

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
КАЗАХСТАН
СЕВЕРО-КАЗАХСТАНСКИЙ УНИВЕРСИТЕТ ИМ. М. КОЗЫБАЕВА
Факультет инженерии и цифровых технологий
Кафедра "Информационно-коммуникационные технологии"

СЕМЕСТРОВЫЙ ПРОЕКТ

На тему: «Разработка ИС «Анализ продаж»»

Выполнили студенты
группы АПО-20-2

Ленгард С.С.
Нартбаев И.А.

Доктор PhD,
доцент кафедры ИКТ

Астапенко Н.В.

Петропавловск, 2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
Аналитическая часть	4
1.1 Характеристика/описание объекта исследования.	4
1.2 Обзор аналогичных проектов.	4
1.3 Техническое задание на разработку системы «Анализ продаж» на VBA. ..	5
1.4 Функциональное моделирование (IDEF0, IDEF3).....	7
2 Проектная часть.....	11
2.1 Проектирование информационных потоков (DFD).	11
2.2 Проектирование средствами UML (6 видов диаграмм).	12
2.3 Проектирование БД (инфологическое проектирование – ERD; дatalogическое проектирование – описание таблиц и схемы данных).....	21
2.4 Проектирование пользовательских интерфейсов.....	23
ЗАКЛЮЧЕНИЕ	33
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ.....	34

ВВЕДЕНИЕ

В современном бизнесе анализ продаж является одним из ключевых инструментов для оценки эффективности и успешности коммерческой деятельности компании. Это особенно актуально для отраслей, где продукция имеет различные характеристики и требует постоянного мониторинга и принятия оперативных решений. В связи с этим, автоматизация анализа продаж позволяет существенно повысить результативность работы менеджеров и специалистов, предоставляя им подробные и точные данные для принятия обоснованных решений.

Цель данного проекта: Разработка и внедрение автоматизированной системы для анализа продаж на основе пользовательской формы в Microsoft Excel.

Задачи проекта:

- Создание пользовательской формы для ввода и обработки данных о продажах и поступлениях товаров.
- Визуализация результатов анализа с использованием диаграмм и графиков.
- Генерация отчетов на основе шаблонов Microsoft Word с автоматическим заполнением информацией, полученной в результате анализа продаж.
- Обеспечение простоты и удобства использования системы для конечных пользователей.

Объект исследования: Процесс анализа продаж и поступлений товаров в компании.

Предмет исследования: Методы и инструменты автоматизации анализа продаж и поступлений товаров с использованием пользовательских форм и макросов в Microsoft Excel и Microsoft Word.

Аналитическая часть

1.1 Характеристика/описание объекта исследования.

Целью проекта является разработка автоматизированной системы анализа продаж на основе пользовательской формы в Microsoft Excel. Приложение будет обладать простым и интуитивно понятным интерфейсом, позволяющим пользователям быстро и легко обрабатывать данные о продажах и поступлениях товаров.

Приложение будет обладать следующими особенностями:

- Автоматический сбор и обработка данных о продажах и поступлениях товаров из листов Excel;
- Визуализация результатов анализа с использованием диаграмм и графиков для наглядного представления информации;
- Генерация отчетов на основе шаблонов Microsoft Word с автоматическим заполнением данными, полученными в результате анализа продаж;
- Возможность сравнения и анализа продаж по различным критериям, таким как даты, коды товаров, названия товаров и т.д.;
- Функция поиска и определения названия товара на основе его кода, используя данные из других листов Excel;
- Простота и удобство использования системы для конечных пользователей.

Система будет предназначена для использования менеджерами и специалистами, ответственными за анализ продаж и поступлений товаров в компании. Она позволит пользователям быстро и точно обрабатывать большие объемы информации, а также генерировать отчеты для дальнейшего использования и анализа.

Таким образом, автоматизированная система анализа продаж на основе пользовательской формы в Microsoft Excel поможет пользователям быстрее и эффективнее обрабатывать данные о продажах и поступлениях товаров, что повысит результативность их работы и поможет принимать обоснованные решения на основе точной и актуальной информации.

1.2 Обзор аналогичных проектов.

В рамках данного проекта по анализу продаж, основанным на пользовательской форме в Microsoft Excel, можно выделить следующих основных конкурентов и аналогов:

Tableau: Tableau - это мощный инструмент анализа данных и визуализации, который предлагает широкий спектр функций для обработки, анализа и представления данных. Он может быть использован для анализа продаж и поступлений товаров, однако может быть излишне сложным и дорогостоящим для многих пользователей.

Power BI: Power BI - это платформа анализа данных от Microsoft, которая предлагает возможности анализа и визуализации данных, а также интеграцию с другими продуктами Microsoft, такими как Excel. Он предлагает набор функций, аналогичных Tableau, но также может быть сложным в использовании для некоторых пользователей.

QlikView: QlikView - это еще один популярный инструмент анализа данных, который предлагает функции обработки, анализа и визуализации данных. Он предлагает интерактивный и интуитивно понятный интерфейс, но может быть дорогостоящим для некоторых пользователей.

Google Sheets: Google Sheets предлагает возможности анализа и обработки данных, а также интеграцию с другими сервисами Google. Он является простым и доступным инструментом для анализа данных, но может не иметь всех функций и возможностей, предлагаемых другими конкурентами.

Apache OpenOffice Calc: Calc - это электронная таблица, входящая в состав бесплатного пакета Apache OpenOffice. Он предлагает функции обработки и анализа данных, а также возможность создания отчетов. Однако его функционал может быть ограниченным по сравнению с другими инструментами анализа данных.

В целом, эти инструменты предлагают различные функции и возможности для анализа продаж и поступлений товаров. Однако наш проект, основанный на пользовательской форме в Microsoft Excel, предлагает простоту, доступность и интеграцию с Microsoft Office, что делает его удобным и эффективным для пользователей, которые не нуждаются в сложных и дорогостоящих решениях.

1.3 Техническое задание на разработку системы «Анализ продаж» на VBA.

Техническое задание на разработку приложения «Анализ продаж» на основе Excel и VBA

1. Общие сведения:

Приложение «Анализ продаж» представляет собой систему на языке VBA, предназначенную для автоматизации процесса анализа продаж компании. Приложение позволяет создавать пользовательские формы для ввода и обработки данных о продажах и поступлениях товаров, визуализировать результаты анализа в виде диаграмм и графиков, а также генерировать отчеты на основе шаблонов Microsoft Word с автоматическим заполнением информацией, полученной в результате анализа продаж. Цель приложения – повышение эффективности и точности анализа продаж, упрощение процесса принятия решений и обеспечение удобства использования системы для конечных пользователей.

2. Основные функции приложения:

- Создание пользовательских форм для ввода и обработки данных о продажах и поступлениях товаров;
- Визуализация результатов анализа в виде диаграмм и графиков;
- Генерация отчетов на основе шаблонов Microsoft Word с автоматическим заполнением информацией, полученной в результате анализа продаж;
- Обеспечение простоты и удобства использования системы для конечных пользователей.

3. Характеристики объектов автоматизации:

К объектам автоматизации системы относятся: • Список продаж; • Наименование товара; • Количество товара; • Цена товара; • Дата продажи; • Дата поступления товара.

4. Требования к функциональности приложения:

- Создание пользовательской формы для ввода и обработки данных о продажах и поступлениях товаров;
- Возможность выбора критериев для анализа продаж (например, по товару, по дате);
- Автоматический расчет суммы продаж и прибыли по выбранным критериям;
- Визуализация результатов анализа в виде диаграмм и графиков;
- Генерация отчетов на основе шаблонов Microsoft Word с автоматическим заполнением информацией, полученной в результате анализа продаж;
- Возможность сохранения и загрузки данных;
- Обеспечение простоты и удобства использования системы для конечных пользователей.

5. Требования к интерфейсу приложения:

- Интуитивно понятный и простой интерфейс;
- Возможность быстрого доступа к функциям приложения;
- Возможность настройки интерфейса под индивидуальные потребности пользователя;

- Наличие инструкции по использованию приложения.

6. Требования к надежности приложения:

- Стабильная работа приложения без сбоев и ошибок;
- Возможность быстрого восстановления работы приложения в случае возникновения ошибок.

7. Требования к безопасности приложения:

- Защита данных от несанкционированного доступа;
- Возможность резервного копирования и восстановления данных.

8. Требования к производительности приложения:

- Быстрая обработка и анализ больших объемов данных;
- Минимальные временные задержки при выполнении операций.

9. Требования к сопровождению приложения:

- Предоставление технической поддержки пользователям;

- Регулярное обновление и улучшение приложения в соответствии с потребностями пользователей и изменениями бизнес-процессов.

10. Требования к документации:

- Наличие подробной инструкции по использованию приложения;
- Наличие руководства администратора;
- Наличие руководства по разработке и модификации приложения.

11. Требования к архитектуре и инструментам разработки:

- Разработка на языке VBA;
- Использование Excel для создания пользовательских форм и обработки данных;

- Использование Microsoft Word для создания шаблонов отчетов;
- Использование современных методов разработки и тестирования программного обеспечения.

12. Требования к условиям эксплуатации:

- Приложение должно быть доступно для использования на персональных компьютерах под управлением операционных систем Windows;

- Для работы приложения должны быть установлены Microsoft Excel и Microsoft Word;

- Должны быть обеспечены достаточные ресурсы для хранения и обработки данных, а также выполнения операций приложения.

1.4 Функциональное моделирование (IDEF0, IDEF3).

IDEF0 (Integration Definition for Function Modeling) - это методология моделирования функций, процессов и систем. Эта нотация была разработана в США в рамках программы ICAM (Integrated Computer-Aided Manufacturing) для улучшения производственной деятельности компаний и стандартизации их процессов.

IDEF0 представляет собой формализованный язык, состоящий из символов и правил, которые позволяют описать функции, потоки данных и управление между функциями и данными в виде блок-схем. Основными элементами нотации являются функциональные блоки, связи между ними, потоки данных и входы/выходы.

Контекстная диаграмма показывает взаимодействие системы с внешней средой и состоит из одной функции, которая обозначает саму систему. Она показывает, что система взаимодействует с одним типом пользователей (механизм): пользователь. На входе: код товара, дата или период и номер телефона клиента. На выходе: добавление поставки или продажи, отредактированная база данных и отчет о количестве поступившего/проданного товара за заданный период. Контекстная диаграмма IDEF0 анализа продаж на VBA представлена в соответствии с рисунком 1.1.

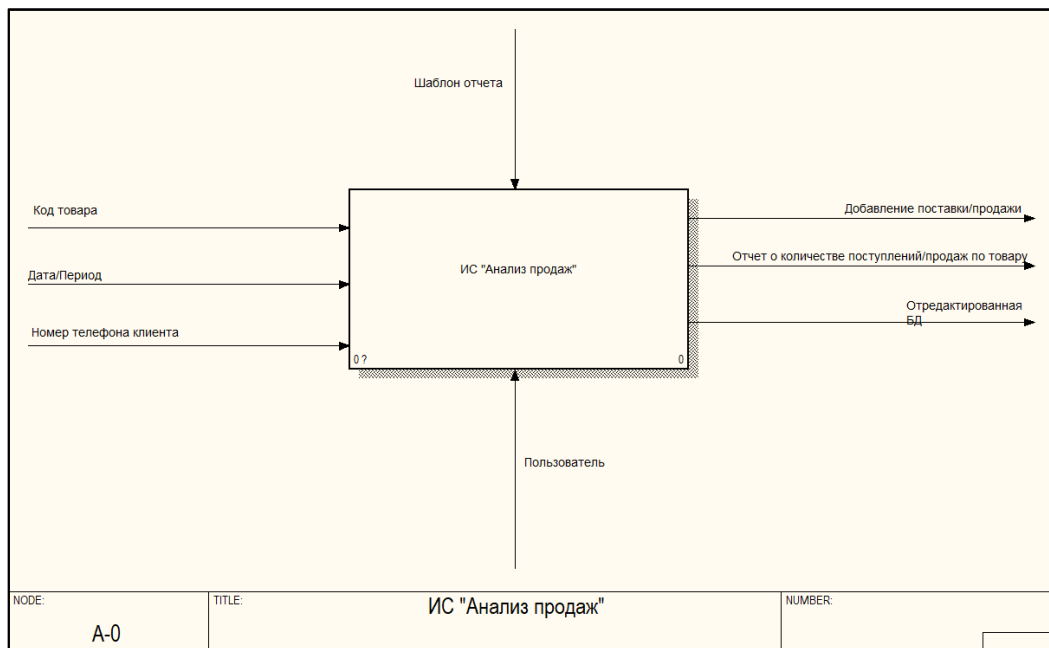


Рисунок 1.1 Контекстная диаграмма IDEF0

Диаграмма декомпозиции разбивает систему на более мелкие подсистемы, каждая из которых выполняет определенные функции. Она состоит из трех функций: изменение базы данных клиентов и доступных товаров, проверка на наличие товара в базе /на складе, использование формы для составления отчёта. Каждая из этих функций разбивается на более детализированные функции, позволяя более точно определить задачи, которые должна выполнять каждая из подсистем. Диаграмма декомпозиции 1-уровня IDEF0 представлена в соответствии с рисунком 1.2.

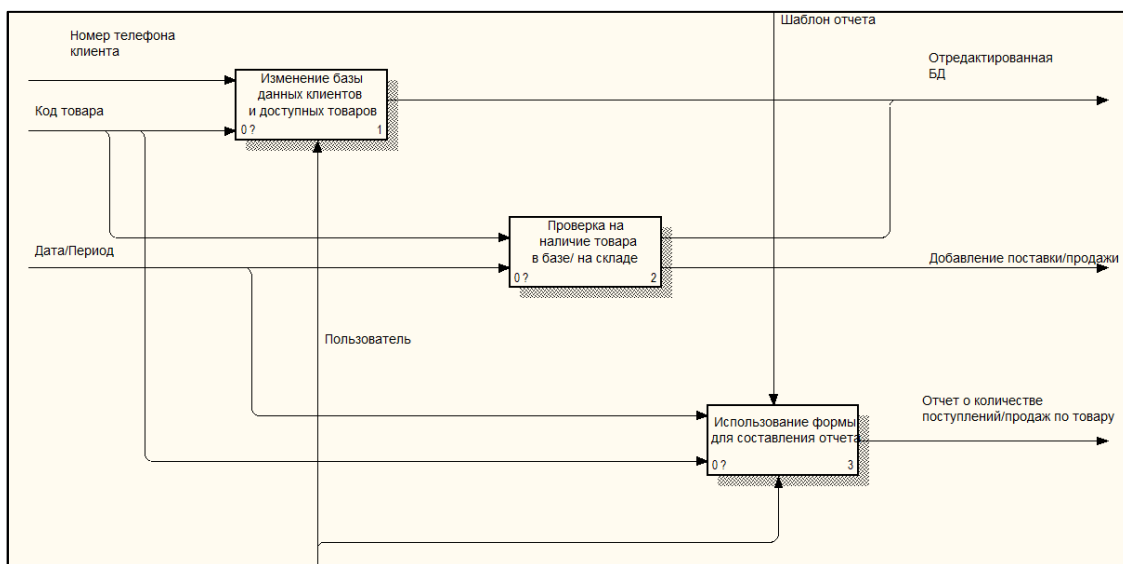


Рисунок 1.2 Диаграмма декомпозиции 1-уровня IDEF0

IDEF3 (Integrated Definition for Process Description Capture Method) является методом, который позволяет описать более детально процессы, которые определены в IDEF0. IDEF3 может быть использован для создания более подробных диаграмм действий, которые необходимы для выполнения каждой функции.

Например, для функции «изменение базы клиентов и доступных товаров» для начала нужно определить какую из баз данных мы будем менять. После выбора редактируемой БД мы производим нужные нам изменения (либо добавить, либо отредактировать, либо удалить) товар или данные о клиенте, затем сохраняем изменения. На выходе получаем сохраненную и измененную базу данных.

Диаграмма декомпозиции 2-уровня IDEF3 «изменение базы клиентов и доступных товаров» представлена в соответствии с рисунком 1.3.

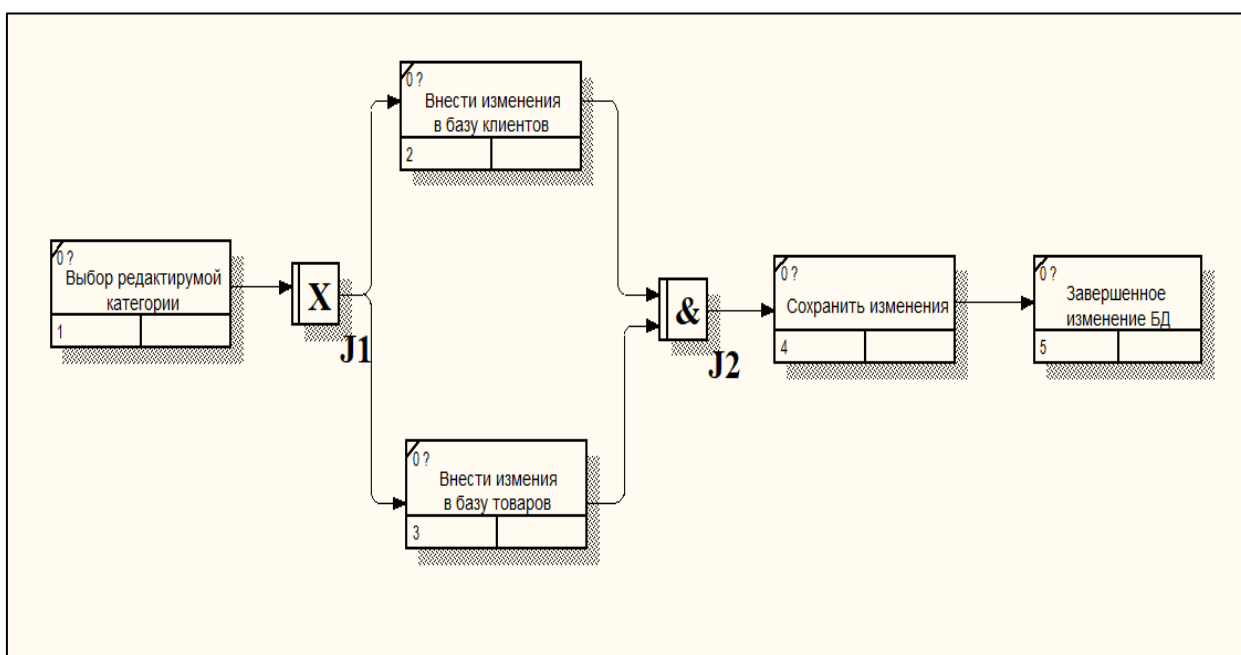


Рисунок 1.3 Диаграмма декомпозиции 2-уровня IDEF3 «изменение базы клиентов и доступных товаров»

Для функции «Проверка на наличие товара в базе/ на складе» можно определить действия, такие как выбор того, что мы хотим сделать, добавить поставку или продажу, затем мы вводим данные (код товара, дата, количество, цена покупки (для поставки) и цена продажи (для продажи), также для продажи уникальным параметром будет номер телефона клиента).

К тому же для продажи существует уникальная проверка на то, было ли достаточное количество товара для продажи на заданную дату в наличии, если не было, то об этом выйдет предупреждение.

Диаграмма декомпозиции 2-уровня IDEF3 «профиль» представлена в соответствии с рисунком 1.4.

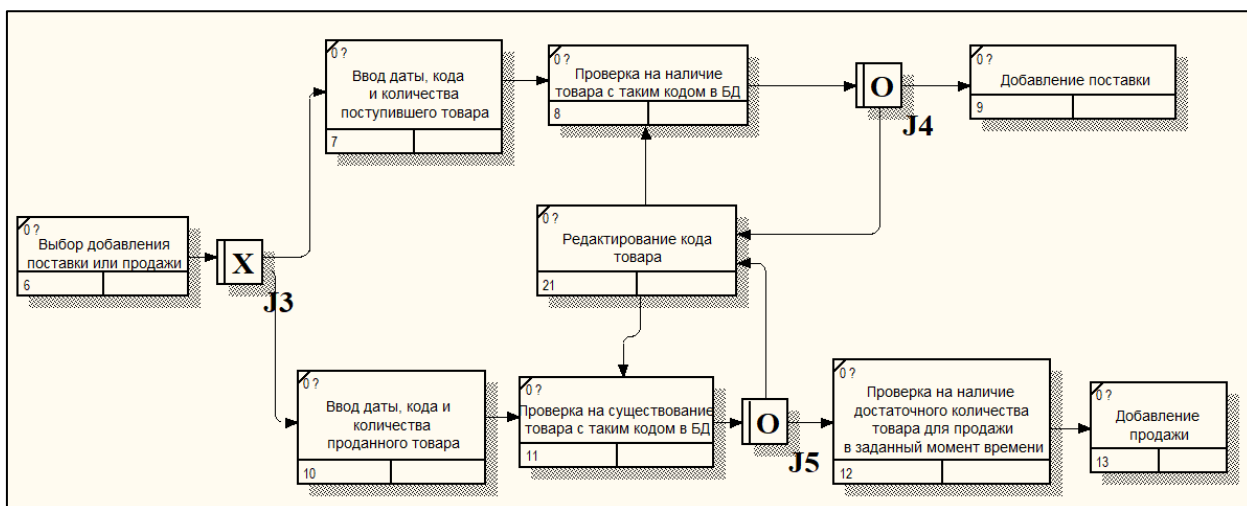


Рисунок 1.4 Диаграмма декомпозиции 2-уровня IDEF3 «профиль»

Для функции «Использование формы для составления отчета» определить порядок действий следующим образом: сначала мы вводим код товара, информацию о котором хотим получить, затем даты начала и конца периода, последующие действия выполняются самой программой. На выходе: мы получаем небольшой отчет с основной информацией по товару за заданный период.

Диаграмма декомпозиции 2-уровня IDEF3 «Использование формы для составления отчета» представлена в соответствии с рисунком 1.5.

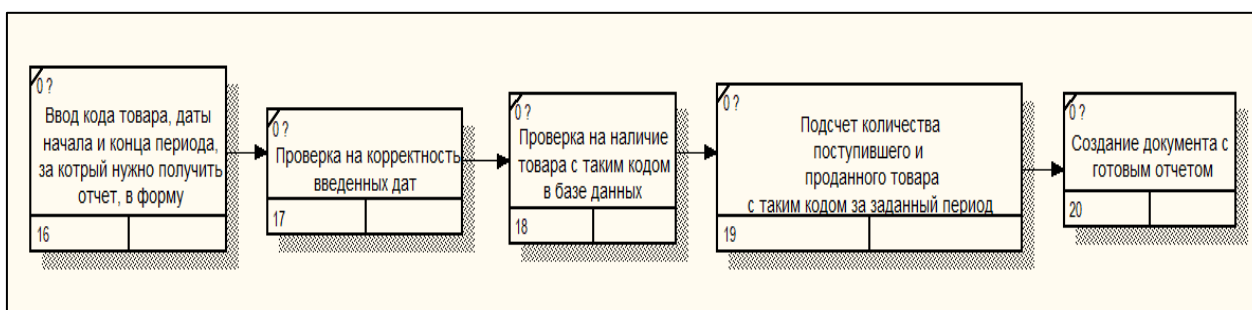


Рисунок 1.5 Диаграмма декомпозиции 2-уровня IDEF3 «Использование формы для составления отчета»

Таким образом, использование методологии IDEF0 и IDEF3 позволяет более детально определить функции и процессы, которые необходимы для создания анализа продаж на VBA. Это поможет создать более эффективный и простой в использовании программный продукт.

2 Проектная часть

2.1 Проектирование информационных потоков (DFD).

Data Flow Diagram (DFD) – это диаграмма, которая показывает поток данных и процессы, которые преобразуют эти данные. В контексте анализа продаж на VBA, DFD поможет понять, как информация проходит через систему и какие процессы нужны для обработки этой информации.

Диаграмма DFD наглядно отображает течение информации в пределах процесса или системы. Для изображения входных и выходных данных, точек хранения информации и путей ее передвижения между источниками и пунктами доставки в таких диаграммах применяются стандартные фигуры, такие как прямоугольники и круги, а также стрелки и краткие текстовые метки.

На первом уровне DFD, находится контекстная диаграмма, которая показывает взаимодействие между пользователем и системой. Контекстная диаграмма DFD представлена в соответствии с рисунком 2.1.

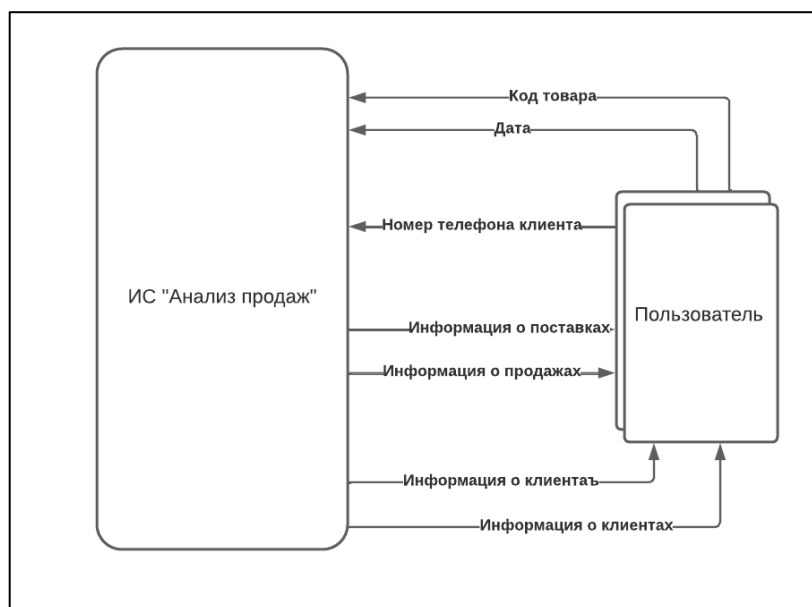


Рисунок 2.1 Контекстная диаграмма DFD

На втором уровне DFD, происходит более подробное описание процессов, которые преобразуют данные.

Подчиняясь правилам и инструкциям выбранной системы, символы отображают четыре компонента диаграммы DFD.

Внешние сущности – внешние системы, из которых поступает или куда направляется информация в результате взаимодействия с изображаемой системой. Иными словами, это источники и пункты доставки информации, которая приходит или уходит из системы. Как правило, располагаются по краям схемы.

Процессы – любые процессы, которые ведут к изменению информации и созданию выходных данных.

Хранилища данных – файлы или репозитории, где хранится информация для последующего использования, например, базы данных.

Потоки данных – маршруты, по которым информация перемещается между внешними сущностями, процессами и хранилищами данных. Потоки данных иллюстрируют взаимодействие между другими компонентами и отображаются в виде стрелок, как правило, с краткими метками. Диаграмма декомпозиции 1-уровня DFD представлена в соответствии с рисунком 2.2.

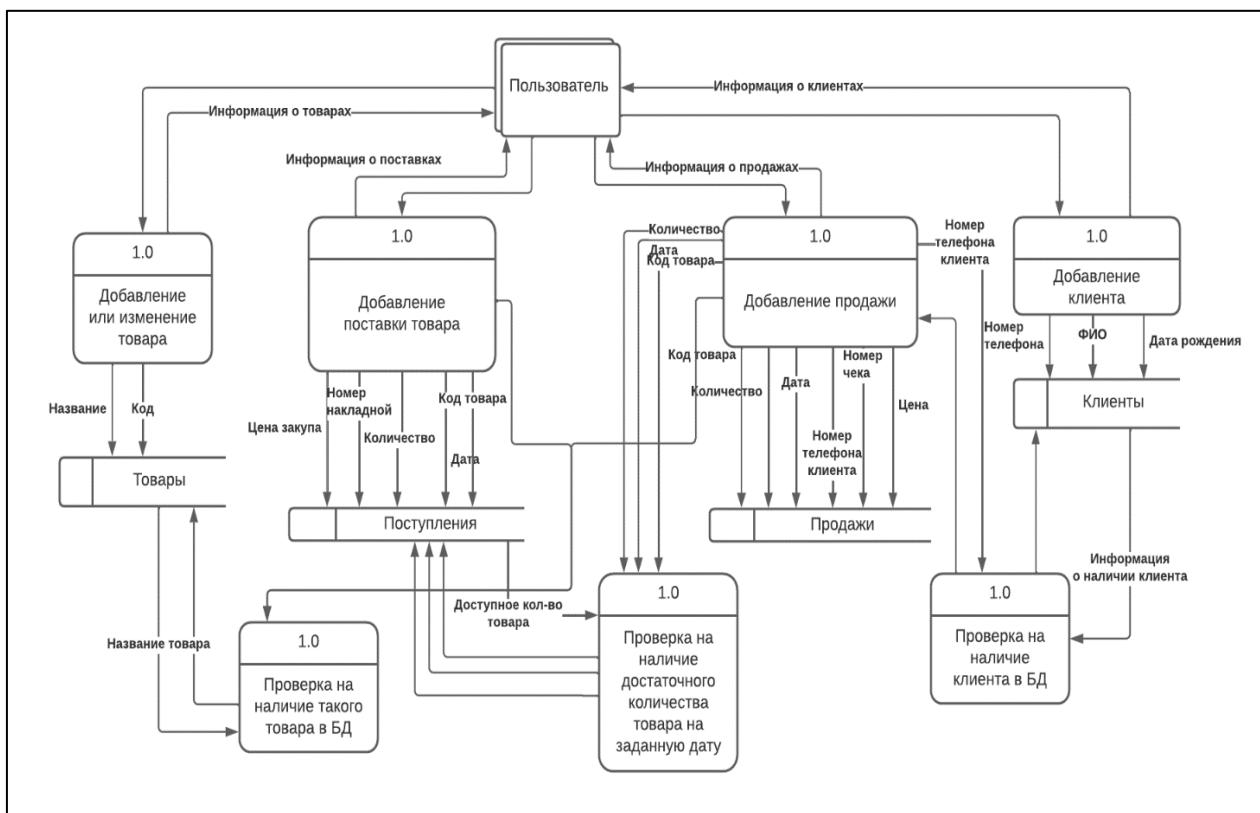


Рисунок 2.2 Диаграмма декомпозиции 1-уровня DFD

Для проекта «Анализ продаж» на диаграмме декомпозиции 1-уровня DFD:

Внешние сущности: пользователь.

Процессы: Добавление или изменение товара, Добавление поставка товара, Добавление Продажи, Добавление клиенты.

Базы данных: Товары, Поступления, Продажи, Клиенты.

Таким образом, DFD поможет понять, как информация проходит через систему, какие процессы нужны для обработки этой информации и как они связаны между собой.

2.2 Проектирование средствами UML (6 видов диаграмм).

Диаграмма вариантов использования (Use Case Diagram) – это диаграмма, которая позволяет описать функциональность системы с точки

зрения ее пользователей. Диаграмма вариантов использования состоит из актеров (Actor) – пользователей системы, и вариантов использования (Use Case) – функциональных возможностей системы, которые могут быть использованы пользователями. Варианты использования описываются в виде эллипсов и связаны с актерами линиями.

Для выражения отношений между актерами и вариантами использования применяются стандартные виды отношений.

Отношение ассоциации применительно к диаграммам вариантов использования служит для обозначения специфической роли актера в отдельном варианте использования. Другими словами, ассоциация определяет семантические особенности взаимодействия актеров и вариантов использования в графической модели системы. Таким образом, это отношение устанавливает, какую конкретную роль играет актер при взаимодействии с экземпляром варианта использования.

Отношение расширения между вариантами использования обозначается пунктирной линией со стрелкой (вариант отношения зависимости), направленной от того варианта использования, который является расширением для исходного варианта использования. Данная линия со стрелкой помечается ключевым словом «extend» («расширяет»).

Отношение включения между двумя вариантами использования указывает, что поведение одного варианта использования включается в качестве составного компонента в последовательность поведения другого варианта использования. Графически данное отношение обозначается пунктирной линией со стрелкой (вариант отношения зависимости), направленной от базового варианта использования к включаемому. При этом данная линия со стрелкой помечается ключевым словом «include» («включает»).

Для проекта «Анализ продаж» диаграмма вариантов использования включает:

Актеры: пользователь.

Варианты использования: работа с продажами (включает в себя просмотр записей продаж; удаление и добавление продаж расширяет); работа с поставками (включает в себя просмотр записей поставок (включает в себя просмотр записей продаж; удаление и добавление поставок расширяет); работа с клиентами (включает в себя просмотр записей с информацией о клиентах, расширяет добавление клиента, удаление клиента и редактирование информации о нем); работа с товарами (включает в себя просмотр записей с товарами, расширяет добавление/редактирование и удаление товара)

Диаграмма вариантов использования представлена в соответствии с рисунком 2.3.

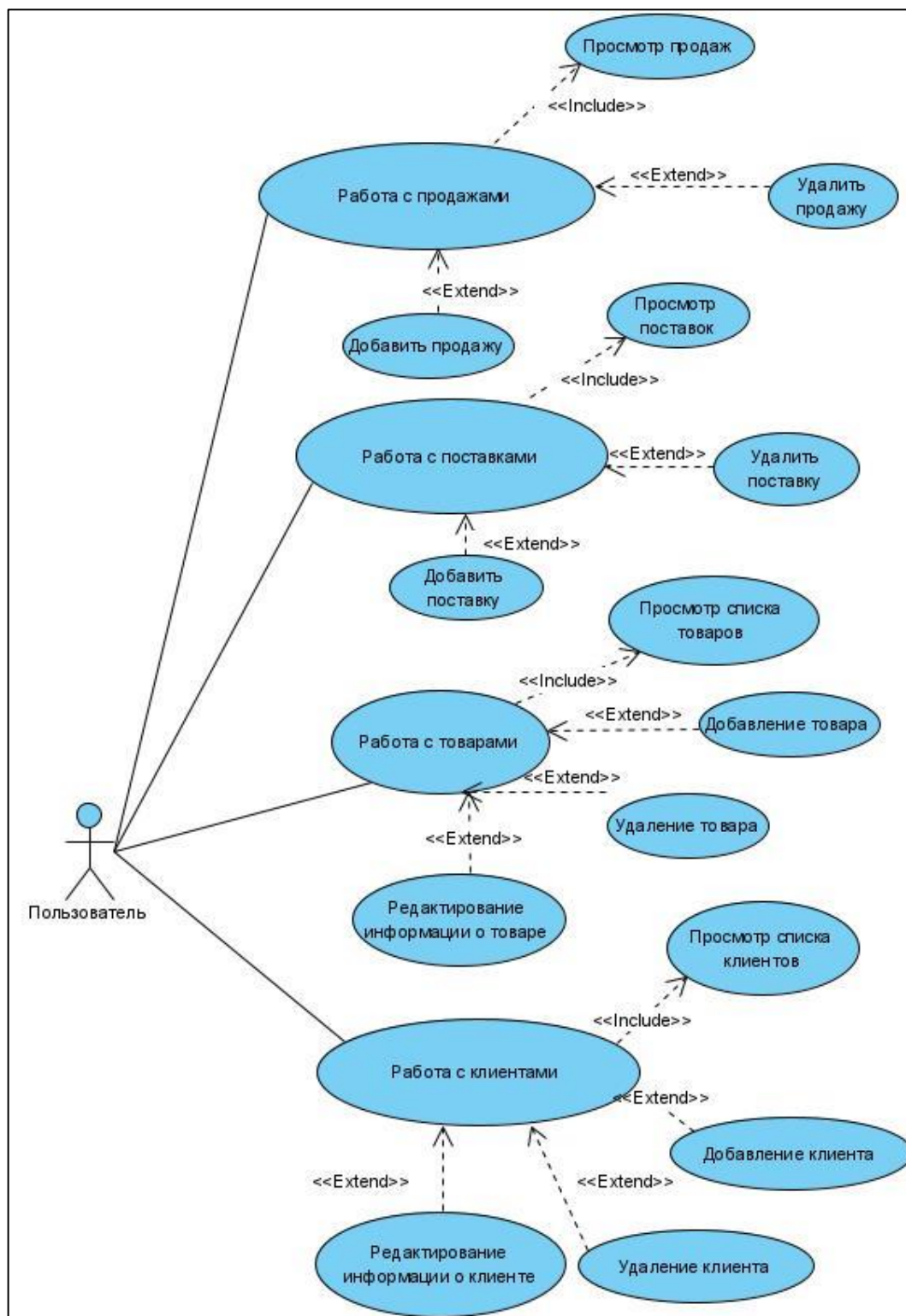


Рисунок 2.3 Диаграмма вариантов использования

Диаграммы классов при моделировании объектно-ориентированных систем встречаются чаще других. На таких диаграммах отображается множество классов, интерфейсов, коопераций и отношений между ними. Диаграмма классов служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного

программирования. Кроме того, диаграммы классов составляют основу еще двух диаграмм – компонентов и развертывания.

Диаграмма классов может отражать различные взаимосвязи между отдельными сущностями предметной области, такими как объекты и подсистемы, а также описывает их внутреннюю структуру и типы отношений.

Основными элементами являются классы и связи между ними. Классы характеризуются при помощи атрибутов и операций.

Классом называется описание совокупности объектов с общими атрибутами, операциями, отношениями и семантикой. Атрибут – это именованное свойство класса, включающее описание множества значений, которые могут принимать экземпляры этого класса. Операция – это некоторый сервис, который предоставляет экземпляр или объект класса по требованию своих клиентов (других объектов, в том числе и экземпляров данного класса). Класс может содержать любое число операций или не содержать их вовсе.

Связи:

Ассоциация (association) – представляет собой отношения между экземплярами классов.

Агрегация (aggregation) – это ассоциация типа «целое-часть». Агрегация в UML представляется в виде прямой с ромбом на конце. Ромб на связи указывает, какой класс является агрегирующим (т.е. «состоящим из»); класс с противоположного конца – агрегированным (т.е. те самые «части»).

Композиция (composition) – это такая агрегация, где объекты-части не могут существовать сами по себе и уничтожаются при уничтожении объекта агрегирующего класса. Композиция изображается так же, как ассоциация, только ромбик закрашен.

Наследование (inheritance) – это отношение типа «общее-частное». Позволяет определить такое отношение между классами, когда один класс обладает поведением и структурой ряда других классов. При создании производного класса на основе базового (одного или нескольких) возникает иерархия наследования.

Для проекта «Анализ продаж» диаграмма классов включает:

Класс «главное окно» с атрибутом Заголовок и функциями открытия окон.

Класс «окно добавления товара» с атрибутами: код товара, название и функциями добавления, редактирования и удаления записей, а также возможностью сохранить документ.

Класс «окно поступлений» с атрибутами: код товара, дата, количество, номер накладной, цена закупа и функциями добавления и удаления записей.

Класс «окно продаж» с атрибутами: код товара, дата, количество, номер чека, цена закупа, номер телефона клиента и функциями добавления и удаления записей.

Класс «окно добавление и просмотр клиентов» с атрибутами: номер телефона клиента, ФИО, дата рождения, сумма покупок и уровень скидки.

Класс «окно добавление и просмотр клиентов» с атрибутами: номер телефона клиента, ФИО, дата рождения, сумма покупок и уровень скидки.

Класс «аналитика по клиентам» с атрибутом уровень скидки и функцией «показать».

Класс «аналитика по продажам» с атрибутами код товара, дата начала и дата конца периода с функцией генерации отчета, он также имеет композицию с классом «окно построения диаграмм, у которой есть функции построения диаграммы 1 и диаграммы 2. Все остальные классы есть агрегация класса главное окно.

Диаграмма классов представлена в соответствии с рисунком 2.4.

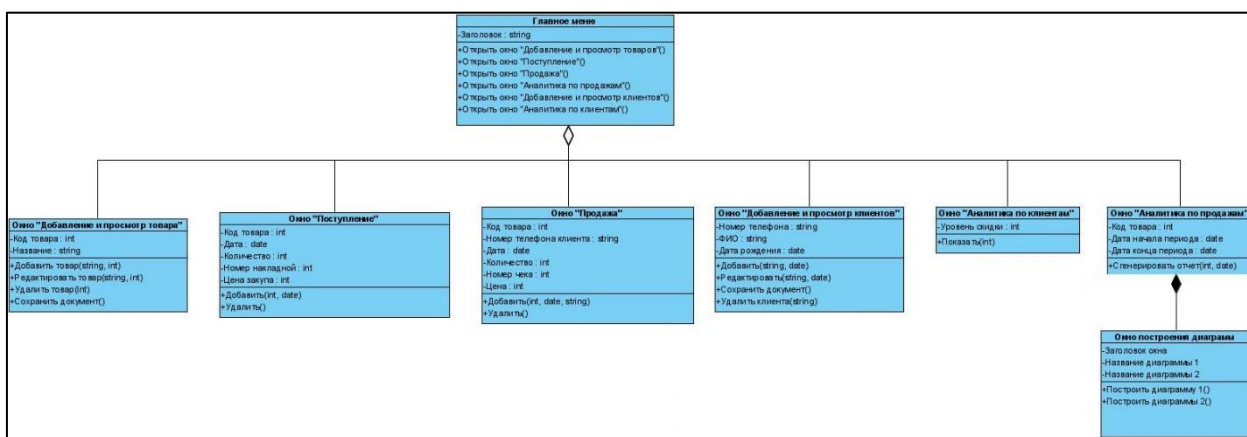


Рисунок 2.4 Диаграмма классов

Диаграмма активностей (видов деятельности) – как и диаграмма состояний, отражает динамические аспекты поведения системы. По существу, эта диаграмма представляет собой блок-схему, которая наглядно показывает, как поток управления переходит от одной деятельности к другой.

Активности на диаграмме «разбросаны» по беговым дорожкам, каждая из которых соответствует поведению одного из объектов. Благодаря этому легко определить, каким из объектов выполняется каждая из активностей. Дорожка – часть области диаграммы деятельности, на которой отображаются только те активности, за которые отвечает конкретный объект. Предназначены дорожки для разбиения диаграммы в соответствии с распределением ответственности за действия. Имя дорожки может означать роль или объект, которому она соответствует.

Для создания диаграммы деятельности используются следующие узлы:

Узел управления (control node) – это абстрактный узел действия, которое координирует потоки действий.

Начальный узел деятельности (или начальное состояние деятельности) (activity initial node) является узлом управления, в котором начинается поток (или потоки) при вызове данной деятельности извне.

Конечный узел деятельности (или конечное состояние деятельности) (activity final node) является узлом управления, который останавливает (stop)

все потоки данной диаграммы деятельности. На диаграмме может быть более одного конечного узла.

Диаграмма деятельности «Анализ продаж» представлена в соответствии с рисунком 2.5.

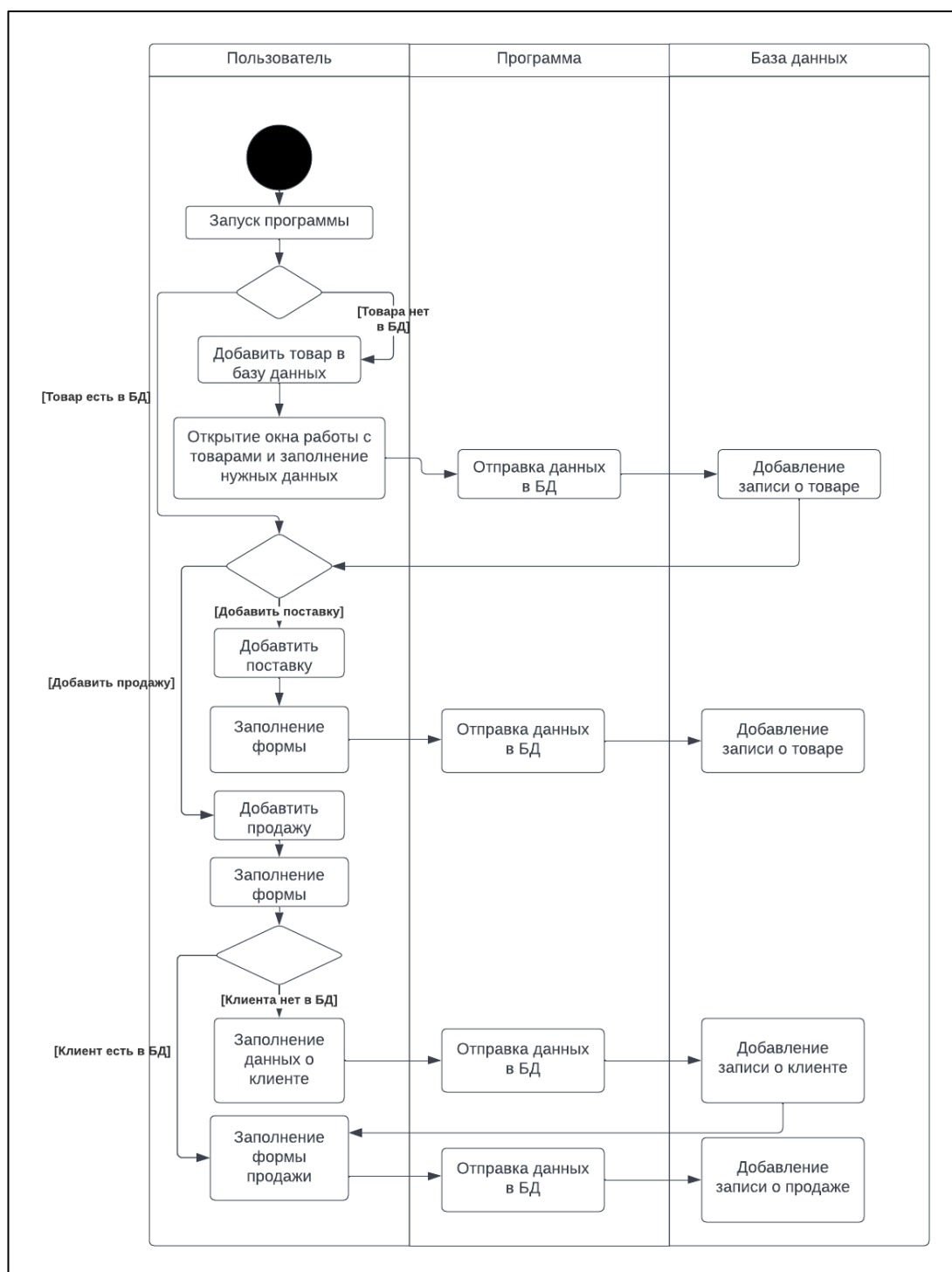


Рисунок 2.5 Диаграмма деятельности «Анализ продаж»

Диаграмма последовательности UML (Unified Modeling Language) – это вид диаграммы, используемый для описания взаимодействия между

объектами или компонентами в системе. Она показывает последовательность сообщений, передаваемых между объектами или компонентами во времени.

Диаграмма последовательности включает в себя:

- Объекты или компоненты, между которыми происходит взаимодействие;
- Сообщения, передаваемые между объектами или компонентами, обозначенные стрелками.

Временные ограничения, обозначенные вертикальными пунктирными линиями, которые показывают длительность взаимодействия между объектами.

Диаграмма последовательности часто используется для описания взаимодействия между компонентами в рамках одного процесса, например, взаимодействия между компонентами веб-приложения или между классами в программе.

Для проекта «Анализ продаж» диаграмма последовательности содержит объекты такие, как: пользователь, программа, БД.

Диаграмма последовательности представлена в соответствии с рисунком 2.8.

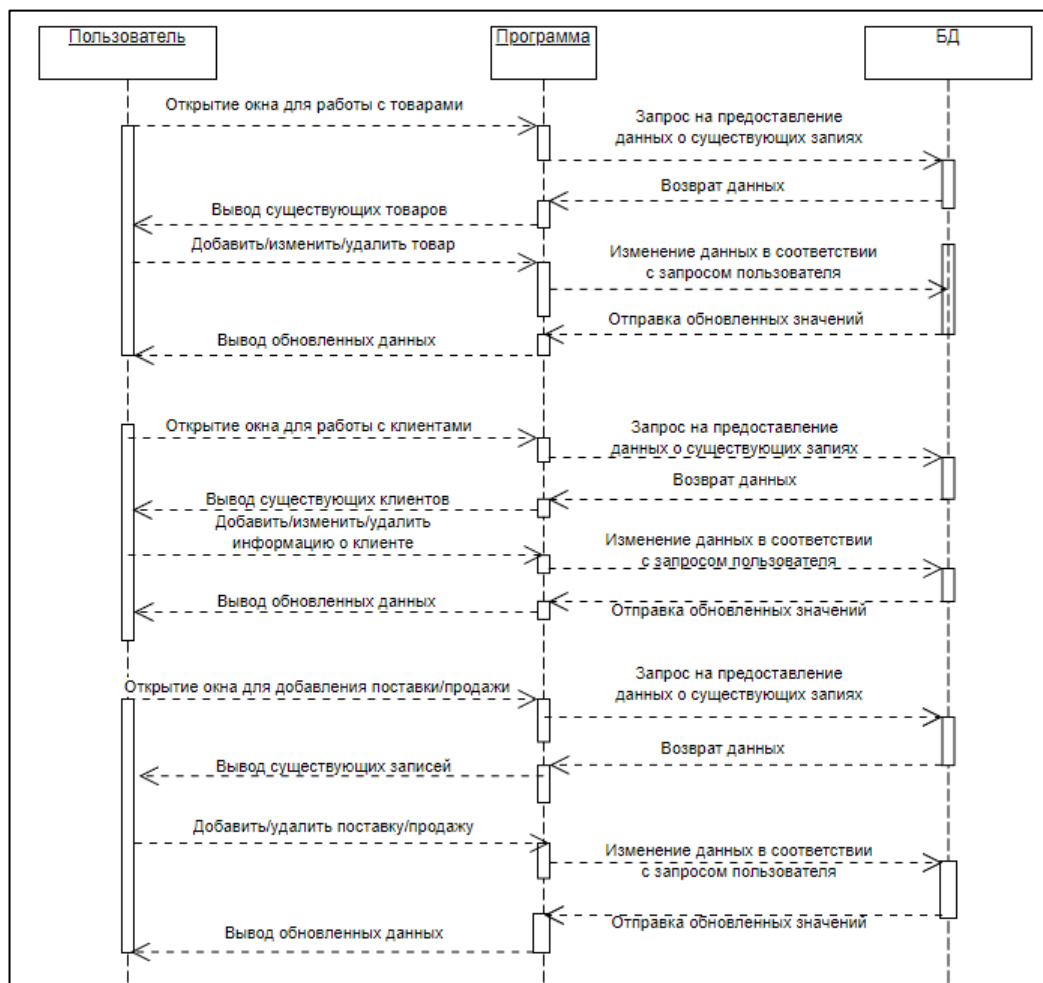


Рисунок 2.8 Диаграмма последовательности

Диаграмма компонентов – это структурная диаграмма, которая показывает, как компоненты приложения связаны друг с другом и с внешними компонентами. Она используется для описания архитектуры приложения, определяя его основные компоненты, их взаимодействие и зависимости.

Символы на диаграмме компонентов

Компонент: прямоугольник со значком компонента и стереотипом компонента.

Предоставляемый интерфейс и требуемый интерфейс: интерфейсы на компонентных схемах показывают, как компоненты соединены друг с другом и взаимодействуют друг с другом. Соединитель сборки позволяет соединить требуемый интерфейс компонента (представленный полукругом и сплошной линией) с предусмотренным интерфейсом (представленный окружностью и сплошной линией) другого компонента. Это показывает, что один компонент предоставляет услугу, которая требуется другому.

Диаграмма компонентов представлена в соответствии с рисунком 2.9.

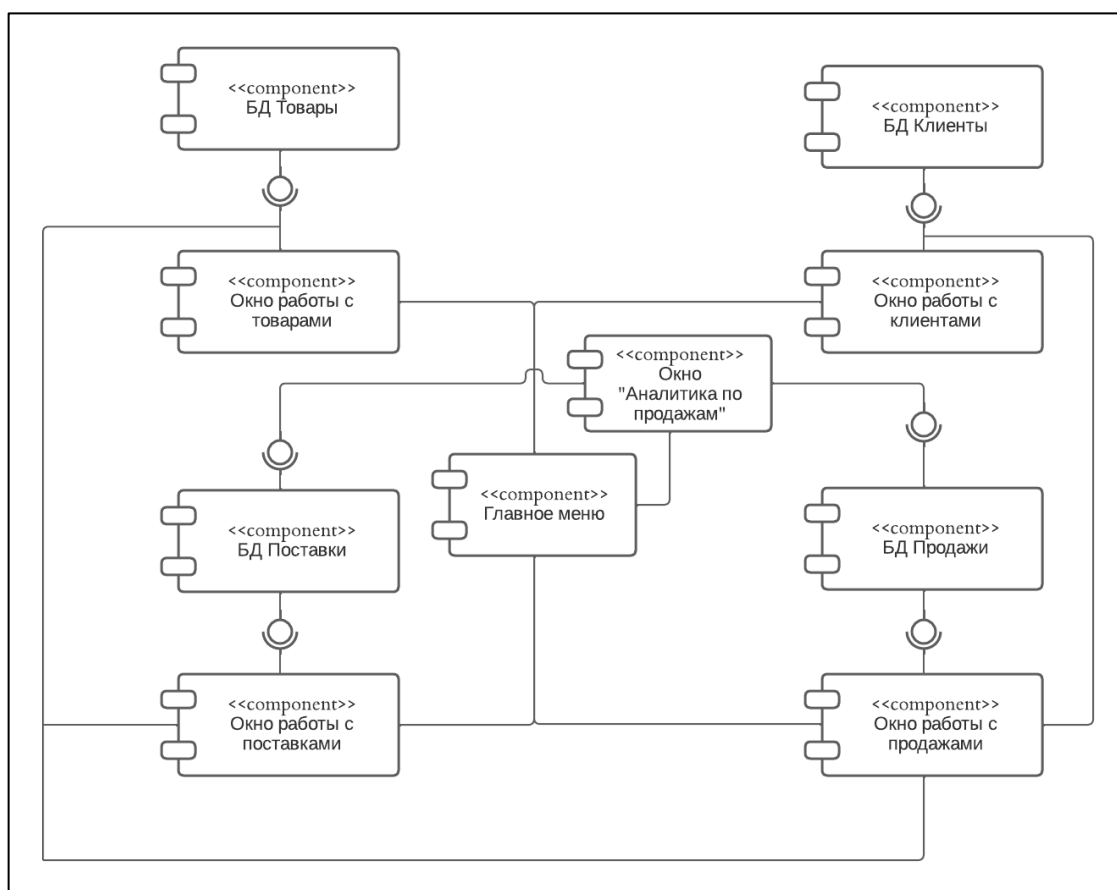


Рисунок 2.9 Диаграмма компонентов

Диаграмма состояний UML – это модель поведения, которая показывает, как объекты проходят через различные состояния во время своей жизненной цикла и как они реагируют на события, которые происходят в

системе. Диаграмма состояний используется для описания поведения сложных систем и управления их состояниями.

Диаграмма состояний состоит из различных состояний объекта, переходов между этими состояниями и действий, которые происходят при переходе между состояниями. Диаграмма состояний имеет схожую семантику с диаграммой деятельности, только деятельность здесь заменена состоянием, переходы символизируют действия. Таким образом, если для диаграммы деятельности отличие между понятиями «Деятельность» и «Действие» заключается в возможности дальнейшей декомпозиции, то на диаграмме состояний деятельность символизирует состояние, в котором объект находится продолжительное количество времени, в то время как действие моментально.

Диаграмма состояний представлена в соответствии с рисунком 2.10.

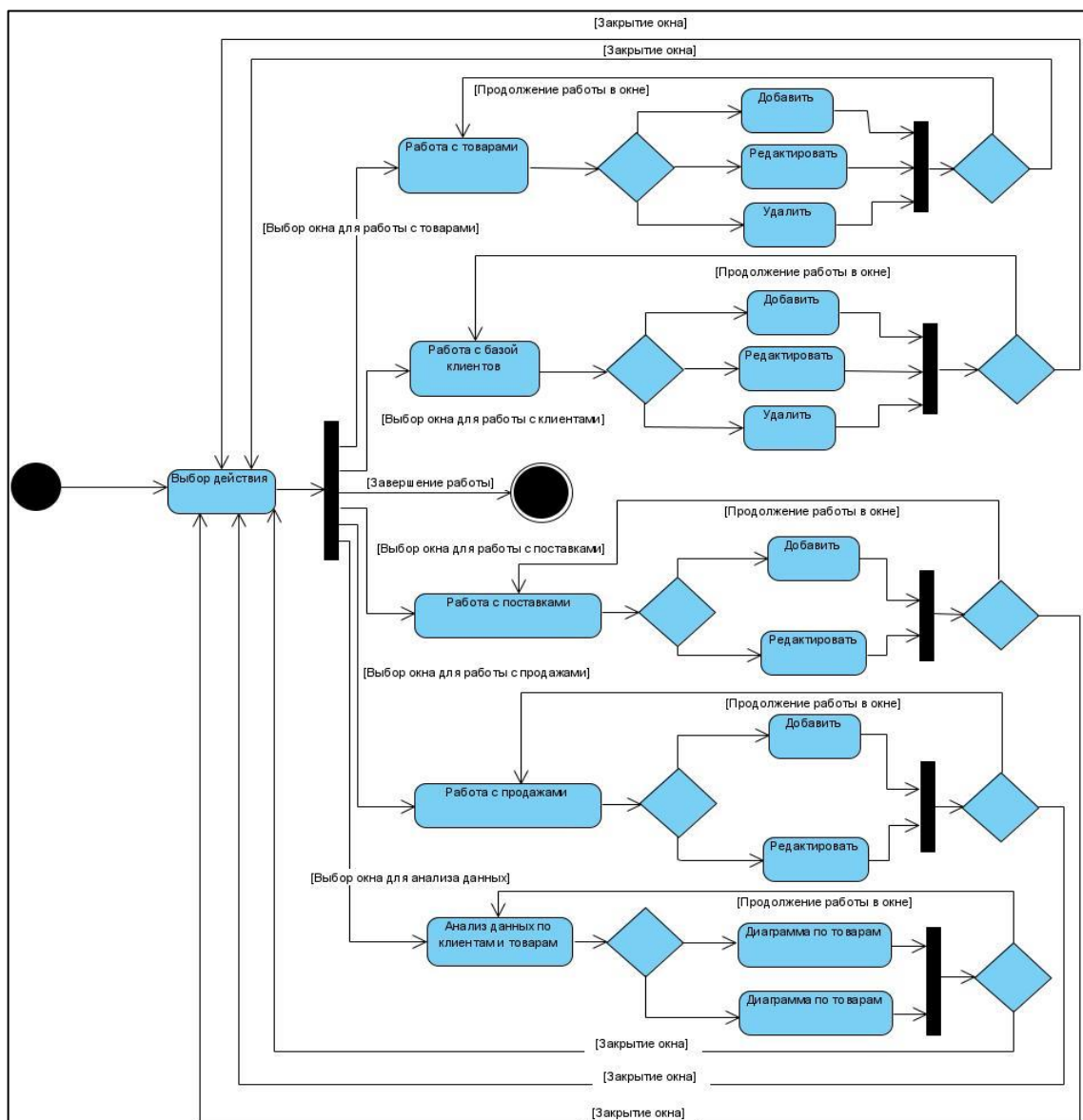


Рисунок 2.10 Диаграмма состояний

UML диаграммы представляют собой визуальное средство моделирования объектно-ориентированных систем. Использование UML диаграмм позволяет упростить процесс разработки и повысить эффективность коммуникации между различными участниками проекта.

Одна из ключевых особенностей UML диаграмм заключается в том, что они могут использоваться на разных стадиях жизненного цикла проекта. Так, на ранних стадиях проектирования UML диаграммы могут быть использованы для определения требований, анализа и проектирования системы, в то время как на более поздних стадиях – для реализации и тестирования системы.

UML диаграммы могут быть использованы для моделирования различных аспектов системы, таких как структура, поведение и взаимодействие между объектами. Каждый тип UML диаграммы имеет свои особенности и используется для определенных целей.

2.3 Проектирование БД (инфологическое проектирование – ERD; даталогическое проектирование – описание таблиц и схемы данных).

Схема «сущность-связь» (также ERD или ER-диаграмма) – это разновидность блок-схемы, где показано, как разные «сущности» (люди, объекты, концепции и так далее) связаны между собой внутри системы. ER-диаграммы (или ER-модели) полагаются на стандартный набор символов, включая прямоугольники, ромбы, овалы и соединительные линии, для отображения сущностей, их атрибутов и связей. Эти диаграммы устроены по тому же принципу, что и грамматические структуры: сущности выполняют роль существительных, а связи – глаголов.

Символы ERD-атрибутов

Под понятием «сущности» подразумеваются объекты или понятия, несущие важную информацию. Обычно обозначаются прямоугольником.

Связи используются в схемах «сущность-связь» для обозначения взаимодействия между двумя сущностями. Грамматически связи, как правило, выражаются глаголами. Обычно обозначаются ромбом.

ERD-атрибуты характеризуют сущности, позволяя пользователям лучше разобраться в устройстве базы данных. Атрибуты содержат информацию о сущностях, выделенных в концептуальной ER-диаграмме. Обычно обозначаются овалом.

В ER-диаграмме для приложения «Анализ продаж» определим такие сущности: «товары», «поставки», «продажи», «клиенты»

Связи: «учитывает».

Диаграмма ER представлена в соответствии с рисунком 2.11.

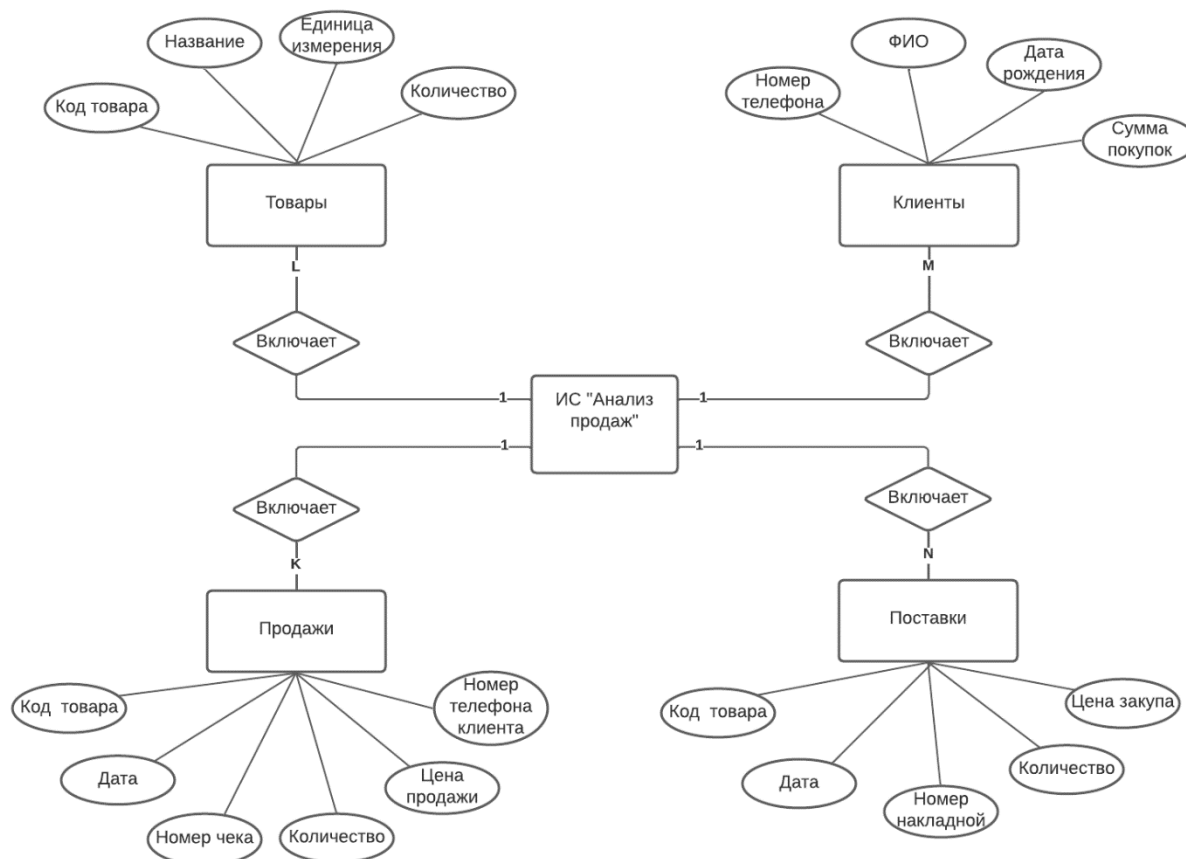


Рисунок 2.11 ER-диаграмма

Даталогическая модель – это модель базы данных, которая описывает реализацию сущностей и связей, определенных в инфологической модели, с учетом специфики используемой СУБД и ее ограничений.

Даталогическая модель описывает все таблицы, их поля и связи между ними, а также ограничения на эти таблицы и связи. Каждая таблица представляет собой отдельную сущность в системе и имеет свое имя и набор полей, которые описывают свойства этой сущности. Связи между таблицами определяют отношения между этими сущностями.

Даталогическая модель также учитывает ограничения на таблицы и связи, такие как уникальность значений в поле, ограничения на ввод значений, ограничения на удаление записей и т.д.

Кроме того, даталогическая модель включает определение типов данных для каждого поля и связи, а также определение индексов, которые используются для ускорения доступа к данным.

Для проекта «Анализ продаж», даталогическая модель будет содержать таблицы, такие как «товары», «поставки», «продажи», «клиенты». Каждая таблица будет содержать свои поля, ограничения и связи с другими таблицами.

Даталогическая модель представлена в соответствии с рисунком 2.12.

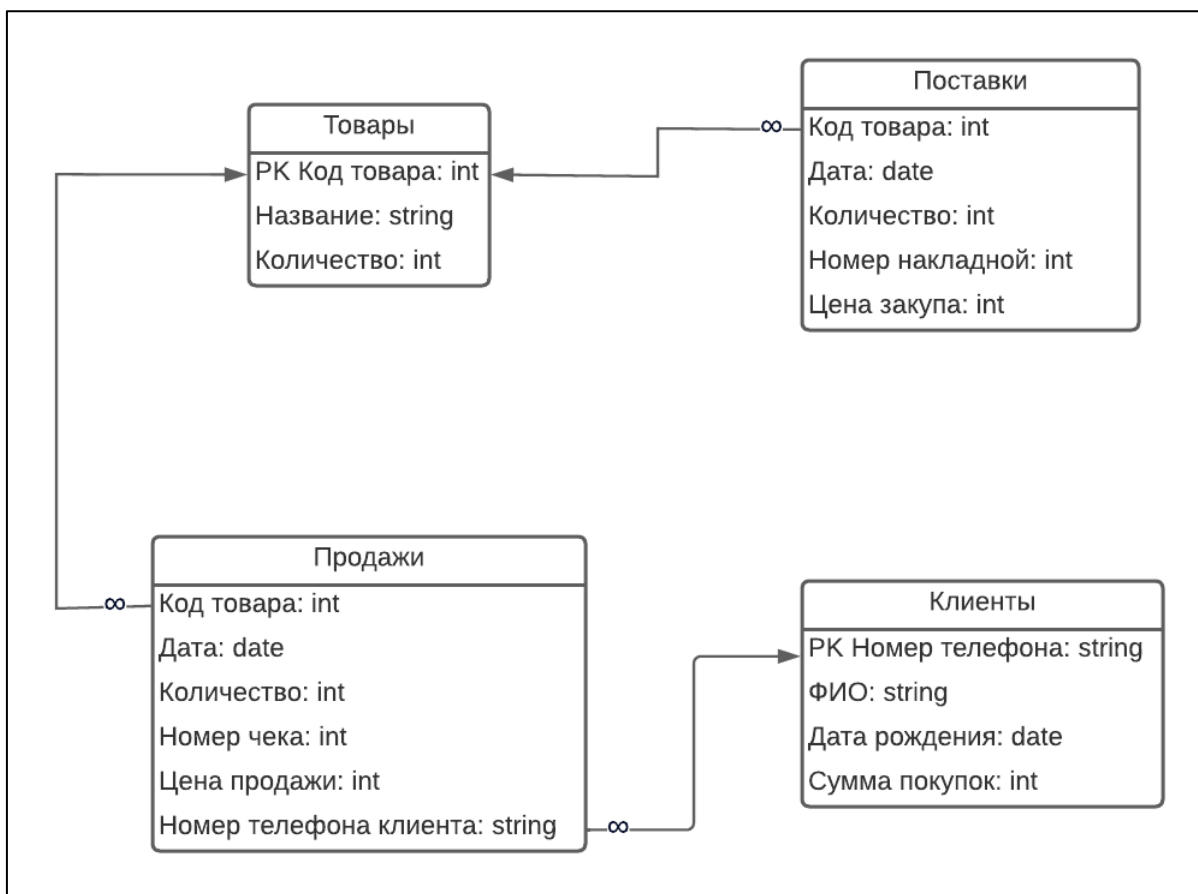


Рисунок 2.12 Даталогическая модель

Концептуальная схема базы данных, полученная в результате даталогического проектирования, является базовой структурой, на основе которой разрабатываются физические схемы базы данных, используемые конкретной СУБД. Даталогическая модель позволяет оптимизировать работу с данными и обеспечить целостность и надежность базы данных. Одним из главных преимуществ даталогической модели является то, что она позволяет более точно определить структуру базы данных и предотвратить возможные проблемы согласованности данных. Она также упрощает процесс создания физической модели базы данных и позволяет эффективнее работать с ней в дальнейшем.

2.4 Проектирование пользовательских интерфейсов

Окно «Главное меню» является основным интерфейсом пользовательского приложения, написанного на VBA. В этом окне имеются четыре кнопки:

«Добавление и просмотр товаров» - при нажатии на данную кнопку пользователь переходит на страницу редактирования записей товаров, там он может добавлять, редактировать и удалять записи.

«Поступление» - кнопка, предназначенная для перехода на страницу поставок товаров, где пользователь может добавить или удалить поставку товара.

«Продажа» - кнопка, которая открывает страницу редактирования базы данных продаж. Пользователь может добавить или удалить поставку.

«Аналитика по товарам» - кнопка, позволяющая пользователю посмотреть количество поступившего и проданного товара.

«Добавление и просмотр клиентов» - кнопка, которая открывает страницу редактирования базы данных клиентов. Здесь пользователь может добавлять, редактировать и удалять записи с данными о клиентах.

«Аналитика по клиентам» - кнопка, позволяющая посмотреть, какие пользователи имеют уровень скидки в 0%, 3%, 5% и 10%.

Окно «Главная» представлено в соответствии с рисунком 2.13.



Рисунок 2.13 Окно «Главное меню»

Окно «Добавление и просмотр товара» предназначено для редактирования записей в базе данных «товары». Оно содержит следующие элементы интерфейса:

- Текстовое поле для кода товара
- Текстовое поля для ввода названия товара
- Кнопку «Добавить товар» для добавления товара

- Кнопку «Редактировать» для редактирования уже существующей записи о товаре.
- Кнопку «Сохранить» для сохранения изменений в документе.
- Кнопку «Удалить» для удаления записи.

Окно «Добавление и просмотр товаров» представлено в соответствии с рисунком 2.14.

Добавление товара

Код товара

Название

Добавить товар Редактировать Сохранить Удалить

Код товара	Наименование	Ед. изм.	Количество
------------	--------------	----------	------------

Рисунок 2.14 Окно «Добавление и просмотр товаров»

Окно «Поступление» содержит следующие элементы:

- Текстовые поля для ввода кода товара, даты, количества, номера накладной и цены закупа
 - Кнопки для увеличения и уменьшения количества товара на единицу и десятков
 - Кнопки «Добавить» и «Удалить»
- Окно «Поступление» представлено в соответствии с рисунком 2.15.

Код	Название	Ед. изм.	№ накладной	Дата	Количеств	Цена	Сумма
101	Товар №2	шт.	1001	01.01.2020	40	500	20000
103	Товар №4	шт.	1001	01.01.2020	10	5500	55000
105	Товар №6	шт.	1001	01.01.2020	100	600	60000
104	Товар №5	шт.	1001	01.01.2020	31	4000	124000
103	Товар №4	шт.	1002	01.06.2020	40	4000	160000
105	Товар №6	шт.	1002	01.06.2020	10	6000	60000
102	Товар №2	шт.	1001	01.01.2020	31	5000	155000
152	Новый товар	шт.	5000	01.05.2023	20	2000	40000
102	Товар №2	шт.	1121	11.11.2023	31	2500	77500
153	Новый товар просто	шт.	12312	01.01.1999	20	123123	2462460
157	редактирование	шт.	12312	01.01.2000	10	12312	123120
157	редактирование	шт.	12312	01.01.2010	20	12312	246240
158	Название товара222	шт.	1231	01.01.2000	11	123123	1354353
158	Название товара222	шт.	1231	01.01.2010	11	123123	1354353

Рисунок 2.15 Окно «Поступления»

- Окно «Продажи» содержит следующие элементы:
- Текстовые поля для ввода кода товара, даты, количества, номера чека, номера клиента и цены закупа
 - Кнопки для увеличения и уменьшения количества товара на единицу и десятков
 - Кнопки «Добавить» и «Удалить»
- Окно «Профиль» представлено в соответствии с рисунком 2.16.

Продажа

Код товара Дата Номер чека

Номер телефона Количество 1 10 Цена

Код товара	Название	Ед. изм.	Чека	Дата	Количество	Цена	Сумма	Номер
101	Товар №1	шт.	88	02.01.2020	40	555	22200	87770848727
102	Товар №2	шт.	11	01.01.2020	1	1	1	87770848727
102	Товар №2	шт.	11	01.01.2020	1	1	1	87770848727
102	Товар №2	шт.	11	01.01.2020	1	1	1	87770848727
102	Товар №2	шт.	1	03.03.2020	1	1	1	87770848727
102	Товар №2	шт.	1	03.03.2020	1	1	1	87770848727
102	Товар №2	шт.	1	01.01.2030	36	1	36	87770848727
103	Товар №3	шт.	1	01.01.2030	1	1000000	1000000	87770222333
103	Товар №3	шт.	1	01.01.2030	1	60000	60000	87770999888
152	Новый товар	шт.	444	02.05.2023	11	444	4884	87000000000
153	Новый товар просто	шт.	123123	01.01.2011	29	12312312	357057048	87770111777
156	Крутое имя новое	шт.	123123	01.01.2000	1	1231231	1231231	87000555666
156	Крутое имя новое	шт.	123123	01.01.2011	21	1231231	25855851	87000555666
158	Название товара222	шт.	34232	01.01.2000	1	2342342	2342342	87770123456
158	Название товара222	шт.	34232	21.12.2012	20	2342342	46846840	87770123456

Рисунок 2.16 Окно «Продажи»

Окно «Аналитика по товарам» содержит следующие элементы:

- Текстовые поля для ввода кода товара, даты начала и конца периода
- Текстовые поля, в которых выводится результат запроса (количество поступивших единиц товара за период и количество проданных единиц товара)
- Кнопка «Расчёт», при нажатии на которую, если введенные значения прошли проверку на правильность ввода, будут отображены количества поступившего и проданного товара.
- Кнопка «Сгенерировать отчет» при нажатии на которую будет генерироваться документ Word, который будет заполняться полученными данными
- Кнопка «Диаграмма», которая отобразит количество полученных и проданных единиц товара в виде круговой диаграммы

Окно «Аналитика по товарам» выглядит в соответствии с рисунком 2.17.

The screenshot shows a window titled 'SalesAnalytics' with a blue border and a close button in the top right corner. The background is a light blue grid. The form contains the following elements:

- Код товара**: A text input field.
- Дата**: A date selection field with 'от' (from) and 'до' (to) labels.
- Расчет**: A button to calculate the data.
- Поступило за период**: A text input field for the period received.
- Продано за период**: A text input field for the period sold.
- Сгенерировать отчет**: A button to generate the report.
- Диаграмма**: A button to view the chart.

Рисунок 2.17 Окно «Статистика»

Окно «Добавление и просмотр клиентов» содержит следующие элементы:

- Текстовые поля для заполнения данных о клиенте (Номер телефона, ФИО, Дата рождения)
- Кнопка «Добавить» для добавления нового клиента
- Кнопка «Редактировать» для редактирования уже существующих записей.
- Кнопка «Сохранить», при нажатии на которую документ сохраняется
- Кнопку «Удалить», нажатие которой приводит к удалению существующей записи

- Таблица с клиентами, где отображается основная информация о них, а также сумма покупок и уровень скидки

Окно «Добавление и просмотр клиентов» представлено в соответствии с рисунком 2.18.

Добавление клиента

Номер телефона:

ФИО

Дата рождения (ДД.ММ.ГГГГ)

Добавить Редактировать Сохранить Удалить

Номер	ФИО	Дата	Сумма	Скидка
-------	-----	------	-------	--------

Рисунок 2.18. Окно «Избранное»

Окно «Аналитика по клиентам» содержит следующие элементы:

- Кнопки-опции «Не выбрано», «0%», «3%», «5%», «10%» для выбора уровня скидки
- Кнопку «Показать», при нажатии на которую отобразятся все клиенты с соответствующим уровнем скидки
- Кнопка «Диаграмма», при нажатии на которую появится отдельное окно с диаграммой, на которой будут показаны 5 клиентов с самой большой суммой покупок

Окно «Аналитика по клиентам» представлено в соответствии с рисунком 2.19.

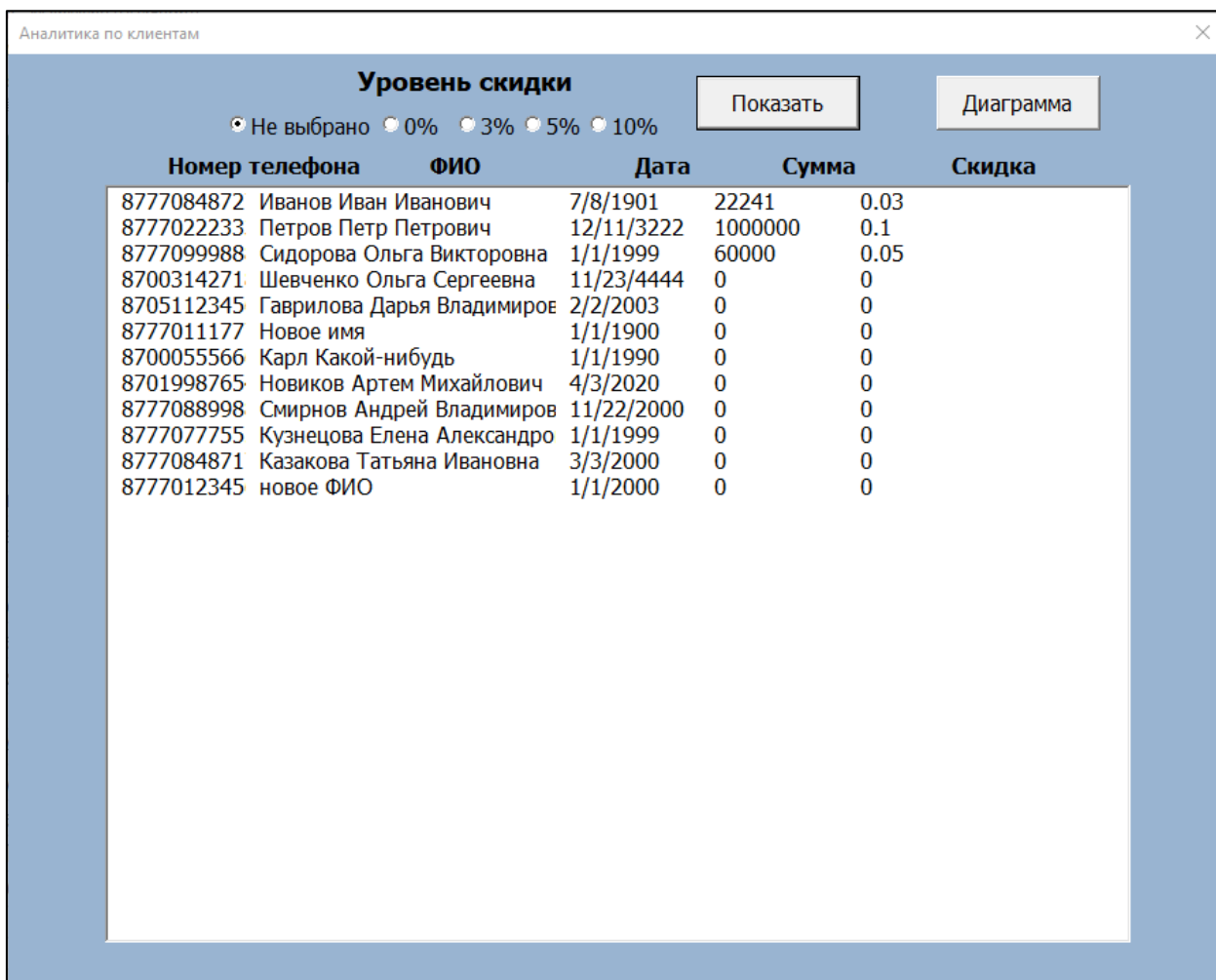


Рисунок 2.19 Окно «Аналитика по клиентам»

При нажатии на кнопку «Диаграмма», появится окно, которое выглядит в соответствии с рисунком 2.20

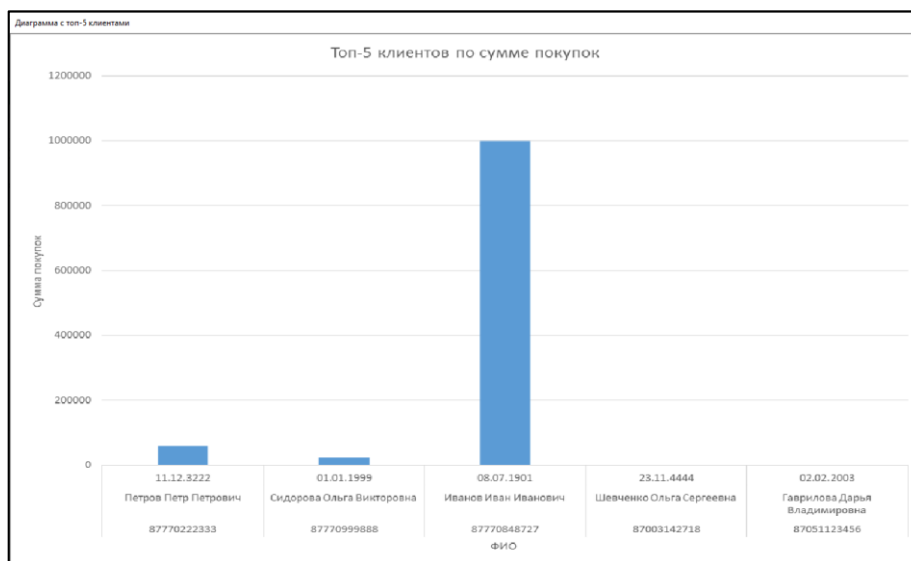


Рисунок 2.20 Окно «5 клиентов с самой большой суммой покупок»

Теперь рассмотрим диаграмму и отчет, который генерируется в окне «Аналитика по товару». Диаграмма. Получаемая в результате нажатия на кнопку «Диаграмма» выглядит в соответствии с рисунком 2.21.

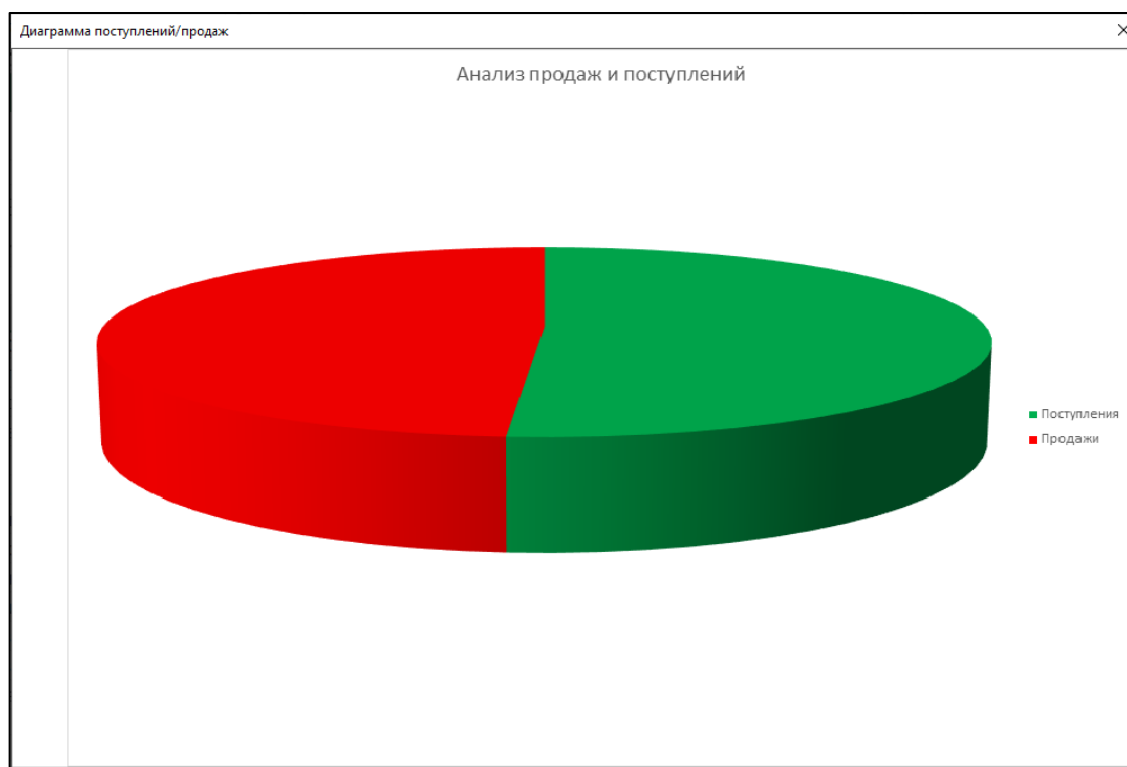


Рисунок 2.21 Окно «Диаграмма поставок/продаж»

Отчет, который генерируется на основе тех же данных выглядит в соответствии с рисунком 2.22.

Дата начала:	1.1.1999
Дата окончания периода:	1.1.2030
Код товара:	158
Поступившее за период количество товара:	22 шт.
Проданное за период количество товара:	21 шт.

Рисунок 2.22 Окно «Диаграмма поставок/продаж»

Проектирование пользовательских интерфейсов на VBA является важной задачей в разработке приложений на этой платформе. Цель создания удобного и понятного интерфейса заключается в том, чтобы обеспечить пользователям простой и интуитивно понятный доступ к функциям приложения.

Для достижения этой цели важно следовать некоторым общим принципам проектирования пользовательского интерфейса, таким как использование простых и понятных элементов управления, правильная организация размещения элементов на форме, создание логической структуры приложения и т.д.

При проектировании пользовательских интерфейсов на VBA также важно учитывать особенности платформы и целевой аудитории приложения.

В целом, проектирование пользовательских интерфейсов на VBA является важным аспектом разработки приложений на этой платформе, и его успешное выполнение может повысить удобство использования приложения для его пользователей и, следовательно, улучшить рейтинг приложения.

ЗАКЛЮЧЕНИЕ

В результате выполнения данного проекта было разработано приложение на VBA Excel под названием «Анализ продаж», которое позволяет пользователям быстро и легко анализировать продажи и поставки товаров, а также контролировать и управлять своей базой данных товаров.

Проект "Анализ продаж" представляет собой приложение на языке VBA Excel, которое предоставляет пользователям возможность эффективного анализа продаж и поставок товаров. Пользователь может вводить данные о продажах и поставках, а приложение автоматически анализирует их, предоставляя сводные данные о продажах и поставках за определенный период. Кроме того, в приложении предусмотрена возможность управления базой данных товаров, включая добавление, изменение и удаление информации о товарах.

В рамках проекта были выполнены все поставленные задачи, такие как разработка базы данных, создание интерфейса приложения, реализация функционала анализа продаж и поставок, разработка системы автоматического заполнения отчетов на основе введенных данных и управление базой данных товаров. Были использованы методы исследования, такие как анализ литературы по теме, изучение аналогов, разработка базы данных и программирование на языке VBA Excel.

Одной из ключевых особенностей приложения является возможность автоматического заполнения отчетов на основе анализа продаж и поставок. Это позволяет пользователям быстро и легко получать информацию о состоянии продаж и поставок в виде готовых отчетов, что упрощает процесс анализа и принятия решений.

Для реализации проекта были использованы различные методы исследования, включая анализ литературы и изучение аналогов на рынке. Также была проведена работа по программированию на языке VBA Excel, что позволило создать удобный и функциональный интерфейс приложения. В результате работы было получено приложение, которое помогает пользователям быстро и легко анализировать продажи и поставки товаров, а также управлять своей базой данных товаров и автоматически создавать отчеты для анализа результатов.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1 Ушакова, Е.В. Методические указания по выполнению дипломных проектов/работ для бакалавриата направления информационно-коммуникационные технологии [Текст] / Е.В. Ушакова, Л.В. Долматова. - Петропавловск: СКГУ им. М.Козыбаева, 2020. – 99 с.;
- 2 Хант Р., Пратт Д. VBA для чайников [Текст] / Хант Р., Пратт Д. – СПб.: Питер, 2017. – 416 с.;
- 3 Леонтьева Т.А. Разработка приложений в Excel на VBA [Текст] / Леонтьева Т.А. – М.: Издательский дом «Экзамен», 2014. – 272 с.;
- 4 Арсентьев А.В. Microsoft Excel VBA [Текст] / Арсентьев А.В. – М.: ДМК Пресс, 2017. – 256 с.;
- 5 Панасенко Н.В. Программирование в Excel на VBA для начинающих [Текст] / Панасенко Н.В. – М.: Логос, 2018. – 384 с.;
- 6 Журавлев А.В. Excel 2016. Программирование на VBA [Текст] / Журавлев А.В. – М.: Лань, 2017. – 320 с.;
- 7 Бондарев А.А. VBA в Excel 2013: Программирование для пользователей [Текст] / Бондарев А.А. – М.: Эксмо, 2013. – 304 с.;
- 8 Киселев С.А. Программирование в Microsoft Excel с помощью VBA [Текст] / Киселев С.А. – М.: Логос, 2013. – 256 с.;
- 9 Серебряков И.А. Разработка приложений в Excel на VBA [Текст] / Серебряков И.А. – М.: Издательство "СамМедиа", 2020. – 416 с.;
- 10 Панасенко Н.В. VBA для Excel 2016: учебник [Текст] / Панасенко Н.В. – М.: Диалектика, 2016. – 320 с.;
- 11 Покровский А.В., Шевелева Е.А. Программирование в VBA для Excel 2013 [Текст]: учебное пособие / А.В. Покровский, Е.А. Шевелева. – СПб.: БХВ-Петербург, 2013. – 336 с.;
- 12 Ракитов А.А. Программирование на VBA в приложениях Microsoft Office 2013 [Текст]: учебное пособие / А.А. Ракитов. – М.: Юрайт, 2013. – 256 с.;
- 13 Черезова Е.В., Шашков А.В. Программирование в Excel 2013. Самоучитель [Текст] / Е.В. Черезова, А.В. Шашков. – М.: Эком, 2014. – 320 с.;
- 14 Соловьев В.В. Информационные технологии и Excel [Текст]: учебное пособие / В.В. Соловьев. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2013. – 223 с.;
- 15 Рогозин Д.В. Программирование на языке VBA в Microsoft Office [Текст]: учебное пособие / Д.В. Рогозин. – СПб.: Питер, 2013. – 336 с.;
- 16 Козлов А.Г. Excel 2013. Практикум по программированию на VBA [Текст] / А.Г. Козлов. – М.: БХВ-Петербург, 2013. – 320 с.