**Министерство образования и молодежной политики**

**Свердловской области**

**ГАПОУ СО «Ирбитский политехникум»**

**Курсовая работа**

**по ПМ.02 Разработка и администрирование баз данных**

**МДК 02.02. Технология разработки и защиты баз данных**

**Тема: Проектирование и разработка базы данных "Спортивный клуб"**

**Выполнил:** студент группы ПКС-405

Иванов Иван Иванович

**Проверил:** преподаватель

Вишнякова Н.В.

Ирбит, 2021

**Содержание**

[Введение 3](#_Toc34315103)

[1. Постановка задачи 4](#_Toc34315104)

[2. Концептуальная модель базы данных "Спортивный клуб" 5](#_Toc34315105)

[3. Логическая модель "Model\_0" базы данных "SportClub" 6](#_Toc34315106)

[4. Разработка физической модели базы данных SportClub 7](#_Toc34315107)

[5. Разработка интерфейса 10](#_Toc34315108)

[Заключение 18](#_Toc34315109)

[Приложение 19](#_Toc34315110)

[Приложение 1 Программный код MainWindow.xaml 19](#_Toc34315111)

[Приложение 2 Программный код MainWindow.xaml.cs 20](#_Toc34315112)

[Приложение 3 Программный код MainPage.xaml 21](#_Toc34315113)

[Приложение 4 Программный код MainPage.xaml.cs 22](#_Toc34315114)

[Приложение 5 Программный код Sportsmans.xaml 23](#_Toc34315115)

[Приложение 6 Программный код Sportsmans.xaml.cs 24](#_Toc34315116)

[Приложение 7 Программный код AddSportsmans.xaml 25](#_Toc34315117)

[Приложение 8 Программный код AddSportsmans.xaml.cs 26](#_Toc34315118)

[Приложение 9 Программный код EditSportsmans.xaml 27](#_Toc34315119)

[Приложение 10 Программный код EditSportsmans.xaml.cs 28](#_Toc34315120)

[Список использованной литературы 29](#_Toc34315121)

# Введение

Спорт (англ. *sport*, сокращение от первоначального старофранц. desport - «игра», «развлечение») - организованная по определённым правилам деятельность людей (спортсменов), состоящая в сопоставлении их физических и (или) интеллектуальных способностей, а также подготовка к этой деятельности и межличностные отношения, возникающие в её процессе.

Происходящие сегодня в обществе социально-экономические изменения предъявляют новые требования к спортивному клубу, выражающиеся, в частности, в том, что ее спортсменов уже недостаточно обладать широкими и разносторонними знаниями, навыками и умениями их самостоятельно пополнять. Они должны быть социально активными, физически и морально здоровыми, а также ответственными людьми, обладающими определенным жизненным опытом, который поможет адаптироваться к суровым требованиям современной жизни.

Формирование ценностного отношения спортсменов к физической культуре и спорту может осуществляться в процессе их подготовки и участия в спортивно-массовых мероприятиях. Существующий традиционный подход к организации спортивно-массовых мероприятий в спортивном клубе не позволяет эффективно влиять на повышение уровня сформированности ценностного отношения тренеров и спортсменов к физической культуре и спорту. Повышение эффективности проведения спортивно - массовых мероприятий в спортивнов клубе невозможно без грамотного планирования на всех ступенях процесса их организации. Необходимость создание клуба, прежде всего связан, с желанием ребят разнообразить свой досуг, укрепить здоровье и поучаствовать в различных соревновательных и конкурсных мероприятиях, расширить круг общения.

**Цель курсовой работы** - проектирование и разработка базы данных "Спортивный клуб" в MS SQL Server и создание приложения в среде Visual Studio с помощью WPF форм, языка C# и технологии Entity Framework.

**Задачи курсовой работы**:

1. Изучить предметную область по деятельности спортивных клубов;
2. Разработать концептуальную модель БД "SportClub";
3. Спроектировать и оптимизировать логическую и физическую модель БД, провести нормализацию;
4. Разработать интерфейс приложения для взаимодействия пользователя с данными;
5. Осуществить тестирование и отладку приложения пользователя;
6. Оформить документацию к программному продукту.

# 1. Постановка задачи

Основные задачи, которые должна решать информационная система:

**хранить данные о спортсменах:**

* Фамилия
* Имя
* Отчество
* Дата рождения
* Пол
* Тренер
* Уровень мастерства: кандидат в мастера спорта (КМС), мастер спорта (МС), заслуженный мастер спорта (ЗМС)
* Адрес
* Телефон
* Травмы (вид травмы, когда получена, когда излечена, фамилия врача).;

**хранить данные о тренерах:**

* Фамилия
* Имя
* Отчество
* Уровень мастерства
* Паспортные данные
* Адрес
* Телефон.

**хранить данные о соревнованиях:**

* Название
* Дата начала
* Дата окончания
* Место проведения
* Возрастные ограничения
* Организатор
* Результаты (участник, показанный результат, занятое место).

**- обеспечивать учёт травм, полученных спортсменами;**

- обеспечить учет уровня мастерства спортсменов (когда получено, номер удостоверения, скан удостоверения) кандидат в мастера спорта (КМС), мастер спорта (МС), заслуженный мастер спорта (ЗМС)

- обеспечивать учет у какого тренера в какой момент тренируется спортсмен.

**помогать в подборе возможных кандидатур на участие в соревнованиях (соответствующего возраста и без текущих травм).**

Каждый спортсмен тренируется у одного тренера.

Каждый тренер может тренировать несколько спортсменов.

Спортсмен может переходить от одного тренера к другому.

каждом соревновании может участвовать несколько спортсменов.

Спортсмен может участвовать в нескольких соревнованиях.

**Техническое Задание**

Спроектировать базу данных спортивного клуба.

# 2. Концептуальная модель базы данных "Спортивный клуб"

|  |  |
| --- | --- |
| **Сущности, связи** | **Атрибуты сущностей, связей** |
| Спортсмены | Идентификатор спортсмена  ФИО спортсмена  Дата рождения  Идентификатор тренера  Идентификатор уровня мастерства  Адрес |
| Тренера | Идентификатор тренера  ФИО тренера  Идентификатор уровня мастерства  Паспортные данные  Адрес  Идентификатор телефона |
| Соревнования | Идентификатор соревнования  Наименование соревнования  Дата начала  Дата окончания  Место проведения  Возрастные ограничения  Организатор  Идентификатор спортсмена |
| Переходы спортсменов | Идентификатор спортсмена  Идентификатор тренера  Дата начала  Дата окончания |
| Травмы спортсменов | Идентификатор спортсмена  Идентификатор травмы  Дата начала  Дата окончания  Фамилия доктора |
| Травмы | Идентификатор травмы  Наименование травмы |
| Уровень мастерства | Идентификатор уровня мастерства  Наименование уровня мастерства |
| Телефон | Идентификатор телефона  Тип телефона |

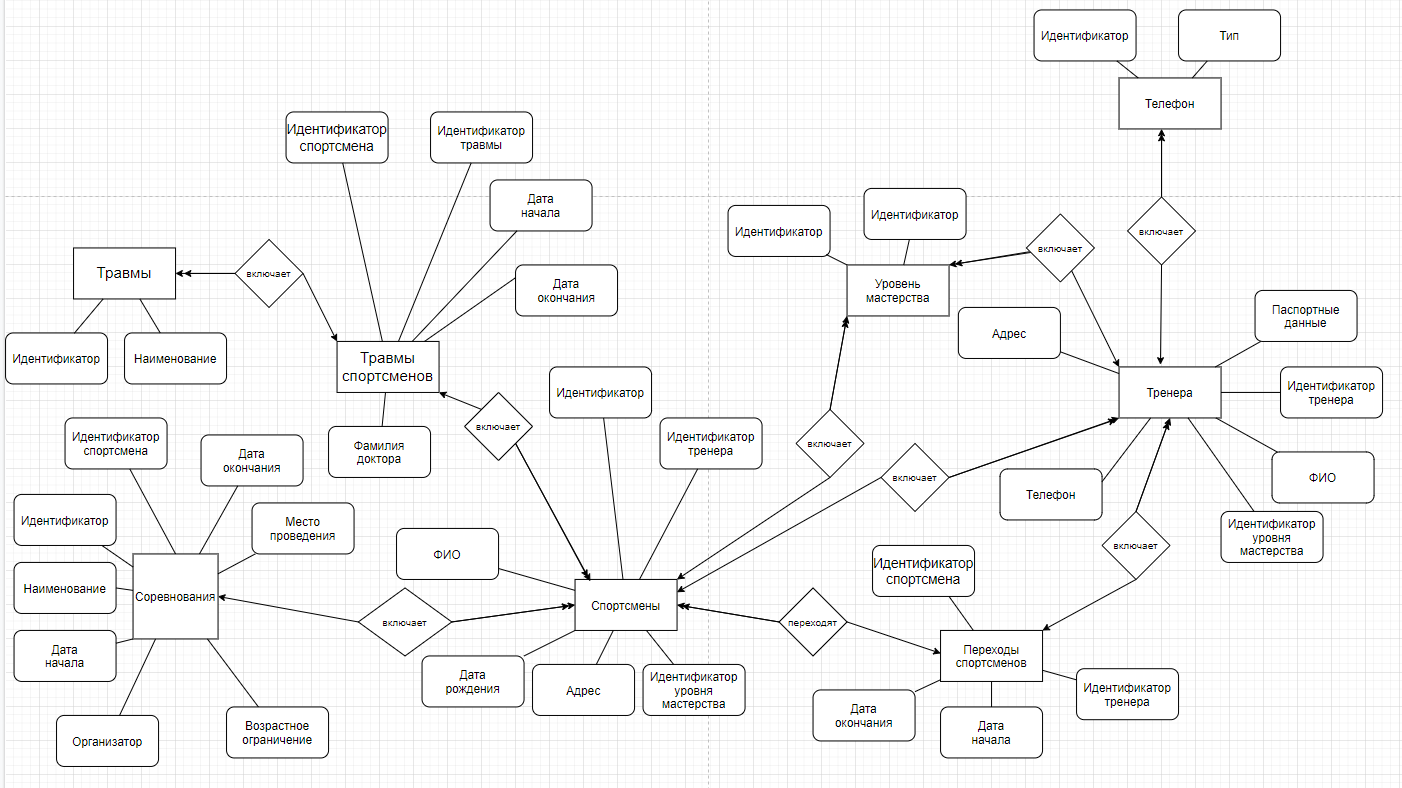


Рис. 1 Модель Сущность-связь БД "Спортивный клуб"

# 3. Логическая модель "Model\_0" базы данных "SportClub"

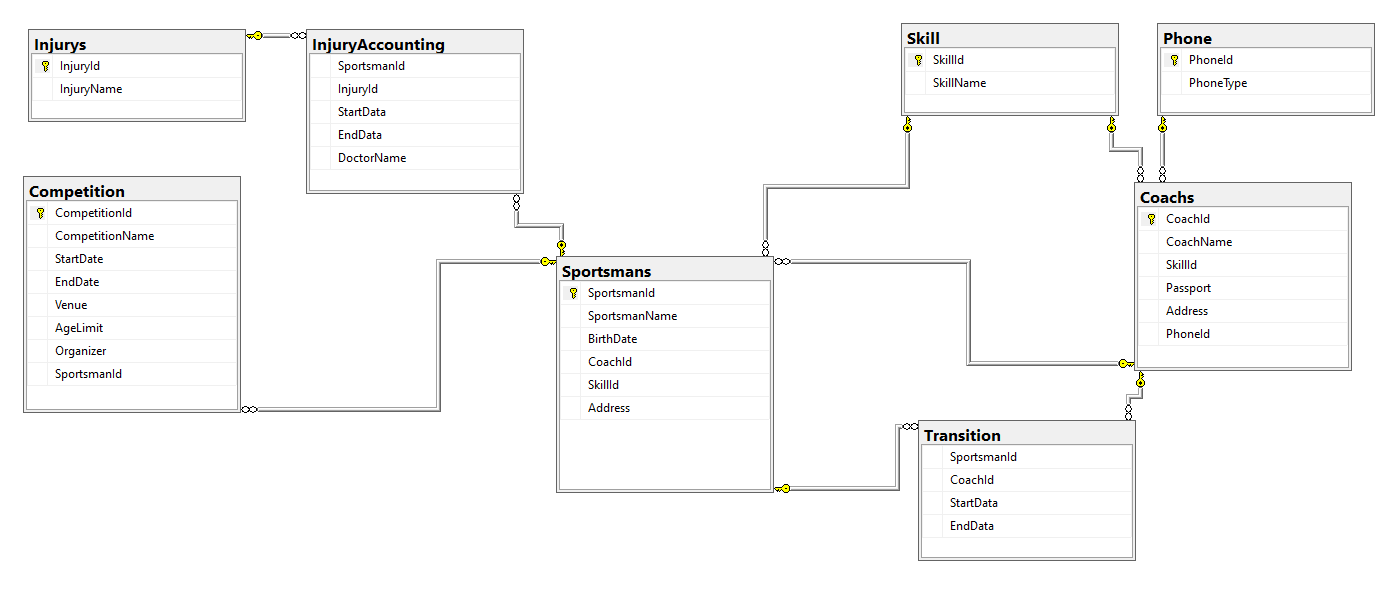


Рис. 2 Схема данных SportClub

Таблица 1

Связи между таблицами базы данных SportClub

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название связи, тип | Главная таблица | Название дочерней таблицы | Поле для звязи гл. таблицы | Поле для дочерней таблицы |
| FK\_Competition\_Sportsmans | Competition | Sportsmans | SportsmanId | SportsmanId |
| FK\_Sportsmans\_Coachs | Sportsmans | Coachs | CoachId | CoachId |
| FK\_Sportsmans\_Skill | Sportsmans | Skill | SkillId | SkillId |
| FK\_Transition\_Sportsmans | Transition | Sportsmans | SportsmanId | SportsmanId |
| FK\_Coachs\_Phone | Coachs | Phone | PhoneId | PhoneId |
| FK\_Coachs\_Skill | Coachs | Skill | SkillId | SkillId |
| FK\_Transition\_Coachs | Transition | Coachs | CoachId | CoachId |
| FK\_InjuryAccounting\_Injurys | InjuryAccounting | Injurys | InjuryId | InjuryId |
| FK\_InjuryAccounting\_Sportsmans | InjuryAccounting | Sportsmans | SportsmanId | SportsmanId |

# 4. Разработка физической модели базы данных SportClub

Структура таблицы Sportsmans

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| SportsmanId | integer | Целый, 4 байта, Первичный ключ | Идентификатор спортсмена |
| SportsmanName | nchar(50) | Строка, 60 байтов | ФИО |
| BirthDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата рождения |
| CoachId | integer | Целый, 4 байта | Идентификатор тренера |
| SkillId | integer | Целый, 4 байта | Уровень мастерства |
| Address | nchar(50) | Строка: 60 байтов | Адрес |

Структура таблицы Coach

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| CoachId | integer | Целый, 4 байта, Первичный ключ | Идентификатор тренера |
| CoachName | nchar(50) | Строка: 60 байтов | ФИО |
| SkillId | integer | Целый, 4 байта | Уровень мастерства |
| Passport | nchar(50) | Строка: 60 байтов | Паспортные данные |
| Address | nchar(50) | Строка: 60 байтов | Адрес |
| PhoneId | integer | Целый, 4 байта | Телефон |

Структура таблицы Competition

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| CompetitionId | integer | Целый, 4 байта, Первичный ключ | Идентификатор соревновании |
| CompetitionName | nchar(50) | Строка: 60 байтов | Наименование |
| SkillId | integer | Целый, 4 байта | Уровень мастерства |
| StartDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата начала |
| EndDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата окончания |
| Venue | nchar(50) | Строка: 60 байтов | Место проведения |
| AgeLimit | nchar(50) | Строка: 60 байтов | Возрастные ограничения |
| Organizer | nchar(50) | Строка: 60 байтов | Организатор |
| SportsmanId | integer | Целый, 4 байта | Идентификатор спортсмена |

Структура таблицы Transition

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| SportsmanId | integer | Целый, 4 байта | Идентификатор спортсмена |
| CoachId | integer | Целый, 4 байта | Идентификатор тренера |
| StartDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата начала |
| EndDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата окончания |

Структура таблицы InjuryAccounting

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| SportsmanId | integer | Целый, 4 байта | Идентификатор спортсмена |
| InjuryId | integer | Целый, 4 байта | Идентификатор травмы |
| StartDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата начала |
| EndDate | data | Формат:DD-MM-YYYY  Размер: 3 байта | Дата окончания |
| DoctorName | nchar(50) | Строка: 60 байтов | Фамилия доктора |

Структура таблицы Injurys

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| InjuryId | integer | Целый, 4 байта, Первичный ключ | Идентификатор травмы |
| InjuryName | nchar(50) | Строка: 60 байтов | Наименование |

Структура таблицы Skill

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| SkillId | integer | Целый, 4 байта, Первичный ключ | Идентификатор уровня мастерства |
| SkillName | nchar(50) | Строка: 60 байтов | Наименование |

Структура таблицы Phone

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Свойства | Описание |
| PhoneId | integer | Целый, 4 байта, Первичный ключ | Идентификатор телефон |
| PhoneType | nchar(50) | Строка: 60 байтов | Тип |

# 5. Разработка интерфейса

При запуске приложения появляется окно MainWindow следующего вида:

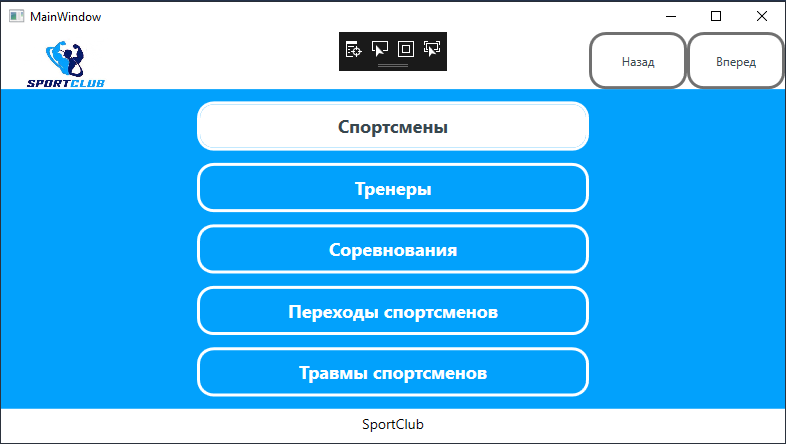


Рис. 3 Кнопочная форма

На рисунке 3 показана главная кнопочная форма. Форма содержит пять кнопок, с помощью которых можно перейти в другие разделы приложения.

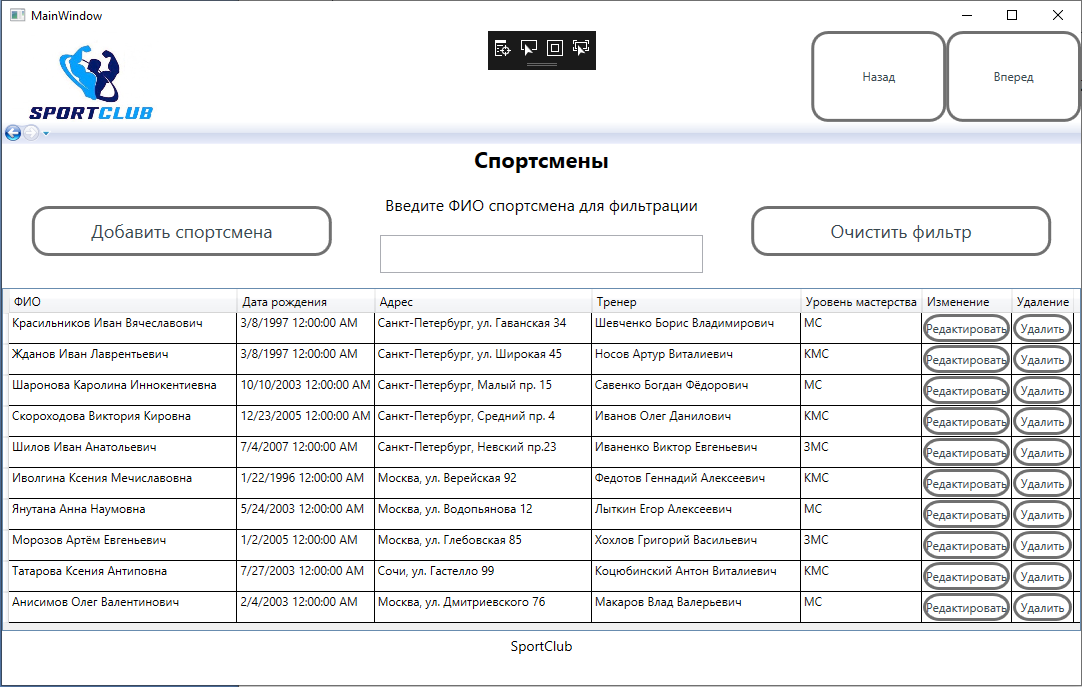


Рис. 4 Страница спортсмены

На данном рисунке показана страница "Спортсмены", также имеются кнопки "Добавить спортсмена", "Редактировать"", "Удалить". Присутсвует фильтр, который можно почистить.

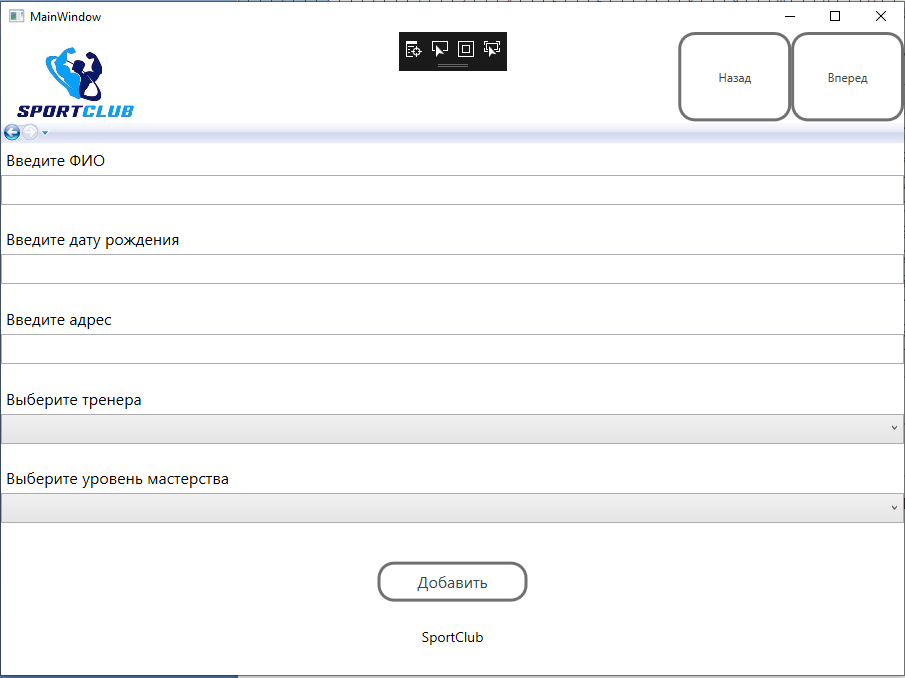


Рис. 5 Страница добавления спортсмена

На рисунке 5 имеется форма добавления спортсмена. Содержит список тренеров и уровней мастерства. Присутствует кнопка "Добавить".

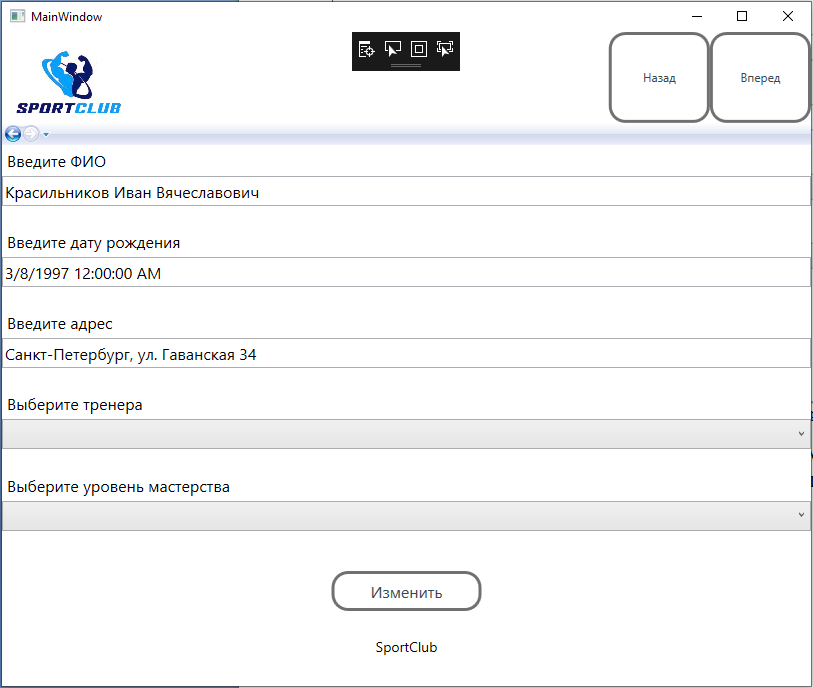


Рис 6 Страница редактирования спортсмена

На рисунке 6 имеется форма редактирования спортсмена. Содержит список тренеров и уровней мастерства. Присутствует кнопка "Изменить".

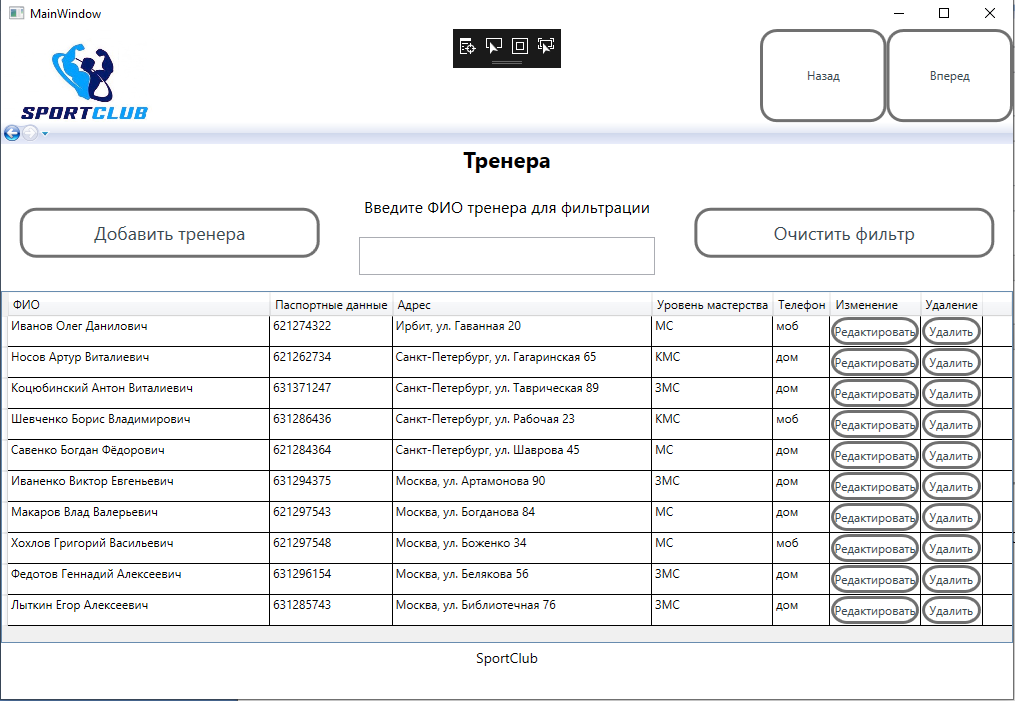


Рис. 7 Страница Тренера

На данном рисунке показана страница "Тренера", также имеются кнопки "Добавить тренера", "Редактировать", "Удалить" "Очистить фильтр". Присутсвует фильтр, который можно почистить.

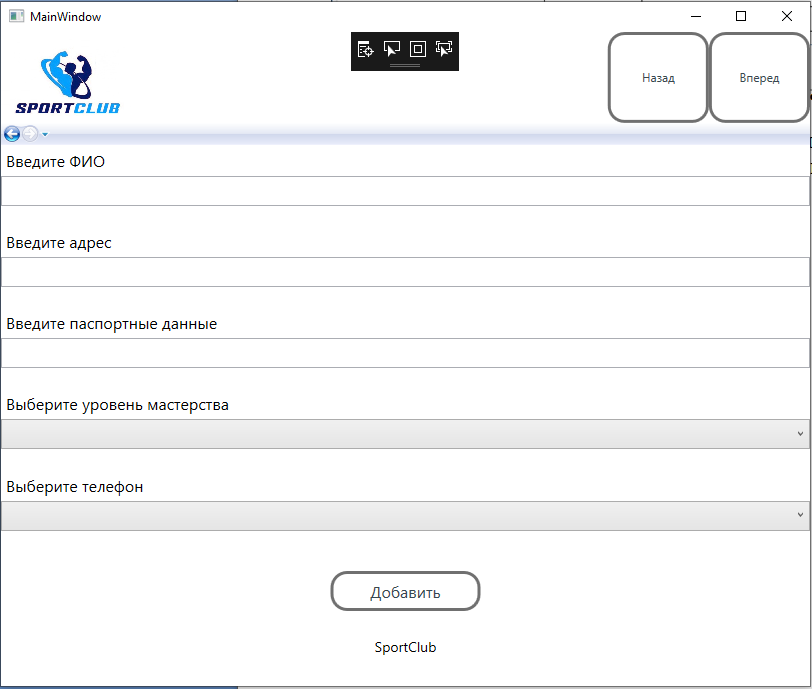


Рис. 8 Страница добавления тренера

На рисунке 8 имеется форма добавления тренера. Содержит список типов телефона и уровней мастерства. Присутствует кнопка "Добавить".

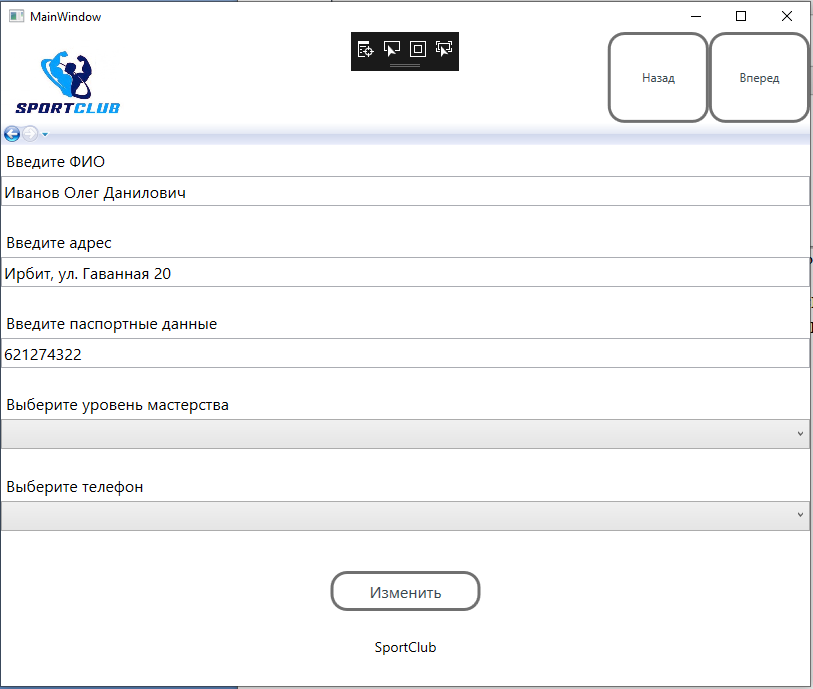


Рис. 9 Страница редактирования тренера

На рисунке 9 имеется форма редактирования тренера. Содержит типов телефона и уровней мастерства. Присутствует кнопка "Изменить".

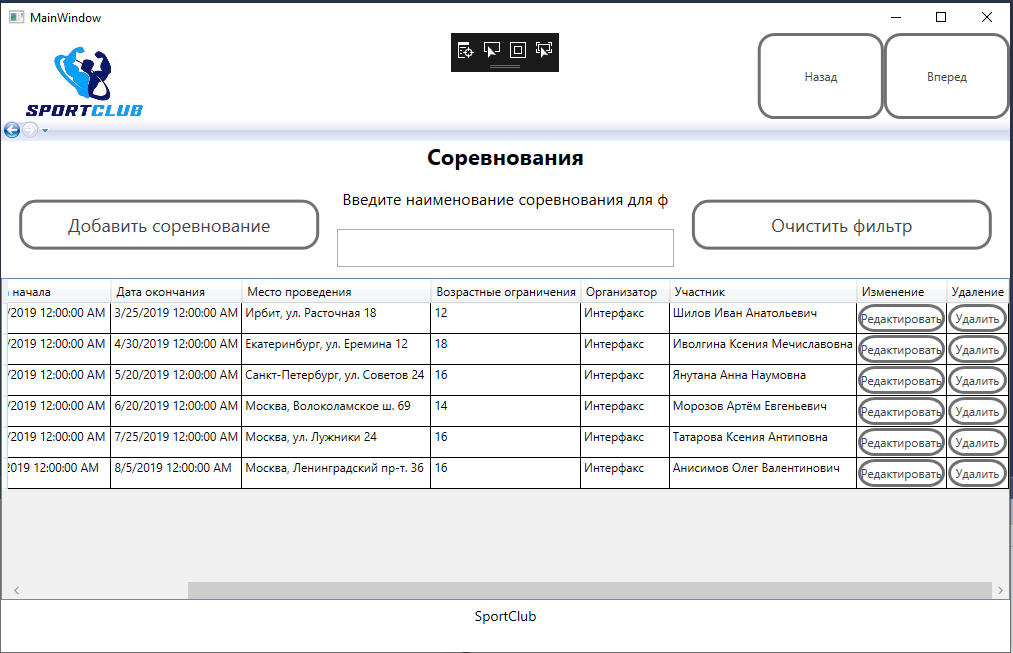


Рис. 10 Страница Соревнования

На данном рисунке показана страница "Соревнования", также имеются кнопки "Добавить соревнование", "Редактировать", "Удалить" "Очистить фильтр". Присутсвует фильтр, который можно почистить.

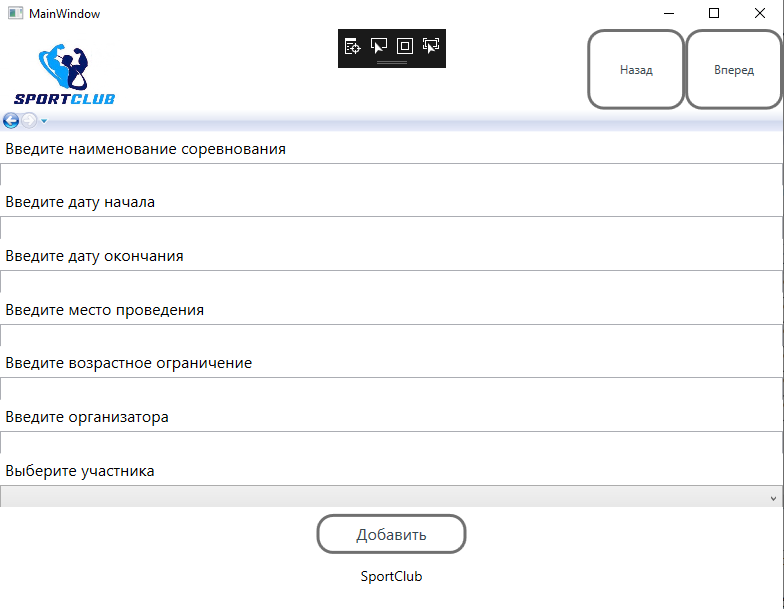


Рис. 11 Страница добавления соревнования

На рисунке 11 имеется форма добавления соревнования. Содержит список участников. Присутствует кнопка "Добавить".

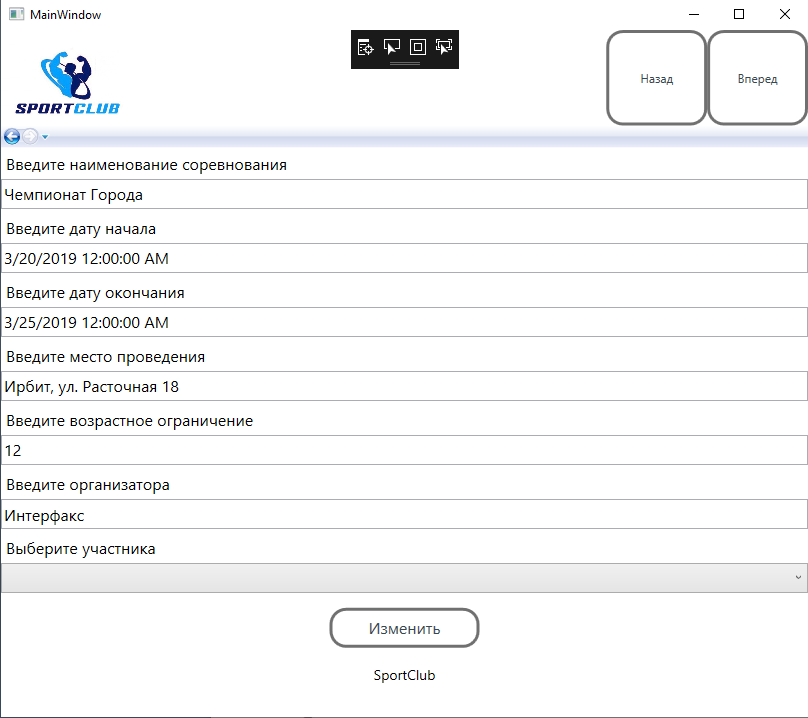


Рис. 12 Страница редактирования соревнования

На рисунке 12 имеется форма редактирования соревнования. Содержит список участников. Присутствует кнопка "Изменить".

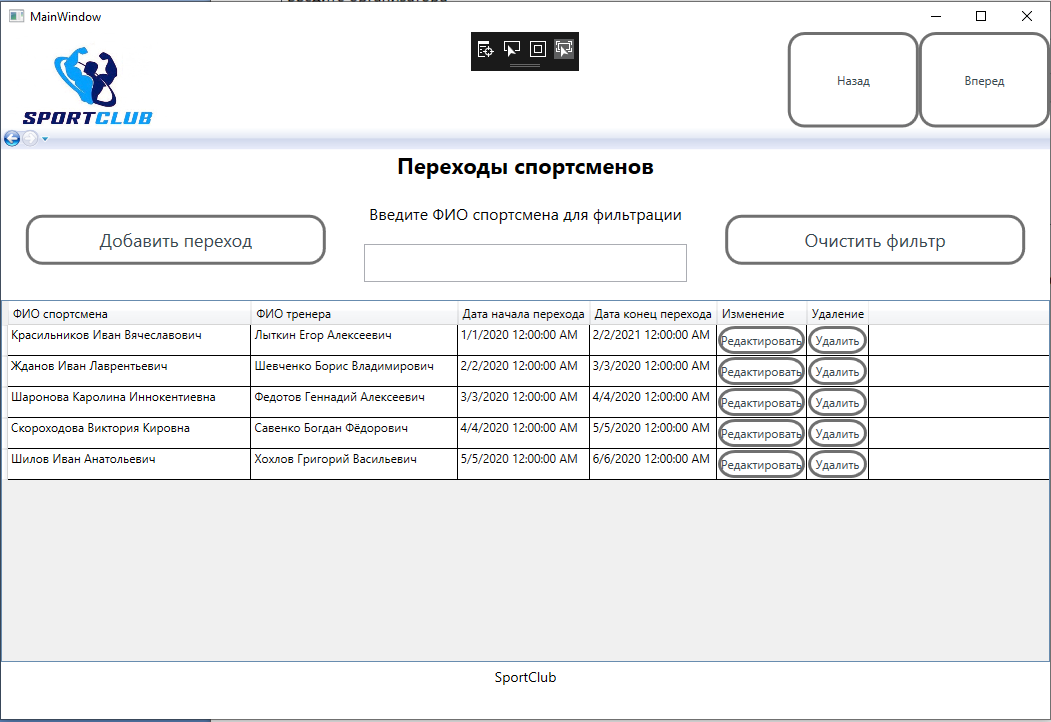


Рис. 13 Страница переходов спортсменов

На данном рисунке показана страница "Переходов спортсменов", также имеются кнопки "Добавить переход", "Редактировать", "Удалить" "Очистить фильтр". Присутсвует фильтр, который можно почистить.

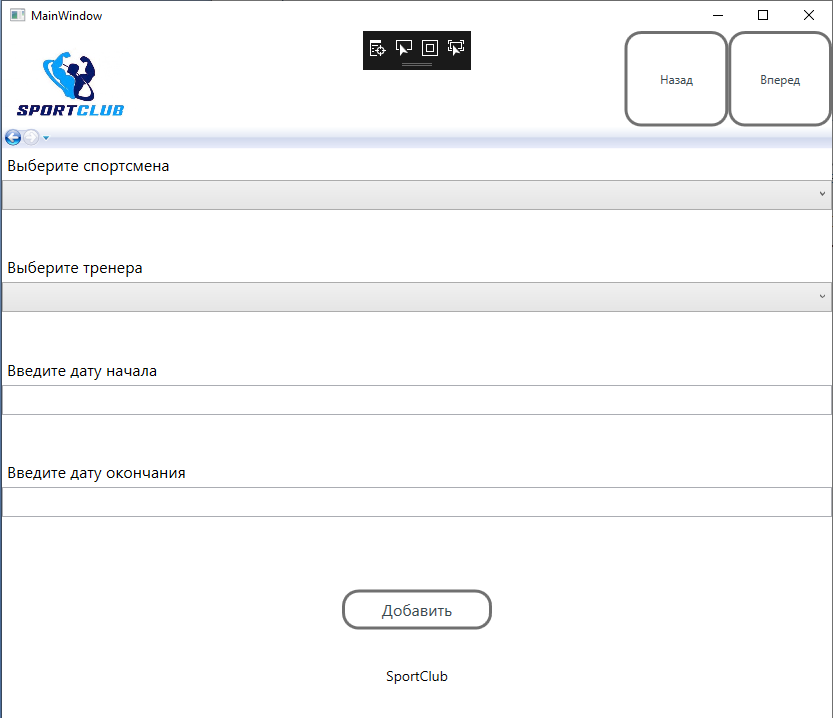


Рис. 14 страница добавления перехода спортсмена

На рисунке 14 имеется форма добавления перехода. Содержит список спортсменов, тренеров. Присутствует кнопка "Добавить".

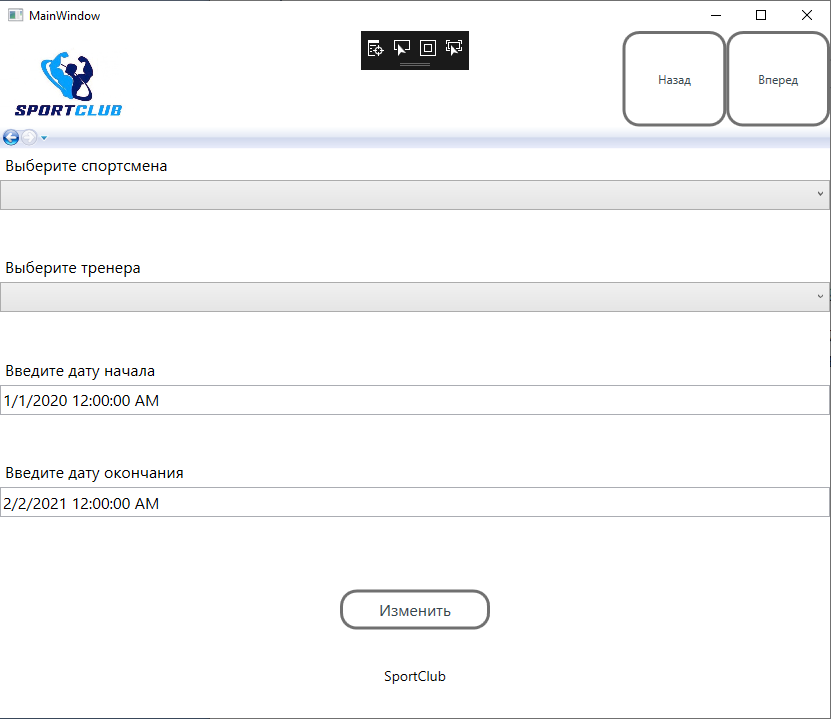


Рис. 15 страница редактирования перехода спортсмена

На рисунке 15 имеется форма редактирования перехода спортсмена. Содержит список спортсменов, тренеров. Присутствует кнопка "Изменить".

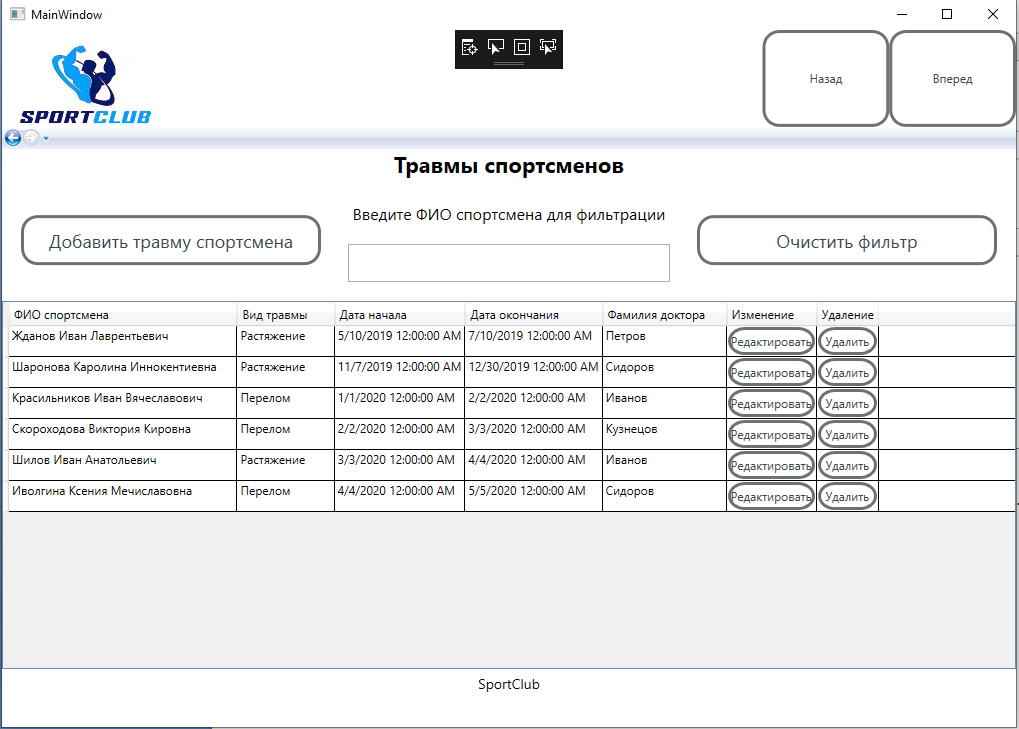


Рис. 16 страница травм спортсменов

На данном рисунке показана страница "Травм спортсменов", также имеются кнопки "Добавить травму спортсмена", "Редактировать", "Удалить" "Очистить фильтр". Присутсвует фильтр, который можно почистить.

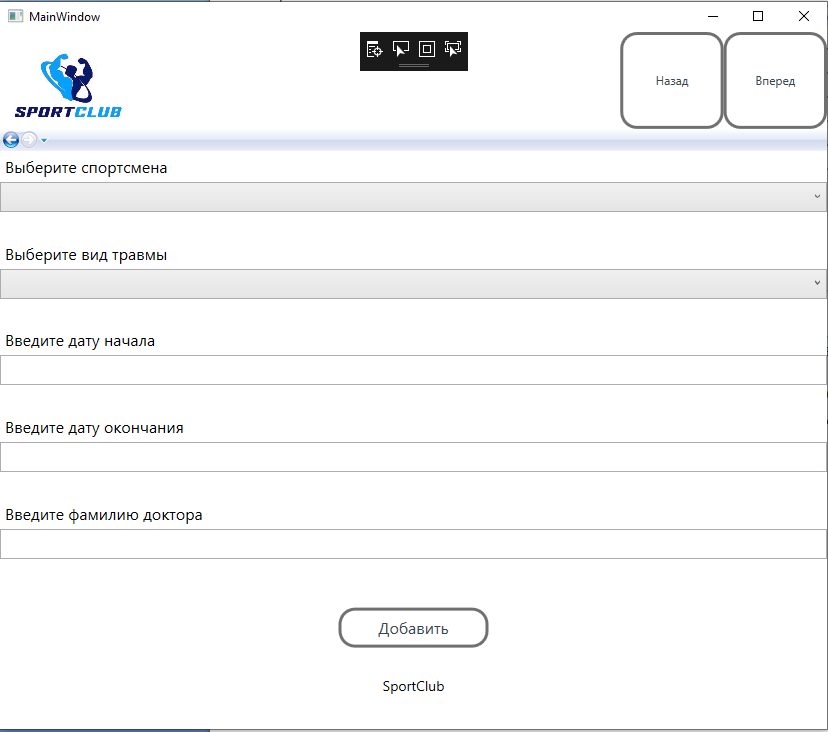


Рис. 17 страница добавления травмы спортсмена

На рисунке 17 имеется форма добавления травмы спортсмена. Содержит список спортсменов, вид травмы. Присутствует кнопка "Добавить".

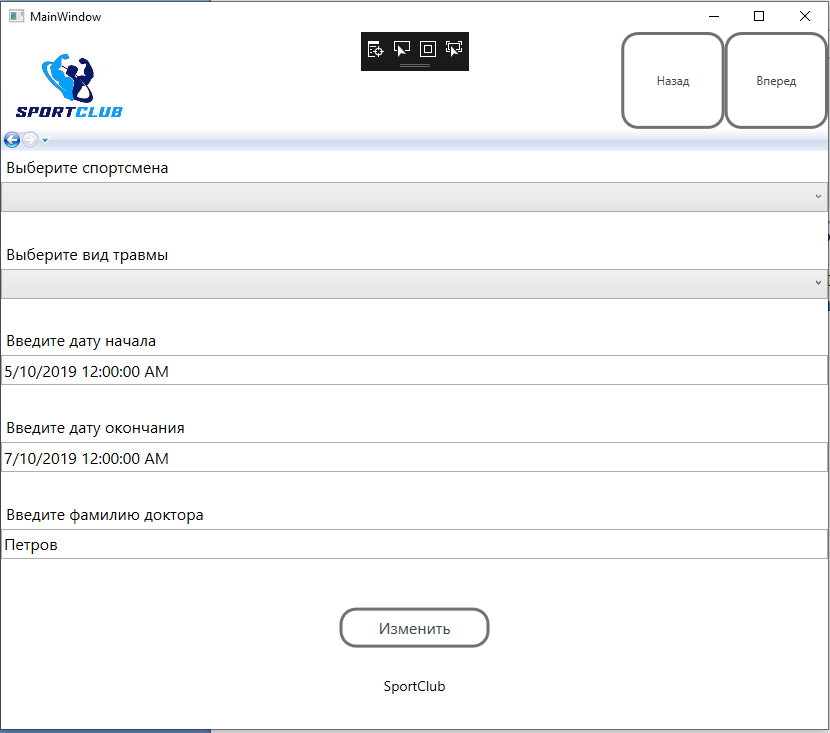


Рис. 18 страница редактирования травмы спортсмена

На рисунке 18 имеется форма редактирования травмы спортсмена. Содержит список спортсменов, вид травмы. Присутствует кнопка "Изменить".

# Заключение

За время курсового проектирования была изучена программа по работе с базами данных Microsoft SQL Server Management Studio 17, среда разработки Microsoft Visual Studio 2017 с модулем Entity Framework 6, с помощью которого можно создавать прикладные приложения с использованием баз данных. Приобретен практический опыт СУБД в MS SQL.

Таким образом, поставленная цель "Проектирование и разработка базы данных "Спортивный клуб" в MS SQL Server и создание приложения в среде Visual Studio с помощью WPF форм, языка C# и технологии Entity Framework" достигнута.

Следует отметить, что данный программный продукт может быть использован на небольших фирмах. При необходимости, данное программное приложение может быть скорректировано под запросы и задачи определенного спортивного клуба. Преимущества данной программы, заключаются в том, что этот продукт позволяет решать конкретные задачи и не требует специальных знаний в программировании, т.к. интерфейс интуитивно понятен.

# Приложение

## Приложение 1 Программный код MainWindow.xaml

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

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="20\*"></ColumnDefinition>

<ColumnDefinition Width="70\*"></ColumnDefinition>

<ColumnDefinition Width="15\*"></ColumnDefinition>

<ColumnDefinition Width="15\*"></ColumnDefinition>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="25\*"></RowDefinition>

<RowDefinition Height="140\*"></RowDefinition>

<RowDefinition Height="15\*"></RowDefinition>

</Grid.RowDefinitions>

<Image Grid.Column="0" Grid.Row="0" Source="images\logo.jpg"></Image>

<Frame Name="MyFrame" Grid.Row="1" Grid.Column="0" Grid.ColumnSpan="4" Background="White"></Frame>

<Button Style="{StaticResource myButton}" Name="ForwardButton" Grid.Row="0" Grid.Column="3" Content="Вперед" Click="ForwardButton\_Click"></Button>

<Button Style="{StaticResource myButton}" Name="BackButton" Grid.Row="0" Grid.Column="2" Content="Назад" Click="BackButton\_Click"></Button>

<Label Grid.Row="3" Grid.ColumnSpan="4" Content="SportClub" HorizontalAlignment="Center" FontSize="14"></Label>

</Grid>

</Window>

## Приложение 2 Программный 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 SportClub

{

/// Логика взаимодействия для MainWindow.xaml

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

MyFrame.Navigate(new pages.MainPage());

}

/// Обработка кнопки Вперед

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

{

if (MyFrame.CanGoForward)

{

MyFrame.GoForward();

}

else

{

MessageBox.Show("Ошибка, вы находитесь на главной странице", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

}

}

/// Обработка кнопки Назад

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

{

if (MyFrame.CanGoBack)

{

MyFrame.GoBack();

}

else

{

MessageBox.Show("Ошибка, вы находитесь на главной странице", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

}

}

}

}

## Приложение 3 Программный код MainPage.xaml

<Page x:Class="SportClub.pages.MainPage"

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:SportClub.pages"

mc:Ignorable="d"

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

Title="MainPage">

<Grid>

<Grid Background="#02A1FC">

<Grid.RowDefinitions>

<RowDefinition Height="10\*"/>

<RowDefinition Height="40\*"/>

<RowDefinition Height="10\*"/>

<RowDefinition Height="40\*"/>

<RowDefinition Height="10\*"/>

<RowDefinition Height="40\*"/>

<RowDefinition Height="10\*"/>

<RowDefinition Height="40\*"/>

<RowDefinition Height="10\*"/>

<RowDefinition Height="40\*"/>

<RowDefinition Height="10\*"/>

</Grid.RowDefinitions>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="20\*"/>

<ColumnDefinition Width="40\*"/>

<ColumnDefinition Width="20\*"/>

</Grid.ColumnDefinitions>

<Button Click="Sportsmans"

Grid.Column="1" Grid.Row="1"

FontSize="18" FontWeight="Bold">Спортсмены</Button>

<Button Click="Coachss"

Grid.Column="1" Grid.Row="3"

FontSize="18" FontWeight="Bold">Тренеры</Button>

<Button Click="Competitions"

Grid.Column="1" Grid.Row="5"

FontSize="18" FontWeight="Bold">Соревнования</Button>

<Button Click="Transition"

Grid.Column="1" Grid.Row="7"

FontSize="18" FontWeight="Bold">Переходы спортсменов</Button>

<Button Click="InjuryAccountings"

Grid.Column="1" Grid.Row="9"

FontSize="18" FontWeight="Bold">Травмы спортсменов</Button>

</Grid>

</Grid>

</Page>

## Приложение 4 Программный код MainPage.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 SportClub.pages

{

/// <summary>

/// Логика взаимодействия для MainPage.xaml

/// </summary>

public partial class MainPage : Page

{

public MainPage()

{

InitializeComponent();

}

private void Sportsmans(object sender, RoutedEventArgs e)

{

this.NavigationService.Navigate(new Sportsman());

//MessageBox.Show("Извините, временно режим не работает");

}

private void Coachss(object sender, RoutedEventArgs e)

{

this.NavigationService.Navigate(new Coachss());

//MessageBox.Show("Извините, временно режим не работает");

}

private void Competitions(object sender, RoutedEventArgs e)

{

this.NavigationService.Navigate(new Competitions());

//MessageBox.Show("Извините, временно режим не работает");

}

private void Transition(object sender, RoutedEventArgs e)

{

this.NavigationService.Navigate(new Trasitions());

//MessageBox.Show("Извините, временно режим не работает");

}

private void InjuryAccountings(object sender, RoutedEventArgs e)

{

this.NavigationService.Navigate(new InjuryAccountings());

//MessageBox.Show("Извините, временно режим не работает");

}

}Приложение 5 Программный код Sportsmans.xaml

<Page x:Class="SportClub.pages.Sportsman"

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:SportClub.pages"

mc:Ignorable="d"

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

Title="Sportsmans">

<Grid Background="#fff">

<Grid.RowDefinitions>

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

<RowDefinition></RowDefinition>

<RowDefinition Height="3\*"></RowDefinition>

</Grid.RowDefinitions>

<TextBlock FontSize="22" FontWeight="Bold" VerticalAlignment="Center" HorizontalAlignment="Center">Спортсмены</TextBlock>

<Grid Grid.Row="1">

<Grid.ColumnDefinitions>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

<ColumnDefinition></ColumnDefinition>

</Grid.ColumnDefinitions>

<Button Click="ClearFilter" Style="{StaticResource myButton}" Grid.Column="2" HorizontalAlignment="Center" VerticalAlignment="Center" Height="50" Width="300" FontSize="18">Очистить фильтр</Button>

<Button Click="AddSportsmans" Style="{StaticResource myButton}" Grid.Row="1" Margin="15" HorizontalAlignment="Center" Height="50" Width="300" FontSize="18">Добавить спортсмена</Button>

<StackPanel VerticalAlignment="Center" HorizontalAlignment="Center" Grid.Column="1">

<Label FontSize="16">Введите ФИО спортсмена для фильтрации</Label>

<TextBox TextChanged="FilterName" Name="NameFilter" Margin="0 15 0 0" Height="38" VerticalContentAlignment="Center"></TextBox>

</StackPanel>

</Grid>

<DataGrid IsReadOnly="True" Name="SportsmansList" Grid.Row="2" AutoGenerateColumns="False" CanUserAddRows="False">

<DataGrid.Columns>

<DataGridTextColumn Header="ФИО" Binding="{Binding SportsmanName}"></DataGridTextColumn>

<DataGridTextColumn Header="Дата рождения" Binding="{Binding BirthDate}"></DataGridTextColumn>

<DataGridTextColumn Header="Адрес" Binding="{Binding Address}"></DataGridTextColumn>

<DataGridTextColumn Header="Тренер" Binding="{Binding Coachs.CoachName}"></DataGridTextColumn>

<DataGridTextColumn Header="Уровень мастерства" Binding="{Binding Skill.SkillName}"></DataGridTextColumn>

<DataGridTemplateColumn>

<DataGridTemplateColumn.Header>Изменение</DataGridTemplateColumn.Header>

<DataGridTemplateColumn.CellTemplate>

<DataTemplate>

<Button Click="EditSportsmans" Style="{StaticResource myButton}" Height="28">Редактировать</Button>

</DataTemplate>

</DataGridTemplateColumn.CellTemplate>

</DataGridTemplateColumn>

<DataGridTemplateColumn>

<DataGridTemplateColumn.Header>Удаление</DataGridTemplateColumn.Header>

<DataGridTemplateColumn.CellTemplate>

<DataTemplate>

<Button Click="DeleteS" Style="{StaticResource myButton}" Height="28">Удалить</Button>

</DataTemplate>

</DataGridTemplateColumn.CellTemplate>

</DataGridTemplateColumn>

</DataGrid.Columns>

</DataGrid>

</Grid>

</Page>

## Приложение 6 Программный код Sportsmans.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 SportClub.pages

{

/// Логика взаимодействия для Sportsmans.xaml

public partial class Sportsman : Page

{

public int? SportsmanId { get; internal set; }

public void LoadList()

{

var test = DB.Context.Sportsmans.ToList();

SportsmansList.ItemsSource = DB.Context.Sportsmans.ToList();

}

public Sportsman()

{

InitializeComponent();

LoadList();

}

private void FilterName(object sender, TextChangedEventArgs e)

{

string name = NameFilter.Text;

var dataFilter = DB.Context.Sportsmans.Where(g => g.SportsmanName.Contains(name));

SportsmansList.ItemsSource = dataFilter.ToList();

}

private void EditSportsmans(object sender, RoutedEventArgs e)

{

var selectedItem = SportsmansList.SelectedItem as Sportsmans;

this.NavigationService.Navigate(new EditSportsmans(selectedItem));

}

private void DeleteS(object sender, RoutedEventArgs e)

{

if (MessageBox.Show("Вы точно хотите удалить запись?", "Сообщение", MessageBoxButton.YesNo, MessageBoxImage.Information) == MessageBoxResult.Yes)

{

var SelectedItem = SportsmansList.SelectedItem as Sportsmans;

DB.Context.Sportsmans.Remove(SelectedItem);

DB.Context.SaveChangesAsync();

MessageBox.Show("Запись успешно удалена", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

LoadList();

}

}

private void AddSportsmans(object sender, RoutedEventArgs e)

{

this.NavigationService.Navigate(new AddSportsmans());

//MessageBox.Show("Извините, временно режим не работает");

}

private void ClearFilter(object sender, RoutedEventArgs e)

{

NameFilter.Text = "";

}

## Приложение 7 Программный код AddSportsmans.xaml

<Page x:Class="SportClub.pages.AddSportsmans"

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:SportClub.pages"

mc:Ignorable="d"

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

Title="AddSportsmans">

<Grid Background="#fff">

<Grid.RowDefinitions>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

</Grid.RowDefinitions>

<StackPanel Grid.Row="0">

<Label FontSize="16">Введите ФИО</Label>

<TextBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="NameSportsmans"></TextBox>

</StackPanel>

<StackPanel Grid.Row="1">

<Label FontSize="16">Введите дату рождения</Label>

<TextBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="BirthDateSportsmans"></TextBox>

</StackPanel>

<StackPanel Grid.Row="2">

<Label FontSize="16">Введите адрес</Label>

<TextBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="AddressSportsmans"></TextBox>

</StackPanel>

<StackPanel Grid.Row="3">

<Label FontSize="16">Выберите тренера</Label>

<ComboBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="CoachSportsmans"></ComboBox>

</StackPanel>

<StackPanel Grid.Row="4">

<Label FontSize="16">Выберите уровень мастерства</Label>

<ComboBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="SkillSportsmans"></ComboBox>

</StackPanel>

<Button FontSize="16" Style="{StaticResource myButton}" VerticalAlignment="Center" HorizontalAlignment="Center" Width="150" Height="40" Click="AddS" Grid.Row="5">Добавить</Button>

</Grid>

</Page>

## Приложение 8 Программный код AddSportsmans.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 SportClub.pages

{

/// <summary>

/// Логика взаимодействия для AddSportsmans.xaml

/// </summary>

public partial class AddSportsmans : Page

{

public AddSportsmans()

{

InitializeComponent();

SkillSportsmans.ItemsSource = DB.Context.Skill.ToList();

SkillSportsmans.DisplayMemberPath = "SkillName";

CoachSportsmans.ItemsSource = DB.Context.Coachs.ToList();

CoachSportsmans.DisplayMemberPath = "CoachName";

}

private void AddS(object sender, RoutedEventArgs e)

{

var selectedItem = SkillSportsmans.SelectedItem as Skill;

var selectedItems = CoachSportsmans.SelectedItem as Coachs;

try

{

var dataS = new Sportsmans

{

SportsmanName = NameSportsmans.Text,

BirthDate = DateTime.Parse(BirthDateSportsmans.Text),

Address = AddressSportsmans.Text,

SkillId = selectedItem.SkillId,

CoachId = selectedItems.CoachId

};

DB.Context.Sportsmans.Add(dataS);

DB.Context.SaveChangesAsync();

MessageBox.Show("Запись успешно добавлена", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

this.NavigationService.Navigate(new Sportsman());

}

catch (Exception)

{

MessageBox.Show("Произошла ошибка при добавлении! Введены некоректные данные!", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

}

}

}

}

## Приложение 9 Программный код EditSportsmans.xaml

<Page x:Class="SportClub.pages.EditSportsmans"

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:SportClub.pages"

mc:Ignorable="d"

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

Title="EditSportsmans">

<Grid Background="#fff">

<Grid.RowDefinitions>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

<RowDefinition></RowDefinition>

</Grid.RowDefinitions>

<StackPanel Grid.Row="0">

<Label FontSize="16">Введите ФИО</Label>

<TextBox FontSize="16" Height="30" VerticalContentAlignment="Center" Text="{Binding tempData.SportsmanName}"></TextBox>

</StackPanel>

<StackPanel Grid.Row="1">

<Label FontSize="16">Введите дату рождения</Label>

<TextBox FontSize="16" Height="30" VerticalContentAlignment="Center" Text="{Binding tempData.BirthDate}"></TextBox>

</StackPanel>

<StackPanel Grid.Row="2">

<Label FontSize="16">Введите адрес</Label>

<TextBox FontSize="16" Height="30" VerticalContentAlignment="Center" Text="{Binding tempData.Address}"></TextBox>

</StackPanel>

<StackPanel Grid.Row="3">

<Label FontSize="16">Выберите тренера</Label>

<ComboBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="CoachSportsmans"></ComboBox>

</StackPanel>

<StackPanel Grid.Row="4">

<Label FontSize="16">Выберите уровень мастерства</Label>

<ComboBox FontSize="16" Height="30" VerticalContentAlignment="Center" Name="SkillSportsmans"></ComboBox>

</StackPanel>

<Button FontSize="16" Style="{StaticResource myButton}" VerticalAlignment="Center" HorizontalAlignment="Center" Width="150" Height="40" Click="saveChangesS" Grid.Row="5">Изменить</Button>

</Grid>

</Page>

## Приложение 10 Программный код EditSportsmans.xaml.cs

namespace SportClub.pages

{

public partial class EditSportsmans : Page

{

public List<Coachs> Coaches { get; set; }

public Sportsmans tempData { get; set; }

public Skill skill;

public Coachs Coachs;

public EditSportsmans(Sportsmans sportsmans)

{

tempData = sportsmans;

Coachs = sportsmans.Coachs;

InitializeComponent();

Coaches = DB.Context.Coachs.ToList();

this.DataContext = this;

CoachSportsmans.ItemsSource = DB.Context.Coachs.ToList();

CoachSportsmans.DisplayMemberPath = "CoachName";

SkillSportsmans.ItemsSource = DB.Context.Skill.ToList();

SkillSportsmans.DisplayMemberPath = "SkillName";

}

private void saveChangesS(object sender, RoutedEventArgs e)

{

int count = 0;

var sportsmans = DB.Context.Coachs.Where(c => c.CoachId == tempData.CoachId).ToList();

foreach (var coach in Coaches)

{

foreach (var Coach in Coaches)

{

if (Coach.CoachId == 1)

{

count++;

}

}

}

if (tempData.Skill.SkillId == 2 && count != 0)

{

MessageBox.Show("Нельзя изменить!", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

}

else

{

if (tempData.SportsmanName.Equals("") || tempData.BirthDate.Equals("") || tempData.Address.Equals("") || tempData.Coachs.CoachName.Equals("") || tempData.Skill.SkillName.Equals(""))

{

MessageBox.Show("Произошла ошибка при изменении! Введены некоректные данные!", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

}

else

{

try

{

DB.Context.SaveChanges();

if (MessageBox.Show("Запись успешно изменена", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information) == MessageBoxResult.OK)

{

this.NavigationService.GoBack();

}

}

catch (Exception)

{

MessageBox.Show("Произошла ошибка при сохранении :(", "Сообщение", MessageBoxButton.OK, MessageBoxImage.Information);

}

# Список использованной литературы

1. Федорова Г. Разработка, администрирование и защита баз данных : учебник для студентов учреждений среднего профессионального образования - М .: Издательский центр «Академия», 2017. — 286 с.
2. Федорова, Г.Н. Основы проектирования баз данных учебное пособие для студ. учрежд. СПО / Г.Н.Федорова. - М.: Академия, 2014.
3. Федорова, Г.Н. Разработка и администрирование баз данных: учебник / Г.Н.Федорова. - М.: Академия, 2015.
4. Семакин И. Г. Основы программирования и баз данных: учебник для студентов учреждений среднего профессионального образования. - 2-е издание, стереотипное. - Москва: Академия, 2017. 219 с.
5. Фуфаев Э. В. Базы данных: учеб. пособие дли студ. учреждений сред, проф. образования / Э. В.Фуфаев, Д. Э.Фуфаев. — 7-е изд., стер. — М .: Издательский центр «Академия», 2012. — 320 с.
6. Котеров Д.В. Симдянов И.В. PHP/7 - СПб.: БХВ-Петербург, 2016., 1088с.
7. Агальцов, В.П. Базы данных. В 2-х т. Т. 2. Распределенные и удаленные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 272 c.
8. Агальцов, В.П. Базы данных. В 2-х т. Т. 1. Локальные базы данных: Учебник / В.П. Агальцов. - М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2013. - 352 c.
9. Карпова, И.П. Базы данных: Учебное пособие / И.П. Карпова. - СПб.: Питер, 2013. - 240 c.
10. Гагарина, Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учебное пособие для студ. учрежд. СПО / Л.Г.Гагарина. - М.: ФОРУМ: ИНФРА-М, 2017.
11. Голицына, О.Л. Основы проектирования баз данных: учебное пособие для студ. учрежд. СПО / О.Л.Голицына. - М.: ФОРУМ: ИНФРА-М, 2017.
12. Голицына О. Л., Максимов Н. В., Попов И. И. Базы данных: учеб. пособие. — 2-е изд., испр. и доп. — М.: ФОРУМ: ИНФРА-М, 2009. — 400 с.: ил. — (Профессиональное образование).
13. Кевин Янк PHP и MySQL от новичка к профессионалу М. : Эксмо, 2013. – 384 с. – (Мировой компьютерный бестселлер).
14. Советов, Б.Я. Базы данных: теория и практика: / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовской. - М.: Юрайт, 2013. - 463 c.
15. К.Э.Плохотникова и др. Методы разработки курсовых работ - М.: СОЛОН-ПРЕСС, 2006