



Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

ИУ «ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ»

КАФЕДРА

ИУ7 «ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»

КУРСОВАЯ РАБОТА

НА ТЕМУ:

*Разработка базы данных для хранения и
обработки данных цветочного магазина*

Студент

ИУ7-65Б

(группа)

(подпись, дата)

(И.О. Фамилия)

Руководитель курсового
проекта

(подпись, дата)

Никольшина Т.А.

(И.О. Фамилия)

Консультант

(подпись, дата)

(И.О. Фамилия)

2025 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Аналитический раздел	4
1.1 Анализ предметной области. Сравнительный анализ существующих решений	4
1.2 Формулировка требований к разрабатываемой базе данных и приложению	6
1.3 Формализация и описание информации, подлежащей хранению в проектируемой базе данных	6
1.4 Проведенный анализ существующих баз данных на основе формализации данных	6
1.5 ER-диаграмма сущностей проектируемой базы данных в нотации Чена	7
1.6 Формализация и описание пользователей проектируемого приложения к базе данных	9
1.7 Диаграмма вариантов использования	9
2 Конструкторский раздел	11
2.1 Диаграмма проектируемой базы данных	11
2.2 Описание сущностей проектируемой базы данных	11
2.3 Описание проектируемых ограничений целостности базы данных	14
2.4 Описание всех проектируемых процедур/функций/триггеров в формате схемы	15
2.5 Описание проектируемой ролевой модели на уровне базы данных	15
3 Технологический раздел	18

3.1	Обоснование выбора средств реализации базы данных и приложения	18
3.2	Описание сущностей реализованной базы данных	18
3.3	Описание реализованных ограничений целостности базы данных	18
3.4	Описание всех реализованных процедур/функций/триггеров в формате схемы	18
3.5	Описание ролевой модели на уровне базы данных	18
3.6	Описание методов тестирования и тестовых кейсов для всех разработанных на стороне базы данных функций	19
3.7	Описание интерфейса доступа к базе данных	19
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ		20

ВВЕДЕНИЕ

Данная работа посвящена проектированию и реализации базы данных и приложения для автоматизации работы цветочного магазина. Основное внимание уделяется разработке структуры базы данных, обеспечению целостности хранимых данных, а также созданию функционального приложения, позволяющего эффективно управлять товарными запасами, учетными записями пользователей и бизнес-процессами магазина.

Цель: создание надежной и масштабируемой базы данных и приложения для цветочного магазина, обеспечивающих автоматизацию ключевых операций.

Задачи:

- 1) сформулировать требования и ограничения к разрабатываемой базе данных и приложению;
- 2) определить роли и описание пользователей системы;
- 3) спроектировать сущности базы данных и ограничения целостности;
- 4) реализовать триггер для автоматического обновления товарных остатков при поступлении новых партий;
- 5) выбрать средства реализации базы данных и приложения;
- 6) разработать и реализовать сущности базы данных с учетом ограничений целостности;
- 7) описать методы тестирования функционала и разработать тестовые сценарии;
- 8) провести исследование производительности запросов в зависимости от объема данных с использованием индексов и без них.

1 Аналитический раздел

1.1 Анализ предметной области. Сравнительный анализ существующих решений

Общая характеристика предметной области

Цветочный бизнес — это динамично развивающаяся отрасль, включающая продажу свежих цветов, комнатных растений, сопутствующих товаров (упаковка, открытки, удобрения) и дополнительные услуги (доставка). Современные цветочные магазины стремятся автоматизировать процессы учета товаров, управления заказами и взаимодействия с клиентами, что требует эффективных IT-решений.

Ключевые процессы, подлежащие автоматизации:

- управление ассортиментом (учет цветов, растений, упаковки);
- контроль остатков на складе;
- оформление и отслеживание заказов (включая доставку);
- управление клиентской базой и маркетинговыми акциями;
- аналитика продаж и формирование отчетности;

Сравнительный анализ существующих решений

На рынке представлено несколько типовых решений для автоматизации работы цветочных магазинов:

1) Ручные системы учета

Наиболее простой вариант, включающий использование Excel и бумажных журналов. Основные недостатки: высокая вероятность ошибок при вводе данных и сложности при расширении бизнеса.

2) Универсальные CRM-платформы

Такие системы как 1С [1] и Битрикс24 предлагают широкий функционал, но их внедрение требует существенных финансовых вложений и часто включает невостребованные возможности.

3) Отраслевые программные продукты

Специализированные решения типа FloristWare и FlowerShop Pro созданы специально для флористики, но имеют ограниченную гибкость настройки

и высокую стоимость лицензий.

Как показано в таблице 1.1, сравнительный анализ существующих решений демонстрирует, что ручные методы учета неэффективны при росте бизнеса из-за высокого риска ошибок и сложности масштабирования. Универсальные CRM-системы (1С, Битрикс24), хотя и предлагают широкий функционал, оказываются избыточными для малого цветочного магазина, требуя значительных затрат на адаптацию. Специализированные решения (FloristWare, FlowerShop Pro), разработанные для флористического бизнеса, часто не покрывают все специфические потребности магазинов и отличаются высокой стоимостью лицензирования.

Таблица 1.1 — Сравнительный анализ функциональных возможностей систем управления для цветочного бизнеса

Критерий	Microsoft Excel	1С:Предприятие	FloristWare Pro
Автоматизация учета	Ручной ввод данных, высокая вероятность ошибок	Полная автоматизация всех учетных операций	Специализированная автоматизация для флористики
Управление складом	Базовый учет остатков без аналитики	Полноценный складской учет с аналитикой сроков годности	Контроль остатков с напоминаниями о пополнении
Обработка заказов	Текстовые записи в ячейках таблицы	Интегрированная CRM-система с историей заказов	Оптимизированный интерфейс для быстрого оформления
Отчетность	Ручное составление отчетов по шаблонам	Встроенные отчетные формы с аналитикой	Стандартные отчеты по специфике цветочного бизнеса
Интеграции	Ограниченные возможности	Широкие возможности интеграции	Специализированные API для флористики
Стоимость	Включена в пакет Office	Лицензия + внедрение)	Подписка

1.2 Формулировка требований к разрабатываемой базе данных и приложению

Функциональные требования:

- управление товарами (хранение информации о цветах и сопутствующих товарах, учет остатков на складе, обновление количества при поступлении);
- управление заказами (регистрация заказов, учет клиентов и истории заказов);
- управление поставщиками (ведение списка поставщиков, учет поступлений товаров).

1.3 Формализация и описание информации, подлежащей хранению в проектируемой базе данных

Требования к БД:

- Сущности:
 - Товары (id, название, цена, количество)
 - Поставщики (id, название, контакты)
 - Пользователи (id, имя, телефон)
 - Поставки товаров (id, товары, дата, ответственный)
 - Заказы (id, клиент_id, статус, ответственный)
 - Склады (id, название, адрес)
- Триггеры:
 - автообновление остатков при поставках.

1.4 Проведенный анализ существующих баз данных на основе формализации данных

В таблице 1.2 представлено сравнение трёх моделей хранения данных: **SQL** (реляционная), **NoSQL** (документная) [2] и **Ключ-Значение** [3], проведенное на основе анализа структуры проектируемой базы данных.

Таблица 1.2 — Сравнение моделей хранения данных

Критерий	SQL (PostgreSQL, MySQL)	NoSQL (MongoDB)	Ключ- Значение (Redis)
Структура	Жёсткая схема (таблицы)	Гибкая (JSON- документы)	Просто ключ + значение
Связи	JOIN, внешние ключи	Вложенные до- кументы	Нет связей
Пример для Зака- за	Отдельные таблицы (Заказ, Контрагент)	Единый доку- мент с вложен- ными данными	order:123 → "JSON-данные"
Транзакции	Полная под- держка (ACID)	Ограниченная	Нет (или про- стые операции)
Масштабируемость	Вертикальное	Горизонтальное	Горизонтальное
Скорость	Оптимизирована для сложных запросов	Быстрые встав- ки/чтения	Максимальная скорость для простых опера- ций
Проблемы	Сложность мас- штабирования	Дублирование данных	Нет сложных за- просов

Для данной базы данных SQL является оптимальным выбором, так как:

- Данные имеют чёткую структуру и связи (например, «Партия → Номенклатура»)
- Требуются транзакции (например, при списании товаров)
- Необходимы сложные запросы с JOIN (анализ продаж, остатков)

NoSQL и Ключ-Значение не подходят для основной системы, но могут использоваться как дополнение.

Вывод: реляционная СУБД лучше всего соответствует требованиям данной системы.

1.5 ER-диаграмма сущностей проектируемой базы данных в нотации Чена

На рисунке 1.1 представлена ER-диаграмма сущностей проектируемой базы данных в нотации Чена для разрабатываемого приложения.

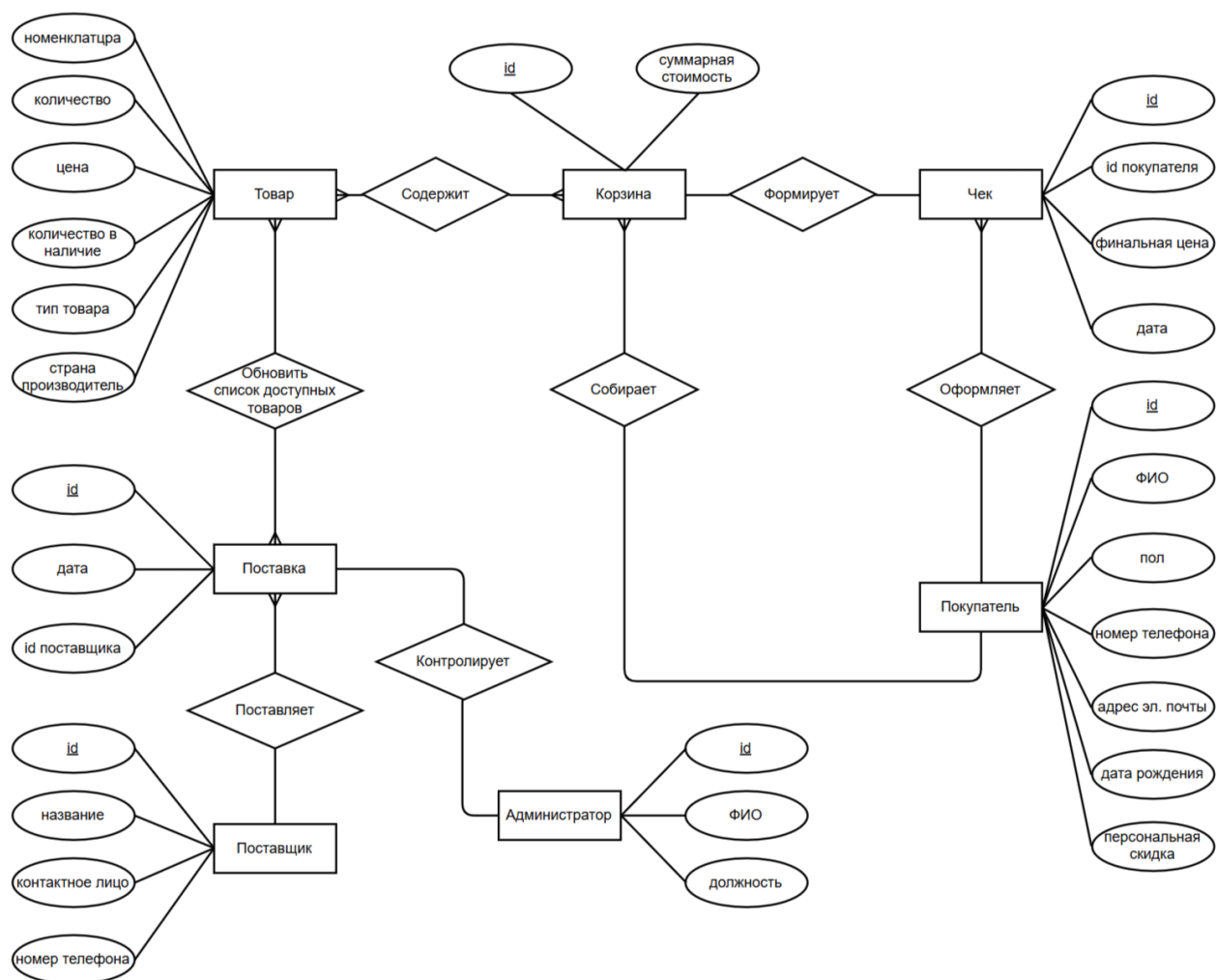


Рисунок 1.1 — ER-диаграмма сущностей проектируемой базы данных в нотации Чена

1.6 Формализация и описание пользователей проектируемого приложения к базе данных

Пользователи приложения:

- администратор (полный доступ);
- продавец (просмотр каталога товаров, работа с заказами);
- кладовщик (загрузка информации о новой партии товара, просмотр каталога товаров)

Роли БД:

- администратор;
- продавец;
- кладовщик;

1.7 Диаграмма вариантов использования

На рисунке 1.2 представлена диаграмма вариантов использования для разрабатываемого приложения.

Вывод

Проведённый анализ предметной области и сравнение существующих решений показали необходимость разработки специализированной системы автоматизации для цветочного бизнеса. На основе формализованных требований была спроектирована структура базы данных с чёткими связями между сущностями. Сравнительный анализ моделей хранения данных подтвердил преимущества реляционного подхода, обеспечивающего целостность данных и поддержку транзакций. В качестве СУБД выбрана PostgreSQL, как наиболее подходящая система благодаря её надежности, производительности и богатому функционалу для работы со сложными запросами. Полученные результаты стали основой для дальнейшего проектирования базы данных и разработки приложения.

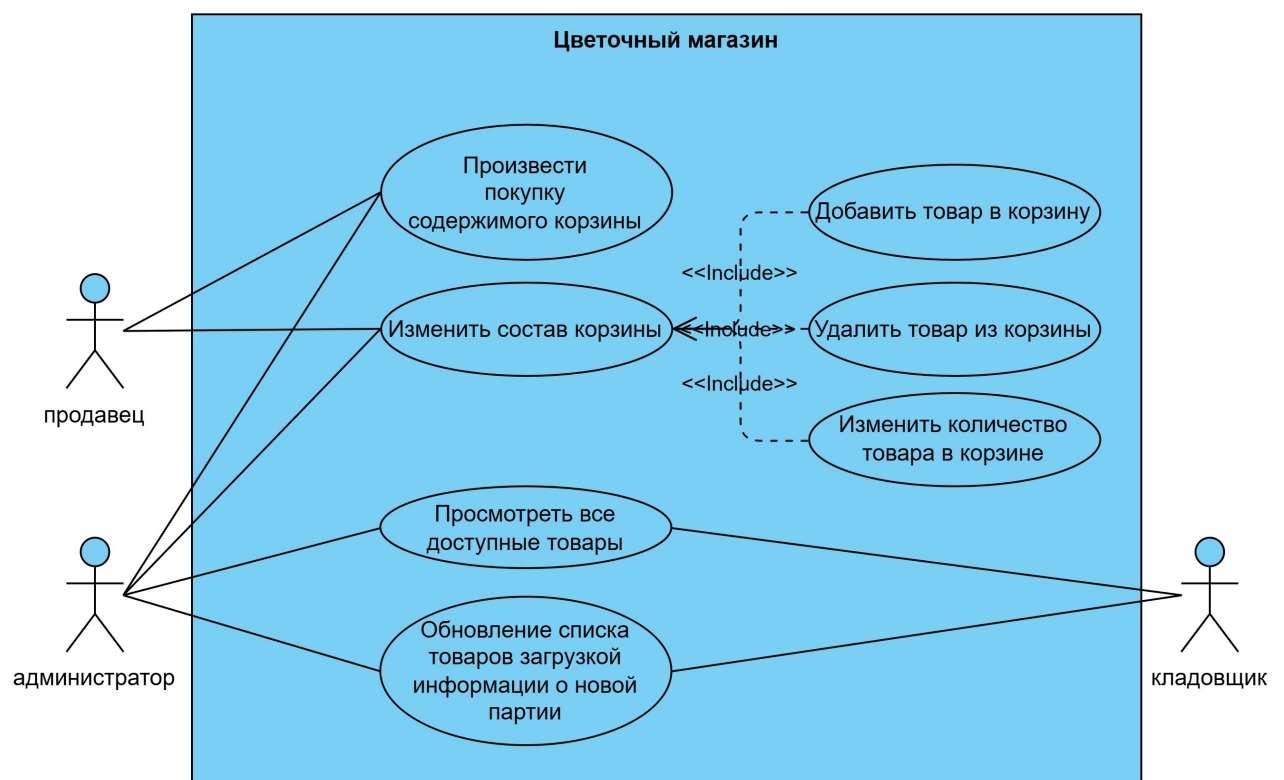


Рисунок 1.2 — Диаграмма вариантов использования

2 Конструкторский раздел

2.1 Диаграмма проектируемой базы данных

На рисунке 2.1 представлена диаграмма проектируемой базы данных.

2.2 Описание сущностей проектируемой базы данных

— Списание

- Основная сущность для учёта списания товаров
- Атрибуты:
 - * id - уникальный идентификатор
 - * номер партии
 - * номенклатура
 - * количество списываемого товара
 - * ответственный за списание (пользователь с ролью кладовщик)

— Пользователи

- Сущность для хранения данных пользователей системы
- Атрибуты:
 - * id - уникальный идентификатор
 - * название (имя пользователя)
 - * роль в системе

— Страна

- Справочник стран
- Атрибуты:
 - * id - уникальный идентификатор
 - * название страны

— Товар на складе

- Сущность учёта текущих остатков на складе
- Атрибуты:
 - * id - уникальный идентификатор
 - * номенклатура
 - * количество товара
 - * место хранения

- * номер партии

— **Заказ**

- Сущность для оформления заказов
- Атрибуты:
 - * id - уникальный идентификатор
 - * дата
 - * контрагент
 - * ответственный за заказ

— **Заказ товар**

- Сущность для сопоставления заказа и товаров
- Атрибуты:
 - * id заказа
 - * id товара
 - * количество
 - * прайс

— **Контрагент**

- Сущность для хранения данных контрагентов
- Атрибуты:
 - * id - уникальный идентификатор
 - * название
 - * тип (поставщик/покупатель)
 - * юридический статус
 - * юридический адрес
 - * контактное лицо
 - * телефон

— **Продажи**

- Сущность для учёта продаж
- Атрибуты:
 - * номер чека
 - * контрагент
 - * заказ
 - * статус продажи
 - * итоговая цена

— **Места хранения**

- Справочник складских помещений
- Атрибуты:
 - * id - уникальный идентификатор
 - * название места хранения
 - * адрес

— **Номенклатура**

- Основной справочник товаров/продукции
- Атрибуты:
 - * id - уникальный идентификатор
 - * название товара
 - * страна производства

— **Партия**

- Учёт поступлений товаров партиями
- Атрибуты:
 - * id - уникальный идентификатор
 - * дата
 - * дата годности
 - * себестоимость
 - * количество
 - * ответственный
 - * поставщик

— **Прайс**

- Справочник цен на товары
- Атрибуты:
 - * id - уникальный идентификатор
 - * номенклатура
 - * цена
 - * партия

2.3 Описание проектируемых ограничений целостности базы данных

до сих пор не понимаю, что тут должно быть

2.4 Описание всех проектируемых процедур/функций/триггеров в формате схемы

На рисунке 2.2 представлена диаграмма алгоритма триггера на обновление товара на складе при поступлении новой партии товара.

На рисунке 2.3 представлена диаграмма алгоритма триггера на обновление товара на складе при покупке.

2.5 Описание проектируемой ролевой модели на уровне базы данных

Три имеющиеся типа пользователей обладают следующим набором прав. Кладовщики – загрузка информации о партии и просмотр каталога доступных товаров; продавцы – внесение информации о заказах и продажах, просмотр каталога доступных товаров; администраторы – всё выше перечисленное.

Вывод

В конструкторском разделе была разработана структура базы данных, включающая 12 нормализованных сущностей, связанных между собой через внешние ключи, что обеспечивает целостность данных. Диаграмма базы данных наглядно демонстрирует взаимосвязи между таблицами: от справочников (номенклатура, страны, места хранения) до операционных сущностей (партии, заказы, продажи, списания). Особое внимание уделено механизмам поддержания актуальности данных через систему триггеров, автоматически обновляющих остатки на складе при поступлении новых партий и продажах. Ролевая модель, реализованная на уровне СУБД, разделяет права доступа между тремя типами пользователей (администраторами, продавцами и кладовщиками), обеспечивая безопасность и соответствие бизнес-процессам компании.

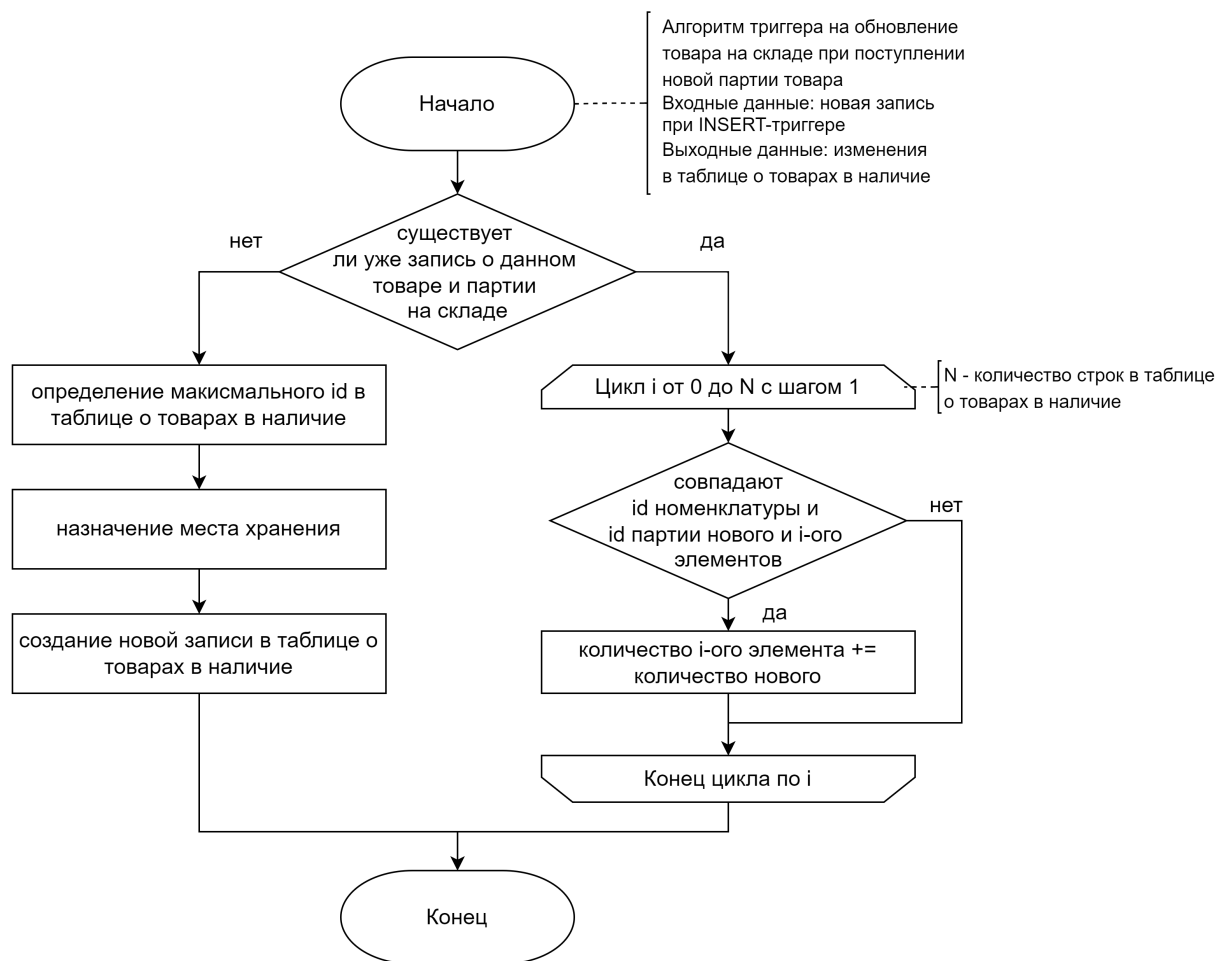


Рисунок 2.2 — Диаграмма алгоритма триггера на обновление товара на складе при поступлении новой партии товара

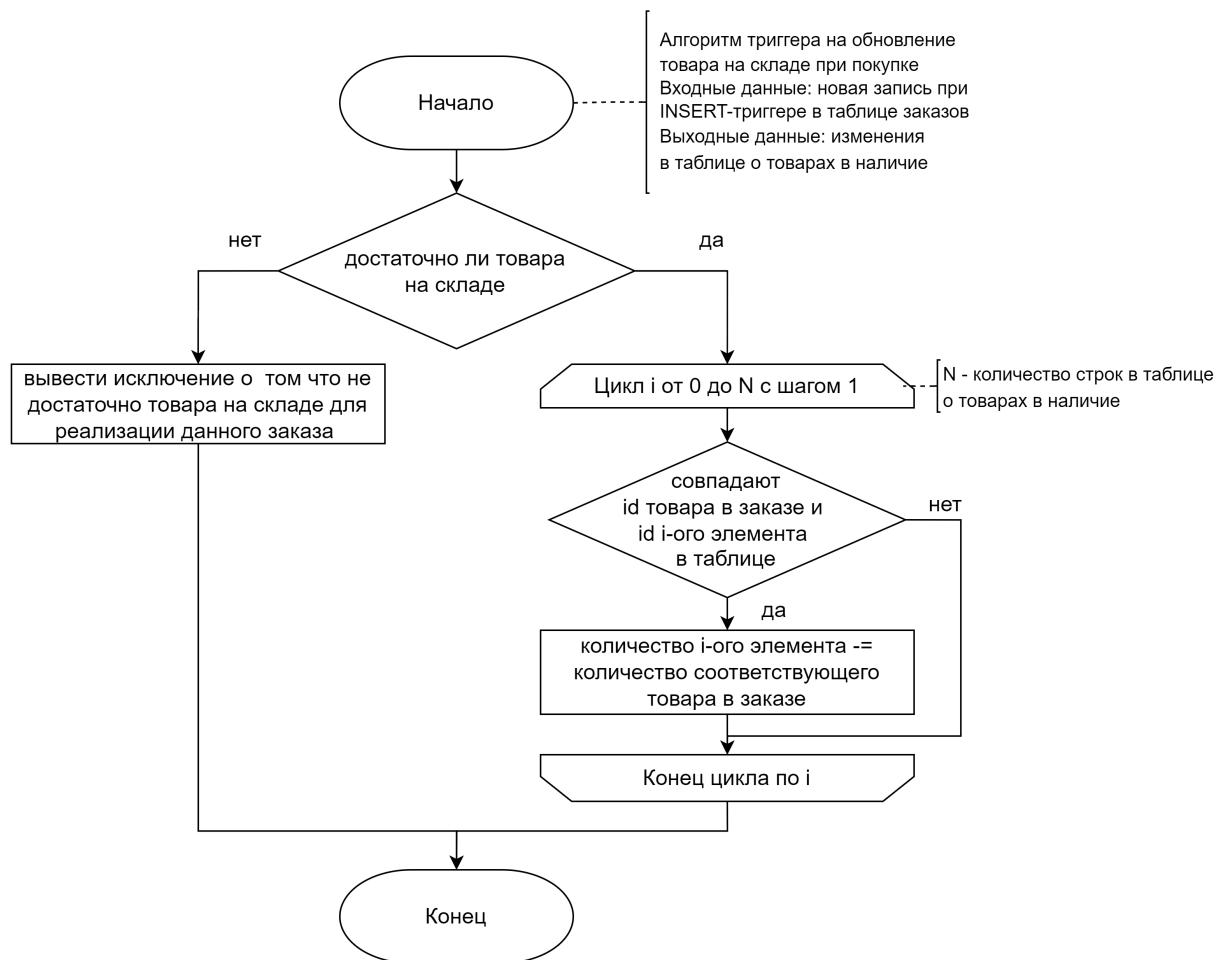


Рисунок 2.3 — Диаграмма алгоритма триггера на обновление товара на складе при покупке

3 Технологический раздел

3.1 Обоснование выбора средств реализации базы данных и приложения

Выбор PostgreSQL обусловлен его надежностью, производительностью и соответствием требованиям предметной области. Система предоставляет: полноценную поддержку ACID-транзакций для гарантии целостности данных, гибкую систему ограничений и триггеров для автоматизации бизнес-процессов, встроенную ролевую модель с детализированным управлением доступом, поддержку сложных типов данных (JSONB, ENUM), что упрощает хранение структурированной информации.

Для клиентской части выбор пал на C# в силу: строгой типизации и высокой производительности исполняемого кода, наличия проверенных инструментов интеграции с PostgreSQL (Npgsql), поддержки современных парадигм программирования и паттернов проектирования.

Данный технологический стек оптимально соответствует требованиям к безопасности, надежности и масштабируемости проектируемой системы учета.

3.2 Описание сущностей реализованной базы данных

ещё раз написать то же самое?

3.3 Описание реализованных ограничений целостности базы данных

тоже не понятно

3.4 Описание всех реализованных процедур/функций/триггеров в формате схемы

вроде бы ровно такой же пункт есть в конструкторской части

3.5 Описание ролевой модели на уровне базы данных

прямо точно какие таблицы могут insert update delete?

3.6 Описание методов тестирования и тестовых кейсов для всех разработанных на стороне базы данных функций

может ли это быть ручное тестирование?

3.7 Описание интерфейса доступа к базе данных

описание интерфейса программы?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. "Полный список документации по программным продуктам 1С [Электронный ресурс]". "Режим доступа: https://its.1c.ru/docs/program_documentation (дата обращения: 21.05.2025)".
2. А.О. Фирсов. Исследование способов применения NoSQL и реляционных баз данных. E-Scio, 2019.
3. ТКАЧЕНКО А.В. ВАСИЛЬЧИКОВА А.В. ГРИШУНОВ С.С. ОБЗОР КЛАССОВ НЕРЕЛЯЦИОННЫХ БАЗ ДАННЫХ. ЭЛЕКТРОННЫЙ ЖУРНАЛ: НАУКА, ТЕХНИКА И ОБРАЗОВАНИЕ Издательство: Общество с ограниченной ответственностью "Манускрипт 2016.