

**Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)**

Колледж информатики и программирования

ОТЧЁТ
По учебной практике

Специальность 09.02.07 «Информационные системы и программирование»
(код) (наименование)

Профессиональный модуль ПМ.02 Осуществление интеграции программных модулей
(код) (наименование)

Междисциплинарный курс МДК.02.01 Технология разработки программного обеспечения
(код) (наименование)

Выполнил:

Студент 4 курса 4ИСИП-121 учебной группы
(номер) (номер)

(подпись) Сакович Н.Д.
(инициалы, фамилия)

Проверил:

Руководитель практики от Колледжа
информатики и программирования
Преподаватель ВКК, к.п.н. Т.Г. Аксёнова
(квалификационная (инициалы, фамилия)
категория или звание,
должность)

Москва – 2024

Перечень работ, выполненных в ходе учебной практики

№ п/п	Виды работ	Оценка
1	Анализ выбранной предметной области	
2	Разработка и оформление технического задания на программное средство	
3	Построение архитектуры программного средства	
4	Анализ требований и построение необходимых диаграмм	

СОДЕРЖАНИЕ

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 1	4
ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 2.....	11
ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 3.....	15
ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 4.....	22

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 1

Тема: Проектирование Архитектуры

Студент: Сакович Н.Д.
Группа: 4ИСИП-421
Преподаватель: Т.Г. Аксёнова
Дата: 19.12.2024

На рисунке 1 предоставлена диаграмма Use Case:

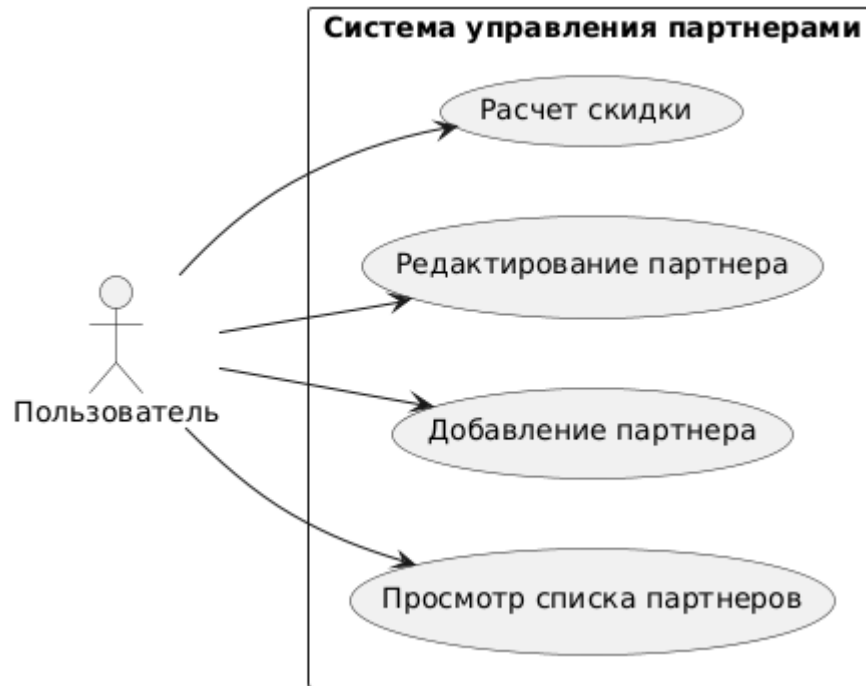


Рисунок 1. Use Case

Листинг кода UML:

```
@startuml
left to right direction
actor "Пользователь" as user

rectangle "Система управления партнерами" {
    usecase "Просмотр списка партнеров" as UC1
    usecase "Добавление партнера" as UC2
    usecase "Редактирование партнера" as UC3
    usecase "Расчет скидки" as UC4
}

user --> UC1
user --> UC2
user --> UC3
user --> UC4
@enduml
```

На рисунке 2 предоставлена диаграмма деятельности:



Рисунок 2. Диаграмма деятельности

Листинг кода UML:

```

@startuml
start
:Загрузка списка партнеров;
:Просмотр списка партнеров;
if (Выбор партнера) then
    :Редактирование партнера;
    :Сохранение изменений;
else
    :Добавление нового партнера;
    :Сохранение изменений;
endif
:Расчет индивидуальной скидки;
:Просмотр детальной информации о партнере;
stop
@enduml
  
```

На рисунке 3 предоставлена диаграмма последовательности:



Рисунок 3. Диаграмма последовательности

Листинг кода UML:

```

@startuml
participant MainWindow as "MainWindow"
participant PartnerEditWindow as "PartnerEditWindow"
participant MaterialDefectEntities as "MaterialDefectEntities"

MainWindow->>MaterialDefectEntities: Загрузка списка партнеров
MaterialDefectEntities-->>MainWindow: Список партнеров
MainWindow->>PartnerEditWindow: Открытие окна редактирования партнера
PartnerEditWindow->>MaterialDefectEntities: Сохранение изменений
MaterialDefectEntities-->>MainWindow: Рейтинг партнера
MainWindow->>MainWindow: Расчет индивидуальной скидки
@enduml
  
```

На рисунке 4 предоставлена диаграмма классов:

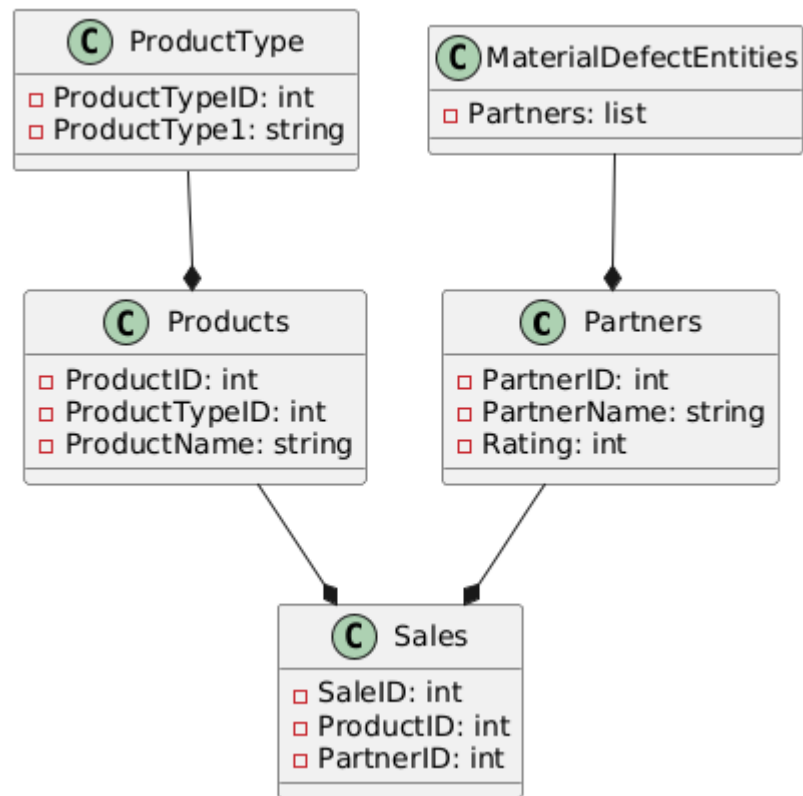


Рисунок 4. Диаграмма классов

Листинг кода UML:

```

@startuml
class Partners {
    - PartnerID: int
    - PartnerName: string
    - Rating: int
}

class ProductType {
    - ProductTypeID: int
    - ProductType1: string
}

class Products {
    - ProductID: int
    - ProductTypeID: int
    - ProductName: string
}

class Sales {
    - SaleID: int
    - ProductID: int
    - PartnerID: int
}

class MaterialDefectEntities {
    - Partners: list
}

Partners --> Sales
ProductType --> Products
Products --> Sales
MaterialDefectEntities --> Partners
@enduml
  
```


На рисунке 5 предоставлена диаграмма пакетов:

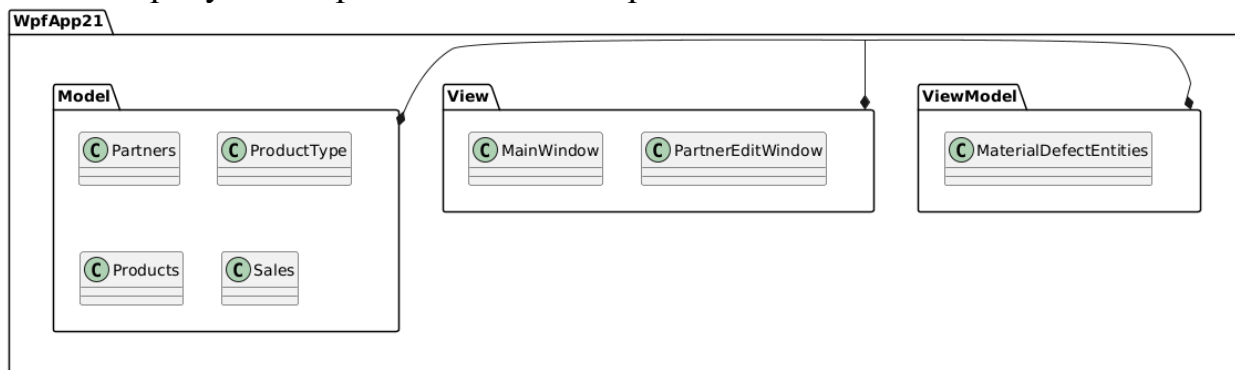


Рисунок 5. Диаграмма пакетов

Листинг кода UML:

```
@startuml
package WpfApp21 {
    package Model {
        class Partners
        class ProductType
        class Products
        class Sales
    }
    package View {
        class MainWindow
        class PartnerEditWindow
    }
    package ViewModel {
        class MaterialDefectEntities
    }
}

WpfApp21 --* Model
WpfApp21 --* View
WpfApp21 --* ViewModel
@enduml
```

На рисунке 6 предоставлена диаграмма бизнес-процессов в нотации IDEF0:

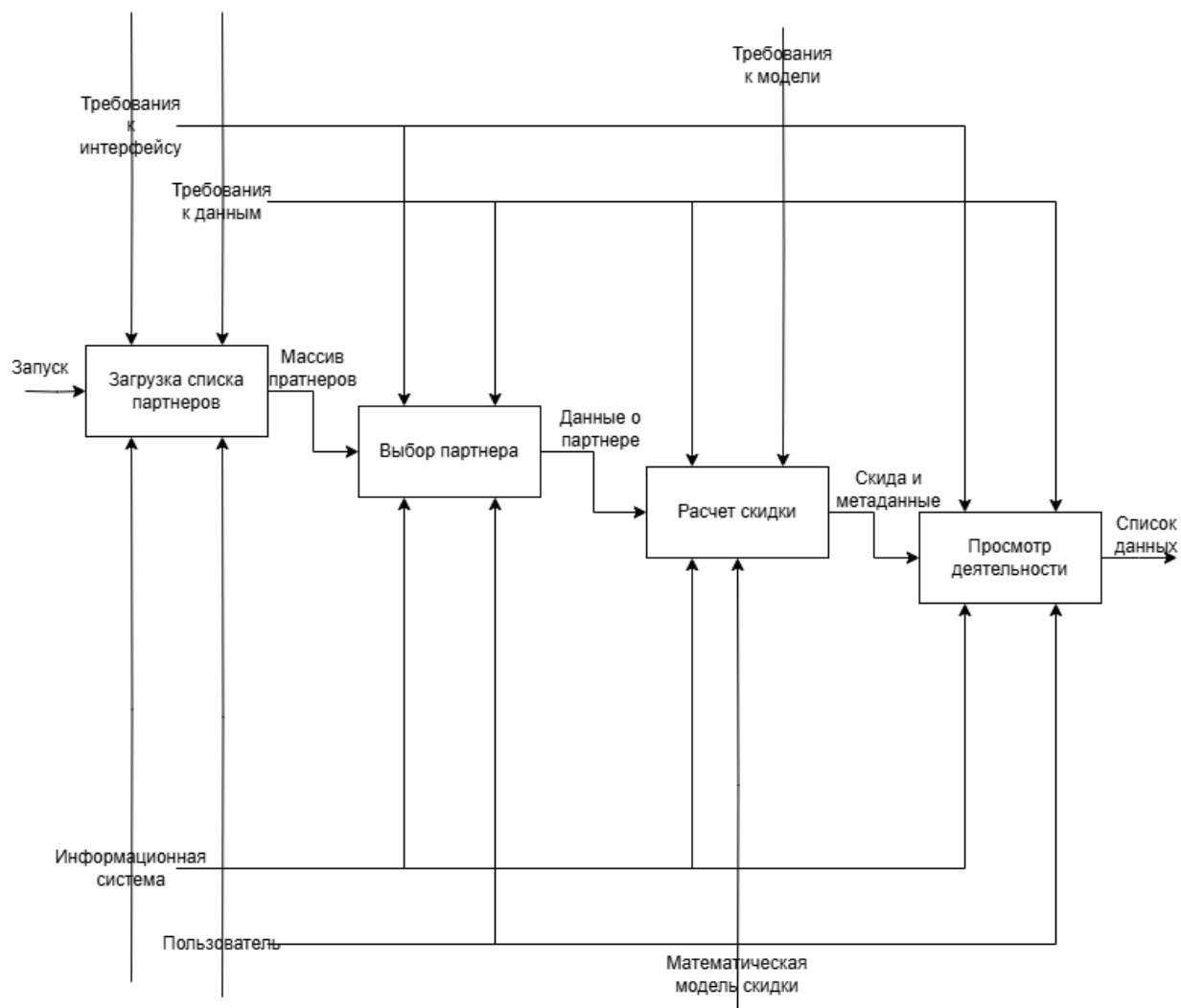


Рисунок 6. IDEF0

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 2

Тема: Проектирование базы данных и разработка объектов базы данных

Студент: Сакович Н.Д.
Группа: 4ИСИП-421
Преподаватель: Т.Г. Аксёнова
Дата: 19.12.2024

ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ И РАЗРАБОТКА ОБЪЕКТОВ БАЗЫ

Цель работы: разработать базу данных в соответствии с заданием, наполнить ее сущностями и атрибутами, сделать ER-диаграмму.

В листинге 1 предоставлен скрипт БД.

Листинг 1

```
USE [MaterialDefect]
GO
/***** Object:  Table [dbo].[MaterialDef]      Script Date: 19.12.2024
21:38:23 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[MaterialDef](
    [MaterialType] [nvarchar](50) NULL,
    [DefectRate] [float] NULL,
    [ProductTypeID] [int] NULL
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Partners]      Script Date: 19.12.2024 21:38:23
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Partners](
    [PartnerID] [int] IDENTITY(1,1) NOT NULL,
    [PartnerType] [nvarchar](50) NULL,
    [PartnerName] [nvarchar](100) NOT NULL,
    [Director] [nvarchar](100) NULL,
    [Email] [nvarchar](100) NULL,
    [Phone] [nvarchar](20) NULL,
    [LegalAddress] [nvarchar](255) NULL,
    [INN] [nvarchar](12) NULL,
    [Rating] [int] NULL,
    PRIMARY KEY CLUSTERED
    (
        [PartnerID] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
    ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
    OFF) ON [PRIMARY],
    UNIQUE NONCLUSTERED
    (
        [PartnerName] ASC
    )WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
    ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
    OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Products]      Script Date: 19.12.2024 21:38:23
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Products](
    [ProductID] [int] IDENTITY(1,1) NOT NULL,
    [ProductTypeID] [int] NOT NULL,
    [ProductName] [nvarchar](255) NOT NULL,
    [Article] [int] NOT NULL,
```

```

        [MinPartnerPrice] [decimal](18, 2) NOT NULL,
PRIMARY KEY CLUSTERED
(
    [ProductID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY],
UNIQUE NONCLUSTERED
(
    [ProductName] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[ProductType]      Script Date: 19.12.2024
21:38:23 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[ProductType](
    [ProductTypeID] [int] IDENTITY(1,1) NOT NULL,
    [ProductType] [nvarchar](50) NOT NULL,
    [Coefficient] [float] NULL,
PRIMARY KEY CLUSTERED
(
    [ProductTypeID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY],
UNIQUE NONCLUSTERED
(
    [ProductType] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Sales]      Script Date: 19.12.2024 21:38:23
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Sales](
    [SaleID] [int] IDENTITY(1,1) NOT NULL,
    [PartnerID] [int] NULL,
    [ProductID] [int] NULL,
    [Quantity] [int] NULL,
    [SaleDate] [datetime] NULL,
PRIMARY KEY CLUSTERED
(
    [SaleID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[MaterialDef] WITH CHECK ADD FOREIGN KEY([ProductTypeID])
REFERENCES [dbo].[ProductType] ([ProductTypeID])
GO
ALTER TABLE [dbo].[Products] WITH CHECK ADD FOREIGN KEY([ProductTypeID])
REFERENCES [dbo].[ProductType] ([ProductTypeID])

```

```

GO
ALTER TABLE [dbo].[Sales] WITH CHECK ADD FOREIGN KEY([PartnerID])
REFERENCES [dbo].[Partners] ([PartnerID])
GO
ALTER TABLE [dbo].[Sales] WITH CHECK ADD FOREIGN KEY([ProductID])
REFERENCES [dbo].[Products] ([ProductID])
GO

```

На рисунке 7 представлена ER-диаграмма БД.

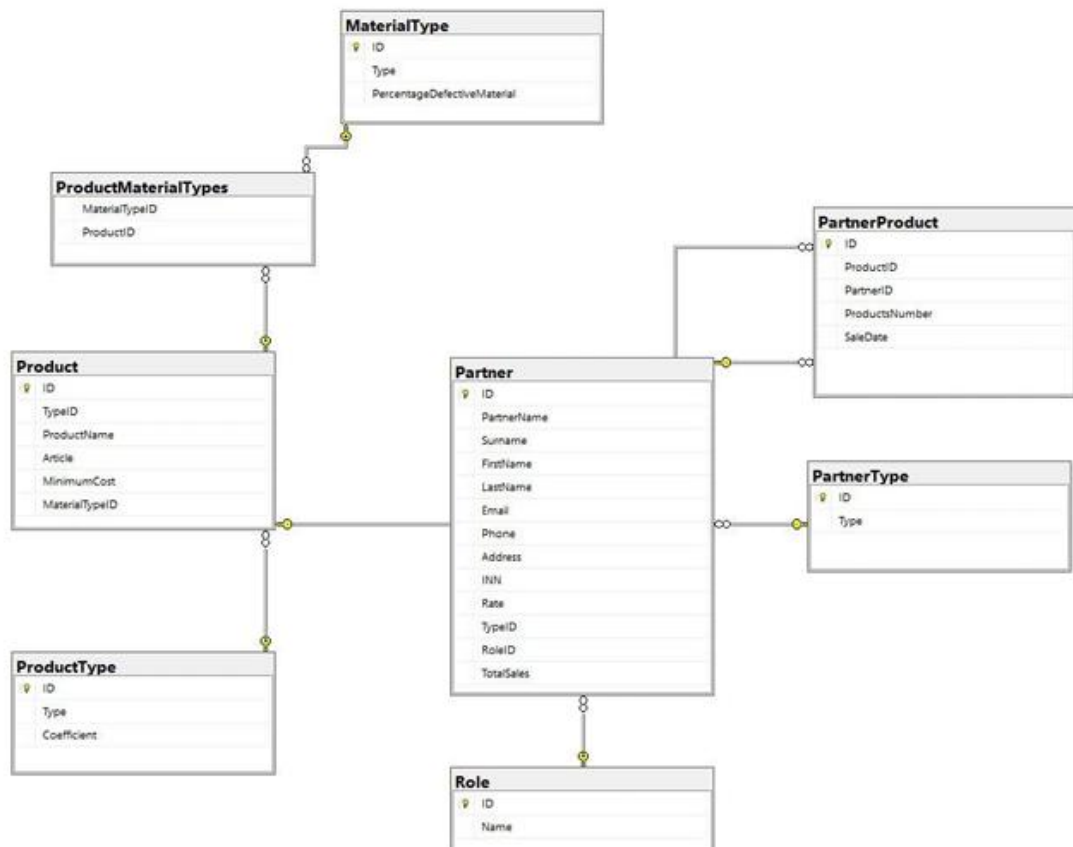


Рисунок 7. ER-диаграмма

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 3

**Тема: Разработка графического приложения для управления
пользователями информационной системы**

Студент: Сакович Н.Д.
Группа: 4ИСИП-421
Преподаватель: Т.Г. Аксёнова
Дата: 19.12.2024

РАЗРАБОТКА ГРАФИЧЕСКОГО ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ ПОЛЬЗОВАТЕЛЯМИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Цель работы: приобрести практические навыки проектирования последовательного пользовательского интерфейса приложения WPF в среде разработки Visual Studio, навигации по страницам WPF, подключения базы данных с помощью технологии Entity Framework ADO.NET.

В листинге 2 предоставлен код страницы MainWindow.xaml

```
<Window x:Class="WpfApp21.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="Система учета партнеров" Height="450" Width="800">
    <Grid>
        <DataGrid x:Name="PartnersDataGrid" HorizontalAlignment="Stretch"
VerticalAlignment="Stretch" Margin="10,10,10,60"
                AutoGenerateColumns="True" IsReadOnly="True"/>
        <Button Content="Рассчитать скидку" HorizontalAlignment="Right"
VerticalAlignment="Bottom" Width="186" Margin="0,0,103,26"
                Click="CalculateDiscountButton_Click" Height="30"/>

        <Button x:Name="OpenPartnerEditWindowButton"
Content="Добавить партнера"
Width="150"
Height="30"
HorizontalAlignment="Left"
Margin="107,371,0,19"
Click="OpenPartnerEditWindowButton_Click"/>
    </Grid>
</Window>
```

В листинге 3 предоставлен код MainWindow.xaml.cs

```
using System.Linq;
using System.Windows;
using System;

namespace WpfApp21
{
    public partial class MainWindow : Window
    {
        private MaterialDefectEntities dbContext;

        public MainWindow()
        {
            InitializeComponent();
            dbContext = new MaterialDefectEntities();
            LoadPartners();
        }

        private void LoadPartners()
        {
            try
            {
                var partners = dbContext.Partners.Select(p => new
                {
                    PartnerType = p.PartnerType,
                    PartnerName = p.PartnerName,
                    Director = p.Director,
                });
            }
        }
    }
}
```



```

        Rating = p.Rating
    }).ToList();

    PartnersDataGrid.ItemsSource = partners;
}
catch (Exception ex)
{
    MessageBox.Show($"Ошибка загрузки данных: {ex.Message}");
}

private void CalculateDiscountButton_Click(object sender,
RoutedEventArgs e)
{
    if (PartnersDataGrid.SelectedItem != null)
    {
        dynamic selectedPartner = PartnersDataGrid.SelectedItem;
        int rating = selectedPartner.Rating;
        double discount = CalculateIndividualDiscount(rating);
        MessageBox.Show($"Индивидуальная скидка для партнера
\"{selectedPartner.PartnerName}\": {discount:F2}%", "Скидка");
    }
    else
    {
        MessageBox.Show("Выберите партнера из списка.");
    }
}

private double CalculateIndividualDiscount(int rating)
{
    if (rating >= 10) return 20.0; // Высокий рейтинг
    if (rating >= 5) return 10.0; // Средний рейтинг
    return 5.0; // Низкий рейтинг
}

private void AddPartnerButton_Click(object sender, RoutedEventArgs
e)
{
    var partnerEditWindow = new PartnerEditWindow(dbContext);
    if (partnerEditWindow.ShowDialog() == true)
    {
        LoadPartners();
    }
}

private void EditPartnerButton_Click(object sender, RoutedEventArgs
e)
{
    if (PartnersDataGrid.SelectedItem is Partners selectedPartner)
    {
        var partnerEditWindow = new PartnerEditWindow(dbContext,
selectedPartner);
        if (partnerEditWindow.ShowDialog() == true)
        {
            LoadPartners();
        }
    }
    else
    {
        MessageBox.Show("Выберите партнёра для редактирования.",
"Информация", MessageBoxButton.OK, MessageBoxImage.Information);
    }
}

```

```

        private void OpenPartnerEditWindowButton_Click(object sender,
RoutedEventArgs e)
        {
            var partnerEditWindow = new PartnerEditWindow(dbContext);
            if (partnerEditWindow.ShowDialog() == true)
            {
                LoadPartners();
            }
        }
    }
}

```

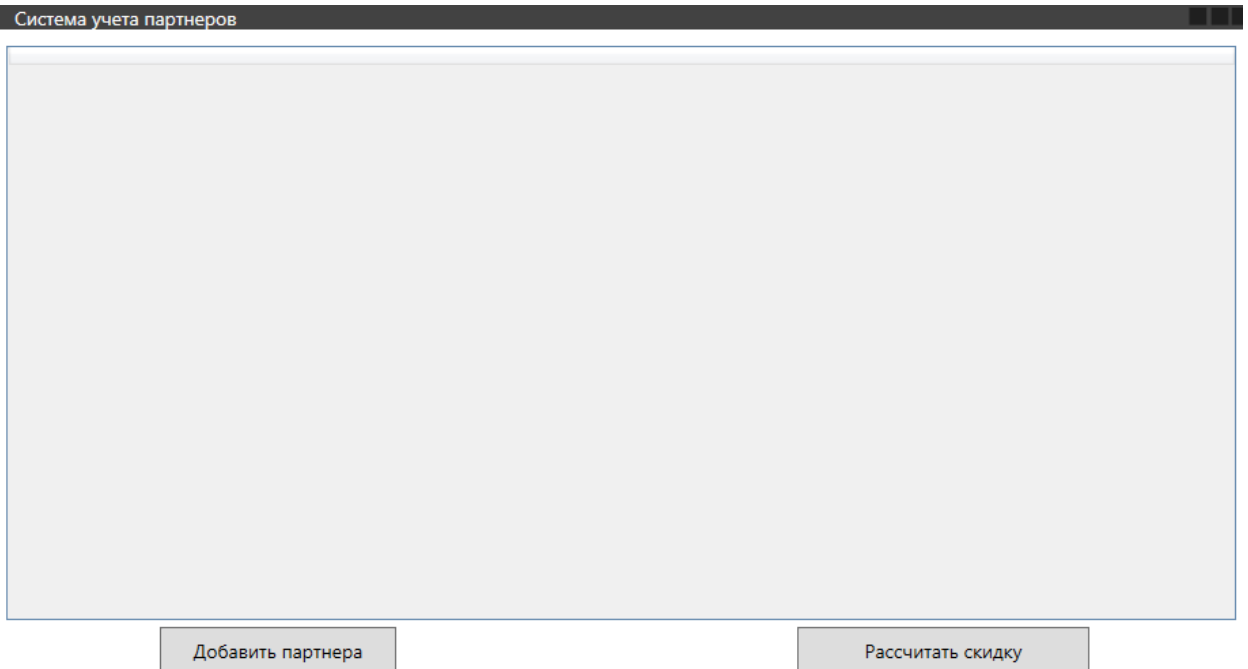


Рисунок 8. Главная страница

В листинге 3 предоставлен код страницы редактора партнёров
PartenrEditWindow.xaml

```

<Window x:Class="WpfApp21.PartnerEditWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="Добавить/Редактировать партнёра" Height="400" Width="400">
    <Grid Margin="10">
        <StackPanel>
            <TextBlock Text="Наименование партнёра:"/>
            <TextBox x:Name="PartnerNameTextBox"/>

            <TextBlock Text="Тип партнёра:"/>
            <ComboBox x:Name="PartnerTypeComboBox">
                <ComboBoxItem Content="Физическое лицо"/>
                <ComboBoxItem Content="Юридическое лицо"/>
            </ComboBox>

            <TextBlock Text="Рейтинг:"/>
            <TextBox x:Name="RatingTextBox"/>

            <TextBlock Text="Адрес:"/>
            <TextBox x:Name="AddressTextBox"/>
        </StackPanel>
    </Grid>

```

```

        <TextBlock Text="ФИО директора:"/>
        <TextBox x:Name="DirectorTextBox"/>

        <TextBlock Text="Телефон:"/>
        <TextBox x:Name="PhoneTextBox"/>

        <TextBlock Text="Email:"/>
        <TextBox x:Name="EmailTextBox"/>

        <StackPanel Orientation="Horizontal" HorizontalAlignment="Right"
Margin="0,20,0,0">
            <Button Content="Сохранить" Width="100" Margin="0,0,10,0"
Click="SaveButton_Click"/>
            <Button Content="Отмена" Width="100"
Click="CancelButton_Click"/>
        </StackPanel>
    </StackPanel>
</Grid>
</Window>

```

В листинге 4 предоставлен код редактора партнёров PartnerEditWindow.xaml.cs

```

using System;
using System.Windows;

namespace WpfApp21
{
    public partial class PartnerEditWindow : Window
    {
        private Partners _partner;
        private MaterialDefectEntities _dbContext;

        public PartnerEditWindow(MaterialDefectEntities dbContext, Partners
partner = null)
        {
            InitializeComponent();
            _dbContext = dbContext;
            _partner = partner;

            if (_partner != null)
            {
                // Заполняем поля для редактирования
                PartnerNameTextBox.Text = _partner.PartnerName;
                PartnerTypeComboBox.SelectedItem = _partner.PartnerType;
                RatingTextBox.Text = _partner.Rating.ToString();
                AddressTextBox.Text = _partner.LegalAddress;
                DirectorTextBox.Text = _partner.Director;
                PhoneTextBox.Text = _partner.Phone;
                EmailTextBox.Text = _partner.Email;
            }
        }

        private void SaveButton_Click(object sender, RoutedEventArgs e)
        {
            try
            {
                // Валидация данных
                if (string.IsNullOrEmpty(PartnerNameTextBox.Text) ||
                    string.IsNullOrEmpty(RatingTextBox.Text) ||
                    !int.TryParse(RatingTextBox.Text, out int rating) ||
rating < 0)
            {

```

```

        MessageBox.Show("Введите корректные данные. Рейтинг
должен быть неотрицательным числом.", "Ошибка", MessageBoxButton.OK,
MessageBoxImage.Error);
        return;
    }

    // Создание или обновление партнёра
    if (_partner == null)
    {
        _partner = new Partners();
        _dbContext.Partners.Add(_partner);
    }

    _partner.PartnerName = PartnerNameTextBox.Text;
    _partner.PartnerType = PartnerTypeComboBox.Text;
    _partner.Rating = rating;
    _partner.LegalAddress = AddressTextBox.Text;
    _partner.Director = DirectorTextBox.Text;
    _partner.Phone = PhoneTextBox.Text;
    _partner.Email = EmailTextBox.Text;

    _dbContext.SaveChanges();
    DialogResult = true;
}
catch (Exception ex)
{
    MessageBox.Show($"Ошибка сохранения данных: {ex.Message}",
"Ошибка", MessageBoxButton.OK, MessageBoxImage.Error);
}

private void CancelButton_Click(object sender, RoutedEventArgs e)
{
    DialogResult = false;
}
}
}

```

Добавить/Редактировать партнёра

Наименование партнёра:

Тип партнёра:

Рейтинг:

Адрес:

ФИО директора:

Телефон:

Email:

Рисунок 9. Редактор партнеров

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 4

Тема: Отладка и тестирование разработанного программного приложения

Студент: Сакович Н.Д.
Группа: 4ИСИП-421
Преподаватель: Т.Г. Аксёнова
Дата: 19.12.2024

ОТЛАДКА И ТЕСТИРОВАНИЕ РАЗРАБОТАННОГО ПРОГРАММНОГО ПРИЛОЖЕНИЯ

Цель работы: провести отладку и тестирование разработанного программного приложения, реализовать обработку исключительных ситуаций в приложении.

В программном приложении было разработано уведомление для пользователей о совершаемых ими ошибках или о запрещенных в рамках задания действиях. Окна сообщений, соответствующих отображаются с соответствующим заголовком. Текст сообщений содержит информацию о совершенных ошибках пользователя.

Сценарий тестирования:

Код	Ввод данных	Ожидаемый результат	Выявление ошибки
1	-10	Введены некорректные данные	нет

При некорректном вводе рейтинга появляется ошибка, как показано на рисунке 10:

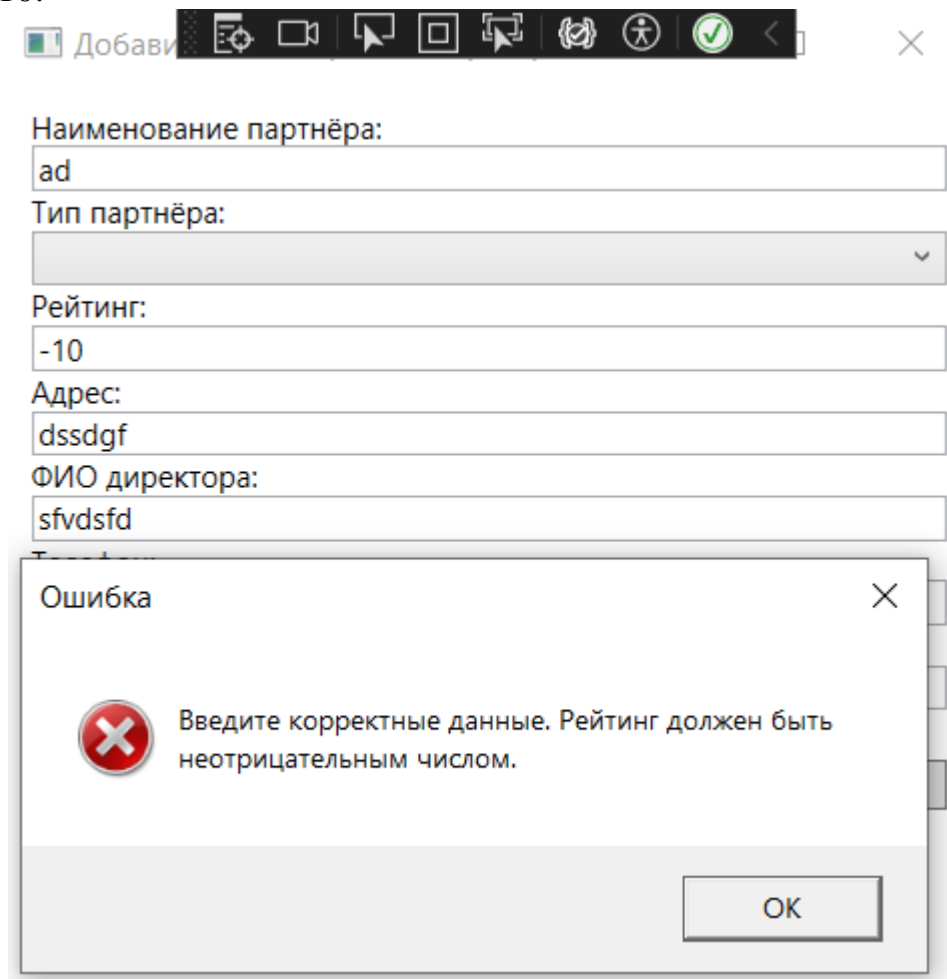


Рисунок 10. Ошибка