Федеральное агентство связи

Сибирский Государственный Университет Телекоммуникаций и Информатики

СибГУТИ

Кафедра прикладной математики и кибернетики

РГР по дисциплине Визуальное программирование и человеко-машинное взаимодействие. Ч. 1

Выполнил: студент 2 курса, группы ИП-014

Обухов Артём Игоревич  
Проверил: ст. преподаватель

Милешко Антон Владимирович

Новосибирск, 2022 г.

**РГР. Часть 1**

# Задание на РГР

Создать ПО для отображения и обработки статистических данных для определённого вида спорта. ПО должно включать 2 основных окна: окно отображающее таблицы БД со статистической информацией и результаты запросов к БД, переключение таблиц и результатов должно быть реализовано через вкладки; и окно для менеджера запросов к БД.

Первое окно должно давать возможность просматривать и изменять все таблицы БД, а также просматривать результаты запросов к БД. Должна иметься возможность удалить вкладки с результатами запросов, но не вкладки с таблицами. Также должна иметься возможность перейти к окну менеджера запросов.

Окно менеджера запросов должно предоставлять интерфейс для создания, сохранения, удаления, редактирования запросов. Созданные запросы должны отображаться в виде списка с названиями запросов, в который можно добавлять новые запросы, удалять, просматривать существующие. Для создания и редактирования запросов должен предоставляться визуальный интерфейс, а не язык запросов. Редактор запросов должен поддерживать операции выборки, соединения, группирования, подзапросы (в качестве подзапроса используются ранее сохранённые запросы).

Ход работы:

1. Исследование предметной области и создание ER диаграммы.
2. Перевод ER диаграммы в реляционную модель, создание и заполнение БД.
3. Проработка визуального интерфейса приложения
4. Создание диаграммы классов приложения
5. Реализация основного окна приложения
6. Реализация менеджера запросов
7. Тестирование и отладка

**Вариант задания - 17**

**15. Дартс (htttps://dartsdb.com, https://americandartsdatabase.com/)**

**Часть 1**

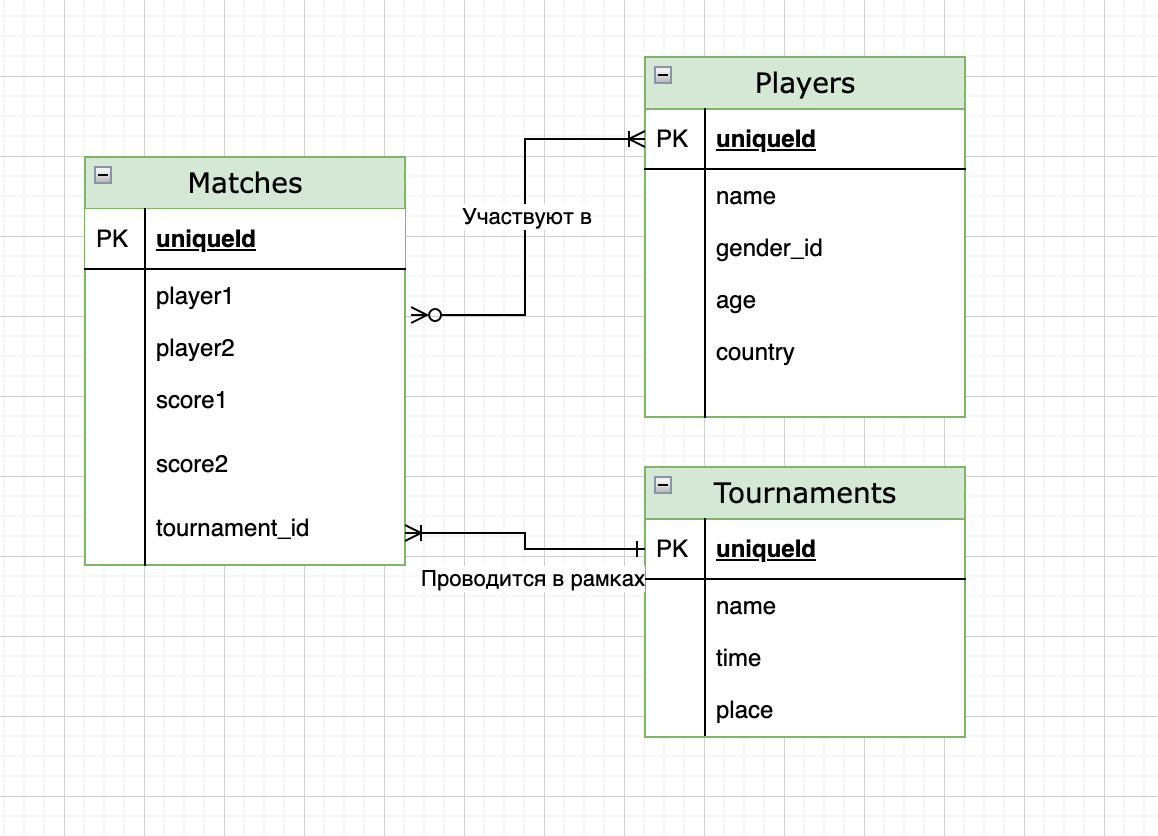
**Предметная область**

Тема: Дартс

1. Players
   1. Имя
   2. Пол
   3. Возраст
   4. Страна
2. Matches
   1. Игрок 1
   2. Игрок 2
   3. Счет первого игрока
   4. Счет второго игрока
   5. Турнир
3. Tournaments
   1. Имя
   2. Время
   3. Место

**ER-диаграмма**

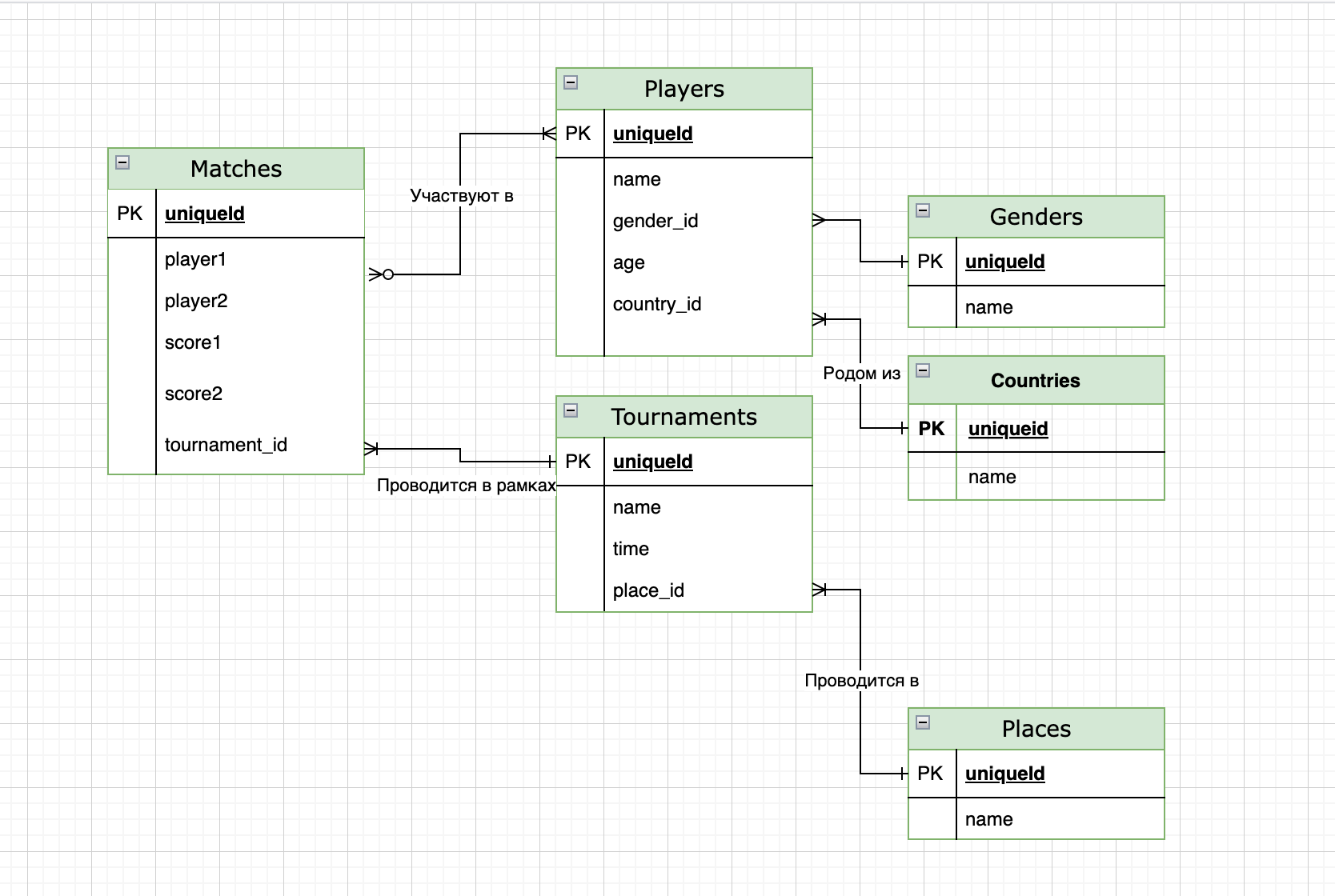
**Сущности**: Players(Игроки), Matches(Результаты игр), Tournaments(Мероприятия, в пределах которых проводятся раунды),



**Часть 2**

**Реляционная модель**

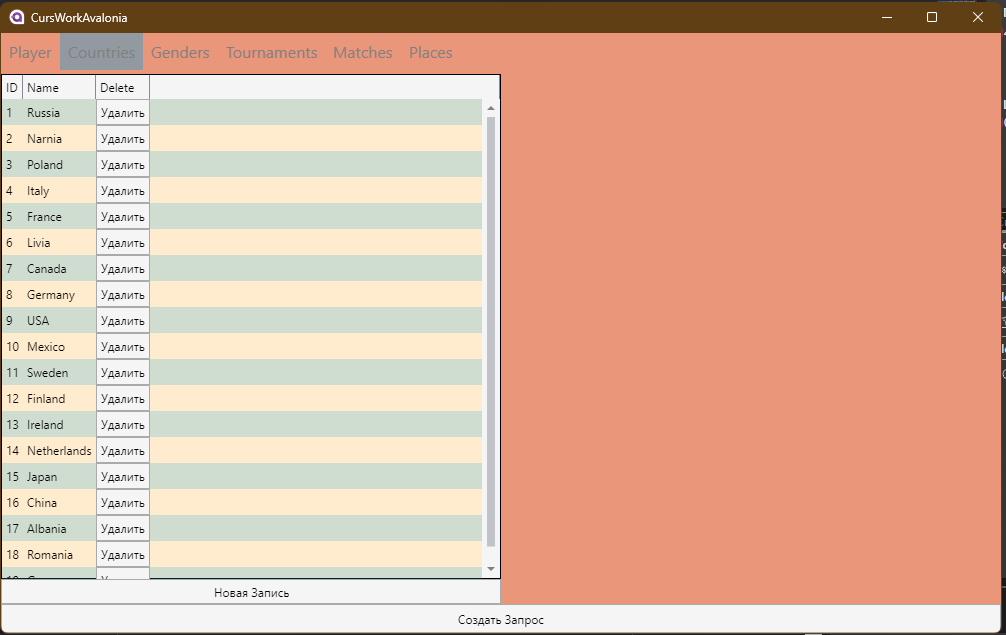
1. players
   1. id (PK)
   2. name (STRING)
   3. gender\_id (Внешний ключ)
   4. country\_id (Внешний ключ)
   5. age (INTEGER)
2. matches
   1. id (PK)
   2. player1\_id (Внешний ключ)
   3. player2\_id (Внешний ключ)
   4. score1 (INTEGER)
   5. score2 (INTEGER)
   6. tournament\_id (Внешний ключ)
3. tournaments
   1. id (PK)
   2. name (STRING)
   3. time (DATETIME)
   4. place\_id (Внешний ключ)
4. genders
   1. id (PK)
   2. name (STRING)
5. countries
   1. id (PK)
   2. name (STRING)
6. places
   1. id(PK)
   2. name (STRING)



**Часть 3**

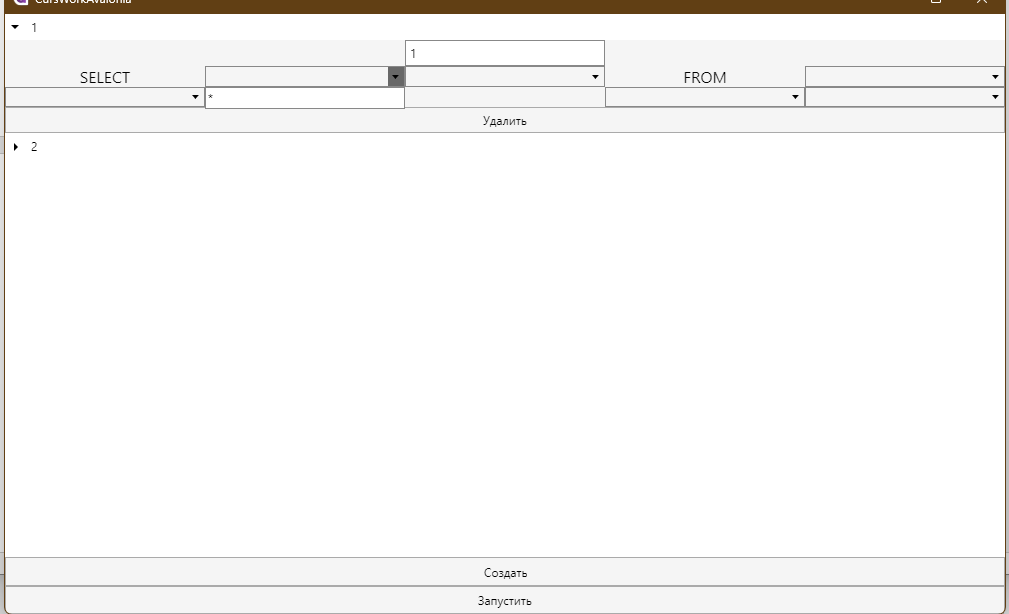
**Проработка визуального интерфейса**

**Основное окно:**



Для отображения списка таблиц использует **<TabControl>**

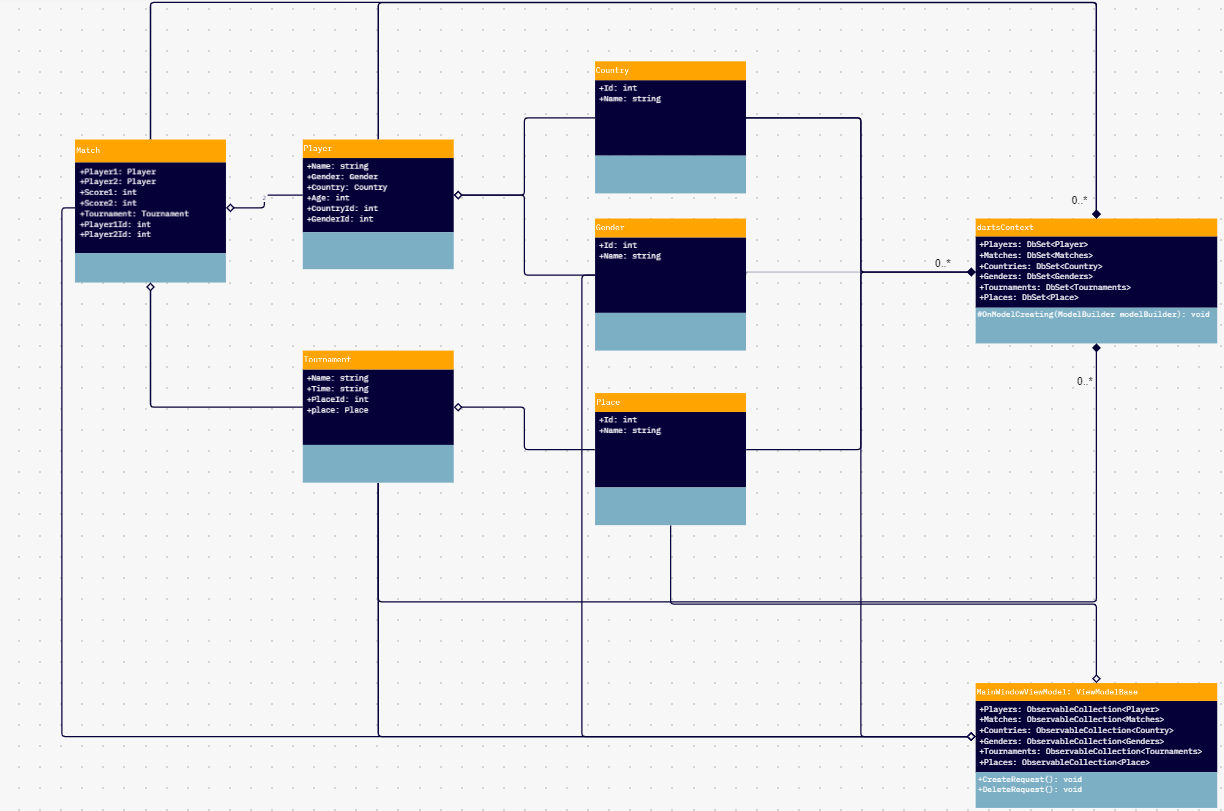
Чтобы отобразить содержимое таблиц внутри **<TabControl>** используется элемент **<DataGrid>**

**Окно запросов:**

Для выбора операторов, полей таблиц и таблиц используется **<ComboBox>** с записанными заранее значениями полей.

**Часть 4**

**Создание диаграммы классов**

****

Классы:

1. Country
   1. public string name
2. Gender
   1. public string name
3. Place
   1. public string name
4. Player
   1. public string name
   2. public Gender gender
   3. public Country country
   4. public int age
5. Match
   1. public Player player1
   2. public Player player2
   3. public int score1
   4. public int score2
   5. public Tournament tournament
6. Tournament
   1. public string name
   2. public DateTime date
   3. public Place place
7. DatabaseContext – класс, который позволяет работать с базой данных через модели. Каждый DbSet<Model> представляет собой сущности в БД
   1. public DbSet<Player> Players
   2. public DbSet<Match> Matches
   3. public DbSet<Country> Countries
   4. public DbSet<Gender> Genders
   5. public DbSet<Tournament> Tournaments
   6. public DbSet<Place> Places
   7. protected void OnModelCreating(ModelBuilder modelBuilder)
8. MainWindowViewModel – здесь представлены все основные данные и методы для работы с View, каждая ObservableCollection – наблюдаемая коллекция, содержимое которых будет отображаться в окне программы, также они непосредственно связаны с моделями из DatabaseContext
   1. public ObservableCollection<Player> Players
   2. public ObservableCollection <Match> Matches
   3. public ObservableCollection <Country> Countries
   4. public ObservableCollection <Gender> Genders
   5. public ObservableCollection <Tournament> Tournaments
   6. public ObservableCollection <Place> Places
   7. public void SQLRequestOpen()
   8. public void SQLRequestRun()

**Часть 5**

**Реализация основного окна приложения**

Было реализовано основное окно приложения, где представлены все таблицы с базы данных с возможностью удаления и добавления записей

