

Министерство образования и молодежной политики
Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Специальность 09.02.07: «Информационные системы и
программирование»

Разработка информационной системы складского учета медицинской
аптеки

Пояснительная записка

к курсовому проекту

КП-ПР-41-14-2026-ПЗ

Разработал:

Студент гр. ПР-41

_____/ С.А. Никитин

Руководитель

_____/ Д.О. Гариев

Министерство образования и молодежной политики

Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Специальность 09.02.07: «Информационные системы и
программирование»

Разработка информационной системы складского учета медицинской
аптеки

Курсовой проект

КП-ПР-41-14-2026-ПЗ

Оглавление

Введение.....	4
1 Постановка задачи.....	5
1.1 Описание предметной области	5
1.2 Требования к программному продукту	8
2 Проектирование.....	12
2.1 Проектирование логической структуры	12
2.2 Проектирование физической структуры	14
3 Разработка и тестирование.....	21
3.1 Реализация основных модулей	21
3.2 Тестирование	23
Заключение	27
Список использованной литературы.....	28
Приложения	29
Приложение исходного кода проекта	30

Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата						
Инв. № подл.						КП-ПР-41-14-2026-ПЗ			
	Лит	Изм.	№ докум.	Подп.	Дата				
	Разраб.					Разработка информационной системы складского учета медицинской аптеки			
	Пров.	Гариев Д.О.							
	Н. контр.	Гариев Д.О.							
	Утв.	Гариев Д.О.							
	Лит	Лист	Листов						
	3								

Введение

Курсовой проект посвящен разработке специализированной системы, направленной на автоматизацию складского учета медицинской аптеки. Проведенный анализ показал, существующие программы не всегда идеально учитывают специфику фармацевтической отрасли, такую как жесткое регулирование, необходимость отслеживания серийных номеров и обязательную отчетность. Это может приводить к ошибкам в учете, затруднять быстрый поиск лекарств и увеличивать время обслуживания покупателей.

Разрабатываемая система призвана решить эти проблемы за счет создания удобного и централизованного инструмента. Она будет обеспечивать точный контроль над всем ассортиментом, автоматизировать процессы инвентаризации и формирования заказов, а также своевременно предупреждать персонал об истекающих сроках годности препаратов.

Целью курсового проекта является создание информационной системы для автоматизации складского учета в медицинской аптеке.

Для достижения поставленной цели необходимо решить следующие задачи:

- проанализировать предметную область и существующие программные решения;
- сформировать требования к программному продукту;
- разработать архитектуру приложения и структуру данных;
- реализовать функционал с использованием WPF, Microsoft SQL Server на языке C#;
- провести тестирование и оценить работоспособность приложения.

Объектом исследования является процесс складского учета в медицинской аптеке. Предметом исследования — методы и программные средства, предназначенные для автоматизации данного процесса.

Инв. № подл.	Подп. и дата	Инв. № дцкл.	Взам. инв. №	Подп. и дата	Инв. № инв.	Подп. и дата	Инв. № подл.	Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-41-14-2026-ПЗ	Лист
														4

1 Постановка задачи

1.1 Описание предметной области

Сущность и структура предметной области

В основе предметной области лежит необходимость точного, оперативного и надежного учета товара в медицинской аптеке. Эти процессы формируют базовые сущности, с которыми ежедневно работает персонал:

- лекарственные средства, обладающие строго регламентируемыми характеристиками;
- партии товаров, поступившие от одного поставщика совокупности товаров с одинаковыми параметрами;
- первичные документы, подтверждающие операции (юридические основания для прихода и расхода товарных ценностей);
- поставщики, осуществляющие поставку фармацевтической продукции контрагенты.

Персонал аптеки ежедневно взаимодействует с этими сущностями, осуществляя приемку товаров, отпуск со склада, проведение инвентаризаций и формирование отчетности.

Чтобы понять, как функционирует складской учет в аптеке, необходимо рассмотреть его ключевые процессы.

Приемка и учет товара

Кладовщик получает товар от поставщика проверяет соответствие наименований, дозировок, производителей, серий и сроков годности. В случае соответствия, товар принимается на склад, данные вносятся в систему и формируется приходный документ.

Управление остатками и сроками годности

Подп. и дата	
Взам. инв. №	
Инв. № дцкл.	
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ				
---------------------	--	--	--	--

Лист
5

Система должна автоматически отслеживать сроки годности всех товарных партий. Это критически важно для предотвращения продажи просроченных препаратов и своевременного их списания.

Инвентаризация

Проводится периодическая сверка фактических остатков товаров на складе с данными системы. Выявленные расхождения фиксируются для поддержания точности учета.

Отпуск и списание товара

Оформляется перемещение товаров в торговый зал или их списание по причине брака или истечения срока годности. Каждая такая операция приводит к уменьшению складских остатков.

Формирование отчетности

На основе актуальных данных система формирует отчеты: по остаткам, движению товаров, оборачиваемости и просрочке, что необходимо для анализа и принятия управленческих решений.

Анализ эффективности работы аптечного склада

В рамках предметной области анализ эффективности представляет собой процесс оценки ключевых показателей работы склада. По итогам отчетных периодов система формирует отчеты, предоставляющие обратную связь о результатах работы – точность учета, оборачиваемость товаров, уровень просроченной продукции.

Проблемы существующей предметной области

Несмотря на наличие различных систем учета, медицинские аптеки сталкиваются с проблемами:

1. Отсутствие оперативного контроля остатков. Невозможность быстро получить точную информацию о наличии товаров приводит к дефициту популярных позиций или излишнему затовариванию.

Инв. № подл.	Подп. и дата	Инв. № дцкл.	Взам. инв. №	Подп. и дата					
Инв. № подл.	Подп. и дата	Инв. № дцкл.	Взам. инв. №	Подп. и дата	КП-ПР-41-14-2026-ПЗ				
Ли	Изм.	№ докум.	Подп.	Дата					
					Лист				
					6				

1. Приемка товара от поставщика и оформление накладной.
2. Регулярный контроль сроков годности и формирование отчетов по просрочке.
3. Отпуск товара в торговый зал или его списание с оформлением расходных документов.
4. Проведение плановых или внеплановых инвентаризаций для сверки остатков.
5. Формирование отчетности для анализа работы аптечного склада.

1.2 Требования к программному продукту

Требования к функциональным характеристикам

Состав выполняемых функций

Система должна обеспечивать выполнение следующих функций:

Управление товарами:

В рамках управления товарами система должна предоставлять возможность добавлять новые товары, включая их название и описание, а также редактировать и архивировать старые позиции. Кроме того, система должна позволять искать и фильтровать товары по различным параметрам.

Работа с поставщиками:

Для работы с поставщиками система должна вести список компаний-поставщиков, содержащий их контакты и реквизиты, и обеспечивать возможность быстрого поиска нужного поставщика в процессе приемки товара.

Отпуск и списание препаратов:

В части отпуска и списания система должна оформлять перемещение товаров в торговый зал и формировать акты списания с обязательным указанием причины. Также система должна контролировать операции, чтобы невозможно было списать количество товара, превышающее фактический остаток.

Прием товара:

При приеме товара система должна создавать приходные накладные и

Инв. № подл.	Подп. и дата	Инв. № дцкл.	Взам. инв. №	Подп. и дата	Инв. № подл.						Лист 8
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-41-14-2026-ПЗ						

автоматически добавлять новые партии товаров с указанием серий и сроков годности. Дополнительной функцией является автоматический расчет общей суммы поставки.

Контроль сроков:

Функция контроля сроков подразумевает, что система будет показывать товары с истекающим сроком годности и автоматически предупреждать о возникновении просрочки.

Инвентаризация:

Для проведения инвентаризации система должна создавать соответствующие ведомости для пересчета товаров, предоставлять возможность вносить данные о фактических остатках и затем сравнивать эти данные с информацией, хранящейся в системе.

Отчеты:

Система должна показывать остатки товаров на любой запрошенный момент времени, формировать отчеты по движению товаров и строить статистику, анализирующую продажи и работу с поставщиками.

Требования к надежности

Стабильность работы:

Система должна обеспечивать стабильность работы, отсутствие зависаний или аварийного завершения при работе пользователей. При возникновении ошибок необходимо выводить понятные сообщения с инструкциями для пользователя.

Сохранность данных:

Для сохранности данных требуется реализовать автоматическое сохранение данных при каждом изменении, а также регулярное резервное копирование базы данных. Система должна обеспечивать восстановление данных после сбоев электропитания.

Целостность данных:

Для обеспечения целостности данных должен быть реализован запрет на удаление товаров, которые есть в остатках или участвуют в документах. Обязательным требованием является сохранение полной истории документов.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ

Защита от ошибочных действий:

Система должна обеспечивать предотвращение дублирования товаров в базе, блокировку списания товаров с отрицательным остатком и автоматическую проверку корректности вводимых дат.

Требования к составу и параметрам технических средств

Минимальные характеристики устройства:

- компьютер с Windows 10;
- 4 ГБ оперативной памяти;
- 500 МБ свободного места;
- доступ к интернету для обновлений.

Требования к информационной и программной совместимости

Система должна быть совместима с операционными системами Windows 10 и 11. В качестве базы данных используется SQL Server. Разработка должна вестись на языке C# с использованием платформы .NET.

Специальные требования

Интерфейс системы должен быть интуитивно понятным. Необходимо обеспечить быстрый поиск товаров по названию. Для ускорения работы пользователей должны быть предусмотрены шаблоны часто используемых документов для быстрого заполнения, а также возможность массовых операций, таких как изменение цен и списание партий. Важным элементом являются визуальные подсказки и цветовое выделение для товаров с низким остатком или истекающим сроком годности.

Анализ аналогов

1С: Управление торговлей - комплексная система для автоматизации бизнеса. Ее сильные стороны включают мощный функционал для крупных компаний, глубокую аналитику и отчетность, а также интеграцию с бухгалтерскими программами. Слабые стороны заключаются в высокой стоимости лицензии и сопровождения, сложности освоения для неподготовленных пользователей и избыточности функций для небольшой аптеки. Моя система

Подп. и дата	
Взам. инв. №	
Инв. № дцкл.	
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ

Лист

10

предлагает упрощенный и специализированный подход, ориентированный именно на нужды аптечного учета, с низким порогом входа.

Excel-таблицы - распространенное решение для учета в малом бизнесе. Их сильные стороны — это полная бесплатность и доступность, гибкость настройки под свои нужды и простота начального освоения. Слабые стороны включают высокий риск ошибок при ручном вводе, отсутствие автоматизированного контроля и невозможность многопользовательской работы. Мое решение обеспечивает надежность данных, автоматизацию рутинных операций и защиту от человеческого фактора.

Специализированные программы для аптек (например, "Фармасофт"). Их сильные стороны — это учет отраслевых особенностей, работа с нормативными документами и отработанные бизнес-процессы. Слабые стороны заключаются в высокой стоимости внедрения и сложности кастомизации. Моя система предлагает более гибкий и экономичный подход с возможностью адаптации под конкретные нужды пользователя.

Инв. № подл	Подп. и дата	Инв. № дцкл.	Взам. инв. №	Подп. и дата					
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-41-14-2026-ПЗ	Лист			
						11			

2 Проектирование

2.1 Проектирование логической структуры

После проведения анализа предметной области и формулирования требований к проекту была разработана UML-диаграмма вариантов использования. Рисунок 1 демонстрирует диаграмму, которая описывает сценарии взаимодействия пользователя с приложением для учёта складского учета.

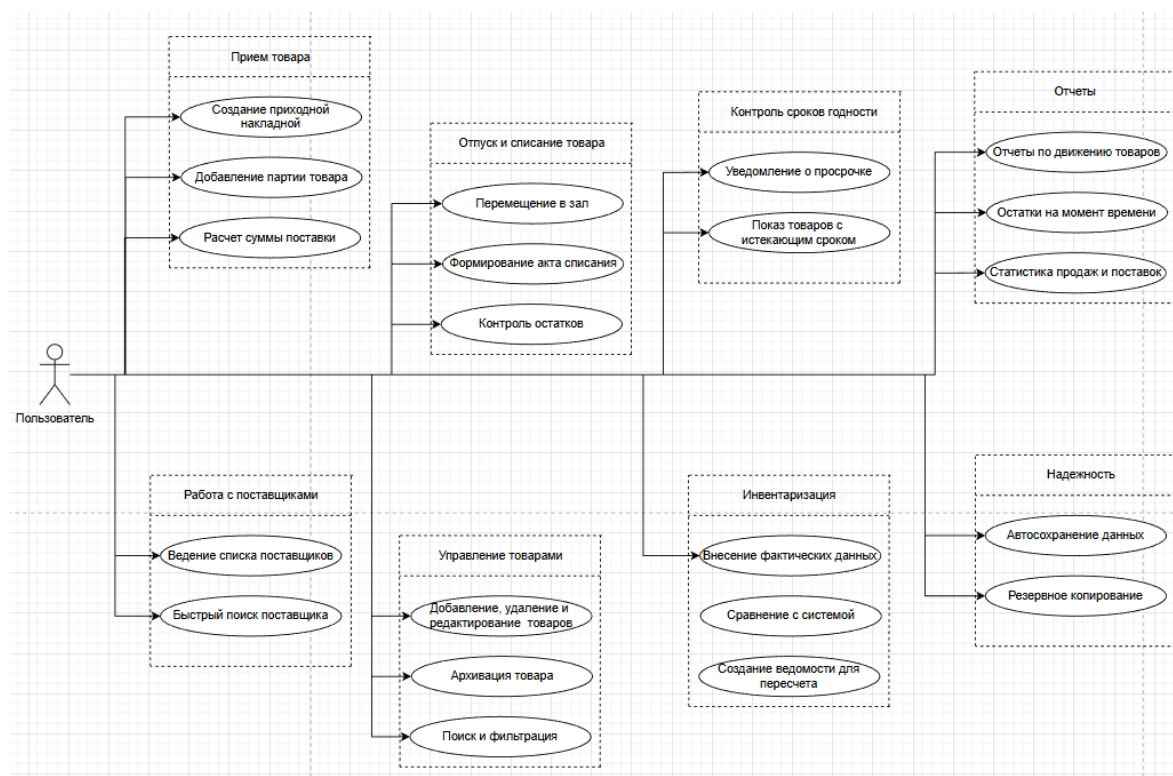


Рисунок 1 - UML-диаграмма прецедентов

На диаграмме показаны функции, связанные с управлением данными: работа с товарами и их категориями, управление поставщиками, ведение складских партий лекарственных средств. Пользователь может выполнять операции добавления, просмотра, редактирования и удаления записей, а также применять фильтрацию и поиск для эффективной работы с большими объемами информации.

Проектирование базы данных

Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ

Лист
12

Для хранения информации, необходимой для работы системы, была спроектирована база данных. Рисунок 2 отображает логическую структуру базы данных, которая определяет состав сущностей, их атрибуты и взаимосвязи. Структура включает ключевые сущности: товары, категории, поставщики, партии товаров, документы, инвентаризации, журнал движений и списания. Связи между таблицами обеспечивают целостность данных и реализацию бизнес-логики, такой как отслеживание остатков, контроль сроков годности и ведение истории операций.

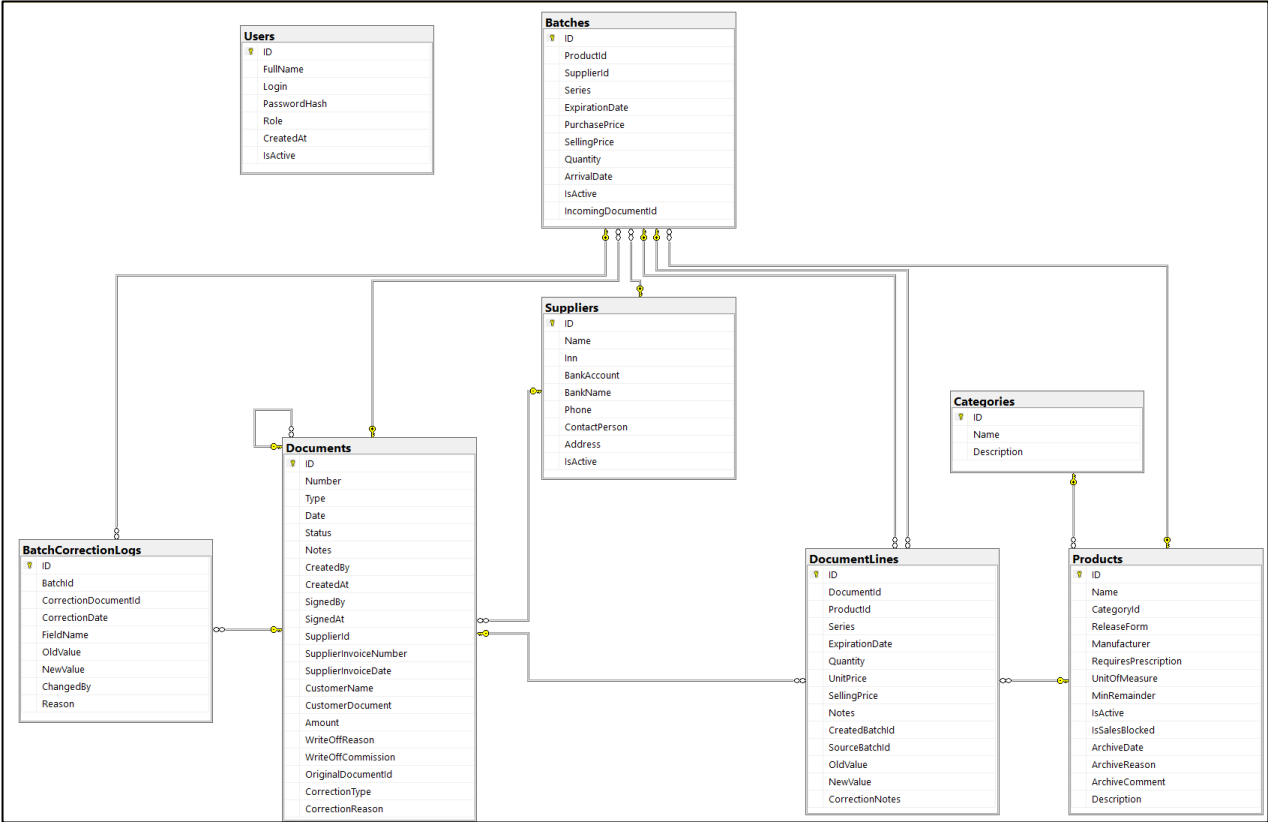


Рисунок 2 - ER-диаграмма базы данных

На основе разработанной схемы базы данных можно описать типичный сценарий взаимодействия пользователя с системой. Структура таблиц и связей между ними определяет возможности приложения.

Пользователь начинает работу с ведения справочной информации. Первым шагом осуществляется добавление новой категории товаров. Затем регистрируется поставщик медицинской продукции с указанием наименования организации, контактных данных и реквизитов. После этого вносится информация о новом

Подп. и дата
Взам. инв. №
Инв. № дцкл.
Подп. и дата
Инв. № подл.

Ли	Изм.	№ докум.	Подп.	Дата
----	------	----------	-------	------

лекарственном средстве. Для товара может быть установлен минимальный остаток, при достижении которого система будет генерировать предупреждения.

При поступлении продукции от поставщика пользователь оформляет приходный документ. В документе информация и список поступивших товаров. Для каждой позиции система автоматически информацию о поступившей партии. После проведения документа обновляется количество товаров на складе увеличиваются.

В процессе работы пользователь может осуществлять контроль за состоянием склада. Система автоматически отслеживает сроки годности всех партий и выделяет те, у которых срок истекает в ближайшее время. При необходимости пользователь инициирует проведение инвентаризации.

При отпуске товара в торговый зал или его списании по причине истечения срока годности либо брака оформляется соответствующий расходный документ. Система контролирует, чтобы количество списываемого товара не превышало фактический остаток. Каждая операция прихода или расхода отражается в журнале движений, что позволяет в любой момент восстановить полную историю по любому товару.

Для анализа эффективности работы склада пользователь обращается к разделу отчетности. Система позволяет сформировать различные отчеты. Отдельные отчеты посвящены контролю сроков годности: система показывает список товаров с истекающим сроком и уже просроченных позиций. Вся статистика доступна в наглядном виде, что позволяет пользователю оперативно принимать управленческие решения.

2.2 Проектирование физической структуры

При проектировании системы был выбран подход клиент-серверной архитектуры. Такой подход позволяет централизованно управлять данными и повышает надежность хранения информации.

Подп. и дата	
Взам. инв. №	
Инв. № дцкл.	
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ				
---------------------	--	--	--	--

Лист
14

При отборе инструментов уделялось внимание технологиям платформы Microsoft .NET, обеспечивающим надежность, производительность, глубокую интеграцию со средой Windows и соответствие поставленным функциональным задачам управления складом аптеки.

Список выбранных технологий:

1. Язык программирования: C#

Преимущества: официальный и основной язык для разработки на платформе .NET, строгая типизация и современный синтаксис, богатая стандартная библиотека, отличная поддержка со стороны интегрированной среды разработки Visual Studio.

Недостатки: ограниченная кроссплатформенность - основная поддержка ориентирована на экосистему Microsoft Windows для десктопной разработки.

Аналоги:

Java - обладает кроссплатформенностью и большим набором библиотек. *Недостатки аналога:* может требовать больше ресурсов и обладает менее глубокой интеграцией с нативными возможностями Windows по сравнению со связкой C# и WPF.

Python - позволяет быстро создавать прототипы и обладает простым синтаксисом. *Недостатки аналога:* более низкая производительность выполнения.

2. Платформа для пользовательского интерфейса: WPF

Преимущества: мощная система для создания современных десктопных интерфейсов с использованием языка разметки XAML, богатые возможности для стилизации, анимации и создания сложных визуальных элементов.

Недостатки: ориентирована преимущественно на операционную систему Windows.

Аналог:

Windows Forms - проста в освоении для создания базовых интерфейсов. *Недостатки аналога:* устаревшая технология с ограниченными возможностями, менее гибкая система привязки данных.

3. База данных: Microsoft SQL Server

Инв. № подл.	Подп. и дата	Инв. № докл.	Взам. инв. №	Подп. и дата						Лист 15
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-41-14-2026-ПЗ					

Преимущества: высокая производительность и надежность, встроенные средства обеспечения целостности, безопасности данных, отличная интеграция с платформой .NET через Entity Framework.

Недостатки: коммерческая лицензия для промышленного использования.

Аналог:

SQLite - легковесная встраиваемая СУБД, не требующая отдельного сервера.

Недостатки аналога: не предназначена для высоконагруженных многопользовательских сценариев.

PostgreSQL - бесплатная, открытая и мощная объектно-реляционная СУБД с высоким уровнем соответствия стандартам SQL. *Недостатки аналога:* инструменты управления и интеграция с .NET несколько менее развиты.

4. Объектно-реляционное отображение (ORM): Entity Framework Core

Преимущества: позволяет работать с базой данных, минимизируя написание SQL-кода, поддержка LINQ для создания типобезопасных запросов, механизмы миграций для управления эволюцией схемы базы данных прямо из кода приложения.

Недостатки: может демонстрировать меньшую производительность в экстремально сложных сценариях выборки.

Аналог:

Dapper - микро-ORM, обеспечивающий высокую скорость работы за счет прямого маппинга результатов SQL-запросов. *Недостатки аналога:* не предоставляет таких развитых механизмов миграции и отслеживания изменений сущностей.

LINQ to SQL - легковесное решение, интегрированное в .NET Framework, для простых сценариев. *Недостатки аналога:* ограниченная функциональность по, отсутствие поддержки .NET Core, устаревшая технология.

5. Архитектура: MVVM (Model-View-ViewModel)

Преимущества: подходит для WPF благодаря мощной системе привязки данных, четкое разделение ответственности между компонентами интерфейса, бизнес-логикой, состоянием и данными.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	КП-ПР-41-14-2026-ПЗ	Лист
						16
Ли	Изм.	№ докум.	Подп.	Дата		

Недостатки: увеличивает количество классов в проекте, ViewModel может стать излишне сложной, если не соблюдать принципы разделения ответственности.

Аналог:

MVC (Model-View-Controller) — классический подход для очень малых проектов. *Недостатки аналога:* часто приводит к смешиванию логики и интерфейса.

Паттерны проектирования

MVVM - выступает основным архитектурным паттерном для слоя представления, организуя взаимодействие между пользовательским интерфейсом, логикой отображения и состояния и доменными моделями.

Observer - активно применяется в WPF через механизм уведомлений об изменении свойств. Это позволяет элементам интерфейса автоматически реагировать на изменения данных в ViewModel без прямых ссылок.

Singleton - гарантирует наличие единственного экземпляра критически важных служб, таких как основной контекст базы данных или сервис для глобальных настроек и логирования.

Adapter - применяется для преобразования данных из формата моделей в формат, удобный для отображения в элементах управления WPF, таких как DataGrid или ComboBox.

Command - паттерн, инкапсулирующий запрос на выполнение действия в объект. В WPF реализуется через интерфейс ICommand и используется для привязки действий пользователя (клики по кнопкам) к методам в ViewModel.

Repository - абстрагирует работу с разными источниками данных, предоставляя компонентам приложения интерфейс для операций с сущностями.

Макеты интерфейса приложения

Макеты служат визуальным прототипом, который демонстрирует, как будут организованы основные экраны и элементы управления, позволяя определить структуру навигации, расположение компонентов, не углубляясь на данном этапе в детали визуального оформления.

Подп. и дата	
Взам. инв. №	
Инв. № дцкл.	
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ

Рисунок 3 демонстрирует макет главного окна системы. Это центральный интерфейс, обеспечивающий доступ ко всем функциям складского учета. На главном окне пользователь может переходить к основным разделам, доступны быстрые действия для создания документов и глобальный поиск по товарам, отображается актуальная информация о состоянии системы, подключении к базе данных и количестве товаров в учете.

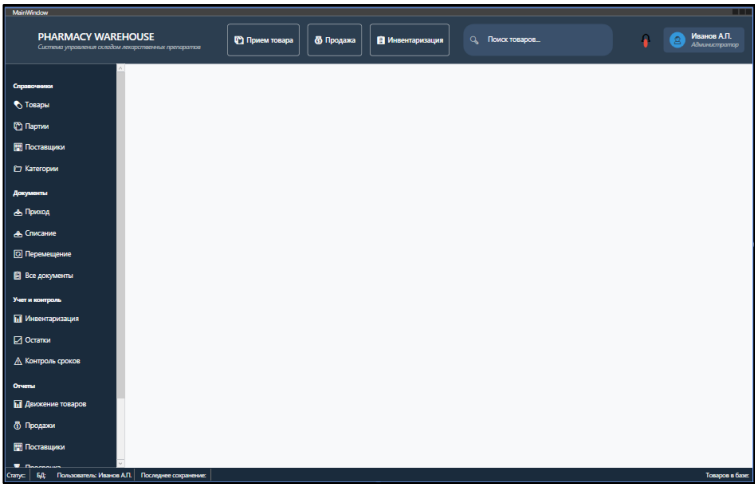


Рисунок 3 - Макет главного окна системы

Рисунок 4 показывает макет страницы управления категориями товаров. На данной странице пользователь может просматривать структуру всех категорий товаров, добавлять новые категории, редактировать или удалять существующие, реализован поиск по названию или описанию категории. В правой части экрана отображается список всех товаров, принадлежащих выбранной категории. Также на этой странице система предоставляет статистику по категориям.

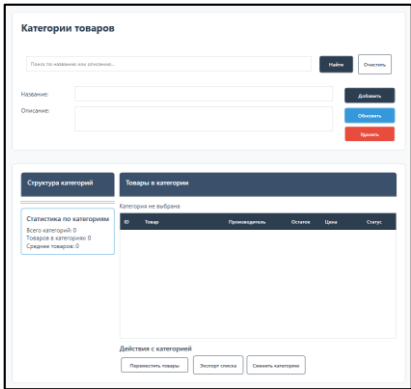


Рисунок 4 - Макет страницы управления категориями товаров

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата
Инв. № подл.

Ли	Изм.	№ докум.	Подп.	Дата

Рисунок 5 отображает макет страницы оформления приходной. На данной странице пользователь может создавать приходную накладную, необходимую для учёта поступления товаров на склад. Она позволяет корректно зафиксировать каждую партию товара, что необходимо для ведения точного складского остатка.

Приходная накладная

Оформление поступления товара от поставщика

Сохранить черновик

Провести документ

Отмена

Копировать номер

Статистика

Номер документа:Новый

Поставщик:*

Дата документа:Выбор даты

Номер счета поставщика:

Дата счета поставщика:Выбор даты

Примечание:

Позиций:0

Общее количество:0

Общая сумма:0.00 руб

Позиции прихода

Выбрано: 0

Товар *	Количество *	Цена закупки *	Сумма	Серия товара*	Срок годности*
			0.00		Выбор даты

Добавить

№	Товар	Серия	Кол-во	Цена	Сумма	Партия	Срок годности
---	-------	-------	--------	------	-------	--------	---------------

Информация о товаре

Не выбран товар

Категория: -

Производитель: -

Текущий остаток: 0

Минимальный остаток: 0

Рисунок 5 - Макет страницы оформление приходной накладной

Рисунок 6 демонстрирует ведение справочника всех лекарственных средств и медицинских изделий, хранящихся на складе. Пользователь может добавлять новые товары, внося полную информацию. Для работы с большим списком предусмотрены инструменты поиска по названию и фильтрации по категории.

Товары и препараты

Управление товарами аптеки

Добавить товар

Редактировать

В архив

Удалить

Обновить

Фильтры и поиск

Поиск (название, производитель, описание)

Категория

Рецептурный

Наличие

Применить

Сбросить

Показывать архивные

Только с истекающим сроком

Только пропущенные

Статистика

Всего: 0

Найдено: 0

В наличии: 0

Истекает срок: 0

Пропущено: 0

Список товаров

Наименование товара	Категория	Форма выпуска	Производитель	Лд, инв.	Остаток	Срок годности	Рецепт	Описание	Активный
---------------------	-----------	---------------	---------------	----------	---------	---------------	--------	----------	----------

Показывать по: 25

Страница 1 из 1

Рисунок 6 - Макет страницы управления товарами

Рисунок 7 это окно которое открывается при создании нового товара или изменении данных существующего.

Подп. и дата
Взам. инв. №
Инв. № дубл.
Подп. и дата
Инв. № подл.

Ли	Изм.	№ докум.	Подп.	Дата

Название товара*

Обязательное поле

Категория

Форма выпуска

Производитель

Единица измерения

Отмена

Сохранить

Рисунок 7 - Макет диалогового окна добавления/редактирования товара

Рисунок 8 демонстрирует взаимосвязи между основными экранами приложения, демонстрирующая логику переходов и структуру навигации, показана

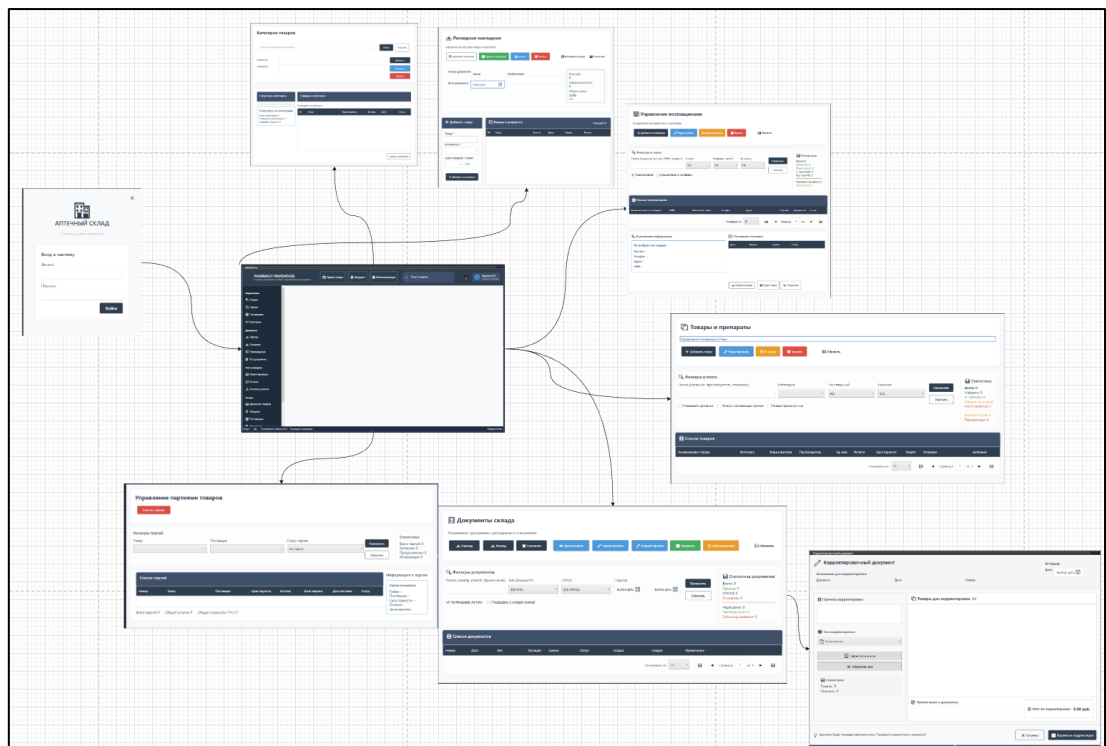


Рисунок 8 - Схема взаимодействия функциональных экранов.

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

<i>Ли</i>	<i>Изм.</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>

3 Разработка и тестирование

3.1 Реализация основных модулей

В данном разделе рассматриваются программные модули системы складского учета аптеки, реализующие основной функционал системы. Каждый модуль решает определенную задачу и взаимодействует с другими компонентами системы.

Система включает следующие ключевые модули:

- модуль управления товарами - обеспечивает ведение полного справочника препаратов, включая основные характеристики;
- модуль документооборота - реализует полный цикл обработки складских документов: от создания приходных накладных с формированием партий до оформления расходных документов со списанием товаров и составления актов списания по различным причинам;
- модуль управления партиями - осуществляет контроль сроков годности каждой партии товара, автоматически рассчитывает статусы (просроченные, истекающие);
- модуль поставщиков - ведет базу данных контрагентов с полной контактной и реквизитной информацией.
- модуль корректировок - позволяет исправлять ошибки в уже проведенных документах путем создания корректировочных документов.

Рисунок 9 иллюстрирует модуль управления партиями товаров. Модуль автоматически рассчитывает статусы партий на основе сроков годности. При проверке каждой партии система определяет количество дней до истечения срока годности и устанавливает соответствующий статус: "Просрочен", "Истекает скоро" или "Норма".

Подп. и дата	
Взам. инв. №	
Инв. № дцкл.	
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ

```
[NotMapped]
Ссылка: 6
public bool IsExpired => ExpirationDate <= DateTime.FromDateTime(DateTime.Now);

[NotMapped]
Ссылка: 6
public bool IsExpiringSoon
{
    get
    {
        var daysUntilExpiration = ExpirationDate.DayNumber - DateTime.FromDateTime(DateTime.Now).DayNumber;
        return daysUntilExpiration > 0 && daysUntilExpiration <= 30;
    }
}

[NotMapped]
Ссылка: 1
public int DaysUntilExpiration => ExpirationDate.DayNumber - DateTime.FromDateTime(DateTime.Now).DayNumber;

[NotMapped]
Ссылка: 0
public string ExpirationStatus
{
    get
    {
        if (IsExpired) return "Просрочен";
        if (IsExpiringSoon) return $"Истекает через {DaysUntilExpiration} дн.";
        return "В норме";
    }
}

[NotMapped]
Ссылка: 0
public Brush ExpirationStatusBrush
{
    get
    {
        if (IsExpired) return Brushes.Red;
        if (IsExpiringSoon) return Brushes.Orange;
        return Brushes.Green;
    }
}
```

Рисунок 9 - Расчет статусов партий

Рисунок 10 демонстрирует модуль создания корректировочных документов, который реализует функцию исправления ошибок в ранее созданных приходных накладных. Модуль обеспечивает загрузку оригинального документа, проверку его типа, создание нового корректировочного документа и установку связи между ними для отслеживания истории изменений.

```
var originalDoc = _db.Documents
    .Include(d => d.DocumentLines)
    .ThenInclude(l => l.CreatedBatch)
    .FirstOrDefault(d => d.Id == originalDocumentId);

if (originalDoc == null)
    throw new Exception("Оригинальный документ не найден");

var correctionDoc = new Document
{
    Type = DocumentType.Correction,
    Number = GenerateDocumentNumber(DocumentType.Correction),
    Date = DateTime.Now,
    OriginalDocumentId = originalDocumentId,
    CorrectionType = Models.CorrectionType.Quantity,
    CorrectionReason = reason,
    CreatedAt = DateTime.Now,
    CreatedBy = Environment.UserName,
    Status = DocumentStatus.Draft
};

_db.Documents.Add(correctionDoc);
_db.SaveChanges();
```

Рисунок 10 - Создание корректировочных документов

Рисунок 11 показывает метод архивации товара. Система проверяет наличие остатков товара на складе и предоставляет пользователю различные сценарии

Подп. и дата
Взам. инв. №
Инв. № дцкл.
Подп. и дата
Инв. № подл.

Ли	Изм.	№ докум.	Подп.	Дата

архивации: полное списание остатков или перевод в режим дораспродажи. Метод сохраняет причину архивации, дату и комментарий для аудита.

```
public void ArchiveProduct(int productId, string reason, bool blockSales,
    string comment = null)
{
    var product = _db.Products
        .Include(p => p.Batches)
        .FirstOrDefault(p => p.Id == productId);

    if (product == null) return;

    product.IsActive = false;
    product.IsSalesBlocked = blockSales;
    product.ArchiveDate = DateTime.Now;
    product.ArchiveReason = reason;
    product.ArchiveComment = comment;

    var archiveNote = $"{DateTime.Now:dd.MM.yyyy HH:mm}" +
        $"{reason}" +
        $"{product.CurrentStock} шт." +
        $"{(blockSales ? "Заблокирован" : "Дораспродажа")}";

    if (!string.IsNullOrEmpty(comment))
        archiveNote += $"{comment}";

    _db.SaveChanges();
    GetAll();
}
```

Рисунок 11 - Архивация товара

3.2 Тестирование

Для проверки работоспособности основных функций системы складского учета было проведено ручное тестирование. Тестирование выполнялось в среде разработки Visual Studio 2022.

Тестированию подверглись следующие модули: управления товарами; документооборота; управления партиями; система фильтрации и поиска.

Количество проверенных функциональных сценариев: 20.

Методика тестирования:

Тестирование проводилось по принципу "черного ящика" - взаимодействие с интерфейсом приложения без доступа к исходному коду.

Таблица 1 показывает результаты тестирования работы с товарами.

Таблица 1 - Тестирование работы с товарами

Действие	Ожидаемый результат	Фактический результат
Добавление нового товара	Товар появляется в списке	Товар добавлен успешно

Подп. и дата	
Взам. инв. №	
Инв. № дцкл.	
Подп. и дата	
Инв. № подл.	

Таблица 1 - Продолжение таблицы

Редактирование данных товара	Изменения сохраняются	Данные обновляются
Архивация товара	Товар помечается как неактивный	Товар архивирован
Фильтрация по категории	Отображаются товары выбранной категории	Фильтрация работает

Таблица 1 -Конец таблицы

Таблица 2 демонстрирует результаты тестирования работы документооборота.

Таблица 2 - Тестирование работы документооборота

Действие	Ожидаемый результат	Фактический результат
Создание приходной накладной	Документ создается, партии формируются	Документ создан успешно
Оформление продажи	Товар списывается, документ создается	Продажа оформлена
Создание акта списания	Товар списывается со склада	Списание выполнено
Корректировка документа	Создается корректировочный документ	Корректировка создана

Таблица 3 демонстрирует результаты тестирования работы с партиями.

Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ

Таблица 3 - Тестирование работы с партиями

Действие	Ожидаемый результат	Фактический результат
Просмотр статусов партий	Отображаются статусы (просрочен, истекает)	Статусы отображаются
Фильтрация по поставщику	Отображаются партии выбранного поставщика	Фильтрация работает
Поиск по серии	Находятся партии с указанной серией	Поиск работает
Просмотр истории партии	Отображается история изменений	История отображается

Таблица 4 отображает результаты тестирования работы системы фильтрации.

Таблица 4 - Тестирование системы фильтрации

Действие	Ожидаемый результат	Фактический результат
Поиск товара по названию	Находятся товары по запросу	Поиск работает
Фильтрация по наличию	Отображаются товары в наличии/отсутствующие	Фильтрация работает
Сортировка по остатку	Товары сортируются по количеству	Сортировка работает
Комбинированная фильтрация	Применяются несколько фильтров одновременно	Фильтрация работает

Ручное тестирование подтвердило работоспособность основных функций системы складского учета. Критические ошибки не обнаружены. Все тестовые сценарии выполнены успешно. Незначительные проблемы интерфейса не влияют

Инв. № подл.	Подп. и дата
Инв. № докл.	Взам. инв. №
Подп. и дата	

на основную функциональность системы. Таким образом, разработанная архитектура и реализация основных модулей полностью соответствуют поставленным требованиям и обеспечивают автоматизацию ключевых процессов складского учета в аптечной организации.

Инв. № подл.	Подп. и дата				Взам. инв. №	Инв. № дцкл.	Подп. и дата	Инв. № подл.	
Ли	Изм.	№ докум.	Подп.	Дата	КП-ПР-41-14-2026-ПЗ				Лист
									26

Список использованной литературы

1. Бен-Ган И. T-SQL. Основы [Текст] / И. Бен-Ган. – Санкт-Петербург: Питер, 2012. – 592 с.
2. Рикарди Д. Программирование на C# для начинающих. Самый простой способ изучить C# [Текст]: учебное пособие / Д. Рикарди. – Санкт-Петербург: Питер, 2022. — 336 с.
3. Леффингуэлл Д. Принципы работы с требованиями к программному обеспечению. Универсальный практикум [Текст] / Д. Леффингуэлл, Д. Уидриг. – Москва: Вильямс, 2020. – 432 с.
4. Скит Дж. C# для профессионалов. Тонкости программирования [Текст] / Дж. Скит. – Москва: Диалектика, 2021. – 608 с.
5. Троелсен Э. Язык программирования C# 9.0 и платформа .NET 5 [Текст] / Э. Троелсен. – Москва: Диалектика, 2021. – 960 с.
6. Автоматизация учета движения товаров в аптеке: программы, журнал, контроль и интеграции с Честным Знаком [Электронный ресурс]. – Режим доступа: <https://www.cleverence.ru/articles/auto-busines/-avtomatizatsiya-ucheta-dvizheniya-tovarov-v-apteke-osobennosti-i-resheniya/> (дата обращения: 22.11.2025).
7. Автоматизация учета товаров в аптеке: этапы и нужное ПО [Электронный ресурс]. – Режим доступа: <https://tu-don.ru/blog/avtomatizatsiya/kak-proiskhodit-protsess-avtomatizatsii-apteki/> (дата обращения: 20.11.2025).
8. Официальная документация по платформе .NET [Электронный ресурс]. — Режим доступа: <https://learn.microsoft.com/ru-ru/dotnet/> (дата обращения: 18.12.2025).

[illegible]

Приложения

[illegible]

Приложение исходного кода проекта

[illegible]



Исходный код:

<https://github.com/ghjgk1/PharmacyWarehouse.git>

Инв. № подл.	Подп. и дата
Инв. № дцкл.	Взам. инв. №
Подп. и дата	
Инв. № подл.	

Ли	Изм.	№ докум.	Подп.	Дата

КП-ПР-41-14-2026-ПЗ