|  |
| --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное автономное образовательное учреждение  высшего образования |
| **ДАЛЬНЕВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ** |
|  |
|  |
|  |
| **ШКОЛА ЕСТЕСТВЕННЫХ НАУК**  **Кафедра прикладной математики, механики, управления и программного обеспечения** |
|  |
| **ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПРОГНОЗИРОВАНИЯ ИСХОДА ТЕННИСНОГО МАТЧА** |
|  |
| **ОТЧЕТ**  по дисциплине «Методы системного анализа и моделирования»  по образовательной программе подготовке бакалавров по направлению 02.03.03  «Математическое обеспечение и администрирование информационных систем»   |  |  | | --- | --- | | Защищен оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) (И.О. Фамилия)  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г. | Выполнили студенты группы Б8116 – 02.03.03  Кондаков И.А.  (подпись)  Гасанов Р.Э.  (подпись)  Руководитель зав. кафедры ПММУиПО, д.т.н., профессор И.Л. Артемьева  \_\_\_\_\_\_\_\_\_\_\_\_«\_\_\_\_» \_\_\_\_\_\_\_\_\_ 2020 г.  (подпись) | |  |  |   г. Владивосток  2020 |

**Содержание**

[**1 Анализ предметной области 3**](#_Toc41410610)

[**1.1 Анализ множества задач профессиональной деятельности, анализ объектов задач 3**](#_Toc41410611)

[**1.2 Анализ знаний предметной области 4**](#_Toc41410612)

[**1.3 Анализ ситуаций предметной области 5**](#_Toc41410613)

[**2 Построение модели предметной области 5**](#_Toc41410614)

[**2.1 Построение модели знаний 5**](#_Toc41410615)

[**2.2 Построение модели ситуаций 7**](#_Toc41410616)

[**2.3 Онтологические соглашения 10**](#_Toc41410617)

[**2.3.1 Ограничения целостности области знаний 10**](#_Toc41410618)

[**2.3.2 Связи онтологий знаний и ситуаций 10**](#_Toc41410619)

[**3 Проект интерфейса редактора базы знаний 12**](#_Toc41410620)

[**3.1 Архитектура системы 12**](#_Toc41410621)

[**3.2 Use-case диаграмма 14**](#_Toc41410622)

[**4.3 Проект базы данных для хранения знаний 14**](#_Toc41410623)

[**4.4 Проект базы данных для хранения исходных данных 15**](#_Toc41410624)

[**4 Проект интерфейса редактора знаний 15**](#_Toc41410625)

[**4.1 Определение частичного порядка на терминах области знаний 15**](#_Toc41410626)

[**4.2 Сценарий диалога со Экспертом при формировании базы знаний 16**](#_Toc41410627)

[**5 Проект интерфейса системы ввода исходных данных 18**](#_Toc41410628)

[**5.1 Определение частичного порядка на терминах области ситуаций 18**](#_Toc41410629)

[**5.2 Сценарий диалога при вводе исходных данных 18**](#_Toc41410630)

[**6 Проект решателя задач 19**](#_Toc41410631)

[**6.1 Алгоритм работы решателя задач 19**](#_Toc41410632)

[**6.2 Сценарий диалога с решателем задач 19**](#_Toc41410633)

[**7 Реализация 20**](#_Toc41410634)

[**8 Тестирование 27**](#_Toc41410635)

[**Заключение 28**](#_Toc41410636)

[**Список литературы 29**](#_Toc41410637)

# **1 Анализ предметной области**

## **1.1 Анализ множества задач профессиональной деятельности, анализ объектов задач**

В данной предметной области решается задача прогнозирования победителя в теннисном матче. Объектом задачи является теннисный матч. Под теннисным матчем понимается противостояние 2-ух теннисных игроков. Результатом решения задачи является исход игры. Исходом игры считается победа первого или второго игрока, которая определяется вероятностью победы. Под вероятностью победы понимается шанс на победу в матче первого или второго игрока.

Для того, чтобы посчитаться вероятность победы перед стартом матча анализируются значения следующих понятий:

* Процент выигранных матчей – кол-во выигранных матчей в процентном соотношении;
* Процент выигранных матчей за последние 60 недель- кол-во выигранных матчей за последние 60 недель в процентном соотношении;
* Процент выигранных BO5 матчей - кол-во выигранных матчей BO5 в процентном соотношении. “BO5” – best of 5 – матч до 3 победных сетов

(т.е победит тот кто первым одержит победу в 3 сетах)

* Рейтинг игрока – позиция игрока в рейтинге теннисистов.
* История встреч игроков – как часто и с каким результатом заканчивались прошлые матчи двух игроков.

Данные понятия являются основными признаками объекта. После анализа признаков объекта подсчитывается вероятность победы. На ее основе выводится исход игры.

Чтобы получить исход игры в текущем матче, вероятность победы подсчитывается с учетом вспомогательных понятий:

* Подачи на вылет – это заработанное очко игроком, посредством введения мяча в игру при условии, что соперник не коснулся мяча;
* Двойные ошибки – это две подряд совершенные ошибки на подаче;
* Очки выигранные на первой подаче – кол-во очков выигранных на первой подаче;
* Очки выигранные на второй подаче – кол-во очков выигранных на второй подаче;
* Очки выигранные с первой подачи соперника – кол-во очков выигранных на первой подаче соперника;
* Очки выигранные со второй подачи соперника – кол-во очков выигранных на второй подаче соперника;
* Реализованные брейкпоинты. Брейкпоинт – очко, когда игрок может выиграть гейм (временной отрезок в игре, в котором разыгрывается 4 очка; если оба игрока набрали по 3 очка, гейм проводится до 2-ух набранных подряд очков одной из сторон состязания) не на своей подаче;
* Активно выигранные мячи – кол-во “забитых” мячей. Мяч считается “забитым”, когда соперник не коснулся мяча;
* Невынужденные ошибки – кол-во ошибок совершенных игроком;
* Геймы выигранные на своей подаче;
* Геймы выигранные на приеме;

Таким образом, вероятность победы зависит от основных признаков, но если матч уже начался, то учитываются и вспомогательные понятия.

## **1.2 Анализ знаний предметной области**

Совокупность теннисных матчей является множеством матчей. Тогда под знаниями предметной области будем понимать множество матчей и их признаков.

Считается, что если теннисный матч еще не начался, то вероятность победы формируется из процента выигранных матчей, процента выигранных матчей за последние 60 недель, процента выигранных BO5 матчей, рейтинга игрока, истории встреч игроков. История встреч может быть пустой, так как первый и второй игрок могут играть впервые

## **1.3 Анализ ситуаций предметной области**

Противостояние между первым и вторым игроком считается матчем. Матч состоит из множества игроков. Под множеством игроков понимаются люди, играющие матч. Матч имеет некий исход игры. Под исходом игры понимаем победу первого или второго игрока в матче, которая определяется с помощью вероятности победы из баз знаний. Первый и второй игрок не могут быть одним и тем же человеком.

Считается, что если теннисный матч уже начался, то вероятность победы формируется не только из процента выигранных матчей, процента выигранных матчей за последние 60 недель, процента выигранных BO5 матчей, рейтинга игрока, истории встреч игроков, но и также из подачи на вылет, двойные ошибки, очки выигранные на первой подаче, очки выигранные на второй подаче, очки выигранные с первой подачи соперника, очки выигранные со второй подачи соперника, реализованные брейкпоинты, активно выигранные мячи, невынужденные ошибки, геймы выигранные на своей подаче, геймы выигранные на приеме. Новые данные будем считать вспомогательными признаками.

Таким образом, ситуацией будет являться матч, множество игроков, исход матча и вспомогательные признаки.

# **2 Построение модели предметной области**

## **2.1 Построение модели знаний**

**cорт** матчи: {} N \ Ø

термин «матч» обозначает конечное не пустое множество матчей

**сорт** игроки: {} N \ Ø

термин «игроки» обозначает конечное не пустое множество игроков

**сорт** очки игрока перед матчем: игроки -> R [0, 1000]

термин «очки игрока перед матчем» обозначает функцию, которая сопоставляет игроку некоторое вещественное число в диапазоне [0, 1000]

**сорт** очки игрока во время матча: игроки -> R [0, 1000]

термин «очки игрока во время матча» обозначает функцию, которая сопоставляет игроку некоторое вещественное число в диапазоне [0, 1000]

**сорт** рейтинг: игроки -> I [1, 200]

термин «рейтинг игрока» обозначает функцию, которая сопоставляет каждому игроку некоторое целое число в диапазоне [1, 200]

**сорт** процент выигранных матчей: игроки -> R [0, 100]

термин «процент выигранных матчей игрока» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент выигранных матчей за последние 60 недель: игроки -> R [0, 100]

термин «процент выигранных матчей игрока за последние 60 недель» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент выигранных BO5 матчей: игроки -> R [0, 100]

термин «процент выигранных BO5 матчей игроком» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** исход матча: матчи -> I [1, 2]

термин «исход матча» обозначает функцию, которая сопоставляет каждому матчу некоторое целое число в диапазоне [1, 2]

**сорт** история матчей: ∀ m ∈ матчи -> (игроки(а) х игроки(б) x I [1, 2])

термин «история матчей» обозначает функцию, которая сопоставляет матчу m из множества матчей тройку игроки, игроки, f(3, история матчей)=исход матча(m).

**сорт** предматчевая статистика: {} ([1,200] x R[0,100] x R[0,100] x R[0,100]) \ Ø

термин «предматчевая статистика» обозначает конечное не пустое множество декартовых произведений, ∀ v ∈ игроки, где f(1, предматчевая статистика)= рейтинг(v), где f(2, предматчевая статистика) = процент побед(v), где f(3, предматчевая статистика)=процент побед в BO5 матчах(v), где f(4, предматчевая статистика)=процент побед за последние 60 недель(v).

## **2.2 Построение модели ситуаций**

**сорт** процент подач на вылет: игроки -> R [0, 100]

термин «процент подач на вылет» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент двойных ошибок: игроки -> R [0, 100]

термин «процент двойных ошибок» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент очков выигранных на первой подаче: игроки -> R [0, 100]

термин «процент очков выигранных на первой подаче» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент очков выигранных на второй подаче: игроки -> R [0, 100]

термин «процент очков выигранных на второй подаче» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент очков выигранных с первой подачи соперника: игроки -> R [0, 100]

термин «процент очков выигранных с первой подачи соперника» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент очков выигранных со второй подачи соперника: игроки -> R [0, 100]

термин «процент очков выигранных со второй подачи соперника» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент реализованных брейкпоинтов: игроки -> R [0, 100]

термин «процент реализованных брейкпоинтов» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент активно выигранных мячей: игроки -> R [0, 100]

термин «процент активно выигранных мячей» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент невынужденных ошибок: игроки -> R [0, 100]

термин «процент невынужденных ошибок» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент геймов выигранные на своей подаче: игроки -> R [0, 100]

термин «процент геймов выигранные на своей подаче» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** процент геймов выигранные на приеме: игроки -> R [0, 100]

термин «процент геймов выигранные на приеме» обозначает функцию, которая сопоставляет каждому игроку некоторое вещественное число в диапазоне [0, 100]

**сорт** статистика во время матча: {} (предматчевая статистика x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100] x R [0, 100])

термин «статистика во время матча» обозначает конечное множество декартовых произведений предматчевой статистики, ∀ v ∈ игроки, где f(2, статистика во время матча) = подачи на вылет(v),где f(3, статистика во время матча) = двойные ошибки(v), где f(4, статистика во время матча) = очков выигранных на первой подаче(v), где f(5, статистика во время матча) = очков выигранных на второй подаче(v), где f(6, статистика во время матча) = очков выигранных с первой подачи соперника(v), где f(7, статистика во время матча) = очков выигранных со второй подачи соперника(v), где f(8, статистика во время матча) = реализованных брейкпоинтов(v), активно выигранных мячей, где f(9, статистика во время матча) = невынужденных ошибок(v), где f(10, статистика во время матча) =геймов выигранных на своей подаче, геймов выигранных на приеме(v).

**сорт** вероятность победы игрока перед матчем: предматчевая статистика -> R [0, 100]

термин «вероятность победы игрока перед матчем» обозначает функцию, которая сопоставляет каждой предматчевой статистике игрока некоторое вещественное число в диапазоне [0, 100].

**сорт** вероятность победы игрока во время матча: статистика во время матча -> R [0, 100]

термин «вероятность победы игрока вовремя матча» обозначает функцию, которая сопоставляет каждой статистике игрока во время матча некоторое вещественное число в диапазоне [0, 100].

## **2.3 Онтологические соглашения**

В этом разделе описаны ограничения на введенные термины онтологии прогнозирования теннисного матча.

### **2.3.1 Ограничения целостности области знаний**

1. У каждого игрока уникальное значение рейтинга.
2. Рейтинг принимает значение от 1 до 200.
3. Из пункта 1 и 2 следует, что кол-во игроков = max(рейтинг)
4. Поле игрок может принимать значения только русскими символами.

### **2.3.2 Связи онтологий знаний и ситуаций**

1. Очки игрока перед матчем вычисляется по формуле:

(M ∈ очки игрока перед матчем)

M=(100 – рейтинг) + (процент побед + процента побед в BO5 матчах + процент побед за последние 60 недель)/3 + история матчей\*100

1. Очки игрока во время матча вычисляется по формуле:

(M ∈ очки игрока во время матча)

M=(100 - рейтинг) + (процент побед + процента побед в BO5 матчах + процент побед за последние 60 недель)/3 + история матчей\*100) +

(подачи на вылет\*20 - двойные ошибки\*50 + очки выигранных на первой подаче\*20 + очки выигранные на второй подаче\*15 + очки выигранные + с первой подачи соперника\*30 + очки выигранные со второй подачи соперника\*25 + реализованные брейкпоинты \*100 + активно выигранные мячи\*10 - невынужденных ошибки\*75 + геймы выигранных на своей подаче\*50 + геймов выигранных на приеме\*100)

1. Вероятность победы игрока перед матчем вычисляется по формуле:

(V ∈ вероятность победы игрока перед матчем)

V=(очки игрока(a) перед матчем/ (очки игрока(a) перед матчем + очки игрока(b) перед матчем))

1. Вероятность победы игрока во время матча вычисляется по формуле:

(V ∈ вероятность победы игрока во время матча)

V=(очки игрока(a) во время матча/ (очки игрока(a) во время матча + очки игрока(b) во время матча))

# **3 Проект интерфейса редактора базы знаний**

## **3.1 Архитектура системы**

На рисунке 1 представлена контекстная диаграмма верхнего уровня системы.

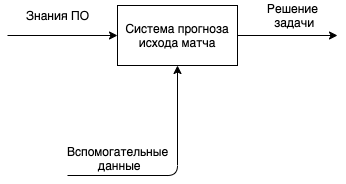
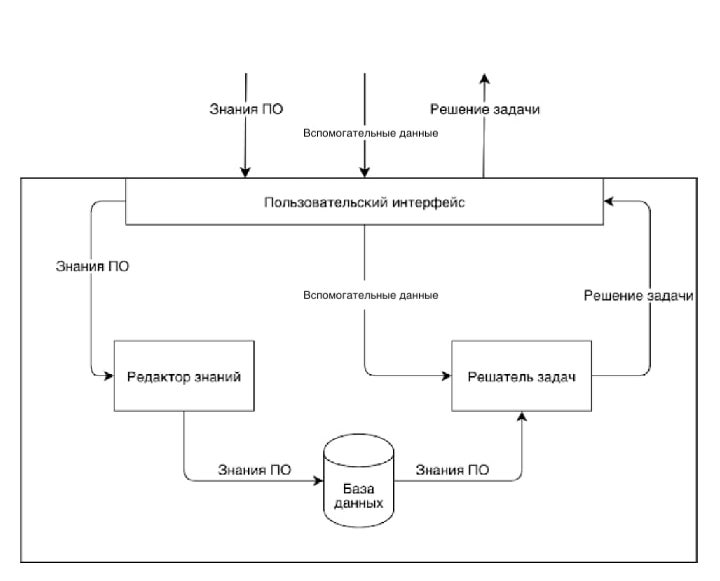


Рисунок 1 – контекстная диаграмма

На вход принемаются знания и вспомогательные данные вводимые профессионалом. Результатом работы системы является прогноз исхода матча в виде указания победителя.

На рисунке 2 представлена архитектурно-контекстная диаграмма системы



База знаний

Рисунок 2 – архитектурно-контекстная диаграмма

На диаграмме представлены основные части системы: редактор знаний, база данных, Профессионалский интерфейс, решатель задач.

Редактор знаний используется для создания и редактирования знаний. Эксперты ПО вносят свои знания о ней.

База данных предназначена для хранения знаний ПО внесенных с помощью редактора знаний.

С помощью Профессионалского интерфейса в систему вводятся знания ПО и вводимые данные профессионалом для решения задач. Здесь же входные данные проходят проверки на корректность и целостность. Так же с помощью Профессионалского интерфейса выводится результат решения задач в понятном профессионалу виде.

Решатель задач выдает результат решения задач на основе входных данных.

## **3.2 Use-case диаграмма**

На рисунке 3 представлена use-case диаграмма программы



Получение прогноза

Профессионал

Эксперт

Рисунок 3 - Use-case диаграмма системы

На диаграмме представлено два вида пользователя: Эксперт и Профессионал. Эксперт может использовать систему для создания и редактирования базы знаний. Так же Эксперт может вводить вспомогательные данные для решения.

Профессионал может выбирать игроков из множества игроков и вводить вспомогательные данные.

## **Проект базы данных для хранения знаний**

На рисунке 4 представлена схема базы знаний.

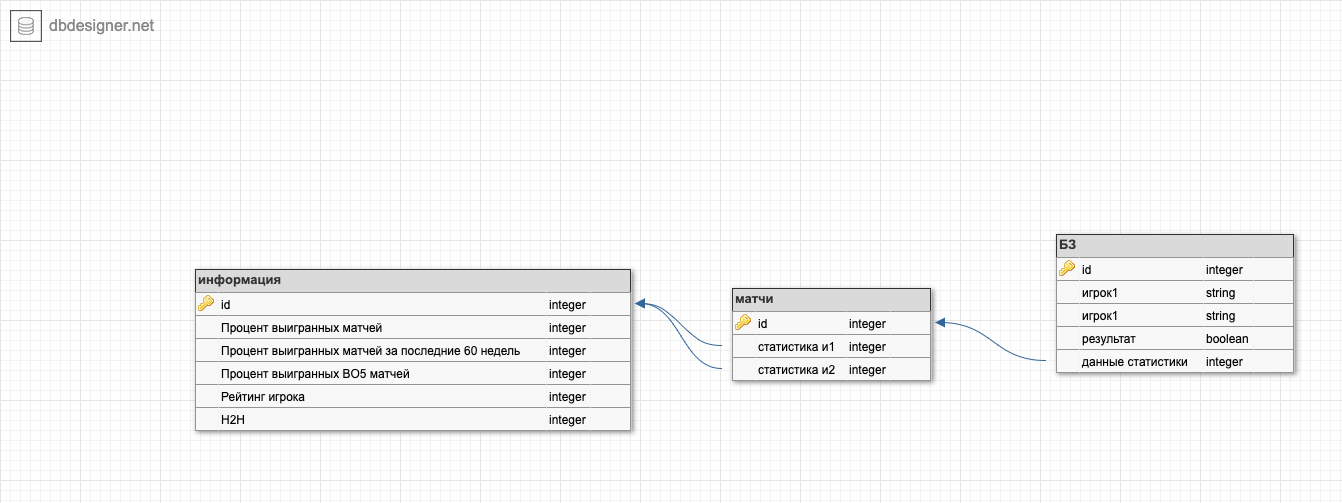


Рисунок 4 – схема базы знаний

## **Проект базы данных для хранения исходных данных**

На рисунке 5 представлена съема базы дынных для хранения исходных данных

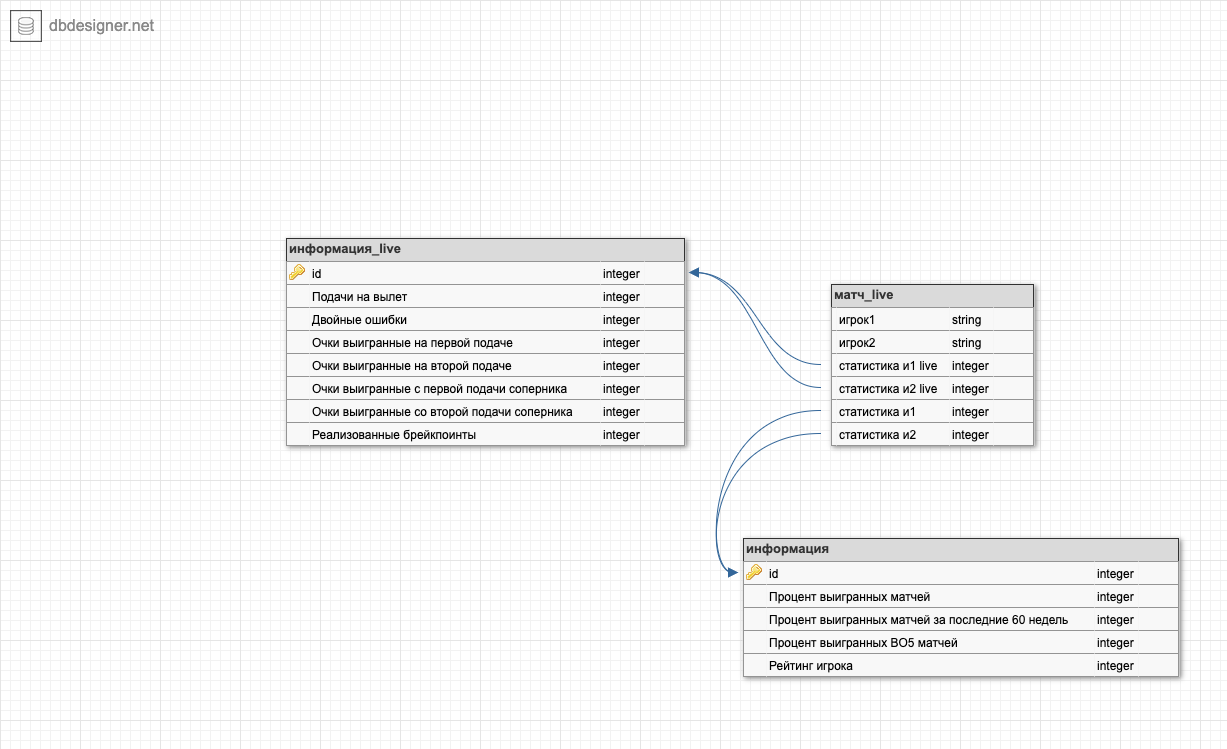


Рисунок 5 – схема базы знаний исходных данных

# **4 Проект интерфейса редактора знаний**

Для Эксперта данной предметной области требуется спроектировать систему редактора базы знаний, который бы согласовывался с моделью предметной области.

* 1. **Определение частичного порядка на терминах области знаний**

На терминах из области знаний вводится структура частичного порядка по отношению зависимости. Эту структуру можно описать графом, приведенным на рисунке 6.

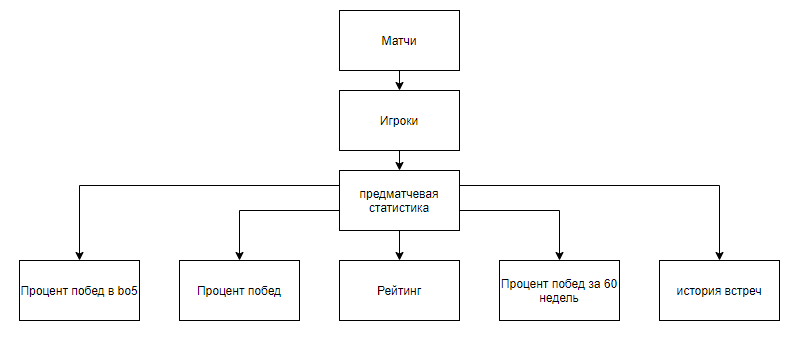


Рисунок 6 – граф зависимостей терминов области знаний

**4.2 Сценарий диалога со Экспертом при формировании базы знаний**

Пример окна редактора знаний для добавления матча показан на рисунке 7.

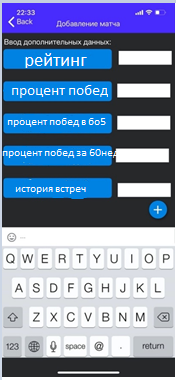


Рисунок 7 - окно редактора знаний для добавления матча

Возможный сценарий диалога состоит из ввода данных статистики

Пример окна редактора знаний для просмотра матчей показан на рисунке 8.

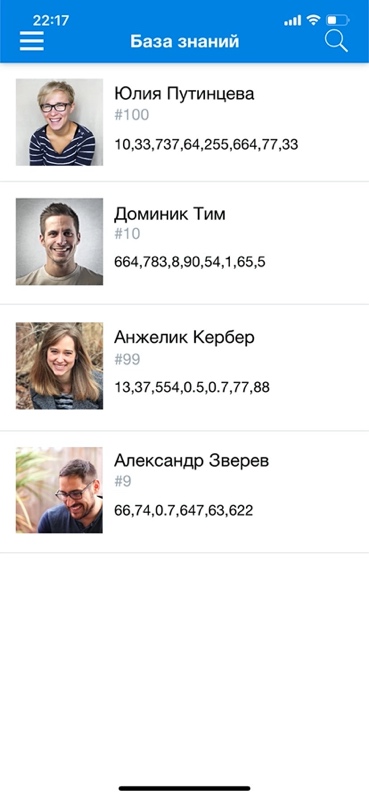


Рисунок 8 - окно редактора знаний для просмотра матчей

Возможный сценарий диалога состоит из ввода данных статистики для каждого игрока.

Примерный вид окна выбора режима программы представлен на рисунке 9.

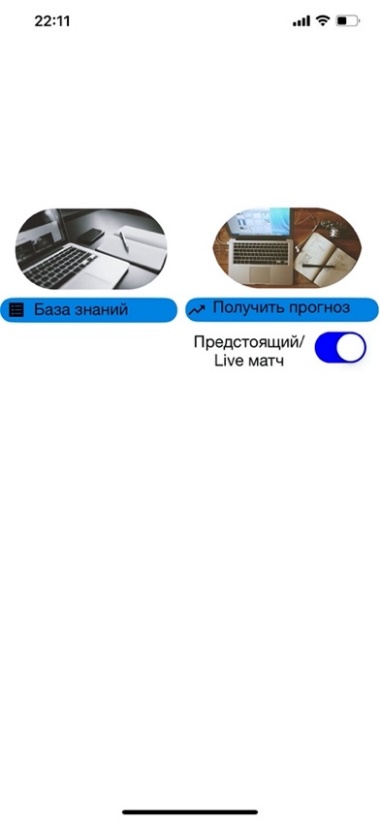


Рисунок 9 - Окно выбора режима

**5 Проект интерфейса системы ввода исходных данных**

**5.1 Определение частичного порядка на терминах области ситуаций**

На терминах из области ситуаций вводится структура частичного порядка по отношению зависимости. Эту структуру можно описать графом, приведенным на рисунке 10.

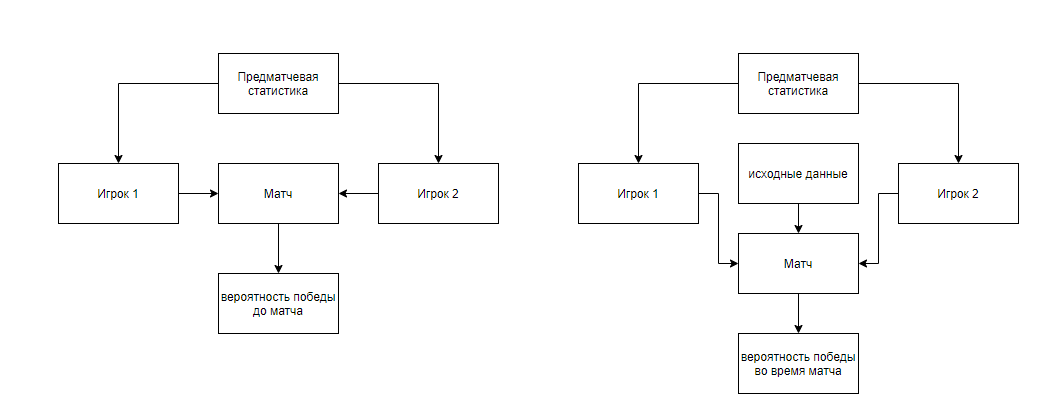


Рисунок 10 – граф зависимостей терминов области ситуаций

**5.2 Сценарий диалога при вводе исходных данных**

Примерный вид окна ввода исходных данных представлен на рисунке 11

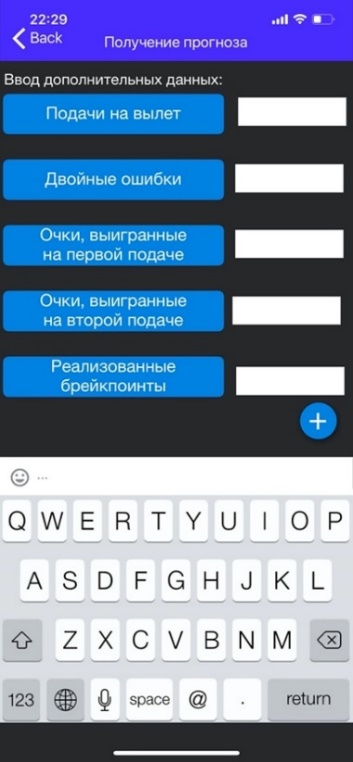


Рисунок 11 - Окно ввода исходных данных

Возможный сценарий диалога: ввод исходных данных

Текст объяснения, формируемого системой, основанной на знаниях, при решении задачи выглядит так:

Прогноз матча: “Имя игрока” : “Вероятность победы”

**6 Проект решателя задач**

В этом разделе описан решатель задач, выполняющий функцию предсказания исхода матча между двумя теннисными игроками, используя знания из предметной области и исходные данные, введенные профессионалом.

Задача предсказания решается формализованным методом, так как ответом является значение полученное входе математических расчетов и сопутствующих данных.

**6.1 Алгоритм работы решателя задач**

1. Для каждого игрока в базе знаний хранится его статистика;
2. Два игрока из множества игроков формируют матч;
3. Если матчи еще не начались, то для каждого такого матча, на основе знаний об игроках решатель задач выводит прогноз на исход матча, на основе вероятности победы того или иного игрока в матче;
4. Если матчи уже идут, то профессионалу предлагается заполнить исходные данные для выбранного им матча;
5. Решатель задач на основе предматчевой статистики и полученных новых исходных данных, формирует прогноз в виде исхода матча, на основе вероятности победы того или иного игрока.

**6.2 Сценарий диалога с решателем задач**

Решатель задач, представленный на рисунке 12 позволяет спрогнозировать исход матча с учетом знаний и исходных данных введенные профессионалом. После заполнения полей, решатель задач обрабатывает знания и исходные данные согласно алгоритму, описанному в пункте 6.1.

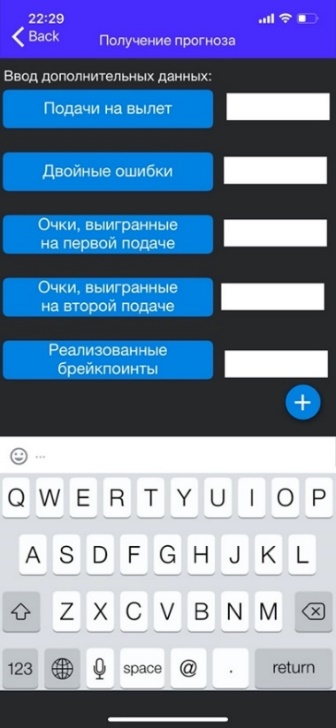


Рисунок 12 - Окно ввода исходных данных

**7 Реализация**

Для решения задачи по реализации редактора баз знаний было решено использовать стек технологий, необходимый для написания мобильного приложения. Был изучен язык программирования Swift и программа для программирования на Swift – Xcode. Xcode – интегрирования среда разработки (IDE) программного обеспечения для платформ macOS, iOS, watchOS и tvOS, разработанная компанией Apple. Первая версия выпущена в 2001 году. Стабильные версии распространяются бесплатно через Mac App Store.

Xcode включается в себя большую часть документации разработчика от Apple и Interface Builder – приложение, использующееся для создания графических интерфейсов. Пакет Xcode включается в себя измененную версию свободного набора компиляторов GNU Compiler Collection и поддерживает языки C, C++, Objective-C, Objective-C++, Swift, Java, Python, AppleScript и Ruby с различными моделями программирования.

Исходя из предметной области и специфики задачи, программное обеспечение было реализовано следующим образом:

1. При запуске программы на телефоне нас встречает главное меню “Матчи” со списком матчей внесенный в базу знаний, изображенный на рисунке 13;

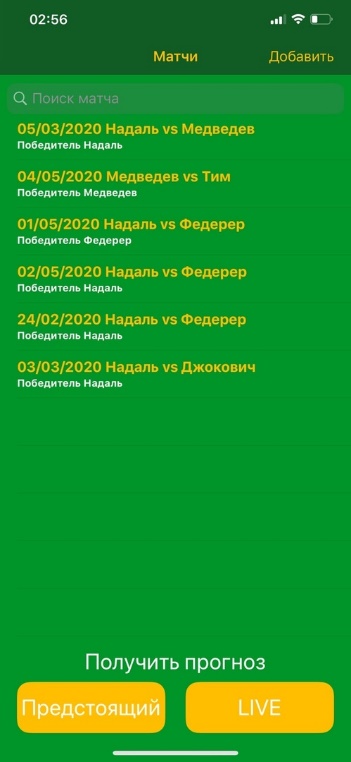


Рисунок 13 – окно программы: “Матчи”

1. В окне “Матчи” реализован поиск по базе знаний, продемонстрированный на рисунке 14;

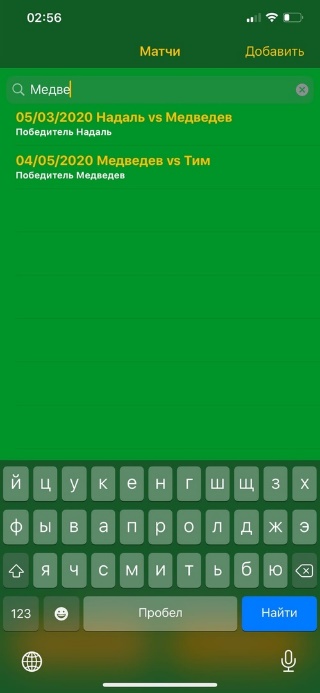


Рисунок 14 – поиск по матчам

1. По длительному нажатию на матч, из списка матчей можно перейти к редактированию выбранной записи в базе знаний. Так на рисунке 15 мы можем увидеть, что в выбранном матче, который уже был занесен в базу знаний до этого, есть возможность редактирования полей, помимо этого тут можно увидеть, что реализованы все описанные ограничения, исходя из онтологии предметной области, так поле очки выигранные на ПП выдало ошибку;



Рисунок 15 – изменение данных о матче

1. После заполнения всех редактируемых полей для сохранения внесенных данных, необходимо нажать на кнопку “изменить матч”;



Рисунок 16 – кнопка “изменить матч”

1. Как можно увидеть на рисунке 13 и 14, справа сверху имеется надпись “Добавить”, которая перенаправляет нас на страницу добавления нового матча, данная страница продемонстрирована на рисунке 17, на данной странице видно, что описанные ранее ограничения работают и мы не сможем добавить матч пока не заполним полям соответствующим образом, матч добавляется по нажатию кнопки “добавить матч”;



Рисунок 17 – добавление матча

1. Теперь перейдем к решателю задач. Целью являлось чтобы данная программа, по мимо редактирования базы знаний выводила прогноз по предстоящему матчу и матчу, который уже начался. Как можно было увидеть на рисунке 13 снизу экрана расположены кнопки “Предстоящий” и “Live”. На рисунке 18 продемонстрировано окно получения прогноза по предстоящему матчу в случая ввода верных данных;



Рисунок 18 – получения прогноза по предстоящему матчу

1. Если введенные нами данные по предстоящему матчу неверные, то мы можем наблюдать следующие сообщения от программы, показанные на рисунках 19 и 20;

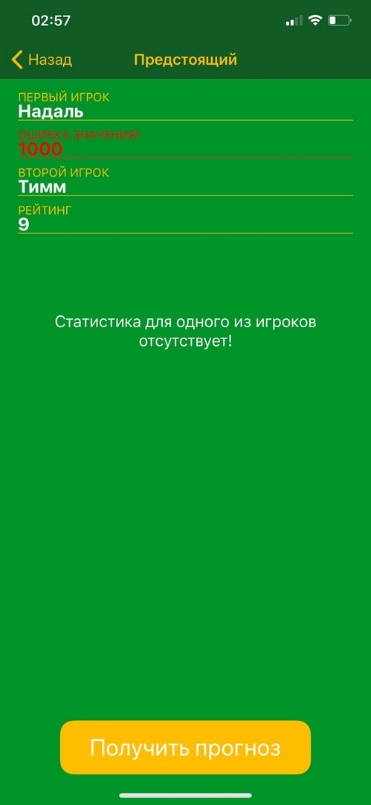


Рисунок 19 – пример ошибки предстояшего матча

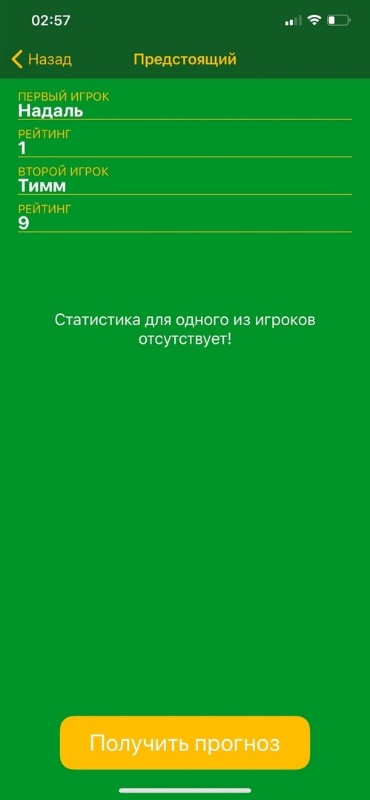


Рисунок 20 – пример ошибки предстояшего матча

1. Теперь посмотрим, что происходит на страничке с текущим матчем, к которой мы можем перейти по кнопке “Live”, показанной ранее. Здесь на рисунке 21 мы видим, что после правильного заполнения всех полей мы также получаем необходимый нам прогноз по матчу.



Рисунок 21 – получение прогноза текущего матча

1. Так же как и в случае с предстоящим матчем мы не сможем получить прогноз, если данные будут введены некорректно или во все будут пустые, данные ошибки продемонстрированы на рисунках ниже.



Рисунок 22 – ошибка в получении прогноза в текущем матче



Рисунок 23 – ошибка в получении прогноза в текущем матче

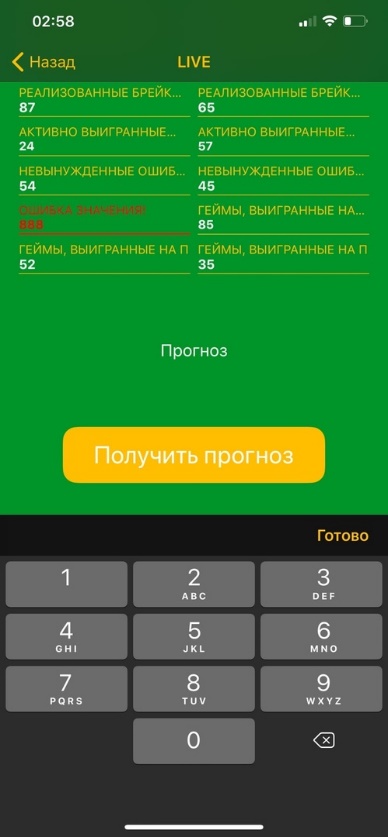


Рисунок 24 – ошибка в получении прогноза в текущем матче

8 Тестирование

8.1 Тестирование добавления, редактирования и удаления данных

1) Ошибка ввода/редактирования данных, на вход подаем значения параметров (рисунок 25).



Рисунок 25 - Выходные данные

2) Ввод/редактирование данных, на вход подаем корректные полностью заполненные данные (рисунки 26.1, 26.2).



Рисунок 26.1 – Входные данные

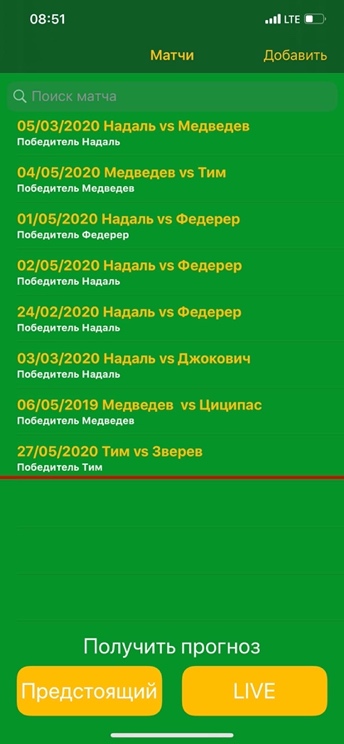


Рисунок 26.2 – Выходные данные

3) Ввод/редактирование данных, на вход подаются пустые данные (рисунки 27.1, 27.2).



Рисунок 27.1 – Входные данные



Рисунок 27.2 – Выходные данные

4) Удаление данных (рисунки 28.1, 28.2).

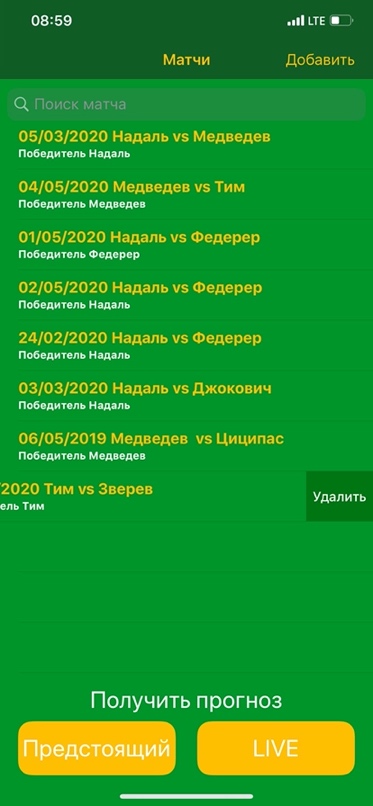


Рисунок 28.1 – Входные данные

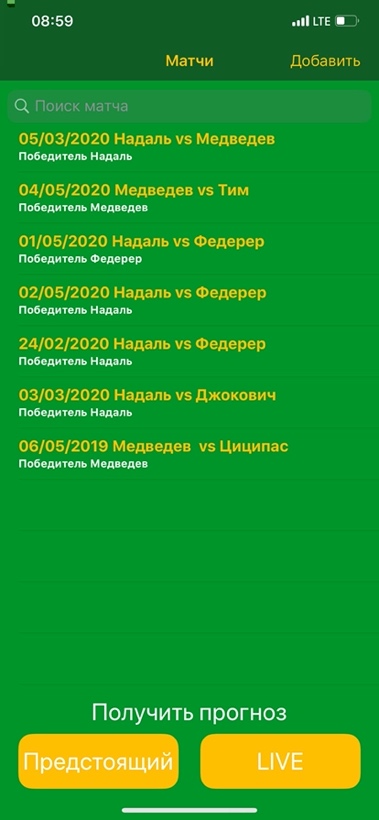


Рисунок 28.2 – Выходные данные

8.2 Тестирование решения задачи прогнозирования теннисного матча

1) Прогноз предстоящего матча с существующими игроками и значениями рейтинга в установленных пределах (рисунок 29).

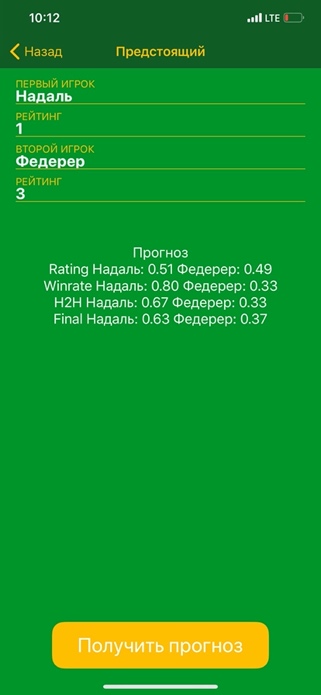


Рисунок 29 – Выходные данные

2) Прогноз LIVE матча с существующими игроками и значениями статистики в установленных пределах (рисунок 30).



Рисунок 30 – Выходные данные

3) Прогноз предстоящего матча с несуществующими игроками (рисунок 31).

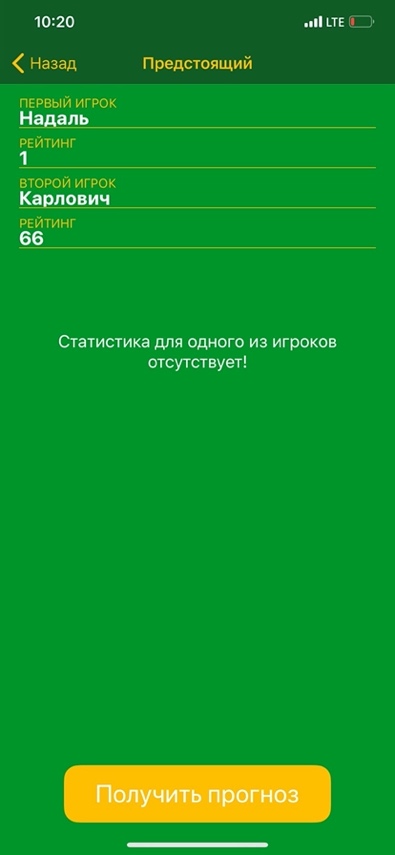


Рисунок 30 – Выходные данные

**Заключение**

В рамках курсового проекта был разработан проект системы, основанной на знаниях, для чего были решены поставленные задачи:

1. проанализирована предметная область;
2. разработана модель предметной области;
3. разработан проект системы для данной предметной области;
4. разработан проект интерфейса системы ввода исходных данных;
5. разработан редактор баз знаний по описанной предметной области.

Таким образом, цель данной лабораторной работы была достигнута.

**Список литературы**

1. Артемьева И. Л. Лекции по дисциплине “Методы системного анализа и моделирования”. Электронный вариант;
2. Документация по языку программирования SWIFT: <https://swift.org/documentation/>