Федеральное государственное образовательное бюджетное учреждение  
высшего образования

«Финансовый университет при Правительстве Российской Федерации»

КОЛЛЕДЖ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

**ОТЧЕТ**

**по результатам проектирования**

**и разработки программного приложения**

по профессиональному модулю ПМ.03 Участие в интеграции программных модулей

специальности 09.02.03 Программирование в компьютерных системах

Выполнил:

студент группы 4ПКС-220

2023

Разоренов С. Т.

Проверили:

преподаватели

\_\_\_\_\_\_\_\_ / Т.Г. Аксёнова /

\_\_\_\_\_\_\_\_/О.А. Тимакин/

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_

# Содержание

[**Содержание 2**](#_Toc154589651)

[**Проектирование программного приложения 3**](#_Toc154589652)

[**Руководство оператора 6**](#_Toc154589653)

[**Листинг программных модулей 10**](#_Toc154589654)

# Проектирование программного приложения

Ниже представлены разработанные мною ER-диаграмма (см. рисунок 1) базы данных и диаграмма вариантов использования (см. рисунок 2) для заданной предметной области.

Диаграмма сущность-связь состоит из трех таблиц базы данных, их полей и типов, что они хранят, в данном случае это таблицы категории (Categories), продавцы (Sellers) и объявления (Announcements), пользователи (Users) таблица Announcements является главной и связана с двумя другими таблицами отношением 1 к 1.

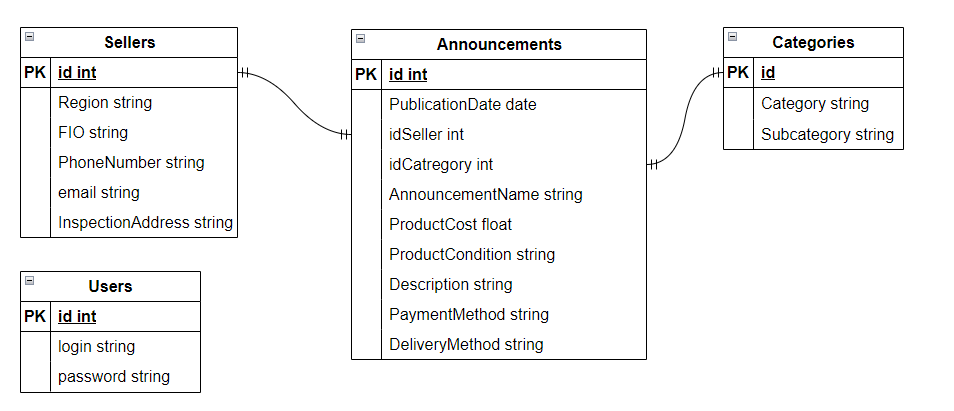


Рисунок 1. Диаграмма сущность-связь (ERD)

Далее представлена диаграмма вариантов использования, которой отображает разницу в функционале между гостем и зарегистрированным пользователем программы, как по ней видно гость не может создавать собственные объявления, но может просматривать объявления других продавцов.



Рисунок 2. Диаграмма вариантов использования (use case diagram)

Приложение должно содержать следующую информацию:

- сведения о категориях (id категории, категория, подкатегория)

- сведения о продавцах (id продавца, регион, Ф.И.О., телефон, e-mail, адрес осмотра)

- объявление (дата публикации, id продавца, id категории, название объявления, стоимость товара, описание, способ оплаты, способ доставки)

# Руководство оператора

Стартовое окно программы выглядит образом, представленным на рисунке 3. Для начала работы пользователю следует нажать кнопку «Просмотреть объявления» если он хочет посмотреть доступные объявления. В том случае если пользователь хочет добавить собственное объявление ему следует зарегистрироваться посредством нажатия на кнопку «Зарегистрироваться» в том случае если у него нет аккаунта или «Войти» если аккаунт имеется.

**Изображение выглядит как текст, снимок экрана, программное обеспечение, компьютер

Автоматически созданное описание**

Рисунок 1. главная страница неавторизированного пользователя

При переходе на страницу регистрации (см. рисунок 4) пользователь должен ввести логин, пароль и подтвердить корректность пароля. В том случае если пользователь не заполнил по крайней мере одно поле – программа выведет соответствующую ошибку. Если же данные введены успешно, программа подтвердит успешную регистрацию (см. рисунок 5) и переведет пользователя на главную с расширенным функционалом.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 2. страница регистрации пользователя

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 5. регистрация пользователя в системе

При переходе на страницу авторизации (см. рисунки 6 и 7) пользователь должен ввести свой логин и пароль, после чего нажать на кнопку «Войти», после чего его также переведет на страницу зарегистрированного пользователя, или выдаст ошибку в зависимости от входных данных.

Изображение выглядит как снимок экрана, текст, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 6. страница авторизации пользователя

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 7. пример успешного входа в систему

При нажатии на кнопку «Посмотреть объявления» пользователя переведет на страницу просмотра объявлений (см. рисунок 8), на которой доступны сортировка и поиск (см. рисунок 9).

**Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание**

Рисунок 8. просмотр объявлений

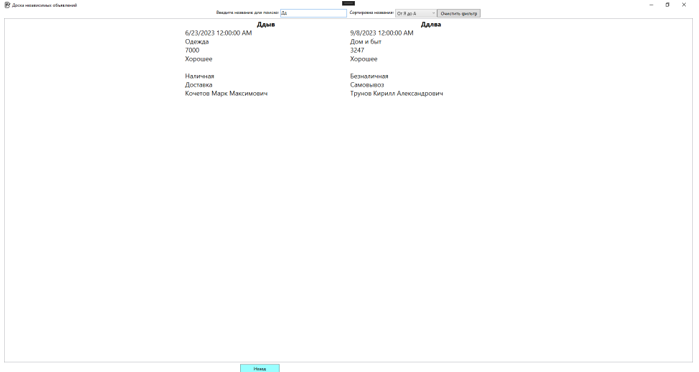


Рисунок 9. альтернативный просмотр, сортировка и поиск

Если пользователь хочет добавить объявление ему следует нажать на кнопку «Добавить объявление» и ввести всю необходимую информацию, а также выбрать продавца и категорию товара, пример приведен на рисунке 10.

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

Автоматически созданное описание

Рисунок 10. добавление объявления

# Листинг программных модулей

Листинг xaml MainWindow:

<Window x:Class="EXAMRAZORENOV.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:EXAMRAZORENOV"

mc:Ignorable="d"

Title="Доска независимых объявлений" Height="450" Width="800" FontFamily="Comic Sans MS">

<Grid>

<Grid.RowDefinitions>

<RowDefinition></RowDefinition>

</Grid.RowDefinitions>

<Frame Name="MainFrame" Grid.Row="0" HorizontalAlignment="Stretch" VerticalAlignment="Stretch" Source="Pages/Main.xaml" NavigationUIVisibility="Hidden"></Frame>

</Grid>

</Window>

Листинг cs MainWindow:using System;

using System.Collections.Generic;

using System.Linq;

using System.Security.Cryptography;

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 EXAMRAZORENOV

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

MainFrame.Content = new Pages.Main();

}

}

}

Листинг xaml Registration:  
<Page x:Class="EXAMRAZORENOV.Pages.Registration"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="Registration">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="30\*"></RowDefinition>

<RowDefinition Height="37\*"></RowDefinition>

<RowDefinition Height="23\*"></RowDefinition>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

</Grid.ColumnDefinitions>

<Label Grid.Column="1" HorizontalContentAlignment="Center" Height="150" VerticalAlignment="Center" FontFamily="Comic Sans MS">Регистрация нового пользователя</Label>

<Label Grid.Column="1" Grid.Row="1" HorizontalContentAlignment="Center" Margin="55,0,55,150" Height="25" VerticalAlignment="Bottom" FontFamily="Comic Sans MS">Введите ваш логин</Label>

<TextBox Name ="LoginTextBox" Grid.Column="1" Grid.Row="1" HorizontalContentAlignment="Center" Margin="73,0,73,125" Height="25" VerticalAlignment="Bottom" FontFamily="Comic Sans MS"/>

<Label Grid.Column="1" Grid.Row="1" HorizontalContentAlignment="Center" Margin="73,0,73,100" Height="25" VerticalAlignment="Bottom" FontFamily="Comic Sans MS">Введите пароль</Label>

<Label Grid.Column="1" Grid.Row="1" VerticalAlignment="Bottom" HorizontalContentAlignment="Center" Margin="70,0,70,54" Height="26" FontFamily="Comic Sans MS">Повторите пароль</Label>

<TextBox Name="PasswordTextBox" Grid.Column="1" Grid.Row="1" HorizontalContentAlignment="Center" Height="25" Margin="73,0,73,75" VerticalAlignment="Bottom" FontFamily="Comic Sans MS"/>

<TextBox Name="ConfirmationTextBox" Grid.Column="1" Grid.Row="1" HorizontalContentAlignment="Center" Margin="73,0,73,29" Height="25" VerticalAlignment="Bottom" FontFamily="Comic Sans MS"/>

<Button Name="RegButton" Content="Зарегистрироваться" Click="RegButton\_Click" Grid.Row="2" Grid.Column="1" Margin="0,0,0,58" Height="30" VerticalAlignment="Bottom" FontFamily="Comic Sans MS" Background="#99FFFF" HorizontalAlignment="Center" Width="130"/>

<Button Name ="CloseButton" Content="Отмена" Click="CloseButton\_Click" Grid.Row="2" Grid.Column="1" Margin="0,62,0,0" Height="30" VerticalAlignment="Top" FontFamily="Comic Sans MS" HorizontalAlignment="Center" Width="130" Background="#99FFFF"/>

</Grid>

</Page>

Листинг cs Registration:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Text;

using System.Text.RegularExpressions;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using EXAMRAZORENOV.Entities;

namespace EXAMRAZORENOV.Pages

{

public partial class Registration : Page

{

public Registration()

{

InitializeComponent();

}

private bool CorrectPasswordCheck()

{

string Password = PasswordTextBox.Text;

int NumberCount = 0;

bool ValidCharsFound(string str)

{

string pattern = @"[a-zA-z]{1}\w";

return Regex.IsMatch(str, pattern);

}

for (int i = 0; i < Password.Length; i++)

{

if (char.IsNumber(Password[i]))

{

NumberCount++;

}

}

if ((Password.Length >= 6 && NumberCount > 0 && ValidCharsFound(PasswordTextBox.Text) == true))

{

return true;

}

return false;

}

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

{

if (string.IsNullOrEmpty(LoginTextBox.Text))

{

MessageBox.Show("Введите логин!", "Ошибка!");

return;

}

if (string.IsNullOrEmpty(PasswordTextBox.Text))

{

MessageBox.Show("Введите пароль!", "Ошибка!");

return;

}

if (string.IsNullOrEmpty(ConfirmationTextBox.Text))

{

MessageBox.Show("Подтвердите пароль!", "Ошибка!");

return;

}

if (PasswordTextBox.Text != ConfirmationTextBox.Text)

{

MessageBox.Show("Пароли не совпадают!", "Ошибка!");

return;

}

if (CorrectPasswordCheck() == false)

{

MessageBox.Show("Пароль не соответствует требованиям!", "Ошибка!");

return;

}

using (var db = new TRPOEntities())

{

var CheckUser = db.Users.AsNoTracking().FirstOrDefault(u => u.login == LoginTextBox.Text);

if (CheckUser != null)

{

MessageBox.Show("Этот логин уже используется в системе!", "Ошибка!", MessageBoxButton.OK);

return;

}

Users user = new Users

{

login = LoginTextBox.Text,

password = PasswordTextBox.Text,

};

db.Users.Add(user);

db.SaveChanges();

}

MessageBox.Show("Регистрация прошла успешно!", "Поздравляем!");

NavigationService.Navigate(new Pages.Auth());

return;

}

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

{

NavigationService.GoBack();

}

}

}

Листинг xaml ProductInfo:

<Page x:Class="EXAMRAZORENOV.Pages.ProductInfo"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="ProductInfo">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="25"/>

<RowDefinition Height="Auto"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<WrapPanel HorizontalAlignment="Center">

<TextBlock Text="Введите название для поиска: " FontFamily="Comic Sans MS"/>

<TextBox x:Name="SearchName" Width="186" TextChanged="SearchName\_TextChanged"/>

<TextBlock Text=" Сортировка названия: " FontFamily="Comic Sans MS"/>

<ComboBox x:Name="CmbBoxSortName" Width="117" SelectionChanged="CmbBoxSortName\_SelectionChanged">

<TextBlock>От А до Я</TextBlock>

<TextBlock>От Я до А</TextBlock>

</ComboBox>

<Button x:Name="Clear" Content="Очистить фильтр" HorizontalContentAlignment="Center" Width="121" Click="Clear\_Click" FontFamily="Comic Sans MS"/>

</WrapPanel>

<ListView Name="UserList" Grid.Row="2" ScrollViewer.HorizontalScrollBarVisibility="Disabled" HorizontalContentAlignment="Center" Margin="0,0,0,34">

<ListView.ItemsPanel>

<ItemsPanelTemplate>

<WrapPanel Orientation="Horizontal" HorizontalAlignment="Center"></WrapPanel>

</ItemsPanelTemplate>

</ListView.ItemsPanel>

<ListView.ItemTemplate>

<DataTemplate>

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="auto"></RowDefinition>

<RowDefinition Height="auto"></RowDefinition>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

</Grid.RowDefinitions>

<TextBlock VerticalAlignment="Center" TextAlignment="Center" Width="450" Grid.Row="0" TextWrapping="Wrap" Text="{Binding AnnouncementName}" HorizontalAlignment="Center" FontSize="18" FontWeight="Bold"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="1" TextWrapping="Wrap" Text="{Binding PublicationDate}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="2" TextWrapping="Wrap" Text="{Binding Categories.Category}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="3" TextWrapping="Wrap" Text="{Binding ProductCost}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="4" TextWrapping="Wrap" Text="{Binding ProductCondition}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="5" TextWrapping="Wrap" Text="{Binding Description}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="6" TextWrapping="Wrap" Text="{Binding PaymentMethod}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="7" TextWrapping="Wrap" Text="{Binding DeliveryMethod}" HorizontalAlignment="Left" FontSize="18"/>

<TextBlock VerticalAlignment="Center" TextAlignment="Left" Width="450" Grid.Row="8" TextWrapping="Wrap" Text="{Binding Sallers.FIO}" HorizontalAlignment="Left" FontSize="18"/>

</Grid>

</DataTemplate>

</ListView.ItemTemplate>

</ListView>

<Button x:Name="Back" Content="Назад" Height="25" Margin="658,0,0,4" Grid.Row="2" Click="Back\_Click" FontFamily="Comic Sans MS" HorizontalAlignment="Left" Width="110" VerticalAlignment="Bottom" Background="#99FFFF"/>

</Grid>

</Page>

Листинг cs ProductInfo:

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 EXAMRAZORENOV.Entities;

namespace EXAMRAZORENOV.Pages

{

/// <summary>

/// Логика взаимодействия для ProductInfo.xaml

/// </summary>

public partial class ProductInfo : Page

{

public ProductInfo()

{

InitializeComponent();

var currentUsers = TRPOEntities.GetContext().Announcement.ToList();

UserList.ItemsSource = currentUsers;

CmbBoxSortName.SelectedIndex = 0;

}

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

{

ClearFilters();

}

private void SearchName\_TextChanged(object sender, TextChangedEventArgs e)

{

UpdateProduct();

}

private void CmbBoxSortName\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

UpdateProduct();

}

private void UpdateProduct()

{

//загружаем всех пользователей в список

var currentUsers = TRPOEntities.GetContext().Announcement.ToList();

//осуществляем поиск по названию без учета регистра букв

currentUsers = currentUsers.Where(x => x.AnnouncementName.ToLower().Contains(SearchName.Text.ToLower())).ToList();

//осуществляем сортировку в зависимости от выбора пользователя

if (CmbBoxSortName.SelectedIndex == 0)

UserList.ItemsSource = currentUsers.OrderBy(x => x.AnnouncementName).ToList();

else UserList.ItemsSource = currentUsers.OrderByDescending(x => x.AnnouncementName).ToList();

}

private void ClearFilters()

{

SearchName.Text = "";

CmbBoxSortName.Text = "";

}

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

{

this.NavigationService.GoBack();

}

}

}

Листинг xaml Main:

<Page x:Class="EXAMRAZORENOV.Pages.Main"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="Main">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="60"/>

<RowDefinition Height="11\*"/>

<RowDefinition Height="60"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="200"></ColumnDefinition>

<ColumnDefinition Width="150\*"></ColumnDefinition>

<ColumnDefinition Width="200"></ColumnDefinition>

</Grid.ColumnDefinitions>

<Button FontFamily="Comic Sans MS" Name="Reg" Content="Зарегистрироваться" Grid.Row="0" Grid.Column="4" Click="RegClick" Margin="40,8,33,32" Background="#99FFFF"/>

<Button FontFamily="Comic Sans MS" Name="Log" Content="Войти" Click="LogClick" Grid.Row="0" Grid.Column="4" Margin="40,32,33,8" Background="#99FFFF"/>

<Button Name="ProductInfo" Content="Посмотреть объявления" Height="30" Click="ProductInfo\_Click" VerticalAlignment="Center" HorizontalAlignment="Center" Width="150" FontFamily="Comic Sans MS" Grid.Column="1" Background="#99FFFF"/>

</Grid>

</Page>

Листинг cs Main:

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 EXAMRAZORENOV.Pages

{

public partial class Main : Page

{

public Main()

{

InitializeComponent();

}

private void RegClick(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Pages.Registration());

}

private void LogClick(object sender, RoutedEventArgs e)

{

NavigationService.Navigate(new Pages.Auth());

}

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

{

this.NavigationService.Navigate(new Uri("/Pages/ProductInfo.xaml", UriKind.Relative));

}

}

}

Листинг xaml AuthUserMain:

<Page x:Class="EXAMRAZORENOV.Pages.AuthUserMain"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="AuthUserMain">

<Grid>

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="60"/>

<RowDefinition Height="11"/>

<RowDefinition Height="60"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="200"></ColumnDefinition>

<ColumnDefinition Width="3\*"></ColumnDefinition>

<ColumnDefinition Width="200"></ColumnDefinition>

</Grid.ColumnDefinitions>

<Button Name="AddAnnouncement" Content="Добавить объявление" Height="30" Click="AddAnnouncement\_Click" VerticalAlignment="Center" HorizontalAlignment="Center" Width="150" FontFamily="Comic Sans MS" Background="#99FFFF"/>

<Button Name="ProductInfo" Content="Посмотреть объявления" Height="30" VerticalAlignment="Center" HorizontalAlignment="Center" Width="150" Grid.Column="1" Click="ProductInfo\_Click" FontFamily="Comic Sans MS" Background="#99FFFF"/>

</Grid>

</Grid>

</Page>

Листинг cs AuthUserMain:

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 EXAMRAZORENOV.Pages

{

/// <summary>

/// Логика взаимодействия для AuthUserMain.xaml

/// </summary>

public partial class AuthUserMain : Page

{

public AuthUserMain()

{

InitializeComponent();

}

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

{

this.NavigationService.Navigate(new Uri("/Pages/AddProduct.xaml", UriKind.Relative));

}

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

{

this.NavigationService.Navigate(new Uri("/Pages/ProductInfo.xaml", UriKind.Relative));

}

}

}

Листинг xaml Auth:

<Page x:Class="EXAMRAZORENOV.Pages.Auth"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="Auth">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

</Grid.ColumnDefinitions>

<Label Grid.Row="0" Grid.Column="1" HorizontalAlignment="Center" Margin="0,0,0,122" Width="267" HorizontalContentAlignment="Center" FontFamily="Comic Sans MS">Вход в систему</Label>

<Label Grid.Row="1" Grid.Column="1" VerticalAlignment="Center" HorizontalAlignment="Right" Width="266" FontFamily="Comic Sans MS">Введите логин:</Label>

<TextBox Grid.Row="1" Grid.Column="1" Height="26" Width="120" Name="LoginTextBox" VerticalAlignment="Center" Margin="115,0,0,0" HorizontalAlignment="Left"></TextBox>

<Label Grid.Row="1" Grid.Column="1" VerticalAlignment="Bottom" HorizontalAlignment="Center" Width="266" FontFamily="Comic Sans MS">Введите пароль:</Label>

<PasswordBox Name="PasswordBox" VerticalAlignment="Bottom" Grid.Row="1" Grid.Column="1" Height="25" Width="120" Margin="115,0,0,1" HorizontalAlignment="Left"></PasswordBox>

<Button FontFamily="Comic Sans MS" Content="Войти" Name="LogButton" Grid.Column="1" Grid.Row="2" Height="30" Width="110" Margin="149,0,0,0" VerticalAlignment="Center" Click="LogButton\_Click" HorizontalAlignment="Left" Background="#99FFFF"></Button>

<Button FontFamily="Comic Sans MS" Content="Отмена" Name="CancelButton" Grid.Column="1" Grid.Row="2" Height="30" Width="110" Margin="10,0,0,0" VerticalAlignment="Center" Click="CancelButton\_Click" HorizontalAlignment="Left" Background="#99FFFF"></Button>

</Grid>

</Page>

Листинг cs Auth:

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 EXAMRAZORENOV.Entities;

namespace EXAMRAZORENOV.Pages

{

/// <summary>

/// Логика взаимодействия для Auth.xaml

/// </summary>

public partial class Auth : Page

{

public Auth()

{

InitializeComponent();

}

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

{

if (string.IsNullOrEmpty(LoginTextBox.Text) || string.IsNullOrEmpty(PasswordBox.Password))

{

MessageBox.Show("Введите логин и пароль!", "Ошибка!");

return;

}

using (var db = new TRPOEntities())

{

var user = db.Users

.AsNoTracking()

.FirstOrDefault(u => u.login == LoginTextBox.Text && u.password == PasswordBox.Password);

if (user == null)

{

MessageBox.Show("Пользователь с такими данными не найден", "Ошибка!");

return;

}

MessageBox.Show("Пользователь найден в системе", "Внимание!");

NavigationService.Navigate(new Pages.AuthUserMain());

}

}

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

{

NavigationService.GoBack();

}

}

}

Листинг xaml AddProductTable:

<Page x:Class="EXAMRAZORENOV.Pages.AddProductTable"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="AddProductTable">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

<RowDefinition Height="\*"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="200"/>

<ColumnDefinition Width="\*"/>

</Grid.ColumnDefinitions>

<TextBlock Text="Дата публикации" Grid.Row="0" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Продавец" Grid.Row="1" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Категория" Grid.Row="2" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Название объявления" Grid.Row="3" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Цена" Grid.Row="4" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Состояние товара" Grid.Row="5" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Описание" Grid.Row="6" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Способ оплаты" Grid.Row="7" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBlock Text="Способ доставки" Grid.Row="8" Grid.Column="0" Margin="10,0,0,0" FontSize="16" FontFamily="Comic Sans MS"/>

<TextBox Grid.Row="0" Grid.Column="1" Text="{Binding PublicationDate}"/>

<ComboBox Name="SallerCmbBox" SelectedItem="{Binding Sallers}" DisplayMemberPath="FIO" Grid.Row="1" Grid.Column="1"/>

<ComboBox Name="CategoryCmbBox" SelectedItem="{Binding Categories}" DisplayMemberPath="Category" Grid.Row="2" Grid.Column="1"/>

<TextBox Grid.Row="3" Grid.Column="1" Text="{Binding AnnouncementName}"/>

<TextBox Grid.Row="4" Grid.Column="1" Text="{Binding ProductCost}"/>

<TextBox Grid.Row="5" Grid.Column="1" Text="{Binding ProductCondition}"/>

<TextBox Grid.Row="6" Grid.Column="1" Text="{Binding Description}"/>

<TextBox Grid.Row="7" Grid.Column="1" Text="{Binding PaymentMethod}"/>

<TextBox Grid.Row="8" Grid.Column="1" Text="{Binding DeliveryMethod}"/>

<Button Name="Save" Content="Сохранить" Grid.Row="9" Height="30" Width="110" Click="Save\_Click" Grid.Column="1" Margin="245,0,245,5" FontFamily="Comic Sans MS" VerticalAlignment="Bottom" Background="#99FFFF"/>

<Button x:Name="Back" Content="Назад" Grid.Row="9" Height="30" Width="110" Click="Back\_Click" FontFamily="Comic Sans MS" Margin="45,0,45,8" VerticalAlignment="Bottom" Background="#99FFFF"/>

</Grid>

</Page>

Листинг cs AddProductTable:

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 EXAMRAZORENOV.Entities;

namespace EXAMRAZORENOV.Pages

{

/// <summary>

/// Логика взаимодействия для AddProductTable.xaml

/// </summary>

public partial class AddProductTable : Page

{

public AddProductTable(Announcement selectedProduct)

{

InitializeComponent();

if (selectedProduct != null)

\_currentProduct = selectedProduct;

DataContext = \_currentProduct;

SallerCmbBox.ItemsSource = TRPOEntities.GetContext().Sallers.ToList();

CategoryCmbBox.ItemsSource = TRPOEntities.GetContext().Categories.ToList();

}

private Announcement \_currentProduct = new Announcement();

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

{

StringBuilder errors = new StringBuilder();

if (\_currentProduct.PublicationDate == null)

errors.AppendLine("Укажите дату публикации!");

if (\_currentProduct.Sallers == null)

errors.AppendLine("Выберите продовца!");

if (\_currentProduct.Categories == null)

errors.AppendLine("Выберите категорию!");

if (string.IsNullOrWhiteSpace(\_currentProduct.AnnouncementName))

errors.AppendLine("Укажите название публикации!");

if (\_currentProduct.ProductCost <= 0)

errors.AppendLine("Укажите цену товара!");

if (string.IsNullOrWhiteSpace(\_currentProduct.ProductCondition))

errors.AppendLine("Укажите состояние товара!");

if (string.IsNullOrWhiteSpace(\_currentProduct.PaymentMethod))

errors.AppendLine("Укажите способ оплаты!");

if (string.IsNullOrWhiteSpace(\_currentProduct.DeliveryMethod))

errors.AppendLine("Укажите способ доставки!");

//Проверяем переменную errors на наличие ошибок

if (errors.Length > 0)

{

MessageBox.Show(errors.ToString());

return;

}

//Добавляем в объект Cars новую запись

if (\_currentProduct.id == 0)

TRPOEntities.GetContext().Announcement.Add(\_currentProduct);

try

{

TRPOEntities.GetContext().SaveChanges();

MessageBox.Show("Данные успешно сохранены!", "Внимание!", MessageBoxButton.OK, MessageBoxImage.Information);

}

catch (Exception ex)

{

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

}

}

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

{

this.NavigationService.Navigate(new Uri("/Pages/AddProduct.xaml", UriKind.Relative));

}

}

}

Листинг xaml AddProduct:

<Page x:Class="EXAMRAZORENOV.Pages.AddProduct"

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:EXAMRAZORENOV.Pages"

mc:Ignorable="d"

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

Title="AddProduct">

<Grid Background="White">

<Grid.RowDefinitions>

<RowDefinition Height="377\*"/>

<RowDefinition Height="73\*"/>

</Grid.RowDefinitions>

<DataGrid Grid.Row="0" Name="DataGridProduct" AutoGenerateColumns="False" IsReadOnly="True">

<DataGrid.Columns>

<DataGridTextColumn Header="Дата публикации" Binding="{Binding PublicationDate}" Width="70" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Продавец" Binding="{Binding Sallers.FIO}" Width="100" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Категория" Binding="{Binding Categories.Category}" Width="100" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Название объявления" Binding="{Binding AnnouncementName}" Width="100" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Цена" Binding="{Binding ProductCost}" Width="60" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Состояние товара" Binding="{Binding ProductCondition}" Width="60" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Описание" Binding="{Binding Description}" Width="60" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Способ оплаты" Binding="{Binding PaymentMethod}" Width="60" FontFamily="Comic Sans MS"/>

<DataGridTextColumn Header="Способ доставки" Binding="{Binding DeliveryMethod}" Width="60" FontFamily="Comic Sans MS"/>

<DataGridTemplateColumn Width="\*">

</DataGridTemplateColumn>

</DataGrid.Columns>

</DataGrid>

<Button Grid.Row="1" ToolTip="Для удаления данных из базы, нажмите на эту кнопку." x:Name="Add" Content="Добавить" HorizontalAlignment="Right" Click="AddEdit\_Click" Height="30" VerticalAlignment="Top" Margin="0,14,680,0" Width="110" FontFamily="Comic Sans MS" Background="#99FFFF"/>

<Button Grid.Row="1" Content="Назад" Name="Back" Height="30" Margin="680,17,0,0" Click="Back\_Click" FontFamily="Comic Sans MS" VerticalAlignment="Top" HorizontalAlignment="Left" Width="110" Background="#99FFFF"/>

</Grid>

</Page>

Листинг cs AddProduct:

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 EXAMRAZORENOV.Entities;

namespace EXAMRAZORENOV.Pages

{

/// <summary>

/// Логика взаимодействия для AddProduct.xaml

/// </summary>

public partial class AddProduct : Page

{

public AddProduct()

{

InitializeComponent();

DataGridProduct.ItemsSource = TRPOEntities.GetContext().Announcement.ToList();

}

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

{

var BooksForRemoving = DataGridProduct.SelectedItems.Cast<Announcement>().ToList();

if (MessageBox.Show($"Вы точно хотите удалить записи в количестве {BooksForRemoving.Count()} элементов?", "Внимание", MessageBoxButton.YesNo, MessageBoxImage.Question) == MessageBoxResult.Yes)

{

try

{

TRPOEntities.GetContext().Announcement.RemoveRange(BooksForRemoving);

TRPOEntities.GetContext().SaveChanges();

MessageBox.Show("Данные успешно удалены!");

DataGridProduct.ItemsSource = TRPOEntities.GetContext().Announcement.ToList();

}

catch (Exception ex)

{

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

}

}

}

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

{

NavigationService.Navigate(new Pages.AddProductTable((sender as Button).DataContext as Announcement));

}

private void BooksTable\_IsVisibleChanged(object sender, DependencyPropertyChangedEventArgs e)

{

if (Visibility == Visibility.Visible)

{

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

DataGridProduct.ItemsSource = TRPOEntities.GetContext().Announcement.ToList();

}

}

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

{

this.NavigationService.GoBack();

}

}

}

Листинг Model context:namespace EXAMRAZORENOV.Entities

{

using System;

using System.Data.Entity;

using System.Data.Entity.Infrastructure;

public partial class TRPOEntities : DbContext

{

private static TRPOEntities context;

public static TRPOEntities GetContext()

{

if (context == null)

context = new TRPOEntities();

return context;

}

public TRPOEntities()

: base("name=TRPOEntities")

{

}

protected override void OnModelCreating(DbModelBuilder modelBuilder)

{

throw new UnintentionalCodeFirstException();

}

public virtual DbSet<Announcement> Announcement { get; set; }

public virtual DbSet<Categories> Categories { get; set; }

public virtual DbSet<Sallers> Sallers { get; set; }

public virtual DbSet<sysdiagrams> sysdiagrams { get; set; }

public virtual DbSet<Users> Users { get; set; }

}

}