Министерство науки и высшего образования Российской Федерации

Бабикова А.С.

ИСТ-19-2б

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный** **исследовательский политехнический университет»**

Факультет: Прикладной математики и механики

Кафедра: Вычислительной математики, механики и биомеханики

Направление: 09.03.02 «Информационные системы и технологии»

Профиль бакалавриата: «Информационные системы и технологии»

**ПРАКТИЧЕСКАЯ РАБОТА**

по дисциплине

**«КОРПОРАТИВНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ»**

Тема: **«Программная реализация ИС»**

Выполнил:

студент гр.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Ф.И.О.)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*(подпись)*

Принял:

ст. преподаватель, Банников Р.Ю.

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(должность, ФИО руководителя)*

*\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(оценка) (подпись)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

**Пермь 2023**

Постановка задачи

Назначение системы

### Корпоративная информационная система, осуществляющая функции редактирования, создания, чтения, удаления и обновления товаров предназначена для рекламы и в последующем заказов товаров определенной категории.

### Состав системы

### В состав системы должны входить следующие компоненты:

### Подсистема управления. Подсистема, предназначенная для работы Администратора системы с данными БД, выполнения настроек.

* Подсистема сбора и обработки данных. Подсистема, предназначенная для регистрации заявок.
* Подсистема защиты информации. Подсистема представляет собой программно-технический комплекс, предназначенный для защиты технических средств, программного обеспечения и данных от несанкционированного доступа к данным системы. Выполняет функции по идентификации и аутентификации сторон, функции по разграничению прав доступа к информационным ресурсам.
* База данных системы. Подсистема, предназначена для хранения и управления данными
* Подсистема отчетности. Подсистема, предназначенная для формирования отчета, который содержит данные, генерируемые в процессе функционирования системы, сохранение отчета в форматах DOCX.

Реализация

*Средства разработки*

Разработка велась в среде Visual Studio на языке C# с использованием Windows Form.

База данных реализована в СУБД MySQL Workbench.

*Функционал*

### Задача авторизации сверить введенные данные пользователя с данными в базе и открыть в зависимости от этого личный кабинет. Если данные введены некорректно, то открывается окно с ошибкой. Рис. 1.

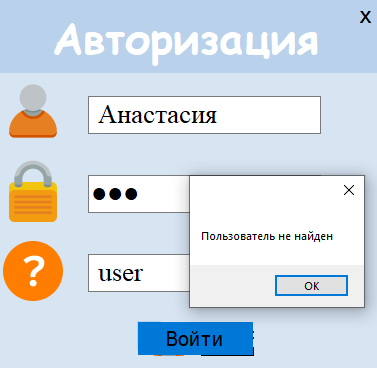


Рисунок 1 – Авторизация.

При успешной авторизации как администратор пользователь видит следующее окно, представленное на рис.2.

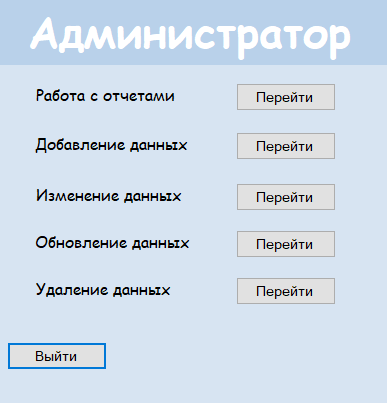


Рисунок 2 – Личный кабинет администратора.

У администратора имеется пять функций такие как:

* Работа с отчетами;
* Добавление данных;
* Изменение данных;
* Обновление данных;
* Удаление данных.

Рассмотрим подробнее каждую.

1. Работа с отчетами представлена на рис. 3.

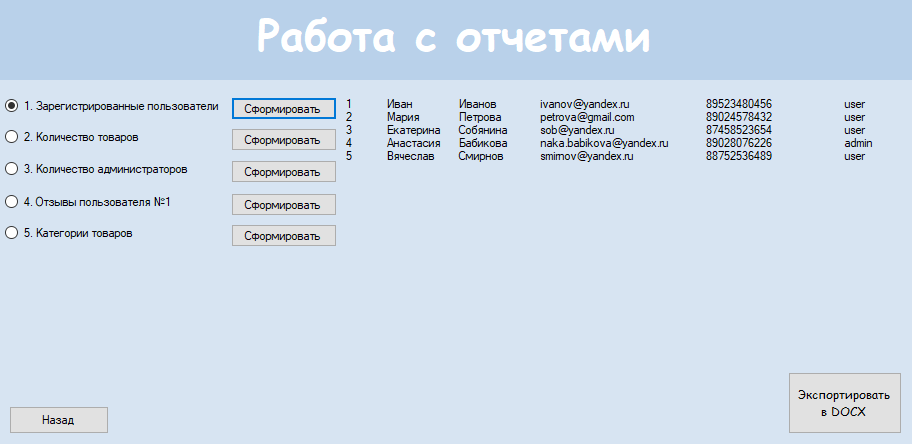


Рисунок 3 – Работа с отчетами.

При нажатии на кнопку «Сформировать», к базе данных отправляется запрос, например чтобы получить зарегистрированных пользователей запрос выглядит следующим образом:

MySqlCommand command = new MySqlCommand("select \* from users", db.getConnection());

### Где первая часть отвечает за сам запрос, а вторая за подключение к базе. Аналогичным образом реализованы остальные запросы.

Для того чтобы экспортировать данный отчет в DOCX необходимо нажать слева от отчета кнопку (radioButton), а затем кнопку «Экспортировать в DOCX». При успешном экспорте выводится всплывающее окно как на рис. 4.

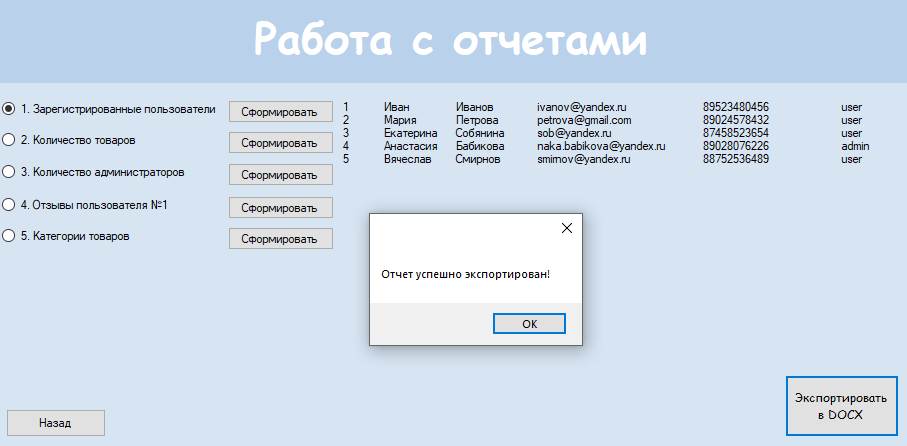


Рисунок 4 – Экспорт отчета.

Отчет в формате DOCX не удалось реализовать в заявленном формате. Вариант отчета представлен на рис.5.

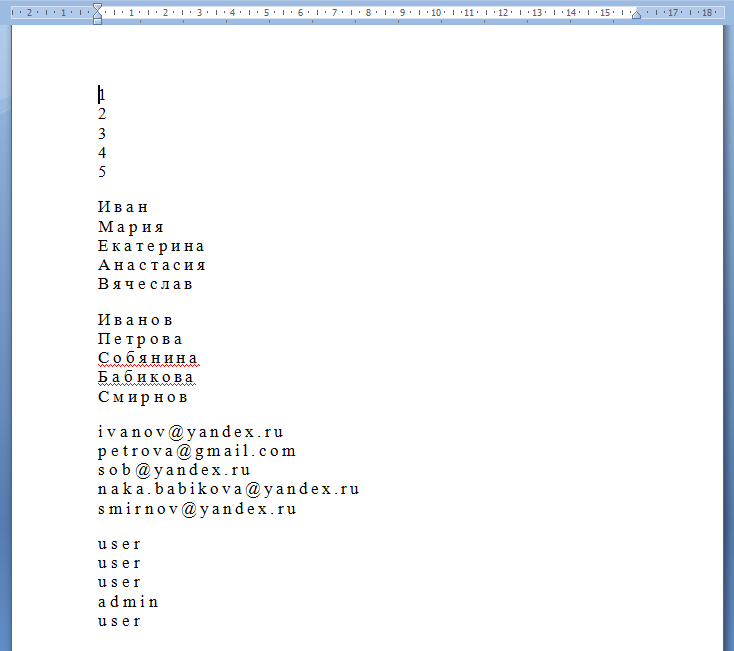


Рисунок 5 – Отчет в формате DOCX.

2. Добавление данных представлено на рис. 6.

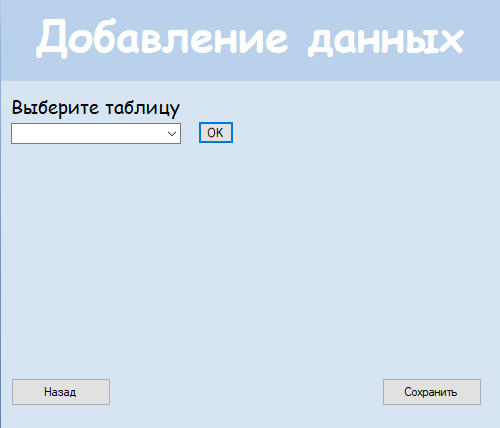


Рисунок 6 – Добавление данных.

Для того чтобы добавить данные для начала необходимо выбрать таблицу и нажать кнопку «ОК».

Если таблица была не выбрана и нажата кнопка «ОК», то выводится всплывающее окно представленное на рис.7.

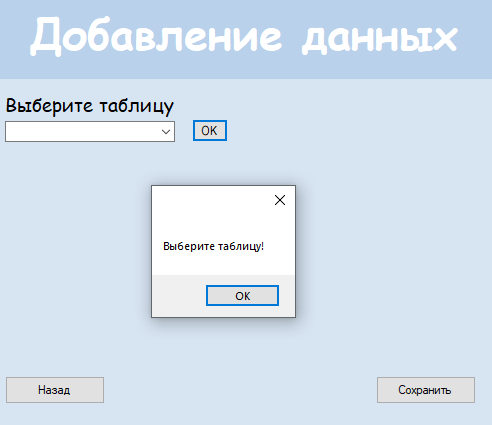


Рисунок 7 – Всплывающее окно с выбором таблицы.

При успешном выборе таблицы для каждой таблицы выводятся соответствующие поля, которые необходимо заполнить. Рис. 8.

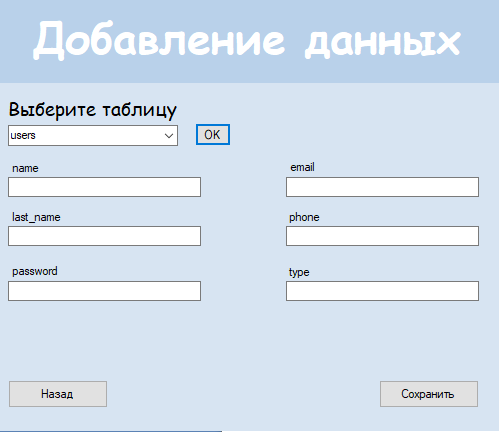


Рисунок 8 – Добавление данных в таблицу «users».

Для некоторых полей реализована проверка на пустоту, если пользователь не заполнил поле и нажал кнопку «Сохранить», то всплывает окно с предупреждением. Рис.9.

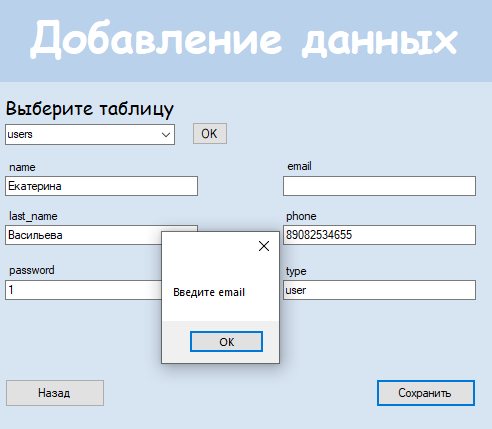


Рисунок 9 – Окно с предупреждением.

При успешном заполнение всех полей после нажатия на кнопку «Сохранить» получаем следующее сообщение. Рис. 10.

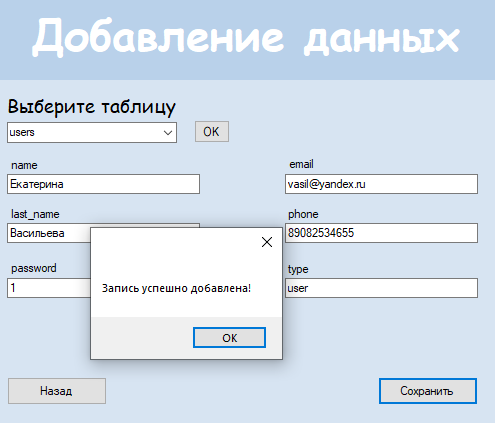


Рисунок 10 – Успешное добавление в базу.

3. Изменение данных представлено на рис. 11.

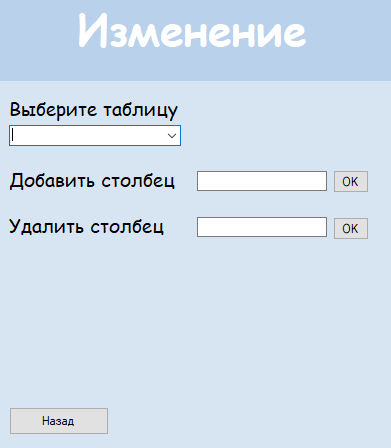


Рисунок 11 – Изменение данных.

4. Обновление данных представлено на рис.12.

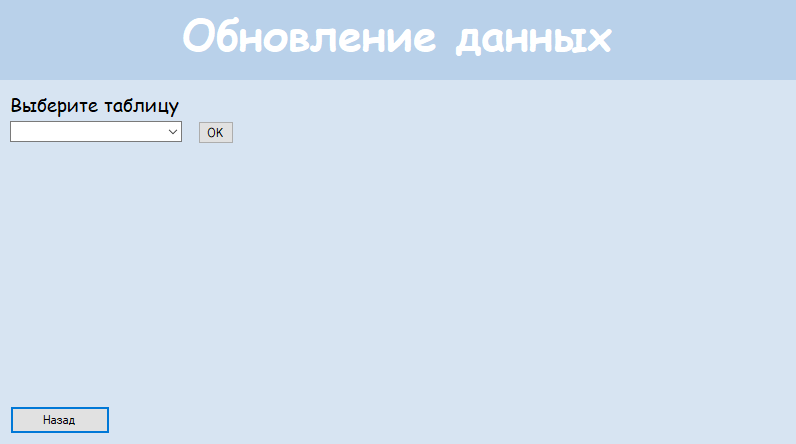


Рисунок 12 – Обновление данных.

Для обновления необходимо выбрать таблицу и нажать кнопку «ОК». Для каждой таблицы обновляемые поля отличаются. На рис. 13 обновление выполнялось для таблицы «orders».

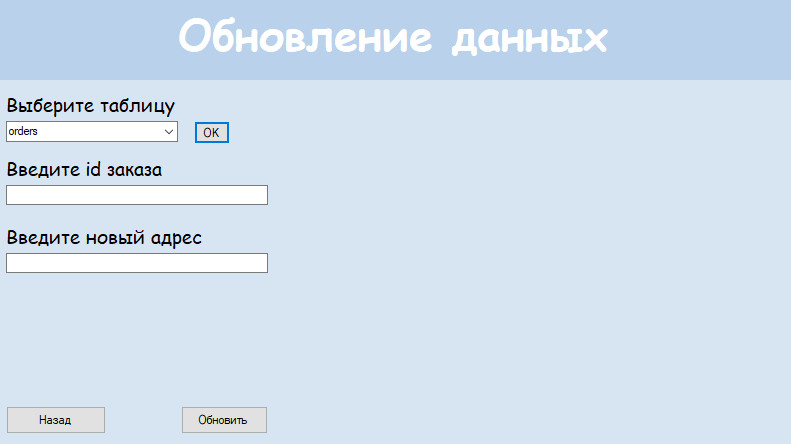


Рисунок 13 – Обновление таблицы «orders».

До обновления таблица в базе выглядела следующим образом. Рис. 14.

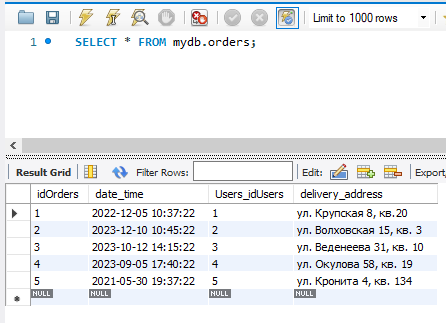


Рисунок 14 – Таблица «orders».

Заполнили поля и нажали кнопку «Обновить». К базе отправился запрос на обновление

MySqlCommand command = new MySqlCommand("UPDATE `orders` SET `delivery\_address`=@adress WHERE `idOrders`=@id", db.getConnection()); Рис.15.

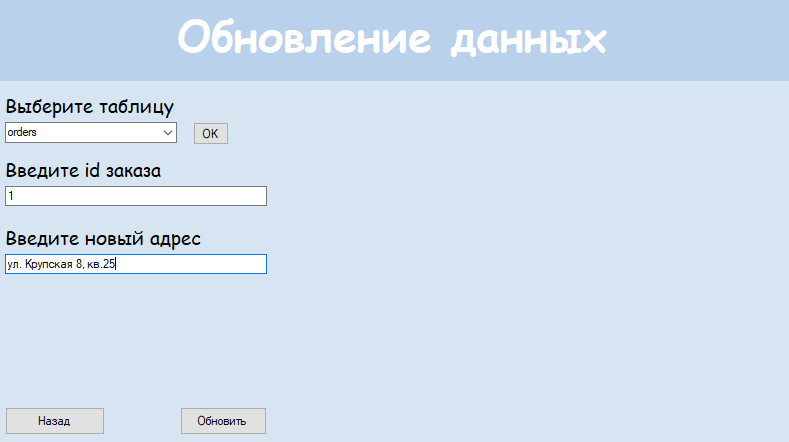


Рисунок 15 – Заполненные поля для обновления.

После успешного обновления было получено всплывающее окно представленное на рис. 16.

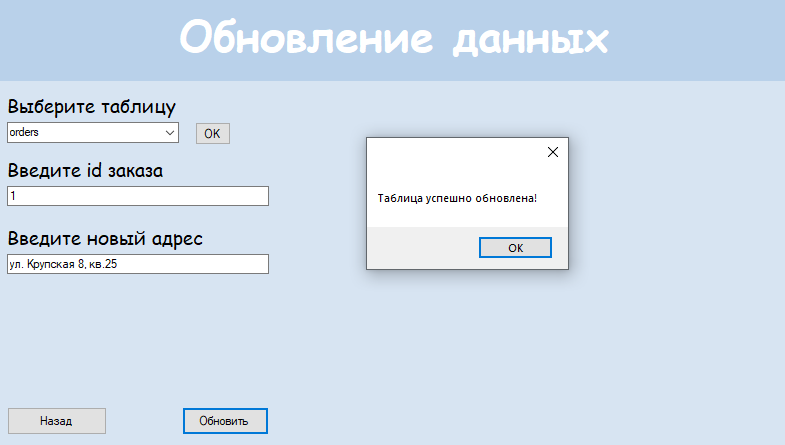


Рисунок 16 – Успешное обновление.

После этого база выглядит следующим образом. Рис. 17.

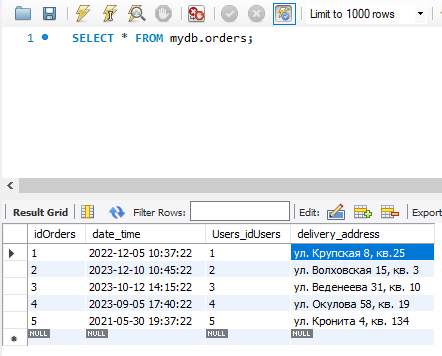


Рисунок 17 – Обновленная таблица «orders»

5. Удаление данных представлено на рис. 18.

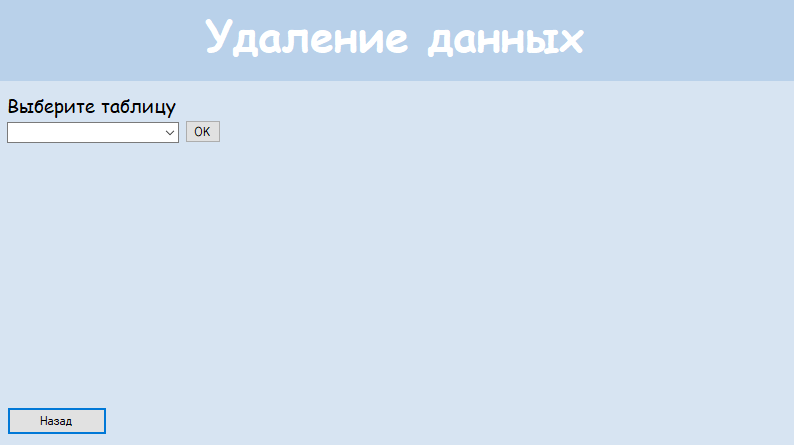


Рисунок 18 – Удаление данных.

Для того чтобы удалить данные необходимо выбрать таблицу и нажать кнопку «ОК». После этого появляется поле, пример представлен на рис.19.

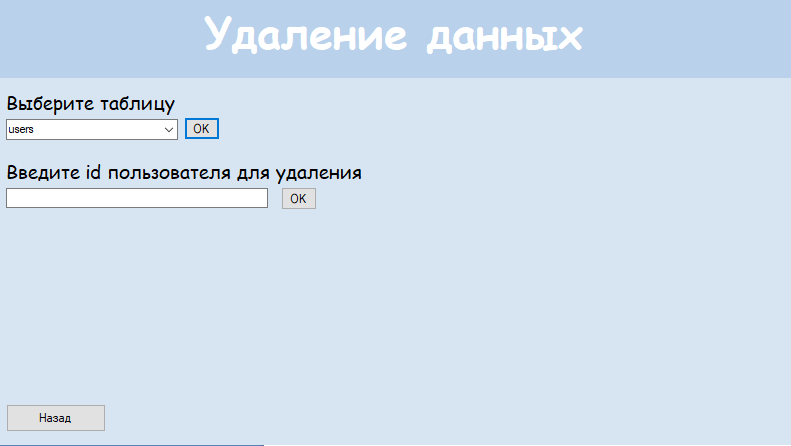


Рисунок 19 – Удаление записи по id.

При успешном удалении всплывает окно, представленное на рис.20.

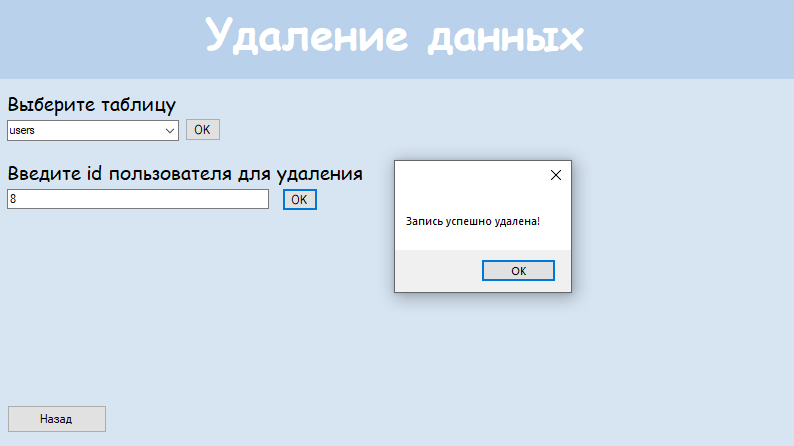


Рисунок 20 – Успешное удаление записи.

При успешной авторизации как пользователь отображается следующее окно, представленное на рис.21.

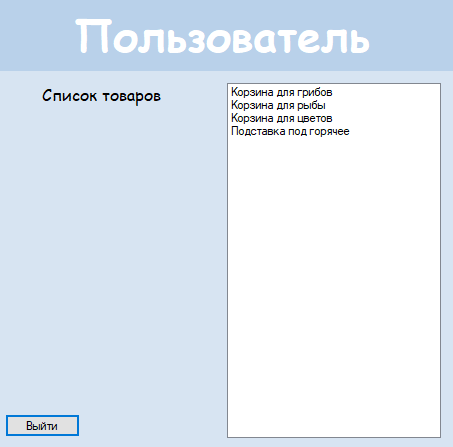


Рисунок 21 – Личный кабинет пользователя.

Пользователь имеет возможность просматривать товары. При нажатии на кнопку «Выйти» пользователь попадает на окно с авторизацией.

Выводы

### В ходе данной лабораторной работы была реализована КИС на языке C#. А именно личные кабинеты пользователя и администратора, реализован отдельный класс для взаимодействия с базой данных (обновление, удаление, добавление, изменение данных), ответы в виде всплывающих окон на различные действия пользователей, формирование отчетов.

На данном этапе обновление реализовано у одной таблицы, некорректно выводятся отчеты в формат DOCX.

На следующем этапе планируется провести ручное тестирование для проверки заявленного функционала в техническом задании.

РиРРРисунро