# Пояснительная записка

# к контрольному домашнему заданию

# по дисциплине «Программирование»

## Образовательная программа «Бизнес-информатика»

## Бакалавриат 1-й курс

Решетцов Иван, группа 164

1. **Тема проекта:** «Разработка приложения (информационной системы) ATP (Ассоциации теннисистов-профессионалов)»
2. **Репозиторий:** <https://github.com/IvanReshettsov>
3. **В проекте доступна** регистрация новых пользователей с последующей авторизацией их в системе. Неавторизованный пользователь может работать в программе в режиме чтения(read-only-mode). Ему доступны страницы PlayerPage и TournamentPage, на которых он имеет возможность выводить в таблицу данные по игрокам(имя, страна, тренер, место в рейтинге, физические параметры, количество выигранных турниров) и по турниром(название, место и дата проведения, супервайзер, призовой фонд). Неавторизованным пользователям также доступна функция поиска элемента в списке(с помощью выпадающего списка в комбобоксе). При введенном имени игрока или названии турнира в комбобокс и при нажатии кнопки SEARCH таблица пролистывается к строке с элементом, удовлетворяющем введенным данным, и выделяет ее. Функция сериализации и десериализации также открыта для неавторизованных пользователей. Навигация в приложении осуществляется с помощью кнопок GO BACK, которые позволяют перемещаться на предыдущую страницу.   
   После авторизации пользователю становятся доступны функции добавления и удаления элементов в списке, которые осуществляются на страницах NewPlayerPage и NewTournamentPage. Также авторизованный пользователь способен выйти из системы, нажав в главном окне кнопку LOG OUT и подтвердив в появившемся окне свой выбор.
4. **В программе использованы технологии:** - Графическое приложение(WPF)

- Операции с данными: создание, редактирование, удаление, чтение (CRUD) + поиск

- Хранение данных в файле

- Защита от некорректного ввода

- Единый стиль именования (Naming Convention)

- Использование GitHub

- Несколько сущностей

- Сериализация

- Страничная навигация (по аналогии с мобильным приложением)

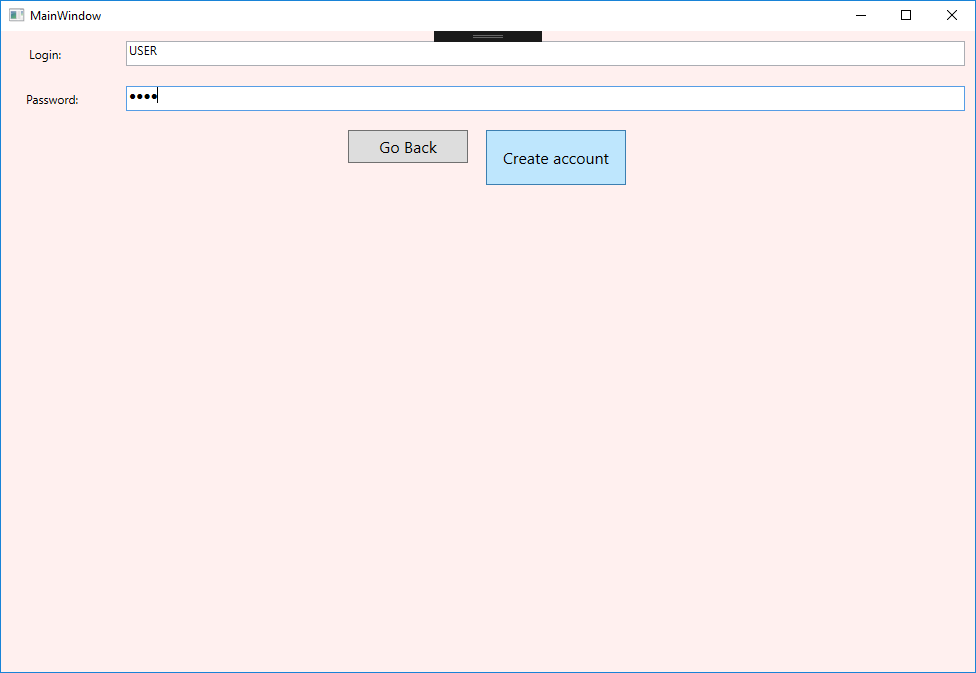
- Анимация(графика)

- Авторизация (доступ в режиме чтения для всех пользователей, редактирование – только для авторизованных)

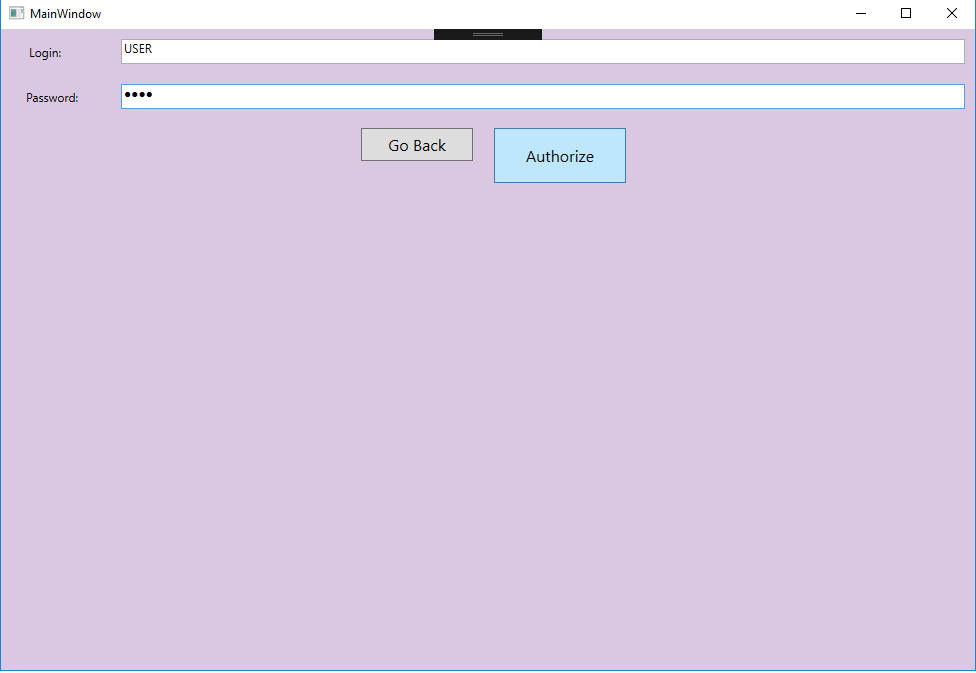
- Регистрация нового пользователя

5. **Главная страница(при запуске программы)**

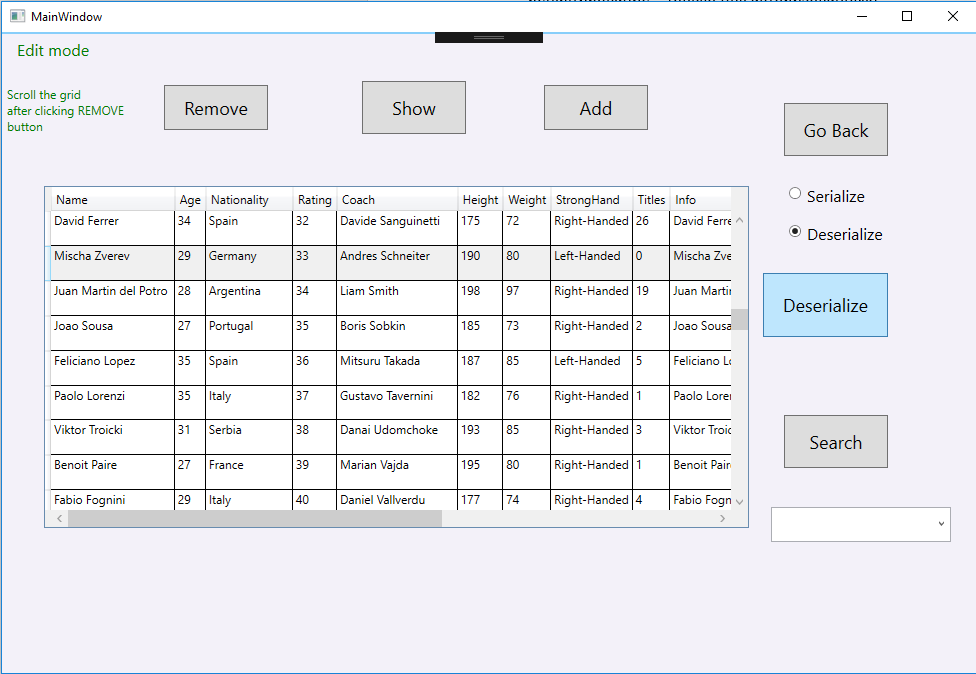


Страница регистрации.

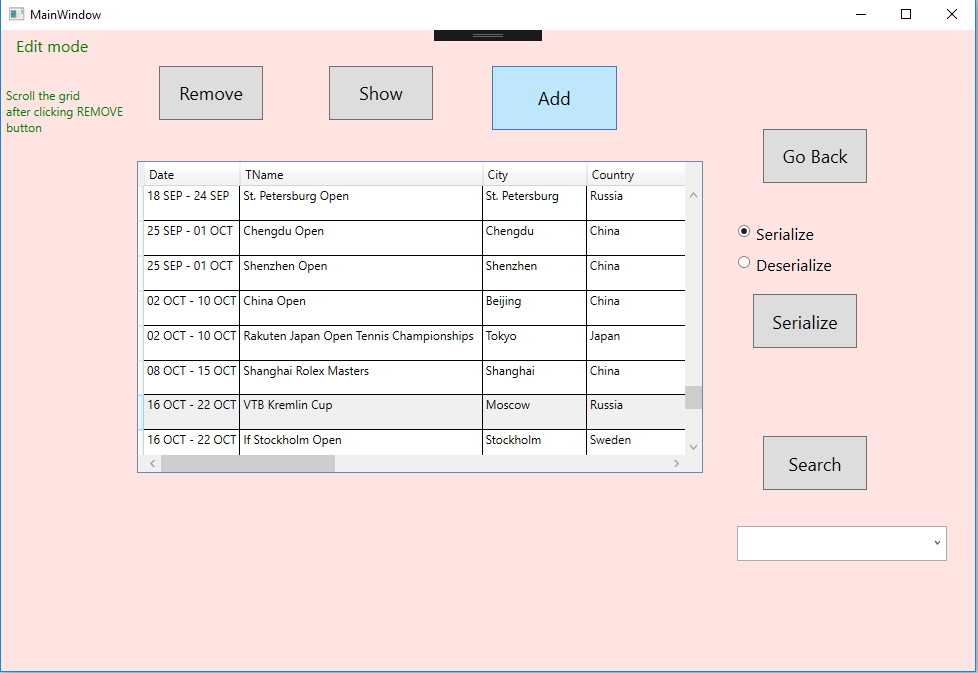
Страница авторизации



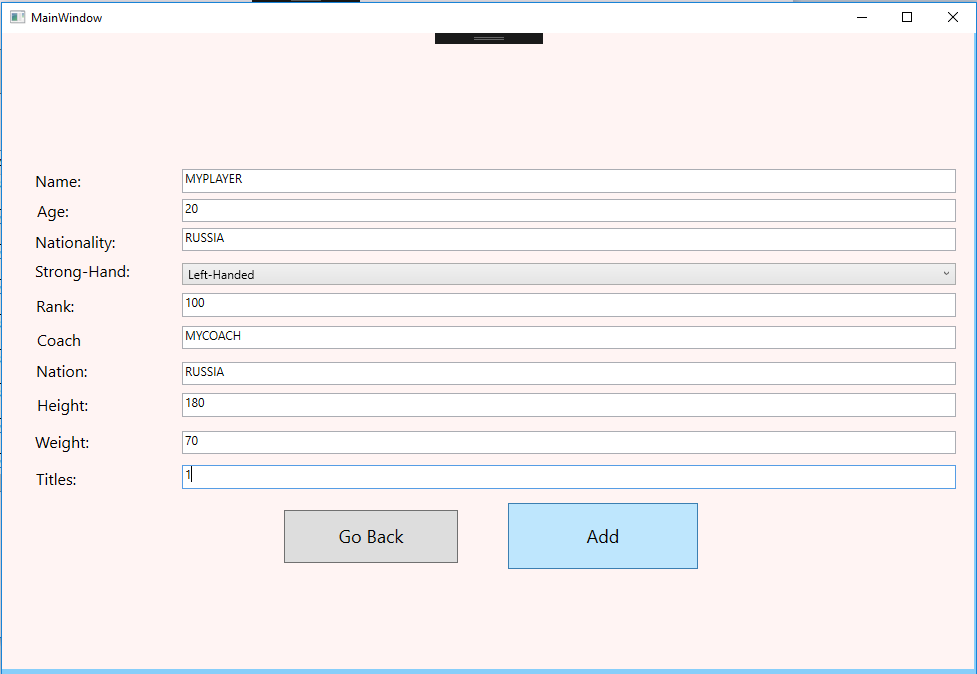
Страница PlayerPage в режиме редактирования. Выделен элемент таблицы, найденный после выполнения функции поиска.



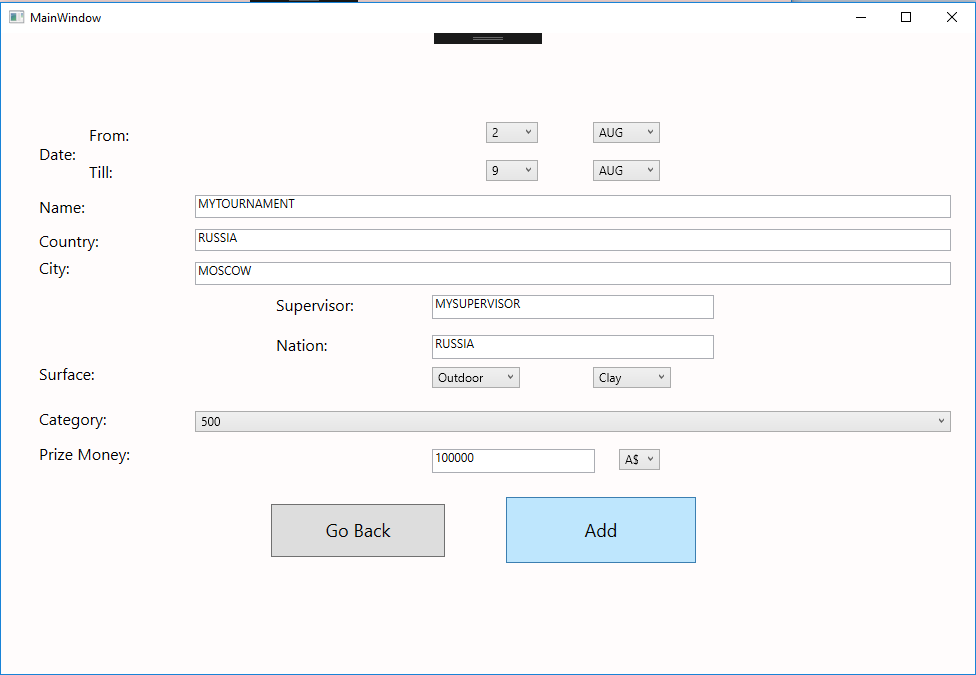
Страница TournamentPage в режиме редактирования. Выделен элемент таблицы, найденный после выполнения функции поиска.



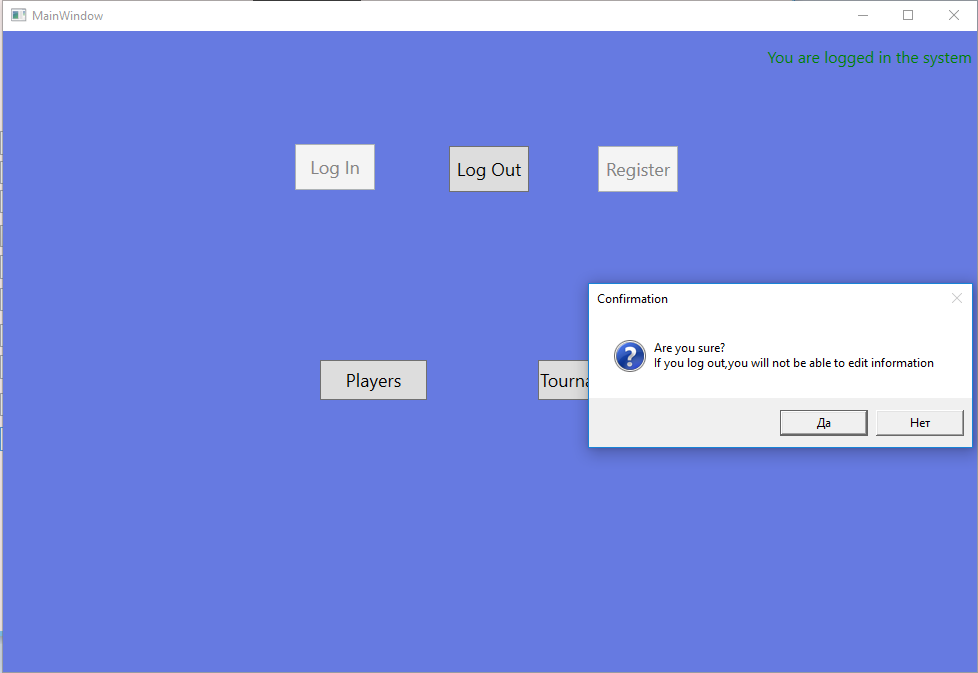
Страница добавления в список нового игрока с заполненными полями ввода информации



Страница добавления в список нового турнира с заполненными полями ввода информации



Главная страница в момент подтверждения выхода пользователя из системы.



6. **Класс Players**

public Players(string name, int age, string nationality, string strongHand, int rating, int height, int weight, int titles)

Полями класса являются параметры, определяющие игрока (имя, возраст, сильная рука, место в рейтинге, рост, вес и количество выигранных титулов).

**Класс Tournaments**

public Tournaments(string date, string tname, string city, string country, string surface, int category,string prizeMoney)

Полями класса являются параметры, определяющие турнир (название, место и дата проведения, призовой фонд, категория и покрытие кортов).

**Классы Coaches и Supervisors (для краткости здесь и выше были приведены только конструкторы классов)**

public Coaches(string name, string nation)

public Supervisors(string name, string nation)

созданы для более подробного описания экземпляров классов Players и Tournaments соответственно. Связь происходит через функцию Player.Coach и Tournament.Supervisor.

**Класс InfoSerializing**

private List<Players> \_players;

public List<Players> Players

{

get { return \_players; }

set { \_players = value; }

}

private List<Tournaments> \_tournaments;

public List<Tournaments> Tournaments

{

get { return \_tournaments; }

set { \_tournaments = value; }

}

Создан для осуществления функций сериализации и десериализации элементов классов Players и Tournaments.

**Класс RegPages**

Создан для обеспечения доступа к страницам, необходимого для осуществления функции страничной навигации **(здесь приведены только поля класса)**

private static CurrentPage \_currentPage = new CurrentPage();

private static LoginPage \_loginPage = new LoginPage();

private static AuthorizePage \_authorizePage = new AuthorizePage();

private static CompletePage \_completePage = new CompletePage();

private static MainPage \_mainPage = new MainPage();

private static PlayerPage \_playerPage = new PlayerPage();

private static NewPlayerPage \_newPlayerPage = new NewPlayerPage();

private static TournamentPage \_tournamentPage = new TournamentPage();

private static NewTournamentPage \_newTournamentPage = new NewTournamentPage();

**Класс User (для краткости приведен только конструктор)**

Создан для осуществления функции регистрации пользователей (данные о каждом зарегистрированном пользователе записываются как переменные этого класса)

public User(string name, string password)

**7. Использованные ресурсы:**

1) https://msdn.microsoft.com/ru-ru/library/system.windows.controls.datagrid(v=vs.110).aspx

2) http://www.itgo.me/a/5703987988945369319/wpf-animate-background-color

3) <https://stackoverflow.com/questions/19554073/use-animation-to-change-window-size-in-wpf>

4) https://stackoverflow.com/questions/28373643/wpf-combobox-binding-itemssource

5) <https://professorweb.ru/my/WPF/base_WPF/level1/info_WPF.php>

6) <http://www.c-sharpcorner.com/search/wpf>

7) <https://drive.google.com/drive/folders/0B3bmYcln-H9kM1hvWGFRTjdtTjg>

8) http://www.cyberforum.ru/wpf-silverlight/thread1523522.html