**Многопользовательское приложение**

Задание 1: Многопользовательское приложение

2. CRM для небольшого бизнеса

MVVM:

Создать ClientModel, OrderModel.

Реализовать CRMViewModel с коллекцией клиентов и заказов.

Создать команды для добавления нового клиента, редактирования и

удаления.

Разделение логики и представления:

Вынести бизнес-логику работы с клиентами в ClientService.

В View оставить только DataBinding и визуальное оформление.

Асинхронное программирование:

Имплементировать загрузку списка клиентов через Task.Run().

Добавить индикатор загрузки.

Листинг программы:

using System;

using System.Collections.ObjectModel;

using System.IO;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Input;

namespace CRMApp{

public partial class MainWindow : Window {

public ObservableCollection<Client> Clients { get; set; }

public MainWindow() {

InitializeComponent();

Clients = new ObservableCollection<Client>();

ClientsListView.DataContext = this;

LoadClients();

}

private void Exit\_Click(object sender, RoutedEventArgs e) {

Close();

}

private void Window\_KeyDown(object sender, KeyEventArgs e) {

if (Keyboard.Modifiers == ModifierKeys.Control) {

switch (e.Key) {

case Key.N:

AddClientCommand(null, null);

break;

case Key.E:

EditClientCommand(null, null);

break;

case Key.D:

DeleteClientCommand(null, null);

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| ФИО:  Тип: string  Контакт:  Тип: string  История взаимодействий:  Тип: string  Тип клиента:  Тип: string | Шаталин, +375331331312, продали, VIP |

Анализ результатов:

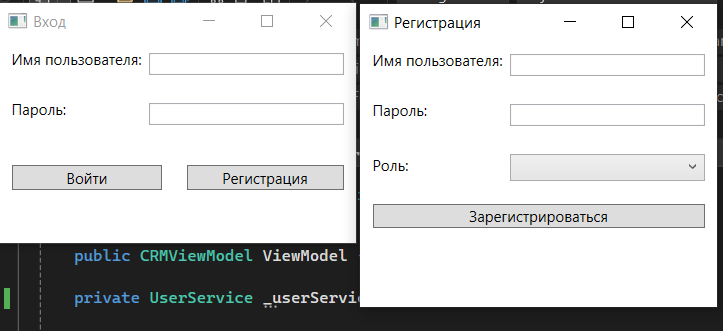


Рисунок 1 – Окно входа и регистрации

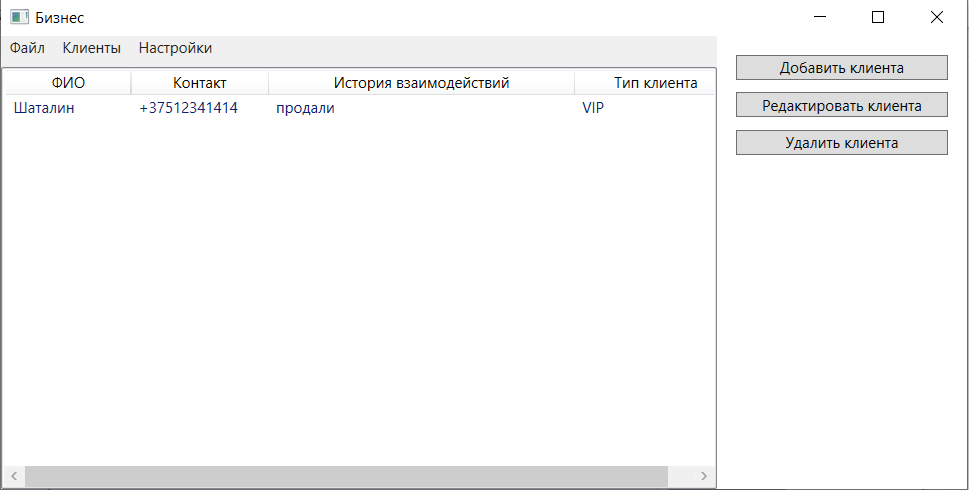


Рисунок 2 – Результат работы программы