Министерство науки и высшего образования РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Институт космических и информационных технологий

Кафедра вычислительной техники

|  |
| --- |
| **КУРСОВОЙ ПРОЕКТ** |
| Информационная система «Оптово-розничная продажа товаров» |
|  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Руководитель |  |  |  |  | В.С. Васильев |
|  |  | подпись, дата |  | инициалы, фамилия |
|  |  |  |  |  |  |
| Студент | КИ22-06б, 032215432 |  |  |  | А.В. Архиреев |
| номер группы, зачетной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2024

**РЕФЕРАТ**

Курсовой проект состоит из 49 страниц текста, 40 рисунков, 6 использованных источников и 1 приложения.

Проектирование программного обеспечения, информационная система, ICONIX, объектно‑ориентированное программирование.

Целью проекта является освоение навыков использования современных инструментальных сред и развитии умений самостоятельной разработки программного обеспечения с графическим интерфейсом.

Суть работы состоит в создании ПО, моделирующую информационную систему, которая позволяет управлять оптово-розничной продажей товаров. ИС будет обрабатывать данные о товарах, включая наименование, оптовую и розничную цену, а также справочную информацию. Также система будет фиксировать данные о покупателях (наименование, адрес, телефон, контактное лицо) и сделки, включая количество купленных товаров и дату покупки.

В первой главе разработана спецификация требований приложения, описаны прецеденты и формы.

Во второй главе спроектированы диаграммы пригодностей и последовательностей.

В третьей главе описаны: реализация приложения, тестирование и инструкция по эксплуатации.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc168506222)

[1 Разработка спецификации требований 4](#_Toc168506223)

[1.1 Постановка задачи 4](#_Toc168506224)

[1.2 Выявление ролей и функций, диаграмма прецедентов 4](#_Toc168506225)

[1.3 Макеты интерфейса и текстовое описание прецедентов 6](#_Toc168506226)

[1.4 Описание форматов данных 24](#_Toc168506227)

[2 Объектно-ориентированное проектирование 26](#_Toc168506228)

[2.1 Диаграммы пригодности и последовательности 26](#_Toc168506229)

[2.2.1 Прецедент «Регистрация» 26](#_Toc168506230)

[2.2.2 Прецедент «Просмотр товаров» 27](#_Toc168506231)

[2.2.3 Прецедент «Совершить покупку» 28](#_Toc168506232)

[2.2.4 Прецедент «Просмотр пользователей» 29](#_Toc168506233)

[2.2.5 Прецедент «Добавить товар» 30](#_Toc168506234)

[2.2.6 Прецедент «Добавить пользователя» 31](#_Toc168506235)

[2.2 ER-диаграмма 32](#_Toc168506236)

[2.3 Диаграмма классов 33](#_Toc168506237)

[3 Объектно-ориентированное программирование 37](#_Toc168506238)

[3.1 Выбор инструментов 37](#_Toc168506239)

[3.2 Реализация 37](#_Toc168506240)

[3.2.1 Графический интерфейс 37](#_Toc168506241)

[3.2.2 База данных 38](#_Toc168506242)

[3.3 Сборка и запуск 38](#_Toc168506243)

[3.4 Первый запуск программы 38](#_Toc168506244)

[3.5 Тестирование 39](#_Toc168506245)

[3.6 Инструкция 43](#_Toc168506246)

[Заключение 44](#_Toc168506247)

[Приложение А 45](#_Toc168506248)

[Список используемых источников 48](#_Toc168506249)

ВВЕДЕНИЕ

Компании, торгующие товарами, должны вести учёт всех покупок. Есть ряд операций, которые выполняются этими компаниями, а именно: продажа товаров, предоставление заказов, аналитика продаж, и выручка компании. При этом операции должны храниться в электронном в виде для упрощения такой задачи.

Целью курсового проекта является создание приложения «Оптово-розничная продажа товаров», которая представляет программный комплекс, предназначенный для автоматизации процессов управления торговыми операциями. Основная цель данной системы заключается в оптимизации учета товарных запасов, обработки заказов и управления продажами как в оптовом, так и в розничном сегментах. Система включает модули для ведения складского учета, анализа продаж, управления клиентскими взаимоотношениями и финансового контроля. Благодаря интеграции различных функций, информационная система обеспечивает повышение эффективности работы предприятия, улучшение качества обслуживания клиентов и снижение операционных затрат.

1 Разработка спецификации требований

1.1 Постановка задачи

Оптово-розничная продажа товаров. Компания торгует товарами из определенного спектра. Каждый товар характеризуется наименованием, оптовой ценой, розничной ценой и справочной информацией. В компанию обращаются покупатели, для каждого из которых в базе данных фиксируются стандартные данные (наименование, адрес, телефон, контактное лицо). По каждой сделке составляется документ, в котором наряду с покупателем фиксируются количество купленного им товара и дата покупки. Обычно покупатели в рамках одной сделки покупают не один товар, а сразу несколько. Также компания решила предоставлять скидки в зависимости от количества закупленных товаров и их общей стоимости.

1.2 Выявление ролей и функций, диаграмма прецедентов

Были выявлены следующие роли: покупатель, администратор, менеджер

На рисунке 1 представлена диаграмма прецедентов покупателя.



Рисунок 1 – Диаграмма прецедентов покупателя

На рисунке 2 представлена диаграмма прецедентов менеджера и администратора.

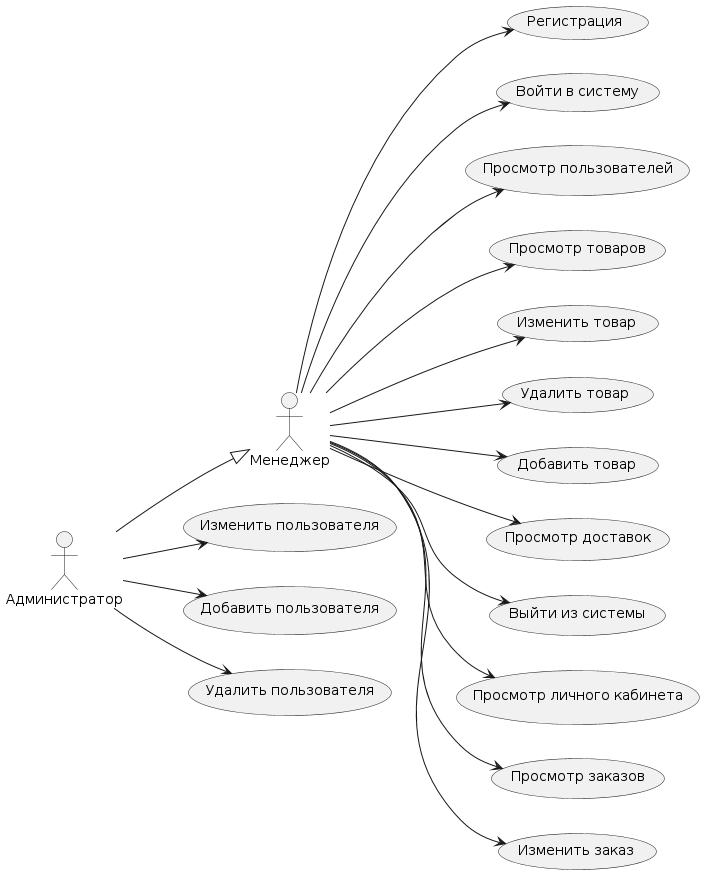


Рисунок 2 - Диаграмма прецедентов менеджера и администратора

В приложении А представлены диаграммы потоков экранов.

1.3 Макеты интерфейса и текстовое описание прецедентов

**Роли:** администратор, покупатель, менеджер.

**Цель сценария:** войти в систему.

**Предусловие:** открыто «окно входа в систему» (рисунок 3).

**Основной сценарий:**

1. Ввести логин и пароль;
2. Нажать кнопку «Войти».

**Постусловие:** В базу данных совершится запрос на поиск данного пользователя. Откроется «Домашнее окно» (рисунок 19-20).

**Альтернативный сценарий**: появится окно об ошибке.

Изображение выглядит как текст, снимок экрана, фиолетовый, Фиолетовый

Автоматически созданное описание

Рисунок 3 – Окно входа в систему

**Роль:** администратор, покупатель, менеджер.

**Цель сценария:** Зарегистрироваться.

**Предусловие:** открыто «окно регистрации» (рисунок 3).

**Основной сценарий:**

1. Ввести данные
2. Нажать кнопку «Зарегистрироваться».

**Постусловие:** В базу данных добавлена учётная запись о пользователе. В окне появится надпись об успешной регистрации.

**Альтернативный сценарий:** В окне появится надпись об ошибке, и так же красным цветом выделится определённая строчка для ввода.

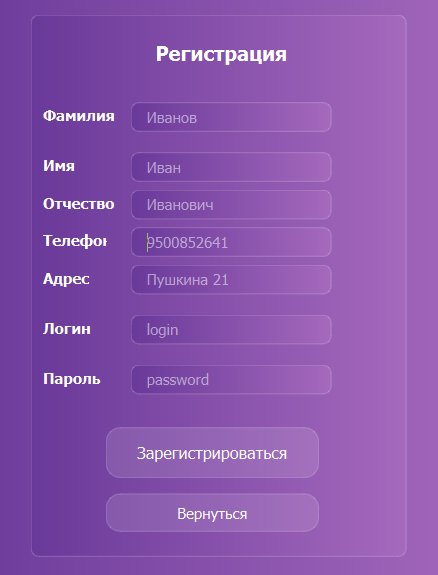


Рисунок 4 – Окно регистрации

**Роль:** покупатель.

**Цель сценария:** посмотреть список продуктов.

**Предусловие:** открыто «Домашнее окно» (рисунок 19).

**Основной сценарий:**

1. Нажать кнопку «Продукты».

**Постусловие:** открыто окно «список продуктов» (рисунок 5), в котором отображается таблица с продуктами.

**Роль:** покупатель.

**Цель сценария:** добавить товар в корзину.

**Предусловие:** открыто окно «Список продуктов» (рисунок 5).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Товара».
2. Нажать кнопку «Добавить выбранный товар в корзину».

**Постусловие:** в таблицу корзины (рисунок 6) добавляется выбранный товар. В окне списка продуктов появляется запись об успешном добавлении товара в корзину.

**Альтернативный сценарий:** при добавлении товара большего, чем в наличии, в окне списка продуктов появляется запись об ошибке.

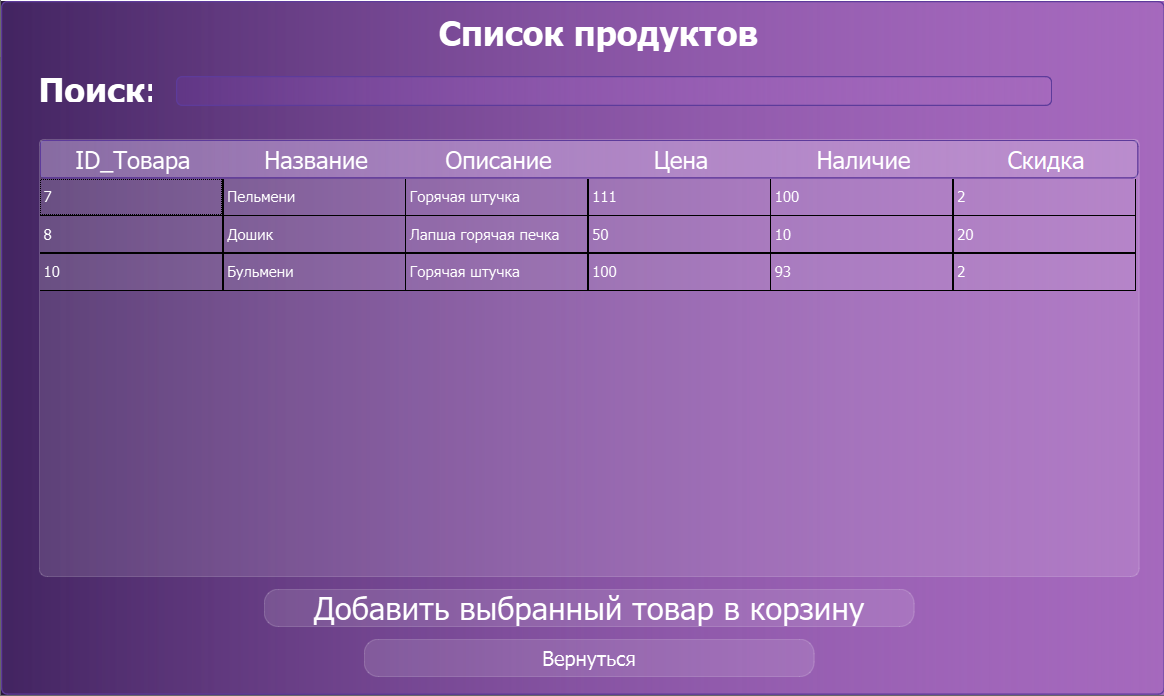


Рисунок 5 – Окно списка продуктов

**Роль:** покупатель.

**Цель сценария:** просмотреть корзину.

**Предусловие:** открыто «Домашнее окно» (рисунок 19).

**Основной сценарий:**

1. Нажать на кнопку «Корзина».

**Постусловие:** открыто окно «корзина» (рисунок 6), в котором отображается таблица с продуктами, добавленными в корзину.

**Роль:** покупатель.

**Цель сценария:** удалить выбранный товар из корзины.

**Предусловие:** открыто окно «корзина» (рисунок 6).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Товара».
2. Нажать на кнопку «Удалить выбранный товар».

**Постусловие:** в колонке у выбранного товара уменьшается количество на 1. Так же изменяется цена у этой записи и общая цена в окне.

**Альтернативный сценарий:** если перед удалением количество = 1, то запись из таблицы удаляется.

**Роль:** покупатель.

**Цель сценария:** удалить выбранный товар полностью.

**Предусловие:** открыто окно «корзина» (рисунок 6).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Товара».

2. Нажать на кнопку «Удалить товар полностью».

**Постусловие:** запись с выбранным товаром удаляется. Изменяется запись об общей цене.

**Роль:** покупатель.

**Цель сценария:** очистить корзину.

**Предусловие:** открыто окно «корзина» (рисунок 6).

**Основной сценарий:**

1. Нажать на кнопку «Очистить корзину».

**Постусловие:** все запись в таблице удаляются. Кнопка «Купить», становится неактивной и пропадает. Запись об общей цене тоже.

**Роль:** покупатель.

**Цель сценария:** совершить покупку.

**Предусловие:** открыто окно «корзина» (рисунок 6).

**Основной сценарий:**

1. Нажать на кнопку «Купить».

2. Нажать на кнопку «Купить».

**Постусловие:** если в таблице присутствуют записи, открывается окно с совершением покупки (рисунок 7), при нажатии в появившемся окне покупки кнопку купить, происходит запись в базу данных.

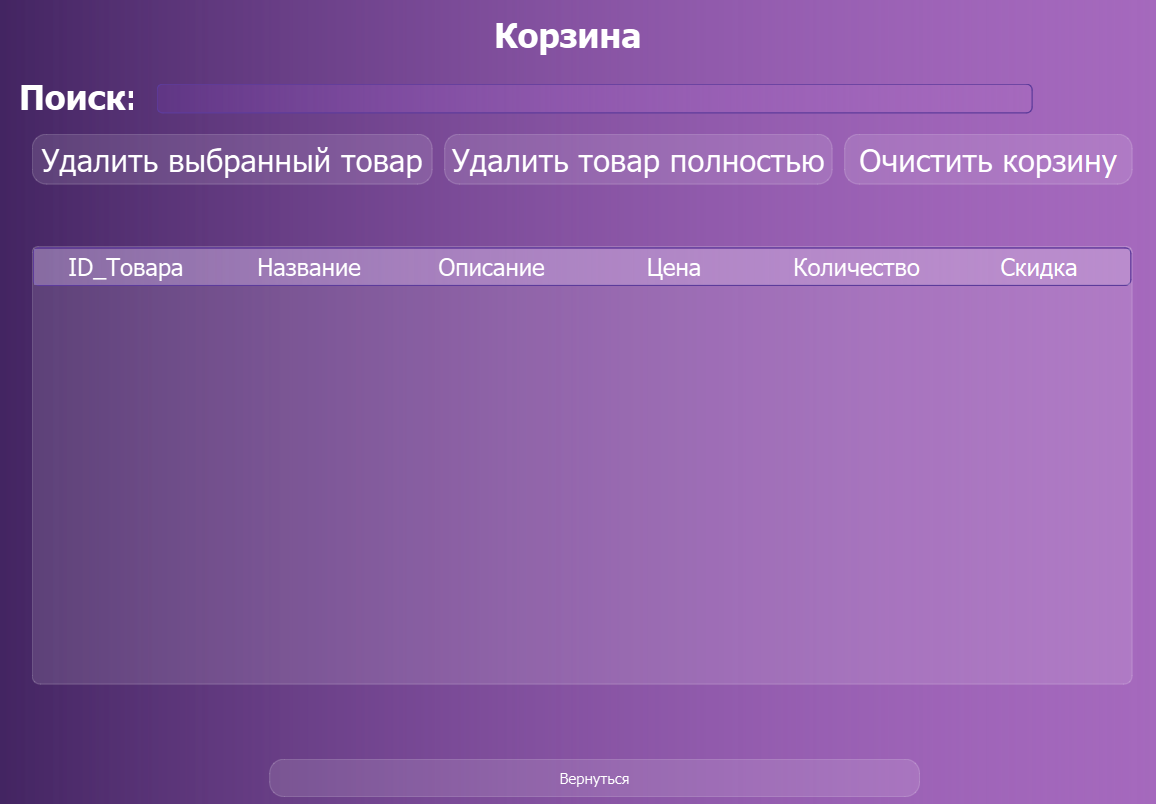


Рисунок 6 – Окно корзины

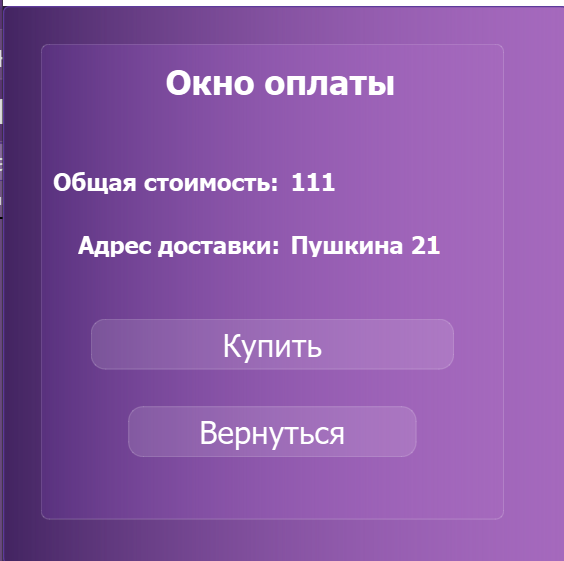


Рисунок 7 – Окно совершения покупки

**Роль:** покупатель.

**Цель сценария:** просмотреть историю покупок.

**Предусловие:** открыто «Домашнее окно» (рисунок 19).

**Основной сценарий:**

1. Нажать на кнопку «История покупок».

**Постусловие:** открыто окно «История покупок» (рисунок 8), в котором отображается таблица с историей покупок.

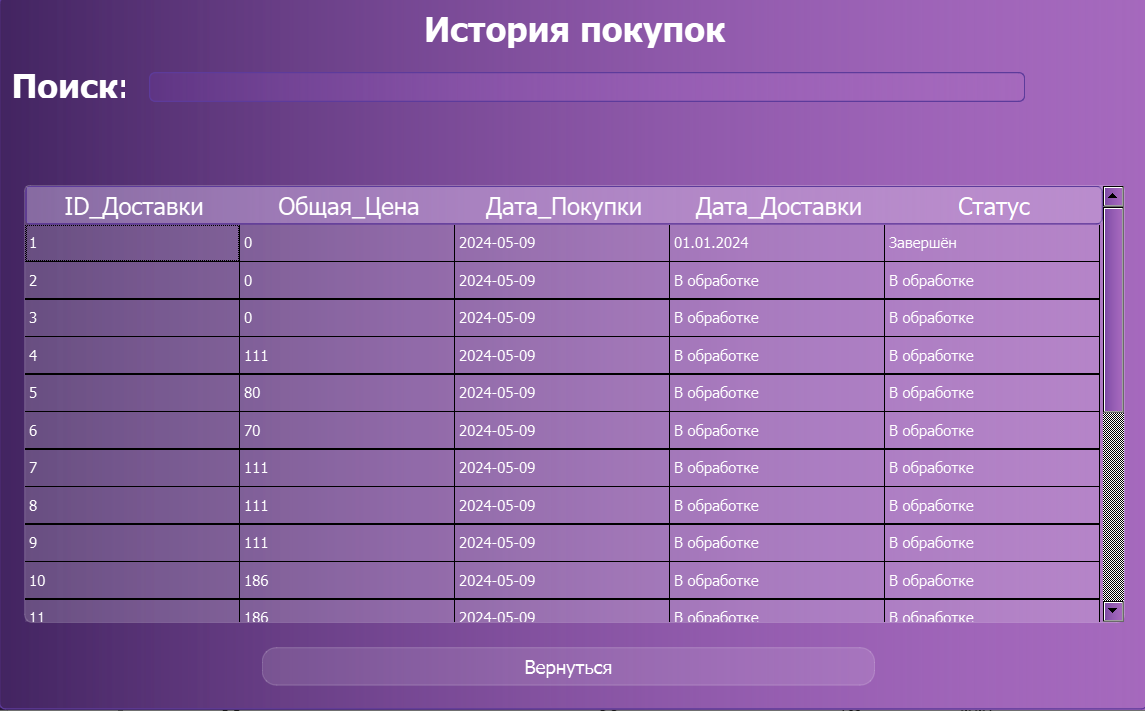


Рисунок 8 – Окно истории покупок

**Роль:** покупатель, менеджер, администратор.

**Цель сценария:** просмотреть информацию в личном кабинете.

**Предусловие:** открыто «Домашнее окно» (рисунок 19-20).

**Основной сценарий:**

1. Нажать на кнопку «Личный кабинет».

**Постусловие:** открыто окно «Личный кабинет» (рисунок 9), в котором отображается информация о пользователе.

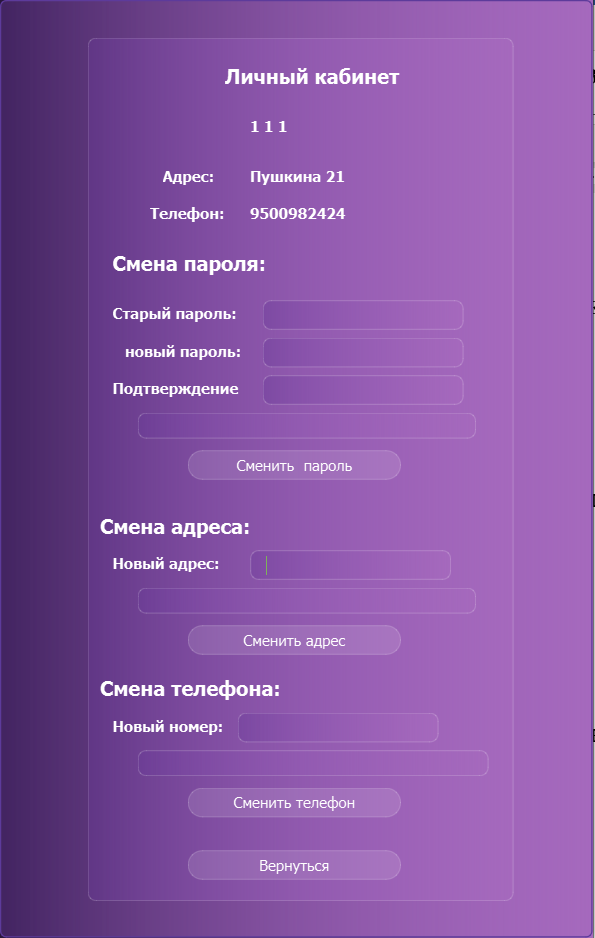


Рисунок 9 – Окно с личным кабинетом

**Роль:** менеджер, администратор.

**Цель сценария:** просмотреть список продуктов.

**Предусловие:** открыто «Домашнее окно» (рисунок 20).

**Основной сценарий:**

1. Нажать на кнопку «Продукты».

**Постусловие:** открыто окно «список продуктов» (рисунок 10), в котором отображается таблица с продуктами.

**Роль:** менеджер, администратор.

**Цель сценария:** добавить продукт.

**Предусловие:** открыто окно «Список продуктов» (рисунок 10).

**Основной сценарий:**

1. Нажать на кнопку «Добавить продукт».

2. В появившемся окне ввести данные о продукте (рисунок 11).

3. Нажать кнопку «Добавить товар».

**Постусловие:** в базу данных вносится запись товара.

**Альтернативный сценарий:** если введённые данные некорректны, в окне добавления товара появляется запись об ошибке.

**Роль:** менеджер, администратор.

**Цель сценария:** изменить продукт.

**Предусловие:** открыто окно «Список продуктов» (рисунок 10).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Товара».

2. Нажать на кнопку «Изменить продукт».

3. В появившемся окне у выбранной строки нажать кнопку «Изменить» (рисунок 12).

4. Ввести данные.

5. Нажать на кнопку «Изменить товар».

**Постусловие:** в базу данных вносится изменения товара.

**Альтернативный сценарий:** если введённые данные некорректны, в окне изменения товара появляется запись об ошибке.

**Роль:** менеджер, администратор.

**Цель сценария:** удалить продукт.

**Предусловие:** открыто окно «Продукты» (рисунок 10).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Товара».

2. Нажать на кнопку «Удалить».

3. В появившемся окне нажать кнопку «Да» (рисунок 18).

**Постусловие:** из базы данных удаляется запись товара.

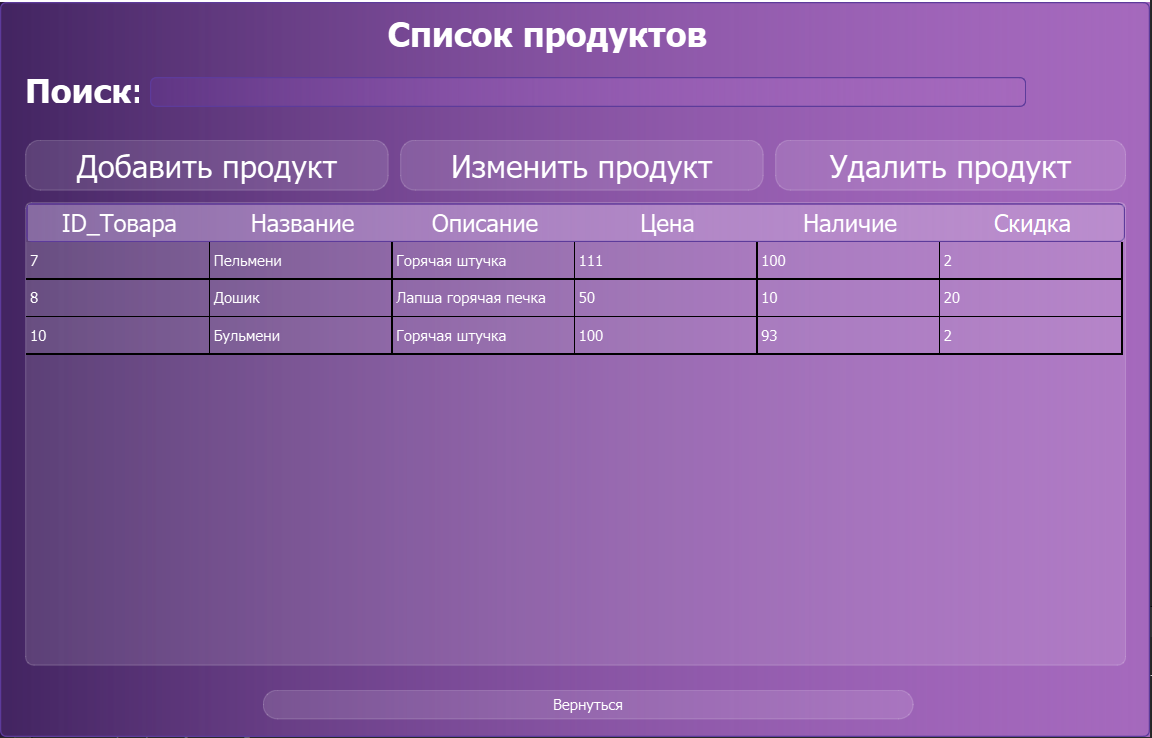


Рисунок 10 – окно со списком продуктов

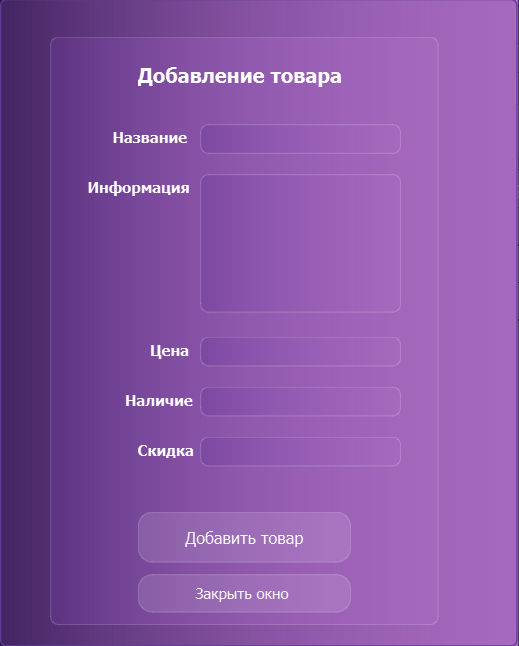


Рисунок 11 – окно добавления товара

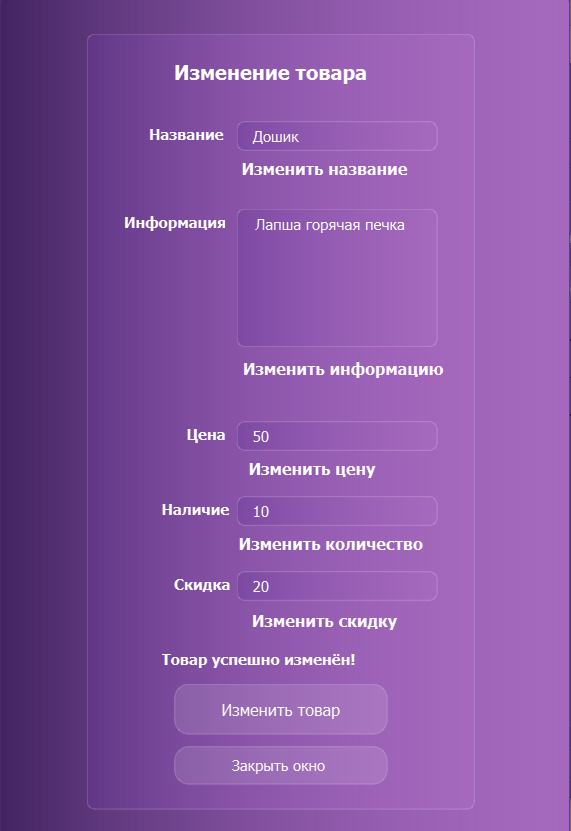


Рисунок 12 – Окно изменения товара

**Роль:** менеджер, администратор.

**Цель сценария:** просмотр списка заказов.

**Предусловие:** открыто «Домашнее окно» (рисунок 20).

**Основной сценарий:**

1. Нажать кнопку «Список заказов».

**Постусловие:** открыто окно «список заказов» (рисунок 12), в котором отображается таблица с заказами.

**Роль:** менеджер, администратор.

**Цель сценария:** изменить заказ.

**Предусловие:** открыто окно «Список заказов» (рисунок 12).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Заказа».

2. Нажать на кнопку «Изменить статус».

3. В появившемся окне у нужной строки нажать кнопку изменить (рисунок 13).

4. Ввести данные.

5. Нажать на кнопку «Изменить статус».

**Постусловие:** в базу данных вносится изменения заказа.

**Альтернативный сценарий:** если введённые данные некорректны, в окне изменения заказа появляется запись об ошибке.

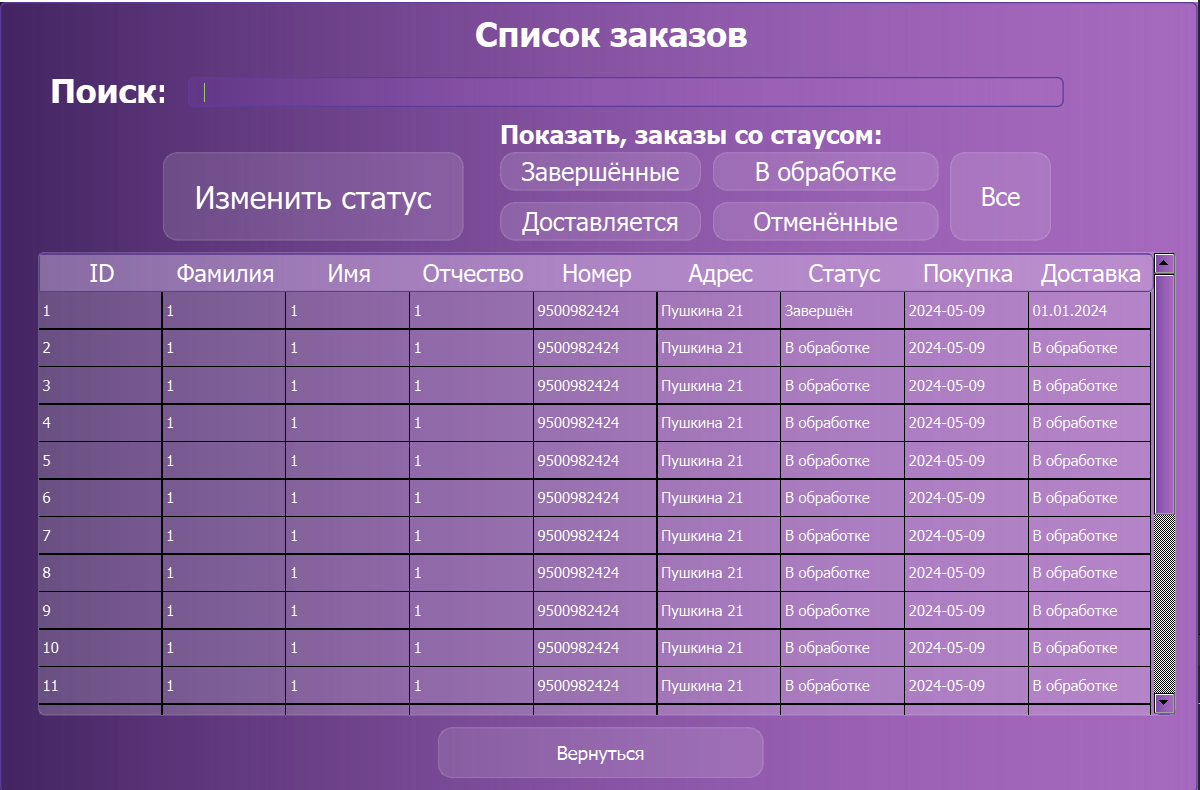


Рисунок 12 – окно списка заказов

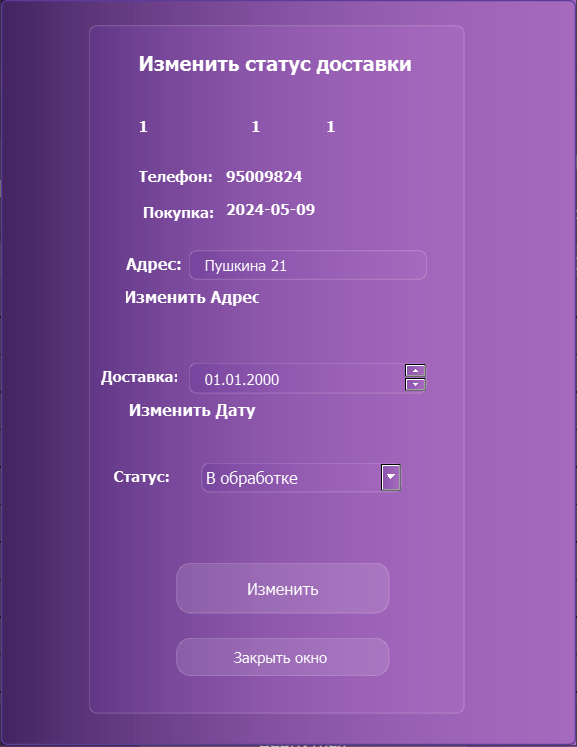


Рисунок 13 – окно изменения статуса доставки

**Роль:** менеджер, администратор.

**Цель сценария:** просмотр списка пользователей.

**Предусловие:** открыто «Домашнее окно» (рисунок 20).

**Основной сценарий:**

1. Нажать кнопку «Список пользователей».

**Постусловие:** открыто окно «список пользователей» (рисунок 14-15), в котором отображается таблица с пользователями.

**Роль:** администратор.

**Цель сценария:** добавить пользователя.

**Предусловие:** открыто окно «Список пользователей» (рисунок 15).

**Основной сценарий:**

1. Нажать на кнопку «Добавить пользователя».

2. В появившемся окне ввести данные (рисунок 16).

3. Нажать на кнопку «Добавить пользователя».

**Постусловие:** в базу данных вносится информация о пользователе.

**Альтернативный сценарий:** если введённые данные некорректны, в окне добавления пользователя появляется запись об ошибке.

**Роль:** администратор.

**Цель сценария:** изменить информацию о пользователе.

**Предусловие:** открыто окно «Список пользователей» (рисунок 15).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Человека».

2. Нажать на кнопку «Изменить пользователя».

3. В появившемся окне у нужной строки нажать кнопку изменить (Рисунок 17).

4. Ввести данные.

5. Нажать на кнопку «Изменить пользователя».

**Постусловие:** в базу данных вносится изменения о пользователе.

**Альтернативный сценарий:** если введённые данные некорректны, в окне изменения пользователя появляется запись об ошибке.

**Роль:** администратор.

**Цель сценария:** удалить пользователя.

**Предусловие:** открыто окно «Список пользователей» (рисунок 15).

**Основной сценарий:**

1. Нажать на ячейку в колонке «ID\_Человека».

2. Нажать на кнопку «Удалить».

3. В появившемся окне нажать кнопку «Да» (рисунок 18).

**Постусловие:** из базы данных удаляется запись пользователя.

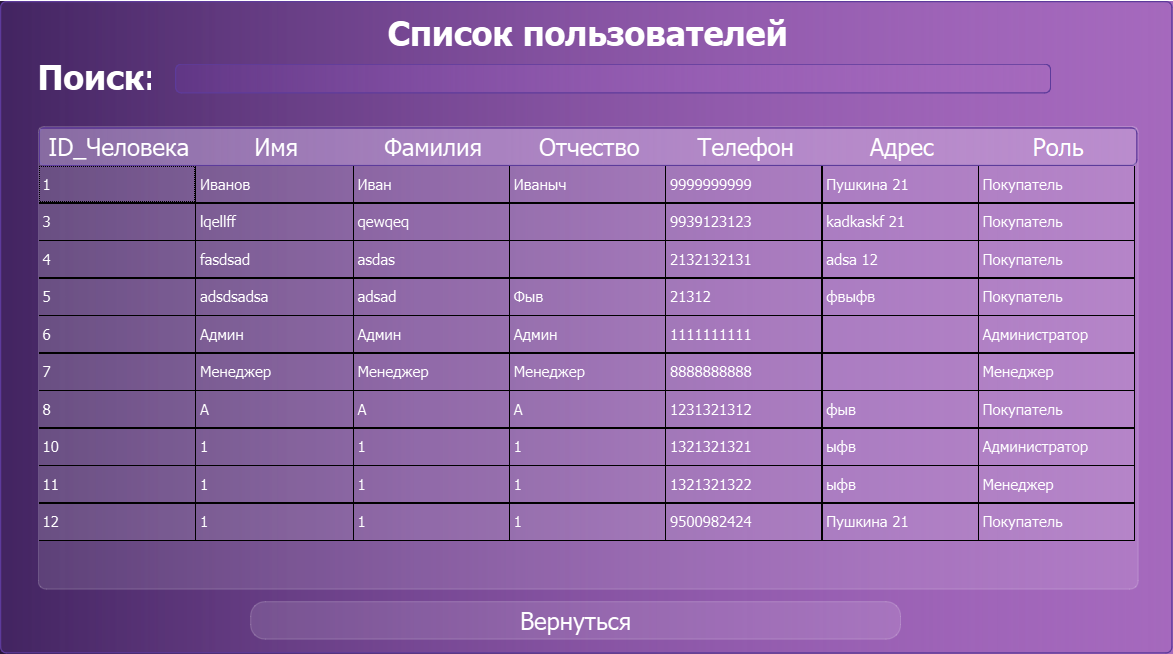


Рисунок 14 – Окно списка пользователей менеджер

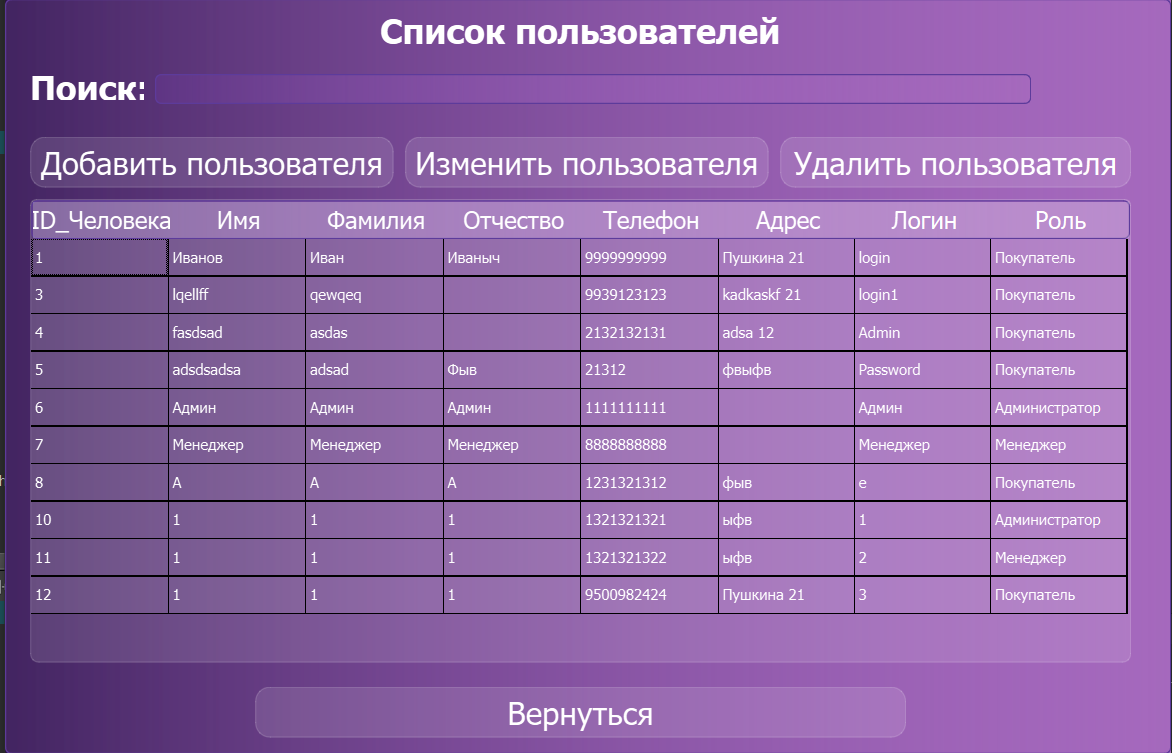


Рисунок 15 – Окно списка пользователей администратора

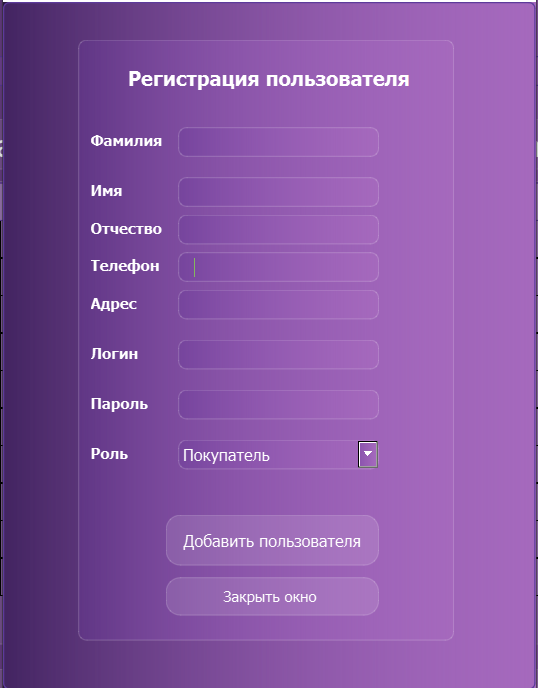


Рисунок 16 – Окно добавления пользователя

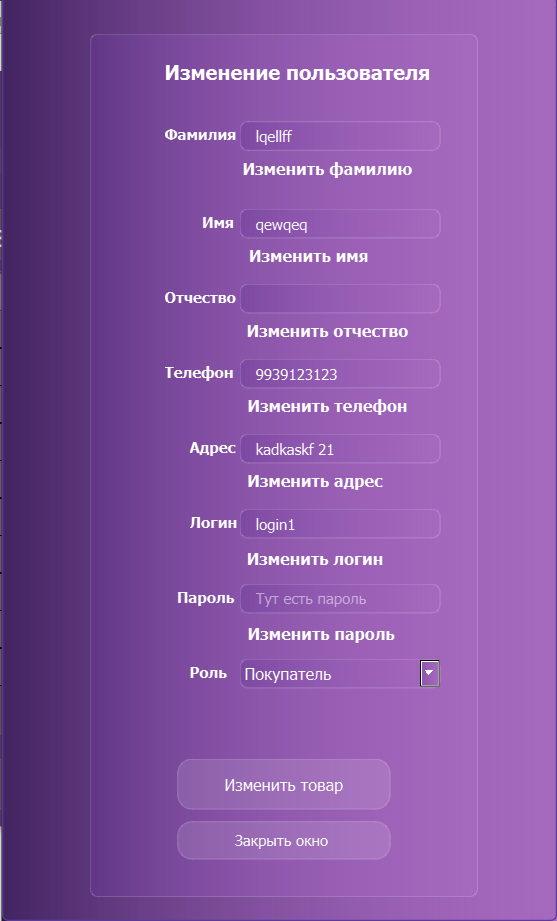


Рисунок 17 – Окно изменения пользователя

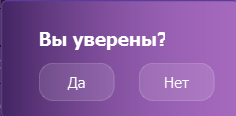


Рисунок 18 – Окно подтверждения

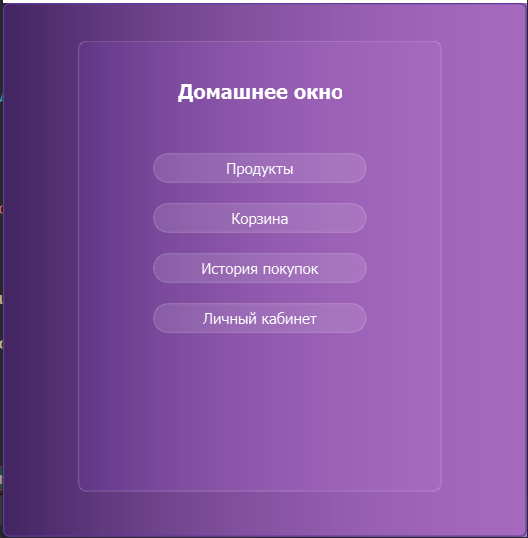


Рисунок 19 – Домашнее окно покупателя

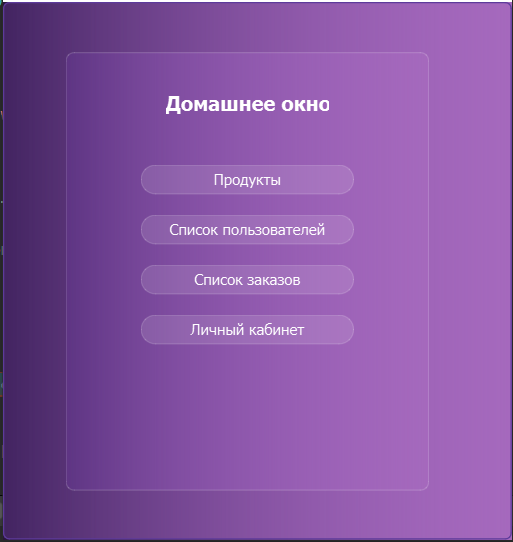


Рисунок 20 – Домашнее окно менеджера и администратора

1.4 Описание форматов данных

Важной частью функционирования системы является хранение и обработка данных. На рисунке 21 представлен пример заполнения пользователей в базу данных.



Рисунок 21 – Пример заполнения пользователей в базу данных

На рисунке 22 представлен пример заполнения товаров в базу данных.

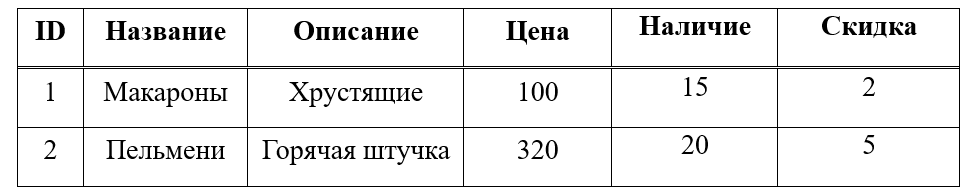


Рисунок 22 – Пример заполнения товаров в базу данных

На рисунке 23 представлен пример заполнения доставки в базу данных.

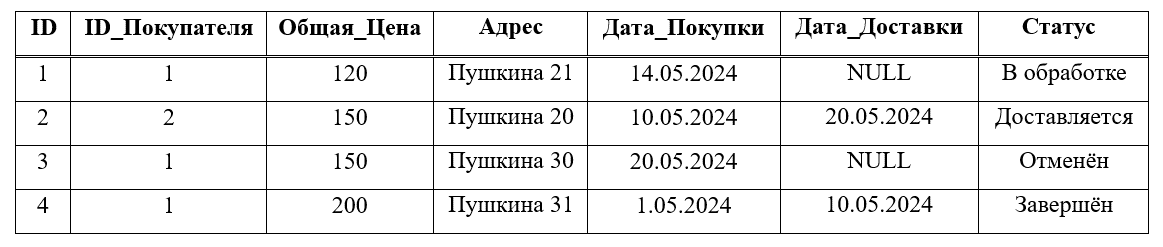


Рисунок 23 – Пример заполнения доставки в базу данных

На рисунке 24 представлен пример заполнение заказа в базу данных.



Рисунок 24 – Пример заполнения заказа в базу данных

2 Объектно**-ориентированное проектирование**

2.1 Диаграммы пригодности и последовательности

2.2.1 Прецедент «Регистрация»

На рисунке 25 представлена диаграмма пригодности для данного прецедента.

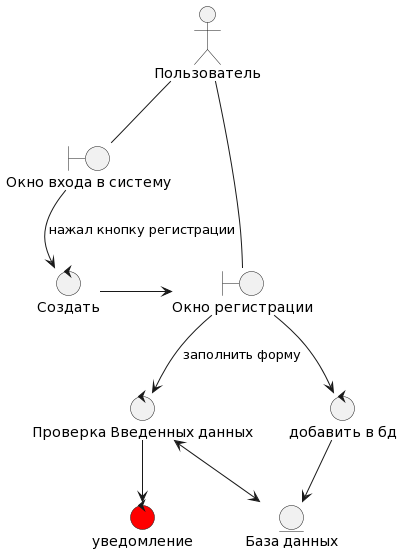


Рисунок 25 – Диаграмма пригодности прецедента «Регистрация»

На рисунке 26 представлена диаграмма последовательности для этого прецедента.

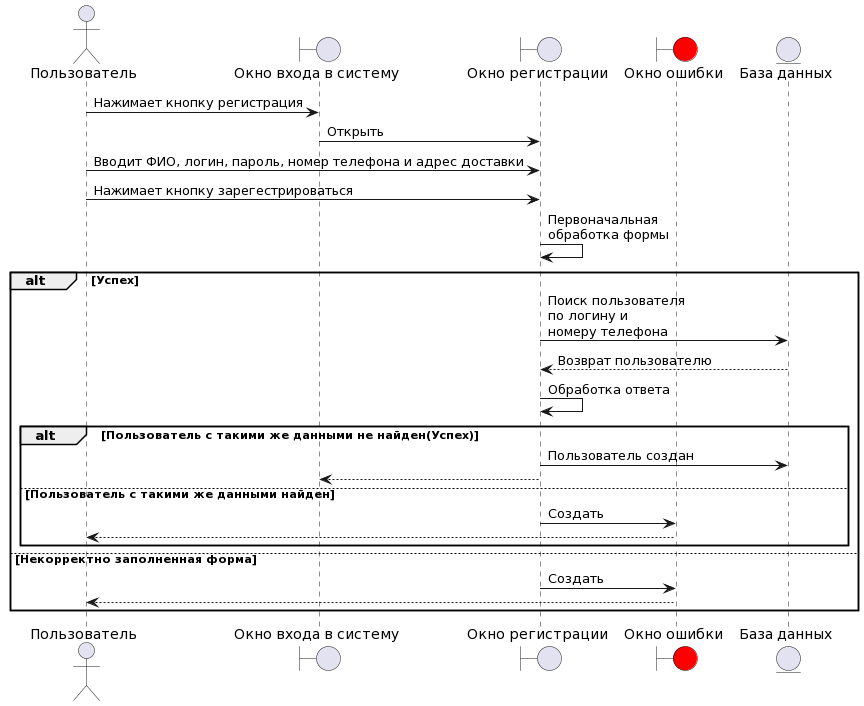


Рисунок 26 – Диаграмма последовательности прецедента «Регистрация»

2.2.2 Прецедент «Просмотр товаров»

На рисунке 27 представлена диаграмма пригодности для данного прецедента.

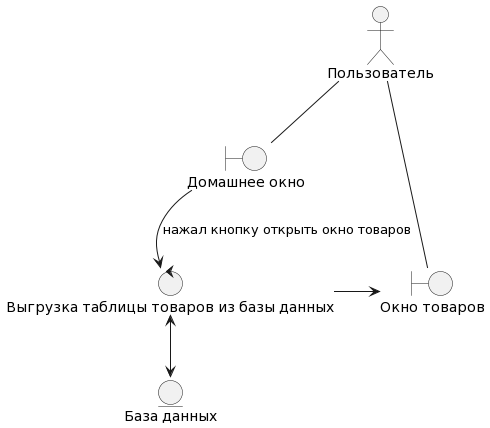


Рисунок 27 – Диаграмма пригодности прецедента «Просмотр товаров»

На рисунке 28 представлена диаграмма последовательности данного прецедента.

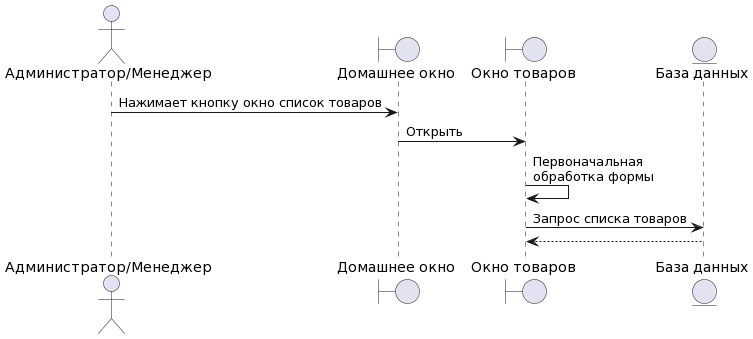


Рисунок 28 – Диаграмма последовательности прецедента «Просмотр товаров»

2.2.3 Прецедент «Совершить покупку»

На рисунке 29 представлена диаграмма пригодности для данного прецедента.



Рисунок 29 – Диаграмма пригодности прецедента «Совершить покупку»

На рисунке 30 представлена диаграмма последовательности для данного прецедента.



Рисунок 30 – Диаграмма последовательности прецедента «Совершить покупку»

2.2.4 Прецедент «Просмотр пользователей»

На рисунке 31 представлена диаграмма пригодности для данного прецедента.

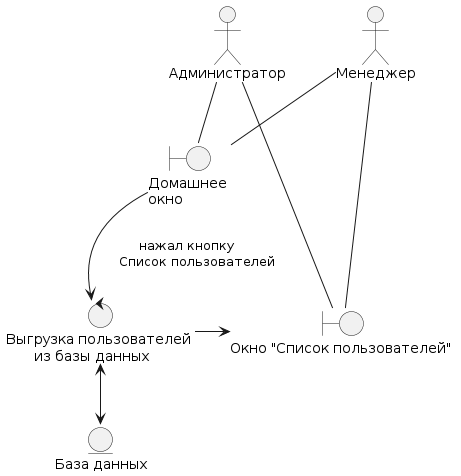


Рисунок 31 – Диаграмма пригодности прецедента «Просмотр пользователей»

На рисунке 32 представлена диаграмма последовательности для данного прецедента.

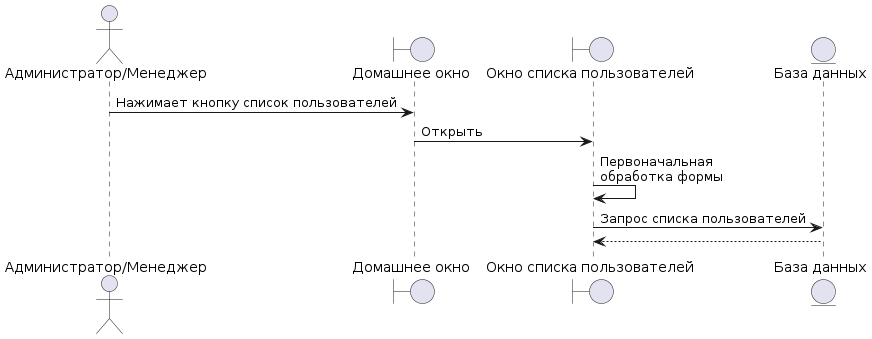


Рисунок 32 – Диаграмма последовательности прецедента «Просмотр пользователей»

2.2.5 Прецедент «Добавить товар»

На рисунке 33 представлена диаграмма пригодности для данного прецедента.

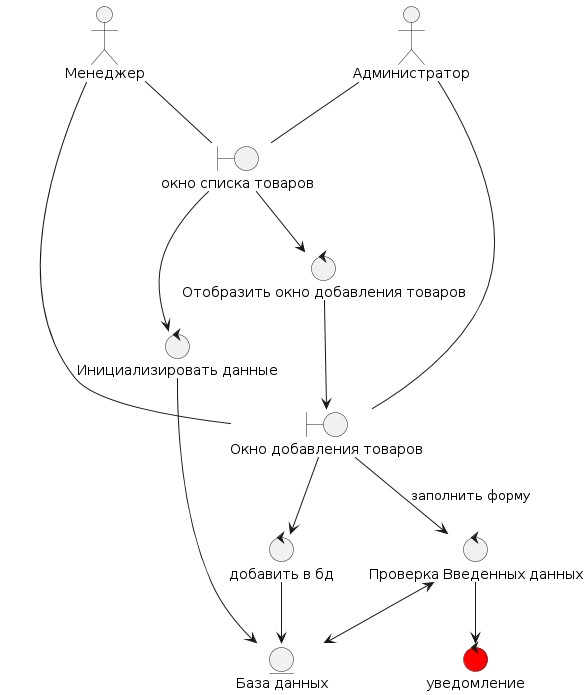


Рисунок 33 – Диаграмма пригодности прецедента «Добавить товар»

На рисунке 34 представлена диаграмма последовательности для данного прецедента.

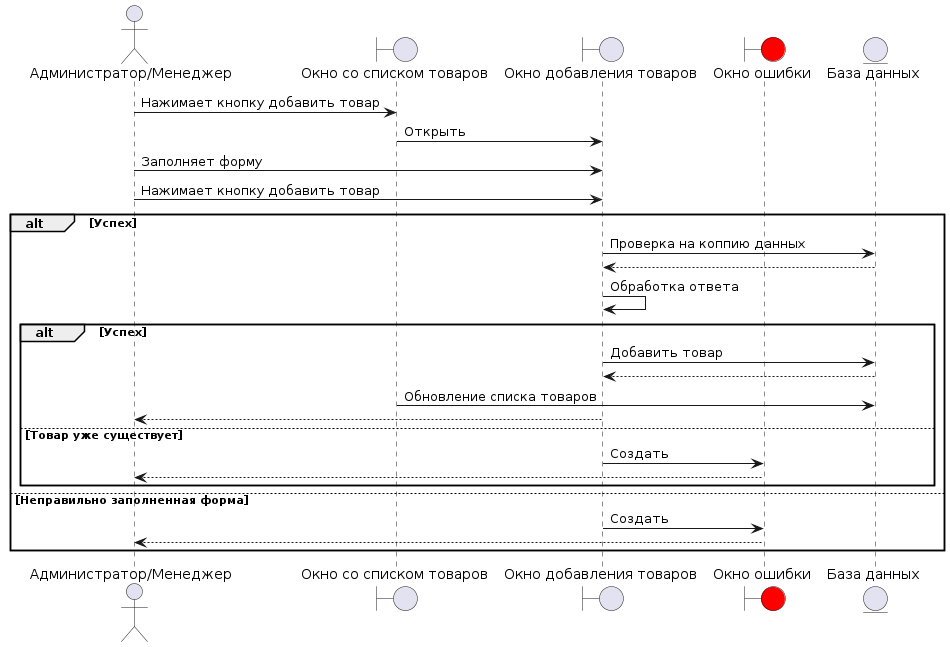


Рисунок 34 – Диаграмма последовательности прецедента «Добавить товар»

2.2.6 Прецедент «Добавить пользователя»

На рисунке 35 представлена диаграмма пригодности для данного прецедента.

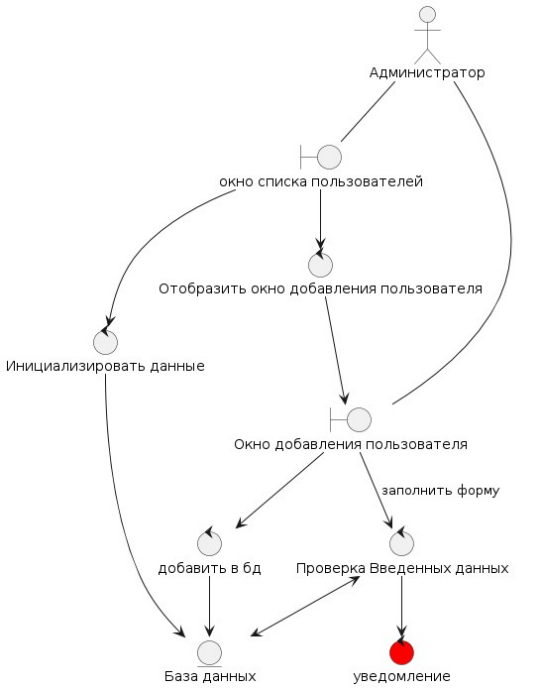


Рисунок 35 – Диаграмма пригодности прецедента «Добавить пользователя»

На рисунке 36 представлена диаграмма последовательности для данного прецедента.

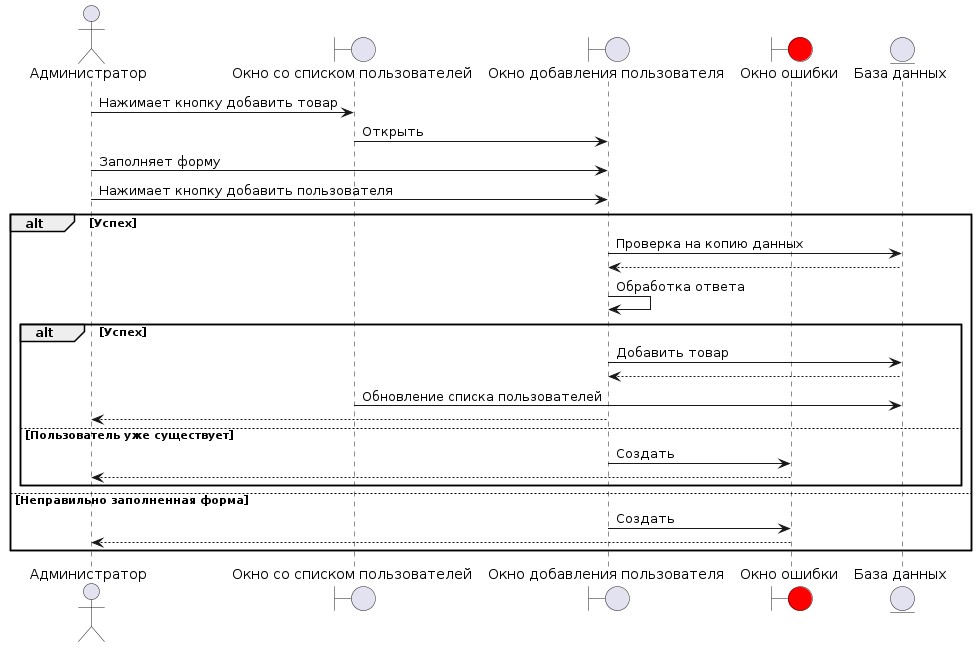


Рисунок 36 ­­– Диаграмма последовательности прецедента «Добавить пользователя»

2.2 ER-диаграмма

На рисунке 37 представлена ER-диаграмма в нотации Мартина.



Рисунок 37 – ER-диаграмма

2.3 Диаграмма классов

На рисунке 38-40 представлены диаграммы классов пользователей.

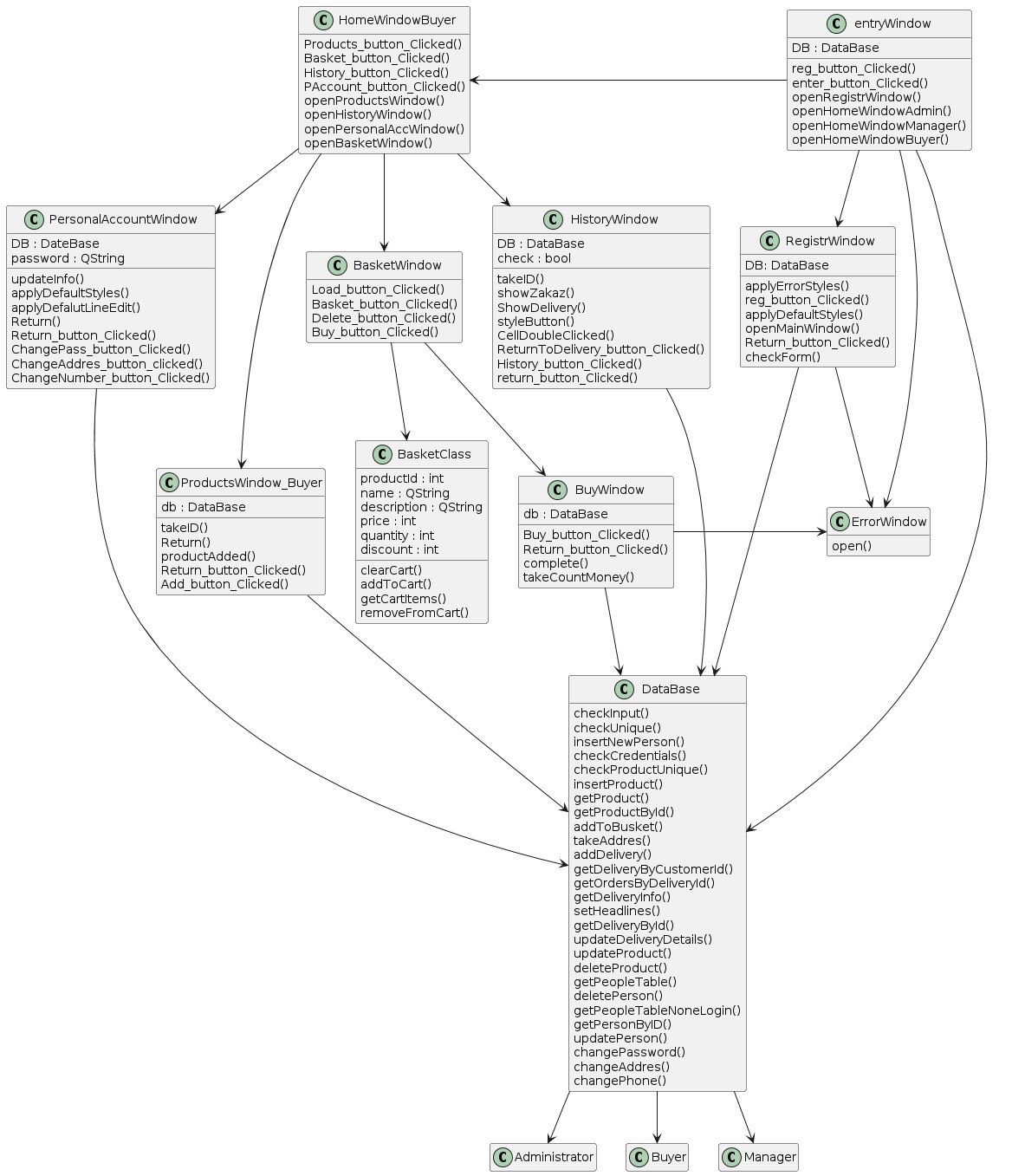


Рисунок 38 – Диаграмма классов покупателя

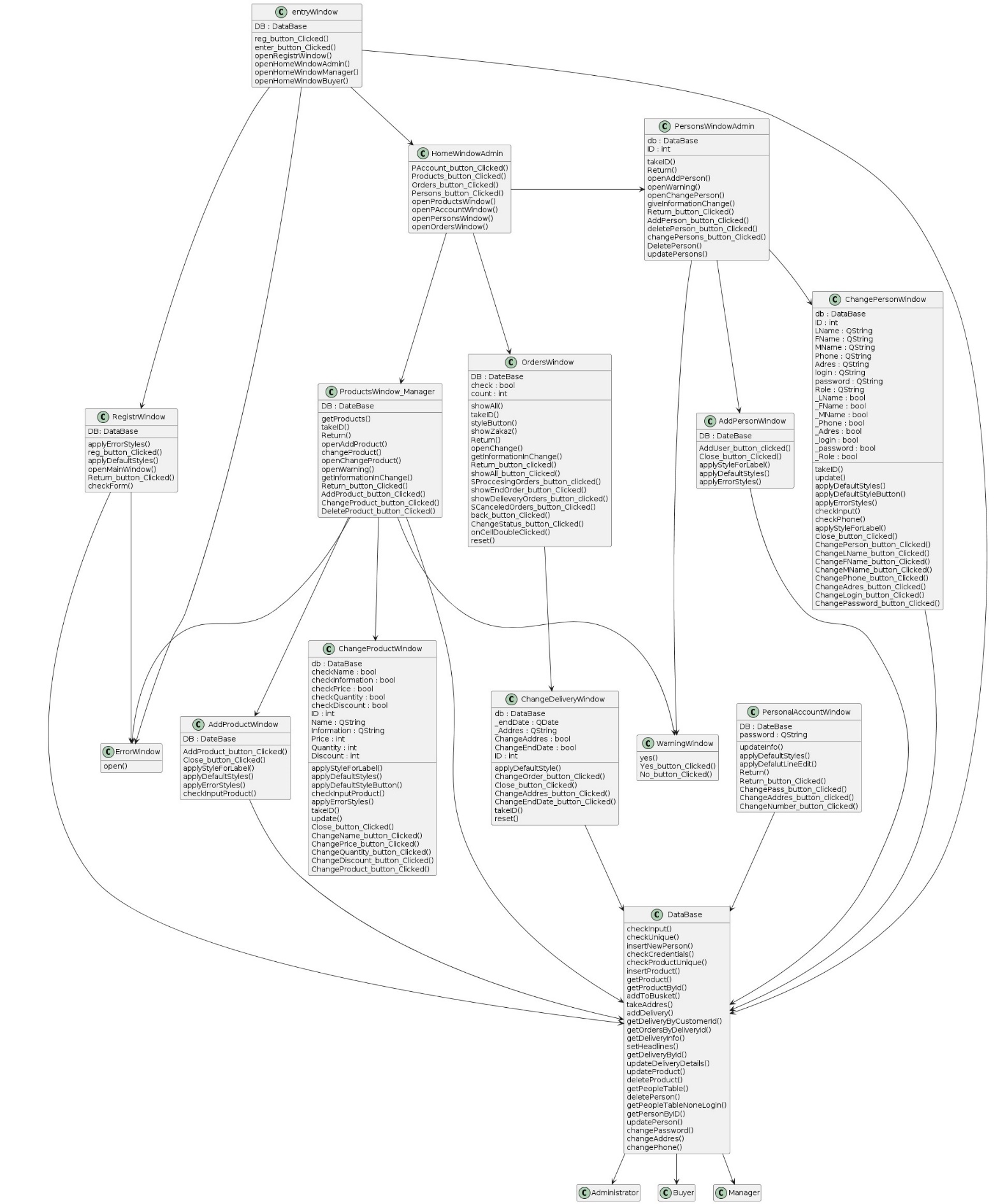


Рисунок 39 – Диаграмма классов администратора

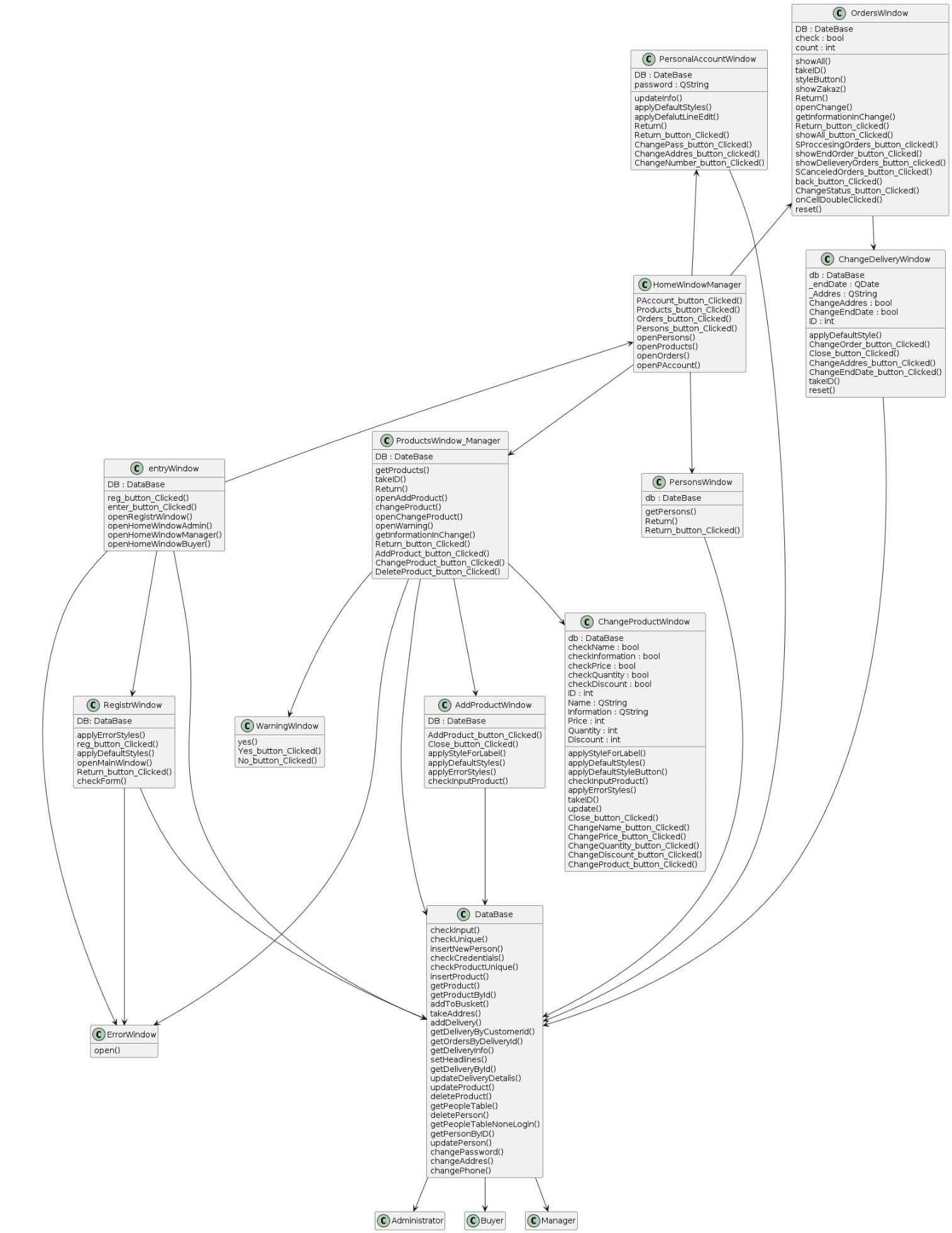


Рисунок 40 – Диаграмма классов менеджера

# 3 **Объектно-ориентированное программирование**

* 1. Выбор инструментов

Для разработки ПК-приложения необходимы библиотеки для создания графического интерфейса, работы с базами данных. Для этих целей отлично подходит язык программирования C++ вместе с фреймворком Qt. Qt предоставляет широкий выбор инструментов для графического интерфейса, включая стандартные Qt Widgets, и содержит всё необходимое для работы с моделями, списками и представлениями.

* 1. Реализация

3.2.1 Графический интерфейс

Для разработки графического интерфейса приложения на ПК удобно использовать стандартный редактор дизайна в Qt. Этот инструмент позволяет визуально создавать интерфейс с помощью Qt Widgets, предлагая множество предустановленных элементов управления. Редактор позволяет легко размещать кнопки, текстовые поля, списки и другие виджеты, обеспечивая удобное создание и редактирование пользовательских интерфейсов.

Приложение содержит множество окон, поэтому необходимо реализовать переходы между ними. В соответствии с диаграммами в приложении А, логика переключения экранов осуществляется через сигналы и слоты. В Qt сигналы и слоты обеспечивают механизм связи между объектами. Когда пользователь взаимодействует с элементом интерфейса, сигнал, например, buttonClicked(), может быть отправлен и пойман соответствующим слотом, который выполнит переключение окна. Этот подход обеспечивает гибкую и модульную архитектуру, упрощая управление переходами между экранами.

3.2.2 База данных

Различные данные будут храниться в базе данных SQLite. Для работы с базой данных используется класс DataBase. Взаимодействие с базой данных реализуются средствами модуля QT SQL.

Класс DataBase необходим для инициализации соединения с базой данных и поддержания этого соединения на протяжении всего времени работы приложения. В конструкторе класса DataBase проверяется наличие файла базы данных и, если он отсутствует, создается новая база данных с необходимыми таблицами.

3.3 Сборка и запуск

Все действия выполняются в консоли и в директории, в которой будет храниться исходный код программы.

1) <https://github.com/linetick/Kursach>

2) В QT Creator выбирается сборка программы.

Если собрать программу автоматическом режиме не получается:

1) <https://github.com/linetick/Kursach>

2) “cd Kurs-master”

3) “qmake Kursach.pro”

4) “make “

5) “./Kurs”

3.4 Первый запуск программы

При первом запуске, программа автоматически создаёт базу данных и заполняет её необходимыми таблицами, так же в таблицу люди добавляется администратор с данными: «Имя – Админ», «фамилия – Администратор», «отчество – Отчество», «телефон – 1234567890», «адрес – Адрес», «логин – админ», «пароль – админ», «роль – Администратор» .

3.5 Тестирование

Тестирование проводилось вручную с целью убедиться, что все функциональные окна системы работают корректно, валидаторы и механизмы обработки данных функционируют правильно, а система обрабатывает все возможные сценарии ввода данных. В этом разделе подробно описывается процесс ручного тестирования, включающего проверку окон регистрации, входа в систему, добавления и изменения товаров, добавления и изменения пользователей, а также изменения доставки.

Ручное тестирование включало поэтапную проверку каждого необходимого для тестирования окна на соответствие требованиям. В качестве примера ниже приведены действия, которые выполнялись при проверке прецедента «Вход в систему»:

- попытка входа с пустыми полями

- попытка входа с отличными от зарегистрированных учетными данными

- попытка входа с правильным логином и неправильным паролем

- попытка входа с правильным паролем и неправильным логином

Пример сообщения об ошибке приведён на рисунке 41.

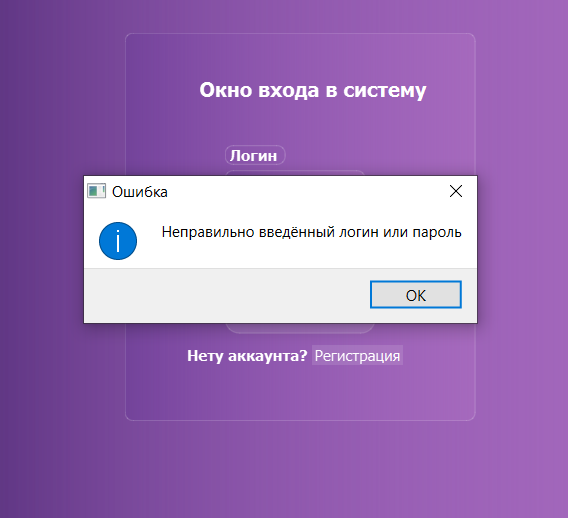


Рисунок 41 – тестирование некорректных данных окна входа в систему

Для прецедента «Регистрация» было проведено тестирование с помощью следующих действий:

- проверка полей «Телефон» и «Логин» на уникальность (рисунок 44)

- попытка регистрации с пустыми полями (рисунок 45)

- попытка регистрации с использованием допустимых, но граничных значений (например, минимальная и максимальная длина пароля)

- проверка корректности обработки ошибок и отображения сообщений пользователю.

Пример сообщения об успешной регистрации приведён на рисунке 43.

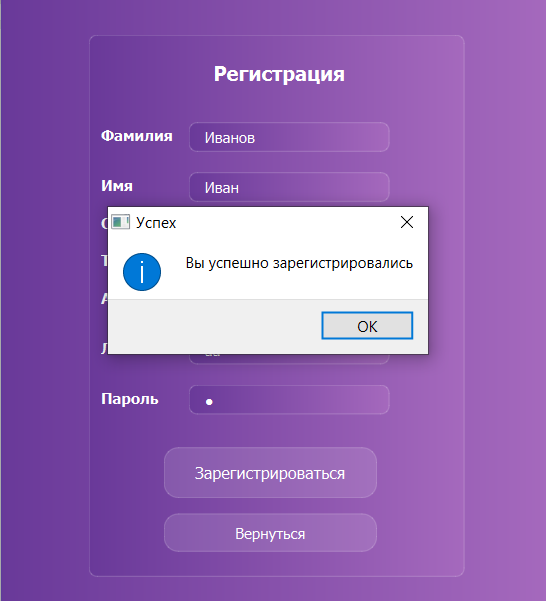


Рисунок 43 – тестирование корректных данных

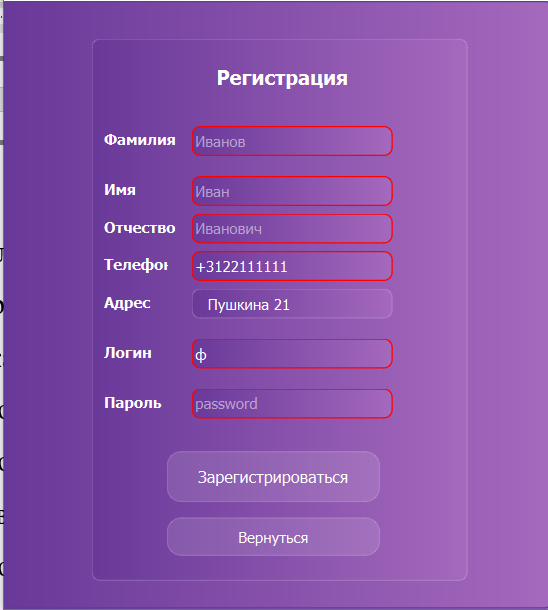


Рисунок 44 – Тестирование полей «Телефон» и «Логин» на уникальность.

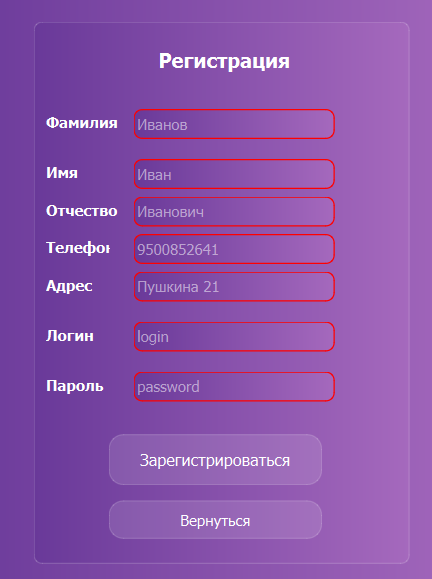


Рисунок 45 – Тестирование всех полей на пустые строки

Для прецедента «Добавить товар» было проведено тестирование с помощью следующих действий:

- попытка добавления товара с уже существующим названием (рисунок 47)

- попытка регистрации с пустыми полями (рисунок 48)

- попытка добавления товара с некорректными значениями в полях:

- Некорректная цена (например, отрицательное значение или буквы).

- Некорректное количество (например, отрицательное значение или буквы).

- Некорректная скидка (например, отрицательное значение, буквы, значение больше 100%).

Пример сообщения об успешном добавлении товара приведён на рисунке 46.

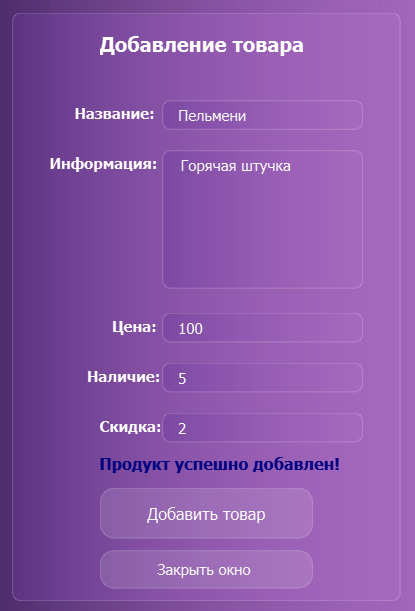


Рисунок 46 – Тестирование корректно заполненной формы

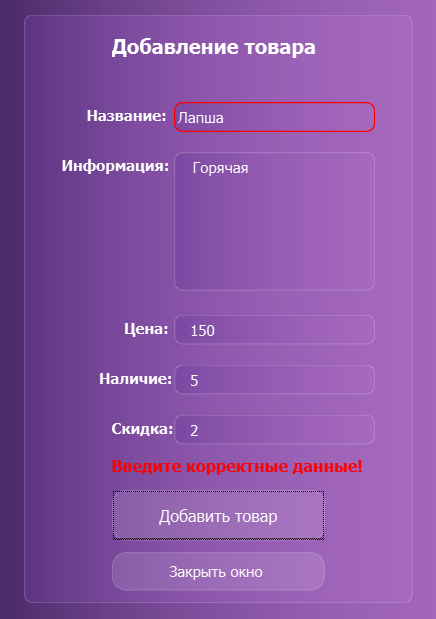


Рисунок 47 – Тестирование поле «Название» на уникальность

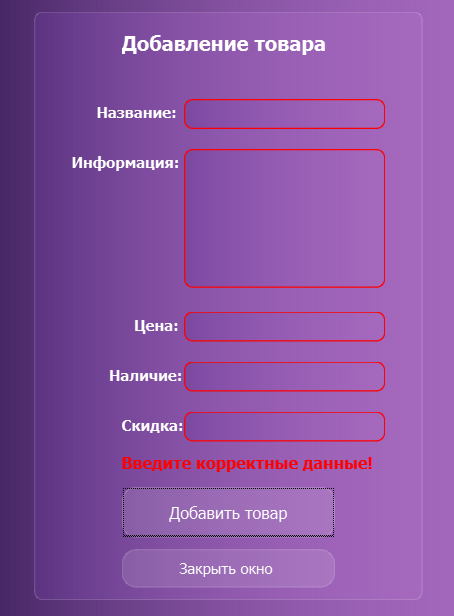


Рисунок 48 – Тестирование полей на пустые строки

Тестирование проводилось для каждого прецедента, описанного в первом разделе.

В результате ручного тестирования всех функциональных окон системы были выявлены и исправлены следующие проблемы:

1. Некорректная обработка спецсимволов в полях ввода.

2. Неверные сообщения об ошибках при вводе некорректных данных.

3. Ошибки при проверке совпадения паролей и других полей.

После внесения изменений система была повторно протестирована, и все тесты прошли успешно. Валидаторы корректно обрабатывают все виды ввода данных, система отображает правильные сообщения об ошибках, и пользователи могут успешно выполнять все операции (регистрация, вход, добавление и изменение товаров и пользователей, изменение доставки) при корректном заполнении всех полей.

3.6 Инструкция

В качестве инструкций для пользователя можно использовать прецеденты, описанные в первом разделе и диаграммы последовательности, разработанные во втором разделе.

Заключение

В результате выполнения курсового проекта была разработана спецификация требования, и по ней разработано приложение «Оптово-розничная продажа товаров».

# Приложение А

**Диаграмма потоков экранов**

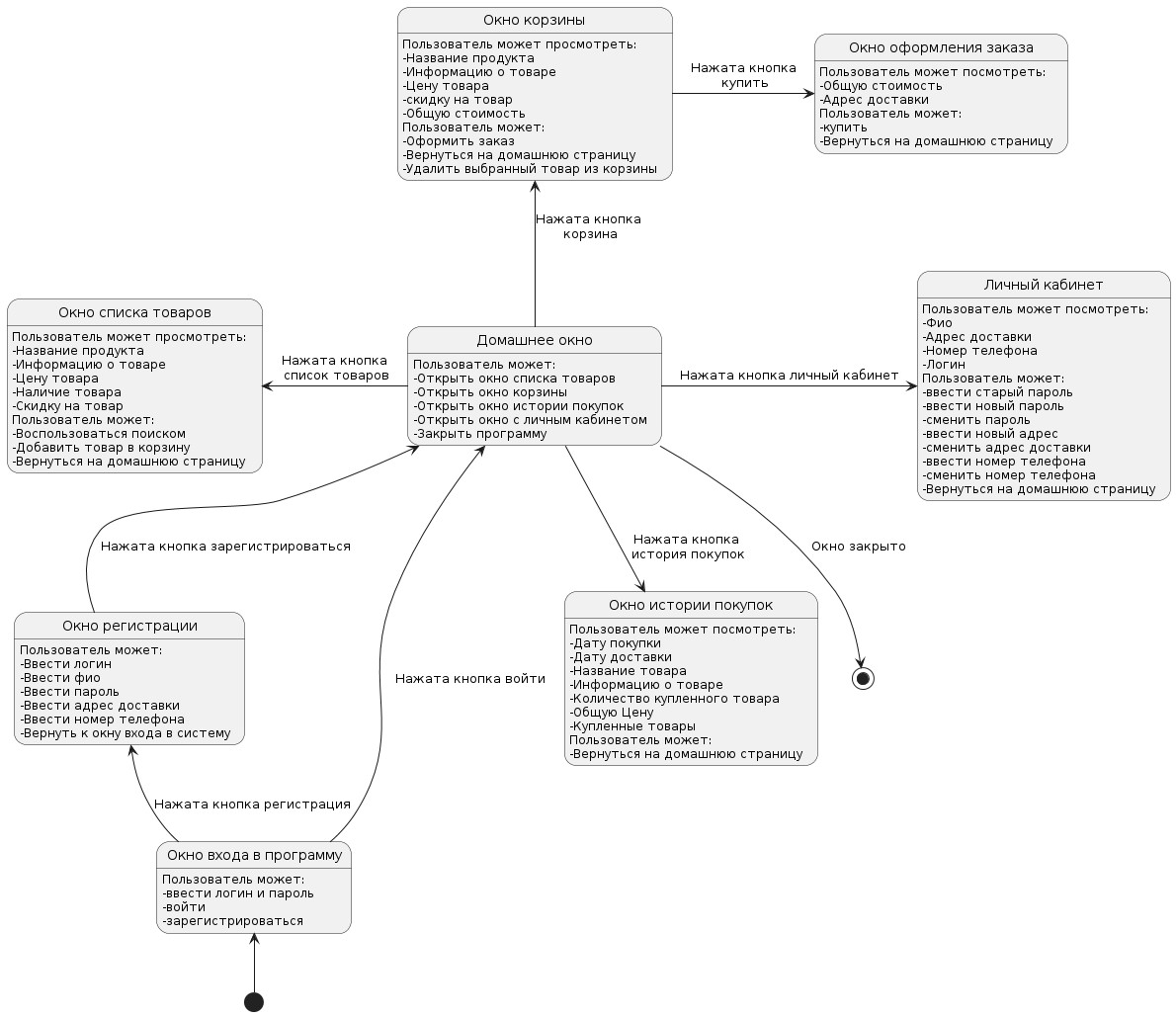


Рисунок А.1 – Диаграмма потоков экранов покупателя

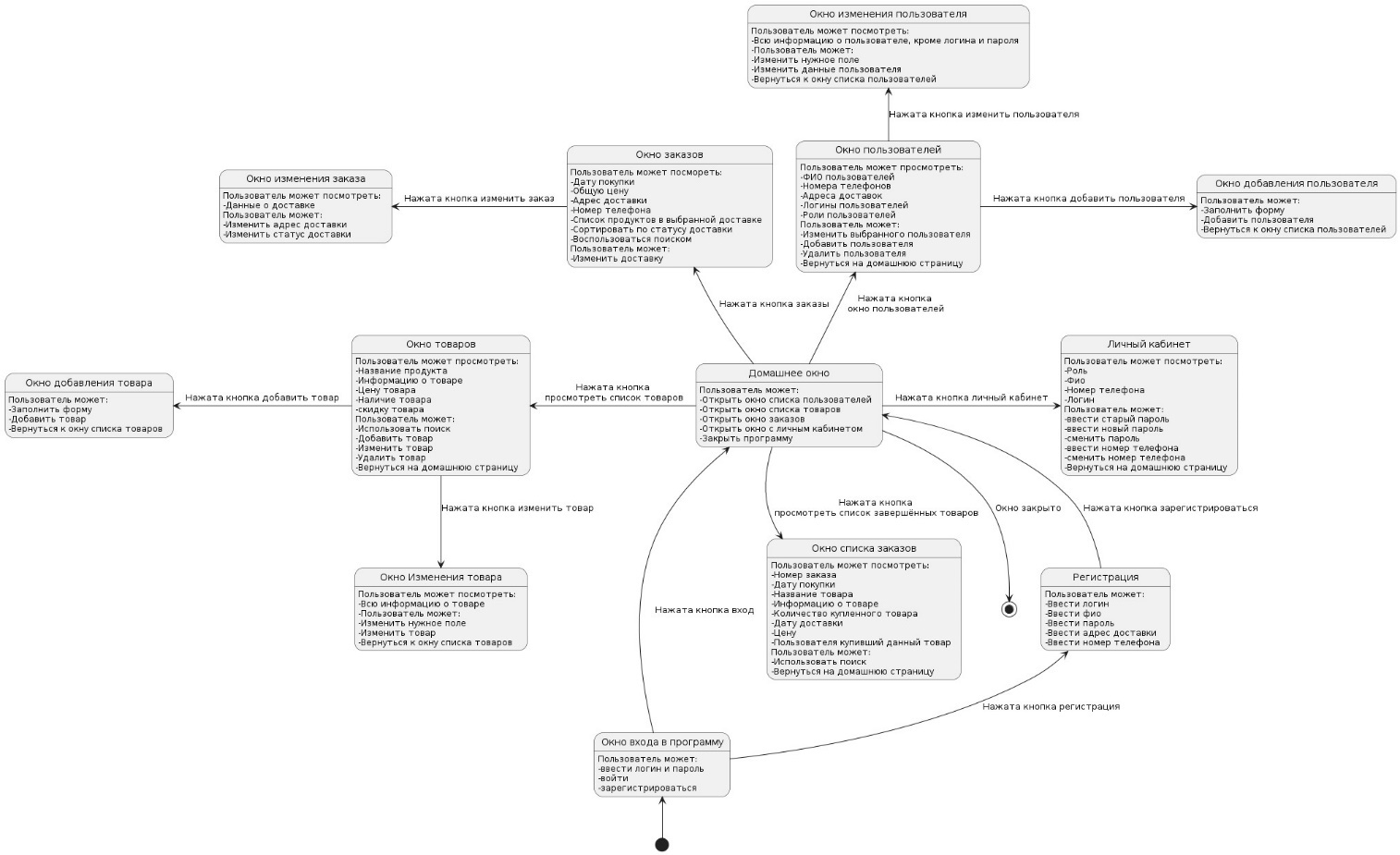


Рисунок А.2 – Диаграмма потоков экранов администратора

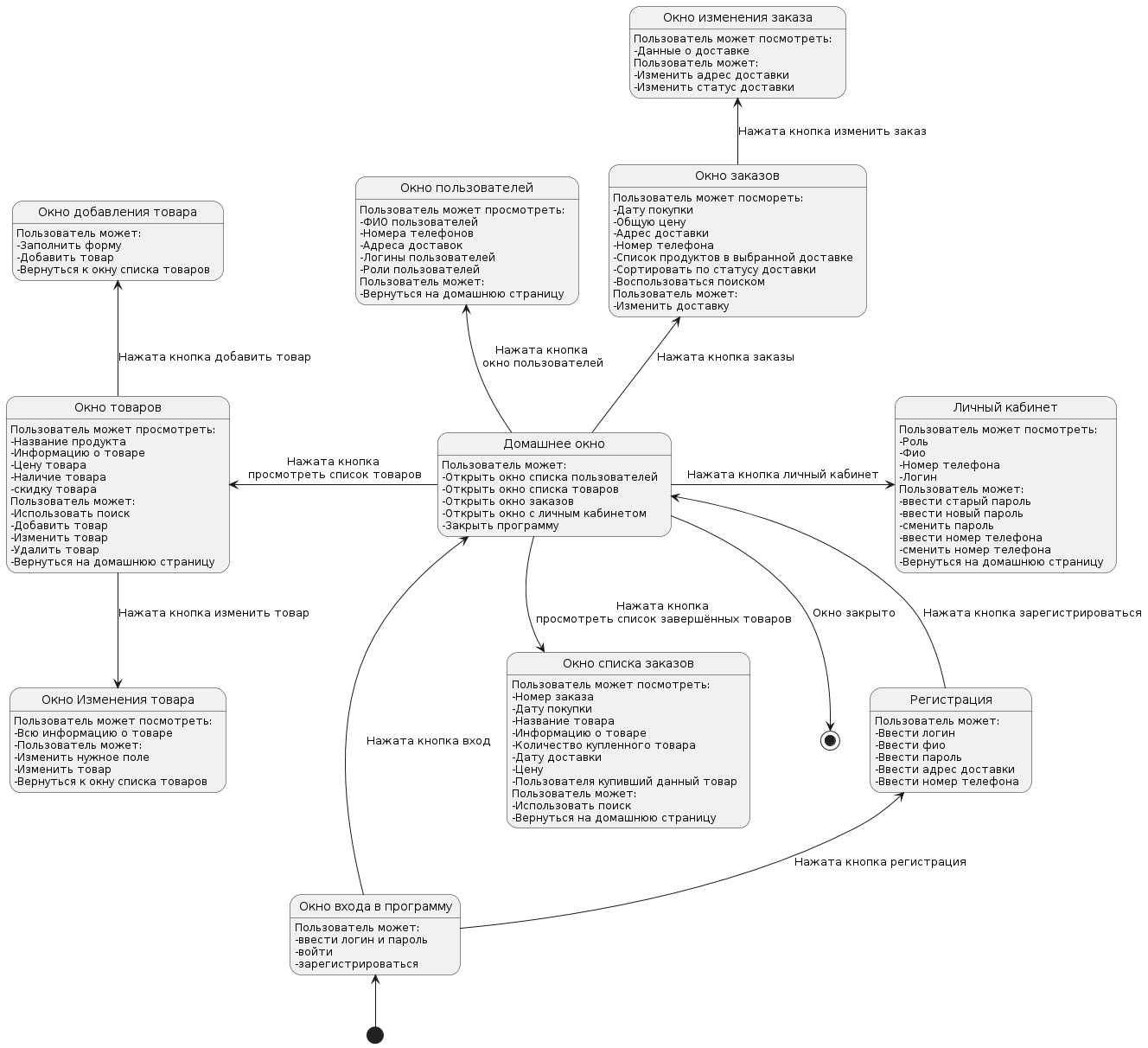


Рисунок А.3 – Диаграмма потоков экрана менеджера

# Список используемых источников

1. СТУ 7.5–07–2021. Стандарт университета «Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности».
2. Документация Qt // QT | Tools for Each Stage of Software Development Lifecycle. – URL: <https://doc.qt.io/> (дата обращения 24.02.2024).
3. Основы UML – диаграммы использования (use-case) // Блог программиста – программирование и алгоритмы URL: <https://pro-prof.com/>.
4. SQLite Documentation : [сайт]. – URL: https://www.sqlite.org/docs.html (дата обращения: 20.05.2024)
5. PlantUML : [сайт]. – URL: https://plantuml.com/ru/ (дата обращения: 20.05.2024).