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.01 Разработка модулей программного обеспечения для компьютерных систем
        (код)   (наименование)

Междисциплинарный курс МДК.01.02 Поддержка и тестирование программных модулей
        (код)   (наименование)

Выполнил:

Студент 4 курса 4ИСИП-422 учебной группы
     (номер)   (номер)

_____   М.А. Желтобрюх
    (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)
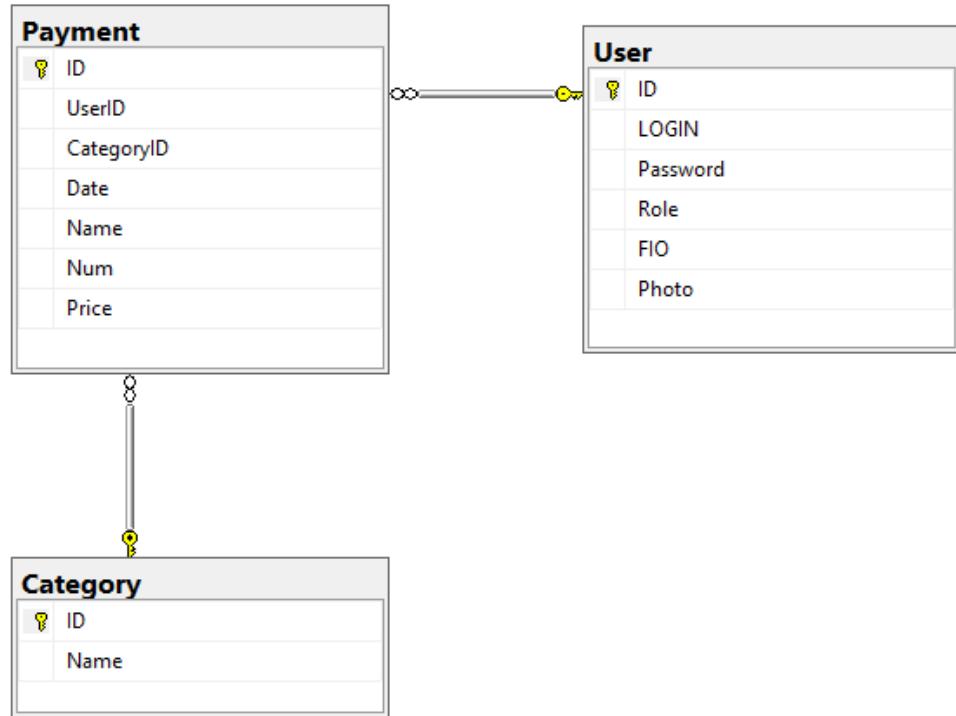
Москва – 2025

Перечень работ, выполненных в ходе учебной практики

№ п/п	Виды работ	Оценка
1	Разработка программного модуля многооконного интерфейса	Отлично
2	Тестирование программного модуля доступа к БД	Отлично

Ссылка на git

Листинг №1 – Скрипт БД

```
USE [master]
GO
/****** Object:  Database [Zheltobryukh_DB_Payments]    Script Date:
06.11.2025 11:17:38 ******/
CREATE DATABASE [Zheltobryukh_DB_Payments]
 CONTAINMENT = NONE
 ON   PRIMARY
( NAME = N'Zheltobryukh_DB_Payments', FILENAME = N'C:\Program
Files\Microsoft SQL
Server\MSSQL17.MSSQLSERVER\MSSQL\DATA\Zheltobryukh_DB_Payments.mdf' , SIZE =
8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )
 LOG ON
( NAME = N'Zheltobryukh_DB_Payments_log', FILENAME = N'C:\Program
Files\Microsoft SQL
Server\MSSQL17.MSSQLSERVER\MSSQL\DATA\Zheltobryukh_DB_Payments_log.ldf' ,
SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
 WITH CATALOG_COLLATION = DATABASE_DEFAULT, LEDGER = OFF
```

```
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET COMPATIBILITY_LEVEL = 160
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [Zheltobryukh_DB_Payments].[dbo].[sp_fulltext_database] @action =
'enable'
end
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ANSI_NULLS OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ANSI_PADDING OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ARITHABORT OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET CURSOR_DEFAULT  GLOBAL
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET  DISABLE_BROKER
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET AUTO_UPDATE_STATISTICS_ASYNC
OFF
GO
```

```sql
ALTER DATABASE [Zheltobryukh_DB_Payments] SET DATE_CORRELATION_OPTIMIZATION
OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET PARAMETERIZATION SIMPLE
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET RECOVERY FULL
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET  MULTI_USER
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET DB_CHAINING OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET FILESTREAM(
NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET TARGET_RECOVERY_TIME = 60
SECONDS
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET DELAYED_DURABILITY = DISABLED
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET ACCELERATED_DATABASE_RECOVERY
= OFF
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET QUERY_STORE = ON
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] 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 [Zheltobryukh_DB_Payments]
```

```
GO
/****** Object:  Table [dbo].[Category]    Script Date: 06.11.2025 11:17:38
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Category](
      [ID] [int] IDENTITY(1,1) NOT NULL,
      [Name] [varchar](200) NOT NULL,
 CONSTRAINT [PK_Category] 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].[Payment]    Script Date: 06.11.2025 11:17:38
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Payment](
      [ID] [int] IDENTITY(1,1) NOT NULL,
      [UserID] [int] NOT NULL,
      [CategoryID] [int] NOT NULL,
      [Date] [date] NOT NULL,
      [Name] [nvarchar](50) NOT NULL,
      [Num] [decimal](18, 0) NOT NULL,
      [Price] [decimal](18, 0) NOT NULL,
 CONSTRAINT [PK__Payment__3214EC277DBFCCB9] 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].[User]    Script Date: 06.11.2025 11:17:38
******/
```

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[User](
      [ID] [int] IDENTITY(1,1) NOT NULL,
      [LOGIN] [varchar](max) NOT NULL,
      [Password] [varchar](max) NOT NULL,
      [Role] [varchar](50) NULL,
      [FIO] [varchar](max) NULL,
      [Photo] [varchar](max) NULL,
 CONSTRAINT [PK_User] 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] TEXTIMAGE_ON [PRIMARY]
GO
SET IDENTITY_INSERT [dbo].[Category] ON
GO
INSERT [dbo].[Category] ([ID], [Name]) VALUES (1, N'Вахо')
GO
INSERT [dbo].[Category] ([ID], [Name]) VALUES (2, N'Фрукты')
GO
INSERT [dbo].[Category] ([ID], [Name]) VALUES (3, N'Овощи')
GO
INSERT [dbo].[Category] ([ID], [Name]) VALUES (4, N'Патроны')
GO
SET IDENTITY_INSERT [dbo].[Category] OFF
GO
SET IDENTITY_INSERT [dbo].[Payment] ON
GO
INSERT [dbo].[Payment] ([ID], [UserID], [CategoryID], [Date], [Name], [Num],
[Price]) VALUES (1, 1, 1, CAST(N'2025-03-05' AS Date), N'2', CAST(1000 AS
Decimal(18, 0)), CAST(100 AS Decimal(18, 0)))
GO
INSERT [dbo].[Payment] ([ID], [UserID], [CategoryID], [Date], [Name], [Num],
[Price]) VALUES (2, 1, 2, CAST(N'2024-08-10' AS Date), N'Яблоко', CAST(10 AS
Decimal(18, 0)), CAST(100 AS Decimal(18, 0)))
GO
```

```sql
INSERT [dbo].[Payment] ([ID], [UserID], [CategoryID], [Date], [Name], [Num],
[Price]) VALUES (3, 2, 3, CAST(N'2025-01-01' AS Date), N'Помидора', CAST(15
AS Decimal(18, 0)), CAST(120 AS Decimal(18, 0)))
GO
INSERT [dbo].[Payment] ([ID], [UserID], [CategoryID], [Date], [Name], [Num],
[Price]) VALUES (4, 4, 4, CAST(N'2023-02-09' AS Date), N'Оплата патрон',
CAST(1000 AS Decimal(18, 0)), CAST(10 AS Decimal(18, 0)))
GO
SET IDENTITY_INSERT [dbo].[Payment] OFF
GO
SET IDENTITY_INSERT [dbo].[User] ON
GO
INSERT [dbo].[User] ([ID], [LOGIN], [Password], [Role], [FIO], [Photo])
VALUES (1, N'user1',
N'D785D63511A645A24875A109E0EF1DA6560DD94D149B6734949A96556CB3449F',
N'User', N'Миргасимов Расим Амирович', NULL)
GO
INSERT [dbo].[User] ([ID], [LOGIN], [Password], [Role], [FIO], [Photo])
VALUES (2, N'adminrasim',
N'D8B736FE94FA63827B4539B50CE891773C62B77021D8139617FD8107445E8806',
N'Admin', N'Джабаров Расим Админович', NULL)
GO
INSERT [dbo].[User] ([ID], [LOGIN], [Password], [Role], [FIO], [Photo])
VALUES (4, N'user2',
N'01DA2F16F6BD99F056D0C4ACE5B9E0B7C6E4CC2E810D69FFC0E002083898873D',
N'User', N'Желтобрюх Заяц Вахович', NULL)
GO
INSERT [dbo].[User] ([ID], [LOGIN], [Password], [Role], [FIO], [Photo])
VALUES (6, N'test',
N'ECD71870D1963316A97E3AC3408C9835AD8CF0F3C1BC703527C30265534F75AE',
N'User', N'ыфв', NULL)
GO
SET IDENTITY_INSERT [dbo].[User] OFF
GO
ALTER TABLE [dbo].[Payment]  WITH CHECK ADD  CONSTRAINT
[FK_Payment_Category] FOREIGN KEY([CategoryID])
REFERENCES [dbo].[Category] ([ID])
GO
ALTER TABLE [dbo].[Payment] CHECK CONSTRAINT [FK_Payment_Category]
GO
ALTER TABLE [dbo].[Payment]  WITH CHECK ADD  CONSTRAINT [FK_Payment_User]
FOREIGN KEY([UserID])
```

```
REFERENCES [dbo].[User] ([ID])
GO
ALTER TABLE [dbo].[Payment] CHECK CONSTRAINT [FK_Payment_User]
GO
USE [master]
GO
ALTER DATABASE [Zheltobryukh_DB_Payments] SET  READ_WRITE
GO
```

Рисунок 1

	Имя столбца	Тип данных	Разрешить ...
🔑	ID	int	☐
	Name	varchar(200)	☐
▶			☐

Рисунок 2

	Имя столбца	Тип данных	Разрешить ...
🔑	ID	int	☐
	LOGIN	varchar(MAX)	☐
	Password	varchar(MAX)	☐
	Role	varchar(50)	☑
	FIO	varchar(MAX)	☑
	Photo	varchar(MAX)	☑
▶			☐

Рисунок 3

	Имя столбца	Тип данных	Разрешить
🔑	ID	int	☐
	UserID	int	☐
	CategoryID	int	☐
	Date	date	☐
	Name	nvarchar(50)	☐
	Num	decimal(18, 0)	☐
	Price	decimal(18, 0)	☐

Рисунок 4

9

Data Dictionary				
Шаблон				
Category				
KEY	FIELD NAME	DATA TYPE / FIELD SIZE	REQUIRED?	NOTES
PK	ID	INT	Y	Auto increment
	Name	VARCHAR(MAX)	Y	
Payment				
KEY	FIELD NAME	DATA TYPE / FIELD SIZE	REQUIRED?	NOTES
PK	ID	INT	Y	Auto increment
FK	UserID	INT	Y	Foreign Key to User
FK	CategoryID	INT	Y	Foreign Key to Category
	Date	DATE	Y	
	Name	NVARCHAR(50)	Y	
	Num	DECIMAL(18,0)	Y	
	Price	DECIMAL(18,0)	Y	
User				
KEY	FIELD NAME	DATA TYPE / FIELD SIZE	REQUIRED?	NOTES
PK	ID	INT	Y	Auto increment
	Login	VARCHAR(MAX)	Y	
	Password	VARCHAR(MAX)	Y	
	Role	VARCHAR(50)	N	
	FIO	VARCHAR(MAX)	N	
	Photo	NVARCHAR(MAX)	N	

Рисунок 5

Сменить стиль

Логин

Пароль

Вход

Регистрация

Смена пароля

06.11.2025 10:17:41

Рисунок 6
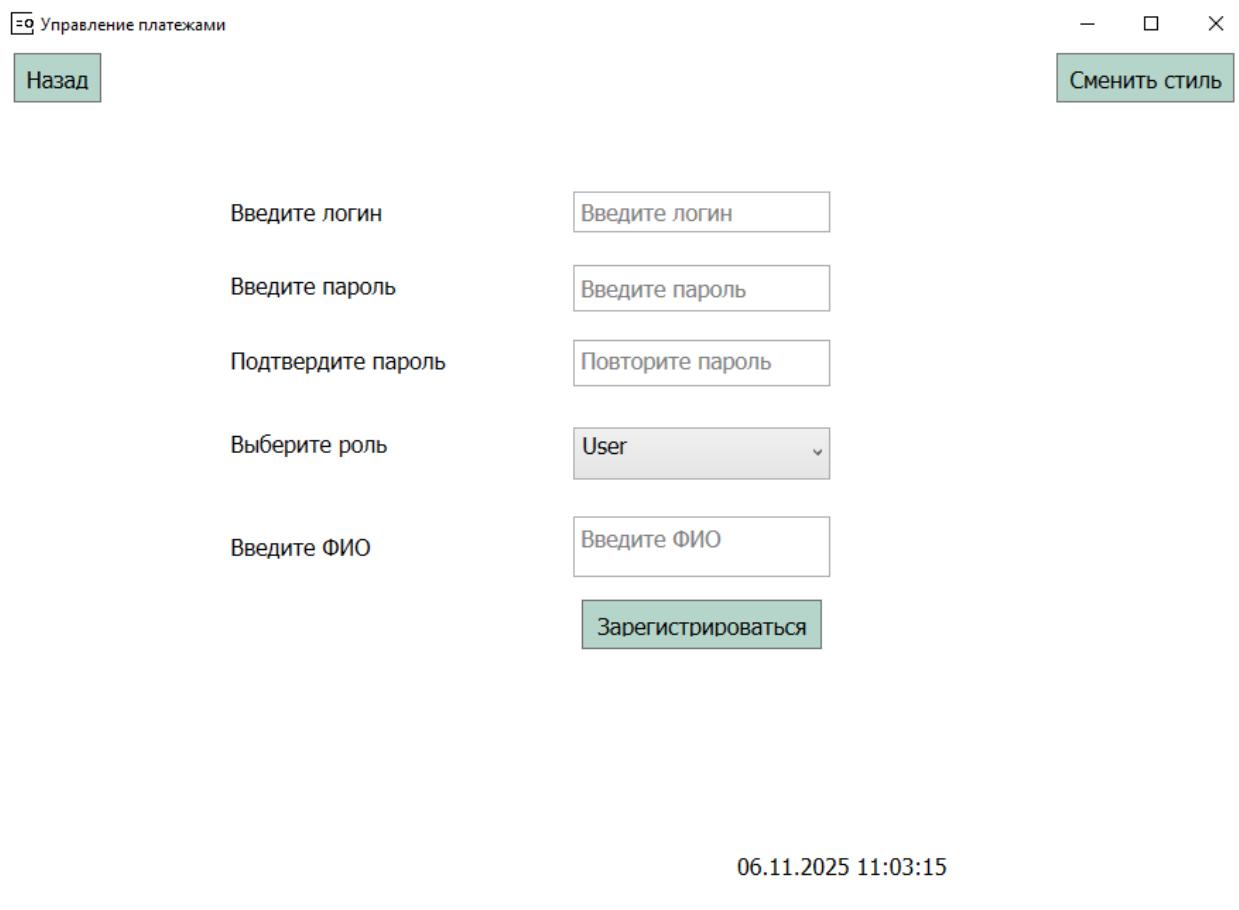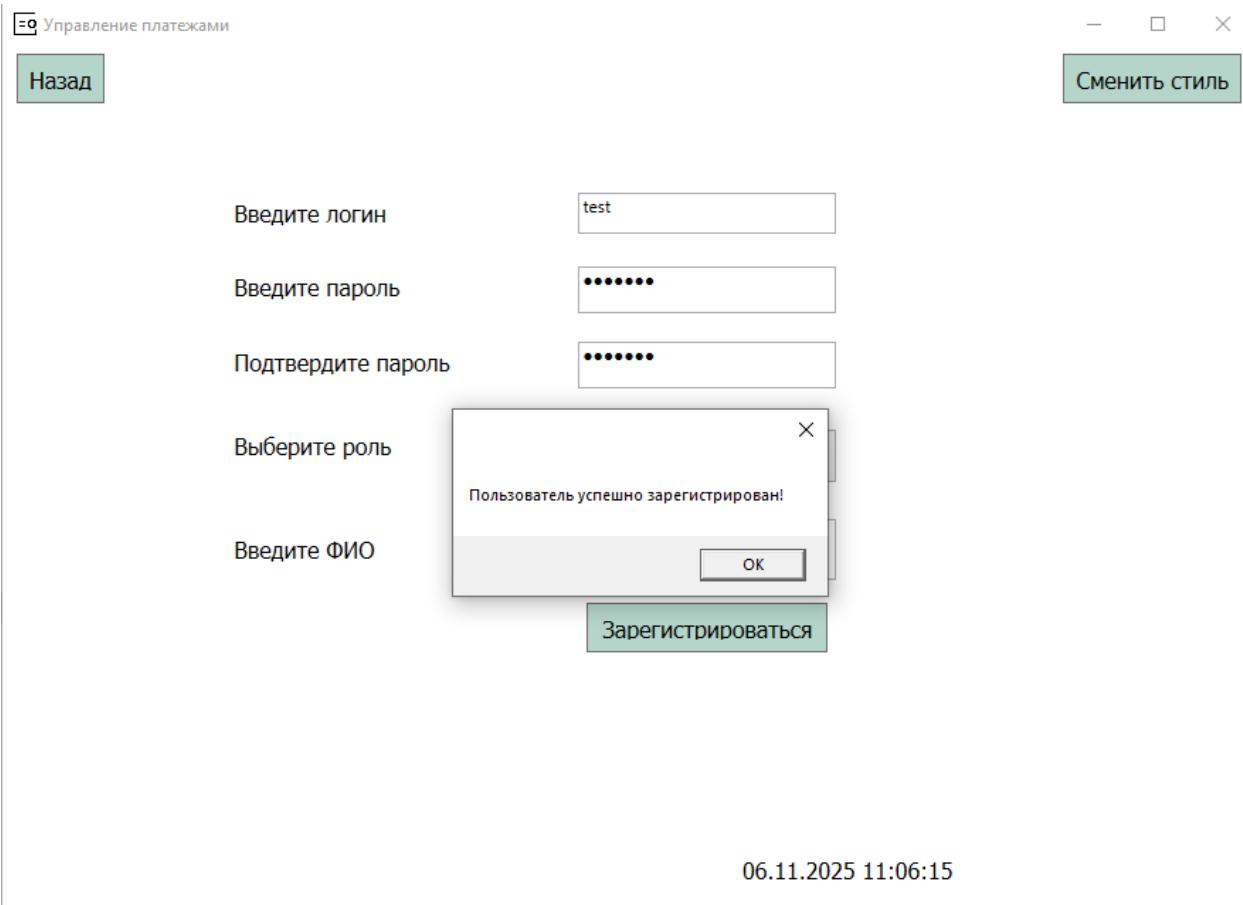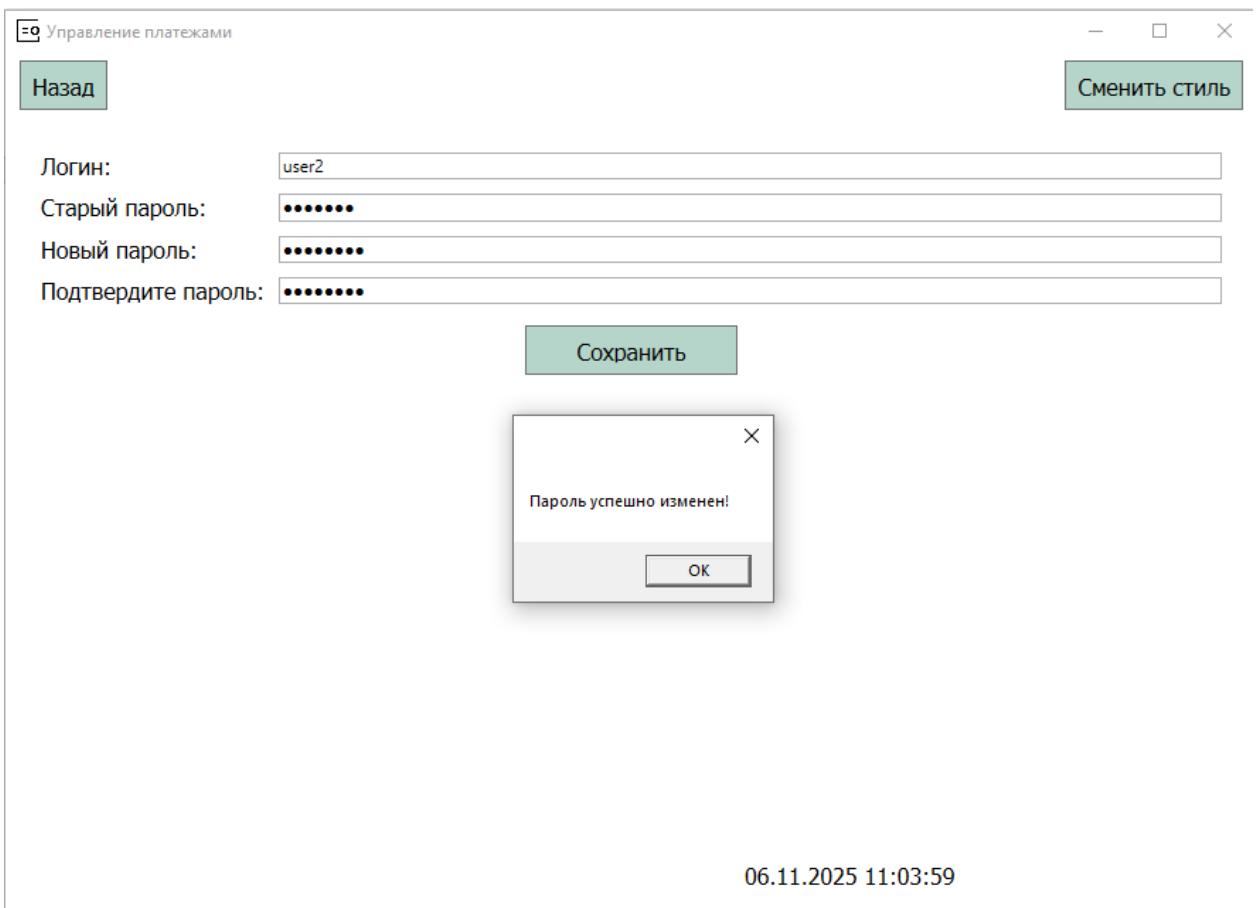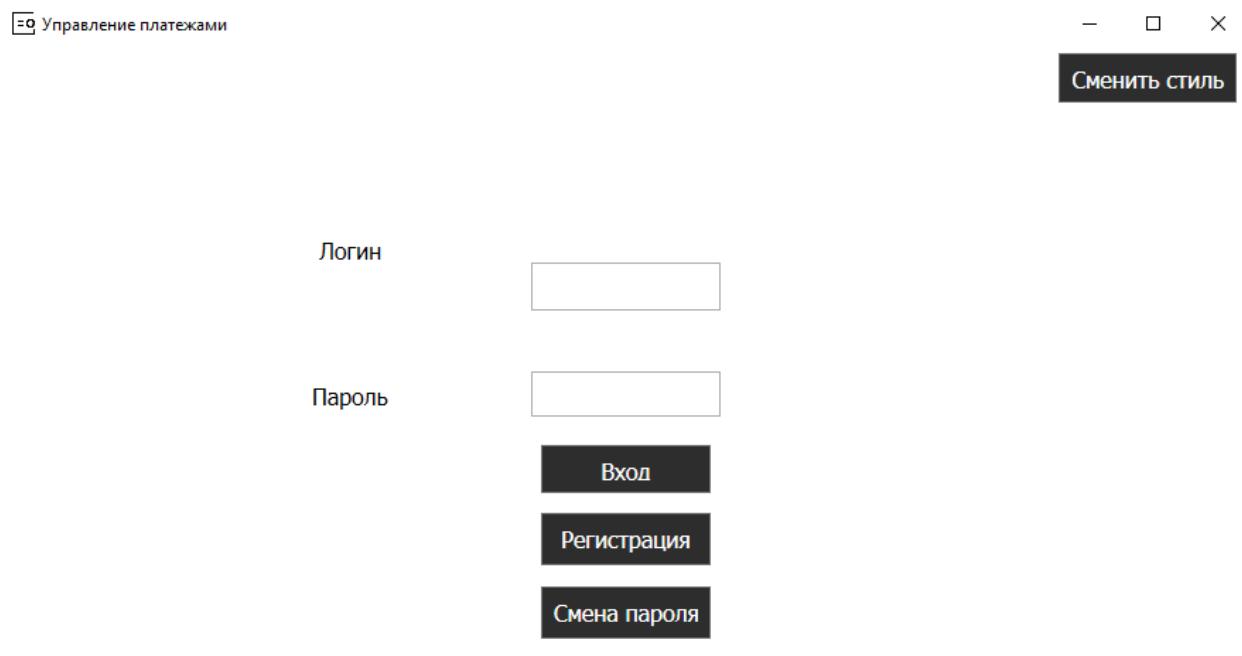
⊟♀ Управление платежами

Назад

Сменить стиль

Введите логин   Введите логин

Введите пароль   Введите пароль

Подтвердите пароль   Повторите пароль

Выберите роль   User

Введите ФИО   Введите ФИО

Зарегистрироваться

06.11.2025 11:03:15

Рисунок 7

Рисунок 8

Назад

Сменить стиль

Логин: user2

Старый пароль: ••••••

Новый пароль: •••••••

Подтвердите пароль: •••••••

Сохранить

Пароль успешно изменен!

OK

06.11.2025 11:03:59

Рисунок 9

Управление платежами

Сменить стиль

Логин

Пароль

Вход

Регистрация

Смена пароля

06.11.2025 10:17:58

Рисунок 10

Джабаров Расим Админович	Желтобрюх Заяц Вахович	Миргасимов Расим Амирович
Admin	User	User

06.11.2025 10:43:01

Рисунок 11

Назад   Сменить стиль

Введите ФИО для поиска: ☐   Сортировка по ФИО: По убыванию ⌄   ☑ Только администраторы

Очистить фильтр

Джабаров Расим Админович

АВД

Admin

06.11.2025 10:54:53

Рисунок 12

Назад   Сменить стиль

Таблица пользователе   Таблица категорий   Таблица платежей

Диаграммы
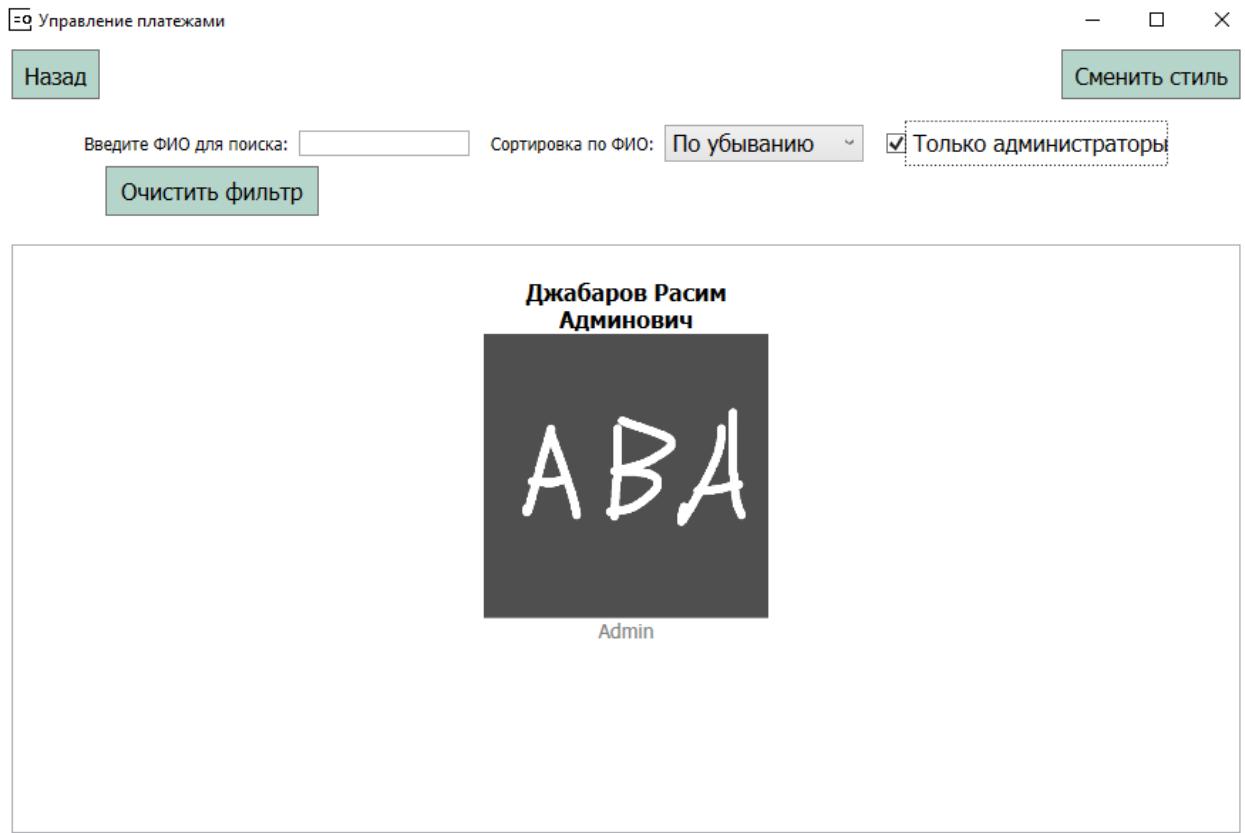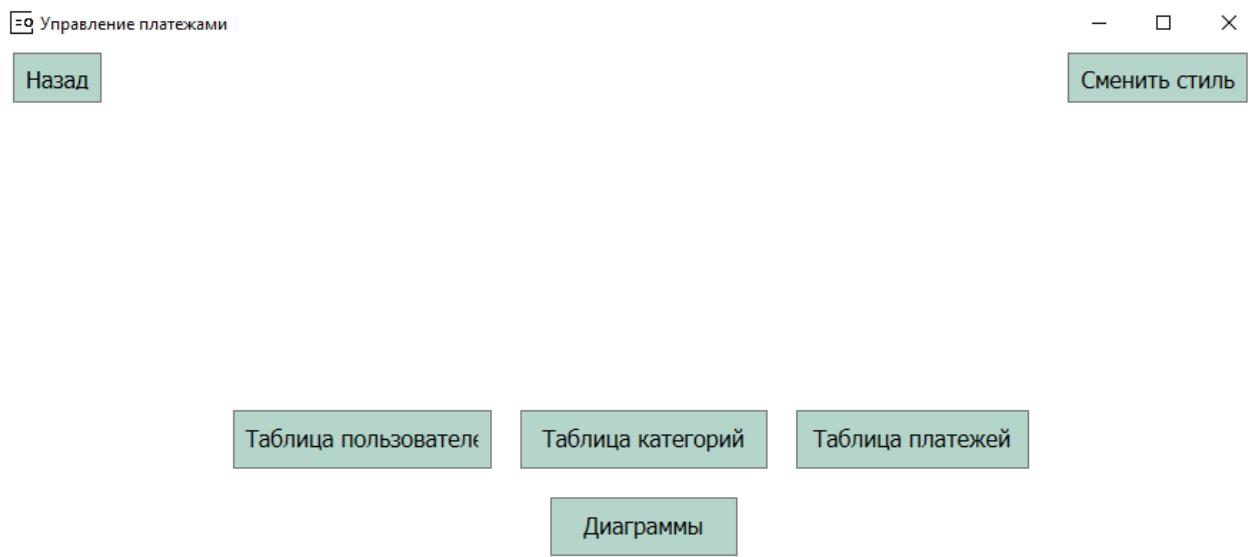
06.11.2025 10:55:46

Рисунок 13

Логин	Пароль	Роль	ФИО	Фото		
user1	D785D63511A645	User	Миргасимов Расим Амирович	ABA	Редакт⊾	
adminrasim	D8B736FE94FA63	Admin	Джабаров Расим Админович	ABA	Редакт⊾	
user2	0C7489C79A5AB4	User	Желтобрюх Заяц Вахович	ABA	Редакт⊾	

Назад **Сменить стиль**

Добавить **Удалить**

06.11.2025 10:56:00

Рисунок 14

Рисунок 15

17
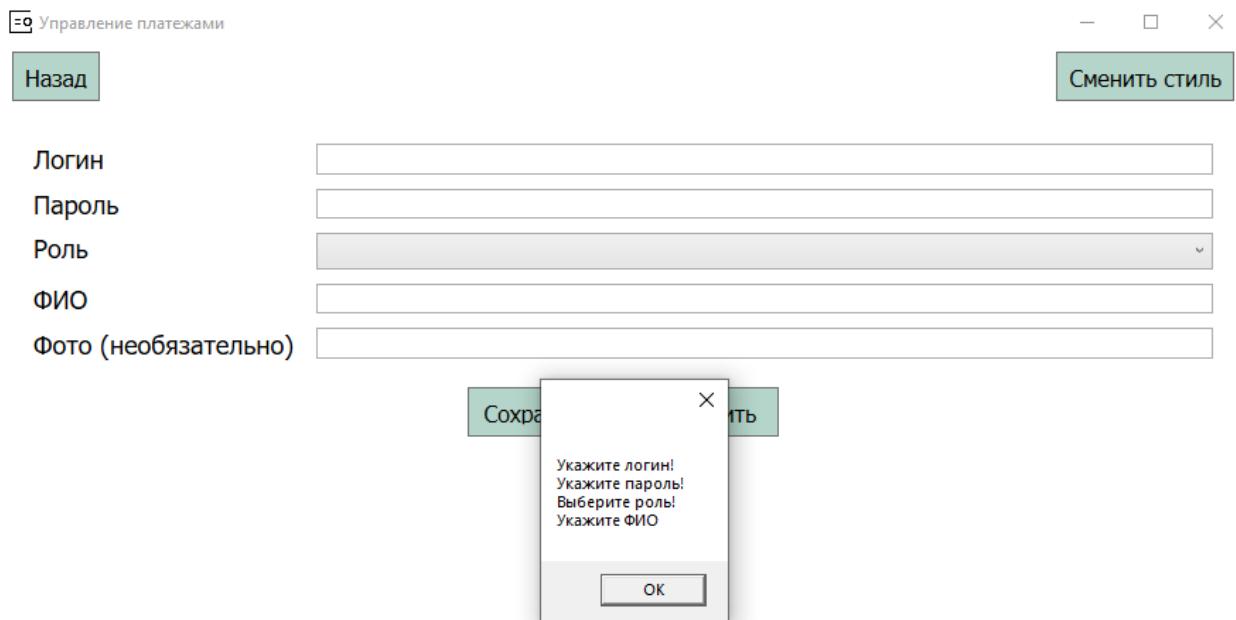
Рисунок 16

Назад

Сменить стиль

Логин

Пароль

Роль ⌄

ФИО

Фото (необязательно)

Сохр... ✕ ...ть

Укажите логин!
Укажите пароль!
Выберите роль!
Укажите ФИО

ОК

06.11.2025 10:56:42

Рисунок 17

С... ✕

Данные успешно сохранены!

ОК

Рисунок 18

Внимание ✕

❓ Вы точно хотите удалить записи в количестве 1 элементов?

Да   Нет

Рисунок 19

Рисунок 20

Управление платежами

Назад

Сменить стиль

Категория	
Вахо	Редактировать
Фрукты	Редактировать
Овощи	Редактировать

Добавить

Удалить

06.11.2025 10:58:09

Управление платежами

Назад

Сменить стиль

Название категории

Патроны

Сохранить   Очистить

Данные успешно сохранены!

OK
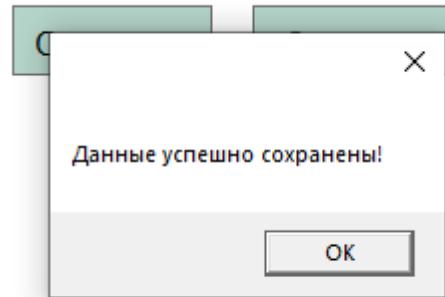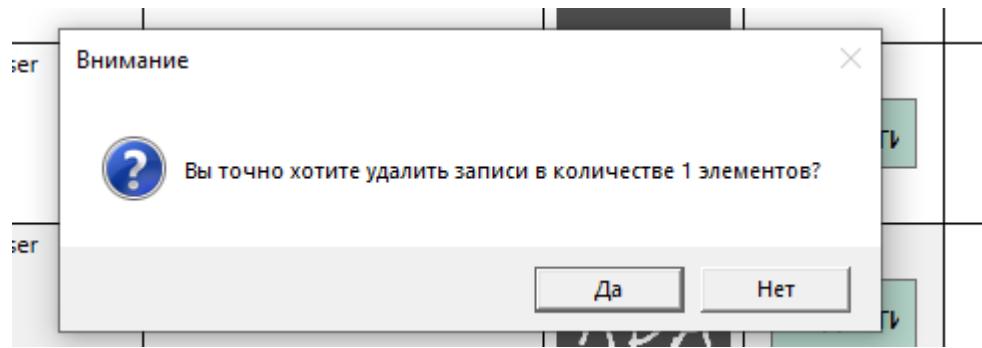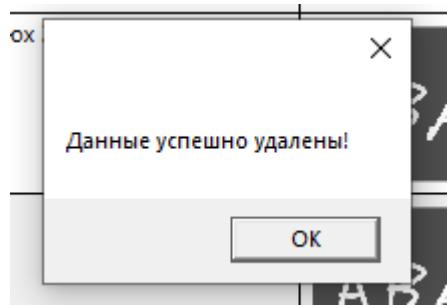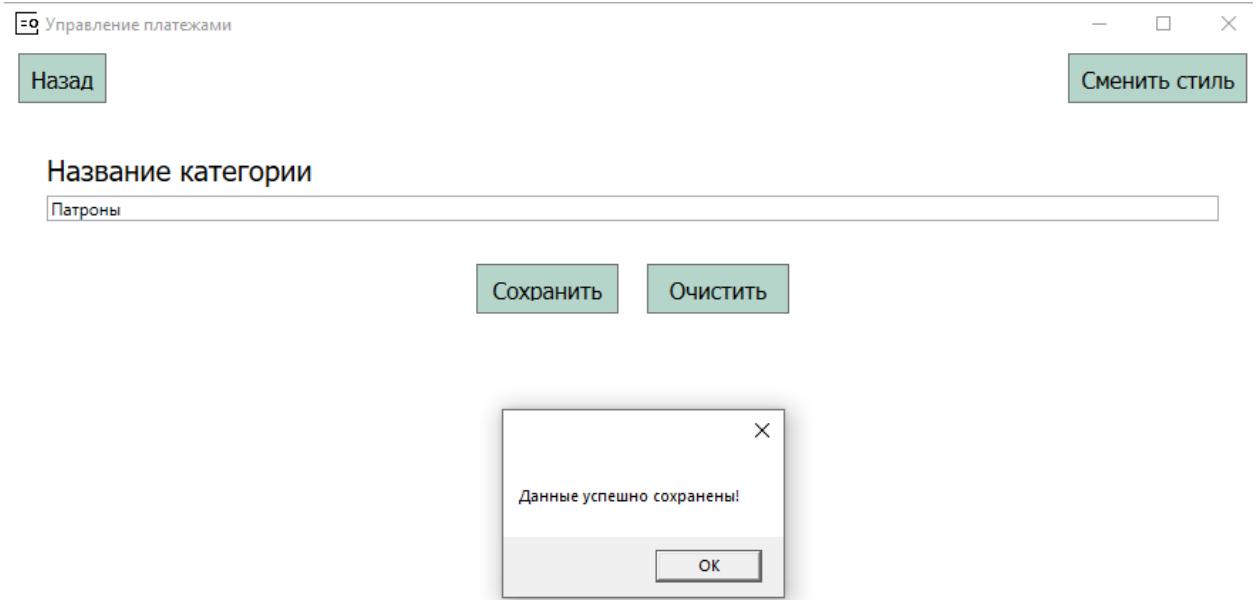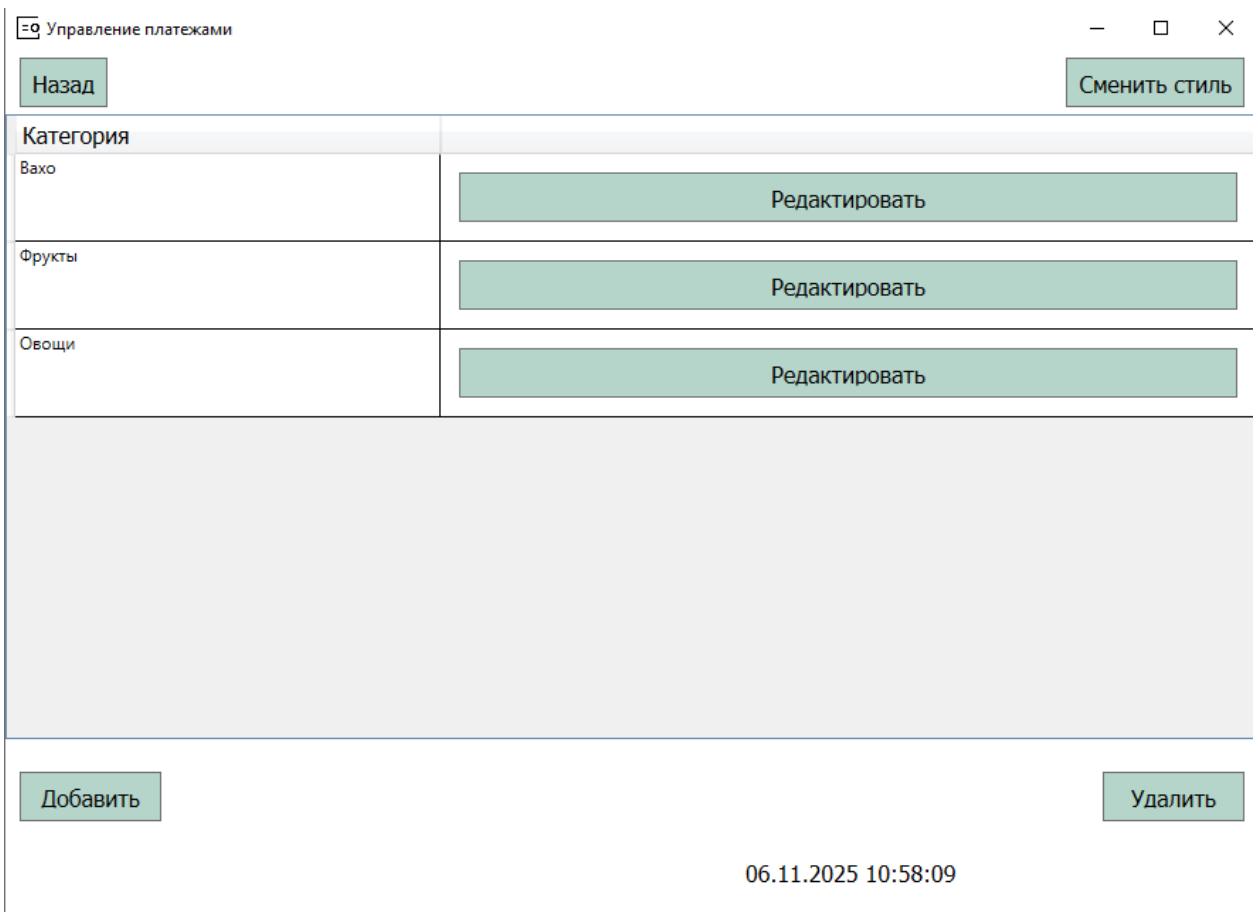
06.11.2025 10:58:48

Рисунок 21
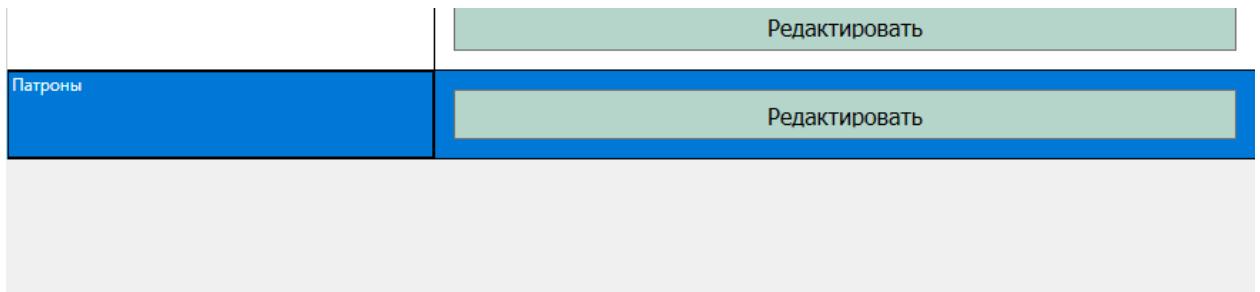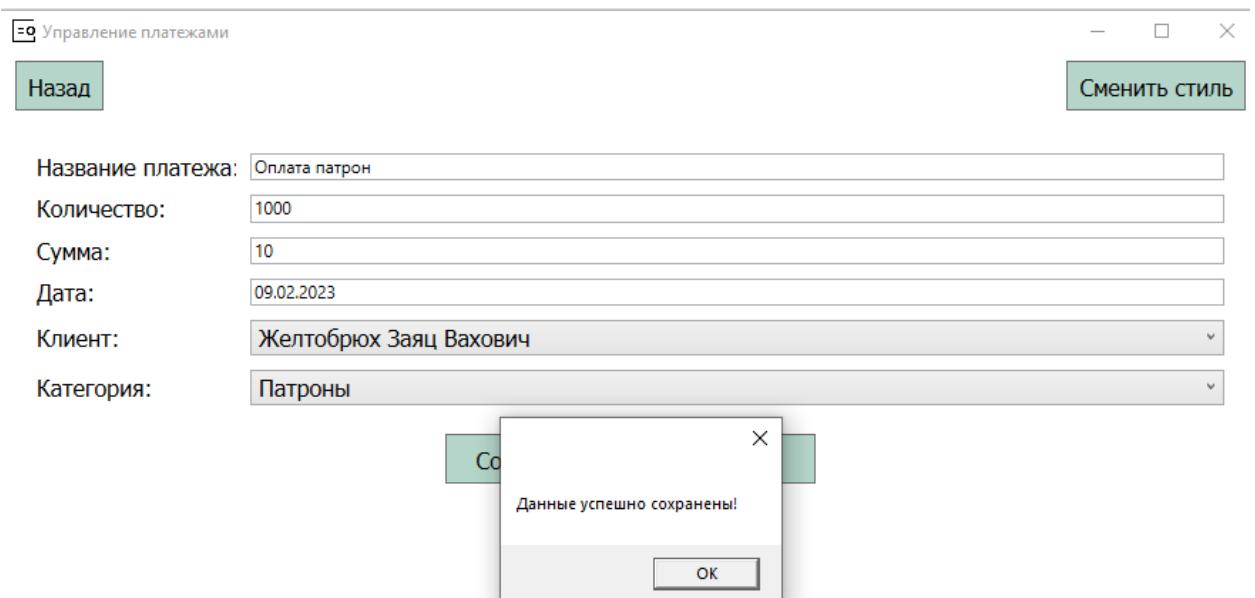
		Редактировать	
Патроны		Редактировать	

Рисунок 22

Дата	Пользователь	Название	Количество	Сумма	Категория	
05.03.2025	Миргасимов Расим Амирс	2	1000	100	Вахо	Редактиро
10.08.2024	Миргасимов Расим Амирс	Яблоко	10	100	Фрукты	Редактиро
01.01.2025	Джабаров Расим Админої	Помидора	15	120	Овощи	Редактиро

Управление платежами

Назад   Сменить стиль

Добавить   Удалить

06.11.2025 10:59:40

Рисунок 23

Рисунок 24

Рисунок 25

Рисунок 26

Рисунок 27

Дата платежа	Название	Стоимость	Количество	Сумма
		Вахо		
05.03.2025	2	100,00	1000	100000,00
			ИТОГО:	100000,00
		Фрукты		
10.08.2024	Яблоко	100,00	10	1000,00
			ИТОГО:	1000,00

Джабаров Расим Админович | Желтобрюх Заяц Вахович | **Миргасимов Расим Амирович** | Общий итог

Рисунок 28

100 % ▼ ✅ Проблемы не найдены. ◄

⊞ Результаты | 📄 Сообщения

	ID	Name
1	1	Вахо
2	2	Фрукты
3	3	Овощи
4	4	Патроны

Рисунок 29

Рисунок 30

Рисунок 31

Листинг №2 - Код "App.xaml"

```xml
<Application x:Class="_422_Zheltobryukh.App"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
             xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
             xmlns:local="clr-namespace:_422_Zheltobryukh"
             StartupUri="MainWindow.xaml">
    <Application.Resources>
        <ResourceDictionary>
            <ResourceDictionary.MergedDictionaries>
                <ResourceDictionary Source="/Dictionary.xaml"/>
            </ResourceDictionary.MergedDictionaries>
        </ResourceDictionary>
    </Application.Resources>
</Application>
```

Листинг №3 - Код "App.xaml.cs"

```csharp
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Linq;
using System.Threading.Tasks;
using System.Windows;

namespace _422_Zheltobryukh
{
    /// <summary>
    /// Interaction logic for App.xaml
    /// </summary>
    public partial class App : Application
    {
    }
```

25

```
}
```

Листинг №4 - Код "Dictionary.xaml"

```xml
<ResourceDictionary
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
                    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">
    <Color x:Key="PrimaryColor">#B5D5CA</Color>
    <Color x:Key="AccentColor">#E0A9AF</Color>
    <Color x:Key="TextColor">#000000</Color>
    <Color x:Key="BackgroundColor">#FFFCD6</Color>



    <SolidColorBrush x:Key="PrimaryBrush" Color="{StaticResource
PrimaryColor}"/>
    <SolidColorBrush x:Key="AccentBrush" Color="{StaticResource
AccentColor}"/>
    <SolidColorBrush x:Key="TextBrush" Color="{StaticResource TextColor}"/>
    <SolidColorBrush x:Key="BackgroundBrush" Color="{StaticResource
BackgroundColor}"/>



    <Style
TargetType="Button">
        <Setter Property="Height" Value="35"/>
        <Setter Property="Padding" Value="8"/>
        <Setter Property="Margin" Value="12"/>
        <Setter Property="FontFamily" Value="Tahoma"/>
        <Setter Property="FontSize" Value="12pt"/>
        <Setter Property="Background" Value="{StaticResource
PrimaryBrush}"/>
        <Setter Property="Foreground" Value="{StaticResource TextBrush}"/>
    </Style>
    <Style
TargetType="TextBlock">
        <Setter Property="FontFamily" Value="Tahoma"/>
        <Setter Property="FontSize" Value="12pt"/>

    </Style>
    <Style TargetType="Window">
        <Setter Property="Background" Value="{StaticResource
BackgroundBrush}"/>
    </Style>

</ResourceDictionary>
```

Листинг №5 - Код "DictionaryDark.xaml"

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


    <Color x:Key="PrimaryColor">#2D2D2D</Color>
    <Color x:Key="AccentColor">#E0A9AF</Color>
    <Color x:Key="TextColor">#FFFFFF</Color>
    <Color x:Key="BackgroundColor">#1E1E1E</Color>


    <SolidColorBrush x:Key="PrimaryBrush" Color="{StaticResource
PrimaryColor}"/>
```

```xml
    <SolidColorBrush x:Key="AccentBrush" Color="{StaticResource
AccentColor}"/>
    <SolidColorBrush x:Key="TextBrush" Color="{StaticResource TextColor}"/>
    <SolidColorBrush x:Key="BackgroundBrush" Color="{StaticResource
BackgroundColor}"/>


    <Style
TargetType="Button">
        <Setter Property="Height" Value="35"/>
        <Setter Property="Padding" Value="8"/>
        <Setter Property="Margin" Value="12"/>
        <Setter Property="FontFamily" Value="Tahoma"/>
        <Setter Property="FontSize" Value="12pt"/>
        <Setter Property="Background" Value="{StaticResource
PrimaryBrush}"/>
        <Setter Property="Foreground" Value="{StaticResource TextBrush}"/>
    </Style>
    <Style
TargetType="TextBlock">
        <Setter Property="FontFamily" Value="Tahoma"/>
        <Setter Property="FontSize" Value="12pt"/>

    </Style>
    <Style TargetType="Window">
        <Setter Property="Background" Value="{StaticResource
BackgroundBrush}"/>
    </Style>

</ResourceDictionary>
```

Листинг №6 - Код "MainWindow.xaml"

```xml
<Window x:Class="_422_Zheltobryukh.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:_422_Zheltobryukh"
        mc:Ignorable="d"
        Loaded="Window_Loaded"
        Closing="Window_Closing"
        Title="Управление платежами"
        Height="650"
        Width="900"
        WindowStartupLocation="CenterScreen">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="45"/>
            <RowDefinition/>
            <RowDefinition Height="45"/>
        </Grid.RowDefinitions>

        <Frame Grid.Row="1" x:Name="MainFrame"
NavigationUIVisibility="Hidden" />

        <TextBlock x:Name="DateTimeNow"
                HorizontalAlignment="Left"
                Margin="522,0,0,0"
                Grid.Row="2"
                VerticalAlignment="Center"
```

```
                              Height="35"
                              Width="194"
                              Grid.Column="1"/>

        <Button Content="Назад"
                Grid.Row="0"
                HorizontalAlignment="Left"
                VerticalAlignment="Top"
                Margin="10,5,0,0"
                Click="BackButton_Click"
                x:Name="BackButton"/>

        <Button Content="Сменить стиль"
                HorizontalAlignment="Right"
                VerticalAlignment="Top"
                Margin="10,5,10,0"
                Click="ChangeStyleButton_Click"/>
    </Grid>
</Window>
```

Листинг №7 - Код "MainWindow.xaml.cs"

```
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 _422_Zheltobryukh
{
    /// <summary>
    /// Interaction logic for MainWindow.xaml
    /// </summary>
    public partial class MainWindow : Window
    {
        private bool _isDarkTheme = false;

        public MainWindow()
        {
            InitializeComponent();
            MainFrame.Source = new Uri("Pages/AuthPage.xaml",
UriKind.Relative);
            MainFrame.Navigated += MainFrame_Navigated;
        }

        private void MainFrame_Navigated(object sender, NavigationEventArgs
e)
        {
            // Обновляем видимость кнопки назад
            UpdateBackButtonVisibility();
        }

        private void UpdateBackButtonVisibility()
        {
```

```
            // Показываем кнопку назад только если есть куда возвращаться
            BackButton.Visibility = MainFrame.CanGoBack ? Visibility.Visible
: Visibility.Collapsed;
        }

        // Публичный метод для очистки истории навигации
        public void ClearNavigationHistory()
        {
            // Очищаем журнал навигации
            while (MainFrame.CanGoBack)
            {
                MainFrame.RemoveBackEntry();
            }
            UpdateBackButtonVisibility();
        }

        private void Window_Loaded(object sender, RoutedEventArgs e)
        {
            var timer = new System.Windows.Threading.DispatcherTimer();
            timer.Interval = new TimeSpan(0, 0, 1);
            timer.IsEnabled = true;
            timer.Tick += (o, t) => {
                DateTimeNow.Text = DateTime.Now.ToString();
            };
            timer.Start();

            // Инициализируем видимость кнопки назад
            UpdateBackButtonVisibility();
        }

        void Window_Closing(object sender,
System.ComponentModel.CancelEventArgs e)
        {
            if (MessageBox.Show("Вы уверены, что хотите закрыть окно?",
"Message",
            MessageBoxButton.YesNo) == MessageBoxResult.No)
                e.Cancel = true;
            else
                e.Cancel = false;
        }

        // Обработчик кнопки "Сменить стиль"
        private void ChangeStyleButton_Click(object sender, RoutedEventArgs
e)
        {
            // Переключаем тему при каждом нажатии
            _isDarkTheme = !_isDarkTheme;

            string themeFile = _isDarkTheme ? "DictionaryDark.xaml" :
"Dictionary.xaml";
            var uri = new Uri(themeFile, UriKind.Relative);

            // Загружаем словарь ресурсов
            ResourceDictionary resourceDict = Application.LoadComponent(uri)
as ResourceDictionary;

            // Очищаем коллекцию ресурсов приложения
            Application.Current.Resources.Clear();

            // Добавляем загруженный словарь ресурсов
Application.Current.Resources.MergedDictionaries.Add(resourceDict);
        }
```

```
        // Обработчик кнопки "Назад"
        private void BackButton_Click(object sender, RoutedEventArgs e)
        {
            if (MainFrame.CanGoBack)
            {
                MainFrame.GoBack();
            }
        }
    }
}
```

Листинг №8 - Код "AddCategoryPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.AddCategoryPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      Title="AddCategoryPage">
    <Grid Margin="20">
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
        </Grid.RowDefinitions>

        <TextBlock Text="Название категории" Grid.Row="0" FontSize="20"
Margin="10"/>

        <TextBox x:Name="TBCategoryName"
                 Text="{Binding Name}"
                 Grid.Row="0" Margin="10,40,10,10"/>

        <StackPanel Grid.Row="1"
Orientation="Horizontal" HorizontalAlignment="Center" Margin="10">
            <Button x:Name="ButtonSaveCategory" Content="Сохранить"
Width="100" Margin="10" Click="ButtonSaveCategory_Click"/>
            <Button x:Name="ButtonClean" Content="Очистить" Width="100"
Margin="10" Click="ButtonClean_Click"/>
        </StackPanel>
    </Grid>
</Page>
```

Листинг №9 - Код "AddCategoryPage.xaml.cs"

```csharp
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 _422_Zheltobryukh.Pages
{
    /// <summary>
    /// Логика взаимодействия для AddCategoryPage.xaml
    /// </summary>
```

```csharp
    public partial class AddCategoryPage : Page
    {
        private Category _currentCategory = new Category();

        public AddCategoryPage(Category selectedCategory)
        {
            InitializeComponent();

            if (selectedCategory != null)
                _currentCategory = selectedCategory;


            DataContext = _currentCategory;
        }


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

            if (string.IsNullOrWhiteSpace(_currentCategory.Name))
                errors.AppendLine("Укажите название категории!");

            if (errors.Length > 0)
            {
                MessageBox.Show(errors.ToString());
                return;
            }

            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                if (_currentCategory.ID == 0)
                {
                    db.Categories.Add(_currentCategory);
                }
                else
                {

                    var existingCategory =
db.Categories.Find(_currentCategory.ID);

                    if (existingCategory != null)
                    {
                        existingCategory.Name = _currentCategory.Name;

                        db.Entry(existingCategory).State =
System.Data.Entity.EntityState.Modified;
                    }
                }

                try
                {
                    db.SaveChanges();
                    MessageBox.Show("Данные успешно сохранены!");
                    NavigationService.GoBack();
                }
                catch (Exception ex)
                {
                    MessageBox.Show($"Ошибка сохранения:
{ex.Message}\n\nInnerException:\n{ex.InnerException?.Message}");
                }
            }
```

```
                }


        private void ButtonClean_Click(object sender, RoutedEventArgs e)
        {

            TBCategoryName.Text = "";
        }
    }
}
```

Листинг №10 - Код "AddPaymentPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.AddPaymentPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      Title="AddPaymentPage">
    <Grid Margin="20">
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="150"/>
            <ColumnDefinition Width="*"/>
        </Grid.ColumnDefinitions>

        <TextBlock Text="Название платежа:" Grid.Row="0" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <TextBox x:Name="TBPaymentName" Text="{Binding Name}" Grid.Row="0"
Grid.Column="1" Margin="5"/>

        <TextBlock Text="Количество:" Grid.Row="1" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <TextBox x:Name="TBCount" Text="{Binding Num}" Grid.Row="1"
Grid.Column="1" Margin="5"/>

        <TextBlock Text="Сумма:" Grid.Row="2" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <TextBox x:Name="TBAmount" Text="{Binding Price}" Grid.Row="2"
Grid.Column="1" Margin="5"/>

        <TextBlock Text="Дата:" Grid.Row="3" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <TextBox x:Name="TBDate" Text="{Binding Date,
StringFormat='dd.MM.yyyy'}" Grid.Row="3" Grid.Column="1" Margin="5"/>

        <TextBlock Text="Клиент:" Grid.Row="4" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <ComboBox x:Name="CBUser" Grid.Row="4" Grid.Column="1" Margin="5"
Height="25"/>

        <TextBlock Text="Категория:" Grid.Row="5" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <ComboBox x:Name="CBCategory" Grid.Row="5" Grid.Column="1"
Margin="5" Height="25"/>
```

```xml
        <StackPanel Grid.Row="6" Grid.ColumnSpan="2"
Orientation="Horizontal" HorizontalAlignment="Center" Margin="5">
            <Button x:Name="ButtonSave" Content="Сохранить" Width="120"
Height="35" Margin="10" Click="ButtonSave_Click"/>
            <Button x:Name="ButtonClean" Content="Очистить" Width="120"
Height="35" Margin="10" Click="ButtonClean_Click"/>
        </StackPanel>
    </Grid>
</Page>
```

Листинг №11 - Код "AddPaymentPage.xaml.cs"

```csharp
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 _422_Zheltobryukh.Pages
{
    /// <summary>
    /// Логика взаимодействия для AddPaymentPage.xaml
    /// </summary>
    public partial class AddPaymentPage : Page
    {
        private Payment _currentPayment = new Payment();

        public AddPaymentPage(Payment selectedPayment)
        {
            InitializeComponent();

            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                CBCategory.ItemsSource = db.Categories.ToList();
                CBUser.ItemsSource = db.Users.ToList();
            }

            CBCategory.DisplayMemberPath = "Name";
            CBUser.DisplayMemberPath = "FIO";

            if (selectedPayment != null)
            {
                _currentPayment = selectedPayment;
            }

            DataContext = _currentPayment;

            if (selectedPayment != null)
            {
                CBUser.SelectedValue = selectedPayment.UserID;
                CBCategory.SelectedValue = selectedPayment.CategoryID;
            }
            else
            {
```

```csharp
                        // Устанавливаем текущую дату для новых записей
                        TBDate.Text = DateTime.Now.ToString("dd.MM.yyyy");
                }
        }

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

            // 1. ПРОВЕРКА НА ПУСТОЕ ИМЯ (решение ошибки валидации)
            if (string.IsNullOrWhiteSpace(_currentPayment.Name))
                errors.AppendLine("Укажите название платежа!");

            if (!DateTime.TryParse(TBDate.Text, out DateTime paymentDate))
            {
                errors.AppendLine("Введите корректную дату (например,
dd.MM.yyyy)!");
            }
            else
            {
                _currentPayment.Date = paymentDate;
            }

            if (!decimal.TryParse(TBCount.Text, out decimal num) || num <=
0)
                errors.AppendLine("Укажите корректное количество!");
            else
                _currentPayment.Num = num;

            if (!decimal.TryParse(TBAmount.Text, out decimal price) || price
<= 0)
                errors.AppendLine("Укажите корректную цену!");
            else
                _currentPayment.Price = price;

            if (CBUser.SelectedItem == null)
                errors.AppendLine("Укажите клиента!");
            else
                _currentPayment.UserID = (CBUser.SelectedItem as User).ID;

            if (CBCategory.SelectedItem == null)
                errors.AppendLine("Укажите категорию!");
            else
                _currentPayment.CategoryID = (CBCategory.SelectedItem as
Category).ID;


            if (errors.Length > 0)
            {
                MessageBox.Show(errors.ToString());
                return;
            }

            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                if (_currentPayment.ID == 0)
                {
                    db.Payments.Add(_currentPayment);
                }
                else
                {
                    // 2. ИСПРАВЛЕНИЕ РЕДАКТИРОВАНИЯ (убираем SetValues)
```

```
                        var existingPayment =
db.Payments.Find(_currentPayment.ID);
                        if (existingPayment != null)
                        {
                            // Вручную обновляем поля
                            existingPayment.Name = _currentPayment.Name;
                            existingPayment.Date = _currentPayment.Date;
                            existingPayment.Num = _currentPayment.Num;
                            existingPayment.Price = _currentPayment.Price;
                            existingPayment.UserID = _currentPayment.UserID;
                            existingPayment.CategoryID =
_currentPayment.CategoryID;
                        }
                    }

                    try
                    {
                        db.SaveChanges();
                        MessageBox.Show("Данные успешно сохранены!");
                        NavigationService.GoBack();
                    }
                    catch
(System.Data.Entity.Validation.DbEntityValidationException ex)
                    {
                        // Более подробная ошибка валидации
                        var errorMessages = ex.EntityValidationErrors
                            .SelectMany(x => x.ValidationErrors)
                            .Select(x => x.ErrorMessage);
                        var fullErrorMessage = string.Join("\n", errorMessages);
                        MessageBox.Show($"Ошибка
валидации:\n{fullErrorMessage}");
                    }
                    catch (Exception ex)
                    {
                        MessageBox.Show($"Ошибка сохранения:
{ex.Message}\n\nInnerException:\n{ex.InnerException?.Message}", "Ошибка");
                    }
                }
        }

        private void ButtonClean_Click(object sender, RoutedEventArgs e)
        {
            TBPaymentName.Text = "";
            TBAmount.Text = "";
            TBCount.Text = "";
            TBDate.Text = DateTime.Now.ToString("dd.MM.yyyy");
            CBCategory.SelectedIndex = -1;
            CBUser.SelectedIndex = -1;
        }
    }
}
```

Листинг №12 - Код "AddUserPage.xaml"

```
<Page x:Class="_422_Zheltobryukh.Pages.AddUserPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      Title="AddUserPage">
    <Grid Margin="20">
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
```

```
                <RowDefinition Height="Auto"/>
                <RowDefinition Height="Auto"/>
                <RowDefinition Height="Auto"/>
            </Grid.RowDefinitions>
            <Grid.ColumnDefinitions>
                <ColumnDefinition Width="200"/>
                <ColumnDefinition Width="*"/>
            </Grid.ColumnDefinitions>

            <TextBlock Text="Логин" Grid.Row="0" Grid.Column="0" FontSize="18"
Margin="5"/>
            <TextBox x:Name="TBLogin" Text="{Binding LOGIN}" Grid.Row="0"
Grid.Column="1" Margin="5"/>

            <TextBlock Text="Пароль" Grid.Row="1" Grid.Column="0" FontSize="18"
Margin="5"/>
            <TextBox x:Name="TBPass" Text="{Binding Password}" Grid.Row="1"
Grid.Column="1" Margin="5"/>

            <TextBlock Text="Роль" Grid.Row="2" Grid.Column="0" FontSize="18"
Margin="5"/>
            <ComboBox x:Name="cmbRole" SelectedItem="{Binding Role}"
Grid.Row="2" Grid.Column="1" Margin="5"/>

            <TextBlock Text="ФИО" Grid.Row="3" Grid.Column="0" FontSize="18"
Margin="5"/>
            <TextBox x:Name="TBFio" Text="{Binding FIO}" Grid.Row="3"
Grid.Column="1" Margin="5"/>

            <TextBlock Text="Фото (необязательно)" Grid.Row="4" Grid.Column="0"
FontSize="18" Margin="5"/>
            <TextBox x:Name="TBPhoto" Text="{Binding Photo}" Grid.Row="4"
Grid.Column="1" Margin="5"/>

            <StackPanel Grid.Row="5" Grid.ColumnSpan="2"
Orientation="Horizontal" HorizontalAlignment="Center" Margin="5">
                <Button x:Name="ButtonSave" Content="Сохранить" Width="100"
Margin="10" Click="ButtonSave_Click"/>
                <Button x:Name="ButtonClean" Content="Очистить" Width="100"
Margin="10" Click="ButtonClean_Click"/>
            </StackPanel>
        </Grid>
</Page>
```

Листинг №13 - Код "AddUserPage.xaml.cs"

```
using System;
using System.Linq;
using System.Text;
using System.Windows;
using System.Windows.Controls;

namespace _422_Zheltobryukh.Pages
{
    public partial class AddUserPage : Page
    {
        private User _currentUser = new User();

        public AddUserPage(User selectedUser)
        {
            InitializeComponent();
```

```
            if (selectedUser != null)
            {
                _currentUser = selectedUser;
            }

            DataContext = _currentUser;

            cmbRole.ItemsSource = new string[] { "Admin", "User" };
            if (selectedUser != null)
            {
                cmbRole.SelectedItem = selectedUser.Role;
            }
        }
        public static string GetHash(string password)
        {
            using (var sha256 =
System.Security.Cryptography.SHA256.Create())
            {
                byte[] bytes = Encoding.UTF8.GetBytes(password);
                byte[] hashBytes = sha256.ComputeHash(bytes);
                return BitConverter.ToString(hashBytes).Replace("-",
"").ToUpper();
            }
        }

        private void ButtonSave_Click(object sender, RoutedEventArgs e)
        {


            StringBuilder errors = new StringBuilder();

            if (string.IsNullOrWhiteSpace(_currentUser.LOGIN))
                errors.AppendLine("Укажите логин!");
            if (string.IsNullOrWhiteSpace(_currentUser.Password))
                errors.AppendLine("Укажите пароль!");
            if (cmbRole.SelectedItem == null)
                errors.AppendLine("Выберите роль!");
            else
                _currentUser.Role = cmbRole.SelectedItem.ToString();
            if (string.IsNullOrWhiteSpace(_currentUser.FIO))
                errors.AppendLine("Укажите ФИО");

            if (errors.Length > 0)
            {
                MessageBox.Show(errors.ToString());
                return;
            }

            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                _currentUser.Password = GetHash(_currentUser.Password);

                if (_currentUser.ID == 0)
                {
                    db.Users.Add(_currentUser);
                }
                else
                {
                    var existingUser = db.Users.Find(_currentUser.ID);
                    if (existingUser != null)
                    {
db.Entry(existingUser).CurrentValues.SetValues(_currentUser);
```

```
                }
            }

            try
            {
                db.SaveChanges();
                MessageBox.Show("Данные успешно сохранены!");
                NavigationService.GoBack();
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message.ToString());
            }
        }
    }

    private void ButtonClean_Click(object sender, RoutedEventArgs e)
    {
        TBLogin.Text = "";
        TBPass.Text = "";
        cmbRole.SelectedIndex = -1;
        TBFio.Text = "";
        TBPhoto.Text = "";
    }
  }
}
```

Листинг №14 - Код "AdminPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.AdminPage"
      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:_422_Zheltobryukh.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="AdminPage">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="*"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="*"/>
        </Grid.RowDefinitions>

        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="*"/>
            <ColumnDefinition Width="Auto"/>
            <ColumnDefinition Width="Auto"/>
            <ColumnDefinition Width="Auto"/>
            <ColumnDefinition Width="*"/>
        </Grid.ColumnDefinitions>

        <!-- Первый ряд кнопок -->
        <Button x:Name="BtnTab1"
                Content="Таблица пользователей"
                Grid.Row="1" Grid.Column="1"
                Margin="10"
                Height="41" Width="181"
                Click="BtnTab1_Click"/>
```

38

```
            <Button x:Name="BtnTab2"
                    Content="Таблица категорий"
                    Grid.Row="1" Grid.Column="2"
                    Margin="10"
                    Height="41" Width="173"
                    Click="BtnTab2_Click"/>

            <Button x:Name="BtnTab3"
                    Content="Таблица платежей"
                    Grid.Row="1" Grid.Column="3"
                    Margin="10"
                    Height="41" Width="163"
                    Click="BtnTab3_Click"/>

            <!-- Второй ряд - кнопка по центру -->
            <Button x:Name="BtnTab4"
                    Content="Диаграммы"
                    Grid.Row="2" Grid.Column="2"
                    Margin="10"
                    Height="41" Width="131"
                    Click="BtnTab4_Click"/>
    </Grid>
</Page>
```

Листинг №15 - Код "AdminPage.xaml.cs"

```
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 _422_Zheltobryukh.Pages
{
    /// <summary>
    /// Interaction logic for AdminPage.xaml
    /// </summary>
    public partial class AdminPage : Page
    {
        public AdminPage()
        {
            InitializeComponent();
        }

        // Переход на страницу пользователей
        private void BtnTab1_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new UsersTabPage());
        }

        // Переход на страницу категорий
        private void BtnTab2_Click(object sender, RoutedEventArgs e)
        {
```

```
                NavigationService?.Navigate(new CategoryTabPage());
        }

        // Переход на страницу платежей
        private void BtnTab3_Click(object sender, RoutedEventArgs e)
        {
                NavigationService?.Navigate(new PaymentTabPage());
        }

        // Переход на страницу диаграмм
        private void BtnTab4_Click(object sender, RoutedEventArgs e)
        {
                NavigationService?.Navigate(new DiagrammPage());
        }
    }
}
```

Листинг №16 - Код "AuthPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.AuthPage"
      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:_422_Zheltobryukh.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="AuthPage">

    <Grid HorizontalAlignment="Center"
          Width="800"
          Height="450"
          VerticalAlignment="Top">
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="271*" />
            <ColumnDefinition Width="529*" />
        </Grid.ColumnDefinitions>
        <Grid.RowDefinitions>
            <RowDefinition Height="217*" />
            <RowDefinition Height="233*" />
        </Grid.RowDefinitions>

        <Label Content="Логин"
               x:Name="labelLogin"
               Margin="178,85,0,0"
               HorizontalAlignment="Left"
               Width="54"
               VerticalAlignment="Top"
               Visibility="Hidden" />

        <Label Content="Пароль"
               x:Name="labelPass"
               Margin="173,188,0,0"
               HorizontalAlignment="Left"
               Width="64"
               Grid.Row="0"
               Grid.RowSpan="2"
               VerticalAlignment="Top"
               Visibility="Hidden" />

        <TextBox x:Name="TextBoxLogin"
                 Background="Transparent"
```

```xml
                    Margin="62,0,0,75"
                    TextWrapping="Wrap"
                    ToolTip="Введите логин или почту"
                    TextChanged="TextBoxLogin_TextChanged"
                    Height="34"
                    VerticalAlignment="Bottom"
                    HorizontalAlignment="Left"
                    Width="134"
                    Grid.Column="1"
                    Visibility="Hidden"
                    CommandManager.PreviewExecuted="textBox_PreviewExecuted"/>

        <TextBox x:Name="txtHintLogin"
                    Grid.Column="1"
                    Margin="62,0,0,75"
                    Height="34"
                    VerticalAlignment="Bottom"
                    HorizontalAlignment="Left"
                    Width="134"
                    IsHitTestVisible="False"
                    IsReadOnly="True"
                    Background="Transparent"
                    Foreground="Gray"
                    Text="Введите логин или почту"
                    Visibility="Hidden" />

        <PasswordBox x:Name="PasswordBox"
                        Background="Transparent"
                        ToolTip="Введите пароль"
                        Margin="0,0,333,0"
                        HorizontalAlignment="Right"
                        Width="134"
                        Height="32"
                        VerticalAlignment="Bottom"
                        Grid.Column="1"
                        Visibility="Hidden"
                        PasswordChanged="PasswordBox_PasswordChanged" />

        <TextBox x:Name="txtHintPass"
                    Grid.Column="1"
                    Margin="0,0,333,0"
                    Width="134"
                    Height="32"
                    VerticalAlignment="Bottom"
                    HorizontalAlignment="Right"
                    IsHitTestVisible="False"
                    IsReadOnly="True"
                    Background="Transparent"
                    Foreground="Gray"
                    Text="Введите пароль"
                    Visibility="Hidden" />

        <Button x:Name="ButtonEnter"
                    Margin="69,20,0,0"
                    Height="34"
                    Content="Вход"
                    ToolTip="Кнопка входа"
                    Grid.Row="1"
                    HorizontalAlignment="Left"
                    Width="120"
                    VerticalAlignment="Top"
                    Grid.Column="1"
                    Visibility="Hidden"
```

```xml
                               Click="ButtonEnter_OnClick" />

          <Button x:Name="ButtonReg"
                  Margin="69,68,0,0"
                  Height="37"
                  Content="Регистрация"
                  ToolTip="Кнопка регистрации"
                  Grid.Row="1"
                  HorizontalAlignment="Left"
                  Width="120"
                  VerticalAlignment="Top"
                  Grid.Column="1"
                  Visibility="Hidden"
                  Click="ButtonReg_Click" />

          <Button x:Name="ButtonChangePassword"
                  Margin="69,120,0,0"
                  Height="37"
                  Content="Смена пароля"
                  ToolTip="Кнопка смены пароля"
                  Grid.Row="1"
                  HorizontalAlignment="Left"
                  Width="120"
                  VerticalAlignment="Top"
                  Grid.Column="1"
                  Visibility="Hidden"
                  Click="ButtonChangePassword_Click" />

          <TextBox Grid.Column="1"
                   CommandManager.PreviewExecuted="textBox_PreviewExecuted"
                   ContextMenu="{x:Null}"
                   Visibility="Visible"
                   x:Name="captcha"
                   Height="42"
                   Foreground="Red"
                   FontStyle="Italic"
                   FontWeight="Heavy"
                   VerticalAlignment="Top"
                   Width="156"
                   Background="{x:Null}"
                   FontStretch="UltraCondensed"
                   IsReadOnly="True"
                   HorizontalContentAlignment="Center"
                   HorizontalAlignment="Right"
                   Margin="0,96,322,0"
                   VerticalContentAlignment="Center"
                   FontSize="20" />

          <TextBox Grid.Column="1"
                   Visibility="Visible"
                   x:Name="captchaInput"
                   HorizontalAlignment="Left"
                   Height="34"
                   Margin="49,183,0,0"
                   TextWrapping="Wrap"
                   Text=""
                   VerticalAlignment="Top"
                   Width="160" />

          <Button x:Name="submitCaptcha"
                  Margin="69,20,0,0"
                  Visibility="Visible"
                  Height="34"
```

```xml
                            Content="Подтвердить"
                            ToolTip="Подтвердить ввод капчи"
                            Grid.Row="1"
                            HorizontalAlignment="Left"
                            Width="120"
                            VerticalAlignment="Top"
                            Grid.Column="1"
                            Click="submitCaptcha_Click" />

        <Label Content="Капча:"
                Name="labelCaptcha"
                Margin="178,104,0,87"
                HorizontalAlignment="Left"
                Width="54"
                Visibility="Visible" />
    </Grid>
</Page>
```

Листинг №17 - Код "AuthPage.xaml.cs"

```csharp
using System;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Input;

namespace _422_Zheltobryukh.Pages
{
    public partial class AuthPage : Page
    {
        private int failedAttempts = 0;
        private User currentUser;

        public AuthPage()
        {
            InitializeComponent();
            CaptchaChange(); // Инициализация капчи при загрузке страницы
            this.Loaded += AuthPage_Loaded;
        }

        private void AuthPage_Loaded(object sender, RoutedEventArgs e)
        {
            if (captcha.Visibility == Visibility.Visible)
            {
                CaptchaSwitch();
            }
            failedAttempts = 0;
        }

        // --- Хеширование SHA256 (совпадает с RegPage) ---
        public static string GetHash(string password)
        {
            using (SHA256 sha256 = SHA256.Create())
            {
                byte[] bytes = Encoding.UTF8.GetBytes(password);
                byte[] hashBytes = sha256.ComputeHash(bytes);
                return BitConverter.ToString(hashBytes).Replace("-",
"").ToUpper();
            }
        }
```

43

```csharp
        // --- События текстовых полей и подсказок ---
        private void TextBoxLogin_TextChanged(object sender,
TextChangedEventArgs e)
        {
            txtHintLogin.Visibility =
string.IsNullOrEmpty(TextBoxLogin.Text)
                ? Visibility.Visible
                : Visibility.Hidden;
        }

        private void PasswordBox_PasswordChanged(object sender,
RoutedEventArgs e)
        {
            txtHintPass.Visibility =
string.IsNullOrEmpty(PasswordBox.Password)
                ? Visibility.Visible
                : Visibility.Hidden;
        }

        private void ButtonChangePassword_Click(object sender,
RoutedEventArgs e)
        {
            NavigationService?.Navigate(new ChangePassPage());
        }

        // --- Авторизация пользователя ---
        private void ButtonEnter_OnClick(object sender, RoutedEventArgs e)
        {
            if (string.IsNullOrEmpty(TextBoxLogin.Text) ||
                string.IsNullOrEmpty(PasswordBox.Password))
            {
                MessageBox.Show("Введите логин и пароль!");
                return;
            }

            string hashedPassword = GetHash(PasswordBox.Password);

            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                var user = db.Users
                    .AsNoTracking()
                    .FirstOrDefault(u => u.LOGIN == TextBoxLogin.Text &&
u.Password == hashedPassword);

                if (user == null)
                {
                    MessageBox.Show("Неверный логин или пароль!");
                    failedAttempts++;

                    if (failedAttempts >= 3 && captcha.Visibility !=
Visibility.Visible)
                    {
                        CaptchaSwitch();
                    }

                    return;
                }

                MessageBox.Show("Вход выполнен успешно!");

                if (NavigationService.CanGoBack)
                {
                    var mainWindow = Window.GetWindow(this) as MainWindow;
```

```csharp
                    if (mainWindow != null)
                    {
                        mainWindow.ClearNavigationHistory();
                    }
                }

                switch (user.Role)
                {
                    case "User":
                        NavigationService?.Navigate(new Pages.UserPage());
                        break;
                    case "Admin":
                        NavigationService?.Navigate(new Pages.AdminPage());
                        break;
                    default:
                        MessageBox.Show("Неизвестная роль пользователя!");
                        break;
                }
            }
        }

        // --- Переключение режима капчи ---
        public void CaptchaSwitch()
        {
            switch (captcha.Visibility)
            {
                case Visibility.Visible:
                    TextBoxLogin.Clear();
                    PasswordBox.Clear();
                    captcha.Visibility = Visibility.Hidden;
                    captchaInput.Visibility = Visibility.Hidden;
                    labelCaptcha.Visibility = Visibility.Hidden;
                    submitCaptcha.Visibility = Visibility.Hidden;
                    labelLogin.Visibility = Visibility.Visible;
                    labelPass.Visibility = Visibility.Visible;
                    TextBoxLogin.Visibility = Visibility.Visible;
                    PasswordBox.Visibility = Visibility.Visible;
                    ButtonChangePassword.Visibility = Visibility.Visible;
                    ButtonEnter.Visibility = Visibility.Visible;
                    ButtonReg.Visibility = Visibility.Visible;
                    return;

                case Visibility.Hidden:
                    CaptchaChange();
                    captcha.Visibility = Visibility.Visible;
                    captchaInput.Visibility = Visibility.Visible;
                    labelCaptcha.Visibility = Visibility.Visible;
                    submitCaptcha.Visibility = Visibility.Visible;
                    labelLogin.Visibility = Visibility.Hidden;
                    labelPass.Visibility = Visibility.Hidden;
                    TextBoxLogin.Visibility = Visibility.Hidden;
                    PasswordBox.Visibility = Visibility.Hidden;
                    ButtonChangePassword.Visibility = Visibility.Hidden;
                    ButtonEnter.Visibility = Visibility.Hidden;
                    ButtonReg.Visibility = Visibility.Hidden;
                    return;
            }
        }

        // --- Генерация новой капчи ---
        public void CaptchaChange()
        {
```

```csharp
            string allowchar =
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz1234567890";
            StringBuilder captchaBuilder = new StringBuilder();
            Random random = new Random();

            for (int i = 0; i < 6; i++)
            {
                captchaBuilder.Append(allowchar[random.Next(0,
allowchar.Length)]);
            }

            captcha.Text = captchaBuilder.ToString();
        }

        // --- Проверка капчи ---
        private void submitCaptcha_Click(object sender, RoutedEventArgs e)
        {
            if (captchaInput.Text != captcha.Text)
            {
                MessageBox.Show("Неверно введена капча!", "Ошибка");
                CaptchaChange();
            }
            else
            {
                MessageBox.Show("Капча введена успешно..", "Успех");
                CaptchaSwitch();
                failedAttempts = 0;
            }
        }

        // --- Блокировка вставки / копирования ---
        private void textBox_PreviewExecuted(object sender,
ExecutedRoutedEventArgs e)
        {
            if (e.Command == ApplicationCommands.Copy ||
                e.Command == ApplicationCommands.Cut ||
                e.Command == ApplicationCommands.Paste)
            {
                e.Handled = true;
            }
        }

        // --- Переход на страницу регистрации ---
        private void ButtonReg_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new RegPage());
        }
    }
}
```

Листинг №18 - Код "CategoryTabPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.CategoryTabPage"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    Title="CategoryTabPage">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="*" />
            <RowDefinition Height="80" />
        </Grid.RowDefinitions>
```

```xml
        <DataGrid x:Name="DataGridCategory" Grid.Row="0"
AutoGenerateColumns="False" IsReadOnly="True">
            <DataGrid.Columns>

                <DataGridTextColumn Header="Категория" Binding="{Binding
Name}" Width="300"/>
                <DataGridTemplateColumn Width="*">
                    <DataGridTemplateColumn.CellTemplate>
                        <DataTemplate>

                            <Button x:Name="ButtonEdit"
Content="Редактировать" Click="ButtonEdit_Click" ToolTip="Редактировать
данные"/>
                        </DataTemplate>
                    </DataGridTemplateColumn.CellTemplate>
                </DataGridTemplateColumn>
            </DataGrid.Columns>
        </DataGrid>

        <Button x:Name="ButtonAdd" Grid.Row="1" Content="Добавить"
HorizontalAlignment="Left" Width="100" Margin="10" Click="ButtonAdd_Click"
ToolTip="Добавить данные"/>

        <Button x:Name="ButtonDel" Grid.Row="1" Content="Удалить"
HorizontalAlignment="Right" Width="100" Margin="10" Click="ButtonDel_Click"
ToolTip="Удалить данные"/>
    </Grid>
</Page>
```

Листинг №19 - Код "CategoryTabPage.xaml.cs"

```csharp
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 _422_Zheltobryukh.Pages
{
    /// <summary>
    /// Логика взаимодействия для CategoryTabPage.xaml
    /// </summary>
    public partial class CategoryTabPage : Page
    {
        public CategoryTabPage()
        {
            InitializeComponent();
            DataGridCategory.ItemsSource = new
Zheltobryukh_DB_PaymentsEntities1().Categories.ToList();
            this.IsVisibleChanged += Page_IsVisibleChanged;
        }

        private void Page_IsVisibleChanged(object sender,
DependencyPropertyChangedEventArgs e)
```

```
            {
                if (Visibility == Visibility.Visible)
                {
                    using (var db = new Zheltobryukh_DB_PaymentsEntities1())
                    {
                        db.ChangeTracker.Entries().ToList().ForEach(x =>
x.Reload());
                        DataGridCategory.ItemsSource = db.Categories.ToList();
                    }
                }
            }

        private void ButtonEdit_Click(object sender, RoutedEventArgs e)
        {
            NavigationService.Navigate(new AddCategoryPage((sender as
Button).DataContext as Category));
        }

        private void ButtonDel_Click(object sender, RoutedEventArgs e)
        {
            var categoryForRemoving =
DataGridCategory.SelectedItems.Cast<Category>().ToList();

            if (MessageBox.Show($"Вы точно хотите удалить
{categoryForRemoving.Count()} элементов?",
                "Внимание", MessageBoxButton.YesNo,
MessageBoxImage.Question) == MessageBoxResult.Yes)
            {
                try
                {
                    using (var db = new Zheltobryukh_DB_PaymentsEntities1())
                    {
                        foreach (var category in categoryForRemoving)
                        {
                            db.Entry(category).State =
System.Data.Entity.EntityState.Deleted;
                        }
                        db.SaveChanges();
                        MessageBox.Show("Данные успешно удалены!");
                        DataGridCategory.ItemsSource =
db.Categories.ToList();
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.Message.ToString());
                }
            }
        }

        private void ButtonAdd_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new AddCategoryPage(null));
        }
    }
}
```

Листинг №20 - Код "ChangePassPage.xaml"

```
<Page x:Class="_422_Zheltobryukh.Pages.ChangePassPage"
    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:_422_Zheltobryukh.Pages"
    mc:Ignorable="d"
    d:DesignHeight="450" d:DesignWidth="800"
    Title="ChangePassPage">

    <Grid Margin="20">
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="Auto"/>
            <ColumnDefinition Width="*"/>
        </Grid.ColumnDefinitions>
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="Auto"/>
        </Grid.RowDefinitions>

        <TextBlock Text="Логин:" Grid.Row="0" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <TextBox x:Name="TbLogin" Grid.Row="0" Grid.Column="1" Margin="5" />

        <TextBlock Text="Старый пароль:" Grid.Row="1" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <PasswordBox x:Name="CurrentPasswordBox" Grid.Row="1"
Grid.Column="1" Margin="5" />

        <TextBlock Text="Новый пароль:" Grid.Row="2" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <PasswordBox x:Name="NewPasswordBox" Grid.Row="2" Grid.Column="1"
Margin="5" />

        <TextBlock Text="Подтвердите пароль:" Grid.Row="3" Grid.Column="0"
VerticalAlignment="Center" Margin="5"/>
        <PasswordBox x:Name="ConfirmPasswordBox" Grid.Row="3"
Grid.Column="1" Margin="5" />

        <Button x:Name="ButtonSave" Content="Сохранить" Grid.Row="4"
Grid.ColumnSpan="2" HorizontalAlignment="Center" Width="150" Margin="10"
Click="ButtonSave_Click"/>

    </Grid>
</Page>
```

Листинг №21 - Код "ChangePassPage.xaml.cs"

```
using System;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Navigation;

namespace _422_Zheltobryukh.Pages
{
    public partial class ChangePassPage : Page
    {
        public ChangePassPage()
        {
            InitializeComponent();
        }
```

```csharp
        public static string GetHash(string password)
        {
            using (SHA256 sha256 = SHA256.Create())
            {
                byte[] bytes = Encoding.UTF8.GetBytes(password);
                byte[] hashBytes = sha256.ComputeHash(bytes);
                return BitConverter.ToString(hashBytes).Replace("-",
"").ToUpper();
            }
        }

        private void ButtonSave_Click(object sender, RoutedEventArgs e)
        {
            if (string.IsNullOrEmpty(TbLogin.Text) ||
                string.IsNullOrEmpty(CurrentPasswordBox.Password) ||
                string.IsNullOrEmpty(NewPasswordBox.Password) ||
                string.IsNullOrEmpty(ConfirmPasswordBox.Password))
            {
                MessageBox.Show("Все поля обязательны к заполнению!");
                return;
            }

            if (NewPasswordBox.Password != ConfirmPasswordBox.Password)
            {
                MessageBox.Show("Новые пароли не совпадают!");
                return;
            }

            if (NewPasswordBox.Password.Length < 6)
            {
                MessageBox.Show("Новый пароль слишком короткий, должно быть
минимум 6 символов!");
                return;
            }

            bool hasEnglish = false;
            bool hasNumber = false;
            foreach (char c in NewPasswordBox.Password)
            {
                if ((c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z'))
                    hasEnglish = true;
                else if (c >= '0' && c <= '9')
                    hasNumber = true;
            }

            if (!hasEnglish)
            {
                MessageBox.Show("Новый пароль должен содержать только
английскую раскладку!");
                return;
            }
            if (!hasNumber)
            {
                MessageBox.Show("Новый пароль должен содержать хотя бы одну
цифру!");
                return;
            }

            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                string currentPasswordHash =
GetHash(CurrentPasswordBox.Password);
```

50

```
                        var user = db.Users.FirstOrDefault(u => u.LOGIN ==
TbLogin.Text && u.Password == currentPasswordHash);

                        if (user == null)
                        {
                            MessageBox.Show("Текущий логин или пароль неверный!");
                            return;
                        }

                        user.Password = GetHash(NewPasswordBox.Password);
                        db.SaveChanges();

                        MessageBox.Show("Пароль успешно изменен!");

                        if (NavigationService.CanGoBack)
                        {
                            NavigationService.GoBack();
                        }
                        else
                        {
                            NavigationService?.Navigate(new AuthPage());
                        }
                    }
                }
            }
}
```

Листинг №22 - Код "DiagrammPage.xaml"

```
<Page x:Class="_422_Zheltobryukh.Pages.DiagrammPage"
      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:_422_Zheltobryukh.Pages"
      xmlns:charts="clr-
namespace:System.Windows.Forms.DataVisualization.Charting;assembly=System.Wi
ndows.Forms.DataVisualization"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="DiagrammPage">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="*"/>
        </Grid.RowDefinitions>

        <StackPanel Grid.Row="0" Orientation="Horizontal" Margin="10">
            <TextBlock Text="Пользователь:" VerticalAlignment="Center"
Margin="5"/>
            <ComboBox x:Name="CmbUser" Width="200" Margin="5"
                    DisplayMemberPath="FIO"
                    SelectionChanged="UpdateChart"/>

            <TextBlock Text="Тип диаграммы:" VerticalAlignment="Center"
Margin="15,5,5,5"/>
            <ComboBox x:Name="CmbDiagram" Width="150" Margin="5"
                    SelectionChanged="UpdateChart"/>

            <Button x:Name="ButtonExcel" Content="Excel" Width="100"
Margin="15,5,5,5"
                    Click="ButtonExcel_Click" ToolTip="Экспорт в Excel"/>
```

51

```
                <Button x:Name="ButtonWord" Content="Word" Width="100"
Margin="5"
                    Click="ButtonWord_Click" ToolTip="Экспорт в Word"/>
        </StackPanel>

        <WindowsFormsHost Grid.Row="1" Margin="10">
            <charts:Chart x:Name="ChartPayments">
                <charts:Chart.Legends>
                    <charts:Legend/>
                </charts:Chart.Legends>
            </charts:Chart>
        </WindowsFormsHost>
    </Grid>
</Page>
```

Листинг №23 - Код "DiagrammPage.xaml.cs"

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Forms.DataVisualization.Charting;
using Word = Microsoft.Office.Interop.Word;
using Excel = Microsoft.Office.Interop.Excel;
using System.Windows.Navigation;

namespace _422_Zheltobryukh.Pages
{
    public partial class DiagrammPage : Page
    {
        private Zheltobryukh_DB_PaymentsEntities1 _context = new
Zheltobryukh_DB_PaymentsEntities1();

        public DiagrammPage()
        {
            InitializeComponent();

            ChartPayments.ChartAreas.Add(new ChartArea("Main"));

            var currentSeries = new Series("Платежи")
            {
                IsValueShownAsLabel = true
            };
            ChartPayments.Series.Add(currentSeries);

            CmbUser.ItemsSource = _context.Users.ToList();
            CmbDiagram.ItemsSource =
Enum.GetValues(typeof(SeriesChartType));
        }

        private void UpdateChart(object sender, SelectionChangedEventArgs e)
        {
            if (CmbUser.SelectedItem is User currentUser &&
                CmbDiagram.SelectedItem is SeriesChartType currentType)
            {
                Series currentSeries =
ChartPayments.Series.FirstOrDefault();
                currentSeries.ChartType = currentType;
                currentSeries.Points.Clear();

                var categoriesList = _context.Categories.ToList();
                foreach (var category in categoriesList)
```

```
                {
                    currentSeries.Points.AddXY(category.Name,
                        _context.Payments.ToList().Where(u =>
                            u.UserID == currentUser.ID && u.CategoryID ==
category.ID).Sum(u => u.Price * u.Num));
                }
            }
        }

        private void ButtonWord_Click(object sender, RoutedEventArgs e)
        {
            var allUsers = _context.Users.ToList();
            var allCategories = _context.Categories.ToList();

            var application = new Word.Application();
            Word.Document document = application.Documents.Add();

            try
            {
                foreach (var user in allUsers)
                {
                    Word.Paragraph userParagraph =
document.Paragraphs.Add();
                    Word.Range userRange = userParagraph.Range;
                    userRange.Text = user.FIO;
                    userParagraph.set_Style("Заголовок 1");
                    userRange.ParagraphFormat.Alignment =
Word.WdParagraphAlignment.wdAlignParagraphCenter;
                    userRange.InsertParagraphAfter();

                    document.Paragraphs.Add();

                    Word.Paragraph tableParagraph =
document.Paragraphs.Add();
                    Word.Range tableRange = tableParagraph.Range;
                    Word.Table paymentsTable =
document.Tables.Add(tableRange, allCategories.Count() + 1, 2);
                    paymentsTable.Borders.InsideLineStyle =
paymentsTable.Borders.OutsideLineStyle = Word.WdLineStyle.wdLineStyleSingle;
                    paymentsTable.Range.Cells.VerticalAlignment =
Word.WdCellVerticalAlignment.wdCellAlignVerticalCenter;

                    Word.Range cellRange;
                    cellRange = paymentsTable.Cell(1, 1).Range;
                    cellRange.Text = "Категория";
                    cellRange = paymentsTable.Cell(1, 2).Range;
                    cellRange.Text = "Сумма расходов";
                    paymentsTable.Rows[1].Range.Font.Name = "Times New
Roman";
                    paymentsTable.Rows[1].Range.Font.Size = 14;
                    paymentsTable.Rows[1].Range.Bold = 1;
                    paymentsTable.Rows[1].Range.ParagraphFormat.Alignment =
Word.WdParagraphAlignment.wdAlignParagraphCenter;

                    for (int i = 0; i < allCategories.Count(); i++)
                    {
                        var currentCategory = allCategories[i];
                        cellRange = paymentsTable.Cell(i + 2, 1).Range;
                        cellRange.Text = currentCategory.Name;
                        cellRange.Font.Name = "Times New Roman";
                        cellRange.Font.Size = 12;

                        cellRange = paymentsTable.Cell(i + 2, 2).Range;
```

```csharp
                                decimal sum = _context.Payments.ToList()
                                        .Where(p => p.UserID == user.ID && p.CategoryID
== currentCategory.ID)
                                        .Sum(p => p.Price * p.Num);
                                cellRange.Text = sum.ToString("N2") + " руб.";

                                cellRange.Font.Name = "Times New Roman";
                                cellRange.Font.Size = 12;
                            }

                            document.Paragraphs.Add();

                            Payment maxPayment = user.Payments.OrderByDescending(u
=> u.Price * u.Num).FirstOrDefault();
                            if (maxPayment != null)
                            {
                                Word.Paragraph maxPaymentParagraph =
document.Paragraphs.Add();
                                Word.Range maxPaymentRange =
maxPaymentParagraph.Range;
                                maxPaymentRange.Text = $"Самый дорогостоящий платеж
{maxPayment.Name} за {(maxPayment.Price * maxPayment.Num).ToString("N2")}
руб. от {maxPayment.Date.ToString("dd.MM.yyyy")}";
                                maxPaymentParagraph.set_Style("Подзаголовок");
                                maxPaymentRange.Font.Color =
Word.WdColor.wdColorDarkRed;
                                maxPaymentRange.InsertParagraphAfter();
                            }

                            document.Paragraphs.Add();

                            Payment minPayment = user.Payments.OrderBy(u => u.Price
* u.Num).FirstOrDefault();
                            if (minPayment != null)
                            {
                                Word.Paragraph minPaymentParagraph =
document.Paragraphs.Add();
                                Word.Range minPaymentRange =
minPaymentParagraph.Range;
                                minPaymentRange.Text = $"Самый дешевый платеж
{minPayment.Name} за {(minPayment.Price * minPayment.Num).ToString("N2")}
руб. от {minPayment.Date.ToString("dd.MM.yyyy")}";
                                minPaymentParagraph.set_Style("Подзаголовок");
                                minPaymentRange.Font.Color =
Word.WdColor.wdColorDarkGreen;
                                minPaymentRange.InsertParagraphAfter();
                            }

                            if (user != allUsers.LastOrDefault())

document.Words.Last.InsertBreak(Word.WdBreakType.wdPageBreak);
                        }

                        foreach (Word.Section section in document.Sections)
                        {
                            Word.HeaderFooter footer =
section.Footers[Word.WdHeaderFooterIndex.wdHeaderFooterPrimary];

footer.PageNumbers.Add(Word.WdPageNumberAlignment.wdAlignPageNumberCenter);
                        }

                        foreach (Word.Section section in document.Sections)
                        {
```

```
                    Word.Range headerRange =
section.Headers[Word.WdHeaderFooterIndex.wdHeaderFooterPrimary].Range;
                    headerRange.Text = DateTime.Now.ToString("dd/MM/yyyy");
                    headerRange.ParagraphFormat.Alignment =
Word.WdParagraphAlignment.wdAlignParagraphCenter;
                    headerRange.Font.ColorIndex = Word.WdColorIndex.wdBlack;
                    headerRange.Font.Size = 10;
                }

                application.Visible = true;
            }
            catch (Exception ex)
            {
                MessageBox.Show("Ошибка при экспорте в Word: " +
ex.Message);
                if (document != null)
                {
                    (document as
Word._Document).Close(Word.WdSaveOptions.wdDoNotSaveChanges);
                }
                if (application != null)
                {
                    (application as Word._Application).Quit();
                }
            }
        }

        private void ButtonExcel_Click(object sender, RoutedEventArgs e)
        {
            var allUsers = _context.Users.ToList().OrderBy(u =>
u.FIO).ToList();
            decimal grandTotal = 0;

            var application = new Excel.Application();
            application.SheetsInNewWorkbook = allUsers.Count();
            Excel.Workbook workbook =
application.Workbooks.Add(Type.Missing);

            try
            {
                for (int i = 0; i < allUsers.Count(); i++)
                {
                    int startRowIndex = 1;
                    Excel.Worksheet worksheet =
application.Worksheets.Item[i + 1];
                    worksheet.Name = allUsers[i].FIO;

                    worksheet.Cells[1][startRowIndex] = "Дата платежа";
                    worksheet.Cells[2][startRowIndex] = "Название";
                    worksheet.Cells[3][startRowIndex] = "Стоимость";
                    worksheet.Cells[4][startRowIndex] = "Количество";
                    worksheet.Cells[5][startRowIndex] = "Сумма";

                    Excel.Range columnHeaderRange =
worksheet.Range[worksheet.Cells[1][1], worksheet.Cells[5][1]];
                    columnHeaderRange.HorizontalAlignment =
Excel.XlHAlign.xlHAlignCenter;
                    columnHeaderRange.Font.Bold = true;
                    startRowIndex++;

                    var userCategories = allUsers[i].Payments.OrderBy(u =>
u.Date)
                        .GroupBy(u => u.Category).OrderBy(u => u.Key.Name);
```

```
                    foreach (var groupCategory in userCategories)
                    {
                            Excel.Range headerRange =
worksheet.Range[worksheet.Cells[1][startRowIndex],
worksheet.Cells[5][startRowIndex]];
                            headerRange.Merge();
                            headerRange.Value = groupCategory.Key.Name;
                            headerRange.HorizontalAlignment =
Excel.XlHAlign.xlHAlignCenter;
                            headerRange.Font.Italic = true;
                            startRowIndex++;

                            foreach (var payment in groupCategory)
                            {
                                worksheet.Cells[1][startRowIndex] =
payment.Date.ToString("dd.MM.yyyy");
                                worksheet.Cells[2][startRowIndex] =
payment.Name;
                                worksheet.Cells[3][startRowIndex] =
payment.Price;
                                (worksheet.Cells[3][startRowIndex] as
Excel.Range).NumberFormat = "0.00";
                                worksheet.Cells[4][startRowIndex] = payment.Num;
                                worksheet.Cells[5][startRowIndex].Formula =
$"=C{startRowIndex}*D{startRowIndex}";
                                (worksheet.Cells[5][startRowIndex] as
Excel.Range).NumberFormat = "0.00";

                                grandTotal += (payment.Price * payment.Num);
                                startRowIndex++;
                            }

                            Excel.Range sumRange =
worksheet.Range[worksheet.Cells[1][startRowIndex],
worksheet.Cells[4][startRowIndex]];
                            sumRange.Merge();
                            sumRange.Value = "ИТОГО:";
                            sumRange.HorizontalAlignment =
Excel.XlHAlign.xlHAlignRight;

                            worksheet.Cells[5][startRowIndex].Formula =
$"=SUM(E{startRowIndex - groupCategory.Count()}:E{startRowIndex - 1})";
                            sumRange.Font.Bold =
worksheet.Cells[5][startRowIndex].Font.Bold = true;
                            startRowIndex++;
                    }

                    Excel.Range rangeBorders =
worksheet.Range[worksheet.Cells[1][1], worksheet.Cells[5][startRowIndex -
1]];

rangeBorders.Borders[Excel.XlBordersIndex.xlEdgeBottom].LineStyle =
Excel.XlLineStyle.xlContinuous;

rangeBorders.Borders[Excel.XlBordersIndex.xlEdgeLeft].LineStyle =
Excel.XlLineStyle.xlContinuous;

rangeBorders.Borders[Excel.XlBordersIndex.xlEdgeRight].LineStyle =
Excel.XlLineStyle.xlContinuous;

rangeBorders.Borders[Excel.XlBordersIndex.xlEdgeTop].LineStyle =
Excel.XlLineStyle.xlContinuous;
```

```
rangeBorders.Borders[Excel.XlBordersIndex.xlInsideHorizontal].LineStyle =
Excel.XlLineStyle.xlContinuous;

rangeBorders.Borders[Excel.XlBordersIndex.xlInsideVertical].LineStyle =
Excel.XlLineStyle.xlContinuous;

                worksheet.Columns.AutoFit();
            }

            Excel.Worksheet summarySheet =
workbook.Worksheets.Add(After:
workbook.Worksheets[workbook.Worksheets.Count]);
            summarySheet.Name = "Общий итог";
            summarySheet.Cells[1, 1] = "Общий итог:";
            summarySheet.Cells[1, 2] = grandTotal;
            (summarySheet.Cells[1, 2] as Excel.Range).NumberFormat =
"0.00";

            Excel.Range summaryRange =
summarySheet.Range[summarySheet.Cells[1, 1], summarySheet.Cells[1, 2]];
            summaryRange.Font.Color = Excel.XlRgbColor.rgbRed;
            summaryRange.Font.Bold = true;
            summarySheet.Columns.AutoFit();

            application.Visible = true;
        }
        catch (Exception ex)
        {
            MessageBox.Show("Ошибка при экспорте в Excel: " +
ex.Message);
            if (workbook != null)
            {
                (workbook as Excel._Workbook).Close(false);
            }
            if (application != null)
            {
                (application as Excel._Application).Quit();
            }
        }
    }
  }
}
```

Листинг №24 - Код "PaymentTabPage.xaml"

```
<Page x:Class="_422_Zheltobryukh.Pages.PaymentTabPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      Title="PaymentTabPage">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="*" />
            <RowDefinition Height="80" />
        </Grid.RowDefinitions>

        <DataGrid x:Name="DataGridPayment" Grid.Row="0"
AutoGenerateColumns="False" IsReadOnly="True">
            <DataGrid.Columns>

                <DataGridTextColumn Header="Дата" Binding="{Binding Date,
StringFormat='dd.MM.yyyy'}" Width="100"/>
```

```
                <DataGridTextColumn Header="Пользователь" Binding="{Binding
User.FIO}" Width="150"/>
                <DataGridTextColumn Header="Название" Binding="{Binding
Name}" Width="150"/>
                <DataGridTextColumn Header="Количество" Binding="{Binding
Num}" Width="100"/>
                <DataGridTextColumn Header="Сумма" Binding="{Binding Price}"
Width="100"/>

                <DataGridTextColumn Header="Категория" Binding="{Binding
Category.Name}" Width="*"/>
                <DataGridTemplateColumn Width="120">
                    <DataGridTemplateColumn.CellTemplate>
                        <DataTemplate>
                            <Button x:Name="ButtonEdit"
Content="Редактировать" Click="ButtonEdit_Click" ToolTip="Редактировать
данные"/>
                        </DataTemplate>
                    </DataGridTemplateColumn.CellTemplate>
                </DataGridTemplateColumn>
            </DataGrid.Columns>
        </DataGrid>

        <Button x:Name="ButtonAdd" Grid.Row="1" Content="Добавить"
HorizontalAlignment="Left" Width="100" Margin="10" Click="ButtonAdd_Click"
ToolTip="Добавить данные"/>

        <Button x:Name="ButtonDel" Grid.Row="1" Content="Удалить"
HorizontalAlignment="Right" Width="100" Margin="10" Click="ButtonDel_Click"
ToolTip="Удалить данные"/>
    </Grid>
</Page>
```

Листинг №25 - Код "PaymentTabPage.xaml.cs"

```
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;
using System.Data.Entity;

namespace _422_Zheltobryukh.Pages
{
    /// <summary>
    /// Логика взаимодействия для PaymentTabPage.xaml
    /// </summary>
    public partial class PaymentTabPage : Page
    {
        public PaymentTabPage()
        {
            InitializeComponent();
            LoadData();
```

```
                    this.IsVisibleChanged += Page_IsVisibleChanged;
        }

        private void LoadData()
        {
            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                DataGridPayment.ItemsSource = db.Payments
                    .Include(p => p.User)
                    .Include(p => p.Category)
                    .ToList();
            }
        }

        private void Page_IsVisibleChanged(object sender,
DependencyPropertyChangedEventArgs e)
        {
            if (Visibility == Visibility.Visible)
            {
                using (var db = new Zheltobryukh_DB_PaymentsEntities1())
                {
                    db.ChangeTracker.Entries().ToList().ForEach(x =>
x.Reload());

                    DataGridPayment.ItemsSource = db.Payments
                        .Include(p => p.User)
                        .Include(p => p.Category)
                        .ToList();
                }
            }
        }

        private void ButtonAdd_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new AddPaymentPage(null));
        }

        private void ButtonDel_Click(object sender, RoutedEventArgs e)
        {
            var paymentForRemoving =
DataGridPayment.SelectedItems.Cast<Payment>().ToList();

            if (MessageBox.Show($"Вы точно хотите удалить
{paymentForRemoving.Count()} элементов?",
                "Внимание", MessageBoxButton.YesNo,
MessageBoxImage.Question) == MessageBoxResult.Yes)
            {
                try
                {
                    using (var db = new Zheltobryukh_DB_PaymentsEntities1())
                    {
                        foreach (var payment in paymentForRemoving)
                        {
                            db.Entry(payment).State =
System.Data.Entity.EntityState.Deleted;
                        }
                        db.SaveChanges();
                        MessageBox.Show("Данные успешно удалены!");
                        DataGridPayment.ItemsSource = db.Payments.Include(p
=> p.User).Include(p => p.Category).ToList();
                    }
                }
                catch (Exception ex)
                {
```

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

        private void ButtonEdit_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new AddPaymentPage((sender as
Button).DataContext as Payment));
        }
    }
}
```

Листинг №26 - Код "RegPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.RegPage"
      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:_422_Zheltobryukh"
      mc:Ignorable="d"
      d:DesignHeight="600" d:DesignWidth="800"
      Title="RegPage">

    <Grid>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="61*" />
            <ColumnDefinition Width="739*" />
        </Grid.ColumnDefinitions>
        <Grid.RowDefinitions>
            <RowDefinition Height="45" />
            <RowDefinition Height="*" />
            <RowDefinition Height="45" />
        </Grid.RowDefinitions>

        <Label Grid.Column="1"
               Content="Введите логин"
               HorizontalAlignment="Left"
               Margin="91,13,0,0"
               Grid.Row="1"
               VerticalAlignment="Top"
               Width="128" />
        <TextBox x:Name="txtbxLog"
                 Background="Transparent"
                 Grid.Column="1"
                 HorizontalAlignment="Left"
                 Margin="339,13,0,0"
                 Grid.Row="1"
                 VerticalAlignment="Top"
                 Width="182"
                 Height="29"
                 ToolTip="Введите логин или почту"
                 TextChanged="txtbxLog_TextChanged" />
        <Label x:Name="lblLogHitn"
               Foreground="Gray"
               Grid.Column="1"
               Content="Введите логин"
               HorizontalAlignment="Left"
               Margin="339,13,0,0"
               Grid.Row="1"
               VerticalAlignment="Top"
               Width="182"
```

60

```xml
                    Height="29"
                    MouseLeftButtonUp="lblLogHitn_MouseLeftButtonUp" />

            <Label Grid.Column="1"
                    Content="Введите пароль"
                    HorizontalAlignment="Left"
                    Margin="91,65,0,0"
                    Grid.Row="1"
                    VerticalAlignment="Top"
                    Width="128" />
            <PasswordBox x:Name="passBxFrst"
                        Background="Transparent"
                        Grid.Column="1"
                        HorizontalAlignment="Left"
                        Margin="339,65,0,0"
                        Grid.Row="1"
                        VerticalAlignment="Top"
                        Width="182"
                        Height="33"
                        ToolTip="Введите пароль"
                        PasswordChanged="passBxFrst_PasswordChanged" />
            <Label x:Name="lblPassHitn"
                    Foreground="Gray"
                    Grid.Column="1"
                    Content="Введите пароль"
                    HorizontalAlignment="Left"
                    Margin="339,67,0,0"
                    Grid.Row="1"
                    VerticalAlignment="Top"
                    Width="182"
                    Height="29"
                    MouseLeftButtonUp="lblPassHitn_MouseLeftButtonUp" />

            <Label Grid.Column="1"
                    Content="Подтвердите пароль"
                    HorizontalAlignment="Left"
                    Margin="91,118,0,0"
                    Grid.Row="1"
                    VerticalAlignment="Top"
                    Width="169" />
            <PasswordBox x:Name="passBxScnd"
                        Background="Transparent"
                        Grid.Column="1"
                        HorizontalAlignment="Left"
                        Margin="339,118,0,0"
                        Grid.Row="1"
                        VerticalAlignment="Top"
                        Width="182"
                        Height="33"
                        ToolTip="Повторите пароль"
                        PasswordChanged="passBxScnd_PasswordChanged" />
            <Label x:Name="lblPassSecHitn"
                    Foreground="Gray"
                    Grid.Column="1"
                    Content="Повторите пароль"
                    HorizontalAlignment="Left"
                    Margin="339,118,0,0"
                    Grid.Row="1"
                    VerticalAlignment="Top"
                    Width="182"
                    Height="33"
                    MouseLeftButtonUp="lblPassSecHitn_MouseLeftButtonUp" />
```

```xml
        <Label Content="Выберите роль"
               HorizontalAlignment="Left"
               Margin="91,177,0,0"
               Grid.Row="1"
               VerticalAlignment="Top"
               Width="128"
               Grid.Column="1" />
        <ComboBox x:Name="comboBxRole"
                  Grid.Column="1"
                  HorizontalAlignment="Left"
                  Margin="339,180,0,0"
                  Grid.Row="1"
                  VerticalAlignment="Top"
                  Width="182"
                  Height="37"
                  ToolTip="Выберите роль">
            <ComboBoxItem Content="User" />
            <ComboBoxItem Content="Admin" />
        </ComboBox>

        <Label Grid.Column="1"
               Content="Введите ФИО"
               HorizontalAlignment="Left"
               Margin="91,250,0,0"
               Grid.Row="1"
               VerticalAlignment="Top"
               Width="128" />
        <TextBox x:Name="txtbxFIO"
                 Background="Transparent"
                 Grid.Column="1"
                 Margin="339,243,0,0"
                 Grid.Row="1"
                 Height="43"
                 VerticalAlignment="Top"
                 HorizontalAlignment="Left"
                 Width="182"
                 ToolTip="Введите ФИО"
                 TextChanged="txtbxFIO_TextChanged" />
        <Label x:Name="lblFioHitn"
               Foreground="Gray"
               Grid.Column="1"
               Content="Введите ФИО"
               HorizontalAlignment="Left"
               Margin="339,244,0,0"
               Grid.Row="1"
               VerticalAlignment="Top"
               Width="182"
               Height="42"
               MouseLeftButtonUp="lblFioHitn_MouseLeftButtonUp" />

        <Button x:Name="regButton"
                Grid.Column="1"
                Content="Зарегистрироваться"
                HorizontalAlignment="Left"
                Margin="345,302,0,0"
                Grid.Row="1"
                VerticalAlignment="Top"
                Width="170"
                ToolTip="Нажмите для регистрации"
                Click="regButton_Click" />
    </Grid>
</Page>
```

Листинг №27 - Код "RegPage.xaml.cs"

```csharp
using System;
using System.Linq;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Input;
using System.Security.Cryptography;
using System.Text;

namespace _422_Zheltobryukh
{
    public partial class RegPage : Page
    {
        public RegPage()
        {
            InitializeComponent();
            comboBxRole.SelectedIndex = 0;
        }

        // --- Обработчики для подсказок (placeholder'ов) ---
        private void lblLogHitn_MouseLeftButtonUp(object sender,
MouseButtonEventArgs e)
        {
            txtbxLog.Focus();
        }

        private void lblPassHitn_MouseLeftButtonUp(object sender,
MouseButtonEventArgs e)
        {
            passBxFrst.Focus();
        }

        private void lblPassSecHitn_MouseLeftButtonUp(object sender,
MouseButtonEventArgs e)
        {
            passBxScnd.Focus();
        }

        private void lblFioHitn_MouseLeftButtonUp(object sender,
MouseButtonEventArgs e)
        {
            txtbxFIO.Focus();
        }

        private void txtbxLog_TextChanged(object sender,
TextChangedEventArgs e)
        {
            lblLogHitn.Visibility = string.IsNullOrEmpty(txtbxLog.Text)
                ? Visibility.Visible
                : Visibility.Hidden;
        }

        private void txtbxFIO_TextChanged(object sender,
TextChangedEventArgs e)
        {
            lblFioHitn.Visibility = string.IsNullOrEmpty(txtbxFIO.Text)
                ? Visibility.Visible
                : Visibility.Hidden;
        }
```

```csharp
        private void passBxFrst_PasswordChanged(object sender,
RoutedEventArgs e)
        {
            lblPassHitn.Visibility =
string.IsNullOrEmpty(passBxFrst.Password)
                ? Visibility.Visible
                : Visibility.Hidden;
        }

        private void passBxScnd_PasswordChanged(object sender,
RoutedEventArgs e)
        {
            lblPassSecHitn.Visibility =
string.IsNullOrEmpty(passBxScnd.Password)
                ? Visibility.Visible
                : Visibility.Hidden;
        }

        // --- Основная логика регистрации ---
        private void regButton_Click(object sender, RoutedEventArgs e)
        {
            // 1  Проверка заполнения всех полей
            if (string.IsNullOrEmpty(txtbxLog.Text) ||
                string.IsNullOrEmpty(txtbxFIO.Text) ||
                string.IsNullOrEmpty(passBxFrst.Password) ||
                string.IsNullOrEmpty(passBxScnd.Password))
            {
                MessageBox.Show("Заполните все поля!");
                return;
            }

            // 2 Проверка совпадения паролей
            if (passBxFrst.Password != passBxScnd.Password)
            {
                MessageBox.Show("Пароли не совпадают!");
                return;
            }

            // 3 Проверка длины пароля
            if (passBxFrst.Password.Length < 6)
            {
                MessageBox.Show("Пароль слишком короткий, должно быть
минимум 6 символов!");
                return;
            }

            // 4 Проверка символов в пароле
            bool hasEnglish = false;
            bool hasNumber = false;

            foreach (char c in passBxFrst.Password)
            {
                if ((c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z'))
                    hasEnglish = true;
                else if (c >= '0' && c <= '9')
                    hasNumber = true;
            }

            if (!hasEnglish)
            {
                MessageBox.Show("Используйте только английскую раскладку!");
                return;
```

64

```
            }

            if (!hasNumber)
            {
                MessageBox.Show("Добавьте хотя бы одну цифру!");
                return;
            }

            // 5 Проверка существующего логина
            using (var db = new Zheltobryukh_DB_PaymentsEntities1())
            {
                var user = db.Users
                    .AsNoTracking()
                    .FirstOrDefault(u => u.LOGIN == txtbxLog.Text);

                if (user != null)
                {
                    MessageBox.Show("Пользователь с таким логином уже
существует!");
                    return;
                }

                // 6 Создание нового пользователя с хешированием SHA256
                User newUser = new User
                {
                    FIO = txtbxFIO.Text,
                    LOGIN = txtbxLog.Text,
                    Password = GetHash(passBxFrst.Password), // ✅ хешируем
перед сохранением
                    Role = comboBxRole.Text
                };

                db.Users.Add(newUser);
                db.SaveChanges();
            }

            // 7 Очистка полей после регистрации
            MessageBox.Show("Пользователь успешно зарегистрирован!");

            txtbxLog.Clear();
            passBxFrst.Clear();
            passBxScnd.Clear();
            txtbxFIO.Clear();
            comboBxRole.SelectedIndex = 0;
        }

        // --- Метод хеширования SHA256 ---
        public static string GetHash(string password)
        {
            using (SHA256 sha256 = SHA256.Create())
            {
                byte[] bytes = Encoding.UTF8.GetBytes(password);
                byte[] hashBytes = sha256.ComputeHash(bytes);
                return BitConverter.ToString(hashBytes).Replace("-",
"").ToUpper();
            }
        }
    }
}
```

65

Листинг №28 - Код "UserPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.UserPage"
      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"
      mc:Ignorable="d"
      Title="UserPage">

    <Grid>
        <!-- ===== Разделение на строки ===== -->
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="*"/>
        </Grid.RowDefinitions>

        <!-- ===== Панель фильтров ===== -->
        <WrapPanel Orientation="Horizontal"
                   HorizontalAlignment="Center"
                   Margin="10"
                   Grid.Row="0">

            <TextBlock Text="Введите ФИО для поиска:"
                       FontSize="12"
                       VerticalAlignment="Center"/>

            <TextBox x:Name="fioFilterTextBox"
                     Margin="7,7,0,7"
                     MinWidth="120"
                     TextChanged="fioFilterTextBox_TextChanged"/>

            <TextBlock Text="Сортировка по ФИО:"
                       Margin="15,0,0,0"
                       FontSize="12"
                       VerticalAlignment="Center"/>

            <ComboBox x:Name="sortComboBox"
                      Margin="7,0,0,0"
                      SelectedIndex="0"
                      VerticalAlignment="Center"
                      VerticalContentAlignment="Center"
                      SelectionChanged="sortComboBox_SelectionChanged"
                      MinWidth="140">
                <ComboBoxItem Content="По возрастанию"/>
                <ComboBoxItem Content="По убыванию"/>
            </ComboBox>

            <CheckBox x:Name="onlyAdminCheckBox"
                      Content="Только администраторы"
                      IsChecked="False"
                      VerticalContentAlignment="Center"
                      Margin="15,0,0,0"
                      Checked="onlyAdminCheckBox_Checked"
                      Unchecked="onlyAdminCheckBox_Unchecked"/>

            <Button x:Name="clearFiltersButton"
                    Content="Очистить фильтр"
                    VerticalAlignment="Center"
                    HorizontalAlignment="Center"
                    Padding="10,5"
                    Margin="15,0,0,0"
                    FontSize="12"
                    Click="clearFiltersButton_Click_1"/>
```

66

```xml
        </WrapPanel>

        <!-- ===== Список пользователей ===== -->
        <ListView x:Name="ListUser"
                  Grid.Row="1"
                  HorizontalContentAlignment="Center"
                  ScrollViewer.HorizontalScrollBarVisibility="Disabled"
                  Margin="10">

            <ListView.ItemsPanel>
                <ItemsPanelTemplate>
                    <WrapPanel Orientation="Horizontal"
                               HorizontalAlignment="Center"
                               Margin="10"/>
                </ItemsPanelTemplate>
            </ListView.ItemsPanel>

            <ListView.ItemTemplate>
                <DataTemplate>
                    <Grid Margin="10">
                        <Grid.RowDefinitions>
                            <RowDefinition Height="Auto"/>
                            <RowDefinition Height="200"/>
                            <RowDefinition Height="Auto"/>
                        </Grid.RowDefinitions>

                        <!-- ФИО пользователя -->
                        <TextBlock Text="{Binding FIO}"
                                   Grid.Row="0"
                                   VerticalAlignment="Center"
                                   TextAlignment="Center"
                                   Width="200"
                                   TextWrapping="Wrap"
                                   HorizontalAlignment="Center"
                                   FontSize="16"
                                   FontWeight="Bold"/>

                        <!-- Фото -->
                        <Image Width="200"
                               Height="200"
                               Grid.Row="1"
                               Stretch="UniformToFill"
                               HorizontalAlignment="Center">
                            <Image.Source>
                                <Binding Path="Photo">
                                    <Binding.TargetNullValue>
                                        <!-- ◆ Укажи путь к своей
стандартной картинке -->

<ImageSource>images/default.png</ImageSource>
                                    </Binding.TargetNullValue>
                                </Binding>
                            </Image.Source>
                        </Image>

                        <!-- Роль -->
                        <TextBlock Text="{Binding Role}"
                                   Grid.Row="2"
                                   VerticalAlignment="Center"
                                   TextAlignment="Center"
                                   Width="200"
                                   HorizontalAlignment="Center"
                                   FontSize="14"
```

```
                                        Foreground="Gray"/>
                    </Grid>
                </DataTemplate>
            </ListView.ItemTemplate>
        </ListView>
    </Grid>
</Page>
```

Листинг №29 - Код "UserPage.xaml.cs"

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows;
using System.Windows.Controls;

namespace _422_Zheltobryukh.Pages
{
    public partial class UserPage : Page
    {
        public UserPage()
        {
            InitializeComponent();
            UpdateUsers(); // сразу загружаем пользователей
        }

        // ===== Очистка фильтров =====
        private void clearFiltersButton_Click_1(object sender,
RoutedEventArgs e)
        {
            fioFilterTextBox.Text = "";
            sortComboBox.SelectedIndex = 0;
            onlyAdminCheckBox.IsChecked = false;
            UpdateUsers();
        }

        // ===== Обновление списка при вводе текста =====
        private void fioFilterTextBox_TextChanged(object sender,
TextChangedEventArgs e)
        {
            UpdateUsers();
        }

        // ===== Обновление при смене сортировки =====
        private void sortComboBox_SelectionChanged(object sender,
SelectionChangedEventArgs e)
        {
            UpdateUsers();
        }

        // ===== Фильтр только администраторы =====
        private void onlyAdminCheckBox_Checked(object sender,
RoutedEventArgs e)
        {
            UpdateUsers();
        }

        private void onlyAdminCheckBox_Unchecked(object sender,
RoutedEventArgs e)
        {
            UpdateUsers();
        }
```

```csharp
        // ===== Основной метод обновления данных =====
        private void UpdateUsers()
        {
            if (!IsInitialized)
                return;

            try
            {
                using (var db = new Zheltobryukh_DB_PaymentsEntities1())
                {
                    List<User> currentUsers = db.Users.ToList();

                    // ◆ Фильтрация по ФИО
                    if (!string.IsNullOrWhiteSpace(fioFilterTextBox.Text))
                    {
                        currentUsers = currentUsers
                            .Where(x => x.FIO != null &&
x.FIO.ToLower().Contains(fioFilterTextBox.Text.ToLower()))
                            .ToList();
                    }

                    // ◆ Фильтр по роли (только админы)
                    if (onlyAdminCheckBox.IsChecked == true)
                    {
                        currentUsers = currentUsers
                            .Where(x => x.Role == "Admin")
                            .ToList();
                    }

                    // ◆ Сортировка (0 — по возрастанию, 1 — по убыванию)
                    if (sortComboBox.SelectedIndex == 0)
                        currentUsers = currentUsers.OrderBy(x =>
x.FIO).ToList();
                    else
                        currentUsers = currentUsers.OrderByDescending(x =>
x.FIO).ToList();

                    // ◆ Отображение списка
                    ListUser.ItemsSource = currentUsers;
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show($"Ошибка при обновлении списка
пользователей:\n{ex.Message}",
                    "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error);
            }
        }
    }
}
```

Листинг №30 - Код "UsersTabPage.xaml"

```xml
<Page x:Class="_422_Zheltobryukh.Pages.UsersTabPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      Title="UsersTabPage">
    <Grid>
        <Grid.RowDefinitions>
```

```xml
                    <RowDefinition Height="*" />
                    <RowDefinition Height="80" />
            </Grid.RowDefinitions>

            <DataGrid Grid.Row="0" x:Name="DataGridUser"
                        AutoGenerateColumns="False" IsReadOnly="True"
                        RowHeight="90">

                <DataGrid.Columns>
                    <DataGridTextColumn Header="Логин" Binding="{Binding LOGIN}"
Width="100"/>
                    <DataGridTextColumn Header="Пароль" Binding="{Binding
Password}" Width="100"/>
                    <DataGridTextColumn Header="Роль" Binding="{Binding Role}"
Width="80"/>
                    <DataGridTextColumn Header="ФИО" Binding="{Binding FIO}"
Width="200"/>

                    <DataGridTemplateColumn Header="Фото" Width="100">
                        <DataGridTemplateColumn.CellTemplate>
                            <DataTemplate>
                                <Image Height="80" Stretch="UniformToFill"
Margin="5">
                                    <Image.Source>
                                        <Binding Path="Photo">
                                            <Binding.TargetNullValue>
<ImageSource>/Pages/images/default.png</ImageSource>
                                            </Binding.TargetNullValue>
                                        </Binding>
                                    </Image.Source>
                                </Image>
                            </DataTemplate>
                        </DataGridTemplateColumn.CellTemplate>
                    </DataGridTemplateColumn>

                    <DataGridTemplateColumn Width="100">
                        <DataGridTemplateColumn.CellTemplate>
                            <DataTemplate>
                                <Button Content="Редактировать"
Click="ButtonEdit_Click"/>
                            </DataTemplate>
                        </DataGridTemplateColumn.CellTemplate>
                    </DataGridTemplateColumn>
                </DataGrid.Columns>
            </DataGrid>

            <Button Grid.Row="1" Content="Добавить" HorizontalAlignment="Left"
Width="100" Margin="10"
                    Click="ButtonAdd_Click"/>
            <Button Grid.Row="1" Content="Удалить" HorizontalAlignment="Right"
Width="100" Margin="10"
                    Click="ButtonDel_Click"/>
    </Grid>
</Page>
```

Листинг №31 - Код "UsersTabPage.xaml.cs"

```csharp
using System;
using System.Linq;
using System.Windows;
using System.Windows.Controls;
```

```csharp
using System.Collections.Generic;

namespace _422_Zheltobryukh.Pages
{
    public partial class UsersTabPage : Page
    {
        private Zheltobryukh_DB_PaymentsEntities1 _context = new
Zheltobryukh_DB_PaymentsEntities1();

        public UsersTabPage()
        {
            InitializeComponent();
            DataGridUser.ItemsSource = _context.Users.ToList();
            this.IsVisibleChanged += Page_IsVisibleChanged;
        }

        private void Page_IsVisibleChanged(object sender,
DependencyPropertyChangedEventArgs e)
        {
            if (Visibility == Visibility.Visible)
            {
                _context.ChangeTracker.Entries().ToList().ForEach(x =>
x.Reload());
                DataGridUser.ItemsSource = _context.Users.ToList();
            }
        }

        private void ButtonAdd_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new AddUserPage(null));
        }

        private void ButtonDel_Click(object sender, RoutedEventArgs e)
        {
            var usersForRemoving =
DataGridUser.SelectedItems.Cast<User>().ToList();

            if (MessageBox.Show($"Вы точно хотите удалить записи в
количестве {usersForRemoving.Count()} элементов?",
                "Внимание", MessageBoxButton.YesNo,
MessageBoxImage.Question) == MessageBoxResult.Yes)
            {
                try
                {
                    _context.Users.RemoveRange(usersForRemoving);
                    _context.SaveChanges();
                    MessageBox.Show("Данные успешно удалены!");
                    DataGridUser.ItemsSource = _context.Users.ToList();
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.Message.ToString());
                }
            }
        }
        private void ButtonEdit_Click(object sender, RoutedEventArgs e)
        {
            NavigationService?.Navigate(new AddUserPage((sender as
Button).DataContext as User));
        }
    }
}
```