

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

Колледж информатики и программирования

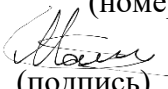
**ОТЧЁТ**  
**По учебной практике**

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

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

Междисциплинарный курс МДК.01.04 Системное программирование  
(код) (наименование)

Выполнил:

Студент 4 курса 4ИСИП-321 учебной группы  
(номер) (номер)  
 (подпись) Д.А. Мамонтов  
(инициалы, фамилия)

Проверил:

Руководитель практики от Колледжа  
информатики и программирования  
Преподаватель 1КК Е.Л. Альшакова  
(квалификационная категория или звание, (инициалы, фамилия)  
должность)

Москва – 2024

### Перечень работ, выполненных в ходе учебной практики

№ п/п	Виды работ	Оценка
1	Разработка структуры, перечня артефактов и протоколов проекта	5
2	Командная работа над проектом с использованием системы контроля версий	5
3	Отладка программного проекта	5

**СОДЕРЖАНИЕ**

**УП.02 .....4**

## УП.02

Анализ предметной области начинается с диаграммы прецедентов (Use Case) для основных пользователей системы, что позволяет сформировать представление о структуре приложения. Диаграмма представлена на рисунке 1.

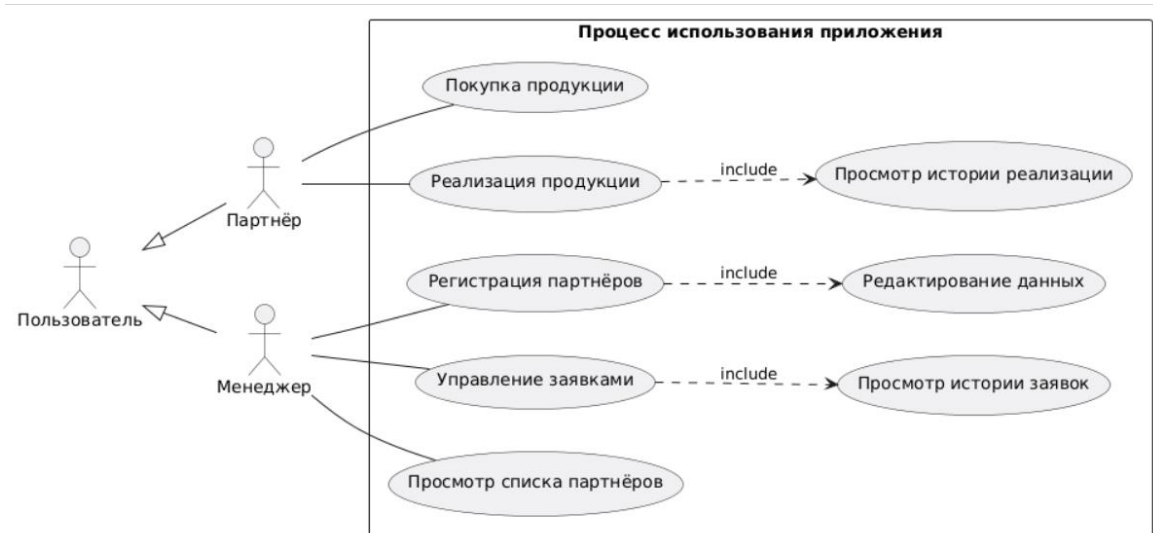


Рисунок 1 – Диаграмма прецедентов

Код для создания диаграммы представлен в листинге 1.

Листинг 1 – Код, формирующий диаграмму прецедентов

```
@startuml
left to right direction
actor "Пользователь" as User
actor "Партнёр" as Partner
actor "Менеджер" as Manager
User <|--Partner
User <|--Manager
rectangle "Процесс использования приложения"{
(Покупка продукции) as UC1
(Реализация продукции) as UC2
(Регистрация партнёров) as UC4
(Управление заявками) as UC6
(Просмотр истории заявок) as UC10
Partner -- UC1
Partner -- UC2
Manager -- UC4
Manager -- (Просмотр списка партнёров)
Manager -- UC6
UC2 ..>(Просмотр истории реализации):include
UC4 ..>(Редактирование данных):include
UC6 ..>UC10:include
}
@enduml
```

Далее переходим к проектированию базы данных, состоящей из 7 таблиц. Для большинства таблиц имеется исходная информация, которая импортируется из Excel, остальные таблицы добавляются вручную. Информация о базе данных представлена на рисунках 2-9.

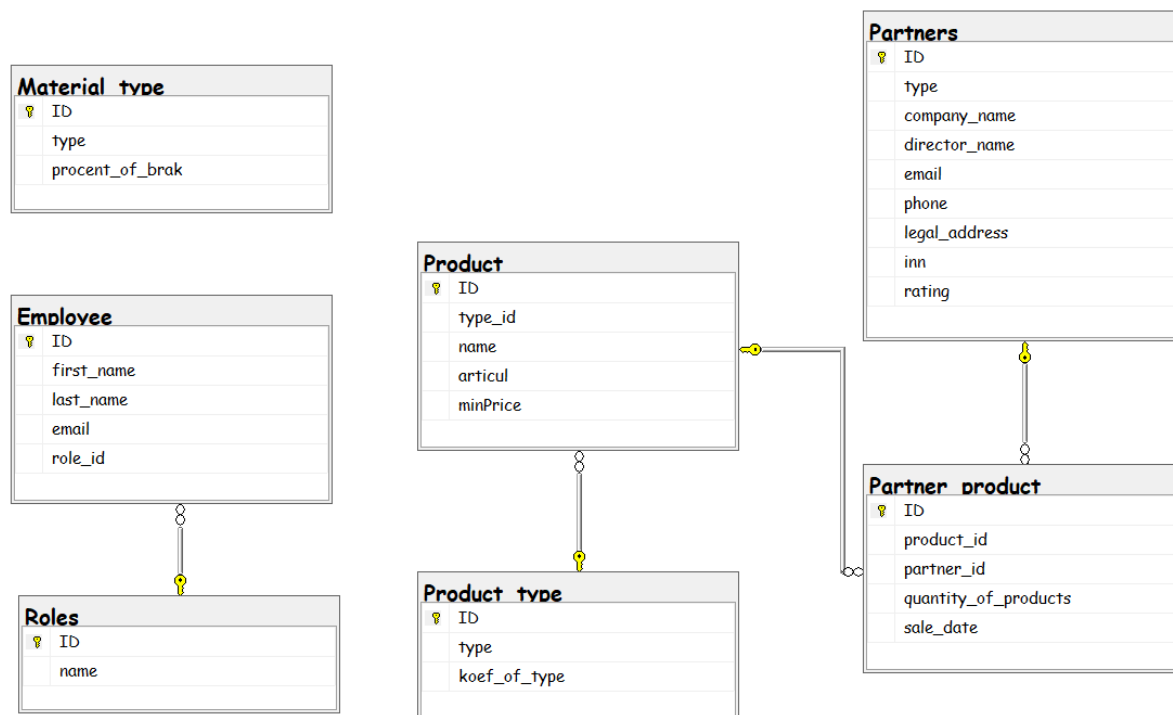


Рисунок 2 – ER-диаграмма БД

Results		Messages			
	ID	first_name	last_name	email	role_id
1	1	Денис	Мамонтов	123@mail.ru	1
2	2	Екатерина	Япрынцева	321@mail.ru	2
3	3	Павел	Петрович	1@mail.ru	3

Рисунок 3 – Таблица “Employee”

Results		Messages	
	ID	type	procent_of_brak
1	1	Тип материала 1	0.10
2	2	Тип материала 2	0.95
3	3	Тип материала 3	0.28
4	4	Тип материала 4	0.55
5	5	Тип материала 5	0.34

Рисунок 4 – Таблица “Material\_type”

Results Messages					
	ID	product_id	partner_id	quantity_of_products	sale_date
1	1	1	1	15500	2023-03-23
2	2	3	1	12350	2023-12-18
3	3	4	1	37400	2024-06-07
4	4	2	2	35000	2022-12-02
5	5	5	2	1250	2023-05-17
6	6	3	2	1000	2024-06-07
7	7	1	2	7550	2024-07-01
8	8	1	3	7250	2023-01-22
9	9	2	3	2500	2024-07-05
10	10	4	4	59050	2023-03-20
11	11	3	4	37200	2024-03-12
12	12	5	4	4500	2024-05-14
13	13	3	5	50000	2023-09-19
14	14	4	5	670000	2023-11-10
15	15	1	5	35000	2024-04-15
16	16	2	5	25000	2024-06-12

Рисунок 5 – Таблица “Partner\_product”

Results Messages									
	ID	type	company_name	director_name	email	phone	legal_address	inn	rating
1	1	ООО	База Строитель	Иванова Александра Ивановна	aleksandraivanova@ml.ru	+79687158466	652050, Кемеровская область, город Юрга, ул. Лесн...	2222455179	5
2	2	ООО	Паркет 29	Петров Василий Петрович	vppetrov@vl.ru	987 123 56 78	164500, Архангельская область, город Северодвинс...	3333888520	7
3	3	ПАО	Стройсервис	Соловьев Андрей Николаевич	ansolovev@st.ru	812 223 32 00	188910, Ленинградская область, город Приморск, ул...	4440391035	7
4	4	ОАО	Ремонт и отделка	Воробьева Екатерина Валерьевна	ekaterina.vorobeva@ml.ru	444 222 33 11	143960, Московская область, город Реутов, ул. Своб...	1111520857	5
5	5	ЗАО	МонтажПро	Степанов Степан Сергеевич	stepanov@stepan.ru	912 888 33 33	309501, Белгородская область, город Старый Оскол...	5552431140	10
6	6	ООО	123	Япрынцев Николай	123@mail.com	+79687158466	Москва	1235434434	5

Рисунок 6 – Таблица “Partners”

Results Messages					
	ID	type_id	name	articul	minPrice
1	1	3	Паркетная доска Ясень темный однополосная 14 мм	8758385	4456,9
2	2	3	Инженерная доска Дуб Французская елка однополосн...	8858958	7330,99
3	3	1	Ламинат Дуб дымчато-белый 33 класс 12 мм	7750282	1799,33
4	4	1	Ламинат Дуб серый 32 класс 8 мм с фаской	7028748	3890,41
5	5	4	Пробковое напольное клеевое покрытие 32 класс 4 мм	5012543	5450,59

Рисунок 7 – Таблица “Product”

Results Messages			
	ID	type	koef_of_type
1	1	Ламинат	2.35
2	2	Массивная доска	5.15
3	3	Паркетная доска	4.34
4	4	Пробковое покрытие	1.50

Рисунок 8 – Таблица “Product\_type”

Results			Messages	
	ID	name		
1	1	Менеджер		
2	2	Пользователь		
3	3	Администратор		

Рисунок 9 – Таблица “Roles”

В листинге 2 представлен скрипт для БД.

## Листинг 2 – Скрипт БД

```
USE [master]
GO
/***** Object:  Database [Mamontov_UP2]      Script Date: 19.12.2024
20:49:03 *****/
CREATE DATABASE [Mamontov_UP2]
    CONTAINMENT = NONE
    ON PRIMARY
    ( NAME = N'Mamontov_UP2_Data', FILENAME =
N'E:\Games\MSSQL15.MSSQLSERVER\MSSQL\DATA\Mamontov_UP2.mdf' , SIZE =
8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )
    LOG ON
    ( NAME = N'Mamontov_UP2_Log', FILENAME =
N'E:\Games\MSSQL15.MSSQLSERVER\MSSQL\DATA\Mamontov_UP2.ldf' , SIZE =
8192KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)
    WITH CATALOG_COLLATION = DATABASE_DEFAULT
GO
ALTER DATABASE [Mamontov_UP2] SET COMPATIBILITY_LEVEL = 150
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [Mamontov_UP2].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [Mamontov_UP2] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [Mamontov_UP2] SET ANSI_NULLS OFF
GO
ALTER DATABASE [Mamontov_UP2] SET ANSI_PADDING OFF
GO
ALTER DATABASE [Mamontov_UP2] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [Mamontov_UP2] SET ARITHABORT OFF
GO
ALTER DATABASE [Mamontov_UP2] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [Mamontov_UP2] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [Mamontov_UP2] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [Mamontov_UP2] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [Mamontov_UP2] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [Mamontov_UP2] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [Mamontov_UP2] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [Mamontov_UP2] SET QUOTED_IDENTIFIER OFF
```

```

GO
ALTER DATABASE [Mamontov_UP2] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [Mamontov_UP2] SET DISABLE_BROKER
GO
ALTER DATABASE [Mamontov_UP2] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [Mamontov_UP2] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [Mamontov_UP2] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [Mamontov_UP2] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [Mamontov_UP2] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [Mamontov_UP2] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [Mamontov_UP2] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [Mamontov_UP2] SET RECOVERY SIMPLE
GO
ALTER DATABASE [Mamontov_UP2] SET MULTI_USER
GO
ALTER DATABASE [Mamontov_UP2] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [Mamontov_UP2] SET DB_CHAINING OFF
GO
ALTER DATABASE [Mamontov_UP2] SET FILESTREAM( NON_TRANSACTED_ACCESS =
OFF )
GO
ALTER DATABASE [Mamontov_UP2] SET TARGET_RECOVERY_TIME = 60 SECONDS
GO
ALTER DATABASE [Mamontov_UP2] SET DELAYED_DURABILITY = DISABLED
GO
ALTER DATABASE [Mamontov_UP2] SET ACCELERATED_DATABASE_RECOVERY = OFF
GO
EXEC sys.sp_db_vardecimal_storage_format N'Mamontov_UP2', N'ON'
GO
ALTER DATABASE [Mamontov_UP2] SET QUERY_STORE = ON
GO
ALTER DATABASE [Mamontov_UP2] SET QUERY_STORE (OPERATION_MODE =
READ_WRITE, CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 30),
DATA_FLUSH_INTERVAL_SECONDS = 900, INTERVAL_LENGTH_MINUTES = 60,
MAX_STORAGE_SIZE_MB = 1000, QUERY_CAPTURE_MODE = AUTO,
SIZE_BASED_CLEANUP_MODE = AUTO, MAX_PLANS_PER_QUERY = 200,
WAIT_STATS_CAPTURE_MODE = ON)
GO
USE [Mamontov_UP2]
GO
/***** Object: Table [dbo].[Employee]      Script Date: 19.12.2024
20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Employee](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [first_name] [varchar](255) NULL,
    [last_name] [varchar](255) NULL,
    [email] [varchar](255) NULL,
    [role_id] [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].[Material_type]      Script Date: 19.12.2024
20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Material_type](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [type] [nvarchar](200) NULL,
    [procent_of_brak] [decimal](10, 2) NULL,
    CONSTRAINT [PK_Material_type] 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].[Partner_product]      Script Date:
19.12.2024 20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Partner_product](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [product_id] [int] NULL,
    [partner_id] [int] NULL,
    [quantity_of_products] [int] NULL,
    [sale_date] [date] NULL,
    CONSTRAINT [PK_Partner_product] 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: 19.12.2024
20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Partners](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [type] [nvarchar](10) NULL,
    [company_name] [nvarchar](255) NULL,
    [director_name] [nvarchar](255) NULL,
    [email] [nvarchar](255) NULL,
    [phone] [nvarchar](20) NULL,
    [legal_address] [nvarchar](255) NULL,
    [inn] [nvarchar](10) NULL,
    [rating] [float] NULL,
    CONSTRAINT [PK__Partners__3214EC27F4FB4FB0] 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].[Product]      Script Date: 19.12.2024
20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Product](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [type_id] [int] NULL,
    [name] [nvarchar](190) NULL,
    [articul] [int] NULL,
    [minPrice] [float] NULL,
    CONSTRAINT [PK_Product] 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].[Product_type]  Script Date: 19.12.2024
20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Product_type](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [type] [nvarchar](100) NULL,
    [koef_of_type] [decimal](10, 2) NULL,
    CONSTRAINT [PK_Product_type] 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].[Roles]      Script Date: 19.12.2024
20:49:03 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Roles](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [name] [nvarchar](100) NULL,
    CONSTRAINT [PK_Roles] 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
ALTER TABLE [dbo].[Employee] WITH CHECK ADD CONSTRAINT
[FK_Employee_Roles] FOREIGN KEY([role_id])
REFERENCES [dbo].[Roles] ([ID])
GO

```

```

ALTER TABLE [dbo].[Employee] CHECK CONSTRAINT [FK_Employee_Roles]
GO
ALTER TABLE [dbo].[Partner_product] WITH CHECK ADD CONSTRAINT
[FK_Partner_product_Partners] FOREIGN KEY([partner_id])
REFERENCES [dbo].[Partners] ([ID])
GO
ALTER TABLE [dbo].[Partner_product] CHECK CONSTRAINT
[FK_Partner_product_Partners]
GO
ALTER TABLE [dbo].[Partner_product] WITH CHECK ADD CONSTRAINT
[FK_Partner_product_Product] FOREIGN KEY([product_id])
REFERENCES [dbo].[Product] ([ID])
GO
ALTER TABLE [dbo].[Partner_product] CHECK CONSTRAINT
[FK_Partner_product_Product]
GO
ALTER TABLE [dbo].[Product] WITH CHECK ADD CONSTRAINT
[FK_Product_Product_type] FOREIGN KEY([type_id])
REFERENCES [dbo].[Product_type] ([ID])
GO
ALTER TABLE [dbo].[Product] CHECK CONSTRAINT [FK_Product_Product_type]
GO
USE [master]
GO
ALTER DATABASE [Mamontov_UP2] SET READ_WRITE
GO

```

При запуске приложения открывается меню с переходом к партнёрам или просмотру истории реализации продукции на соответствующие кнопки. Демонстрация меню изображена на рисунке 10.

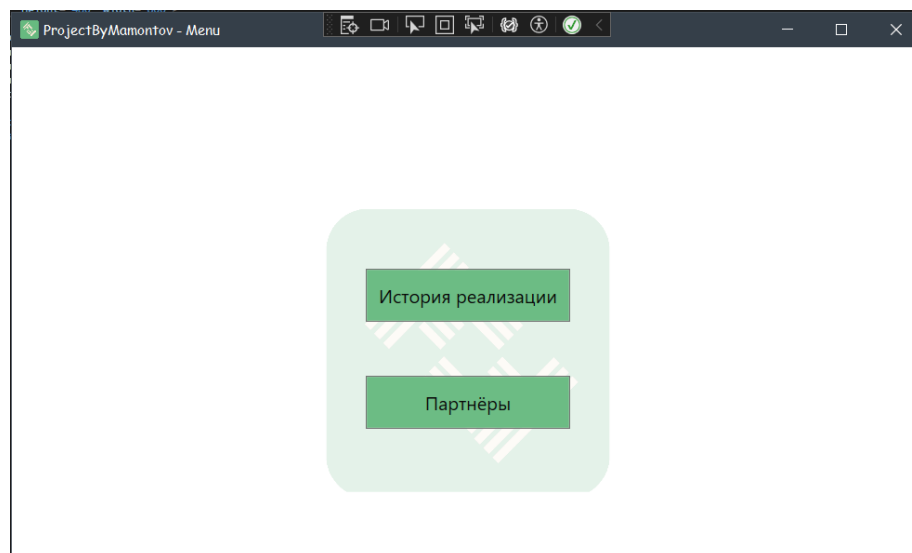


Рисунок 10 – Меню приложения

В приложении реализован выход с подтверждением от пользователя, это представлено на рисунке 11.

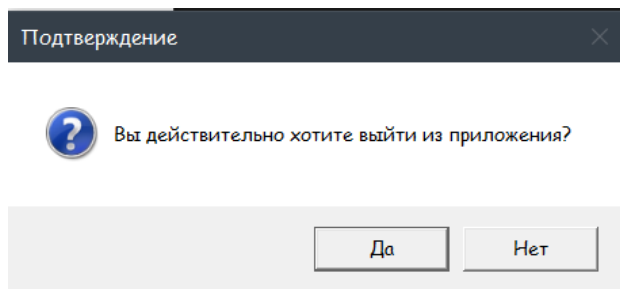


Рисунок 11 – Подтверждение о выходе

В листинге 3 представлена разметка главного окна приложения “MainWindow”

Листинг 3 – Разметка окна “MainWindow”

```
<Window x:Class="UP2.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:UP2"
    mc:Ignorable="d"
    Title="MainWindow" Height="480" Width="800">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="58"></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition Height="24.04"></RowDefinition>
        </Grid.RowDefinitions>

        <Frame x:Name="MainFrame" Grid.Row="1"
NavigationUIVisibility="Hidden" Source="Pages/Menu.xaml"
VerticalAlignment="Center" HorizontalAlignment="Center" Width="800"
Navigated="MainFrame_Navigated" Height="334"
Background="White"></Frame>
        <Button x:Name="BackButton" Grid.Row="0" Margin="20,14,0,9"
Content="Назад" Click="BackButton_Click" HorizontalAlignment="Left"
Width="79" BorderThickness="0"></Button>

    </Grid>
</Window>
```

Листинг 4 – Код для окна “MainWindow”

```
using System;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Navigation;

namespace UP2
{
    public partial class MainWindow : Window
    {
        public MainWindow()
        {
            InitializeComponent();
            this.Closing += MainWindowClosing;
        }
    }
}
```

```

        private void MainWindowClosing(object sender,
System.ComponentModel.CancelEventArgs e)
        {
            MessageBoxResult result = MessageBox.Show("Вы действительно
хотите выйти из приложения?", "Подтверждение", MessageBoxButton.YesNo,
MessageBoxImage.Question);

            if (result == MessageBoxResult.No) e.Cancel = true;
        }
        private void MainFrame_Navigated(object sender,
NavigationEventArgs e)
        {
            if (!(e.Content is Page page)) return;
            this.Title = $"ProjectByMamontov - {page.Title}";

            if (page is Pages.Menu)
                BackButton.Visibility = Visibility.Hidden;
            else
                BackButton.Visibility = Visibility.Visible;
        }
        private void BackButton_Click(object sender, RoutedEventArgs e)
        {
            MainFrame.NavigationService.GoBack();
        }
    }
}

```

На листингах 5-6 представлен код для страницы “Menu”, которая и открывается при запуске приложения.

#### Листинг 5 – Разметка страницы “Menu”

```

<Page x:Class="UP2.Pages.Menu"
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:UP2.Pages"
mc:Ignorable="d"
d:DesignHeight="450" d:DesignWidth="800"
Title="Menu">

    <Grid Opacity="0.97">

        <Grid.RowDefinitions>
            <RowDefinition Height="51*"></RowDefinition>
            <RowDefinition Height="368*"></RowDefinition>
            <RowDefinition Height="31*"></RowDefinition>
        </Grid.RowDefinitions>
        <Image Margin="278,30,278,0" Source="/Мастер пол.png"
Stretch="Fill" Opacity="0.185" Grid.Row="1" Height="248"
VerticalAlignment="Top"/>

        <Button x:Name="Partners_Button" Content="Партнёры"
Grid.Row="1" VerticalAlignment="Top" Height="46"
Click="Partners_Button_Click" HorizontalAlignment="Center" Width="176"
Margin="0,173,0,0"/>
    </Grid>
</Page>

```

```

        <Button x:Name="HistoryOfRealization_Button" Content="История
реализации" Grid.Row="1" Width="176"
Click="HistoryOfRealization_Button_Click" HorizontalAlignment="Center"
Height="46" VerticalAlignment="Top" Margin="0,81,0,0" />

    </Grid>
</Page>

```

## Листинг 6 – Код для страницы “Menu”

```

using System;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Navigation;

namespace UP2.Pages
{
    public partial class Menu : Page
    {
        public Menu()
        {
            InitializeComponent();
        }

        private void HistoryOfRealization_Button_Click(object sender,
RoutedEventArgs e)
        {
            NavigationService.Navigate(new RealizationHistory());
        }

        private void Partners_Button_Click(object sender,
RoutedEventArgs e)
        {
            NavigationService.Navigate(new List_of_partners());
        }
    }
}

```

При нажатии на кнопку “Партнёры” открывается страница со списком всех партнёров. Здесь представлена вся информация о них, а также рассчитывается скидка. Данный список представлен на рисунке и реализован на странице “List\_of\_partners”. Данная страница представлена на рисунке 12.

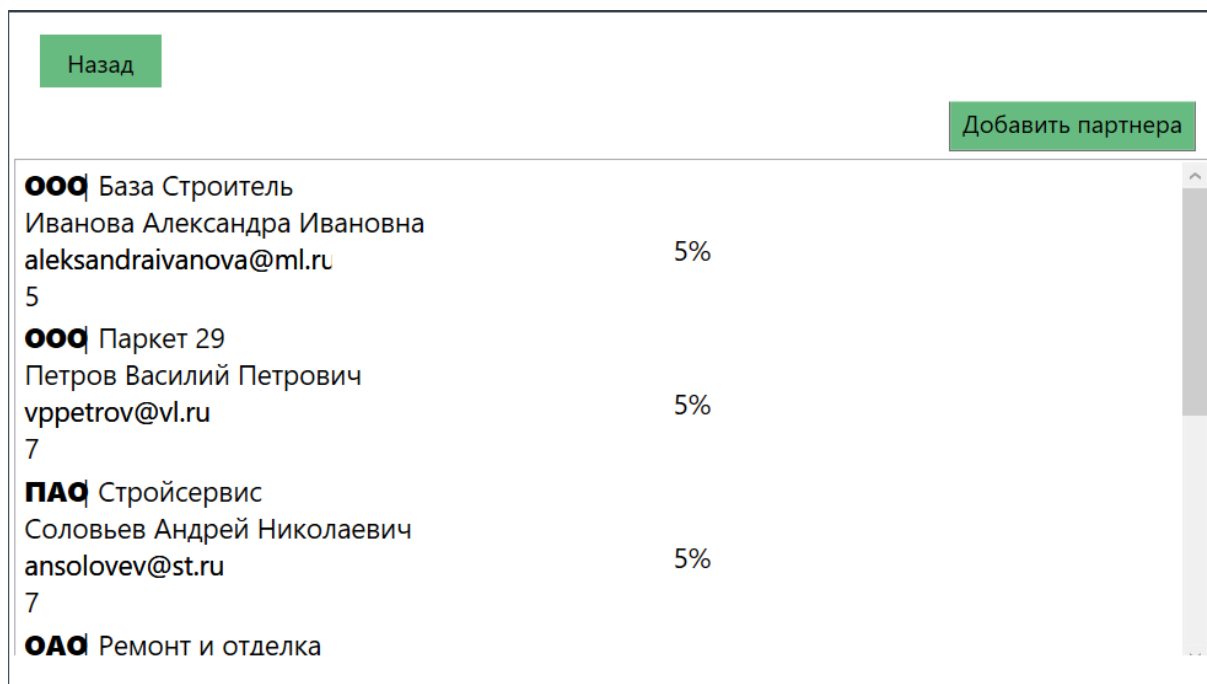


Рисунок 12 – Страница со списком партнёров

На листингах 7-8 представлен код для данной страницы.

#### Листинг 7 – Разметка страницы “List\_of\_partners”

```
<Page x:Name="Is" x:Class="UP2.Pages.List_of_partners"
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:UP2.Pages"
mc:Ignorable="d"
d:DesignHeight="450" d:DesignWidth="800"
Title="List_of_partners" IsVisibleChanged="Is_IsVisibleChanged"
>

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="19*" />
            <RowDefinition Height="131*" />
        </Grid.RowDefinitions>

        <Button ToolTip="Нажмите, чтобы добавить партнера"
Content="Добавить партнера" HorizontalAlignment="Left"
VerticalAlignment="Center" Margin="619,0,0,0"
Click="AddPartnerButton_Click" Height="38" Width="161" />

        <ListView Name="ListUser" Grid.Row="1"
SelectionChanged="ListUser_SelectionChanged_1" Margin="10,0,10,0" >
            <ListView.ItemTemplate>
                <DataTemplate>
                    <Grid>
                        <Grid.RowDefinitions>
                            <RowDefinition
Height="auto"></RowDefinition>
                            <RowDefinition
Height="auto"></RowDefinition>
                            <RowDefinition
Height="auto"></RowDefinition>
                        </Grid.RowDefinitions>
```

```

                                <RowDefinition
Height="auto"></RowDefinition>
                                <RowDefinition
Height="auto"></RowDefinition>
                                <RowDefinition
Height="auto"></RowDefinition>
                                </Grid.RowDefinitions>
                                <Grid.ColumnDefinitions>
                                    <ColumnDefinition
Width="auto"></ColumnDefinition>
                                    <ColumnDefinition
Width="auto"></ColumnDefinition>
                                    <ColumnDefinition
Width="auto"></ColumnDefinition>
                                    <ColumnDefinition
Width="auto"></ColumnDefinition>
                                    <ColumnDefinition
Width="auto"></ColumnDefinition>
                                </Grid.ColumnDefinitions>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Grid.Column="0" Text="{Binding type}"
FontSize="18" FontWeight="Bold" Margin="0,0,133,0"/>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="180" Grid.Column="0"
HorizontalAlignment="Left" FontSize="18" Text="{Binding company_name}"
Margin="48,0,0,0"/>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="10" Grid.Column="0"
HorizontalAlignment="Left" FontSize="18" Text="|" Margin="38,0,0,0"/>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="370" Grid.Row="2"
HorizontalAlignment="Left" FontSize="18" Text="{Binding
director_name}"/>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="200" Grid.Row="3"
HorizontalAlignment="Left" FontSize="18" Text="{Binding email}"/>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="200" Grid.Row="3"
HorizontalAlignment="Left" FontSize="18" Text="{Binding phone}"/>
                                <TextBlock VerticalAlignment="Center"
TextAlignment="Left" Width="200" Grid.Row="4"
HorizontalAlignment="Left" FontSize="18" Text="{Binding rating}"/>

                                <TextBlock VerticalAlignment="Top"
TextAlignment="Right" Width="200" Grid.Row="2"
HorizontalAlignment="Left" FontSize="18" Text="{Binding Discount,
StringFormat={}{0}%}" Margin="247,19,0,0" Grid.ColumnSpan="2"
Grid.RowSpan="2"/>
                                </Grid>
                            </DataTemplate>
                        </ListView.ItemTemplate>
                    </ListView>

                </Grid>
            </Page>

```

## Листинг 8 – Код для страницы “List\_of\_partners”

```

using System;
using System.Collections.Generic;
using System.Linq;

```



```

using System.Security.Cryptography;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;

namespace UP2.Pages
{
    public class PartnerWithDiscount
    {
        public string type { get; set; }
        public string company_name { get; set; }
        public string director_name { get; set; }
        public string email { get; set; }
        public string phone { get; set; }
        public decimal TotalSales { get; set; }
        public string rating { get; set; }
        public decimal Discount { get; set; }

        public void CalculateDiscount()
        {
            if (TotalSales <= 10000) { Discount = 0;}
            else if (TotalSales <= 50000){Discount = 5;}
            else if (TotalSales <= 300000){Discount = 10;}
            else{Discount = 15;}
        }
    }

    public partial class List_of_partners : Page
    {
        private void AddPartnerButton_Click(object sender,
RoutedEventArgs e)
        {
            NavigationService.Navigate(new AddPartner(null));
        }

        public List_of_partners()
        {
            InitializeComponent();
            var currentUsers = Entities.GetContext().Partners.ToList();
            ListUser.ItemsSource = currentUsers;
            var skidkiPachet =
Entities.GetContext().Partner_product.ToList();

            var groupedByPartner = skidkiPachet.GroupBy(x =>
x.Partners);
            var partners = Entities.GetContext().Partners.ToList();

```

```

        var partnerProducts =
Entities.GetContext().Partner_product.ToList();

        var partnerSales = partnerProducts
        .GroupBy(pp => pp.ID)
        .Select(g => new
        {
            PartnerId = g.Key,
            TotalSales = g.Sum(pp => pp.quantity_of_products)
        })
        .ToList();

        var partnersWithDiscount = partners.Select(partner =>
        {

            var totalSales = partnerSales
            .FirstOrDefault(ps => ps.PartnerId ==
partner.ID)?.TotalSales ?? 0;

            var partnerWithDiscount = new PartnerWithDiscount
            {
                type = partner.type,
                company_name = partner.company_name,
                director_name = partner.director_name,
                email = partner.email,
                phone = partner.email,
                TotalSales = totalSales,
                rating = partner.rating.ToString(),
            };

            partnerWithDiscount.CalculateDiscount();

            return partnerWithDiscount;
        }).ToList();

        ListUser.ItemsSource = partnersWithDiscount;
        ListUser.Items.Refresh();

    }

    private void ListUser_SelectionChanged_1(object sender,
SelectionChangedEventArgs e)
    {
        if (ListUser.SelectedItem != null)
        {
            var selectedPartner =
(PartnerWithDiscount)ListUser.SelectedItem;
            var partner = Entities.GetContext().Partners.First(p =>
p.company_name == selectedPartner.company_name);
            NavigationService.Navigate(new AddPartner(partner));
        }
    }

    private void Is_IsVisibleChanged(object sender,
DependencyPropertyChangedEventArgs e)
    {
        if (Visibility == Visibility.Visible)
        {
            var currentUsers =
Entities.GetContext().Partners.ToList();
            ListUser.ItemsSource = currentUsers;

```

```

        var skidkiPachet =
Entities.GetContext().Partner_product.ToList();

        var groupedByPartner = skidkiPachet.GroupBy(x =>
x.Partners);
        var partners = Entities.GetContext().Partners.ToList();

        var partnerProducts =
Entities.GetContext().Partner_product.ToList();

        var partnerSales = partnerProducts
        .GroupBy(pp => pp.ID)
        .Select(g => new
        {
            PartnerId = g.Key,
            TotalSales = g.Sum(pp =>
pp.quantity_of_products)
        })
        .ToList();

Entities.GetContext().ChangeTracker.Entries().ToList().ForEach(x =>
x.Reload());

        var partnersWithDiscount = partners.Select(partner =>
        {

            var totalSales = partnerSales
            .FirstOrDefault(ps => ps.PartnerId ==
partner.ID)?.TotalSales ?? 0;

            var partnerWithDiscount = new PartnerWithDiscount
            {
                type = partner.type,
                company_name = partner.company_name,
                director_name = partner.director_name,
                email = partner.email,
                phone = partner.email,
                TotalSales = totalSales,
                rating = partner.rating.ToString(),
            };

            partnerWithDiscount.CalculateDiscount();

            return partnerWithDiscount;
        }).ToList();

        ListUser.ItemsSource = partnersWithDiscount;
        ListUser.Items.Refresh();
    }
}
}
}

```

При нажатии на партнера открывается страница редактирования. Есть кнопка для добавления нового партнёра. Присутствуют различные проверка, как на пустые поля при попытке сохранить данные, так и на ввод данных. К примеру таких относится проверки на ввод значение в поле ФИО директора, поле рейтинга, телефона и т.д. На рисунке 13 представлена возможность

редактирования данных о партнёре, а на рисунке 14 – страница при его создании.

The screenshot shows a web application window titled "ProjectByMamontov - AddPartner". In the top-left corner, there is a green button labeled "Назад". The form contains the following fields:

- Тип: A dropdown menu with "ООО" selected.
- Наименование: A text input field containing "База Строитель".
- ФИО директора: A text input field containing "Иванова Александи".
- Рейтинг: A text input field containing "5".
- Контактный телефон: A text input field containing "+79687158466".
- Почта: A text input field containing "aleksandraivanova@".
- Адрес: A text input field containing "652050, Кемеровск".
- ИНН: A text input field containing "2222455179".

At the bottom of the form, there are two green buttons: "Отмена\Назад" on the left and "Сохранить" on the right.

Рисунок 13– Редактирование существующего партнёра

The screenshot shows the same "ProjectByMamontov - AddPartner" window, but with empty input fields for creating a new partner. The "Назад" button is still present in the top-left corner. The form fields are:

- Тип: A dropdown menu.
- Наименование: An empty text input field.
- ФИО директора: An empty text input field.
- Рейтинг: An empty text input field.
- Контактный телефон: An empty text input field.
- Почта: An empty text input field.
- Адрес: An empty text input field.
- ИНН: An empty text input field.

At the bottom, the "Отмена\Назад" and "Сохранить" buttons are also present.

Рисунок 14 – Добавление нового партнёра

На рисунке 15 представлен пример ошибки при неправильно введённом значении.

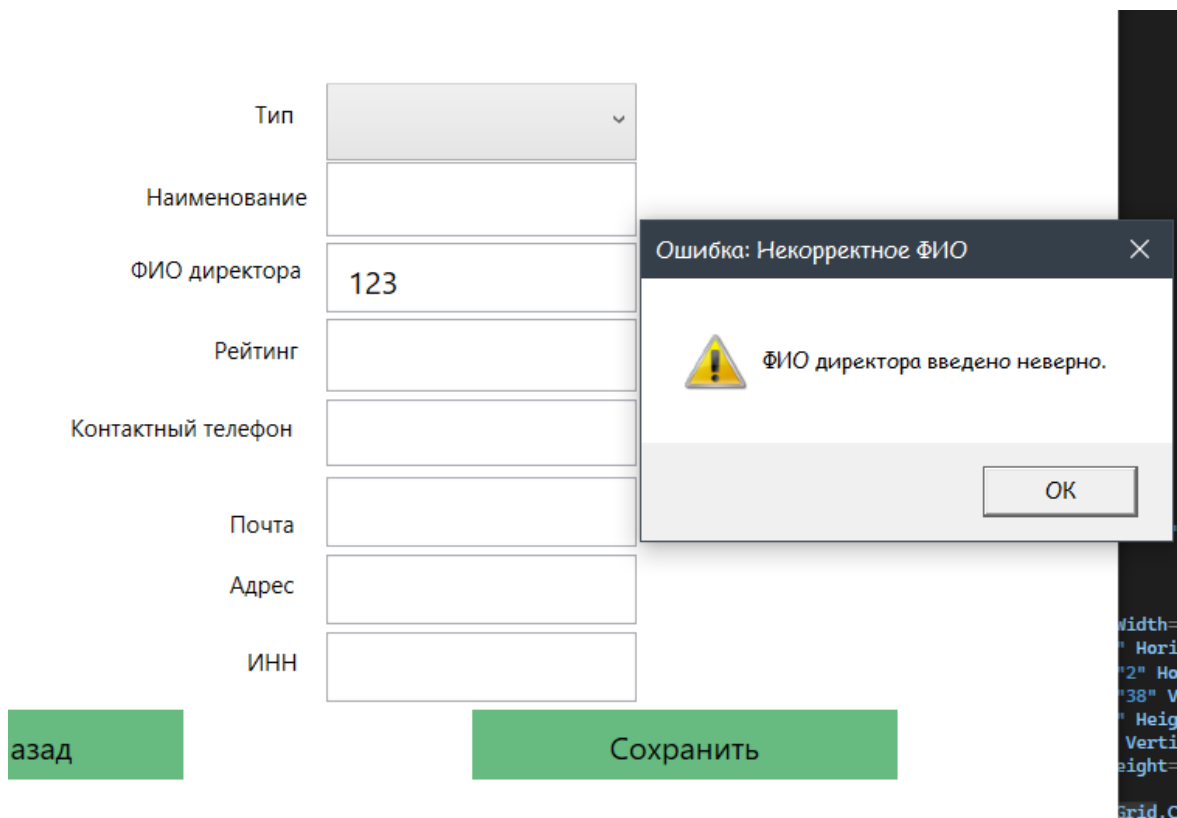


Рисунок 15 – Проверка на ввод

В листингах 9-10 представлен код для страницы “AddPartner”. Именно эта страница отвечает за добавление или редактирование данных.

#### Листинг 9 – Разметка страницы “AddPartner”

```
<Page x:Class="UP2.Pages.AddPartner"
      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:UP2.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="AddPartner">

    <Grid Margin="0,10,0,10">
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="381*"></ColumnDefinition>
            <ColumnDefinition Width="419*"></ColumnDefinition>
        </Grid.ColumnDefinitions>
        <Grid.RowDefinitions>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
            <RowDefinition></RowDefinition>
        </Grid.RowDefinitions>
    </Grid>
</Page>
```

```

        <TextBlock Text="Тип" Margin="348,0,0,0" Height="24"
VerticalAlignment="Center" HorizontalAlignment="Left" Width="27"/>
        <TextBlock Text="Наименование" Margin="292,0,0,0" Grid.Row="1"
HorizontalAlignment="Left" Width="89" Height="20"
VerticalAlignment="Center"/>
        <TextBlock Text="ФИО директора" Margin="284,0,0,0" Grid.Row="2"
HorizontalAlignment="Left" Width="91" Height="24"
VerticalAlignment="Center"/>
        <TextBlock Text="Рейтинг" Margin="327,0,0,0" Grid.Row="3"
RenderTransformOrigin="0.517,0.348" Height="22"
VerticalAlignment="Center" HorizontalAlignment="Left" Width="48"/>
        <TextBlock Text="Контактный телефон" Margin="253,0,0,0"
Grid.Row="4" HorizontalAlignment="Left" Width="122" Height="22"
VerticalAlignment="Center"/>
        <TextBlock Text="Почта" Margin="335,18,0,0" Grid.Row="5"
Height="22" VerticalAlignment="Top" HorizontalAlignment="Left"
Width="36"/>

        <ComboBox VerticalContentAlignment="Center" ToolTip="Выбор
типа" Text="{Binding type}" x:Name="TypeComboBox" Margin="4,0,0,0"
HorizontalAlignment="Left" Width="160" Height="40"
VerticalAlignment="Center" Grid.Column="1"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="Впишите
наименование" Text="{Binding company_name}" x:Name="NameTextBox"
Margin="4,0,0,0" Grid.Row="1" HorizontalAlignment="Left" Width="160"
Height="38" VerticalAlignment="Center" Grid.Column="1"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="Впишите
директора" Text="{Binding director_name}" x:Name="DirectorTextBox"
Margin="4,0,0,0" Grid.Row="2" HorizontalAlignment="Left" Width="160"
Height="36" VerticalAlignment="Center" Grid.Column="1"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="Впишите
рейтинг" Text="{Binding rating}" x:Name="RatingTextBox"
Margin="4,0,0,0" Grid.Row="3" Height="38" VerticalAlignment="Center"
HorizontalAlignment="Left" Width="160" Grid.Column="1"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="Впишите
контактный телефон" Text="{Binding phone}" x:Name="PhoneTextBox"
Margin="4,0,0,0" Grid.Row="4" Height="34" VerticalAlignment="Center"
HorizontalAlignment="Left" Width="160" Grid.Column="1"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="Впишите
почту" Text="{Binding email}" x:Name="EmailTextBox" Margin="4,0,0,0"
Grid.Row="5" Height="36" VerticalAlignment="Center"
HorizontalAlignment="Left" Width="160" Grid.Column="1"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="Впишите
адрес" Text="{Binding legal_address}" x:Name="AddressTextBox"
Margin="4,0,0,0" Grid.Row="6" Height="36" VerticalAlignment="Center"
HorizontalAlignment="Left" Width="160" Grid.Column="1"/>

        <Button BorderThickness="0" ToolTip="Кнопка для отмены
действий/Возврат назад" Content="Отмена\Назад" x:Name="BackButton"
Grid.Row="8" Margin="92,0,0,0" Grid.Column="0" Click="BackButtonClick"
Height="36" VerticalAlignment="Center" HorizontalAlignment="Left"
Width="219"></Button>

        <Button ToolTip="Кнопка для сохранения" Content="Сохранить"
x:Name="SaveButton" Grid.Row="8" Margin="79,0,0,0" Grid.Column="1"
Height="36" VerticalAlignment="Center" BorderThickness="0"
HorizontalAlignment="Left" Width="219" Click="SaveButton_Click"/>
        <TextBox VerticalContentAlignment="Center" ToolTip="ИНН"
Text="{Binding inn}" x:Name="innTextBox" Margin="4,0,0,0" Grid.Row="7"
Height="36" VerticalAlignment="Center" HorizontalAlignment="Left"
Width="160" Grid.Column="1" />
        <TextBlock Text="Адрес" Margin="335,0,0,0" Grid.Row="6"
Height="22" VerticalAlignment="Center" HorizontalAlignment="Left"
Width="36"/>

```

```

        <TextBlock Text="ИИИ" Margin="344,0,0,0" Grid.Row="7"
Height="22" VerticalAlignment="Center" HorizontalAlignment="Left"
Width="27"/>
    </Grid>
</Page>

```

## Листинг 10 – Код для страницы “AddPartner”

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Text.RegularExpressions;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;

namespace UP2.Pages
{
    /// <summary>
    /// Логика взаимодействия для AddPartner.xaml
    /// </summary>
    public partial class AddPartner : Page
    {
        private Partners _currentPartner = new Partners();
        private static readonly Regex FIOregex = new Regex(@"^[А-ЯЁ][а-яё]+(?: [А-ЯЁ][а-яё]+)*$");
        private static readonly Regex EmailRegex = new
Regex(@"^[^@\s]+@^[^@\s]+\.[^@\s]+$");
        private static readonly Regex INNRegex = new
Regex(@"^\d{10}$");
        private static readonly Regex PhoneNumberRegex = new
Regex(@"^\+?[1-9]\d{9,14}$");
        public AddPartner(Partners selectedPartner)
        {
            InitializeComponent();
            TypeComboBox.ItemsSource = new[] { "ЗАО", "ООО", "ПАО",
"ОАО" }; // Здесь укажите реальные типы
            if (selectedPartner != null)
            {
                _currentPartner = selectedPartner;
                if (_currentPartner.type.ToString() == "ЗАО")
                {
                    TypeComboBox.SelectedItem = TypeComboBox.Items[0];
                }
                if (_currentPartner.type.ToString() == "ООО")
                {
                    TypeComboBox.SelectedItem = TypeComboBox.Items[1];
                }
                if (_currentPartner.type.ToString() == "ПАО")
                {
                    TypeComboBox.SelectedItem = TypeComboBox.Items[2];
                }
                else { TypeComboBox.SelectedItem =
TypeComboBox.Items[3]; }
            }
        }
    }
}

```

```

    }
    DataContext = _currentPartner;

    DataContext = _currentPartner;
}

private void BackButtonClick(object sender, RoutedEventArgs e)
{
    NavigationService.GoBack();
}

private void SaveButton_Click(object sender, RoutedEventArgs e)
{
    StringBuilder errors = new StringBuilder();

    if
(string.IsNullOrEmpty(_currentPartner.company_name))
        errors.AppendLine("Укажите наименование!");
    if (string.IsNullOrEmpty(_currentPartner.phone))
        errors.AppendLine("Укажите телефон!");
    if (string.IsNullOrEmpty(_currentPartner.email))
        errors.AppendLine("Укажите почту!");
    if
(string.IsNullOrEmpty(_currentPartner.director_name))
        errors.AppendLine("Укажите ФИО директора!");
    if
(string.IsNullOrEmpty(_currentPartner.legal_address))
        errors.AppendLine("Укажите адрес");

    if (!FIOregex.IsMatch(DirectorTextBox.Text))
    {
        MessageBox.Show("ФИО директора введено неверно.",
"Ошибка: Некорректное ФИО", MessageBoxButton.OK,
MessageBoxImage.Warning);
        return;
    }

    if (!PhoneNumberRegex.IsMatch(PhoneTextBox.Text))
    {
        MessageBox.Show("Номер телефона введён в неверном
формате.", "Ошибка: Некорректный номер телефона", MessageBoxButton.OK,
MessageBoxImage.Warning);
        return;
    }

    if (!EmailRegex.IsMatch(EmailTextBox.Text))
    {
        MessageBox.Show("Электронная почта введена
неверно.", "Ошибка: Некорректная электронная почта",
MessageBoxButton.OK, MessageBoxImage.Warning);
        return;
    }

    if (!INNRegex.IsMatch(innTextBox.Text))
    {
        MessageBox.Show("ИНН введён неверно.", "Ошибка:
Некорректный ИНН", MessageBoxButton.OK, MessageBoxImage.Warning);
        return;
    }
}

```



```

        if (!int.TryParse(RatingTextBox.Text, out int
ratingValue) || ratingValue < 1 || ratingValue > 10)
        {
            MessageBox.Show("Рейтинг должен быть числом от 1 до
10.", "Ошибка: Некорректный ввод", MessageBoxButton.OK,
MessageBoxImage.Warning);
            return;
        }

        if (errors.Length > 0)
        {
            MessageBox.Show(errors.ToString());
            return;
        }
        if (_currentPartner.ID == 0)
Entities.GetContext().Partners.Add(_currentPartner);
        try
        {
            Entities.GetContext().SaveChanges();
            MessageBox.Show("Данные успешно сохранены!");
        }

        catch (Exception ex)
        {
            MessageBox.Show(ex.Message.ToString());
        }
        NavigationService.GoBack();
    }
}
}

```

При нажатии в главном меню на кнопку “История реализации” откроется страница “RealizationHistory” с историей реализации продукции (рисунок 16).

Назад

Продукция	Партнер	Количество	Дата продажи
Паркетная доска Ясень темный с	База Строитель	15500	23.03.2023
Ламинат Дуб дымчато-белый 33	База Строитель	12350	18.12.2023
Ламинат Дуб серый 32 класс 8 м	База Строитель	37400	07.06.2024
Инженерная доска Дуб Француз	Паркет 29	35000	02.12.2022
Пробковое напольное клеевое п	Паркет 29	1250	17.05.2023
Ламинат Дуб дымчато-белый 33	Паркет 29	1000	07.06.2024
Паркетная доска Ясень темный с	Паркет 29	7550	01.07.2024
Паркетная доска Ясень темный с	Стройсервис	7250	22.01.2023
Инженерная доска Дуб Француз	Стройсервис	2500	05.07.2024
Ламинат Дуб серый 32 класс 8 м	Ремонт и отделка	59050	20.03.2023
Ламинат Дуб дымчато-белый 33	Ремонт и отделка	37200	12.03.2024
Пробковое напольное клеевое п	Ремонт и отделка	4500	14.05.2024
Ламинат Дуб дымчато-белый 33	МонтажПро	50000	19.09.2023

Рисунок 16 – Страница с историей реализации

Присутствует и возможность отбора по партнёру (рисунок 17).

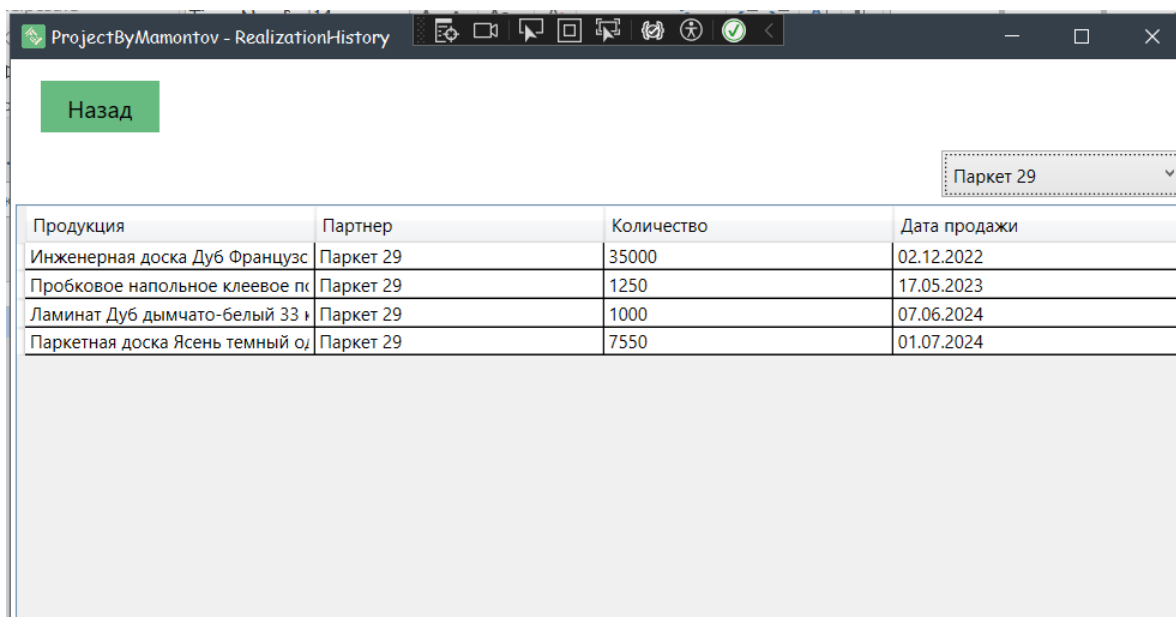


Рисунок 17 – Выбор конкретного партнёра из списка

### Листинг 11 – Разметка страницы «просмотр истории реализации»

```
<Page x:Class="UP2.Pages.RealizationHistory"
      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:UP2.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="RealizationHistory">

    <Grid Background="White">
        <Grid.RowDefinitions>
            <RowDefinition Height="377*"/>
            <RowDefinition Height="50*"/>
        </Grid.RowDefinitions>
        <ComboBox x:Name="PartnerComboBox" Height="31"
Margin="628,14,0,0" Grid.Row="0" VerticalAlignment="Top"
SelectionChanged="PartnerComboBox_SelectionChanged"
HorizontalAlignment="Left" Width="162" FontSize="12" >
            <ComboBox.ItemTemplate>
                <DataTemplate>
                    <TextBlock Text="{Binding company_name}"/>
                </DataTemplate>
            </ComboBox.ItemTemplate>
        </ComboBox>

        <DataGrid x:Name="HistoryDataGrid" Margin="10,50,10,10"
AutoGenerateColumns="False" IsReadOnly="True">
            <DataGrid.Columns>
                <DataGridTextColumn Header="Продукция" Width="*"
Binding="{Binding ProductName}"/>
                <DataGridTextColumn Header="Партнер" Width="*"
Binding="{Binding PartnerName}"/>
            </DataGrid.Columns>
        </DataGrid>
    </Grid>
</Page>
```

```

        <DataGridTextColumn Header="Количество" Width="*"
Binding="{Binding quantity_of_products}"/>
        <DataGridTextColumn Header="Дата продажи" Width="*"
Binding="{Binding sale_date, StringFormat='dd.MM.yyyy'}"/>
    </DataGrid.Columns>
</DataGrid>
</Grid>
</Page>

```

## Листинг 12 – Код для страницы «просмотр истории реализации»

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;

namespace UP2.Pages
{
    /// <summary>
    /// Логика взаимодействия для RealizationHistory.xaml
    /// </summary>
    public partial class RealizationHistory : Page
    {
        public RealizationHistory()
        {
            InitializeComponent();
            LoadPartners();
            LoadHistory();
        }

        private void LoadPartners()
        {
            var partners = Entities.GetContext().Partners.Select(p =>
new
            {
                p.ID,
                p.company_name
            }).ToList();

            PartnerComboBox.ItemsSource = partners;
            PartnerComboBox.SelectedValuePath = "ID";
        }

        private void LoadHistory(int? partnerId = null)
        {
            var query =
Entities.GetContext().Partner_product.AsQueryable();

            if (partnerId.HasValue)

```

```

        {
            query = query.Where(pp => pp.partner_id ==
partnerId.Value);
        }

        var partnerProducts = query.Select(pp => new
        {
            ProductName = pp.Product.name,
            PartnerName = pp.Partners.company_name,
            pp.quantity_of_products,
            pp.sale_date
        }).ToList();

        if (partnerProducts.Count == 0)
        {
            MessageBox.Show("Данные не найдены.");
        }
        else
        {
            HistoryDataGrid.ItemsSource = partnerProducts;
        }
    }

    private void PartnerComboBox_SelectionChanged(object sender,
SelectionChangedEventArgs e)
    {
        if (PartnerComboBox.SelectedValue != null)
        {
            int partnerId = (int)PartnerComboBox.SelectedValue;
            LoadHistory(partnerId);
        }
        else
        {
            LoadHistory();
        }
    }
}

```

Также для стилизации приложения был создан словарь стилей, который представлен в листинге 13.

### Листинг 13 – Словарь стилей

```

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

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">
    <Style
        TargetType="Button">
        <Setter Property="Height" Value="35"/>
        <Setter Property="Padding" Value="8"/>
        <Setter Property="Margin" Value="12"/>
        <Setter Property="FontFamily" Value="Segoe UI"/>
        <Setter Property="FontSize" Value="12pt"/>
        <Setter Property="Background" Value="#67BA80"/>
        <Setter Property="Foreground" Value="Black"/>
    </Style>
    <Style TargetType="TextBox">
        <Setter Property="Height" Value="35"/>
    </Style>

```

```

        <Setter Property="Padding" Value="8"/>
        <Setter Property="Margin" Value="12"/>
        <Setter Property="FontFamily" Value="Segoe UI"/>
        <Setter Property="FontSize" Value="12pt"/>
        <Setter Property="Background" Value="White"/>
        <Setter Property="Foreground" Value="Black"/>
    </Style>

    <Style TargetType="ComboBox">
        <Setter Property="Height" Value="40"/>
        <Setter Property="Padding" Value="8"/>
        <Setter Property="Margin" Value="12"/>
        <Setter Property="FontFamily" Value="Segoe UI"/>
        <Setter Property="FontSize" Value="12pt"/>
        <Setter Property="Background" Value="#67BA80"/>
        <Setter Property="Foreground" Value="Black"/>
    </Style>
    <Style TargetType="TextBlock">
        <Setter Property="FontFamily" Value="Segoe UI"/>
    </Style>
    <Style TargetType="Window">
        <Setter Property="Background" Value="#F4E8D3"/></Setter>
    </Style>
</ResourceDictionary>

```

Для метода расчета количества материала, требуемого для производства продукции, был создан отдельный проект Function. Листинг самого метода представлен в листинге 14.

#### Листинг 14 – Метод для расчёта количества материала

```

using System;
using System.Collections.Generic;
using System.Data.Entity;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Function
{
    internal class Function
    {
        public int CalculateMaterial(
            int productId,
            int materialTypeId,
            int productCount,
            double parameter1,
            double parameter2)
        {
            if (productId <= 0 || materialTypeId <= 0 ||
productCount <= 0 || parameter1 <= 0 || parameter2 <= 0)
            {
                return -1;
            }
            var productType = getProductType(productId);
            if (productType == null)
            {
                return -1;
            }
            var materialType = getMaterialType(materialTypeId);

```

```

        if (materialType == null)
        {
            return -1;
        }
        try
        {
            double materialPerUnit = parameter1 * parameter2 *
Convert.ToDouble(productType.koef_of_type);
            double totalMaterial = materialPerUnit *
productCount;
            double defectMultiplier = 1 +
(double)materialType.procent_of_brak;
            totalMaterial *= defectMultiplier;
            return (int)Math.Ceiling(totalMaterial);
        }
        catch
        {
            return -1;
        }
    }

    private Product_type getProductType(int id)
    {
        using (var context = new Entities())
        {
            return context.Product_type.FirstOrDefault(ch =>
ch.ID == id);
        }
    }

    private Material_type getMaterialType(int id)
    {
        using (var context = new Entities())
        {
            return context.Material_type.FirstOrDefault(ch =>
ch.ID == id);
        }
    }
}

```

Тестовые сценарии для программы расположены в отдельном word файле.