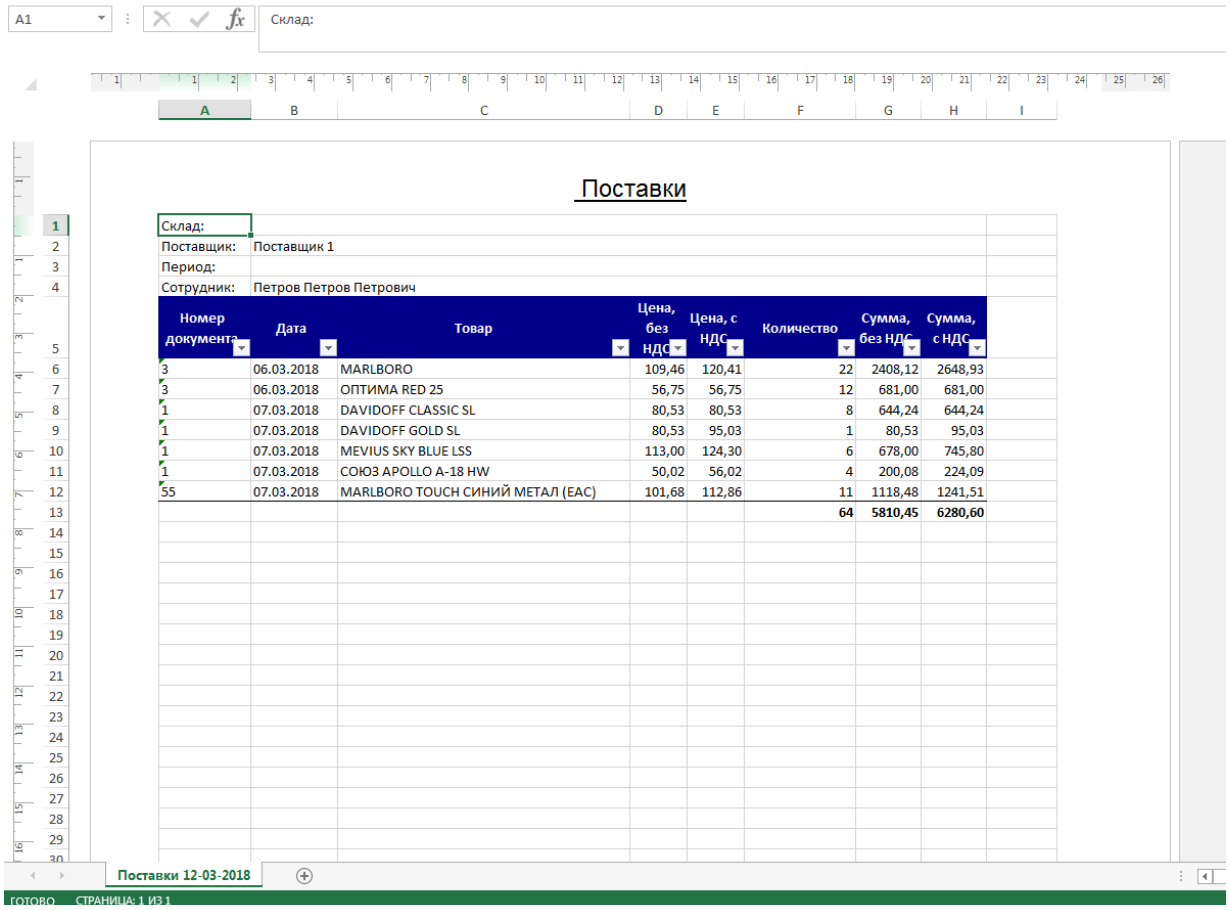


Рис. 73. Форма выбора критериев отчета поставки



Поставки

Склад:	
Поставщик:	Поставщик 1
Период:	
Сотрудник:	Петров Петров Петрович

Номер документа	Дата	Товар	Цена, без НДС	Цена, с НДС	Количество	Сумма, без НДС	Сумма, с НДС
3	06.03.2018	MARLBORO	109,46	120,41	22	2408,12	2648,93
3	06.03.2018	ОПТИМА RED 25	56,75	56,75	12	681,00	681,00
1	07.03.2018	DAVIDOFF CLASSIC SL	80,53	80,53	8	644,24	644,24
1	07.03.2018	DAVIDOFF GOLD SL	80,53	95,03	1	80,53	95,03
1	07.03.2018	MEVIUS SKY BLUE LSS	113,00	124,30	6	678,00	745,80
1	07.03.2018	СОЮЗ APOLLO A-18 HW	50,02	56,02	4	200,08	224,09
55	07.03.2018	MARLBORO TOUCH СИНИЙ МЕТАЛ (EAC)	101,68	112,86	11	1118,48	1241,51
					64	**5810,45**	**6280,60**

Рис. 74. Вид отчета «Поставки»

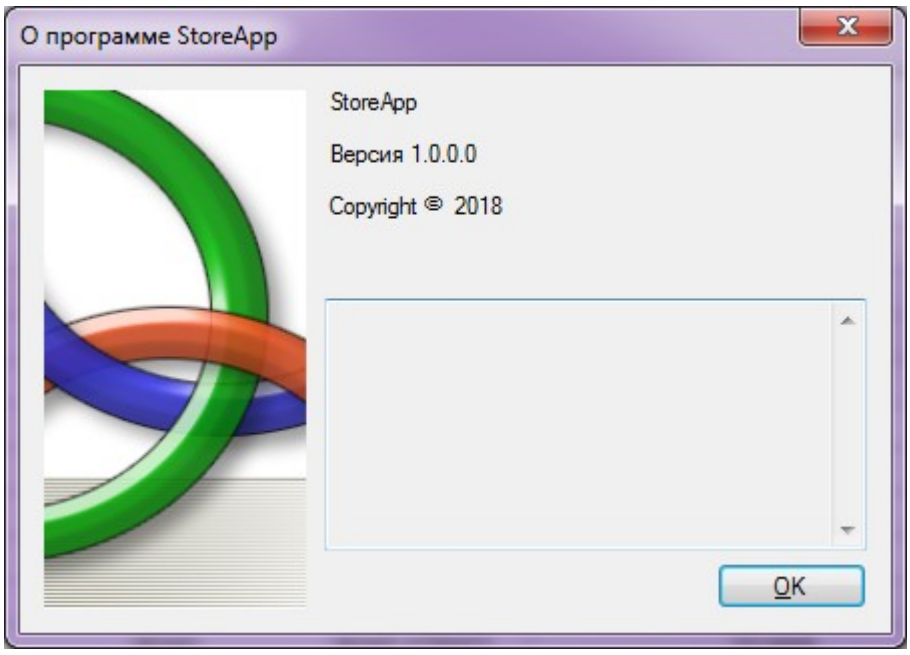


Рис. 75. Форма о программе

Ниже представлены формы, которые меняют свой вид или доступность в режиме администратора:

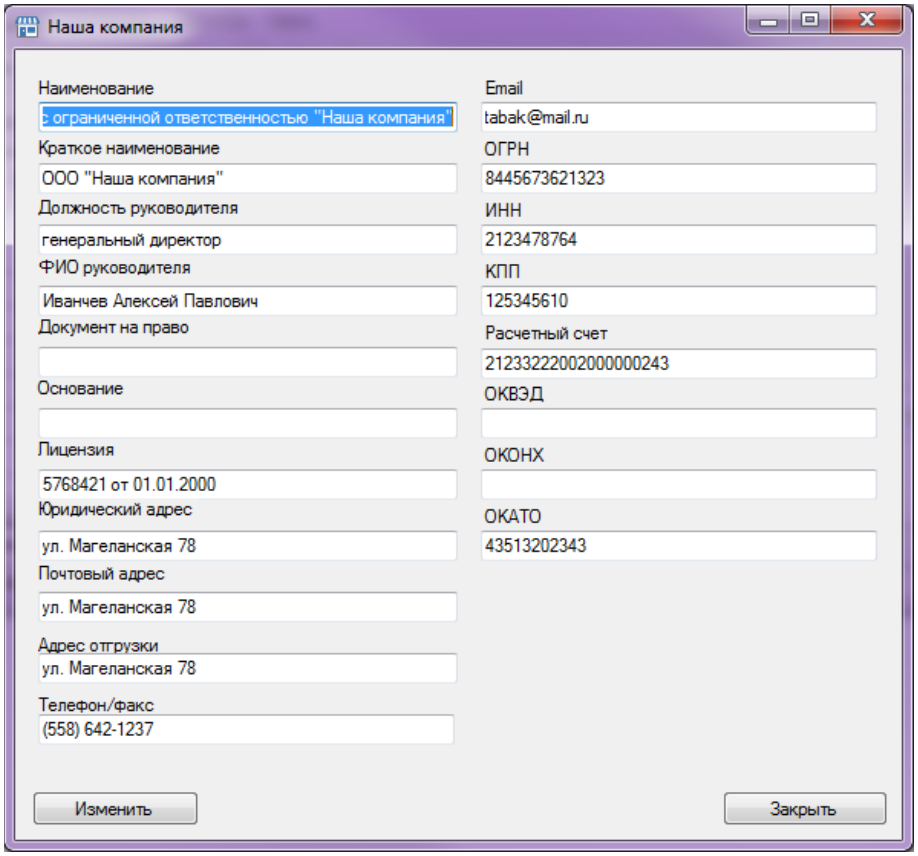


Рис. 76. Форма наша компания (роль администратор)

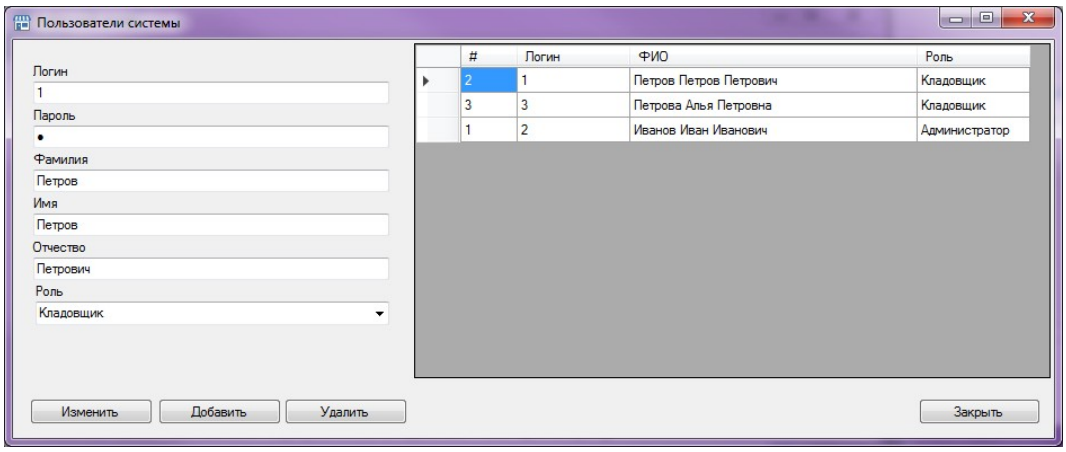


Рис. 77. Форма справочник сотрудник/пользователей (роль администратор)

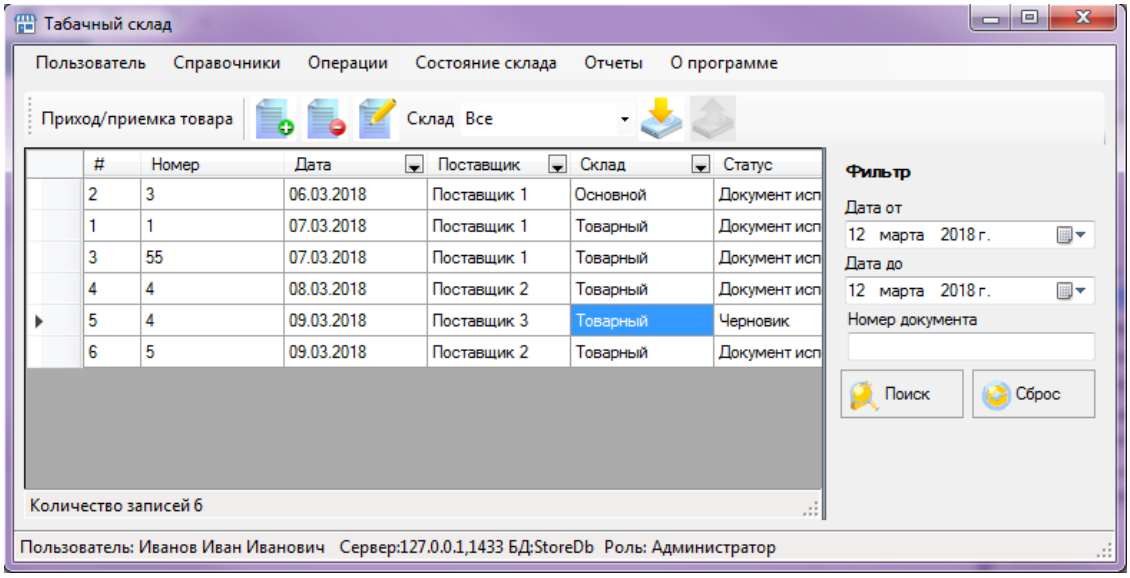


Рис. 78. Приемка товара (роль администратор)

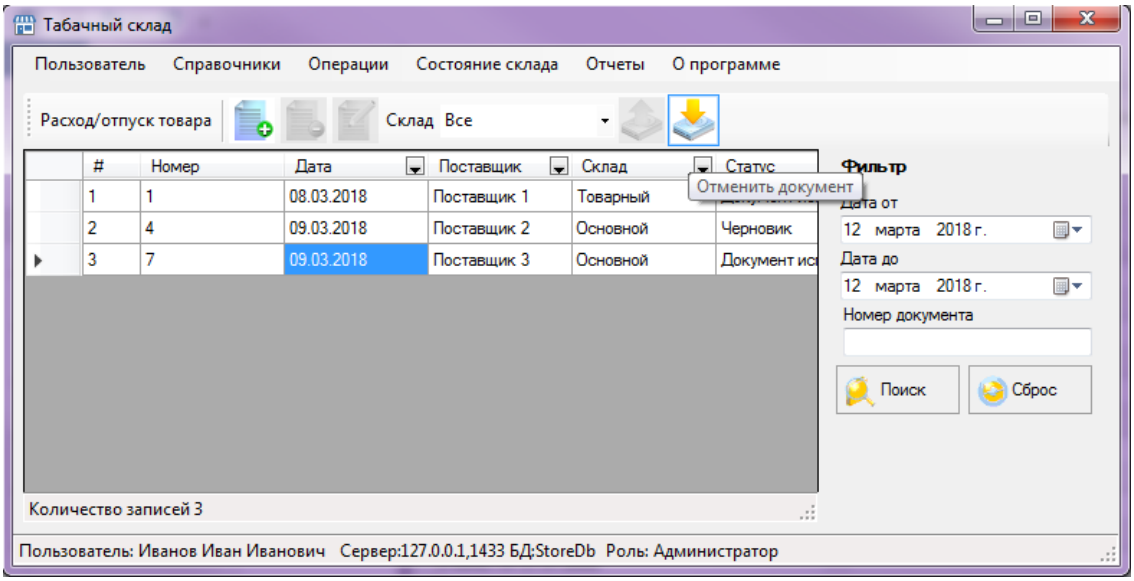


Рис. 79. Отгрузка товара (роль администратор)

3.3. Внедрение разработанной автоматизированной информационной системы «Табачный склад»

Назначение программы

Программа предназначена для автоматизации работы табачного склада, которая позволит сократить временные затраты по внесению данных, их обработке, а также позволит быстро составлять отчеты, получать своевременно необходимую информацию. Также позволит значительно сократить временные затраты на поиск информации. Повысит достоверность получаемой, обрабатываемой и хранимой информации.

Архитектура приложения и условия применения

На сервере установлена операционная система Microsoft Server 2012. Для хранения данных использоваться СУБД Microsoft SQL Server 2016. Также на сервере установлена утилита SQL Server Management Studio (SSMS) — утилита для конфигурирования, управления и администрирования всех компонентов Microsoft SQL Server [5]. Утилита включает скриптовый редактор и графическую программу, которая работает с объектами и настройками сервера.

Минимальные требования для нормального функционирования системы (рабочее место оператора):

- 32-разрядный (x86) или 64-разрядный (x64) процессор с тактовой частотой 1 ГГц или выше;
- 1 ГБ (для 32-разрядного процессора) или 2 ГБ (для 64-разрядного процессора) ОЗУ;
- Графическое устройство DirectX 9 с драйвером WDDM 1.0 или более поздней версии;
- Свободное дисковое пространство на жестком диске 30Мб;
- Операционной системой Microsoft Windows 7 и выше;
- Установленный пакет .Net Framework версии не ниже 4.5

В качестве протокола взаимодействия между компонентами системы на сетевом уровне необходимо использовать протокол TCP/IP.

Инструкция пользователя

Для начала эксплуатации АИС «Табачный склад» необходимо на сервере в MS SQL Server восстановить базу данных, из предлагающегося к программе бэкапа bd.bak, как указано на рисунке 80 и 81.

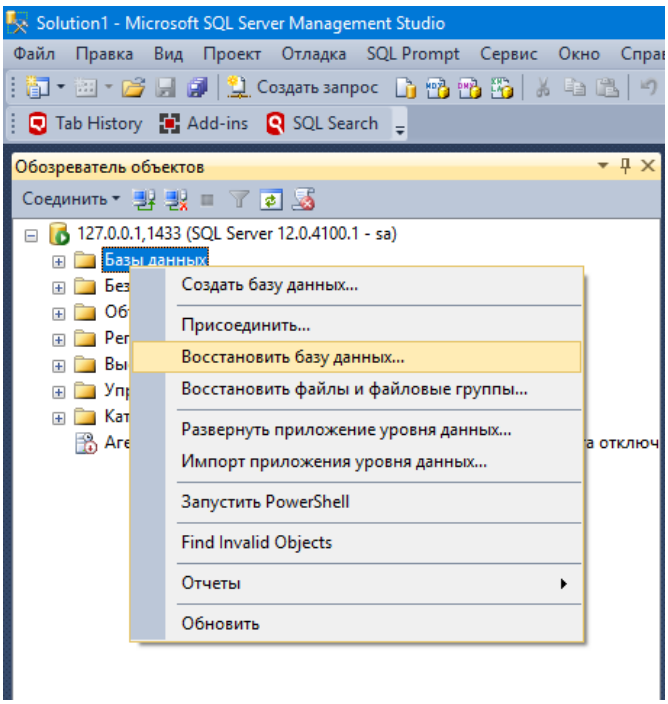


Рис. 80. Восстановление базы данных из бэкапа

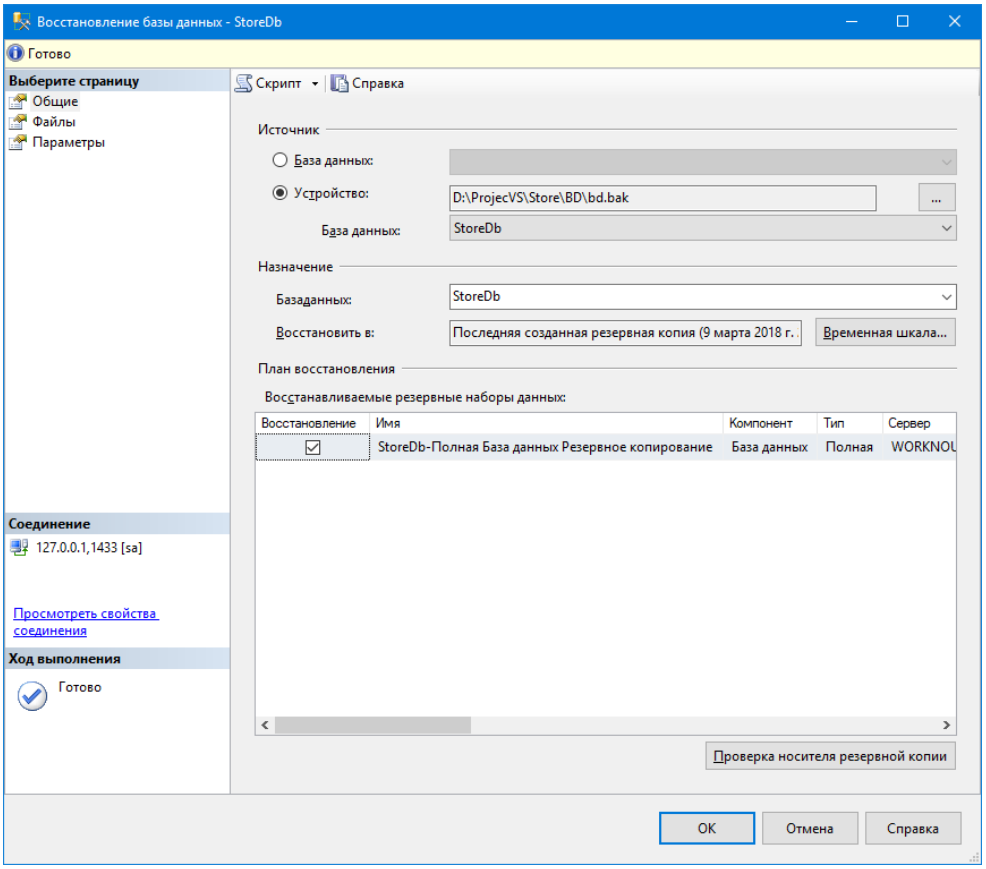


Рис. 81. Восстановление базы данных из бэкапа bd.bak

После восстановления можно приступить к работе с программой, как описано в пункте 3.2.

3.4. Расчет показателей экономической эффективности проекта

Эффективность – это одна из возможных характеристик качества системы, а именно её характеристика с точки зрения соотношения затрат и результатов функционирования системы [18].

Экономический эффект – результат внедрения, какого – либо мероприятия, выраженный в стоимостной форме, в виде экономии от его осуществления. Так, для организаций, использующих программные изделия, основными источниками экономии являются:

- Улучшение показателей их основной деятельности, происходящее в результате использования программного изделия;
- Повышение технического уровня, качества и объёмов вычислительных работ; увеличение объёмов и сокращение сроков переработки информации [18].

Как правило, внедрение средств вычислительной техники для обработки информации направлено на повышение эффективности производства через совершенствование процессов управления, что проявляется в показателях производственно-хозяйственной деятельности объекта управления. Поскольку реализация мероприятий по использованию средств вычислительной техники, средств - связи и оргтехники для целей управления требует значительных финансовых затрат, то расходы по внедрению проектов информационно-вычислительного обслуживания пользователей должны окупаться в как можно более короткие сроки. При создании автоматизированной системы управления предприятие несет единовременные затраты на ее разработку, приобретение необходимых технических и программных средств, текущие затраты на функционирование системы, подготовку и переподготовку кадров. Экономия от функционирования АИС определяется с учетом затрат на ее эксплуатацию[19].

Стандартная методика расчета показателей экономической эффективности включает в себя расчет суммы годовой экономии, коэффициента экономической эффективности капитальных вложений и срока окупаемости капитальных вложений [18].

Экономическое обоснование эффективности инвестиций характеризует отношение результатов — величины прибыли к величине суммарных затрат на создание и эксплуатацию системы. Для этого необходимо определить цену разрабатываемой программы, которая включает:

- Затраты на основную заработную плату разработчиков;
- Затраты на дополнительную заработную плату;
- Отчисления в социальную сферу;
- Отчисления органам социального страхования от несчастных случаев;

- Накладные расходы;
- Затраты на разработку и отладку программы.

Разработанная система предназначена для использования в пределах одной организации. Кроме того, разработка программы осуществлялась по заказу организации, соответственно все права на данный модуль принадлежат единственному заказчику.

Созданием системы по заказу потребителя занимался специалист, знающий предметную область, в тоже время являясь разработчиком ИТ и выполняющего помимо этого функции менеджера проекта и автора технической документации.

Разработка велась в помещении и на оборудовании заказчика.

Процесс разработки делится на этапы и соответствующие им события (таблица 1). Эти этапы в основном соответствуют обычному процессу разработки любого программного обеспечения: постановка задачи, анализ требований, проектирование, кодирование и т.п.

Таблица 1 - Перечень событий

Номер этапа	Наименование события
00	Получение задания на разработку
01	Изучение предметной области
02	Постановка требований к системе
03	Изучение требований к системе
04	Построение функциональной модели
05	Построение инфологической модели
06	Разработка общей структуры программ системы
07	Проектирование и программирование модулей системы
08	Тестирование
09	Разработка требований к доработке по результатам тестирования
10	Доработка по результатам тестирования
11	Разработка программной документации
12	Установка и внедрение системы
13	Обучение персонала

На основе таблицы 1 составляется перечень работ, который отражает их взаимосвязь и логическую последовательность. На разных этапах разработки системы специалист – программист тратил различное время в зависимости от трудоемкости и сложности этапа. Эти показатели приведены в таблице 2.

Таблица 2 - Перечень работ

Код работы	Содержание работы	Время выполнения, дни
00-01	Изучение предметной области	3
01-03	Постановка, изучение требований к системе	1
03-04	Построение функциональной модели	1
04-05	Построение инфологической модели	2
05-06	Разработка общей структуры программ подсистемы	2
06-07	Проектирование и программирование модулей подсистемы	24
07-08	Тестирование	4
08-09	Разработка требований к доработке по результатам тестирования	1
09-10	Доработка по результатам тестирования	3
10-11	Разработка программной документации	3
11-12	Установка и внедрение подсистемы	1
12-13	Обучение персонала	2

Продолжительность разработки составляет 47 дней.

Для определения трудоемкости и стоимости разрабатываемой ИТ необходимо рассчитать следующие статьи затрат, которые приведены с формулами расчетов в таблице 3.

Таблица 3 – Статьи затрат

Статьи затрат	Обозначения	Порядок расчета
Затраты на разработку и отладку программы	$З_{ОП}$	$T_m \cdot l_r$
Основная заработная плата	$З_о$	$\sum t_i \cdot B_i$
Дополнительная заработная плата	$З_Д$	$(10\% \cdot З_О)/100\%$
Отчисления в социальную сферу, в том числе:	$З_{ЕСН}$	$30,2\% \cdot (З_О + З_Д)/100\%$
в пенсионный фонд	$З_{ПФ}$	$22\% \cdot (З_О + З_Д)/100\%$
в фонд социального страхования	$З_{СС}$	$2,9\% \cdot (З_О + З_Д)/100\%$
в фонд медицинского страхования	$З_{МС}$	$5,1\% \cdot (З_О + З_Д)/100\%$
Отчисления органам социального страхования от несчастных случаев	$З_{НС}$	$0,2\% \cdot (З_О + З_Д)/100\%$
Накладные расходы	$Р_{НО}$	$(40\% \cdot З_О)/100\%$

Где $T_м$ – машинное время ЭВМ на разработку и отладку программы, машиночас;

l_r – эксплуатационные расходы, приходящиеся на один час машинного времени,

руб./машиночас;

$\sum t_i$ - время на разработку программы работника i-ой квалификации, чел./месяц;

B_i – основная заработная плата разработчика i-ой квалификации, руб./месяц.

Эксплуатационные расходы одного часа машинного времени находятся по формуле

$$l_r = (C_{эл} + C_{обсл} + A) / \Phi_г ,$$

где $C_{эл}$ – стоимость электроэнергии, потребляемой компьютером за год, руб.;

$C_{обсл}$ – расходы на обслуживание одного компьютера в течение года, руб.;

A – амортизационные отчисления, руб.;

$\Phi_г$ – годовой полезный фонд времени работы ЭВМ, ч.

Компьютер работает в течение 8 часов в сутки, пять дней в неделю, 248 дней в году. Годовой полезный фонд времени работы компьютера при таких условиях составляет:

$\Phi_г = 248 \cdot 8 = 1984$ часа

Стоимость потребляемой за год электроэнергии рассчитывается по формуле:

$C_{эл} = M_п \cdot Ц_{эл} \Phi_г$,

где $M_п$ – потребляемая мощность ЭВМ, кВт;

$Ц_{эл}$ – цена 1 кВт/ч электроэнергии, руб.

Потребление электроэнергии компьютером 0,17 кВт, стоимость одного кВт энергии составляет 5,04 руб. Расчет равен:

$C_{эл} = 0,17 \cdot 5,04 \cdot 1984 = 1699,89$ руб.

Расходы на обслуживание одного компьютера за год определяются по формуле:

$C_{обсл} = ЗП_{обсл} \cdot 12$,

где $ЗП_{обсл}$ - заработная плата обслуживающего персонала по компьютерам, включающая в себя отчисления ЕСН, руб.;

12 – число месяцев в году.

Заработная плата персонала по обслуживанию одного компьютера составляет 6 000 рублей в месяц. Таким образом,

$C_{обсл} = 6000 \cdot 12 = 72000$ руб.

Амортизационные отчисления в организации определяются по методу уменьшаемого остатка по формуле:

$A = H_a \cdot k_{уск} \cdot C / 100$,

где H_a – норма амортизации,

$k_{уск}$ - коэффициент ускорения,

С – балансовая стоимость оборудования, руб.,

$t_{полез}$ - срок полезного использования оборудования, год.

Норма амортизации рассчитывается:

$H_а = 100\%/t_{полез}$.

Срок полезного использования компьютера ($t_{полез}$) на предприятии берется из расчета 8 лет. Балансовая стоимость одного компьютера на время разработки программы составляет 18 000 рублей. $H_а$ равна 12,5%. Коэффициент ускорения равен 2. Поэтому амортизация ЭВМ на первый год использования равна

$А_{ЭВМ1}$=12,5 · 2 ·18000/100 = 4500 руб.

Эксплуатационные расходы на один компьютер определяются по формуле (3.10) и составляют в один час использования

l_r = (1618,94+72000+4500)/1984 = 39,37 руб.

В разработке использовался один компьютер. Оборудование применяли, начиная с 5-го этапа разработки. $Т_м$ вычисляется с учетом восьмичасового рабочего дня.

Программист – 41 день, $Т_м = 41 \cdot 8 = 320$ часов;

В соответствии с этими данными затраты на разработку и отладку программы по формуле составляют

$З_{ОП} = 39,37 \cdot 320 = 12598,4$ руб.

На основании договора найма, заключенного с разработчиком, определяется месячная и суточная заработная плата специалиста, исходя из пятидневной рабочей недели, которые приведены в таблице 4.

Таблица 4 - Заработная плата

Специалист	Заработная плата в месяц (B_i), руб.	Суточная ставка, руб.
Программист	35 000	1590,91

Расчет заработной платы и дополнительных расходов производится в соответствии с формулами. Результаты приведены в таблице 5.

Таблица 5 - Заработная плата участников разработки

Специалист	Программист
Отработанные дни	47
Зо, руб.	74772,77
Зд, руб.	7477,28
Зпф, руб.	18095,01
Змс, руб.	4194,75
Зсс, руб.	2385,25
Знс, руб.	164,50
Рно, руб.	29909,11
Общие затраты, руб	136998,67

Таким образом, расходы на оплату труда составляют 136998,67 руб.

Расчет цены программы определяется по формуле

$Ц_п = (1+ \alpha_р) (З_о+З_д+З_{ЕСН}+З_{нс}.+Р_{но}.+З_{оп})$,

где $\alpha_р$ - норматив рентабельности, учитывающий прибыль организации, разрабатывающей данную программу.

$Ц_п = (1+0,15) \cdot (136998,67 +12598,4) = 149597,07$ руб.

Итак, цена программного продукта, полученного в ходе дипломного проектирования, составляет 149597,07 руб. при норме рентабельности 15%.

Анализ эффекта от внедрения автоматизированной информационной системы.

Экономическая эффективность разработки оценивается с точки зрения заказчика, т.е. пользователя системы. В этом случае необходимо сравнить, насколько эффективнее использовать разработанный продукт по сравнению с ручным способом решения поставленной задачи[26].

Затраты, производимые организацией с использованием ручного способа учета, включают в себя:

- Затраты на основную заработную плату работников по учету и контролю складских запасов, продаж товаров и поставок продукции;
- Затраты на дополнительную заработную плату;
- Отчисления в социальную сферу;
- Отчисления органам социального страхования от несчастных случаев;
- Накладные расходы.

Система разработана:

- Для кладовщиков, ведущих учет продаж, также учет запасов;
- Для администраторов, ведущих учет и контроль за движением товаров;

- Для бухгалтеров.

До введения системы необходимая информация считалась и располагалась на бумажных носителях, вне автоматизированной системы.

Кладовщика занимался учетом проданных единиц товара, занося все продажи в журнал продаж, сверяя в конце рабочего дня итог по кассе и по журналу. На учет продаж и проверку итогов отводилось в среднем по 2,5 часа в день, что составляет 6,875 дней в месяц и около 8 часов в месяц подсчет запасов (по 2 часа каждую неделю). Всего на выше перечисленные операции у кладовщика уходило по 7,875 дней в месяц.

Администратор ввел учет поставок товара и запасов на основном складе, что занимало в зависимости от частоты поставок в среднем по 10 часов в месяц. Кроме этого администратор ввел учет и составлял сводный отчет о движении товаров по нескольким точкам продаж, что занимало в свою очередь около 30 часов в месяц. Что составляло в месяц 5 дней.

У бухгалтера на введение учета движения товаров в среднем в день уходило 1,5 часа рабочего времени. На составление бухгалтерской отчетности в конце месяца у специалиста уходило около 3 дней. В сумме на учет и подготовку необходимой документации в месяц уходило в среднем 7,125 дня

Таким образом, расходы на заработную плату работников по учету и контролю за движением товаров до введения программы при окладе кладовщика 30000 рублей, администратора 40000 рублей и бухгалтера 50000 рублей без отчислений в социальную сферу рассчитываются в соответствии с формулами.

Организация работает при пятидневной неделе 22 рабочих дня в месяц. Основная зарплата оператора ведется из расчета его занятости в учете, то есть

$З_о = B_i/t_м \cdot \sum t_i$,

где B_i – месячный оклад специалиста, руб.;

$t_м$ – число рабочих дней в месяц;

$\sum t_i$ - число отработанных дней в месяц специалистом.

$З_о = 30000/22 \cdot 7,875 = 10738,64$ руб.

Основная заработная плата администратора относительно работы по учету движения товаров составит:

$З_о = 40000/22 \cdot 5 = 9090,91$ руб.

Основная заработная плата бухгалтера относительно работы по учету движения товаров составит:

$З_о = 50000/22 \cdot 7,125 = 16193,2$ руб.

Результаты представлены в таблице 6.

Таблица 6 - Заработная плата работников по учету и контролю над движением товаров

Специалист	Отработанные дни	$З_О$, руб.	$З_Д$, руб.	$З_{ПФ}$, руб.	$З_{МС}$, руб.	$З_{СС}$, руб.	$З_{НС}$, руб.	$Р_{НО}$, руб.	Общие затраты, руб.
Оператор	7,88	10738,64	1073,86	2598,75	602,44	342,56	23,63	4295,46	19072,9
Администратор	5,00	9090,91	909,09	2200	510	290	20	3636,34	16656,34
Бухгалтер	7,13	16193,2	1619,32	3918,75	908,44	516,56	35,3	6477,28	29668,85
Всего	20,01	36022,75	3602,28	8717,5	2020,88	1149,12	78,93	14409,08	65398,09

Таким образом, расходы на оплату труда сотрудников организации по учету за движением товаров составляют 65398,09 руб. в месяц. В год эти расходы будут равны:

$65398,09 \cdot 12 = 784777,08$ руб.

В сопоставление полученным данным необходимо рассчитать производимые затраты с использованием разработанной программы, которые включают в себя:

- Расход электроэнергии вычислительной техникой;
- Обслуживание и ремонт оборудования дополнительным персоналом;
- Расходы на оплату труда сотрудников организации по учету за движением товаров, но уже с использованием программы.

Стоимость потребляемой за год электроэнергии рассчитывается по формуле. Потребление электроэнергии одним компьютером составляет 0,17 кВт в час, стоимость одного кВт энергии составляет 5,04 руб. Расчет в первый год эксплуатации системы равен:

$С_{эл} = 0,17 \cdot 5,04 \cdot 1984 = 1699,89$ руб.

В связи с тем, что программа используется на уже установленных компьютерах, которые обслуживаются и ремонтируются, то расчет затрат не имеет смысла. И $С_{обсл}$ будет равно 0.

Таким образом, материальные затраты организации после введения системы на первый год составят:

$З_{пр} = С_{эл} + С_{обсл}$

$З_{пр} = 1618,94$ руб.

Срок полного использования программы (горизонт расчета $Т_и$) можно рассчитать

из расчета темпа морального устаревания

$T_u = 1/k_u = 1/0,20 = 5,$

где k_u – коэффициент, учитывающий время годового устаревания - 0,003-0,4.

В этом случае срок полезного действия программы стоит принять за 4 лет.

Последующие 4 года применения программы организация производит амортизационные отчисления по оборудованию и автоматизированной системе учета и контроля складских запасов, продаж и поставок товаров методом уменьшаемого остатка. В состав оборудования входит только компьютер-сервер, используемый основный инструментом во время разработки системы. Балансовая стоимость ЭВМ на время расчета амортизации равна:

$С_{ЭВМ} = 18000 – 4500 = 13500$ руб.

Отчисления на амортизацию за второй год использования оборудования при $Н_а$ 12,5% в соответствии с формулой составят:

$А_{ЭВМ2}= 12,5 \cdot 2 \cdot 13500 /100 = 3375$ руб.

Балансовая стоимость программного продукта соответствует цене системы, которая равна 149597,07 руб. Срок полезного использования 4 лет, по формуле $Н_а$ для ПО составляет 25%, поэтому амортизация за первый год использования равна:

$А_{пр1} = 25 \cdot 2 \cdot 149597,07 /100 = 74798,54$ руб.

После введения программы количество времени используемые для учета и контроля за движением товара уменьшилось и составило:

- Кладовщик – 5;
- Бухгалтер – 3;
- Администратор – 2,5.

Таким образом, расходы на оплату труда сотрудников организации по учету за движением товаров составят:

(6818,18 + 6818,18+ 4545,44)*12= 218181,6 руб.

В сводную таблицу 7 выведены данные, показывающие в сравнении учет ручным способом и с помощью системы на последующие 4 года. Амортизационные отчисления представлены в виде суммы отчислений за оборудование и ПО. Причем расчеты учитывают годовую инфляцию 4%.

Таблица 7 - Сводные экономические показатели по разработке подсистемы

Год использования	Ручной счет, руб.	С использованием программы, руб.	Экономия, руб.	Амортизационные отчисления, руб.
1	784777,08	218181,6	566595,48	74798,54
2	816168,16	226908,86	589259,3	37399,27
3	848814,89	235985,22	612829,67	18699,63
4	882767,49	245424,63	637342,86	9349,82
Всего	3332527,62	926500,31	2406027,31	140247,26

Расчет показателей эффективности инвестиционного проекта

Экономическая эффективность инвестиционного проекта характеризуется системой показателей, отражающих соотношение результатов (притока, поступлений, дохода) и затрат (оттока, выплат) от операционной, инвестиционной и финансовой деятельности. В целях экономического обоснования эффективности вложений необходимо произвести расчет показателей:

- Чистого дисконтированного дохода (ЧДД);
- Срока окупаемости вложений ($Т_{ок}$);
- Индекса рентабельности (IP).

Чистый дисконтированный доход (ЧДД) определяется как разница между суммой чистого денежного потока за жизненный цикл и суммой инвестиционных затрат на его реализацию

$$ЧДД = \sum_{1}^{Tu} (P_t - З_t)/(1+E)^t = \sum_{1}^{Tu} Э_t/(1+E)^t,$$

где P_t – результаты, полученные в t периоде, руб.;

$З_t$ – затраты, произведенные в t периоде, руб.;

Tu – горизонт расчета (срок полного использования программы);

E – норма дисконта, равная приемлемой для инвестора нормы с дохода на капитал, %;

Эt – экономия от внедрения системы, руб.

В предыдущих пунктах получено, что окончательная цена разрабатываемого ПО составляет 149597,07 руб. Так же выявлена годовая экономия от внедрения системы 566595,48 руб., что составляет доход организации. Горизонт расчетов ($Т_u$) принять 4 года,

как срок полезного действия программы. Необходимо проверить является ли инвестиционный проект эффективным.

Денежный поток капиталовложений соответствует движению поступлений и затрат в t периоде.

Вычисление ставки дисконтирования:

E = 7,75 + 4 = 11,75%,

где 12% – ставка рефинансирования ЦБ РФ;

4% - коэффициент, учитывающий риск.

Коэффициент доходности рассчитывается по формуле:

$$КД = 1/(1+E)^t.$$

Соответственно на каждый год использования программы вычисляется свой коэффициент доходности. Данные о полученном КД, расчет ЧДД представлен в таблице 8.

Таблица 8 - Расчет чистой дисконтированной стоимости (дохода)

Год эксплуатации	Инвестиции, руб.	$Э_t$, руб.	Амортизация, руб.	Денежный поток, руб.	КД	ЧДД, руб.
0	149597,07			-149597,07	1	-149597,07
1	0	566595,48	74798,54	641394,02	0,8949	573983,51
2	0	589259,3	37399,27	626658,57	0,8008	501828,18
3	0	612829,67	18699,63	631529,3	0,7166	452553,9
4	0	637342,86	9349,82	646692,68	0,6412	414659,35
Всего	149597,07	2406027,31	140247,26	2396677,5		1793427,87

ЧДД величина положительная, поэтому инвестиционный проект жизнеспособен.

Индекс рентабельности (доходности) (IP) определяется отношением суммы приведенных эффектов к размеру капитальных вложений:

$$IP = \sum_{1}^{Tu} (P_t - З_t)/(1+E)^t / K,$$

где K - размер капитальных вложений, руб.

IP = 1793427,87 / 149597,07= 11,98

Так как при IP>1 проект является эффективным, при IP <1 - убыточным, а рассчитанный индекс доходности равен 11,98, следовательно, проект разработки ПО является эффективным.

Срок окупаемости – это период, за который вложения и другие затраты на проект покрываются суммарными результатами от его реализации. Он определяется:

$$Т_{ок} = \frac{З_и}{Д_{сг}},$$

где $З_и$ – чистые инвестиционные затраты, руб.;

Дсг – среднегодовой приток денежных средств от проекта, руб.

Дсг = 2406027,31 / 4 = 601506,83 руб.

Ток = 149597,07/601506,83 = 0,249 года или 2,988 месяца.

Исходя из полученных результатов, можно сделать вывод, что разработанная система жизнеспособна и эффективна при использовании, причем при полном сроке использования 4 года программа окупится 2,988года или 2,988 месяца, то есть последующие 3 года применения системы дают покупателю чистую прибыль (выгоду).

Заключение

Перед началом работы над автоматизацией любой предметной области первым делом необходимо тщательно изучить эту сферу деятельности. Заказчик может требовать

от разработчика автоматизацию любого процесса в производстве или в делопроизводстве. Положительный эффект будет достигнут только после полного и правильного осознания принципа работы в этой сфере.

Задачи, которые были реализованы в выпускной квалификационной работе:

- Проанализирована предметная область;
- Изучен весь материал по данной теме;
- Смоделированы диаграммы бизнес-процессов;
- Создан и протестирован программный продукт.

В результате был создан программный продукт, который может быть использована на предприятии занимающимся дистрибьюцией табачной продукции.

АИС позволяет вести учет поставок и отгрузок товара, контролировать остатки и движение товара, формировать отчеты по работе склада. Программа проста и понятна для любого пользователя, для ее использования не нужно специального обучения, при этом она выполняет в основном все функции, необходимые для работы.

В результате проектирования и разработки системы – было получено приложение для работы с базой данных с целью обеспечения автоматизации деятельности табачного склада, позволяющее упростить работу работников склада в области получения оперативной информации, а также контроля его деятельности. Данное приложение может быть интересно любым коммерческим организациям, ведущим складскую деятельность.

Анализ экономической эффективности показал целесообразность разработки АИС, при полном сроке использования 4 года программа окупится 2,988года или 2,988 месяца, то есть последующие 3 года применения системы дают покупателю чистую прибыль (выгоду).

Список литературы

Законодательные акты и нормативные документы

1. ГОСТ 19. 505 – 79 Руководство оператора
2. ГОСТ 19.201 – 78 Техническое задание

3. ГОСТ 19.402-78 Описание программы
4. ГОСТ 24.703-85 Единая система стандартов автоматизированных систем управления. Типовые проектные решения в АСУ. Основные положения.
5. ГОСТ 34.602-89. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
6. ГОСТ Р ИСО/МЭК 9126-93 Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению. – М.: Издательство стандартов, 1994. – 19 с.
7. Об информации, информационных технологиях и о защите информации: федеральный закон, 27.07.2006 г. № 149-ФЗ: принят Гос. Думой .08.07.2006 г.; одобрен Советом Федерации 14.07.2006 г.; опубликован в «РГ» – Федеральный выпуск № 4131 от 29.07.2006 г.

Учебная и научная литература

8. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. (4-е издание), издательство: ООО Издательство «Питер», 2013. – 896с.
9. Modeler: метод. указ. / Сост.: Ю.А. Соловьева, Н.Б.Соколова; СибГИУ. – Новокузнецк, 2009.
10. Вендров А.М. Объектно-ориентированный анализ и проектирование с использованием языка UML и Rational Rose, практикум. – 2004 – 54 с.
11. Грошев А.С. Разработка информационной системы — СПб.: БХВ-Петербург, 2001. — 656 с.
12. Калянов Г.Н.: Моделирование, анализ, реорганизация и автоматизация биз-нес-процессов. - М.: Финансы и статистика, 2007 – 240 с.
13. Карпова Т. С. Базы данных: модели, разработка, реализация. — СПб.: Питер, 2001. — 304 с.
14. КватраниТ.. Rational Rose 2000 и UML. Визуальное проектирование: Пер. с англ. – М.: ДМК Пресс, 2001. – 176 с.: ил. (Серия «Объектно-ориентированные технологии в программировании»).
15. Климов А. П. C# Советы программистам. БХВ-Петербург, 2012. – 103 с.
16. Копейкин М.В. Базы данных. Основы SQL реляционных баз данных. Учебное пособие/ Копейкин М.В., Спиридонов В.В., Шумова Е.О.; ред. Садчикова И.Н. – СПб, 2005.
17. Леоненков А. Самоучитель UML. – СПб.: БХВ – Петербург, 2007.
18. Маглинец Ю. А. Анализ требований к автоматизированным информационным системам. – Бином, 2008.

19. Маклаков С.В. Моделирование бизнес – процессов с BPwin 4.0 - М.: Диалог – МИФИ, 2002.
20. Маклаков С.В. Создание информационных систем с AllFusion Modeling Suite. - 2003 - 427 с.
21. Маслов А.В. Практикум по проектированию информационных систем в экономике: учеб. пособие / Маслов А. В., Исаков В. В. – Томск: изд-во Томского политехнического университета, 2009. – 193 с.
22. Методология функционального моделирования IDEF0: руководящий документ. – М.: ИПК Издательство стандартов, 2000 г. – Официальное издание. – 75 с.
23. Овчинникова Е.В. Моделирование бизнес – процессов с помощью AlFusion Process Modeler, – Уральский государственный университет путей сообщения (УрГУПС), 2007. – 105 с.
24. Работа с базами данных на языке C#. Технология ADO.NET: Учебное пособие сост. О. Н. Евсеева, А. Б. Шамшев. - Ульяновск: УлГТУ, 2009. - 170 с.
25. Расчет показателей экономической эффективности дипломного проекта: методические указания / Абрамов А.А., Гришина Н.И. – Новгород, 2013. – 23с.
26. Соловьева Ю.А. Создание модели данных с помощью AllFusion ERwin Data
27. Технология программирования: учебное пособие / Ю.Ю. Громов, О.Г. Иванова, М.П. Беляев, Ю.В. Минин. – Тамбов: Изд-во ФГБОУ ВПО «ТГТУ», 2013. – 172 с. – экз.

Электронные ресурсы

28. Microsoft Developer Network. Разработка приложений в Visual Studio 2017. [Электронный ресурс]. – Режим доступа: msdn.microsoft.com
29. Инструментальные средства программирования [Электронный ресурс]. –URL: http://life-prog.ru/1_18845_instrumentalnie-sredstva-programmirovaniya.html
30. Основные принципы разработки приложений [Электронный ресурс]. – URL: http://life-prog.ru/2_18421_glava--osnovnie-printsipi-razrabotki-prilozheniy.html
31. Программирование на C#. Алгоритмы, исходные коды. [Электронный ресурс]. – Режим доступа: www.codeproject.com

ПРИЛОЖЕНИЯ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«СЕВЕРО-ВОСТОЧНЫЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ ЕСТЕСТВЕННЫХ НАУК И МАТЕМАТИКИ

КАФЕДРА ИНФОРМАТИКИ

УТВЕРЖДАЮ

Руководитель выпускной

квалификационной работы

_____Сироткин А.В.

«_____» _____ 2018 года

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку автоматизированной информационной системы «Табачный склад» для предприятия АО «ТК «Мегаполис» в рамках выпускной

квалификационной работы

Техническое задание на 6 листах

Действует с «_____» _____ 201_ года

РАЗРАБОТЧИК

Студент 4 курса группы ПИ-41

_____ Прокопюк С.В.

«_____» _____ 2018 года

Магадан 2018

Оглавлени

e

1. Общие сведения.....114

1.1. Наименование системы....114
1.2. Основание дляпроведения работ....114
1.3. Наименование организации – Заказчика и Разработчика....114
1.4. Плановые сроки начала и окончания работы....114
1.5. Источники и порядок финансирования....114
1.6. Порядок оформления и предъявления заказчику результатов работы....114
2. Назначение и цели создания системы....115
2.1. Назначение системы....115
2.2. Цели создания системы....115
3. Характеристика объектов автоматизации....115
4. Требования к системе....115
4.1. Требование к функциям системы....115
4.2. Требования к системе в целом....116
4.2.1. Требования к структуре системы....116
4.2.2. Требования к режимам функционирования системы....116
4.2.3. Требования к численности и квалификации персонала и режиму его работы....116
4.2.4. Требования к надежности....116
4.2.5. Требования по обеспечению безопасности информации....116
4.2.6. Требования к упровлению доступом....117
4.2.7. Требования к регистрации событий безопасности....117
4.2.8. Требования к обеспечению целосности системы....117
4.2.9. Требования по эргономике и технической эстетике....117
4.2.10. Требования по сохранению информации при авариях....117
4.2.11. Требования по стандартизации и унификации....117
4.2.12. Требования к программному обеспечению системы....118
5. Состав и содержание работ по созданию системы....119

5.1. Основные этапы создания АИС «Табачный склад».....119
6. Перечень подлежащих разработке комплектов и видов документов.....119
7. Источники разработки.....119
8. Лист согласования технического задания.....120

1. Общие сведения

1.1. Наименование системы

Полное наименование системы

Автоматизированная информационная система «Табачный склад»

Краткое наименование

АИС «Табачный склад»

1.2. Основание для проведения работ

На основание приказа об утверждение тем выпускных квалификационных работ №1569/ст от 08.12.2017 г. ФГБОУ ВО «Северо-Восточный государственный университет».

1.3. Наименование организации – Заказчика и Разработчика

Заказчик

Заказчик: АО «ТК «Мегаполис».

Разработчик

Разработчик: Прокопюк Сергей Валерьевич

1.4. Плановые сроки начала и окончания работы

Плановое начало выполнения работ: с 19 ноября 2018 года. Плановое окончание работ: 29 декабря 2018 года.

1.5. Источники и порядок финансирования

Источники финансирования отсутствуют. Работа выполняется в рамках выпускной квалификационной работы.

1.6. Порядок оформления и предъявления заказчику результатов работы.

Предъявление результатов работы выполняется в соответствии с учебным планом.

2. Назначение и цели создания системы

2.1. Назначение системы

АИС «Табачный склад» предназначена для автоматизации следующих процессов:

1. Поступление товара на склад;
2. Отгрузка товара со склада;
3. Формирование отчетов по деятельности склада.

2.2. Цели создания системы

Основными целями создания системы являются:

1. Сокращение временных затрат на учет приемки и отгрузки продукции;
2. Упрощение составления отчетов;
3. Повышение достоверности получаемой, обрабатываемой и хранимой информации;
4. Сокращение временных затрат на поиск информации;
5. Повышение уровня удовлетворенности сотрудников за счет улучшения условий труда.

3. Характеристика объектов автоматизации

Задача заключается в разработке программного продукта табачного склада, позволяющего автоматизировать обработку и хранения данных по складу.

Для создания такой информационной системы анализируется предметная область. Предметной областью данного проекта является область деятельности табачного склада АО «ТК Мегаполис».

Объектом автоматизации является: табачный склад АО «ТК Мегаполис».

4. Требования к системе

4.1. Требование к функциям системы

АИС «Табачный склад» должна обладать следующими функциями:

1. Ввод приемки, изменение приемки, удаление;
2. Ввод отгрузки, изменение, удаление;
3. Формирование отчета по отгрузке и по приемке;
4. Формование отчета по остаткам на складе;
5. Ведение справочников номенклатуры товара, контрагентов, складов;
6. Выгрузка отчетных документов в Microsoft Excel.

4.2. Требования к системе в целом

4.2.1. Требования к структуре системы

В АИС «Табачный склад» должна быть предусмотрена возможность защиты данных от несанкционированного внесистемного доступа.

- Система должна поддерживать единую технологию обработки и предоставления данных;
- Система должна функционировать как единое целое;
- Система должна быть реализована по принципу однократного ввода данных;
- Использование в рамках системы единого набора инструментальных средств;
- Открытость структур хранения информации.

4.2.2. Требования к режимам функционирования системы

АИС «Табачного склада» должна предоставлять два режима работы:

1. Режим администратора.
2. Режим пользователя.

4.2.3. Требования к численности и квалификации персонала и режиму его работы

Количество пользователей АИС «Табачный склад» определяется количеством специалистов штата предприятия. Для организации технического обслуживания системы следует определить ответственного работника, выполняющего функцию администратора системы и имеющего доступ к административному режиму работы системы.

Пользователи должны иметь базовые навыки работы с операционными системами Microsoft (любой из версий: Microsoft Windows XP, 7, 8, 10). Техническое обслуживание системы должно выполняться специалистами, получившими соответствующую квалификацию и навыки выполнения указанных работ.

4.2.4. Требования к надежности

Система должна иметь возможность восстановления в случае сбоя в работе. Время восстановления работоспособности при сбоях не должно превышать одного рабочего дня. Должно осуществляться разграничение прав доступа к системе и её программным модулям.

4.2.5. Требования по обеспечению безопасности информации

В системе должны быть предусмотрены программные инструменты по обеспечению безопасности информации:

- Идентификация пользователей;
- Управление доступом;

4.2.6. Требования к управлению доступом

- Управление учетными записями пользователей;
- Реализация необходимых методов, типов (чтение, запись, выполнение или иной тип) и правил разграничения доступа;
- Аутентификацию пользователя.

4.2.7. Требования к регистрации событий безопасности

- Определение событий безопасности, подлежащих регистрации, и сроков их хранения;
- Определение состава и содержания информации о событиях безопасности, подлежащих регистрации;
- Сбор, запись и хранение информации о событиях безопасности в течение установленного времени хранения.

4.2.8. Требования к обеспечению целостности системы

- Контроль целостности программного обеспечения;
- Ограничение прав пользователей по вводу информации в систему;
- Контроль ошибочных действий пользователей по вводу и предупреждение пользователей об ошибочных действиях.

4.2.9. Требования по эргономике и технической эстетике

- Визуальное взаимодействие пользователя с системой должно строиться на основе интуитивно-понятного интерфейса;
- Для наиболее частых операций должны быть предусмотрены «горячие» клавиши;
- Размер шрифта должен быть удобен для чтения;
- Цветовая палитра должна быть комфортна для длительной работы с интерфейсом, все надписи должны быть отчётливо читаемы.

4.2.10. Требования по сохранению информации при авариях

В системе должен быть предусмотрен механизм резервного копирования данных.

4.2.11. Требования по стандартизации и унификации

Взаимодействие пользователей с системой должен осуществляться посредством визуального графического интерфейса. Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Интерфейс должен соответствовать современным эргономическим требованиям и обеспечивать удобный доступ к основным функциям и операциям системы.

Все надписи экранных форм, а также сообщения, выдаваемые пользователю должны быть на русском языке.

Экранные формы должны проектироваться с учетом требований унификации:

- Все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне;
- Для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы;
- Термины, используемые для обозначения типовых операций (добавление информационной сущности, редактирование поля данных), а также последовательности действий пользователя при их выполнении, должны быть унифицированы;
- Внешнее поведение сходных элементов интерфейса (реакция на наведение указателя «мыши», переключение фокуса, нажатие кнопки) должны реализовываться одинаково для однотипных элементов.

4.2.12. Требования к программному обеспечению системы

При проектировании и разработке системы необходимо максимально эффективным образом использовать ранее закупленное программное обеспечение.

Информационная система может работать под управлением семейства операционных систем Windows (Windows 7, Windows 8 и т.д.) с установленным пакетом MS OFFICE (MS WORD).

При разработке программного продукта используются:

- Система управления базами данных (СУБД) – Microsoft SQL Server 2016;
- Среда объектно-ориентированного программирования – Microsoft Visual Studio 2017 и язык программирования C#;
- CASE - средство – Rational Rose.

Программа должна иметь инсталлятор. Установленная версия программы должна содержать все необходимые для работы библиотеки и запускаться на всех версиях Microsoft Windows, начиная с Windows 7.

Все функциональные кнопки, выполненные в виде иконок должны содержать текстовое описание, отображаемое в виде всплывающей подсказки.

5. Состав и содержание работ по созданию системы

5.1. Основные этапы создания АИС «Табачный склад»

План работ по созданию системы АИС «Табачный склад»		
№ п/п	**Наименование этапа работ**	**Продолжительность**
1	Анализ предметной области.	3 недель
2	Моделирование.	4 недель
3	Проектирование	3 недель
4	Ввод в действие	2 неделя

6. Перечень подлежащих разработке комплектов и видов документов

1. Модель IDEF0;
2. Модель IDEF1;
3. Функциональная архитектура классов (UML);
4. Текстовое описание моделей вариантов использования;
5. Описание классов проектирования.

7. Источники разработки

Настоящее Техническое Задание разработано на основе следующих документов и информационных материалов.

1. ГОСТ 34.003-90 «Автоматизированные системы. Термины и определения».
2. ГОСТ 34.201-89 «Виды, комплектность и обозначение документов при создании автоматизированных систем».
3. ГОСТ 34.601-90 – «Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания».
4. ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы».

8. Лист согласования технического задания

СОСТАВЛЕНО	**СОГЛАСОВАНО**
Студент 4 курса	КТН, доцент
группы ПИз -41	
_____ Прокопюк С.В.	_____ Сироткин А.В.
«_____»_____ 2018 года	«_____»_____ 2018 года