Министерство образования Московской области

Государственное образовательное учреждение высшего образования Московской области

«Государственный гуманитарно-технологический университет»

**Ликино-Дулевский политехнический колледж – филиал ГГТУ**

**КУРСОВОЙ ПРОЕКТ**

«Разработка приложения по автоматизации регистрации пользователей»

МДК 02.01 «Технология разработки программного обеспечения»

**Выполнил:**

Абуталыблы Ильяс Ядигар оглы

Студент 4 курса группы ИСП.20А

09.02.07 Информационные системы и программирование

очной формы обучения

**Руководитель:**

Селиверстова Ольга Михайловна

Оценка

\_\_\_\_\_\_\_\_\_/\_\_\_­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Подпись руководителя

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ликино-Дулево

2023 год

**Оглавление**

[Введение 2](#_Toc154041397)

[1. Назначение разработки 5](#_Toc154041398)

[1.2 Требования к программе 5](#_Toc154041399)

[1.2.1 Требования к функциональным характеристикам 5](#_Toc154041400)

[1.2.2 Требования к надежности и безопасности 6](#_Toc154041401)

[1.2.3 Требования к составу и параметрам технических средств 6](#_Toc154041402)

[1.2.4 Требования к информационной и программной совместимости 6](#_Toc154041403)

[2. Разработка технического проекта 7](#_Toc154041404)

[2.1. Обоснование выбора CASE – средств 7](#_Toc154041405)

[2.2. Проектирование модели данных 7](#_Toc154041406)

[2.2.1 CASE – средство MS Visio 7](#_Toc154041407)

[3. Реализация 11](#_Toc154041408)

[3.1 Обоснование выбора средств разработки 11](#_Toc154041409)

[3.2 Руководство программиста 11](#_Toc154041410)

[3.3 Руководство пользователя 14](#_Toc154041411)

[4. Тестирование и откладка 20](#_Toc154041412)

[5. Методы и средства проведения расчётов оценки трудоёмкости разработки проекта (или Методы и средства защиты БД) 24](#_Toc154041413)

[Заключение 24](#_Toc154041414)

[Приложение 25](#_Toc154041415)

[Список литературы 27](#_Toc154041416)

Введение

Многофункциональный центр (МФЦ), полное название —Многофункциональный центр предоставления государственных и муниципальных услуг) — категория бюджетных учреждений в России, предоставляющих государственные и муниципальные услуги по принципу «одного окна» после однократного обращения заявителя с соответствующим запросом. При этом взаимодействие с органами, предоставляющими государственные услуги, или органами, предоставляющими муниципальные услуги, осуществляется многофункциональным центром без участия заявителя.

Заявленные цели:

* повышение качества и доступности государственных услуг;
* снижение издержек бизнеса на преодоление административных барьеров;
* повышение эффективности деятельности органов исполнительной власти и межведомственной координации;
* повышение открытости и прозрачности для общества.

25 октября 2005 года в России была принята концепция административной реформы и план мероприятий по её проведению в 2006—2010 годах. Одной из основных целей реформы было провозглашено повышение доступности и качества госуслуг.

В центрах «Мои документы» по месту жительства можно воспользоваться следующими услугами по трудоустройству: составление резюме, подбор вакансий из базы данных, присвоение статуса безработного, выплата пособий и выдача направления на профессиональное обучение. Список услуг включает: карьерную консультацию; профориентацию; психологическую помощь; тренинги по поиску работы; информирование о положении на рынке труда; определение стратегии обучения новой профессии; организацию стажировок и практик; организацию временного трудоустройства; осуществление социальных выплат гражданам, признанным в установленном порядке безработными; организацию ярмарок вакансий и учебных рабочих мест; организацию сопровождения при содействии занятости инвалидов; содействие безработным гражданам при переезде и безработным гражданам и членам их семей в переселении в другую местность для трудоустройства по направлению органов службы занятости.

В соответствии с правилами организации деятельности многофункциональных центров предоставления государственных и муниципальных услуг МФЦ использует автоматизированную информационную систему (АИС МФЦ). Основными принципами построения АИС МФЦ являются:

* процессно-ориентированная модель предоставления услуги;
* обеспечение информационной безопасности и защиты персональных данных;
* взаимодействие с внешними информационными системами в соответствии с требованиями действующего законодательства.

**1. Назначение разработки**

Автоматизированная информационная система «MFC» предназначена для оказания услуг клиенту по регистрации пользователя по указанному адресу. Пользователями программы выступает сотрудник центра. Прописка заказчика осуществляется на основании договоров Регистрация гражданина Российской Федерации по месту жительства, в которых оговариваются условия регистрации. Акте перерегистрации указываются: первый адрес, где клиент прописан на данный момент и конечный адрес куда он будет регистрироваться.

Данные первичных документов фиксируются в карточках учета, которые выполняют роль регистров регистрационного учета.

# 1.2 Требования к программе

## 1.2.1 Требования к функциональным характеристикам

Функциональные требования:

* Регистрация пользователя.
* Авторизация пользователя по логину и паролю.
* Добавление, редактирование и удаление данных из таблиц:
  + - «Адреса»;
    - «Паспорта»;
    - «Регистрации»;
    - «Работники»;
* Поиск данных по основным атрибутам таблиц.
* Сортировка: пользователей по логину и ФИО, остальных данных по всем доступным атрибутам.
* Фильтрация данных по различным критериям: названию города, типу паспорта, ФИО(Регистрации), ФИО(Работника)

## 1.2.2 Требования к надежности и безопасности

Программа должна быть в достаточной степени надёжна от сбоев. На крайний случай предусмотрено сохранение данных БД в приложении «Microsoft SQL» или восстановление данных в случае завершения работы.

### 1.2.3 Требования к составу и параметрам технических средств

Таблица №1. «Состав технических средств и их характеристики»

|  |  |
| --- | --- |
| Процессор | Intel® Core™ i5-6400 CPU @ 2.70GHz |
| Оперативная Память | 2 x 4 ГБ, DDR4, DIMM, 2133 МГц |
| Разрешение экрана | 1920x1080 |
| Размер экрана | 20 дюйма |
| Устройства ввода | Мышь, клавиатура |
| Дисковое пространство | 1Гб |
| Подключаемые устройства | Принтер |
| Операционная система | Windows 10 11 |

### 1.2.4 Требования к информационной и программной совместимости

Для корректной работы программы необходимо:

ОС Windows 10/11 – операционная система, сделанная корпорацией Microsoft;

Visual Studio 2022 – лучшая интегрированная среда разработки для создания многофункциональных, привлекательных кроссплатформенных приложений для Windows;

Microsoft SQL Server Management Studio 18 – интегрированная среда для управления любой инфраструктурой SQL, от SQL Server до баз данных;

# 2. Разработка технического проекта

## Обоснование выбора CASE – средств

Широкий набор функциональности: MS Visio предоставляет разнообразные типы диаграмм, включая диаграммы потоков данных, диаграммы классов, диаграммы вариантов использования и многое другое. Это позволяет разработчикам выбирать наиболее подходящий тип диаграммы для визуализации своих концепций и идей.

Интуитивно понятный пользовательский интерфейс: MS Visio имеет дружественный и интуитивно понятный пользовательский интерфейс, что делает процесс создания и редактирования диаграмм максимально простым и удобным. Разработчики смогут быстро освоить инструмент и создавать профессионально выглядящие диаграммы.

Расширяемость и настраиваемость: MS Visio позволяет расширять свои возможности с помощью сторонних плагинов и шаблонов. Разработчики могут использовать готовые шаблоны или создавать собственные, чтобы соответствовать особым требованиям и стандартам их организации.

Исходя из выбранного подхода к проектированию выбрано CASE – средство MS Visio, полностью удовлетворяющее запросам разработки программного средства.

## Проектирование модели данных

### 2.2.1 CASE – средство MS Visio

Microsoft Visio — программа для создания всевозможных видов схем. К их числу относятся блок-схемы, органиграммы, планы зданий и этажей, диаграммы DFD, схемы технологических процессов, модели бизнес-процессов, диаграммы плавательных дорожек, трехмерные карты и так далее. Основные возможности Visio:

* Диаграмма прецедентов - это графическое представление взаимодействия между системой и ее окружением, фокусирующееся на функциональности, которую система предоставляет своим пользователям (актерам). Она описывает различные прецеденты (сценарии использования) системы, используемые для взаимодействия с внешними сущностями. Диаграммы прецедентов часто используются в методологии Unified Modeling Language (UML) для моделирования требований к системе.
* Диаграмма действий – это графическое представление последовательности шагов или действий в конкретном процессе или сценарии. Она помогает визуально отобразить последовательность выполнения операций, решения проблем или других действий.
* Таблица операций обычно используется для документирования различных операций, процедур или действий, которые выполняются в рамках определенной системы или процесса.
* Таблицы описания документов могут быть использованы для организации и управления информацией о документах в организации или проекте. Эти таблицы могут включать информацию о названии документа, его типе, авторе, дате создания, версии, изменениях, статусе и другие сведения, которые помогают эффективно управлять документами. Такие таблицы облегчают отслеживание и доступ к документам в организации

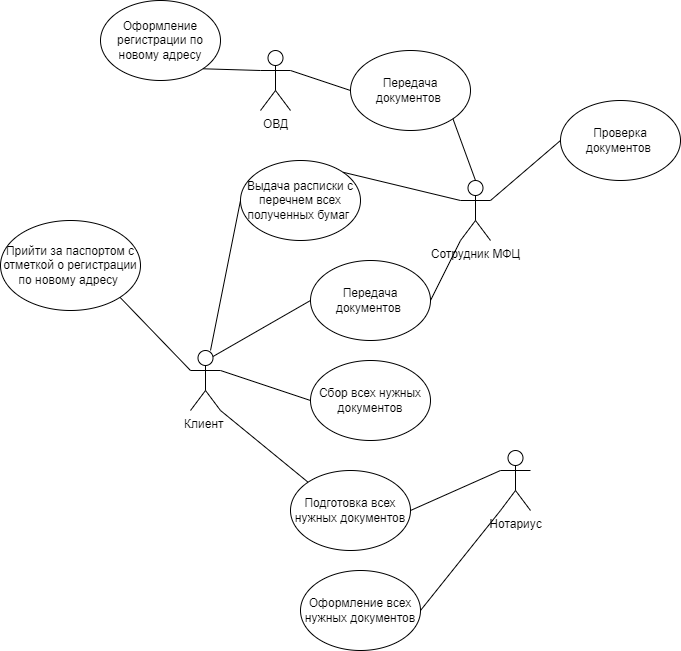


Рис. 2 «Диаграмма прецедентов»

Определение бизнес-процессов:

Таблица №2 «Таблица Бизнес-процессов»

|  |  |
| --- | --- |
| **Номер бизнес-процесса** | **Название бизнес-процесса** |
| 1 | Оформление всех нужных документов |
| 2 | Подготовка всех нужных документов |
| 3 | Сбор всех нужных документов |
| 4 | Передача документов |
| 5 | Выдача расписки с перечнем всех полученных бумаг |
| 6 | Проверка документов |
| 7 | Передача документов |
| 8 | Оформлении регистрации по новому адресу |
| 9 | Прийти за паспортом с отметкой о регистрации по новому адресу |

Словесный алгоритм бизнес-процесса «Регистрация» осуществляется следующим образом:

1. Оформление всех нужных документов: - Оформление всех нужных документов у нотариуса

2. Подготовка всех нужных документов: - передача заказчику готовых документов и информация о том какие документы нужны помимо

3. Сбор всех нужных документов: Сбор всех требуемых документов для регистрации

4. Передача документов: - Передача всех требуемых документов Сотруднику МФЦ

5. Выдача расписки с перечнем всех полученных бумаг: - Получение расписки о списке всех полученных сотрудником МФЦ документов

6. Проверка документов: - Сотрудник МФЦ проверяет документы на достоверность данных и их актуальность

7. Передача документов: - Сотрудник МФЦ передает все нужные документы в ОВД

8. Оформление регистрации по новому адресу – Сотрудник ОВД в соответствии с законом регистрирует вас по указанному адресу

9. Прийти за паспортом с отметкой о регистрации по новому адресу – Необходимо прийти и забрать готовый документ

Диаграмма действий:

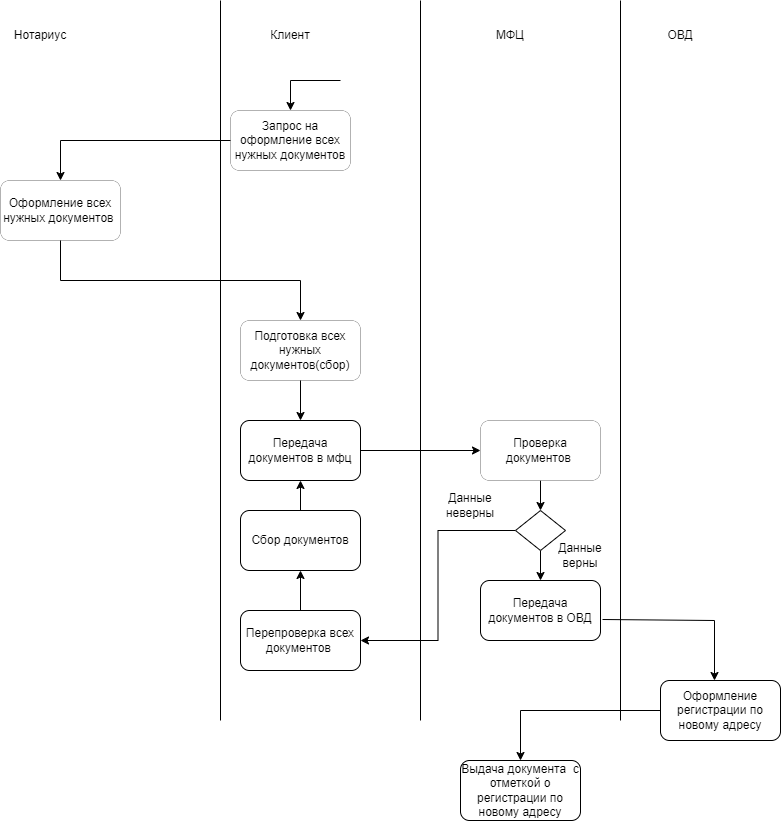


Рис. 3 «Бизнес-процесс «Регистрация»

Таблица №3 «Таблица описания операций

«Операция по выполнению услуги»

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Диаграмма и номер операции на  диаграмме | Составляемый документ  (Исходящий документ) | Операция | Исполнитель | Как часто | Входящие  документы  (документы-  основания) | Реестр, в котором регистрируется документ |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 Пр\_Т\_1 | 1.Заявление на услугу | Подача заявления | Клиент | По мере необходимости | - | Реестр регистрации документов |
| 1 Пр\_Т\_2 | 2.Регистрация заявления в системе | Регистрация заявления | Сотрудник регистрации | По мере необходимости | Заявление на услугу | Реестр заявлений |
| 1 Пр\_Т\_3 | 3.Уведомление о принятии заявления | Уведомление о принятии заявления | Сотрудник регистрации | Ежедневно | Регистрация заявления в системе | Реестр уведомлений |
| 1 Пр\_Т\_4 | 4.Заявление об отказе в услуге | Отказ в предоставлении услуги | Сотрудник регистрации | По мере необходимости | Регистрация заявления в системе | Реестр отказов |
| 1 Пр\_Т\_5 | 5.Уведомление о готовности услуги | Уведомление о готовности услуги | Сотрудник регистрации | По мере необходимости | Заявление на услугу | Реестр уведомлений |

# 3. Реализация

## 3.1 Обоснование выбора средств разработки

Был выбран Visual Studio 2022 как мощный инструмент для разработки. Разработчики зачастую сталкиваются с проблемами обработки информации, такими как необходимость быстрого и управляемого обработки данных, повышения производительности и мобильности разработчиков, а также с требованиями сокращения затрат на информационные технологии и расширения инфраструктуры в соответствии с растущими требованиями. В этом контексте Microsoft SQL Server 2022 является решением для управления и анализа данных. MS SQL Server 2018 предлагает интегрированное решение для управления и анализа данных, которое помогает:

- Строить, развертывать и управлять высоконадежными промышленными приложениями.

- Повышать производительность информационных технологий путем упрощения процессов разработки, развертывания и управления приложениями.

- Обеспечивать интеграцию данных между различными платформами, приложениями и устройствами.

- Оптимизировать стоимость без ущерба для качества выполнения, доступности, масштабируемости и безопасности.

Исходя из перечисленных выше преимуществ MS SQL Server 2018, этот продукт был выбран в качестве сервера базы данных для обеспечения повышенной эффективности и надежности работы системы.

## 3.2 Руководство программиста

Разработана схема данных в среде в Visual Studio.

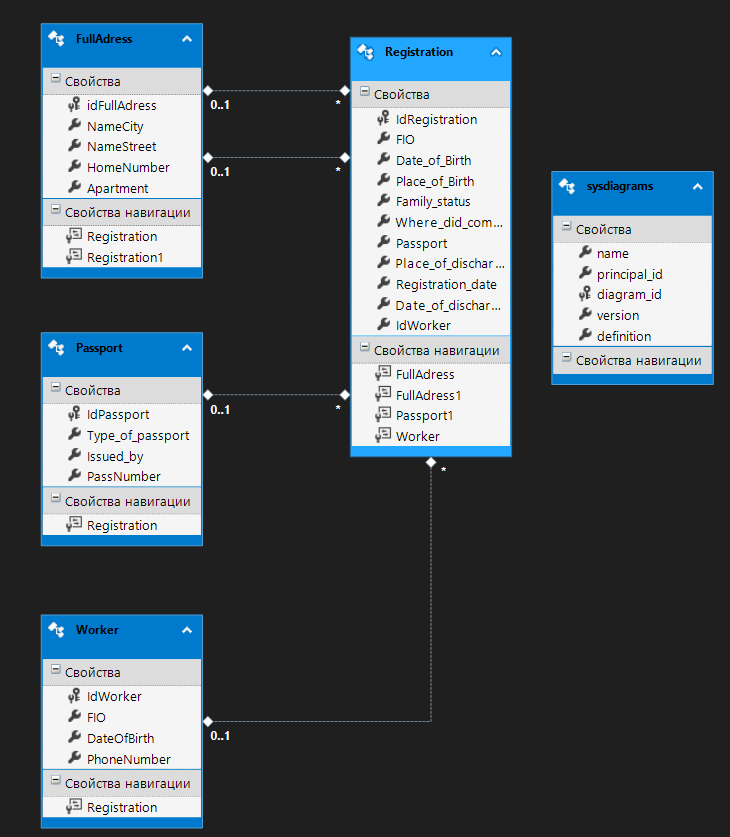


Рис. 3 «Модель данных»

Структура приложения в обозревателе решений:

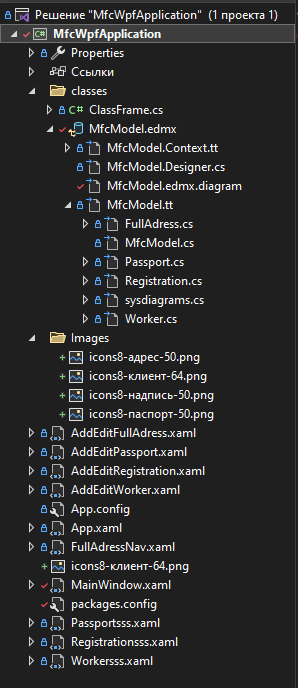


Рис. 4 «Структура приложения»

Класс Frame, служащий для вывода данных с базы данных:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Controls;

namespace SalesOfMaterials.Classes

{

public class ClassFrame

{

public static Frame frmObj;

}

}

Класс MFCDBEntities, служащий для связи с базой данных:

using System;

using System.Data.Entity;

using System.Data.Entity.Infrastructure;

public partial class MFCDBEntities : DbContext

{

private static MFCDBEntities \_context;

public MFCDBEntities()

: base("name=MFCDBEntities")

{

}

public static MFCDBEntities GetContext()

{

if (\_context == null)

\_context = new MFCDBEntities();

return \_context;

}

protected override void OnModelCreating(DbModelBuilder modelBuilder)

{

throw new UnintentionalCodeFirstException();

}

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

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

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

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

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

}

**Входные и выходные данные**

Таблица №2 «Словарь данных»

|  |  |  |  |
| --- | --- | --- | --- |
| **ключ** | **поле** | **обязательное** | **примечание** |
| **Адрес** | | | |
| Первичный | IdExpenseIvoices | Да | Идентификатор |
|  | NameCity | Да | Название города |
|  | NameStreet | Да | Название улицы |
|  | HomeNumber | Да | Номер дома |
|  | Apartment | Да | Номер квартиры |
| **Паспорт** | | | |
| Первичный | IdPassport | Да | Идентификатор |
|  | Type\_of\_passport | Да | Тип Паспорта |
|  | Issued\_by | Да | Кем выдан |
|  | PassNumber | Да | Номер Паспорта |
| **Работники** | | | |
| Первичный | IdWorker | Да | Идентификатор |
|  | FIO | Да | ФИО |
|  | DateOfBirth | Да | Дата рождения |
|  | PhoneNumber | Да | Номер телефона |
| **Тип пользователей** | | | |
| Первичный | IdRegistration | Да | Идентификатор |
|  | FIO | Да | ФИО |
|  | Date\_of\_Birth | Да | Дата рождения |
| Внешний | Place\_of\_Birth | Да | Внешний ключ к таблице адрес |
|  | Family\_status | Да | Семейное положение |
| Внешний | Where\_did\_come\_from | Да | Внешний ключ к таблице адрес |
| Внешний | Passport | Да | Внешний ключ к таблице паспорт |
| Внешний | Place\_of\_discharge | Да | Внешний ключ к таблице адрес |
|  | Registration\_date | Да | Дата регистрации |
|  | Date\_of\_discharge | Да | Дата когда выбыл |
| Внешний | IdWorker | Да | Внешний ключ к таблице работник |

## 3.3 Руководство пользователя

При открытии приложения выпадает главное окно с 4 кнопками для переходов на страницы: Адреса, Паспорта, Регистрации, Работники

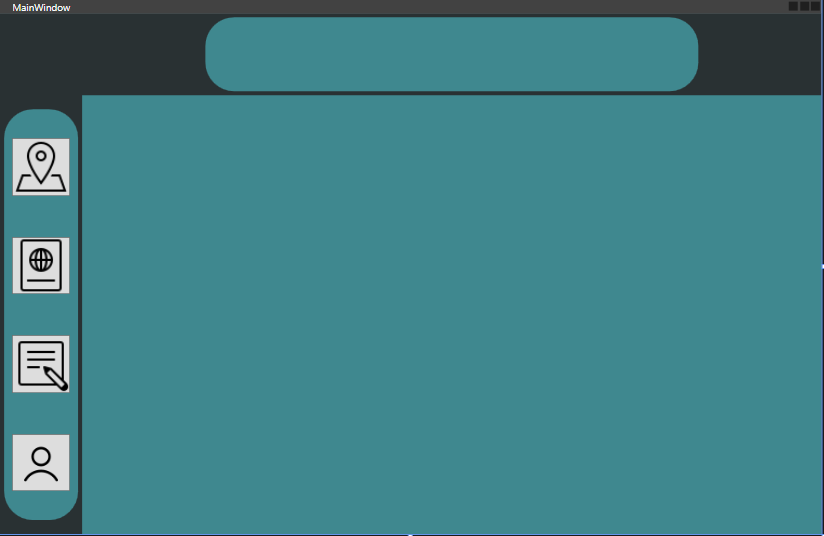


Рис. 4 «Главное меню»

Страница Адресов на этой странице мы можем добавлять удалять и редактировать адреса

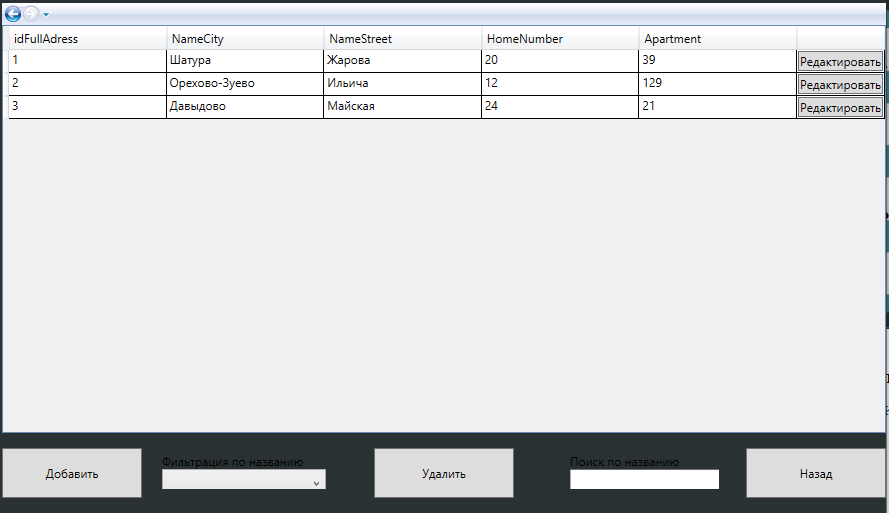


Рис. 5 «Страница адреса»

Страница Паспортов на этой странице мы можем добавлять удалять и редактировать Паспортные данные а также фильтровать по типу паспорта и осуществлять поиск по типу пасспорта

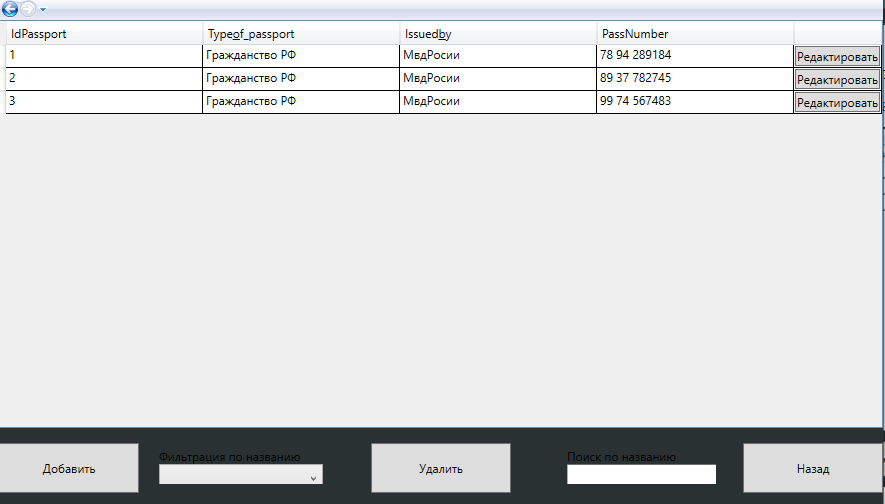


Рис. 6 «Страница паспорта»

Страница Регистрации на этой странице мы можем добавлять удалять и редактировать Регистрационные данные а также фильтровать по ФИО и осуществлять поиск по ФИО

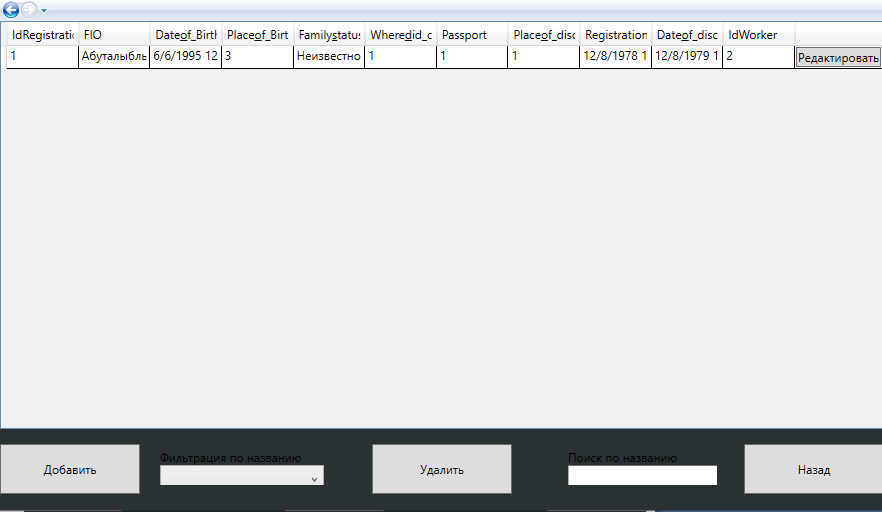


Рис. 7 «Страница Регистрации»

Страница Работников на этой странице мы можем добавлять удалять и редактировать данные о работниках а также фильтровать по ФИО и осуществлять поиск по ФИО

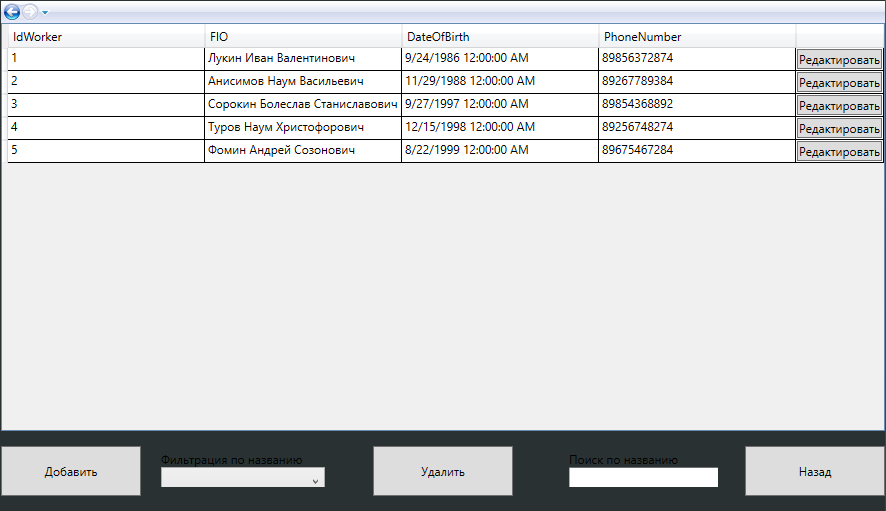


Рис. 8 «Страница Работников»

Страница добавления на примере Работников:



Рис. 9 «Страница Добавления»

Страница добавления на примере работников:

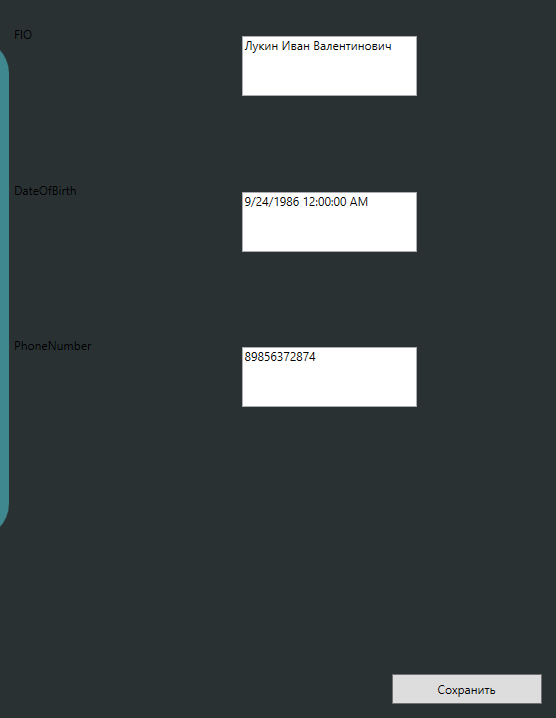


Рис. 10 «Страница Редактирования»

Функция Фильтрации по названию:

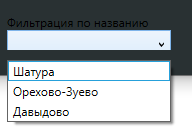


Рис. 11 «Функция фильтрации»

Функция Поиска по названию:

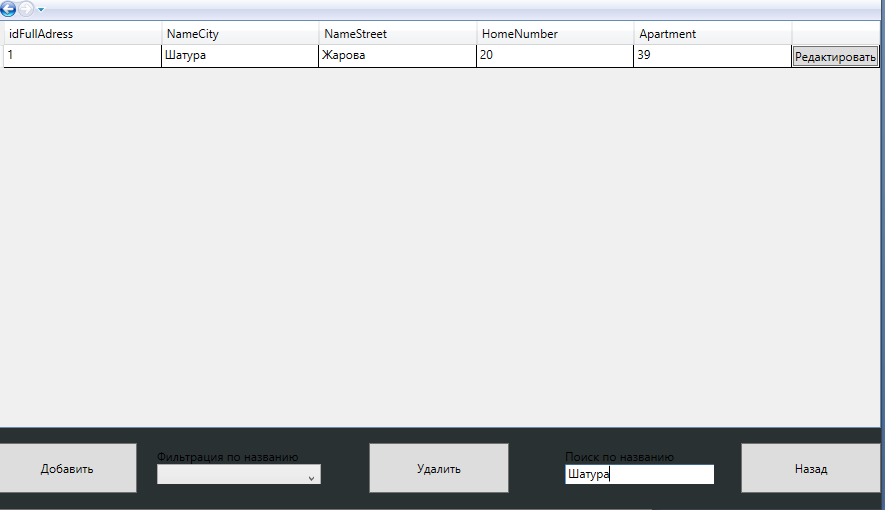


Рис. 12 «Функция поиска по названию»

Функция удаления на примере Работника:

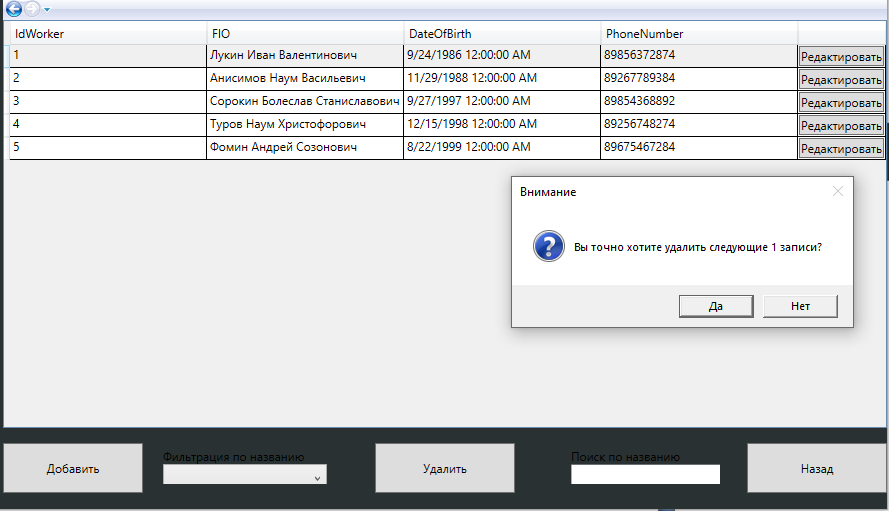


Рис. 13 «Функция удаления до удаления»

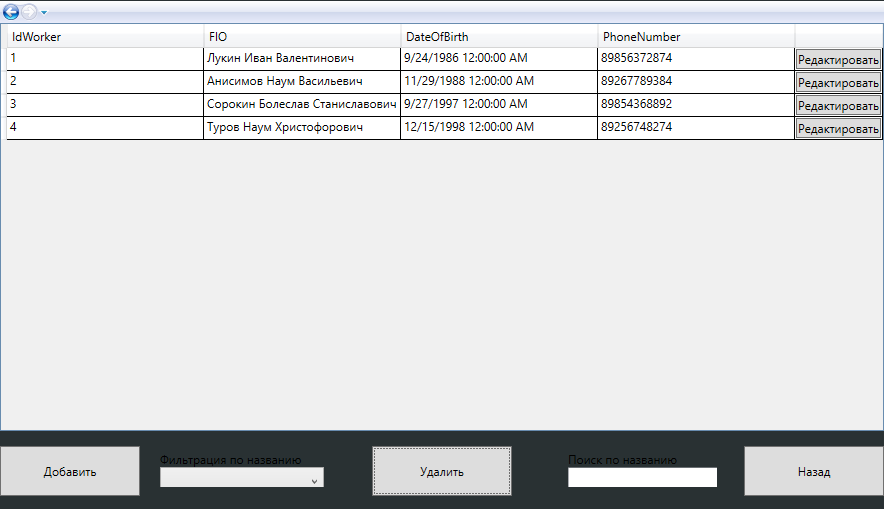


Рис. 14 «Функция удаления после удаления»

Диалоговые окна для корректной работы пользователя с приложением:

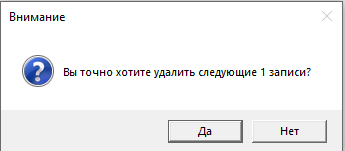


Рис. 15 «Диалоговое окно удаления»

# 4. Тестирование и откладка

Жизненный цикл тестирования программного обеспечения — это процесс выполнения различных действий в ходе проведения тестирования. Как пример, сюда относится составление тест-плана, анализ требований, поиск дефектов, поддержка после релиза и так далее.

Тестирование ПО — процесс исследования, испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов, выбранных определённым образом (ISO/IEC TR 19759:2005).

Цель тестирования — проверка соответствия ПО предъявляемым требованиям, обеспечение уверенности в качестве ПО, поиск очевидных ошибок в программном обеспечении, которые должны быть выявлены до того, как их обнаружат пользователи программы.

Тестовый сценарий (test case) — это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.

Методы тестирования:

* Тестирование белого ящика — метод тестирования ПО, который предполагает, что внутренняя структура/устройство/реализация системы известны тестировщику.
* Тестирование серого ящика — метод тестирования ПО, который предполагает комбинацию White Box и Black Box подходов. То есть, внутреннее устройство программы нам известно лишь частично.
* Тестирование чёрного ящика — также известное как тестирование, основанное на спецификации или тестирование поведения — техника тестирования, основанная на работе исключительно с внешними интерфейсами тестируемой системы.

Тестовые сценарии, выполненные по методу белого ящика:

Тестовый сценарий № 1:

|  |  |
| --- | --- |
| **Тестовый пример #** | 1 |
| **Приоритет тестирования** | Низкий |
| **Заголовок/название теста** | Перейдите на страницу «Адреса» и нажмите на пункт «Фильтр по названию улицы» и выберете любой параметр |
| **Краткое изложение теста** | Должно поменяться количество записей при использовании фильтрации |
| **Этапы теста** | 1. Запустить программу через Visual Studio  2. Попасть в главное меню. Оттуда перейти на страницу «Адреса»  3. Затем нажать в контекстном меню у пункта «Фильтр по названию улицы» любой параметр |
| **Тестовые данные** | После использования «Фильтра по названию улицы» с любым параметром все записи должны отфильтроваться по выбранному параметру |
| **Ожидаемый результат** | Если будет использоваться фильтрация, то количество записей должно изменяться |
| **Фактический результат** | Количество записей не изменилось |
| **Статус** | Незачет |
| **Предварительное условие** | Исправьте код, чтобы фильтр по названию улицы была выполнена корректно |
| **Постусловие** | Подсчет количества записей работает совместно с фильтрацией |
| **Примечания/комментарии** | Подсчёт количества осуществляется через присвоение значения, полученного от метода «Count» у коллекции таблицы, текстовому элементу |

Тестовый сценарий № 2:

|  |  |
| --- | --- |
| **Тестовый пример #** | 2 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Перейдите на страницу «Адреса» и введите текст в «Поиск» |
| **Краткое изложение теста** | Должен выводиться список контрагентов c поиском по всем параметрам |
| **Этапы теста** | 1. Запустить программу через Visual Studio.  2. Попасть в главное меню. Оттуда перейти на страницу «Адреса».  3. Затем нажать ввести в поиск любой текст |
| **Тестовые данные** | После ввода текста, находящегося в одном из полей строки, данные выводятся правильно |
| **Ожидаемый результат** | Если будет использоваться поиск, то он будет выводить данные по всем полям |
| **Фактический результат** | Список Адреса выводится корректно |
| **Статус** | Зачет |
| **Предварительное условие** | - |
| **Постусловие** | Вывод данных с поиском по всем полям |
| **Примечания/комментарии** | Поиск осуществляется через выражение LINQ |

Участок кода с навигацией на страницу Адрес работающий некорректно:

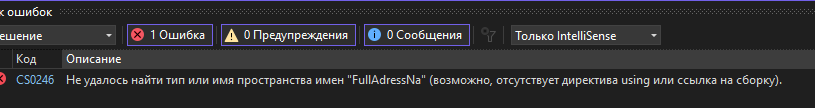


Рис. 16 «Неправильный тип или имя пространства имен»

Участок кода с навигацией на страницу Адрес работающий корректно:

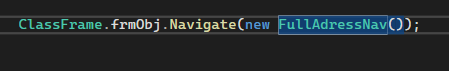


Рис. 17 «Навигация на страницу FullAdressNav»

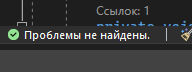


Рис. 18 «Исправленная функция»

Участок кода с присвоением переменной данных из БД работающий некорректно:



Рис. 19 «Неправильное название функции Binding»

Участок кода с присвоением переменной данных из БД работающий корректно:

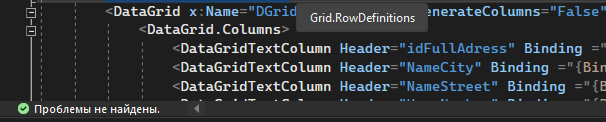


Рис. 20 «Исправленное название функции Binding»

# 5. Методы и средства проведения расчётов оценки трудоёмкости разработки проекта (или Методы и средства защиты БД)

## Заключение

В заключении данной курсовой работы можно сделать несколько ключевых выводов.

Во-первых, разработка программного обеспечения для предприятия МФЦ имеет огромное значение для оптимизации его работы. Вместо ручной обработки и хранения данных, автоматизированная система позволит значительно повысить эффективность работы, сократить время на выполнение процессов и минимизировать возможные ошибки.

Во-вторых, важно правильно спланировать и проанализировать требования предприятия МФЦ при разработке программного обеспечения. Только путем тщательного изучения потребностей и задач предприятия можно создать ПО, которое будет максимально соответствовать его потребностям и особенностям работы.

В-третьих, использование соответствующих технологий и методологий разработки, таких как Agile или DevOps, может значительно повысить эффективность процесса разработки программного обеспечения для предприятия МФЦ. Внедрение современных подходов позволит ускорить разработку, улучшить качество и обеспечить более гибкую адаптацию системы к изменениям требований.

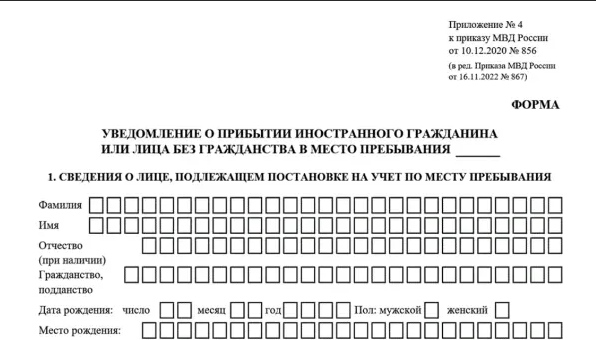
Наконец, стоит отметить, что разработка ПО для предприятия МФЦ является долгосрочным и постоянным процессом. После разработки и внедрения системы, необходимо обеспечить ее постоянную поддержку, обновление и развитие с учетом изменяющихся требований и технологических потребностей предприятия.

В целом, разработка ПО для предприятия МФЦ имеет большой потенциал для улучшения работы и повышения эффективности предприятия. Правильно спланированная и реализованная система сможет существенно сократить время и ресурсы, улучшить качество обслуживания и сделать работу персонала более продуктивной и комфортной.

## Приложение

Документы

Приложение 1 «Форма Регистрации»



Приложение 2 «Свидетельство»



## Список литературы

1. Joseph, J. Bambara SQL Server® Developer's Guide / Joseph J. Bambara, Paul R. Allen. - Москва: Мир, 2016. - 235 c.

2. Kalen, Delaney Inside Microsoft® SQL Server(TM) 2005: Query Tuning and Optimization / Kalen Delaney и др. - М.: Microsoft Press, 2014. - 448 c.

3. Аллен, Г. Тейлор SQL для чайников / Аллен Г. Тейлор. - М.: Диалектика, Вильямс, 2015. - 416 c.

4. Грабер, Мартин SQL для простых смертных / Мартин Грабер. - М.: ЛОРИ, 2014. - 378 c.

5. Кузнецов, Владимир Новейший самоучитель по 1С:Бухгалтерии 8 / Владимир Кузнецов , Сергей Засорин. - М.: БХВ-Петербург, 2017. - 336 c.

6. Чистов, Д. В. Практикум по программе "1С:Упрощенка 8" / Д.В. Чистов, С.А. Харитонов. - М.: 1С-Паблишинг, 2020. - 436 c.

7. Гудсон, Джон Практическое руководство по доступу к данным (+ DVD-ROM) / Джон Гудсон , Роб Стюард. - М.: БХВ-Петербург, 2013. - 304 c.

8. Филатова, В. 1С для начинающих. Понятный самоучитель / В. Филатова. - М.: Питер, 2018. - 256 c.

9. Рудаков А.В., Федорова Г.Н. – Технология разработки программных продуктов. - М.: Академия, 2018. - 206 с.

10. Рудаков А.В., Федорова Г.Н. – Технология разработки программных продуктов. Практикум. - М.: Академия, 2014. - 189 с.

11. Бьюли, А. Изучаем SQL / А. Бьюли. - М.: Символ-плюс, 2014. - 108 c.