МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение

высшего образования

**«Северный (Арктический) федеральный университет имени М.В. Ломоносова»**

**Высшая школа информационных технологий и автоматизированных систем**

(наименование высшей школы)

**РАСЧЕТНО-ГРАФИЧЕСКАЯ РАБОТА**

|  |  |  |
| --- | --- | --- |
| По дисциплине | | Технологии программирования |
|  | | |
|  | | |
| На тему | Разработка программной подсистемы (сервиса) в соответствии с вариантом | |
|  | | |

|  |  |
| --- | --- |
|  | Выполнил обучающийся:  Кульба Георгий Васильевич |
|  | (Ф.И.О.) |
|  | Направление подготовки:  09.03.01 Системное и сетевое администрирование |
|  | (код и наименование) |
|  | Курс: 2 |
|  | Группа: 351919 |
|  | Руководитель:  Веретнов М.Ю. |
|  | (Ф.И.О. руководителя, должность / уч. степень / звание) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Отметка о зачете |  |  |  |  |
|  |  | (отметка прописью) |  | (дата) |
| Руководитель |  |  |  | М. Ю. Веретнов |
|  |  | (подпись руководителя) |  | (инициалы, фамилия) |

Архангельск 2021

Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение

высшего образования

**«Северный (Арктический) федеральный университет имени М. В. Ломоносова**»

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | **Высшая школа информационных технологий и автоматизированных систем** | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | (наименование кафедры) | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | **ЗАДАНИЕ НА РАСЧЕТНО-ГРАФИЧЕСКУЮ РАБОТУ** | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | по | | | | Технологиям программирования | | | | | | | | | | | | | | | | | | | | | |  |
|  |  | | | | | | | | | | (наименование дисциплины) | | | | | | | | | | | | | | | |  |
|  | студенту | | | | | | 2 | | |  | | | | 351919 | | | | группы | |  |
|  | Кульба Георгий Васильевич | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | (фамилия, имя, отчество студента) | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | |  | | --- | | 09.03.01 Информатика и вычислительная техника | | (код и наименование направления подготовки/специальность) | | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  |  | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | ТЕМА: | | | | | Разработка программной подсистемы (сервиса) в соответствии с вариантом | | | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | | | | |
|  |  | | |
|  |  | | |
|  | ИСХОДНЫЕ ДАННЫЕ: | | | | | | | Разработать подсистему для работы с товарами магазина мебели, которая должна включать в себя следующий функционал:  • просмотр списка товаров,  • добавление/удаление/редактирование данных о товарах,  • предложения дополнительных товаров к основным,  • просмотр истории покупок товаров магазина | | | | | | | | | | | | | | | | | | |  |
|  |  | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  |  | | |
|  |  | | |
|  | Руководитель проекта | | | | | | | | | | | |  | |  |  | | | | | |  | Веретнов М.Ю. | | | |  |
|  |  | | | | | | | | | |  | | (должность) | |  | (подпись) | | | | | |  | (инициалы,  фамилия) | | | |  |
|  |  | | | | | | | | | |  | |  | |  |  | | | | | |  |  | | | |  |
|  |  | | | | | | | | | |  | |  | |  |  | | | | | |  |  | | | |  |
|  | |  |  | | | | | |  | | |  | | | | |  | |
| Архангельск 2021 | | | | | | | | | | | | | | | | | | | | | | | | |  |

ЛИСТ ДЛЯ ЗАМЕЧАНИЙ

ОГЛАВЛЕНИЕ

[Введение 6](#_Toc76042551)

[1 Определение требований 7](#_Toc76042552)

[1.1 Описание предметной области 7](#_Toc76042553)

[1.3 Определение требований к стилю 7](#_Toc76042554)

[1.4 Определение требований к GUI 8](#_Toc76042555)

[1.4.1 Список товаров 8](#_Toc76042556)

[1.4.2 Добавление/редактирование товара 8](#_Toc76042557)

[2 Проектирование и реализация база данных 10](#_Toc76042558)

[3 Реализация подсистемы 17](#_Toc76042559)

[4 Проектирование графического пользовательского интерфейса 20](#_Toc76042560)

[3.1 Авторизации 20](#_Toc76042561)

[3.2 Добавление товара 22](#_Toc76042562)

[Заключение 24](#_Toc76042563)

[Список используемых источников 25](#_Toc76042564)

НОРМАТИВНЫЕ ССЫЛКИ

В текстовом документе использованы ссылки на следующие нормативные документы:

СТО 60-02.2.3-2018. Общие требования к оформлению и изложению документов учебной деятельности обучающихся.

ВВЕДЕНИЕ

Для выполнения расчетно-графической работы, первым делом стоит создать базу данных, а также создать приложение для работы с этой базой данных, в которой хранится информация о мебели в пригодном виде.

Чтобы выполнить расчетно-графическую работу, нужно:

* создать базу данных;
* выполнить настройку и заполнить базу данных;
* проверить корректность заполненных данных;
* написать приложение на языке C# для подключения к базе данных, использования необходимых функций;

Необходимые требования к базе данных:

* наличие не менее 3-х сущностей;
* раздельное хранение учетных данных клиентов и сотрудников.

Требования к приложению:

* платформа пользовательского интерфейса WPF;
* разграничение доступа по занимаемой должности;
* наличие карточки товара;
* функционал для просмотра, добавления, изменения данных товара;
* возможность добавления рекомендуемых товаров;
* наличие корзины для выбранных товаров.

**1 ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ**

* 1. Описание предметной области

Цель работы заключается в разработке системы для магазина мебели. То есть, данная система должна предоставить возможность работы в ней персонала и пользователей.

1.2 Определение задания на разработку

Для персонала система должна предусматривать все функции присущие пользователям, а также, она должна иметь возможность добавлять новые, удалять старые и редактировать существующие данные об имеющихся товарах. Для пользователей же должны быть предусмотрены следующие функции: просмотр всей мебели и ее характеристики, возможность добавления товара в корзину с последующим заказом товара, а также фильтрация товара по нужным видам мебели.

1.3 Определение требований к стилю

Все видимые надписи должны иметь шрифт «*Tw Cen MT*». В качестве основного фона используется белый цвет, а для привлечения внимания на целевое действия интерфейса, должен использоваться голубой цвет. Подробнее про использование цветов в системе, можно посмотреть в таблице 1.

Таблица 1 - Используемые цвета

|  |  |  |
| --- | --- | --- |
| Основной фон | Дополнительный фон | Привлечение внимания |
| RGB (255,255,255) | RGB (255,255,255) | RGB (66,227,245) |
|  |  |  |

1.4 Определение требований к GUI

1.4.1 Список товаров

Следует отобразить список товаров в виде общей таблицы,   
в которой будет вся мебель и ее характеристики, также из этой таблицы должна быть возможность перейти в таблицы с описаниями предложенных товаров, так и в виде кнопки-картинки для каждого типа мебели.

Каждый товар должен содержать следующую информацию: идентификационный номер, название, описание, цена, изображение и состояние наличия. Описание не является обязательным к заполнению.

Должна присутствовать возможность сортировки списка   
по убыванию и возрастанию любого из столбцов таблицы товаров.

А также, нужно реализовать возможность удаления персоналом определенного товара. При удалении самого товара, его описание также должно удаляться.

1.4.2 Добавление/редактирование товара

Следует добавить возможность редактирования данных существующего товара, а также добавление нового товара.

На форме должны быть предусмотрены следующие поля: название, стоимость, тип, состояние наличия и описание.

При переходе в режим редактирования все поля выбранного объекта должны быть подгружены в соответствующие поля из базы данных.

При добавлении нового товара идентификатор должен автоматически генерироваться. В случае если администратор редактирует товар, то поле с идентификатором доступно только для чтения.

Стоимость товара может включать сотые значения, а также не может быть отрицательной.

После редактирования/добавления товара, данные в окне списка мебели должны быть обновлены автоматически.

1.4.3 Предложение дополнительных товаров

Необходимо добавить функционал для добавления определённых дополнительных товаров, которые косвенно относятся к товару, который покупает заказчик.

В данном приложении данная функция не была осуществлена.

1.4.4 История продажи товаров

Необходимо реализовать список товаров, которые заказчик уже оформил, но в данном приложении такая функция не была реализована.

**2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ БАЗА ДАННЫХ**

Для работы была взята база данных MySQL, а для работы с ней использовался «*MySQL Workbench*». Всего в базе данных было создано 11 таблиц. Столбец «*id*» является ключом и заполняется автоматически, это сделано для того, чтобы при дальнейшем заполнении таблиц не возникало никаких ошибок. Настройки таблицы «*tovar*» показаны на рисунке 1.1.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1.1 – Настройки таблицы «*tovar*»

Ко всем стокам, кроме «*Izobrazhenie*», применяется параметр «*Not null*», запрещающий оставлять строки не заполненными. У всех первичных ключей, во всех таблицах имеется параметр «*Auto increment*», позволяющий генерировать уникальный номер при вставке новой записи в таблицу. Связь таблиц происходит через внешние ключи, структура которых в данном случае ключами являются записи под названием «*idKategoriya\_Tovarov*» и «*idManufacturer*».

Настройки данных таблицы «*additonal*» показаны на рисунке 1.2.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1.2 – Настройки таблицы «*additional*»

В данной таблице всего 3 строки, где первичным ключом является «*idAdditonal*», и два внешних ключа «*idTovar1*», «*idTovar2*». Также все строки имеют параметр «*Not null*».

Настройки данных таблицы «*dolzhnost*» показаны на рисунке 1.3.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1.3 – Настройки таблицы «*dolzhnost*»

Данная таблица имеет всего 2 строки. Первичным ключом является «*idManufacturer*», к которой принимается параметр «*Not null*» и «*Auto increment*», соответственно. И строка «*Name*», необязательная к заполнению.

Настройки данных таблицы «*dostavka*» показаны на рисунке 1.4.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1.4 – Настройки таблицы «*dostavka*»

В таблице, представленной на рисунке 10, первичным ключом является «*idDostavkal*». К тому же, в данной таблице присутствуют два внешних ключа, «*idKlient*» и «*idZakazi*». Все строки имеют параметр «*Not null*».

Настройки данных таблицы «*kategoriya\_tovarov*» показаны на рисунке 1.5.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1.5 – Настройки таблицы «*kategoria\_tovarov*»

Первичным ключом этой таблицы является «*idKategoriya\_tovarov*», строки «*Nazvanie*» и «*Opisanie*», вновь обязательны к заполнению, и имеют параметр «*Not null*». Внешних ключей, как и две предыдущих таблицы, данная таблица не имеет.

Настройки данных таблицы «*klient*» показаны на рисунке 1.6.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1.6 – Настройки таблицы «*klient*»

Таблица не имеет внешних ключей. Первичным же ключом, является «*idKlient*». К каждой строке таблицы применяется параметр «*Not null*».

Настройки данных таблицы «*manufacturer*» показаны на рисунке 1.7.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1.7 – Настройки таблицы «*manufacturer*»

Данная таблица имеет всего 2 строки. Первичным ключом является «*idManufacturer*», к которой принимается параметр «*Not null*» и «*Auto increment*», соответственно. И строка «*Name*», необязательная к заполнению.

Настройки данных таблицы «*secret*» показаны на рисунке 1.8.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1.8 – Настройки таблицы «*secret*»

Данная таблица служит хранилищем для паролей сотрудников, чтоб защитить их, она вынесена отдельно от паролей клиентов. В этой таблице первичным ключом будет «*idSecret*», вторичным же ключом будет является «*idSotrudnika*». И соответсвтенно, все строки таблицы имеют параметр «*Not null*».

Настройки данных таблицы «*sostavzakaza*» показаны на рисунке 1.9.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1.9 – Настройки таблицы «*sostavzakaza*»

Эта таблица имеет два внешних ключа, которыми являются «*idZakaz*» и «*idTovar*», и первичный ключ «*idSostavzakaza*», а также, строчку «*Kolichestvo*», которая, как и всех строки данной таблицы имеет параметр «*Not null*».

Настройки данных таблицы «*sotrudnik*» показаны на рисунке 1.10.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1.10 – Настройки таблицы «*sotrudnik*»

В данной таблице параметры строк ввода текстовой информации заданны ко всем стокам, кроме «*Otchestvo*» и «*Stazh*», применяется параметр «*Not null*», запрещающий оставлять строки не заполненными. Первичным ключом являетя «*idSotrudnik*». Связь таблиц происходит через внешний ключ «*idDolzhnost*».

Настройки данных таблицы «*zakaz*» показаны на рисунке 7.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1.11 – Настройки таблицы «*zakaz*»

Первичным ключом таблицы является «*idZakaz*», строки «*Summa*» и «*Status*», обязательны к заполнению, и имеют параметр «*Not null*». Внешних ключей таблица не имеет.

*ER* диаграмма данного проекта выглядит следующим образом   
на рисунке 2.

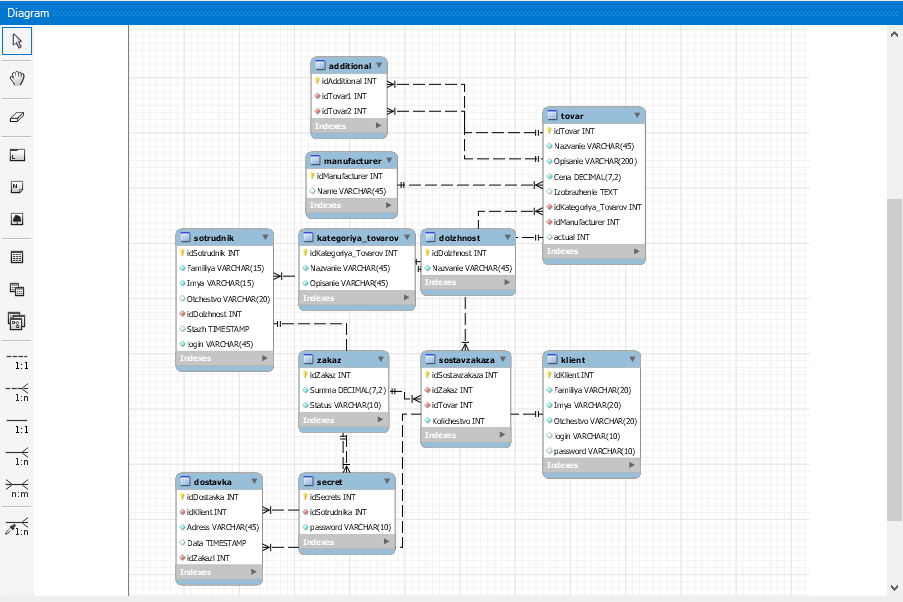


Рисунок 2 – *ER* диаграмма

**3 РЕАЛИЗАЦИЯ ПОДСИСТЕМЫ**

Для реализации проекта был развернут MSQL server, на котором была создана база данных «kulba\_rgr», так же был подключен пакет MDT (MaterialDesignThemes) – это пакет NuGet.

3.1 Описание структуры проекта

Структура проекта довольно проста, было реализовано 2 окна:

* Авторизация
* Главное окно (основное окно для выбора товара)

3.1.1 Авторизация

При включении программы пользователь видит первым окно авторизации, в которой разработана возможность авторизации. Для этого были созданы с помощью MDT поля для ввода данных, и кнопка «Войти», обрабатывающая завершение авторизации, сравнивая введенные данные с данными из таблиц бд. Если данные совпадают, то выводится уведомление с приветствием пользователя и происходит переход на главную страницу. Пример реализации приведен на рисунке 3.

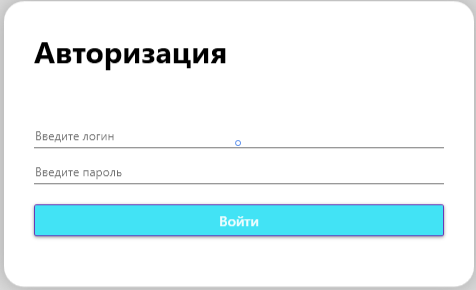


Рисунок 3 – Окно авторизации

3.1.2 Главное окно

При переходе на главное окно пользователю демонстрируются карточки товара. Каждое окно товара основана на основе макета карточки. Которая собирается на основе данных из бд.

3.2 Описание созданных классов

Для упрощения работы был создан файл, в котором описаны классы для взаимодействия с таблицами бд. Пример реализации приведен на рисунке 4[1].

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4 – Пример класса для работы с бд

Для взаимодействия бд с приложением был создан класс DB в котором прописана вся логика для взаимодействия бд и приложения. Пример реализации приведен на рисунке 5[2].

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5 – Класса для работы со списками бд

**4 ПРОЕКТИРОВАНИЕ ГРАФИЧЕСКОГО ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА**

Основным инструментом для разработки интерфейса является бесплатная графическая подсистема *Windows Presentation Foundation* (*WPF*). Для возможности создания более детальных форм, с возможностью тонкой настройки интерфейса использовался *MaterialDesignThemes.* Пример интерфейса показан на рисунке 6[3].

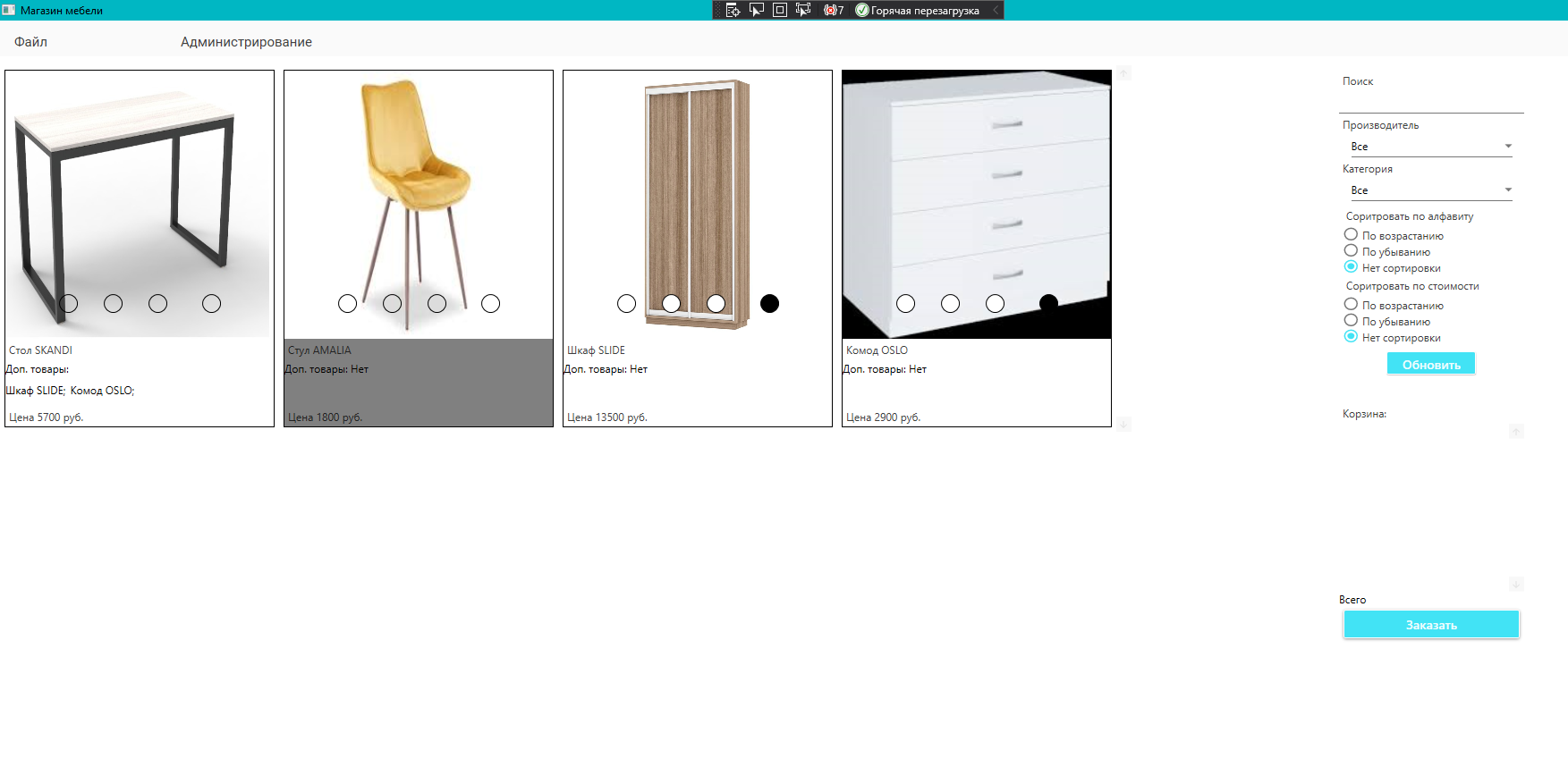


Рисунок 6 – Интерфейс

4.1 Авторизация

Создана возможность авторизации пользователей. Для реализации этой функции необходимо с помощью MaterialDesignThemes создать поля для ввода данных, и кнопка «Войти», обрабатывающая завершение авторизации. Их создание и реализация показаны на рисунках 7-8[4].



Рисунок 7 – Создания компонентов для авторизации

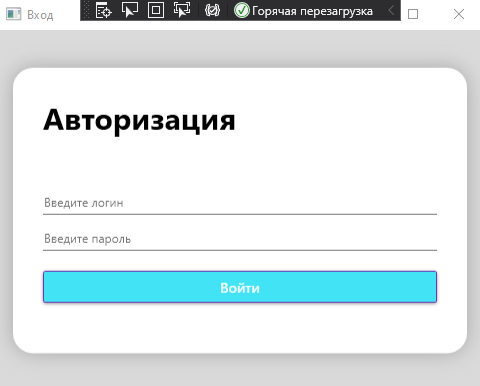


Рисунок 8 – Окно авторизации

Далее создана функция, обрабатывающая клик по кнопке «Войти». При нажатии на эту кнопку система подключается   
к созданной ранее базе данных и проверяет наличие введенного значения   
в таблице «*sotrudnik*». Затем программа проверяет на совпадение логина   
и пароля, введённого пользователем с данными из базы данных. В случае совпадения производится вход под учетной записью с учетом роли пользователя. Если совпадения нет, производится поиск совпадений логина и пароля из таблицы «*klient*». Если есть совпадение с данными из таблицы «*klient*», то производится вход под учетной записью клиента. Если совпадений не найдено в обеих таблицах, то выводится окно с ошибкой, указывающей на неправильность ввода данных[5].

4.2Добавление товара

Осуществлена возможность добавления товара. Для этой функции необходимо создать поля для ввода данных, и кнопки «Выбрать изображения» и «Добавить», обрабатывающая завершение добавления товара. Их создание и реализация показаны на рисунках 9-11.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 9 – Создания компонентов для добавления товара

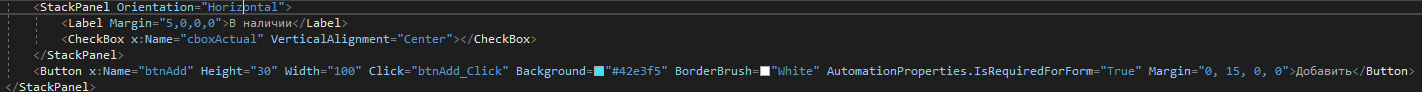


Рисунок 10 – Создания компонентов для добавления товара

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11 – Окно добавления товара

Далее создана функция, обрабатывающая клик по кнопке «Добавить». При нажатии на эту кнопку система подключается   
к созданной ранее базе данных и вносит необходимые данные в таблицу «Tovar».

ЗАКЛЮЧЕНИЕ

В ходе работы были выполнены следующие работы:

- Настройка базы данных;

- Заполнение базы данных необходимой информацией;

- Выполнена проверка корректности заполненных данных;

- Написано приложение на языке C#;

Создан интерфейс программы;

Изучив имеющиеся требования к работе, выяснилось, что база данных содержит все необходимые данные и выполняет все функции, возложенные на нее.

Приложение предоставляет пользователю весь необходимый функционал. Интерфейс обеспечивает удобное выполнение пользователем всех необходимых действий с данными в базе данных.

Готовая работа удовлетворяет всем требованиям, обеспечивает выполнение всех функций в расчетно-графической работе.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Мэтью Мак-Дональд. WPF: Windows Presentation Foundation в .NET 4.5 с примерами на C# 5.0 для профессионалов, 4-е издание = Pro WPF 4.5 in C# 2012: Windows Presentation Foundation in .NET 4.5, 4th edition. — М.: «Вильямс», 2013. — 1024 с. — ISBN 978-5-8459-1854-3.

2 Мэтью Мак-Дональд. WPF: Windows Presentation Foundation в .NET 4.0 с примерами на C# 2010 для профессионалов = Pro WPF in C# 2010: Windows Presentation Foundation with .NET 4.0. — М.: «Вильямс», 2011. — С. 1024. — ISBN 978-5-8459-1657-0.

3 Андерсон, Крис. Основы Windows Presentation Foundation. — СПб.: БХВ-Петербург, 2008. — 432 с. — ISBN 978-5-9775-0265-8.

4 Julia Lerman. Programming Entity Framework. — 2nd Edition. — O’Reilly, 2010. — 920 p. — ISBN 0-596-80726-0.

5 Эндрю Троелсен. Язык программирования C# 2010 и платформа .NET 4.0 = Pro C# 2010 and the .NET 4.0 Platform, 5ed. — М.: «Вильямс», 2010. — С. 1392. — ISBN 978-5-8459-1682-2.