

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федерального государственного бюджетного образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
Московский техникум космического приборостроения

СПЕЦИАЛЬНОСТЬ: 09.02.03 Программирование в компьютерных системах

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

по учебной практике по профессиональному модулю МДК 03.01
«Технология разработки программного обеспечения»

Студент

(подпись)

Москва 2019

СОДЕРЖАНИЕ

1 Постановка задачи	3
2 Структура программы	4
3 Отладка программы.....	7
4 Оптимизация программы.....	8
5 Тестирование программы	9
5.1 Тестирование в нормальных условиях	9
5.2 Тестирование в экстремальных условиях	11
5.3 Тестирование в исключительных ситуациях	11
6 Руководство пользователя	12
Заключение.....	17
Список использованных источников.....	18
Приложение А Листинги программы	19
А.1 Листинг формы Авторизация	20
А.2 Листинг формы Основная_форма	22
Приложение Б Результаты выполнения программы	34

1 Постановка задачи

Разработать информационную систему для агентства недвижимости. У агентства находится некоторое количество объектов жилой недвижимости различных стоимостей и типов. Все клиенты при обращении проходят обязательную регистрацию, при которой о них собирается стандартная информация. Система должна обеспечивать ведение списка объектов недвижимости, покупателей и договоров покупки. Информационная система должна содержать следующие сведения:

Объекты недвижимости	Покупатель	Договор покупки	Пользователи
Тип жилья	Фамилия	Покупатель	Пользователь
Тип дома	Имя	Объект недвижимости	Пароль
Населенный пункт	Отчество	Стоимость	
Тип отделки	Адрес	Дата продажи	
Тип окон	Телефон	Дата окончания действия скидки	
Виды покрытия пола		ID номер договора	
Виды потолков			
Этаж			
Стоимость			
Площадь			

1) Создать таблицы «Договор покупки», «Покупатель», «Объект недвижимости», «Тип дома», «Тип потолка». Заполните таблицы «Объект недвижимости», «Тип дома», «Тип потолка».

2) Создать формы «Объект недвижимости», «Тип дома», «Тип потолка» соответствующие таблицам базы данных с возможностью просмотра, добавление, удаление, обновление записей таблиц.

3) На форме договора аренды осуществить поиск недвижимости по параметрам: «Тип дома», «Тип потолка». Рассчитать стоимость по формуле:

$$\text{Стоимость} = \text{Объект_недвижимости.Стоимость} \times \text{Скидка.}$$

4) Создать форму для авторизации зарегистрированных пользователей. Необходимо предусмотреть возможность регистрации пользователей. При регистрации пароль должен отвечать следующим требованиям: минимум 7 символов, минимум 1 прописная буква.

2 Структура программы

На рисунке 2.1 представлена структура программы.

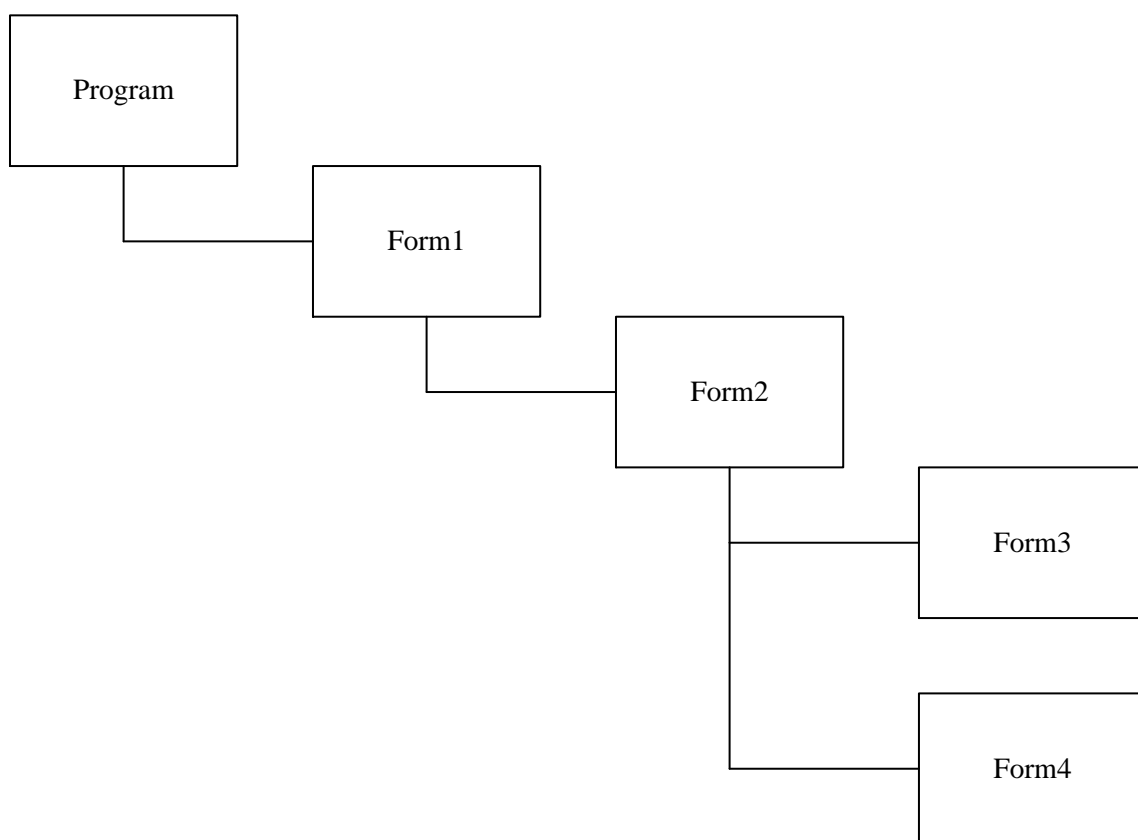


Рисунок 2.1 – Структура программы

Формы, используемые в основной программе, приведены в таблице 2.1.

Таблица 2.1

Форма	Назначение
Form1	Форма авторизации пользователя
Form2	Основная форма

Функции, используемые в форме Авторизация, приведены в таблице 2.2.

Таблица 2.2

Функция	Назначение
button1_Click	Проверка логина и пароля пользователя, вызов формы Form2
Form_FormClosing	Выход из формы

Функции, используемые в форме Основная_форма, приведены в таблице 2.3.

Таблица 2.3

Функция	Назначение
Select	Функция вывода содержимого
del_add	Функция удаления и добавления содержимого в базу данных
Основная форма_Load_1	Действия при входе в форму
button1_Click	Добавление записи в таблицы NEDVIJIMOST
button2_Click	Изменение значений в таблице NEDVIJIMOST
button3_Click	Удаление записи из таблицы NEDVIJIMOST
button7_Click	Выбор параметров таблицы NEDVIJIMOST
button8_Click	Добавление записи в таблицы DOM
button9_Click	Изменение значений в таблице DOM
button12_Click	Удаление из таблицы DOM
button15_Click	Добавление записи в таблицу POTOLKI

Продолжение таблицы 2.3

Функция	Назначение
button14_Click	Изменение значений полей в таблице POTOLKI
button13_Click	Удаление из таблицы POTOLKI
button5_Click	Добавление покупателя в базу данных
button11_Click	Выбор данных покупателей
button16_Click	Расчет 30% скидки
button10_Click	Добавление договора в базу данных
button6_Click	Добавление нового пользователя
slova	Проверка на ввод русских букв
cisla	Проверка ввода только чисел
dataGridView1_SelectionChanged	Вывод параметров выбранного объекта недвижимости
dataGridView5_SelectionChanged	Вывод параметров выбранного покупателя
Form_FormClosing	Выход из формы

3 Отладка программы

Отладка программы представляет собой процесс локализации и исправления ошибок. Локализацией называют процесс определения оператора программы, выполнение которого вызвало нарушение нормального вычислительного процесса. Для исправления ошибки необходимо определить ее причину, то есть определить оператор или фрагмент, содержащий ошибку. Отладка программы занимает значительную часть рабочего времени программиста, нередко большую, чем составление программы. Практически любая программа перед началом отладки содержит как минимум одну ошибку.

В соответствии с этапом обработки, на котором проявляются ошибки, различают следующие виды ошибок:

- 1) синтаксические ошибки – ошибки, обнаруживаемые компилятором при выполнении синтаксического и частично семантического анализа;
- 2) ошибки компоновки – ошибки, обнаруживаемые компоновщиком при объединении модулей программы;
- 3) ошибки выполнения – ошибки, обнаруживаемые ОС, аппаратными средствами или пользователем при выполнении программы.

При вводе даты используется элемент управления `dateTimePicker`, что также позволяет избежать ошибку неправильного формата исходных данных.

При изменении значений полей в таблице DOM была допущена ошибка выполнения. При написании значение было присвоено другому полю.

```
“Sqlstr = "insert into DOM values ( " + textBox27.Text + " );”.
```

Данная ошибка была исправлена:

```
“Sqlstr = "insert into DOM values ( " + textBox25.Text + " );”.
```

Все ошибки были исправлены. Для того чтобы убедиться в правильности работы программы, было проведено тестирование.

4 Оптимизация программы

Оптимизация программы – это процесс модификации кода программы с целью улучшения ее эффективности.

Основной задачей программирования является создание правильных, а не эффективных программ. Зачастую для повышения производительности программ большую значимость имеют такие факторы как:

- четко выработанные требования к программе до начала ее разработки;
- хорошая архитектура программы;
- хорошо спроектированные модули;
- правильно выбранный компилятор со встроенными средствами оптимизации.

Ввиду данных факторов основной целью разработки являлось создание правильной, хорошо спроектированной программы.

В большинстве случаев к оптимизации следует приступать только после того, как программа создана и выдает требуемые результаты. В противном случае можно получить не просто неудобочитаемый код и неработающую программу, но еще и неоптимизированное решение. Однако неизвестно, насколько эффективной будет программа, оптимизированная после ее разработки, поэтому определять требования к эффективности следует на стадии проектирования.

Несмотря на вышеперечисленные факты, оптимизация оказывается очень полезной, когда программа работает правильно. Производя небольшие изменения в некоторых фрагментах кода, называемых критическими областями, оптимизация помогает повысить эффективность программы в несколько раз.

Так как современные ЭВМ отличаются высоким быстродействием, то оптимизация отдельных, редко встречающихся операторов является бесполезной тратой времени. Однако повышение эффективности на мощных машинах можно получить за счет правильной оптимизации циклов и операторов, находящихся внутри тела циклов.

Программа, разрабатываемая в рамках практической работы, была оптимизирована по памяти. Для оптимизации по памяти программа была разделена на классы, что позволило существенно сэкономить память.

Вышеприведенный прием помог существенно повысить эффективность работы программы.

5 Тестирование программы

Тестирование – это набор процедур и действий, предназначенных для демонстрации правильности работы программы в заданных режимах и внешних условиях. Цель тестирования – выявить наличие ошибок или убедительно продемонстрировать их отсутствие.

Процесс тестирования проходит в 3 этапа:

- проверка поведения программы в нормальных условиях;
- проверка поведения программы в экстремальных условиях;
- проверка поведения программы в исключительных ситуациях.

Каждый из этапов предполагает задание определенного, характерного для данного этапа набора данных.

5.1 Тестирование в нормальных условиях

Тестирование в нормальных условиях было проведено со всеми формами. Были протестированы переходы от формы к форме, корректная выборка данных из базы данных. Особое внимание было уделено проверке регистрации нового пользователя. Тестирование регистрации нового пользователя представлено на рисунках 5.1, 5.2.

The screenshot shows a window titled 'Form2' with a tabbed interface. The active tab is 'Регистрация пользователя'. Inside the window, there is a form titled 'Новый пользователь'. The form contains four input fields: 'Логин' (Login) with the text 'Пользователь', 'Пароль' (Password) with masked characters '*****', 'Пароль админа' (Admin Password) with masked characters '...', and 'Доступ' (Access) with a dropdown menu showing 'Менеджер'. A 'Подтвердить' (Confirm) button is located at the bottom of the form.

Рисунок 5.1 –Добавление нового пользователя»

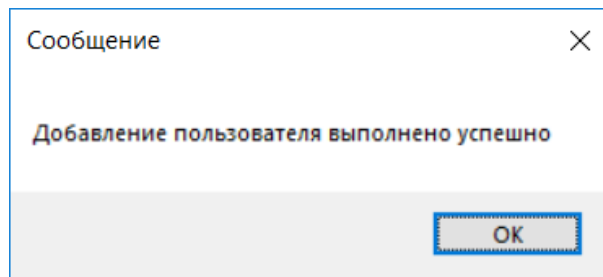


Рисунок 5.2 – Окно, подтверждающее правильность введенных данных

Во время тестирования программы в нормальных условиях ошибок не выявлено, так как окно подтверждения, представленное на рисунке 5.2, появляется только в случае правильности ввода данных. Тестирование было подтверждено визуально, и оно показало, что программа работает правильно.

5.2 Тестирование в экстремальных условиях

Провести проверку в экстремальных условиях невозможно в связи с особенностями элементов управления языка C#, в которых нет ограничений на ввод данных.

5.3 Тестирование в исключительных ситуациях

Произвести тестирование исключительных ситуациях в данной программе невозможно, так как в ней используется защита от ошибок, исключающая возможность ввода неправильных данных.

Так как тестирование в нормальных, экстремальных условиях и исключительных ситуациях не выявило наличие ошибок, можно сказать, что программа работает правильно. Листинг программы представлен в приложении А, а результаты программы – в приложении Б.

6 Руководство пользователя

Минимальные системные требования:

- Windows 7;
- процессор от 1,2 ГГц;
- ОЗУ от 64 МБ;
- свободное место на жестком диске 256 КВ;
- .NET Framework 4.5 и выше.

Для работы с программой следуйте пошаговому руководству, представленному ниже:

- 1) Для запуска приложения следует запустить файл «Агентство.exe»;
- 2) При запуске приложения на экране появится главная форма. Она представляет форму авторизации пользователя (рисунке 6.1);

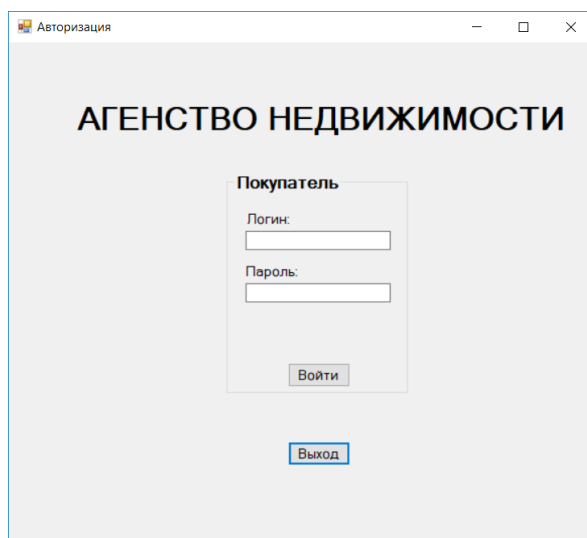


Рисунок 6.1 – Форма авторизации пользователя

- 3) После авторизации пользователь видит главную форму, представленную в виде нескольких вкладок. Одна из них «Объект недвижимости», которая позволяет производить поиск, добавление, изменение, удаление полей таблицы из базы данных. Чтобы произвести необходимые действия нужно ввести данные в полях «Параметры» и выбрать соответствующую кнопку (рисунок 6.2). Каждое ваше действие подтверждается выводом на экран окна об успешном завершении действия или окном об ошибке (рисунок 6.3);

Номер	Тип жилья	Тип дома	Населенный пункт	Тип отделки	Тип окон
1	Элитный	Кирпичный	Москва	Чистовая	Металлопласт
2	Эконом	Панельный	Балашиха	Чистовая	Деревянные
3	Премиум-класс	Монолитный	Волгоград	Без отделки	Пластиковые
4	Элитный	Кирпичный	Ростов	Чистовая	Деревянные
5	Бизнес-класс	Панельный	Балашиха	Без отделки	Металлопласт

Рисунок 6.2 – Главная форма

Рисунок 6.3 – Окно подтверждения действия

4) При нажатии на вкладку «Тип дома» откроется окно добавления, изменения и удаления записей типов дома. Данная форма позволяет добавить новый тип дома или изменить, удалить уже существующие. Данная вкладка показана на рисунке 6.4;

5) При нажатии на вкладку «Тип потолка» откроется окно добавления, изменения и удаления записей типов потолка. Данная форма позволяет добавить новый тип потолка или изменить, удалить уже существующие. Данная вкладка показана на рисунке 6.5;

Номер	Тип дома
1	Кирпичный
2	Монолитный
3	Панельный

Рисунок 6.4 – Вкладка «Тип дома»

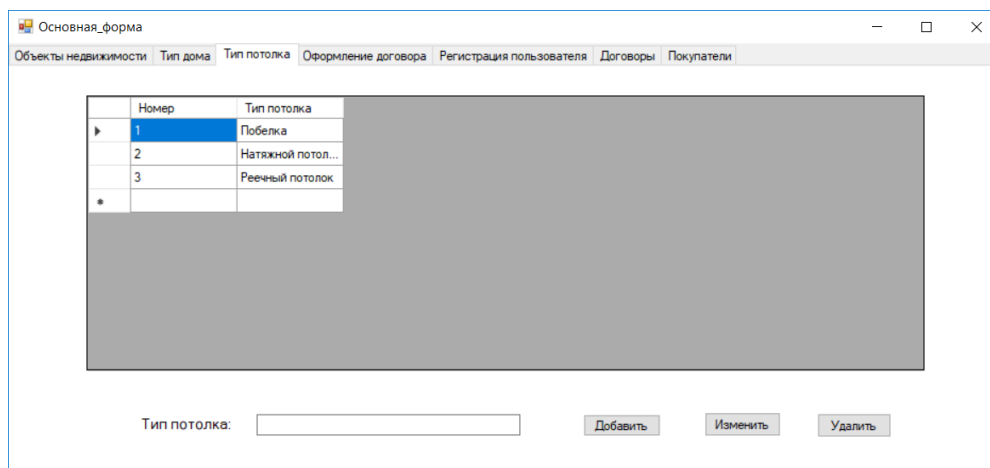


Рисунок 6.5 – Вкладка «Тип потолка»

6) При нажатии на вкладку «Покупатели», откроется окно, в котором покупатель может внести свои данные, а так же просмотреть список остальных покупателей (рисунок 6.6);

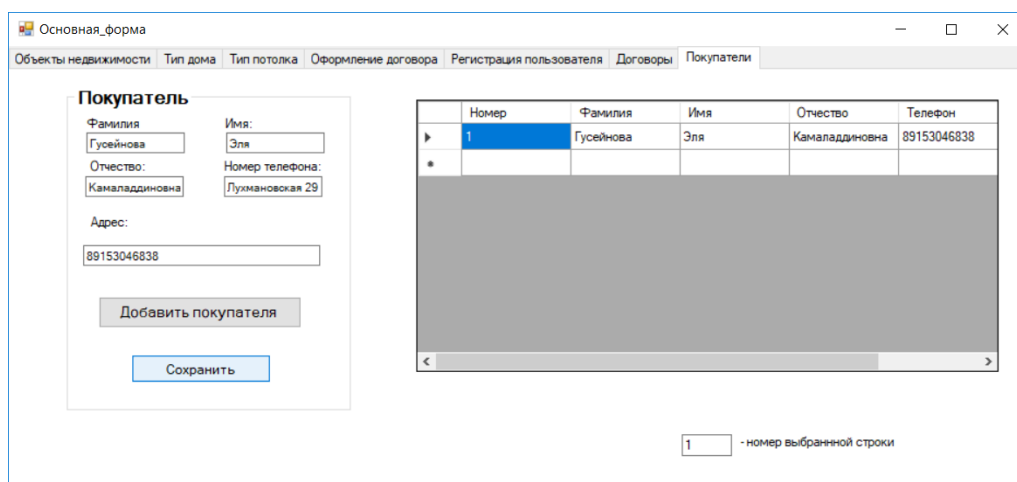


Рисунок 6.6 – Вкладка «Покупатели»

7) При выборе записи в таблице «Объект недвижимости» во вкладке «Параметры» и нажатии кнопки «Сохранить» данные о выбранном пользователем объекте недвижимости появляются во вкладке «Создать договор» в разделе «Параметры». При нажатии кнопки «Сохранить» во вкладке «Покупатели» данные о выбранном пользователем покупателе появляются во вкладке «Создать договор» в разделе «Покупатель». Данные действия позволяют пользователю избежать ошибок при вводе данных о параметрах объекта недвижимости, а так же о покупателях при составлении договора (рисунок 6.7);

8) На вкладке «Оформление договора» следует ввести промокод для расчета стоимости со скидкой, затем нажать кнопку «Оформить договор» для оформления договора (рисунок 6.7). На рисунке 6.8 представлено окно, подтверждающее составление договора;

Основная_форма

Объекты недвижимости | Тип дома | Тип потолка | **Оформление договора** | Регистрация пользователя | Договоры | Покупатели

Параметры

Тип жилья	Вид пола
Премиум-класс	Ламинат
Тип дома	Тип потолка
Монолитный	Побелка
Пункт назначения	Этаж
Волгоград	1
Тип отделки	Стоимость (руб)
Без отделки	15000000,0000
Тип окон	Площадь (кв метр)
Пластиковые	32

Покупатель

Фамилия	Имя:
Гусейнова	Эля
Отчество:	Номер телефона:
Камаладинова	89153046838
Адрес:	
Лузмановская 29	

Условия договора

Дата продажи:	Промокод
31.05.2019	999
Скидка 30%	
Стоимость с учетом скидки:	
10500000	

Оформить договор

Рисунок 6.7 – Вкладка «Составление договора»

Сообщение

Запись добавлена.

OK

Рисунок 6.8 – Окно, подтверждающее составление договора

9) При выборе вкладки «Регистрация пользователя» имеется возможность создать нового пользователя, пароль которого соответствует правилам, указанным в Постановке задачи данной работы. Чтобы пользователь не ошибся в написании пароля, необходимо повторно его ввести. Данная вкладка представлена на рисунке 6.9;

Основная_форма

Объекты недвижимости | Тип дома | Тип потолка | Оформление договора | **Регистрация пользователя** | Договоры | Покупатели

Новый пользователь

Логин:	Пароль:
Пароль админа:	Доступ:

Подтвердить

Рисунок 6.9 – Вкладка «Регистрация пользователя»

10) Для того, чтобы просмотреть составленные договоры, следует перейти во вкладку «Договоры» (рисунок 6.10);

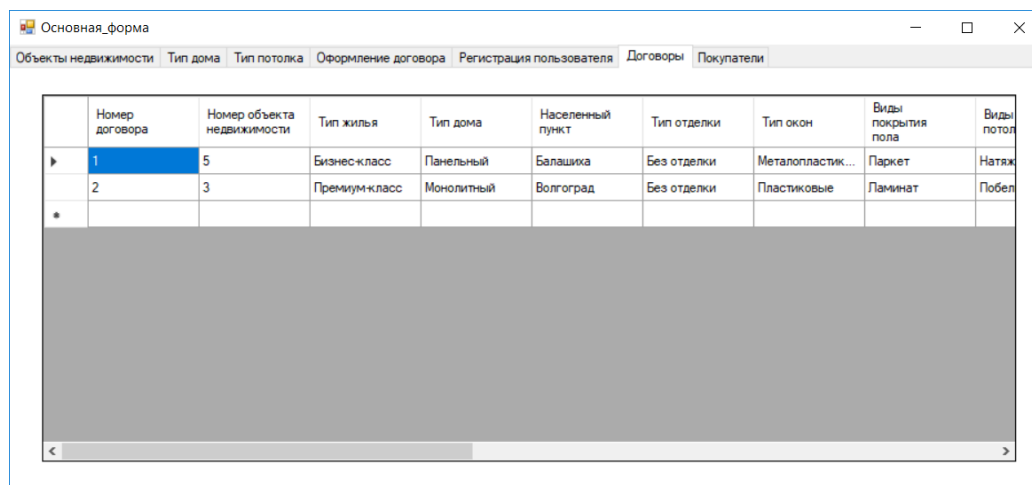


Рисунок 6.10 – Вкладка «Договоры»

- 11) Для того, чтобы закончить работу с программой, закройте главную форму (рисунок 6.11) нажатием на крестик в верхнем правом углу формы.

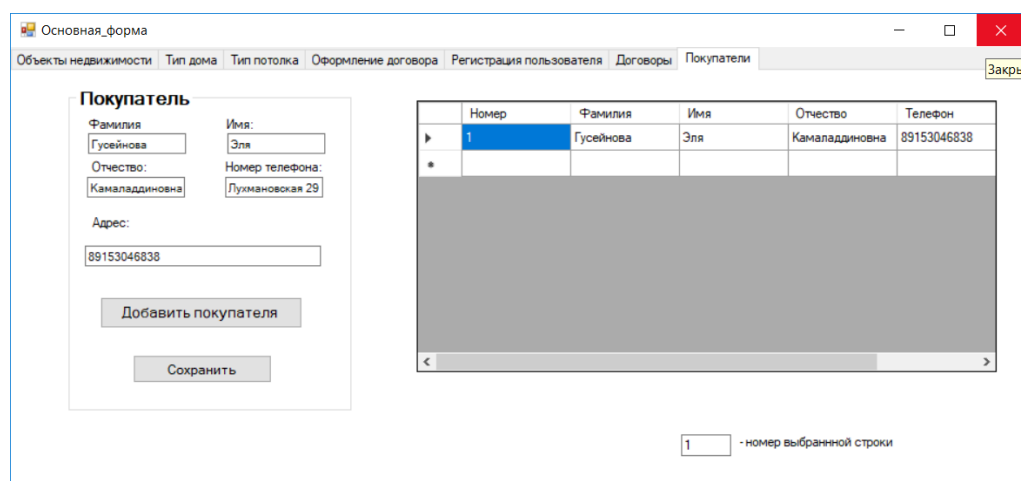


Рисунок 6.11 – Закрытие формы

ЗАКЛЮЧЕНИЕ

Целью разработки данной практической работы являлось написание информационной системы для агентства недвижимости, которая была достигнута в ходе выполненной работы.

В ходе написания данной работы были получены практические навыки в области проектирования программы. Программа разрабатывалась с использованием стратегии анализа сообщений, реализующей метод нисходящий разработки. Анализ основывался на изучении потоков данных, обрабатываемых программой. Опыт, полученный при написании работы, подтвердил необходимость в первую очередь правильной выработки требований к программе и грамотного подхода к ее проектированию перед началом написания кода.

Так же были получены практические навыки по оптимизации и тестированию программы. Появилось понимание важности отладки, ее длительности по времени относительно всего процесса разработки, необходимости знания аспектов из разных областей IT-технологий, как, например, знание аппаратной части, операционных систем, самих реализуемых процессов, природы и специфики различных ошибок и т.д.

Результатом проведенной работы является правильно спроектированное приложение, а также работающий, эффективный, отлаженный и корректно отвечающий тестам соответствующих условий тестирования проект.

При разработке программы были изучены и применены такие элементы управления как TextBox, DataGridView, ComboBox, DateTimePicker и другие.

Программа имеет надлежащую структуру и необходимые комментарии.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Абрамян М.Э. Visual C# на примерах. «Бином», 2008
- 2 Ботт Эд. Windows 10 для IT-специалистов. «ЭКОМ Паблишерз», 2016
- 3 Виссер Джуст. Разработка обслуживающих программ на C#. «АНК», 2017
- 4 Кормен Томас. Алгоритмы: построение и анализ. «Вильямс», 2016
- 5 Макконелл Стив. Совершенный код. «БХВ-Петербург», 2017
- 6 Матрин Роберт. Чистый код. «Питер», 2012
- 7 Рихтер Дж. Программирование на языке C#. 4-е изд. «Питер», 2007
- 8 Стилмен Э., Грин Д. Изучаем C#. «Питер» 2014
- 9 Спиридонов Э.С. Операционные системы. «ЛИБРОКОМ», 2017
- 10 Танебаум Эндрю, Бос Х. Современные операционные системы. «Питер», 2017
- 11 https://ru.wikipedia.org/wiki/C_Sharp
- 12 <https://geekbrains.ru/courses/21>
- 13 <https://habr.com/ru/post/203190/>
- 14 <https://metanit.com/sharp/tutorial/>
- 15 <http://mycsharp.ru/>

ПРИЛОЖЕНИЕ А

(обязательное)

Листинги программы

А.1 Листинг формы Авторизация

```
/* Практическая работа по профессиональному модулю МДК 03.01 "Технология разработки
программного
* обеспечения"
* Название: Агенство
* Разработал: , группа ТМП- .
* Дата и номер версии: 28.05.2019 v2.2
* Язык: C#
* Краткое описание:
* Данная программа является информационной системой для агентства недвижимости.
* Задание:
*
*
* Авторизация - форма авторизации пользователей;
* Основная форма - форма для добавления, изменения и удаления полей в таблицах,
* составления договоров и добавления пользователей, вывода информации о договорах и
* покупателях.
*/
```