|  |
| --- |
| ТЕХНИЧЕСКИЙ ПРОЕКТ  Информационной системы для фирмы по продаже комплектующих для ПК |
|  |

2024**Содержание**

[1. Уточнение структуры данных: 3](#_Toc181985069)

[2. Формы представления данных 6](#_Toc181985070)

[3. Разработка алгоритма решения задачи: 11](#_Toc181985071)

[4. Определение языка, структуры программы и требований к техническим средствам: 23](#_Toc181985072)

[5. Требования к техническим средствам: 24](#_Toc181985073)

# **Уточнение структуры данных:**

База данных информационной системы для фирмы по продаже комплектующих для ПК изображена в виде логической и физической ER-диаграммы (см. рис. 1 и 2).

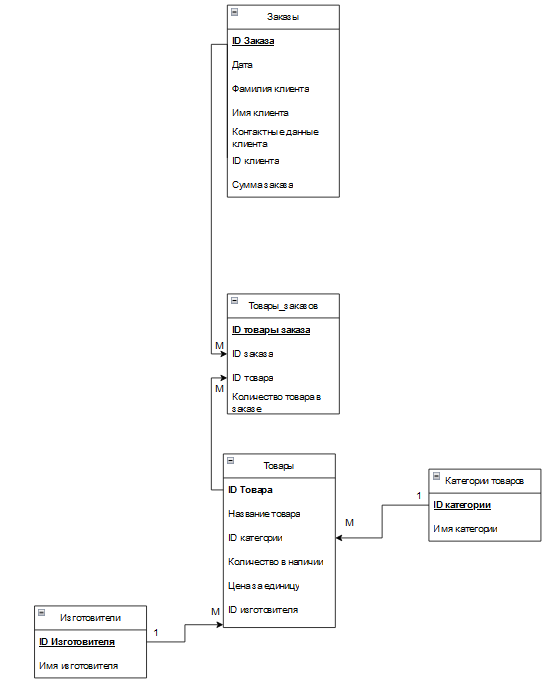


Рисунок 1 – Логическая ER-диаграмма

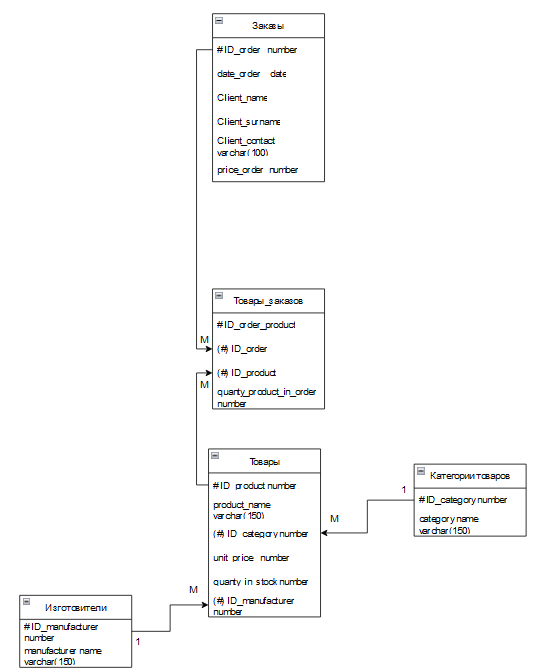


Рисунок 2 – Физическая ER-диаграмма

В базе данных представлено 5 таблиц: «Заказы», «Товары заказов», «Товары», «Изготовители» и «Категории».

Таблица «Заказы» содержит следующую информацию:

* Id заказа: уникальный идентификатор заказа (первичный ключ).
* Дата заказа: дата, когда был оформлен заказ.
* Фамилия клиента: фамилия клиента, оформившего заказ (внешний ключ к таблице клиентов, если есть).
* Имя клиента: имя клиента.
* Контактные данные клиента: информация для связи с клиентом.
* Сумма заказа: общая стоимость заказа.

Таблица «Товары заказов» хранит данные о каждом товаре, входящем в заказ:

* Id товара заказов: уникальный идентификатор записи (первичный ключ).
* Id заказа: ссылка на таблицу «Заказы», идентифицирующая заказ (внешний ключ).
* Id товара: ссылка на таблицу «Товары», идентифицирующая конкретный товар (внешний ключ).
* Количество товара в заказе: количество единиц товара, заказанных в данном заказе.

Таблица «Товары» описывает каждый товар, доступный для заказа:

* Id товара: уникальный идентификатор товара (первичный ключ).
* Название товара: наименование товара.
* Id категории: ссылка на таблицу «Категории», указывающая на категорию товара (внешний ключ).
* Цена за единицу: стоимость одной единицы товара.
* Количество в наличии: количество доступных единиц на складе.
* Id изготовителя: ссылка на таблицу «Изготовители», указывающая на производителя товара (внешний ключ).

Таблица «Изготовители» содержит информацию о поставщиках:

* Id изготовителя: уникальный идентификатор изготовителя (первичный ключ).
* Название изготовителя: название компании-производителя.

Таблица «Категории» хранит данные о категориях товаров:

* Id категории: уникальный идентификатор категории (первичный ключ).
* Название категории: наименование категории товаров.

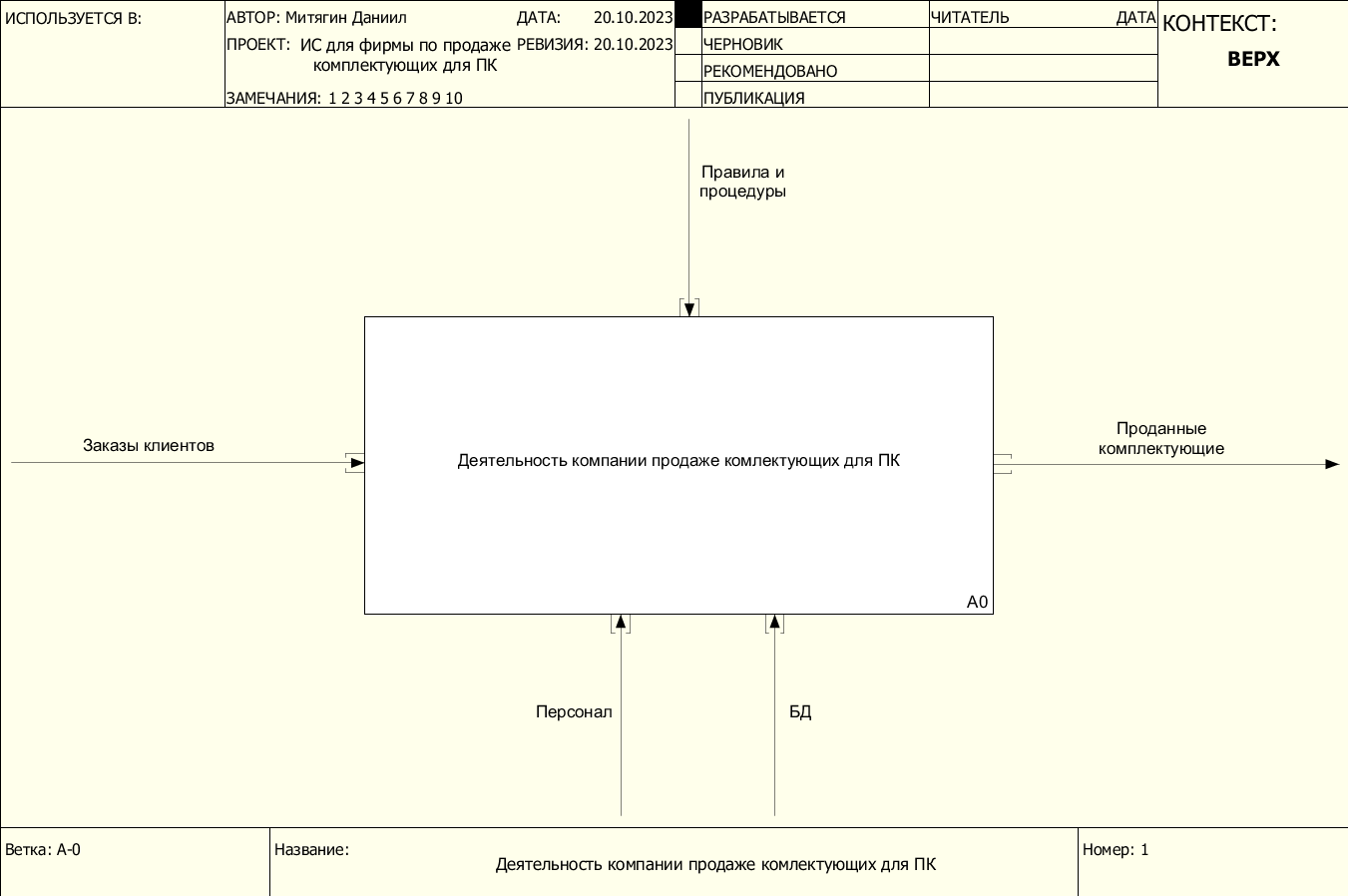


Рисунок 3 – Контекстная диаграмма

На рисунке 3 изображена общая диаграмма IDEF0 деятельности компании по продаже комплектующих для ПК.

Внешней входящей информацией являются:

* Заказы клиентов

Управляющим воздействием будут являться:

* Правила и процедуры

Механизмом управления будут являться:

* Персонал
* БД

Внешней исходящей информацией будут являться:

* Проданные комплектующие

На рисунке 4 показана декомпозиция общей диаграммы. На этом рисунке подробно описана работа ИС.

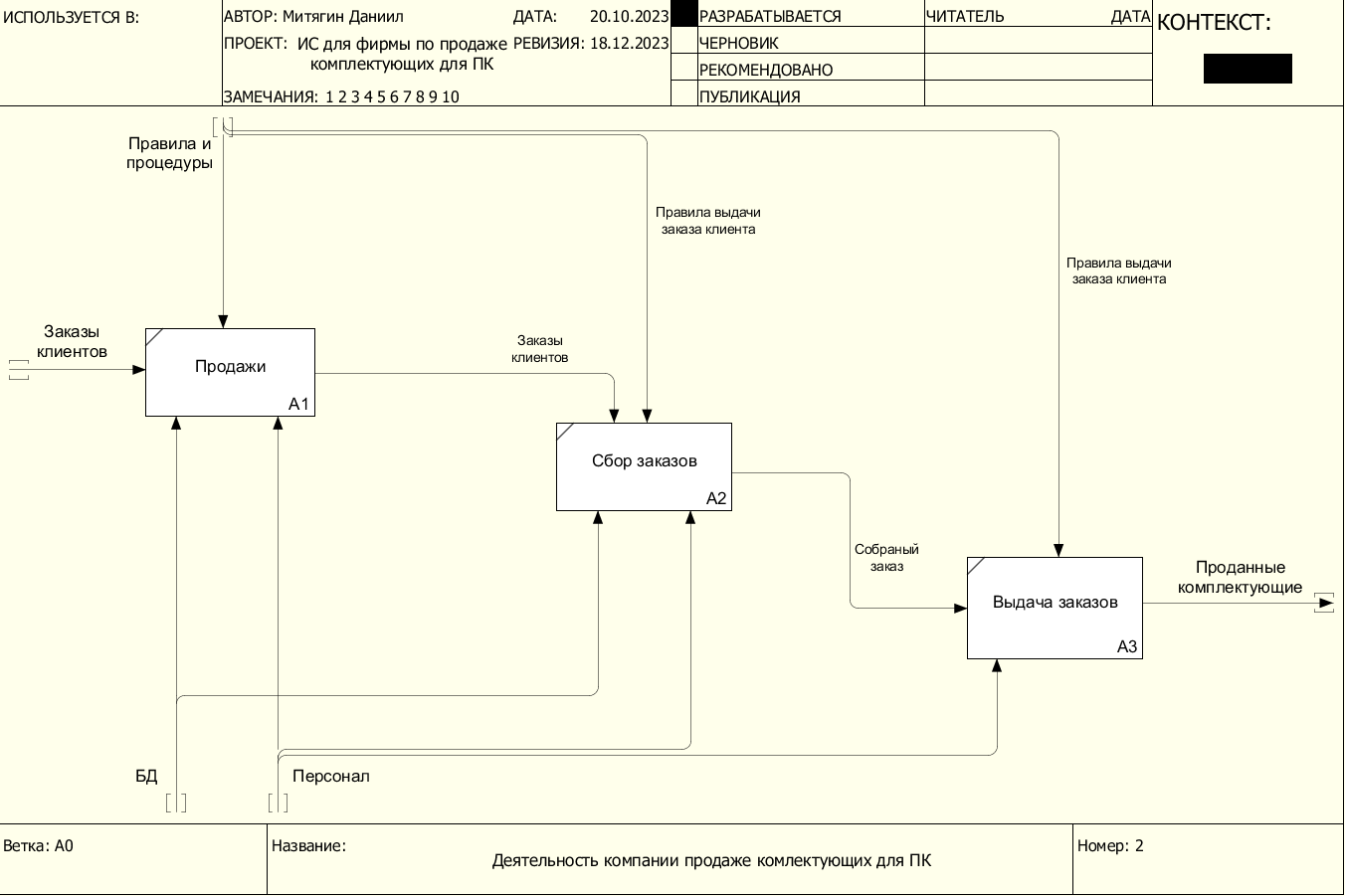


Рисунок 4 – Декомпозиция диаграммы

На этой диаграмме изображены три блока: «Продажи», «Сбор заказов» и «Выдача заказов».

1. На вход блока **«Продажи»** поступает **«Заказ клиента»**, который обрабатывается, после чего формируется **«Заявка клиента»**, направляемая в следующий блок.
2. В блоке **«Сбор заказов»** на вход поступает **«Заявка клиента»**, на основе которой осуществляется комплектование заказа. На выходе блока получается **«Собранный заказ»**, который направляется в блок выдачи.
3. В блоке **«Выдача заказов»** на вход поступает **«Собранный заказ»**, который передается клиенту. На выходе блока формируется **«Проданные комплектующие»**.

В процессе работы используются правила для приема, сбора и выдачи заказов, а также персонал и база данных, которые обеспечивают корректность выполнения операций.

На рисунке 5 представлена диаграмма в нотации DFD.

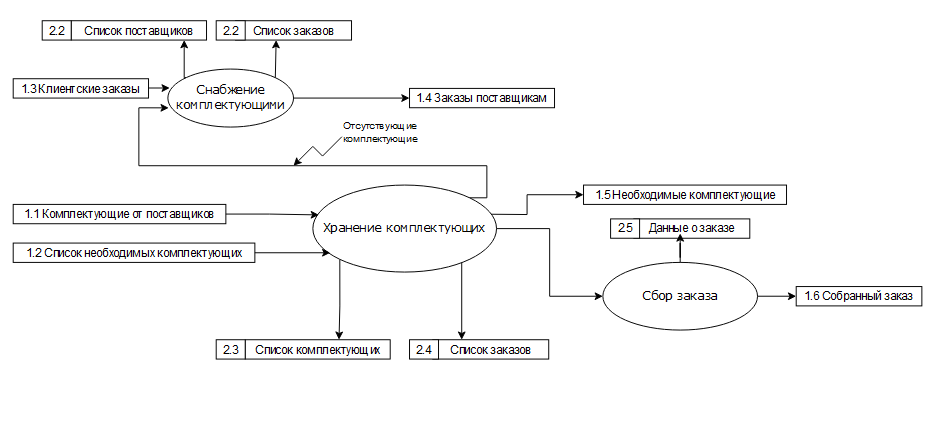


Рисунок 5 – Диаграмма в нотации DFD

В данной диаграмме есть 3 процесса: «Снабжение комплектующими», «Хранение комплектующих» и «Сбор заказа». На вход процессу «Хранение комплектующих» поступают внешние сущности «Комплектующие от поставщиков» и «Список необходимых комплектующих», а на выход внешние данные «Необходимые комплектующие». У процесса «Хранение комплектующих» есть два хранилища данных «Список комплектующих» и «Список заказов». Из процесса «Хранение комплектующих» идет поток данных «Отсутствующие комплектующие» на вход в процесс «Снабжение комплектующими». Также на вход поступают входные данные «Клиентские заказы», и на выходе получаются «Заказы поставщикам». У процесса «Снабжение комплектующими» также есть два хранилища данных: «Список поставщиков» и «Список заказов». Из процесса «Хранение комплектующих» идет еще один поток данных и входит в процесс «Сбор заказа». У процесса «Сбор заказа» есть хранилище данных «Данные о заказе», а также внешняя сущность «Собранный заказ».

# **Формы представления данных**

На прототипе оконной формы с информацией о товарах (см. рис. 3) расположены 3 кнопки “Добавить товар”, “Удалить товар”, “Изменить товар” и “Оформить” с помощью которых можно выйти в меню, добавлять, удалять и изменять данные в базе данных. С помощью добавления товаров в список заказа и кнопки “Оформить” можно оформить заказ. Также в верхней части располагаются 6 кнопки “Товары”, “Категории”, “Заказы” и “Отчеты” с помощью которых можно перемещаться между вкладками.

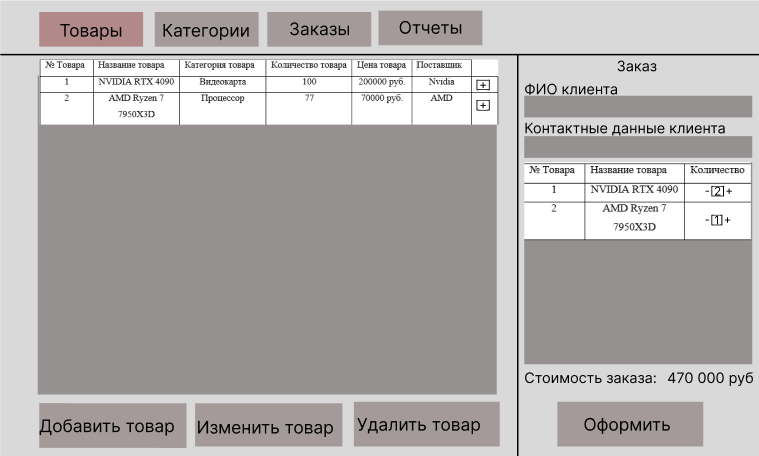
****

Рисунок 6 – Прототип оконной формы с информацией о товарах

На рисунке 7 изображено окно для добавления товара.

****

Рисунок 7 – Прототип оконной формы добавления товара

На рисунке 8 изображено окно для изменения информации о товаре.

****

Рисунок 8 – Прототип оконной формы изменения информации о товаре

На рисунке 9 изображен прототип оконной формы с информацией о категориях на которой расположены кнопки “Добавить категорию”, “Удалить категорию” и “Изменить категорию” с помощью которых можно добавлять, удалять и изменять данные в базе данных.

****

Рисунок 9 – Прототип оконной формы с информацией о категориях

На рисунке 10 изображено окно для добавления категории.

****

Рисунок 10 – Прототип оконной формы добавления категории

На рисунке 11 изображено окно для изменения информации о категории.

****

Рисунок 11 – Прототип оконной формы изменения информации о категории

На рисунке 12 изображен прототип оконной формы с информацией о заказах на которой расположены кнопки “О заказе” и “Удалить запись” с помощью которых можно узнать информацию о заказе и удалять данные в базе данных.

****

Рисунок 12 – Прототип оконной формы с информацией о категориях

На рисунке 13 изображено окно с информацией о заказе.

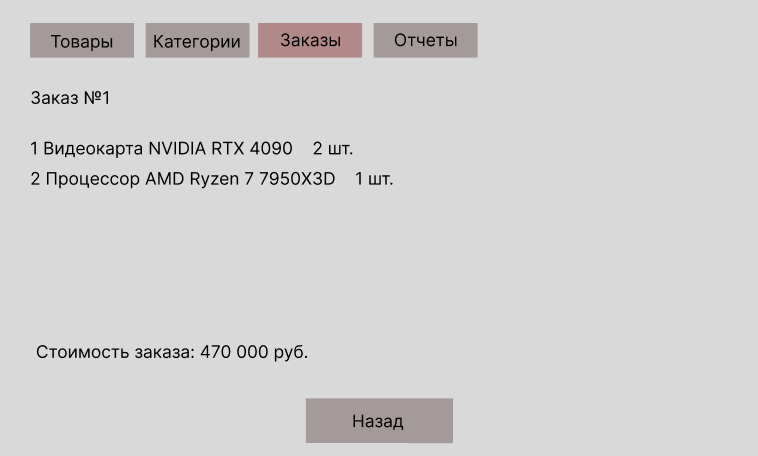
****

Рисунок 13 – Прототип оконной формы с информацией о заказе

На рисунке 14 изображено окно с отчетами. На данной вкладке можно выбрать тип отчета и создать его кнопкой “Создать отчет”.

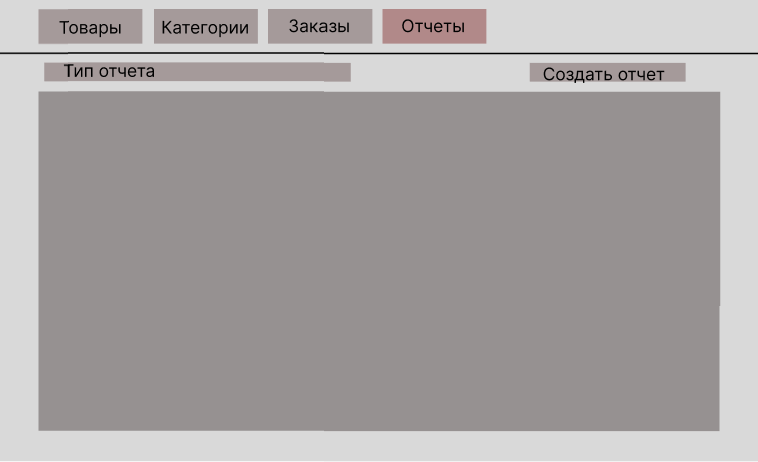
****

Рисунок 14 - Прототип оконной формы с отчетами

# **Разработка алгоритма решения задачи:**

Таблица 1 – Пользовательские сценарии

|  |  |  |
| --- | --- | --- |
| № | Пользователь | Система |
| 1 | Нажимает кнопку "Добавить товар". | Система открывает окно для ввода данных нового товара. |
| 2 | Заполняет поля и нажимает "Добавить". | Система проверяет данные и добавляет новый товар в базу данных. |
| 3 | Нажимает кнопку "Изменить товар". | Система открывает окно для изменения данных о товаре. |
| 4 | Вносит изменения и нажимает "Сохранить". | Система обновляет данные товара в базе. |
| 5 | Нажимает кнопку "Удалить товар". | Система удаляет выбранный товар из базы данных. |
| 6 | Нажимает кнопку "Оформить" после добавления товаров в список. | Система оформляет заказ и обновляет данные о наличии товаров. |
| 7 | Нажимает кнопку "Добавить категорию". | Система открывает окно для добавления новой категории. |
| 8 | Заполняет поля и нажимает "Добавить". | Система сохраняет новую категорию в базе данных. |
| 9 | Нажимает кнопку "Изменить категорию". | Система открывает окно для изменения информации о категории. |
| 10 | Вносит изменения и сохраняет их. | Система обновляет данные категории в базе. |
| 11 | Нажимает кнопку "Удалить категорию". | Система удаляет выбранную категорию из базы данных. |
| 12 | Нажимает кнопку "О заказе" в списке заказов. | Система открывает окно с подробной информацией о заказе. |
| 13 | Нажимает кнопку "Удалить запись" в окне заказов. | Система удаляет выбранный заказ из базы данных. |
| 14 | Нажимает кнопку "Создать отчет" на вкладке отчетов. | Система генерирует и выводит отчет на основе выбранных критериев. |

Для лучшего понимания алгоритма «Система открывает окно для ввода данных нового товара» была создана схема алгоритма, которая изображена ниже.

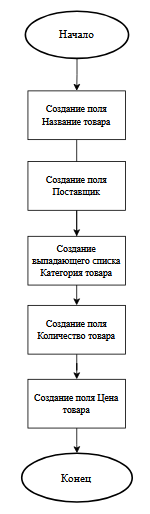


Рисунок 15 - Алгоритм «Система открывает окно для ввода данных нового товара»

Для лучшего понимания алгоритма «Система проверяет данные и добавляет новый товар в базу данных» была создана схема алгоритма, которая изображена ниже.

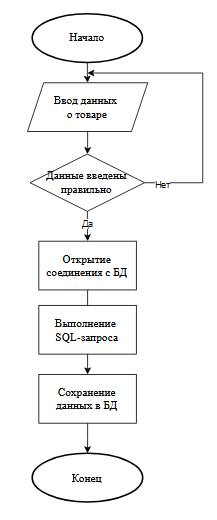


Рисунок 16 - Алгоритм «Система проверяет данные и добавляет новый товар в базу данных»

Для лучшего понимания алгоритма «Система открывает окно для изменения данных о товаре» была создана схема алгоритма, которая изображена ниже.



Рисунок 17 - Алгоритм «Система открывает окно для изменения данных о товаре»

Для лучшего понимания алгоритма «Система обновляет данные товара в базе» была создана схема алгоритма, которая изображена ниже.

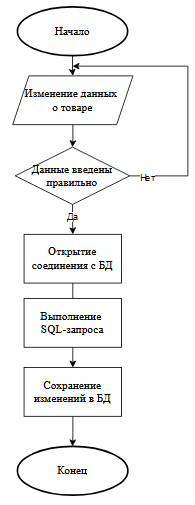


Рисунок 18 - Алгоритм «Система обновляет данные товара в базе»

Для лучшего понимания алгоритма «Система удаляет выбранный товар из базы данных» была создана схема алгоритма, которая изображена ниже.

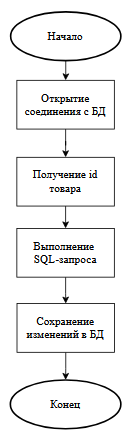


Рисунок 19 - Алгоритм «Система удаляет выбранный товар из базы данных»

Для лучшего понимания алгоритма «Система оформляет заказ и обновляет данные о наличии товаров» была создана схема алгоритма, которая изображена ниже.

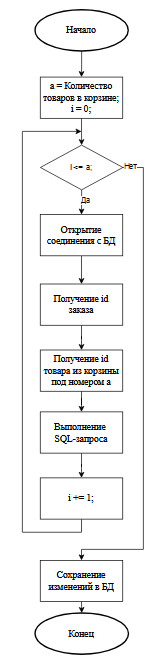


Рисунок 20 - Алгоритм «Система оформляет заказ и обновляет данные о наличии товаров»

Для лучшего понимания алгоритма «Система открывает окно для добавления новой категории» была создана схема алгоритма, которая изображена ниже.



Рисунок 21 - Алгоритм «Система открывает окно для добавления новой категории»

Для лучшего понимания алгоритма «Система сохраняет новую категорию в базе данных» была создана схема алгоритма, которая изображена ниже.

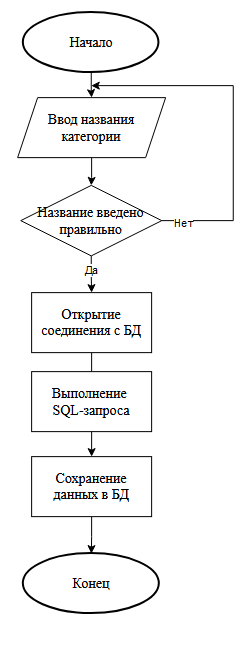


Рисунок 22 - Алгоритм «Система сохраняет новую категорию в базе данных»

Для лучшего понимания алгоритма «Система открывает окно для изменения информации о категории» была создана схема алгоритма, которая изображена ниже.

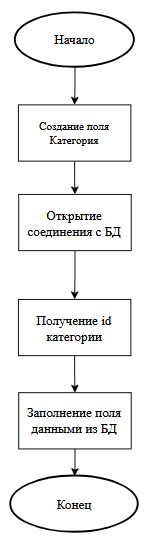


Рисунок 23 - Алгоритм «Система открывает окно для изменения информации о категории»

Для лучшего понимания алгоритма «Система обновляет данные категории в базе» была создана схема алгоритма, которая изображена ниже.

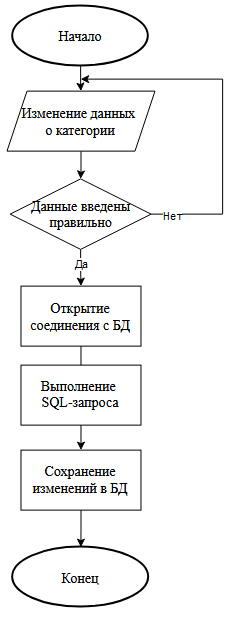


Рисунок 24 - Алгоритм «Система обновляет данные категории в базе»

Для лучшего понимания алгоритма «Система удаляет выбранную категорию из базы данных» была создана схема алгоритма, которая изображена ниже.

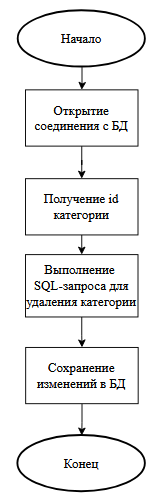


Рисунок 25 - Алгоритм «Система удаляет выбранную категорию из базы данных»

Для лучшего понимания алгоритма «Система открывает окно с подробной информацией о заказе» была создана схема алгоритма, которая изображена ниже.

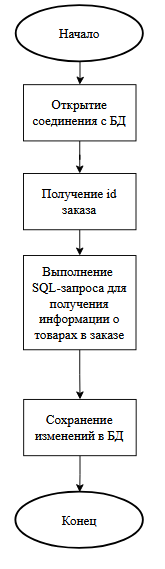


Рисунок 26 - Алгоритм «Система открывает окно с подробной информацией о заказе»

Для лучшего понимания алгоритма «Система удаляет выбранный заказ из базы данных» была создана схема алгоритма, которая изображена ниже.

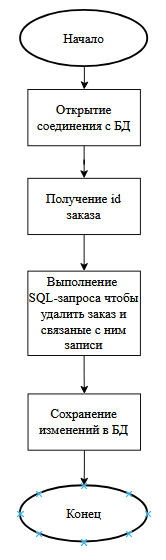


Рисунок 27 - Алгоритм «Система удаляет выбранный заказ из базы данных»

Для лучшего понимания алгоритма «Система генерирует и выводит отчет на основе выбранных критериев» была создана схема алгоритма, которая изображена ниже.

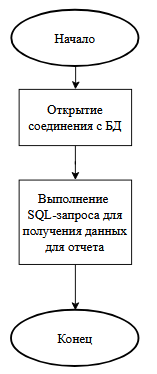


Рисунок 28 - Алгоритм «Система генерирует и выводит отчет на основе выбранных критериев»

# **Определение языка, структуры программы и требований к техническим средствам:**

Программирование: в основе проекта лежит язык Python, который выбран за его универсальность и обширную поддержку библиотек для различных задач разработки, включая работу с базами данных и разработку интерфейсов.

Графический интерфейс: в качестве фреймворка для создания интерфейса используется PyQt5. Этот инструмент позволяет создавать удобные и функциональные графические приложения с широким выбором элементов управления.

Хранение данных: для локального хранения данных используется SQLite, обеспечивающий компактность, простоту настройки и достаточную производительность для приложений с умеренными объемами данных.

# **Требования к техническим средствам:**

Минимальные технические требования для запуска информационной системы:

* Процессор с частотой 2 ГГц;
* Оперативная память: 4 ГБ;
* Монитор;
* Компьютерная мышь или тачпад;
* Клавиатура.