**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-321 учебной группы

(номер) (номер)

\_\_\_\_\_\_\_\_\_\_ \_В. Г. Мартоян\_



(подпись) (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 | 5 |
| 2 | Разработка и оформление технического задания на программное средство | 5 |
| 3 | Построение архитектуры программного средства | 5 |
| 4 | Анализ требований и построение необходимых диаграмм | 5 |

Оглавление

[Модуль 1 – Разработка, администрирование и защита баз данных 4](#_Toc185351631)

[Модуль 2 – Разработка модулей программного обеспечения для компьютерных систем 10](#_Toc185351632)

[Модуль 3 – Сопровождение и обслуживание программного обеспечения компьютерных систем 18](#_Toc185351633)

[Модуль 4 – Осуществление интеграции программных модулей 25](#_Toc185351634)

# **Модуль 1 – Разработка, администрирование и защита баз данных**

На рисунке 1 расположен список таблиц созданной базы данных:

Изображение выглядит как текст, Шрифт, снимок экрана, алгебра

Автоматически созданное описание

Рисунок 1 – таблицы базы данных

На рисунках 2-7 изображены заполненные данными таблицы.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 2 – таблица типов материалов

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 3 – таблица продуктов партнера

Изображение выглядит как текст, число, Шрифт, программное обеспечение

Автоматически созданное описание

Рисунок 4 – таблица партнёров

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 5 – таблица типов партнёров

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 6 – таблица продуктов

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 7 – таблица типов продуктов

В листинге 1 представлен скрипт создания базы данных.

Листинг 1 – скрипт базы данных

USE [master]

GO

CREATE DATABASE [ProductionDB]

 CONTAINMENT = NONE

 ON  PRIMARY

( NAME = N'ProductionDB', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\ProductionDB.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )

 LOG ON

( NAME = N'ProductionDB\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\ProductionDB\_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )

 WITH CATALOG\_COLLATION = DATABASE\_DEFAULT

GO

ALTER DATABASE [ProductionDB] SET COMPATIBILITY\_LEVEL = 150

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [ProductionDB].[dbo].[sp\_fulltext\_database] @action = 'enable'

end

GO

ALTER DATABASE [ProductionDB] SET ANSI\_NULL\_DEFAULT OFF

GO

ALTER DATABASE [ProductionDB] SET ANSI\_NULLS OFF

GO

ALTER DATABASE [ProductionDB] SET ANSI\_PADDING OFF

GO

ALTER DATABASE [ProductionDB] SET ANSI\_WARNINGS OFF

GO

ALTER DATABASE [ProductionDB] SET ARITHABORT OFF

GO

ALTER DATABASE [ProductionDB] SET AUTO\_CLOSE ON

GO

ALTER DATABASE [ProductionDB] SET AUTO\_SHRINK OFF

GO

ALTER DATABASE [ProductionDB] SET AUTO\_UPDATE\_STATISTICS ON

GO

ALTER DATABASE [ProductionDB] SET CURSOR\_CLOSE\_ON\_COMMIT OFF

GO

ALTER DATABASE [ProductionDB] SET CURSOR\_DEFAULT  GLOBAL

GO

ALTER DATABASE [ProductionDB] SET CONCAT\_NULL\_YIELDS\_NULL OFF

GO

ALTER DATABASE [ProductionDB] SET NUMERIC\_ROUNDABORT OFF

GO

ALTER DATABASE [ProductionDB] SET QUOTED\_IDENTIFIER OFF

GO

ALTER DATABASE [ProductionDB] SET RECURSIVE\_TRIGGERS OFF

GO

ALTER DATABASE [ProductionDB] SET  ENABLE\_BROKER

GO

ALTER DATABASE [ProductionDB] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [ProductionDB] SET DATE\_CORRELATION\_OPTIMIZATION OFF

GO

ALTER DATABASE [ProductionDB] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [ProductionDB] SET ALLOW\_SNAPSHOT\_ISOLATION OFF

GO

ALTER DATABASE [ProductionDB] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [ProductionDB] SET READ\_COMMITTED\_SNAPSHOT OFF

GO

ALTER DATABASE [ProductionDB] SET HONOR\_BROKER\_PRIORITY OFF

GO

ALTER DATABASE [ProductionDB] SET RECOVERY SIMPLE

GO

ALTER DATABASE [ProductionDB] SET  MULTI\_USER

GO

ALTER DATABASE [ProductionDB] SET PAGE\_VERIFY CHECKSUM

GO

ALTER DATABASE [ProductionDB] SET DB\_CHAINING OFF

GO

ALTER DATABASE [ProductionDB] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF )

GO

ALTER DATABASE [ProductionDB] SET TARGET\_RECOVERY\_TIME = 60 SECONDS

GO

ALTER DATABASE [ProductionDB] SET DELAYED\_DURABILITY = DISABLED

GO

ALTER DATABASE [ProductionDB] SET ACCELERATED\_DATABASE\_RECOVERY = OFF

GO

ALTER DATABASE [ProductionDB] SET QUERY\_STORE = OFF

GO

USE [ProductionDB]

GO

*/\*\*\*\*\*\* Object:  User [user1]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

CREATE USER [user1] FOR LOGIN [user1] WITH DEFAULT\_SCHEMA=[dbo]

GO

*/\*\*\*\*\*\* Object:  Table [dbo].[MaterialTypes]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[MaterialTypes](

    [ID] [int] IDENTITY(1,1) NOT NULL,

    [Type] [nvarchar](50) NOT NULL,

    [DefectRate] [float] NOT NULL,

PRIMARY KEY CLUSTERED

(

    [ID] 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].[PartnerProducts]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[PartnerProducts](

    [ID] [int] IDENTITY(1,1) NOT NULL,

    [ProductID] [int] NOT NULL,

    [PartnerID] [int] NOT NULL,

    [ProductionCount] [int] NOT NULL,

    [SalesDate] [date] NOT NULL,

PRIMARY KEY CLUSTERED

(

    [ID] 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].[Partners]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Partners](

    [ID] [int] IDENTITY(1,1) NOT NULL,

    [PartnerTypeID] [int] NOT NULL,

    [PartnerName] [nvarchar](100) NOT NULL,

    [DirectorFirstName] [nvarchar](50) NULL,

    [DirectorLastName] [nvarchar](50) NULL,

    [DirectorPatronymic] [nvarchar](50) NULL,

    [Email] [nvarchar](100) NULL,

    [TelephoneNumber] [nvarchar](20) NULL,

    [LegalAddress] [nvarchar](200) NULL,

    [INN] [nvarchar](20) NULL,

    [Rating] [float] NULL,

    [Discount] [decimal](5, 2) NULL,

PRIMARY KEY CLUSTERED

(

    [ID] 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].[PartnersTypes]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[PartnersTypes](

    [ID] [int] IDENTITY(1,1) NOT NULL,

    [PartnerTypeName] [nvarchar](50) NOT NULL,

PRIMARY KEY CLUSTERED

(

    [ID] 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: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Products](

    [ID] [int] IDENTITY(1,1) NOT NULL,

    [TypeID] [int] NOT NULL,

    [ProductName] [nvarchar](100) NOT NULL,

    [Article] [nvarchar](50) NOT NULL,

    [MinCost] [decimal](18, 2) NOT NULL,

    [MaterialTypeID] [int] NULL,

PRIMARY KEY CLUSTERED

(

    [ID] 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].[ProductTypes]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[ProductTypes](

    [ID] [int] IDENTITY(1,1) NOT NULL,

    [Type] [nvarchar](50) NOT NULL,

    [Coefficient] [float] NOT NULL,

PRIMARY KEY CLUSTERED

(

    [ID] 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].[sysdiagrams]    Script Date: 1/13/2025 10:42:58 AM \*\*\*\*\*\*/*

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[sysdiagrams](

    [name] [nvarchar](128) NOT NULL,

    [principal\_id] [int] NOT NULL,

    [diagram\_id] [int] IDENTITY(1,1) NOT NULL,

    [version] [int] NULL,

    [definition] [varbinary](max) NULL,

PRIMARY KEY CLUSTERED

(

    [diagram\_id] 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] TEXTIMAGE\_ON [PRIMARY]

GO

ALTER TABLE [dbo].[PartnerProducts]  WITH CHECK ADD  CONSTRAINT [FK\_PartnerProducts\_Partners] FOREIGN KEY([PartnerID])

REFERENCES [dbo].[Partners] ([ID])

GO

ALTER TABLE [dbo].[PartnerProducts] CHECK CONSTRAINT [FK\_PartnerProducts\_Partners]

GO

ALTER TABLE [dbo].[PartnerProducts]  WITH CHECK ADD  CONSTRAINT [FK\_PartnerProducts\_Products] FOREIGN KEY([ProductID])

REFERENCES [dbo].[Products] ([ID])

GO

ALTER TABLE [dbo].[PartnerProducts] CHECK CONSTRAINT [FK\_PartnerProducts\_Products]

GO

ALTER TABLE [dbo].[Partners]  WITH CHECK ADD  CONSTRAINT [FK\_Partners\_PartnersTypes] FOREIGN KEY([PartnerTypeID])

REFERENCES [dbo].[PartnersTypes] ([ID])

GO

ALTER TABLE [dbo].[Partners] CHECK CONSTRAINT [FK\_Partners\_PartnersTypes]

GO

ALTER TABLE [dbo].[Products]  WITH CHECK ADD  CONSTRAINT [FK\_Products\_MaterialTypes] FOREIGN KEY([MaterialTypeID])

REFERENCES [dbo].[MaterialTypes] ([ID])

GO

ALTER TABLE [dbo].[Products] CHECK CONSTRAINT [FK\_Products\_MaterialTypes]

GO

ALTER TABLE [dbo].[Products]  WITH CHECK ADD  CONSTRAINT [FK\_Products\_ProductTypes] FOREIGN KEY([TypeID])

REFERENCES [dbo].[ProductTypes] ([ID])

GO

ALTER TABLE [dbo].[Products] CHECK CONSTRAINT [FK\_Products\_ProductTypes]

GO

USE [master]

GO

ALTER DATABASE [ProductionDB] SET  READ\_WRITE

GO

# **Модуль 2 – Разработка модулей программного обеспечения для компьютерных систем**

В листинге 2 представлена разметка окна MainWindow

Листинг 2 - разметка окна MainWindow

<Window x:Class="MartoyanWGUP2.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:MartoyanWGUP2"

mc:Ignorable="d"

Title="Martoyan" Height="450" Width="800">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="40"/>

<RowDefinition/>

</Grid.RowDefinitions>

<Button x:Name="Back" Grid.Row="0" Content="Назад" VerticalAlignment="Center" HorizontalAlignment="Left" Margin="5,0,0,0" Click="Back\_Click"/>

<Frame x:Name="Partners" Grid.Row="1" NavigationUIVisibility="Hidden" Navigated="Partners\_Navigated"/>

<Image HorizontalAlignment="Right" Height="38" Margin="0,0,0,0" VerticalAlignment="Top" Width="38" Grid.Row="0" Source = "Resources/MasterPol.png"/>

</Grid>

</Window>

В листинге 3 представлен код окна MainWindow.

Листинг 3 - код окна MainWindow

using MartoyanWGUP2.Classes;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Navigation;

namespace MartoyanWGUP2

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

Partners.NavigationService.Navigate(new Pages.PartnerListPage());

ProductionClass pc = new ProductionClass();

}

private void Back\_Click(object sender, RoutedEventArgs e)

{

if (Partners.CanGoBack) Partners.GoBack();

}

private void Partners\_Navigated(object sender, NavigationEventArgs e)

{

if (!(e.Content is Page page)) return;

Title = $"{page.Title}";

if (page is Pages.AddPartnerPage)

Back.Visibility = Visibility.Visible;

else if (page is Pages.PartnerRealisationHistoryPage)

Back.Visibility = Visibility.Visible;

else

Back.Visibility = Visibility.Hidden;

}

}

}

На рисунке 8 изображена страница списка партнёров.

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

Рисунок 8 – страница партнёров

В листинге 4 представлена XAML-разметка страницы партнёров.

Листинг 4 – XAML-разметка страницы партнёров

<Page x:Class="MartoyanWGUP2.Pages.PartnerListPage"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:MartoyanWGUP2.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="PartnerListPage">

<Grid>

<Button x:Name="AddButton" Grid.Row="0" Content="Добавить" VerticalAlignment="Top" HorizontalAlignment="Right" Margin="0,-35,45,0" Click="AddButton\_Click"/>

<Button x:Name="RealizationHystoryButton" Grid.Row="0" Content="История реализации" VerticalAlignment="Top" HorizontalAlignment="Right" Margin="0,-35,170,0" Click="RealizationHystoryButton\_Click"/>

<ListView x:Name="PartnersList" ScrollViewer.VerticalScrollBarVisibility = "Hidden">

<ListView.ItemContainerStyle>

<Style TargetType="ListViewItem">

<EventSetter Event="MouseDoubleClick" Handler="ListViewItem\_MouseDoubleClick"/>

</Style>

</ListView.ItemContainerStyle>

<ListView.ItemTemplate>

<DataTemplate>

<Border BorderThickness="1" BorderBrush="Gray" Margin="2" Padding="5" CornerRadius="3" Background="#F4E8D3">

<Grid Style="{StaticResource parList}">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="10\*"/>

<ColumnDefinition Width="1\*"/>

</Grid.ColumnDefinitions>

<StackPanel Grid.Column="0" Margin="0,0,0,0">

<TextBlock>

<Run Text="{Binding PartnersTypes.PartnerTypeName}" FontSize="17"/>

<Run Text=" | " FontSize="17"/>

<Run Text="{Binding PartnerName}" FontSize="17"/>

</TextBlock>

<TextBlock>

<Run Text="Директор: "/>

<Run Text="{Binding DirectorFirstName}"/>

<Run Text="{Binding DirectorLastName}"/>

<Run Text="{Binding DirectorPatronymic}"/>

</TextBlock>

<TextBlock>

<Run Text="+7 "/>

<Run Text="{Binding TelephoneNumber}"/>

</TextBlock>

<TextBlock>

<Run Text="Рейтинг: "/>

<Run Text="{Binding Rating}"/>

</TextBlock>

</StackPanel>

<TextBlock Grid.Column="1" VerticalAlignment="Top" HorizontalAlignment="Right" FontSize="17">

<Run Text="{Binding Discount, TargetNullValue='0'}"/>

<Run Text="%"/>

</TextBlock>

</Grid>

</Border>

</DataTemplate>

</ListView.ItemTemplate>

</ListView>

</Grid>

</Page>

На рисунке 9 расположен метод для расчёта скидки для партнёров.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 9 – метод расчёта скидки для партнёра.

В листинге 5 представлен код страницы партнёров.

Листинг 5 - код страницы партнёров

using System.Collections.Generic;

using System.Linq;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Input;

using System.Windows.Navigation;

namespace MartoyanWGUP2.Pages

{

public partial class PartnerListPage : Page

{

ProductionDBEntities \_context = ProductionDBEntities.GetContext();

public PartnerListPage()

{

InitializeComponent();

PartnersList.ItemsSource = ProductionDBEntities.GetContext().Partners.ToList();

CountDiscountByPartners();

}

private void CountDiscountByPartners()

{

var partnerSales = \_context.PartnerProducts

.GroupBy(x => x.PartnerID)

.Select(group => new

{

PartnerId = group.Key,

TotalSales = group.Sum(p => p.ProductionCount)

})

.ToList();

foreach (var partner in \_context.Partners)

{

var salesData = partnerSales.FirstOrDefault(ps => ps.PartnerId == partner.ID);

if (salesData != null)

{

if (salesData.TotalSales < 10000)

{

partner.Discount = 0;

}

else if (salesData.TotalSales >= 10000 && salesData.TotalSales < 50000)

{

partner.Discount = 5;

}

else if (salesData.TotalSales >= 50000 && salesData.TotalSales < 300000)

{

partner.Discount = 10;

}

else partner.Discount = 15;

}

}

\_context.SaveChanges();

}

private void ListViewItem\_MouseDoubleClick(object sender, MouseButtonEventArgs e)

{

if (PartnersList.SelectedItem is Partner partner)

{

NavigationService.Navigate(new AddPartnerPage(partner));

}

}

private void RealizationHystoryButton\_Click(object sender, RoutedEventArgs e)

{

List<Partner> partners = PartnersList.SelectedItems.Cast<Partner>().ToList();

if (partners.Count > 1)

{

MessageBox.Show("Выберите только одного партнера!", "Внимание");

}

else if (partners.Count == 0) MessageBox.Show("Выберите партнера!", "Внимание");

else NavigationService?.Navigate(new PartnerRealisationHistoryPage(partners[0]));

}

private void AddButton\_Click(object sender, RoutedEventArgs e)

{

AddPartnerPage addpage = new AddPartnerPage(null);

NavigationService?.Navigate(addpage);

addpage.OnPartnerAdded += (newPartner) =>

{

var list = \_context.Partners.ToList();

list.Add(newPartner);

PartnersList.ItemsSource = list;

};

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

Button button = (Button)sender;

Partner partners = (Partner)button.DataContext;

NavigationService?.Navigate(new PartnerRealisationHistoryPage(partners));

}

}

}

На листинге 5 можно увидеть код словаря стилей приложения

Листинг 5 – словарь стилей приложения

<ResourceDictionary xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">

<Style TargetType="Button">

<Setter Property="Width" Value="120"/>

<Setter Property="Height" Value="30"/>

<Setter Property="Background" Value="#67BA80"/>

</Style>

<Style TargetType="StackPanel">

<Setter Property="Background" Value="#F4E8D3"/>

<Setter Property="Width" Value="500"/>

</Style>

<Style TargetType="ListViewItem">

<Setter Property="Height" Value="200"/>

<Setter Property="Padding" Value="3"/>

<Setter Property="HorizontalContentAlignment" Value="Left" />

</Style>

<Style x:Key="AddPageTextBlock" TargetType="TextBlock">

<Setter Property="VerticalAlignment" Value="Center"/>

<Setter Property="HorizontalAlignment" Value="Center"/>

</Style>

<Style TargetType="TextBox">

<Setter Property="Width" Value="200"/>

<Setter Property="Height" Value="30"/>

</Style>

<Style TargetType="Grid">

<Setter Property="Background" Value="#FFFFFF"/>

</Style>

<Style TargetType="Grid" x:Key="parList">

<Setter Property="Background" Value="#F4E8D3"/>

<Setter Property="Width" Value="500"/>

</Style>

<Style TargetType="DataGridCell">

<Setter Property="Background" Value="#F4E8D3"/>

</Style>

<Style TargetType="DataGrid">

<Setter Property="AutoGenerateColumns" Value="False"/>

<Setter Property="IsReadOnly" Value="True"/>

</Style>

</ResourceDictionary>

# **Модуль 3 – Сопровождение и обслуживание программного обеспечения компьютерных систем**

На рисунке 10 расположена разметка окна добавления партнёра.

Изображение выглядит как текст, программное обеспечение, снимок экрана, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 10 – разметка окна добавления партнёра

В листинге 7 представлена разметка окна добавления партнёра.

Листинг 7 - разметка окна добавления партнёра.

<Page x:Class="MartoyanWGUP2.Pages.AddPartnerPage"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:MartoyanWGUP2.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="AddPartnerPage">

<Grid>

<!-- Определение строк и столбцов -->

<Grid.RowDefinitions>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="Auto"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<!-- Наименование -->

<TextBlock Text="Наименование:" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<TextBox x:Name="PartnerName" ToolTip="Наименование партнёра" Text="{Binding PartnerName}" Grid.Column="1" Margin="5"/>

<!-- Тип -->

<TextBlock Text="Тип:" Grid.Row="1" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<ComboBox x:Name="PartnerType" Grid.Column="1" Grid.Row="1" Height="20" Width="120"

DisplayMemberPath="PartnerTypeName"

SelectedValuePath="ID"

SelectedValue="{Binding PartnerTypeID}" Margin="5"/>

<!-- Рейтинг -->

<TextBlock Text="Рейтинг:" Grid.Row="2" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<TextBox x:Name="PartnerRate" ToolTip="Рейтинг партнёра" Grid.Column="1" Grid.Row="2" Text="{Binding Rating}" PreviewTextInput="PartnerRate\_PreviewTextInput" Margin="5"/>

<!-- Юр. адрес -->

<TextBlock Text="Юр. адрес:" Grid.Row="3" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<TextBox x:Name="PartnerAddress" ToolTip="Юридический адрес организации" Grid.Column="1" Grid.Row="3" Text="{Binding LegalAddress}" Margin="5"/>

<!-- ФИО директора -->

<TextBlock Text="ФИО директора:" Grid.Row="4" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<StackPanel Grid.Column="1" Grid.Row="4" Orientation="Horizontal" Margin="5">

<TextBox x:Name="DirectorSurname" ToolTip="Фамилия директора организации" Width="100" Text="{Binding DirectorFirstName}" Margin="0,0,5,0"/>

<TextBox x:Name="DirectorName" ToolTip="Имя директора организации" Width="100" Text="{Binding DirectorLastName}" Margin="0,0,5,0"/>

<TextBox x:Name="DirectorLastName" ToolTip="Отчество директора организации" Width="100" Text="{Binding DirectorPatronymic}"/>

</StackPanel>

<!-- Телефон -->

<TextBlock Text="Телефон:" Grid.Row="5" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<TextBox x:Name="PartnerPhone" ToolTip="Телефон организации" Grid.Column="1" Grid.Row="5" Text="{Binding TelephoneNumber}" PreviewTextInput="PartnerPhone\_PreviewTextInput" Margin="5"/>

<!-- Email -->

<TextBlock Text="Email:" Grid.Row="6" Style="{StaticResource AddPageTextBlock}" VerticalAlignment="Center"/>

<TextBox x:Name="PartnerEmail" ToolTip="Адрес электронной почты" Grid.Column="1" Grid.Row="6" Text="{Binding Email}" Margin="5"/>

<!-- Кнопки -->

<Button x:Name="CancelButton" Content="Отмена" Grid.Row="7" Grid.ColumnSpan="2" HorizontalAlignment="Left" Margin="5" Click="CancelButton\_Click"/>

<Button x:Name="SaveButton" Content="Сохранить" Grid.Row="7" Grid.Column="1" HorizontalAlignment="Right" Margin="5" Click="SaveButton\_Click"/>

</Grid>

</Page>

На рисунке 11 отображена проверка на заполненность всех полей.

Изображение выглядит как текст, снимок экрана, диаграмма, План

Автоматически созданное описание

Рисунок 11 – сообщение о незаполненности полей

На рисунке 12 расположено сообщение об успешном сохранении данных.

Изображение выглядит как текст, снимок экрана, диаграмма, число

Автоматически созданное описание

Рисунок 12 – сообщение об успешном сохранении данных

В листинге 8 представлен код окна добавления партнёра.

Листинг 8 – код окна добавления партнёра

using System;

using System.Linq;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Input;

using System.Windows.Navigation;

namespace MartoyanWGUP2.Pages

{

public partial class AddPartnerPage : Page

{

ProductionDBEntities db = ProductionDBEntities.GetContext();

public event Action<Partner> OnPartnerAdded;

Partner \_currentpartner;

public AddPartnerPage(Partner partner)

{

InitializeComponent();

if (partner != null)

{

Title = "Редактирование партнера";

\_currentpartner = partner;

}

else

{

Title = "Добавление партнера";

\_currentpartner = new Partner();

}

PartnerType.ItemsSource = db.PartnersTypes.ToList();

DataContext = \_currentpartner;

}

private void CancelButton\_Click(object sender, RoutedEventArgs e)

{

if (this.NavigationService.CanGoBack) NavigationService.GoBack();

}

private void SaveButton\_Click(object sender, RoutedEventArgs e)

{

if (PartnerName.Text == null

|| PartnerType.SelectedIndex == -1

|| PartnerRate.Text == null

|| PartnerAddress.Text == null

|| DirectorSurname.Text == null

|| DirectorName.Text == null

|| DirectorLastName.Text == null

|| PartnerPhone.Text == null

|| PartnerEmail.Text == null)

{

MessageBox.Show("Заполните все поля!");

return;

}

if (\_currentpartner.ID == 0)

{

db.Partners.Add(\_currentpartner);

OnPartnerAdded.Invoke(\_currentpartner);

}

try

{

db.SaveChanges();

MessageBox.Show("Информация сохранена!");

if (NavigationService.CanGoBack) NavigationService.GoBack();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString());

}

}

private void PartnerRate\_PreviewTextInput(object sender, TextCompositionEventArgs e)

{

TextBox textBox = sender as TextBox;

textBox.Text = textBox.Text.Replace(" ", string.Empty);

textBox.SelectionStart = textBox.Text.Length;

if (!IsValidInput(textBox.Text, e.Text[0]))

{

e.Handled = true;

}

}

private void PartnerPhone\_PreviewTextInput(object sender, TextCompositionEventArgs e)

{

TextBox textBox = sender as TextBox;

if (!IsValidInputNumber(textBox.Text, e.Text[0]))

{

e.Handled = true;

}

}

private bool IsValidInput(string text, char input)

{

if (!char.IsDigit(input))

return false;

if (text.Length > 1)

return false;

return true;

}

private bool IsValidInputNumber(string text, char input)

{

if (!char.IsDigit(input))

return false;

if (text.Length > 12)

return false;

return true;

}

}

}

# **Модуль 4 – Осуществление интеграции программных модулей**

На рисунке 13 расположена разметка страницы продаж партнёров с привязкой столбцов к выборке из базы данных.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 13 – разметка истории реализации партнёров

В листинге 9 представлена разметка страницы истории реализации партнёров.

Листинг 9 – разметка страницы истории реализации партнёров

<Page x:Class="MartoyanWGUP2.Pages.PartnerRealisationHistoryPage"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:local="clr-namespace:MartoyanWGUP2.Pages"

mc:Ignorable="d"

d:DesignHeight="450" d:DesignWidth="800"

Title="PartnerRealisationHistoryPage">

<Grid>

<DataGrid x:Name="PartnerProductsDataGrid" HeadersVisibility="Column">

<DataGrid.Columns>

<DataGridTextColumn Header="Наименование продукции" Binding="{Binding Product.ProductName}" Width="auto"/>

<DataGridTextColumn Header="Количество" Binding="{Binding ProductionCount}" Width="auto"/>

<DataGridTextColumn Header="Дата продажи" Binding="{Binding SalesDate}" Width="\*"/>

</DataGrid.Columns>

</DataGrid>

</Grid>

</Page>

На рисунке 14 изображён фрагмент года с передачей выборки из базы данных в таблицу.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 14 – передача записей из базы данных в таблицу

В листинге 9 представлен код страницы продаж партнёров

using System.Linq;

using System.Windows.Controls;

namespace MartoyanWGUP2.Pages

{

public partial class PartnerRealisationHistoryPage : Page

{

public PartnerRealisationHistoryPage(Partner partner)

{

InitializeComponent();

Title = $"История реализации товара {partner.PartnersType.PartnerTypeName} {partner.PartnerName}";

PartnerProductsDataGrid.ItemsSource = partner.PartnerProducts.ToList();

}

}

}

На листинге 10 показан класс с методом для расчёта необходимого количества материала

Листинг 10 – класс с методом для расчёта необходимого количества материала

using System;

using System.Linq;

namespace MartoyanWGUP2.Classes

{

internal class ProductionClass

{

ProductionDBEntities \_context = ProductionDBEntities.GetContext();

public int GetProductionCount(int ProdID, int MatTypeID, int CountNeeded, double Fp, double Sp)

{

if (Fp < 0 || Sp < 0)

{

return -1;

}

double Coeficent = (double)\_context.ProductTypes.FirstOrDefault(x => x.ID == ProdID).Coefficient;

double DefectRate = (double)\_context.MaterialTypes.FirstOrDefault(x => x.ID == MatTypeID).DefectRate;

if (Coeficent == 0 || DefectRate == 0)

{

return -1;

}

double MaterialForOneProduction = Fp \* Sp \* Coeficent;

return (int)Math.Ceiling(Convert.ToDouble(CountNeeded) / (MaterialForOneProduction \* (1.0 + DefectRate)));

}

}

}

На рисунке 15 изображён процесс установки пользовательской иконки приложения.

Изображение выглядит как текст, программное обеспечение, снимок экрана, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 15 – установка пользовательского значка приложения