

# **ПРИЛОЖЕНИЕ А**

**(Обязательное)**

**Техническое задание**

**Листов 8**

**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

---

УТВЕРЖДАЮ

Заведующий кафедрой ИУ6

А.В. Пролетарский  
« \_\_\_\_\_ » \_\_\_\_\_ 2020 г.

**СИСТЕМА ПОДДЕРЖКИ ОРГАНИЗАЦИИ ФУТБОЛЬНОГО ТУРНИРА**

Техническое задание

Листов 8

Студент

ИУ6-85Б  
(Группа)



(Подпись, дата)

Э.А. Степанов  
(И.О. Фамилия)

Руководитель

(Подпись, дата)

А.А. Малахов  
(И.О. Фамилия)

2020 г.

## 1 ВВЕДЕНИЕ

Настоящее техническое задание распространяется на разработку системы поддержки организации футбольного турнира (СПОФТ), используемой для помощи организатору в проведении спортивного мероприятия, чтобы минимизировать организационные издержки, возникающие в связи с человеческим фактором связанным различными событиями и предназначенной для любого пользователя, имеющего права доступа к системе.

Актуальность разработки СПОФТ обусловлена стремительным развитием технологий и возможностью улучшить существующую систему организации футбольного турнира. Система предназначена для людей, желающих получить на основе входящих и исходящих данных наиболее актуальные данные на данный момент времени.

## 2 ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

СПОФТ разрабатывается в соответствии с учебным планом кафедры ИУ6 «Компьютерные системы и сети» факультета «Информатика и системы управления» МГТУ им. Н.Э. Баумана, утвержденным в установленном порядке.

## 3 НАЗНАЧЕНИЕ РАЗРАБОТКИ

### 3.1 Функциональное назначение программного продукта

Назначением программного продукта является получение пользователем полностью удачного сгенерированного списка футбольных матчей. Основой для обработки являются данные, которые вводятся пользователями при регистрации.

### 3.2 Эксплуатационное назначение программного продукта

Пользователями СПОФТ могут являться любые лица, имеющие подходящее информационно-коммуникационное устройство.

## 4 ИСХОДНЫЕ ДАННЫЕ, ЦЕЛИ И ЗАДАЧИ

### 4.1. Исходные данные

4.1.1. Исходными данными для разработки являются следующие материалы:

4.1.1.1. Введение в многомерный анализ [Электронный ресурс]. URL: <https://habr.com/ru/post/126810/>

4.1.1.2 Принципы построения систем, ориентированных на анализ данных [Электронный ресурс] URL: <https://ami.nstu.ru/~vms/lecture/lecture9/LECTURE9.HTM>

4.1.1.3 Введение в OLAP и многомерные базы данных [Электронный ресурс] URL: <https://www.cfin.ru/itm/olap/intro.shtml>

4.1.1.4 Подключение БД и создание консольного Java приложения [Электронный ресурс] URL: <https://javarush.ru/groups/posts/1952-vvedenie-v-sql>

4.1.1.5 Введение в ORM [Электронный ресурс] URL: <http://internetka.in.ua/orm-intro/>

4.1.1.6 Основы Hibernate [Электронный ресурс] UR: <https://habr.com/ru/post/29694/>

### 4.2. Цель работы

Целью работы является система СПОФТ для наиболее быстрой обработки и анализа входящих данных от пользователей, в организации футбольного турнира.

### 4.3. Решаемые задачи

4.3.1 Анализ существующих систем для организации футбольного турнира.

4.3.2 Сравнительный анализ систем обработки данных.

4.3.3 Проектирование структурной схемы и разработка функциональных диаграмм.

4.3.4 Проектирование серверной части, а также основного алгоритма системы.

4.3.5 Разработка интерфейса.

4.3.6 Разработка технологий тестирования и тестирование системы.

## 5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ СИСТЕМЕ

### 5.1. Требования к функциональным характеристикам

#### 5.1.1. Выполняемые функции

##### 5.1.1.1. Для Участника (Менеджера)

- Регистрация;
- Авторизация;
- Ввод данных о команде.
- Просмотр списка матчей;
- Просмотр результатов матчей и статистики игроков;

##### 5.1.1.2 Для организатора

- Авторизация;
- Доступ к данным о командах;
- Изменение данных о команде;
- Просмотр и изменения списка матчей;
- Просмотр результатов матчей и статистики игроков;
- Диалог с участником (менеджером).

##### 5.1.1.3 Для зрителя

- Регистрация;
- Авторизация;
- Просмотр списка матчей;
- Просмотр результатов матчей и статистики игроков;
- Покупка билетов на матч.

#### 5.1.2. Исходные данные:

- Вводимые первоначальные данные о команде
- Информация о местоположения команды;

- Информация о стадионах;
- Информация о месте и времени предстоящего матча.

#### 5.1.3. Результаты:

- Список предстоящих матчей;
- Список игроков, участвующих в турнире;
- Результаты матчей и статистика игроков.

### 5.2. Требования к надежности

5.2.1. Предусмотреть контроль вводимой информации.

5.2.2. Предусмотреть защиту от некорректных действий пользователя.

5.2.3. Обеспечить целостность информации в базе данных.

### 5.3. Условия эксплуатации

5.3.1. Условия эксплуатации в соответствии с СанПиН 2.2.2/2.4.1340-03.

### 5.4. Требования к составу и параметрам технических средств

5.4.1. Программное обеспечение должно функционировать на IBM-совместимых персональных компьютерах.

5.4.2. Минимальная конфигурация технических средств:

5.4.2.1 Тип процессора .....Intel Core i3, AMD Athlon.

5.4.2.2 Объем ОЗУ .....2048 Мб.

5.4.2.3 Подключение к сети Интернет.

### 5.5. Требования к информационной и программной совместимости

5.5.1. Программное обеспечение персонального компьютера должно работать под управлением операционных систем Windows 7, Windows Vista, Windows XP, Windows 8, Windows 10

## 5.6. Требования к маркировке и упаковке

Требования к маркировке и упаковке не предъявляются.

## 5.7. Требования к транспортированию и хранению

Требования к транспортировке и хранению не предъявляются.

## 5.8. Специальные требования

Специальные требования не предъявляются.

# 6. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

6.1. Разрабатываемые программные модули должны быть самодокументированы, т.е. тексты программ должны содержать все необходимые комментарии.

6.2. В состав сопровождающей документации должны входить:

6.2.1. Расчетно-пояснительная записка на 50-80 листах формата А4 (без приложений 6.2.2, 6.2.3 и 6.2.4).

6.2.2. Техническое задание (Приложение А).

6.2.3. Руководство пользователя (Приложение Б).

6.2.4. Фрагменты исходных кодов программы (Приложение В).

6.3. Графическая часть должна быть выполнена на 6 листах формата А1 (копии формата А3, А4 включить в качестве приложений к расчетно-пояснительной записке):

6.3.1. Результаты анализа существующих систем

6.3.2. Структурная схема информационной системы

6.3.3. Функциональная модель IDEF0

6.3.4. Схема структуры базы данных

6.3.5. Диаграмма вариантов использования системы

6.3.6. Результаты тестирования системы

6.3.7. Формы интерфейса

## 7. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

Требования к технико-экономическому обоснованию разработки не предъявляются.

## 8. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

№	Название этапа	Срок, даты, %	Отчетность
1.	Разработка технического задания	2.02.2019 - 20.02.2020 5 %	Утвержденное техническое задание.
2.	Анализ требований и уточнение спецификаций	21.02.2020 - 07.03.2020 15%	Спецификации программного обеспечения.
3.	Проектирование структуры программного обеспечения, проектирование компонентов	08.03.2020 - 28.03.2020 25%	Схема структурная информационной системы, диаграмма вариантов использования, графы абстрактных диалогов, диаграмма классов концептуального уровня, диаграммы классов приложения уровня реализации.



№	Название этапа	Срок, даты, %	Отчетность
4.	Реализация компонентов и автономное тестирование компонентов.  Сборка и комплексное тестирование.	29.03.2020 - 18.04.2020  50%	Тексты программ- ных компонентов.  Таблицы тестов, ре- зультаты тестирова- ния.
5.	Разработка документации.	19.04.2020 - 25.05.2020  75%	Расчетно-поясни- тельная записка.
6.	Прохождение нормоконтроля, проверка на антиплагиат, получение рецензии, подготовка доклада и предзащита.	25.05.2020 - 6.06.2020  90 %	Иллюстративный материал, доклад, рецензия, справки о нормоконтроле и проценте плагиата.
7.	Защита выпускной квалификационной работы.	8.06.2020 - 04.07.2020  100 %	Доклад и представ- ление выпускной квалификационной работы на заседании государственной эк- заменационной ко- миссии (ГЭК).

## 9. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

### 9.1 Порядок контроля

Контроль выполнения осуществляется руководителем еженедельно.

### 9.2 Порядок защиты

Защита осуществляется перед государственной экзаменационной комиссией (ГЭК).

### 9.3 Срок защиты

Срок защиты определяется в соответствии с планом заседаний ГЭК.

## 10. ПРИМЕЧАНИЕ

В процессе выполнения работы возможно уточнение отдельных требований технического задания по взаимному согласованию руководителя и исполнителя.

## **ПРИЛОЖЕНИЕ Б**

(Обязательное)

Руководство пользователя

Листов 7

**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

---

**ФАКУЛЬТЕТ Информатика и системы управления  
КАФЕДРА Компьютерные системы и сети**

**СИСТЕМА ПОДДЕРЖКИ ОРГАНИЗАЦИИ ФУТБОЛЬНОГО ТУРНИРА**

Руководство пользователя

Листов 7

Студент

ИУ6-85Б  
(Группа)



\_\_\_\_\_  
(Подпись, дата)

Э. А. Степанов  
(И.О. Фамилия)

Руководитель

\_\_\_\_\_  
(Подпись, дата)

А. А. Малахов  
(И.О. Фамилия)

2020 г.

A screenshot of a web application's login page. The page has a dark blue background. At the top, there is a header with a colorful geometric pattern on the left and standard window control buttons (minimize, maximize, close) on the right. The main heading is 'Вход' (Login) in white. Below it are two input fields: 'Логин' (Login) and 'Пароль' (Password), both with light blue text and underlined. Below the password field are two red buttons: 'Войти' (Login) and 'Регистрация' (Registration).

Рисунок Б.1 – Вход в систему

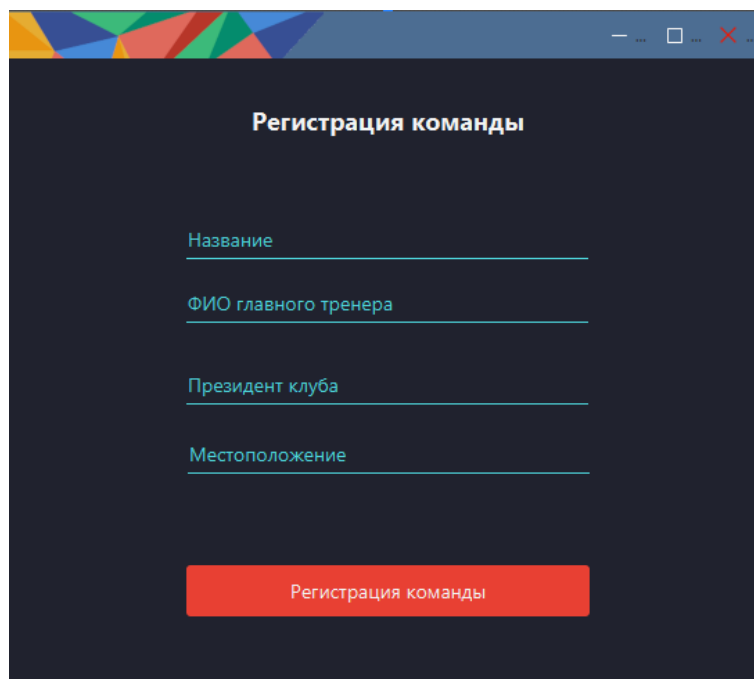
1.Для входа в систему необходимо ввести логин и пароль и нажать на кнопку «Войти».

2.Если пользователь не зарегистрирован в системе, то необходимо пройти процесс регистрации нажать на кнопку «Регистрация».

A screenshot of a web application's registration page. The page has a dark blue background. The main heading is 'Создание нового менеджера' (Create new manager) in white. Below it are three input fields: 'ФИО' (Full Name), 'Логин' (Login), and 'Пароль' (Password), all with light blue text and underlined. Below the password field is a red button labeled 'Сохранить' (Save).

Рисунок Б.2 – Создание нового менеджера

Чтобы создать нового менеджера пользователю необходимо заполнить необходимые поля и нажать на кнопку «Сохранить».

A screenshot of a web application window titled "Регистрация команды" (Team Registration). The window has a dark blue background and a colorful geometric header. It contains four text input fields stacked vertically: "Название" (Name), "ФИО главного тренера" (Coach's Full Name), "Президент клуба" (Club President), and "Местоположение" (Location). At the bottom, there is a red button labeled "Регистрация команды" (Register Team).

Регистрация команды

Название

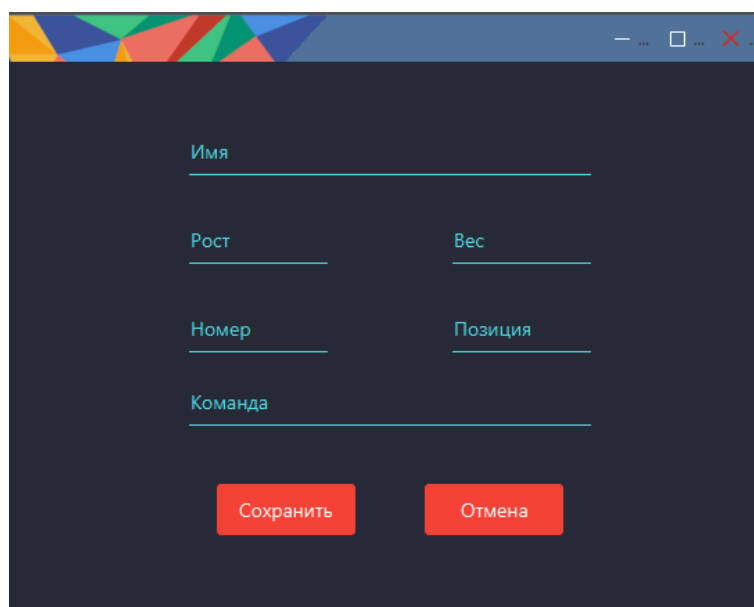
ФИО главного тренера

Президент клуба

Местоположение

Регистрация команды

Рисунок Б.3 – Регистрация команды

A screenshot of a web application window titled "Добавление игрока в команду" (Add Player to Team). The window has a dark blue background and a colorful geometric header. It contains five text input fields: "Имя" (Name), "Рост" (Height), "Вес" (Weight), "Номер" (Number), and "Позиция" (Position). Below these is a field labeled "Команда" (Team). At the bottom, there are two red buttons: "Сохранить" (Save) and "Отмена" (Cancel).

Имя

Рост

Вес

Номер

Позиция

Команда

Сохранить

Отмена

Рисунок Б.4 – Добавление игрока в команду

Менеджер команды, после входа в систему, должен зарегистрировать команду и добавить игроков. Для регистрации команды, пользователю необходимо заполнить все необходимые поля и нажать на кнопку «Регистрация команды». Для добавления игроков в команду, ему необходимо заполнить все поля, кроме «Команда» - это поле заполняется автоматически, так как за одним менеджером команды закреплён один клуб.

Название	Год основания	Тренер	Лига	Рейтинг
ЦСКА	1977	Гончаренко.В.М	РПЛ	32
Реал Мадрид	1902	Зидан	Ла Лига	1
Бавария	1900	Ханс-Дитер Флик	Бундес Лига	5
Ювентус	1897	Мурицио Сарри	Серия А	4
Локомотив	1922	Семин Ю.П	РПЛ	53
Рубин	1958	Слуцкий Л.В	РПЛ	196
Арсенал	1886	Микель Артета	АПЛ	10
Бенфика	1904	Бруну Лаже	Примейра	22
Севилья	1890	Лопетеги	Ла Лига	15
Барусия	1900	Люсьен Фавр	Бундес Лига	13
Милан	1899	Стефано Пиоли	Серия А	63

Рисунок Б.5 – Списки команд

На рисунке Б.5 изображен список команд, которые были зарегистрированы в турнире.

Команды

Игроки

Добавление

Выход

Главная

Игроки

Поиск

Имя	Рост	Вес	Номер	Позиция	Клуб
Тимур Магомедов	185	88.0	9	ВРТ	Барселона
Эрнест Степанов	181	73.0	10	ЦАП	ЦСКА
Андрей Дикарев	178	80.0	99	ЦЗ	Севилья
Алексадр Королев	175	70.0	11	ФРВ	АНЖИ
Леонель Месси	170	72.0	4	ПФА	АНЖИ
Алан Дзагоев	185	79.0	10	ЦП	АНЖИ
Эден Азар	168	69.0	2	ЛФА	Бавария
Поль Погба	189	82.0	32	ЦП	Базель
Ибрагим Мамад...	188	88.0	23	ЦЗ	Ахмат
Игорь Акинфеев	189	85.0	43	ВРТ	ЦСКА
Александр Голов...	178	72.0	34	ЦАП	Базель

Рисунок Б.6 – Списки всех игроков в системе

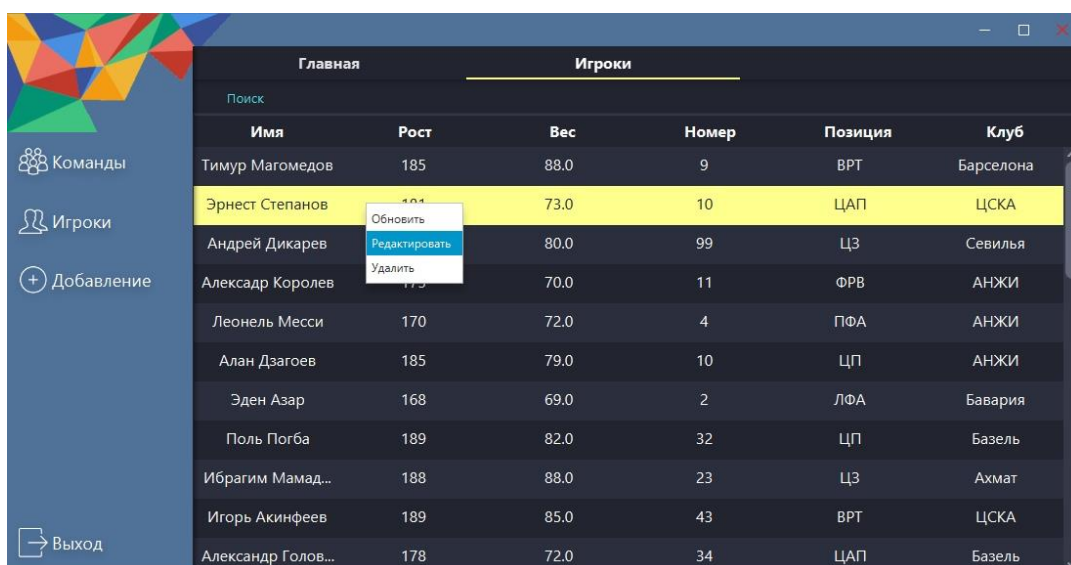


Рисунок Б.7 – Редактирование игрока

Менеджер команды или организатор, могут редактировать или удалить уже существующих игроков. Для этого необходимо выбрать игрока, нажать ПКМ и выбрать «Редактировать» или «Удалить».

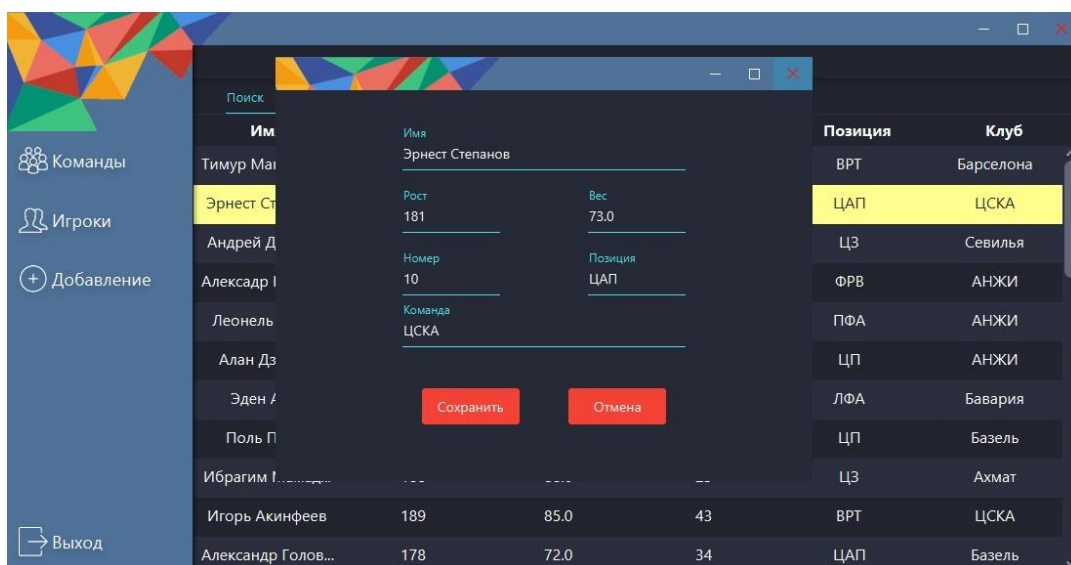


Рисунок Б.8 – Окно редактирования игроков

После нажатия на кнопку «Редактировать» всплывает окно редактирования игрока уже с заполненными данными. менеджеру команды или администратору остается внести необходимые изменения и нажать на кнопку «Сохранить».



Команды	Стадион	Дата	Время
ЦСКА - Спартак	ВЕБ Арена	21.04.2020	18:00
Реал Мадрид - Челси	Лужники	07.04.2020	14:15
Бавария - Боруссия	Альянс Арена	19.03.2020	17:00
Ювентус - Ливерпуль	Энфилд	13.03.2020	15:00
Зенит - Локомотив	РЖД Арена	09.04.2020	20:30
Рубин - Севилья	Казань Арена	10.04.2020	19:45
Шахтер - Арсенал	Металлист	15.03.2020	19:00
Бенфика - Милан	Эштадиу да Луш	04.04.2020	18:30
Байер - Базель	Бай Арена	24.03.2020	12:30
Вердер - Зальцбург	Дженари	09.03.2020	16:00
Порту - Бенфика	Драган	18.03.2020	21:00

480  
Игроков

32  
Команды

Рисунок Б.9 – Списка матчей

На рисунке Б.9 показан список предстоящих матчей турнира, место, дата и время.

Матч	Результат	Проход в следующий этап
ЦСКА - Спартак	2 - 0	ЦСКА
Реал Мадрид - Челси	3 - 1	Реал Мадрид
Бавария - Боруссия	1 - 0	Бавария
Ювентус - Ливерпуль	0 - 2	Ливерпуль
Зенит - Локомотив	0 - 1	Локомотив
Рубин - Севилья	1 - 2	Севилья
Шахтер - Арсенал	2 - 3	Арсенал
Бенфика - Милан	1 - 2	Милан
Байер - Базель	0 - 1	Базель
Вердер - Зальцбург	2 - 1	Вердер
Порту - Бенфика	1 - 3	Бенфика

480  
Игроков

32  
Команды

Рисунок Б.10 – Окно результатов матчей

На рисунке Б.10 показаны результаты сыгранных матчей, итоговый счет, а также команда, которая прошла в следующий этап турнира.

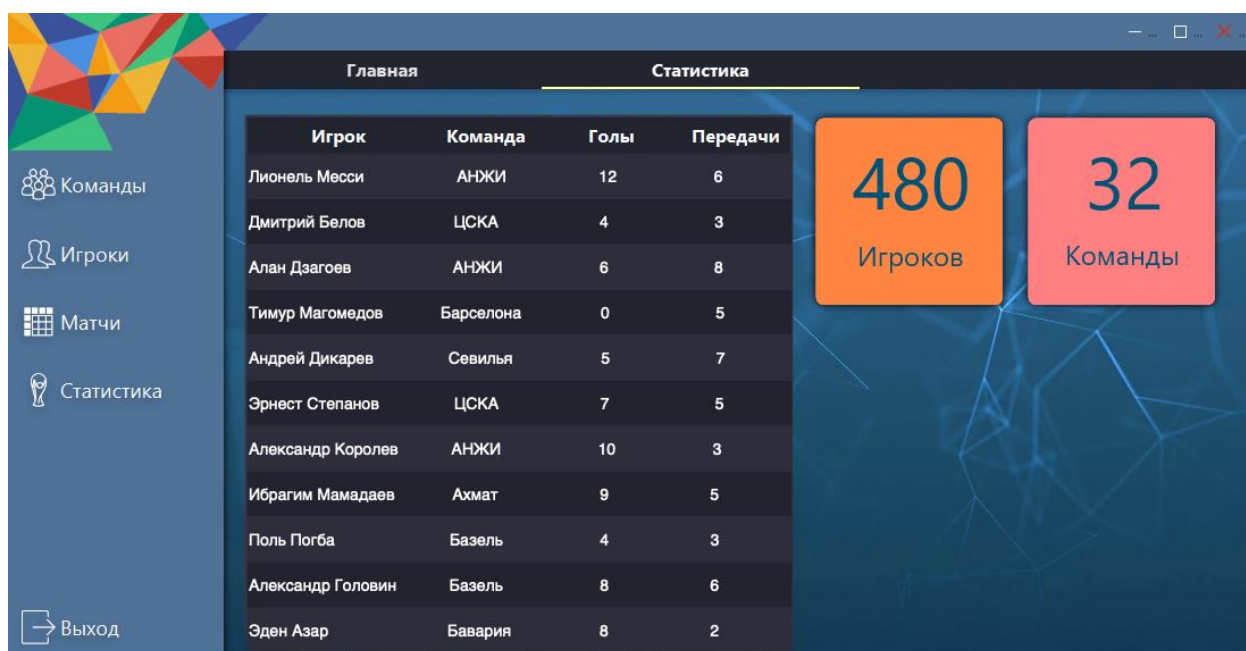


Рисунок Б.11 – Окно статистики игроков

На рисунке Б.11 показана статистика игроков за все матчи турнира. Пользователь может посмотреть информацию, о интересующем игроке.

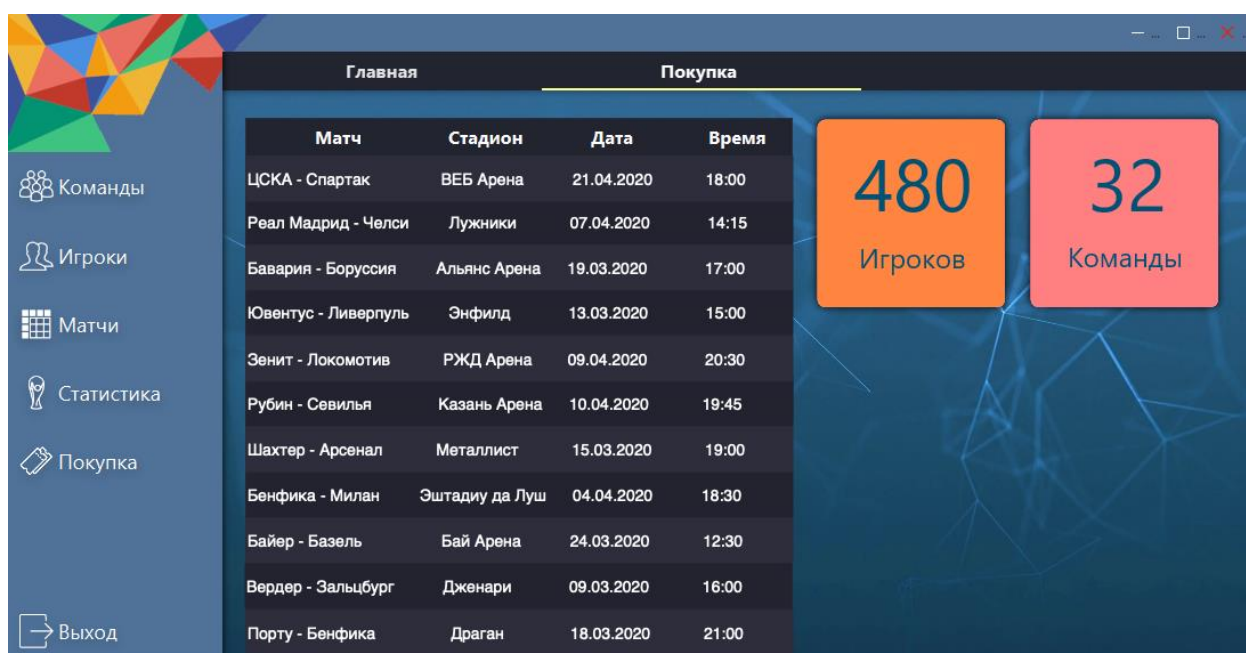


Рисунок Б.12 – Список матчей для покупки билета

Зрителю, чтоб приобрести билет на матчи футбольного турнира нужно сначала выбрать интересующий его матч.



Рисунок Б.13 – Просмотр доступных билетов.

После выбора матча, зритель предоставляется список доступных билетов. Он выбирает билет по критериям: цена, сектор и оставшееся число билетов на выбранный сектор.

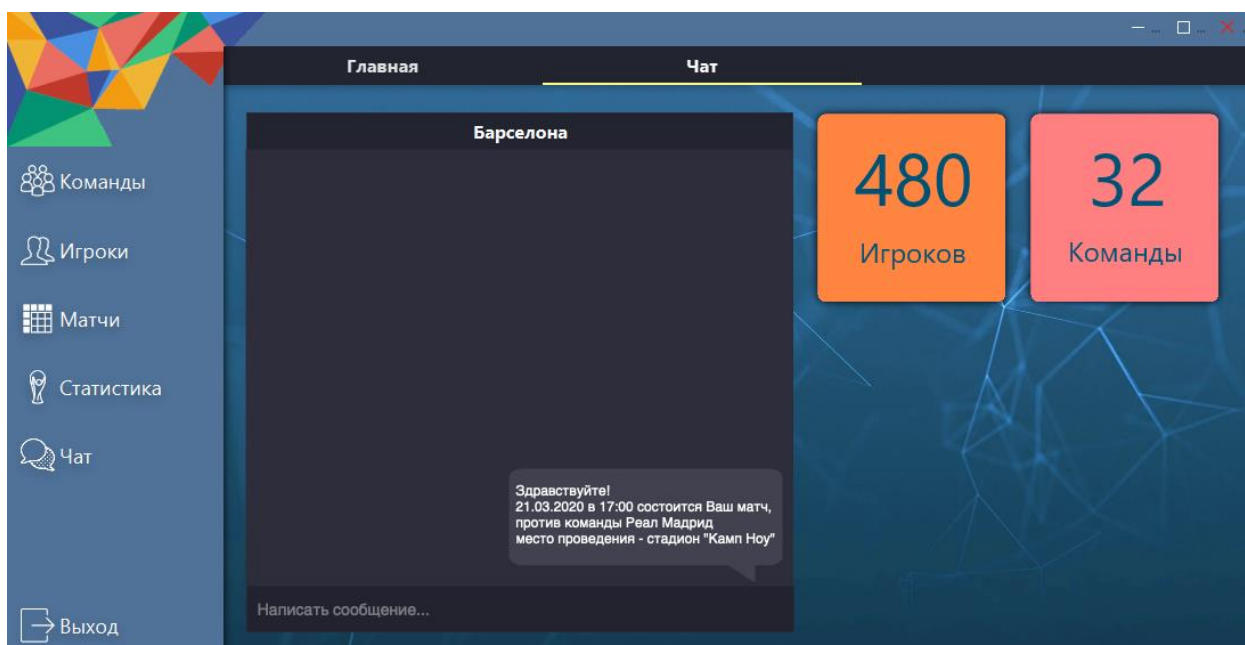


Рисунок Б.14 – Окно диалога между участником и организатор.

На рисунке Б.14 изображено диалоговое окно. Организатор оповещает команду о будущем матче, а также если команде необходим трансфер, то организатор оповещает команду, к какому времени прибудет машина.

## **ПРИЛОЖЕНИЕ В**

(Обязательное)

Фрагменты исходных кодов программы

Листов 14

**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

---

**ФАКУЛЬТЕТ Информатика и системы управления  
КАФЕДРА Компьютерные системы и сети**

**СИСТЕМА ПОДДЕРЖКИ ОРГАНИЗАЦИИ ФУТБОЛЬНОГО ТУРНИРА**

Фрагменты исходного кода

Листов 14

Студент

ИУ6-85Б  
(Группа)



\_\_\_\_\_  
(Подпись, дата)

Э. А. Степанов  
(И.О. Фамилия)

Руководитель

\_\_\_\_\_  
(Подпись, дата)

А. А. Малахов  
(И.О. Фамилия)

2020 г.

## 1. Фрагмент кода редактирования игрока

```
private int ID_player() throws SQLException {
    DatabaseHandler handler = DatabaseHandler.getInstance();
    String nam = name_team.getText();
    String position = role.getText();
    return handler.getId(nam, position);
}

@FXML
private void addMember(ActionEvent event) {
    String mName = name.getText();
    String mHeight = height.getText();
    String mWeight = weight.getText();
    String mNumber = numder.getText();
    String mRole = role.getText();
    String mTeam_name = name_team.getText();

    boolean flag = mName.isEmpty() || mHeight.isEmpty() || mWeight.isEmpty()
|| mNumber.isEmpty() || mRole.isEmpty() || mTeam_name.isEmpty();
    if (flag) {
        AlertMaker.showMaterialDialog(rootPane, mainContainer, new Ar-
rayList<>(), "Insufficient Data", "Please enter data in all fields.");
        return;
    }

    if (isInEditMode) {
        try {
            handleUpdateMember();
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return;
    }
}

void infalteUI(Player player) {
    name.setText(player.getName());
    height.setText(String.valueOf(player.getHeight()));
    weight.setText(String.valueOf(player.getWeight()));
    numder.setText(String.valueOf(player.getNumber()));
    role.setText(player.getRole());
    role.setEditable(false);
}
```

```

        name_team.setText(player.getTeam_name());
        name_team.setEditable(false);
        isInEditMode = Boolean.TRUE;
    }

    private void handleUpdateMember() throws SQLException {
        Player player = new Player(name.getText(), Integer.parse-
seInt(height.getText()), Float.parseFloat(weight.getText()), Integer.parseInt(num-
der.getText()), role.getText(), name_team.getText());
        if (DatabaseHandler.getInstance().updateMember(player, ID_player()) {
            AlertMaker.showMaterialDialog(rootPane, mainContainer, new Ar-
rayList<>(), "Success", "Member data updated.");
        } else {
            AlertMaker.showMaterialDialog(rootPane, mainContainer, new Ar-
rayList<>(), "Failed", "Cant update member.");
        }
    }
}

```

## 2. Фрагмент кода отображения таблицы игроков

```

private void initCol() {
    name.setCellValueFactory(new PropertyValueFactory<>("name"));
    height.setCellValueFactory(new PropertyValueFactory<>("height"));
    weight.setCellValueFactory(new PropertyValueFactory<>("weight"));
    number.setCellValueFactory(new PropertyValueFactory<>("number"));
    role.setCellValueFactory(new PropertyValueFactory<>("role"));
    team_name.setCellValueFactory(new PropertyValueFac-
tory<>("team_name"));
}

private void loadData() {
    list.clear();

    DatabaseHandler handler = DatabaseHandler.getInstance();
    String qu = "SELECT player.name,\n" +
        "    player.height,\n" +
        "    player.weight,\n" +
        "    player.numder,\n" +
        "    player.role_position,\n" +
        "    club.team_name                AS CLUB\n" +
        "FROM player,\n" +
        "    club\n" +
        "WHERE player.club_id = club.id";
    ResultSet rs = handler.execQuery(qu);
    try {

```

```

while (rs.next()) {
    String name = rs.getString("NAME");
    int height = rs.getInt("HEIGHT");
    float weight = rs.getFloat("WEIGHT");
    int number = rs.getInt("NUMDER");
    String role = rs.getString("ROLE_POSITION");
    String team_name = rs.getString("CLUB");

    list.add(new Player(name, height, weight, number, role, team_name));
}
} catch (SQLException ex) {
    ex.printStackTrace();
}

tableView.setItems(list);
}

private void initFilter() {
    filter.textProperty().addListener(o -> {
        if (filter.textProperty().get().isEmpty()) {
            tableView.setItems(list);
            return;
        }
        ObservableList<Player> tableItems = FXCollections.observableAr-
rayList();
        ObservableList<TableColumn<Player, ?>> cols = tableView.getCol-
umns();
        for (Player aFilterData : list) {
            for (TableColumn<Player, ?> col1 : cols) {
                String cellValue = ((TableColumn) col1).getCellData(aFilter-
Data).toString();
                cellValue = cellValue.toLowerCase();
                if (cellValue.contains(filter.textProperty().get().toLowerCase())) {
                    tableItems.add(aFilterData);
                    break;
                }
            }
        }
        tableView.setItems(tableItems);
    });
}

```



### 3. Фрагмент кода соединения и работы с БД

```
public final class DatabaseHandler {

    private static DatabaseHandler handler = null;
    private static Connection conn = null;
    private static Statement stmt = null;

    static {
        createConnection();
    }

    private DatabaseHandler() {
    }

    public static DatabaseHandler getInstance() {
        if (handler == null) {
            handler = new DatabaseHandler();
        }
        return handler;
    }

    Connection getConnection() {
        return conn;
    }

    private static void createConnection() {
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver").newInstance();
            Locale.setDefault(Locale.ENGLISH);
            conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE", "USERTEST", "1111");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public ResultSet execQuery(String query) {
        ResultSet result;
        try {
            stmt = conn.createStatement();
            result = stmt.executeQuery(query);
        } catch (SQLException ex) {
            ex.printStackTrace();
            return null;
        }
    }
}
```

```

    }
    return result;
}

public static void main(String[] args) {
    DatabaseHandler.getInstance();
}

public ResultSet singUser(String name, String password) throws SQLException
{
    String sql = "SELECT * FROM MANAGER WHERE LAST_NAME = ?
AND PASSWORD = ?";
    PreparedStatement ps = conn.prepareStatement(sql);
    ps.setString(1, name);
    ps.setString(2, password);
    return ps.executeQuery();
}

public int getId(String team_name, String role_position) throws SQLException
{
    String sql = "SELECT player.id FROM player WHERE player.CLUB_ID =
(SELECT id FROM club WHERE team_name = ?) AND player.role_position =
?";
    PreparedStatement ps = conn.prepareStatement(sql);
    ps.setString(1, team_name);
    ps.setString(2, role_position);
    ResultSet resultSet = ps.executeQuery();
    if (resultSet.next())
        return resultSet.getInt("id");
    return -1;
}

public boolean updateMember(Player player, int ID) {
    try {
        String update = "UPDATE USER007.PLAYER t\n" +
            "SET t.\"NAME\"      = ?,\n" +
            "    t.HEIGHT      = ?,\n" +
            "    t.WEIGHT       = ?,\n" +
            "    t.NUMDER        = ?,\n" +
            "    t.ROLE_POSITION = ?,\n" +
            "    t.CLUB_ID       = (SELECT id\n" +
            "                        FROM club\n" +
            "                        WHERE team_name = ?)\n" +
            "WHERE t.ID = ?";
    }
}

```

```

        PreparedStatement stmt = conn.prepareStatement(update);
        stmt.setString(1, player.getName());
        stmt.setInt(2, player.getHeight());
        stmt.setFloat(3, player.getWeight());
        stmt.setInt(4, player.getNumber());
        stmt.setString(5, player.getRole());
        stmt.setString(6, player.getTeam_name());
        stmt.setString(7, String.valueOf(ID));
        int res = stmt.executeUpdate();
        return (res > 0);
    } catch (SQLException ex) {
        ex.printStackTrace();
    }
    return false;
}

public boolean deleteMember(Player player) {
    try {
        String deleteStatement = "DELETE FROM player WHERE NAME = ?";
        PreparedStatement stmt = conn.prepareStatement(deleteStatement);
        stmt.setString(1, player.getName());
        int res = stmt.executeUpdate();
        if (res == 1) {
            return true;
        }
    } catch (SQLException ex) {
        ex.printStackTrace();
    }
    return false;
}
}

```

#### **4. Фрагмент кода создания нового менеджера**

```

public static boolean insertNewManager(Manager manager) {
    try {
        PreparedStatement statement = DatabaseHandler.getInstance().getConnection().prepareStatement(
            "INSERT INTO MANAGER(ID, FIRST_NAME, LAST_NAME,
            PASSWORD) VALUES (MANAGER_SEQ.nextval, ?, ?, ?)");
        statement.setString(1, manager.getName());
        statement.setString(2, manager.getLogin());
        statement.setString(3, manager.getPassword());
        return statement.executeUpdate() > 0;
    } catch (SQLException ex) {

```

```

        ex.printStackTrace();

    }
    return false;
}

public static boolean isBookExists(String login) {
    try {
        String checkstmt = "SELECT COUNT(*) FROM manager WHERE
LAST_NAME = ?";
        PreparedStatement stmt = DatabaseHandler.getInstance().getConnection().prepareStatement(checkstmt);
        stmt.setString(1, login);
        ResultSet rs = stmt.executeQuery();
        if (rs.next()) {
            int count = rs.getInt(1);
            System.out.println(count);
            return (count > 0);
        }
    } catch (SQLException ex) {
        ex.printStackTrace();
    }
    return false;
}
}

```

### **5. Фрагмент кода создания Базы Данных**

```

CREATE TABLE calendar_of_games (
    id          NUMBER(10) NOT NULL,
    data_games  TIMESTAMP(0) NOT NULL,
    away_team   NUMBER(10) NOT NULL,
    host_team   NUMBER(10) NOT NULL,
    ticket_id   NUMBER(10) NOT NULL,
    championship_id NUMBER(10) NOT NULL
)
LOGGING;

ALTER TABLE calendar_of_games ADD CONSTRAINT calendar_of_games_pk
PRIMARY KEY ( id );

CREATE TABLE championship (
    id      NUMBER(10) NOT NULL,
    name    VARCHAR2(100) NOT NULL,
    rating  NUMBER(10)

```

```
)  
LOGGING;
```

```
ALTER TABLE championship ADD CONSTRAINT championship_pk PRIMARY KEY ( id );
```

```
CREATE TABLE city (  
    id          NUMBER(10) NOT NULL,  
    name        VARCHAR2(100) NOT NULL,  
    country_id  NUMBER(10) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE city ADD CONSTRAINT city_pk PRIMARY KEY ( id );
```

```
CREATE TABLE club (  
    id          NUMBER(10) NOT NULL,  
    team_name    VARCHAR2(100) NOT NULL,  
    year_of_foundation DATE,  
    president    VARCHAR2(100),  
    championship_id NUMBER(10) NOT NULL,  
    stadium_id   NUMBER(10) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE club ADD CONSTRAINT club_pk PRIMARY KEY ( id );
```

```
CREATE TABLE coach (  
    id          NUMBER(10) NOT NULL,  
    last_name   VARCHAR2(50),  
    name        VARCHAR2(50),  
    country_id  NUMBER(10) NOT NULL,  
    club_id     NUMBER(10) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE coach ADD CONSTRAINT coach_pk PRIMARY KEY ( id );
```

```
CREATE TABLE country (  
    id          NUMBER(10) NOT NULL,  
    name        VARCHAR2(100) NOT NULL,  
    language    VARCHAR2(50)  
)  
LOGGING;
```

```
ALTER TABLE country ADD CONSTRAINT country_pk PRIMARY KEY ( id );
```

```
CREATE TABLE field (  
    id      NUMBER(2) NOT NULL,  
    coating VARCHAR2(50) NOT NULL,  
    "long"  VARCHAR2(50) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE field ADD CONSTRAINT field_pk PRIMARY KEY ( id );
```

```
CREATE TABLE golden_shoe (  
    id          NUMBER(10) NOT NULL,  
    goals       NUMBER(3) NOT NULL,  
    coefficient  NUMBER(2,1) NOT NULL,  
    points      NUMBER(5),  
    player_id   NUMBER(2) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE golden_shoe ADD CONSTRAINT golden_shoe_pk PRIMARY  
KEY ( id );
```

```
CREATE TABLE player (  
    id          NUMBER(10) NOT NULL,  
    last_name   VARCHAR2(50) NOT NULL,  
    name        VARCHAR2(50) NOT NULL,  
    date_of_birth DATE NOT NULL,  
    height      NUMBER(3) NOT NULL,  
    weight      NUMBER(4,1) NOT NULL,  
    number      NUMBER(2) NOT NULL,  
    country_id  NUMBER(10) NOT NULL,  
    role_position VARCHAR2(10) NOT NULL,  
    club_id     NUMBER(10) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE player ADD CONSTRAINT player_pk PRIMARY KEY ( id );
```

```
CREATE TABLE rating_uefa (  
    id      NUMBER(10) NOT NULL,  
    club_id NUMBER(10) NOT NULL,  
    rating  NUMBER(7,3) NOT NULL
```

```
)  
LOGGING;
```

```
ALTER TABLE rating_uefa ADD CONSTRAINT rating_uefa_pk PRIMARY  
KEY ( id );
```

```
CREATE TABLE role (  
    position VARCHAR2(10) NOT NULL,  
    name      VARCHAR2(100) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE role ADD CONSTRAINT role_pk PRIMARY KEY ( position );
```

```
CREATE TABLE stadium (  
    id          NUMBER(10) NOT NULL,  
    name        VARCHAR2(50) NOT NULL,  
    capacity     NUMBER(6) NOT NULL,  
    year_of_foundation DATE,  
    field_id     NUMBER(2) NOT NULL,  
    city_id      NUMBER(10) NOT NULL  
)  
LOGGING;
```

```
ALTER TABLE stadium ADD CONSTRAINT stadium_pk PRIMARY KEY ( id  
);
```

```
CREATE TABLE table_of_scorers (  
    id          NUMBER(10) NOT NULL,  
    goals       NUMBER(3) NOT NULL,  
    assist      NUMBER(3) NOT NULL,  
    player_id   NUMBER(10) NOT NULL,  
    kpd         NUMBER(3) NOT NULL,  
    rating      NUMBER(7)  
)  
LOGGING;
```

```
ALTER TABLE table_of_scorers ADD CONSTRAINT table_of_scorers_pk PRI-  
MARY KEY ( id );
```

```
CREATE TABLE ticket (  
    id          NUMBER(10) NOT NULL,  
    price       NUMBER(6,2) NOT NULL  
)
```

LOGGING;

ALTER TABLE ticket ADD CONSTRAINT ticket\_pk PRIMARY KEY ( id );

```
CREATE TABLE tournament_table (  
  id          NUMBER(10) NOT NULL,  
  games       NUMBER(2) NOT NULL,  
  win         NUMBER(2) NOT NULL,  
  draw        NUMBER(2) NOT NULL,  
  defeat      NUMBER(2) NOT NULL,  
  points      NUMBER(3) NOT NULL,  
  championship_id NUMBER(10) NOT NULL,  
  club_id     NUMBER(10) NOT NULL  
)  
LOGGING;
```

ALTER TABLE tournament\_table ADD CONSTRAINT tournament\_table\_pk  
PRIMARY KEY ( id );

```
CREATE TABLE transfer_list (  
  id          NUMBER(10) NOT NULL,  
  value       NUMBER(12,2) NOT NULL,  
  wage        NUMBER(12,2) NOT NULL,  
  current_club NUMBER(10) NOT NULL,  
  future_club  NUMBER(10) NOT NULL,  
  player_id   NUMBER(10) NOT NULL  
)  
LOGGING;
```

ALTER TABLE transfer\_list ADD CONSTRAINT transfer\_list\_pk PRIMARY  
KEY ( id );

```
ALTER TABLE calendar_of_games  
  ADD CONSTRAINT calendar_of_games_club_fk FOREIGN KEY (  
away_team )  
  REFERENCES club ( id )  
  NOT DEFERRABLE;
```

```
ALTER TABLE calendar_of_games  
  ADD CONSTRAINT calendar_of_games_club_fkv2 FOREIGN KEY (  
host_team )  
  REFERENCES club ( id )  
  NOT DEFERRABLE;
```



```
ALTER TABLE calendar_of_games
  ADD CONSTRAINT calendar_of_games_ticket_fk FOREIGN KEY ( ticket_id
)
  REFERENCES ticket ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE tournament_table
  ADD CONSTRAINT championship_fk FOREIGN KEY ( championship_id )
  REFERENCES championship ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE calendar_of_games
  ADD CONSTRAINT championship_fkv2 FOREIGN KEY ( championship_id )
  REFERENCES championship ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE city
  ADD CONSTRAINT city_country_fk FOREIGN KEY ( country_id )
  REFERENCES country ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE club
  ADD CONSTRAINT club_championship_fk FOREIGN KEY ( champion-
ship_id )
  REFERENCES championship ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE tournament_table
  ADD CONSTRAINT club_fk FOREIGN KEY ( club_id )
  REFERENCES club ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE club
  ADD CONSTRAINT club_stadium_fk FOREIGN KEY ( stadium_id )
  REFERENCES stadium ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE coach
  ADD CONSTRAINT coach_club_fk FOREIGN KEY ( club_id )
  REFERENCES club ( id )
  NOT DEFERRABLE;
```

```
ALTER TABLE coach
  ADD CONSTRAINT coach_country_fk FOREIGN KEY ( country_id )
```

```
REFERENCES country ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE player  
ADD CONSTRAINT player_club_fk FOREIGN KEY ( club_id )  
REFERENCES club ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE player  
ADD CONSTRAINT player_country_fk FOREIGN KEY ( country_id )  
REFERENCES country ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE golden_shoe  
ADD CONSTRAINT player_fk FOREIGN KEY ( player_id )  
REFERENCES player ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE table_of_scorers  
ADD CONSTRAINT player_fkv1 FOREIGN KEY ( player_id )  
REFERENCES player ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE player  
ADD CONSTRAINT player_role_fk FOREIGN KEY ( role_position )  
REFERENCES role ( position )  
NOT DEFERRABLE;
```

```
ALTER TABLE rating_uefa  
ADD CONSTRAINT rating_uefa_club_fk FOREIGN KEY ( club_id )  
REFERENCES club ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE stadium  
ADD CONSTRAINT stadium_city_fk FOREIGN KEY ( city_id )  
REFERENCES city ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE stadium  
ADD CONSTRAINT stadium_field_fk FOREIGN KEY ( field_id )  
REFERENCES field ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE transfer_list
```

```
ADD CONSTRAINT transfer_list_club_fk FOREIGN KEY ( current_club )  
    REFERENCES club ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE transfer_list  
    ADD CONSTRAINT transfer_list_club_fk2 FOREIGN KEY ( future_club )  
        REFERENCES club ( id )  
NOT DEFERRABLE;
```

```
ALTER TABLE transfer_list  
    ADD CONSTRAINT transfer_list_player_fk FOREIGN KEY ( player_id )  
        REFERENCES player ( id )  
NOT DEFERRABLE;
```

```
CREATE SEQUENCE field_seq  
    MINVALUE 1  
    START WITH 1  
    INCREMENT BY 1;
```

```
CREATE SEQUENCE country_seq  
    MINVALUE 1  
    START WITH 1  
    INCREMENT BY 1;
```

```
CREATE SEQUENCE city_seq  
    MINVALUE 1  
    START WITH 1  
    INCREMENT BY 1;
```

```
CREATE SEQUENCE stadium_seq  
    MINVALUE 1  
    START WITH 1  
    INCREMENT BY 1;
```

## **ПРИЛОЖЕНИЕ Г**

**(Обязательное)**

Копии листов графической части

Листов 8

## Результаты анализа существующих систем

№	Критерии	Сервис		
		Eventbrite	SmartSheet	Eventleaf
1	Доступность информации о системе	-	+	+
2	Простота использования	+	-	-
3	Стоимость использования	-	-	-
4	Возможность проведения масштабного мероприятия	+	+	+
5	Быстрая обработка данных	+	+	-
6	Удобная система регистрации	+	+	+

# Анализ существующих систем

После анализа существующих систем, можно сделать вывод, какие особенности системы нужно учесть при ее разработке. Необходимо будет осуществить следующие

1) Быстрый анализ и обработке входящих данных;

2) удобный интерфейс и простота использования системы;

### 3) Доступность к системе

№	Критерий	Вид OLAP системы		
		МOLAP	ROLAP	НOLAP
1	Безопасность и администрирование обеспечивается реляционными СУБД	+	+	+
2	Возможность манипулирования большими объемами данных	-	+	+
3	Эффективность	+	-	+
4	Производительность	-	+	+
5	Возможность использования систем с хранилищами данных и различными OLTP-системами	+	+	+
6	Высокая скорость обработки запросов	-	+	+
7	Сложно извлекать данные без промежуточной агрегации.	+	+	+
8	Нет необходимости в специализированных инструментах для формирования кубов	+	-	+

### Сравнение OLAP систем

Характеристика	ОЦР	ОДР
Частота обновления данных	Высшая частота	Медлая частота
Источники данных	В основном, внутренние	По отношению к аналитической системе, в основном, внешние
Возраст данных	Текущие	Любой
Уровень агрегации данных	Детализированные данные	В основном агрегированные данные
Возможности аналитических операций	Разнонаправленные отчеты	Последовательность, универсальных отчетов, вычисление изменений иррегулярной агрегации и срезов данных
Назначение системы	Фактсация, оперативный поиск и обработка данных, разделение/обработка аналитическая обработка	Работа с любым видом данных, аналитическая обработка, прогнозирования, моделирование
Объем данных	Медлый	Быстрый

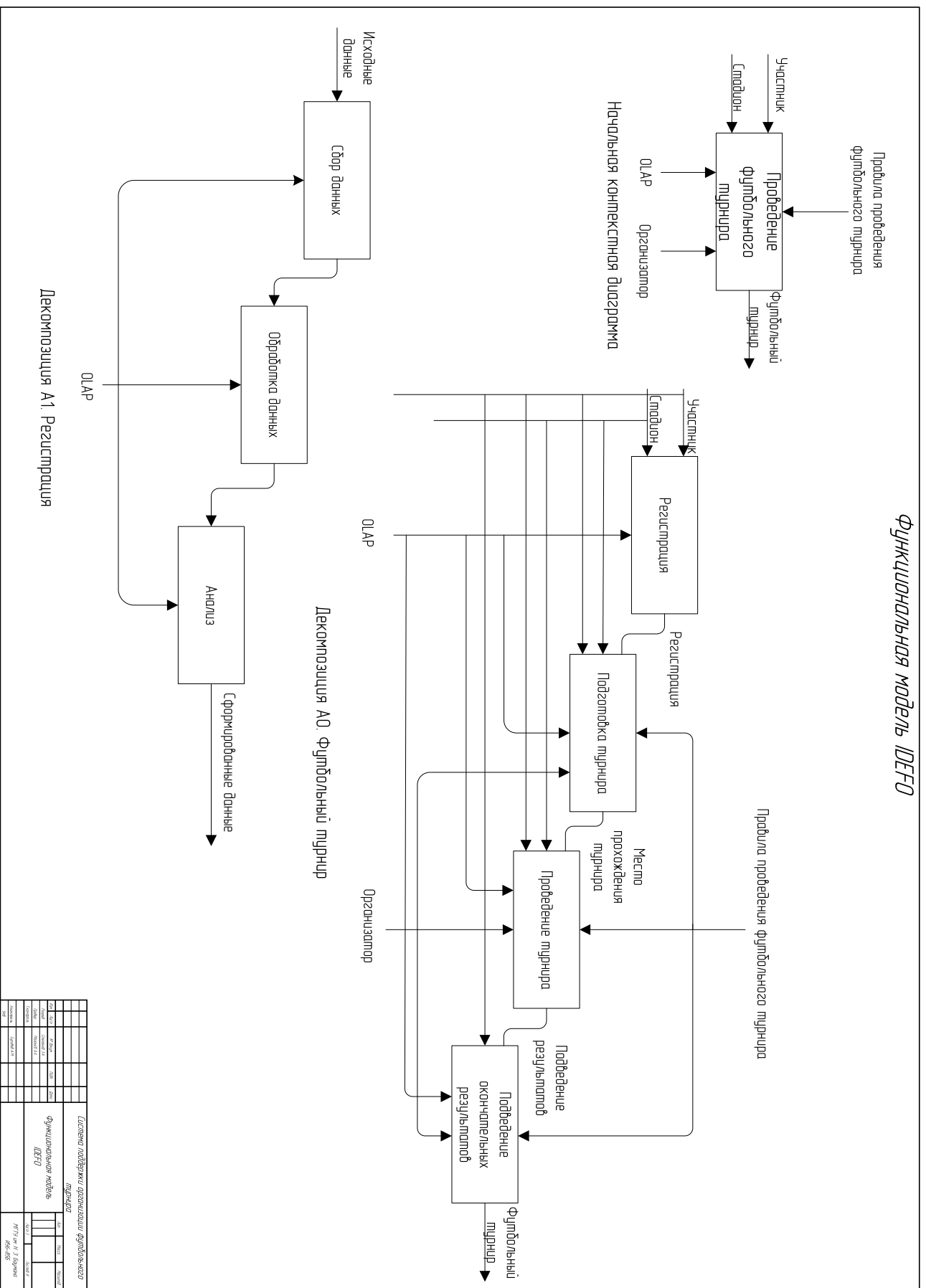
*Сравнительный анализ систем по обработке данных*

По результатам и сравнительному анализу всех OLAP систем, можно сделать вывод, что наиболее удобный инструмент для организации футбольного турнира является мпг НОLAP. Так как он включает в себя как НОLAP Систему со своим явным, так и ROLAP Систему со своим явным. Организация такого мероприятия, как футбольный турнир требует оперативности действий и эффективности проведенной работы.

Судимые по делу о преступлении, совершенном в период наказания									
Результаты дознания судимых по делу									
№ п/п	Имя	Фамилия	П.И.	Дата рождения	Место рождения	Место жительства	Место работы	Место проживания	Место нахождения
1	Иванов	Иван	Иванович	1980-01-01	Москва	Москва	Москва	Москва	Москва
2	Петров	Петр	Петрович	1985-03-15	Санкт-Петербург	Санкт-Петербург	Санкт-Петербург	Санкт-Петербург	Санкт-Петербург
3	Сидоров	Сидор	Сидорович	1990-07-20	Новосибирск	Новосибирск	Новосибирск	Новосибирск	Новосибирск
4	Климов	Климов	Климович	1988-11-05	Казань	Казань	Казань	Казань	Казань
5	Васильев	Василий	Васильевич	1982-09-10	Волгоград	Волгоград	Волгоград	Волгоград	Волгоград
6	Михайлов	Михаил	Михайлович	1987-04-25	Уфа	Уфа	Уфа	Уфа	Уфа
7	Попов	Попов	Попович	1983-12-08	Томск	Томск	Томск	Томск	Томск
8	Смирнов	Смирнов	Смирнович	1986-06-12	Омск	Омск	Омск	Омск	Омск
9	Зайцев	Зайцев	Зайцевич	1989-02-18	Красноярск	Красноярск	Красноярск	Красноярск	Красноярск
10	Кузнецов	Кузнецов	Кузнецович	1984-08-22	Иркутск	Иркутск	Иркутск	Иркутск	Иркутск
11	Лебедев	Лебедев	Лебедевич	1981-05-03	Хабаровск	Хабаровск	Хабаровск	Хабаровск	Хабаровск
12	Новиков	Новиков	Новикович	1987-10-14	Владивосток	Владивосток	Владивосток	Владивосток	Владивосток
13	Осипов	Осипов	Осипович	1983-07-27	Магнитогорск	Магнитогорск	Магнитогорск	Магнитогорск	Магнитогорск
14	Рябов	Рябов	Рябович	1986-11-09	Самара	Самара	Самара	Самара	Самара
15	Соловьев	Соловьев	Соловьевич	1982-03-21	Тюмень	Тюмень	Тюмень	Тюмень	Тюмень
16	Тихонов	Тихонов	Тихонович	1988-06-04	Ярославль	Ярославль	Ярославль	Ярославль	Ярославль
17	Федотов	Федотов	Федотович	1985-09-17	Иваново	Иваново	Иваново	Иваново	Иваново
18	Харьков	Харьков	Харькович	1989-12-02	Воронеж	Воронеж	Воронеж	Воронеж	Воронеж
19	Цыганов	Цыганов	Цыганович	1984-04-16	Брянск	Брянск	Брянск	Брянск	Брянск
20	Чайков	Чайков	Чайкович	1987-07-29	Пенза	Пенза	Пенза	Пенза	Пенза
21	Шаров	Шаров	Шарович	1981-10-11	Саратов	Саратов	Саратов	Саратов	Саратов
22	Шевченко	Шевченко	Шевченко	1986-01-24	Тамбов	Тамбов	Тамбов	Тамбов	Тамбов
23	Шестаков	Шестаков	Шестакович	1983-05-07	Тверь	Тверь	Тверь	Тверь	Тверь
24	Шибанов	Шибанов	Шибанович	1988-08-19	Тюмень	Тюмень	Тюмень	Тюмень	Тюмень
25	Шумов	Шумов	Шумович	1982-11-03	Ульяновск	Ульяновск	Ульяновск	Ульяновск	Ульяновск
26	Щеглов	Щеглов	Щеглович	1987-02-15	Хмельницкий	Хмельницкий	Хмельницкий	Хмельницкий	Хмельницкий
27	Щербинин	Щербинин	Щербинин	1984-06-28	Челябинск	Челябинск	Челябинск	Челябинск	Челябинск
28	Щукин	Щукин	Щукин	1989-09-10	Якутск	Якутск	Якутск	Якутск	Якутск
29	Югов	Югов	Югович	1983-12-22	Ярославль	Ярославль	Ярославль	Ярославль	Ярославль
30	Яковлев	Яковлев	Яковлевич	1986-04-05	Иркутск	Иркутск	Иркутск	Иркутск	Иркутск



# Функциональная модель IDEFO



Функциональная модель IDEF0 (Продолжение)

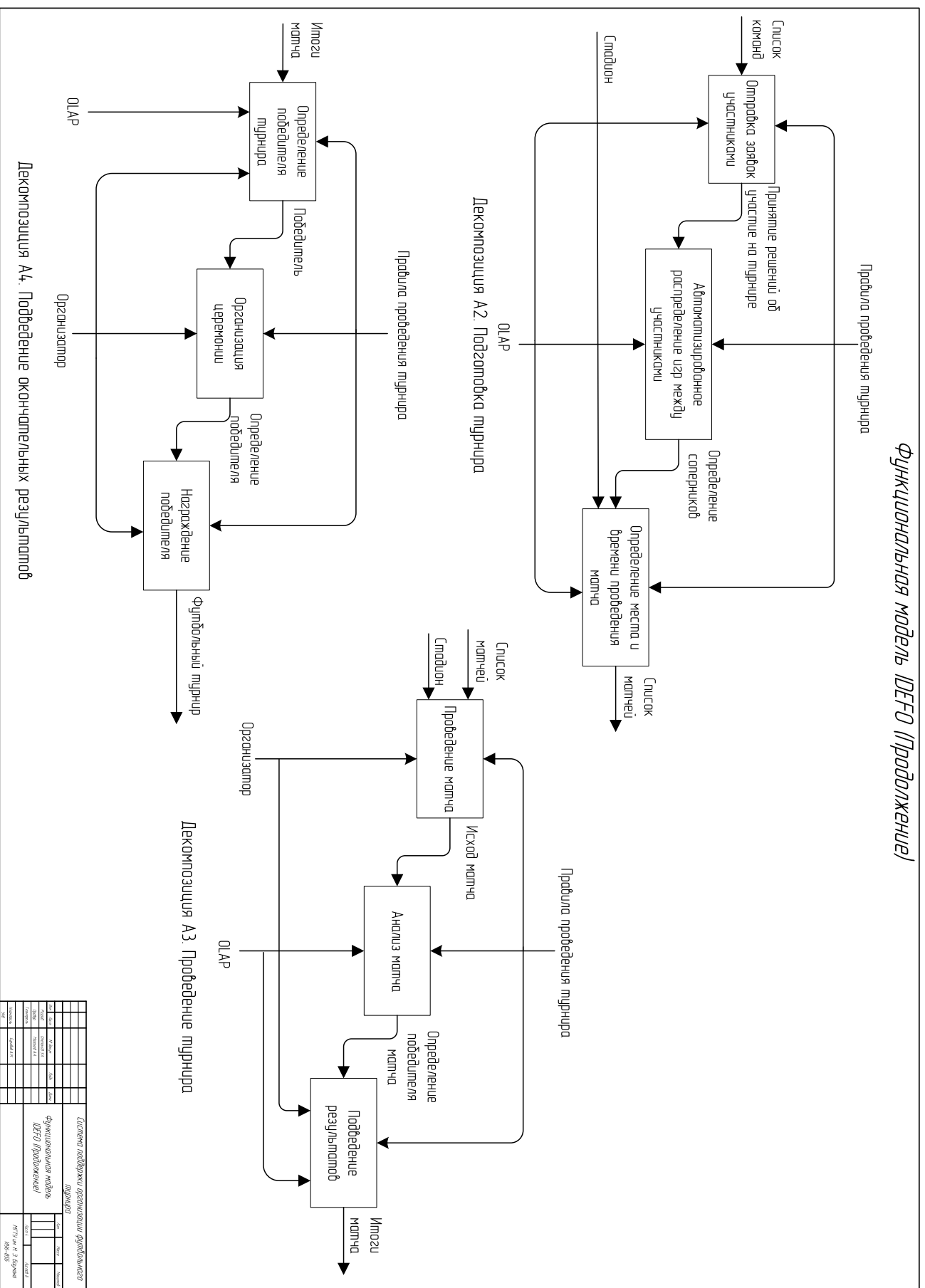
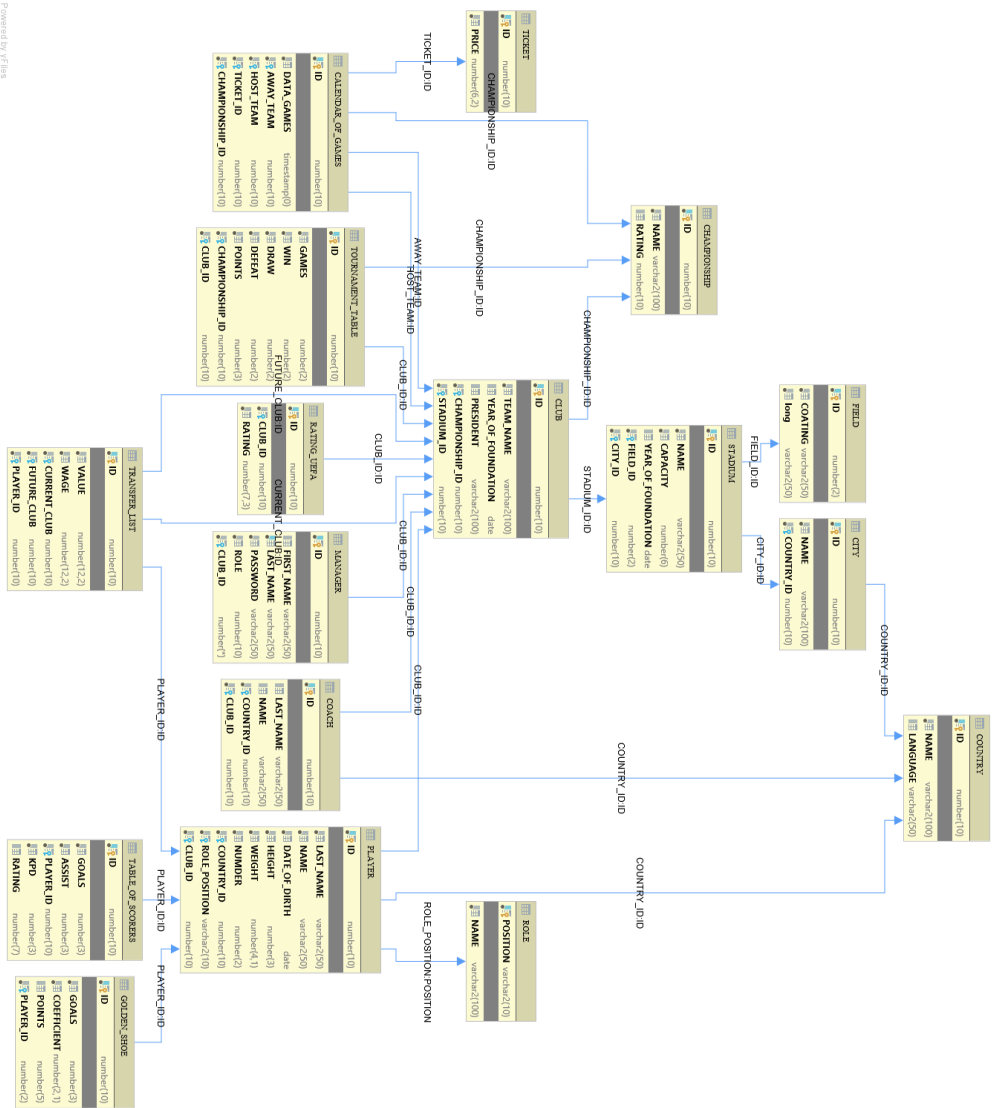




Схема структуры базы данных



Структурная схема базы данных

Средняя нагрузка пользователей									
по дням									
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
Средняя нагрузка пользователей									
по дням									
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
Средняя нагрузка пользователей									
по дням									
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40



Результаты тестирования системы

№	Исход	Путь	Ожидаемый результат	Полученный результат
1	Ничего не выбрано	Ничего не выбрано	Сообщение об ошибке	Сообщение о необходимости ввести данные
2	Введен логин, ничего не выбрано	Ничего не выбрано	Сообщение об ошибке	Сообщение о необходимости ввести данные
3	Введен логин не существует	Ничего не выбрано	Сообщение об ошибке	Сообщение о необходимости ввести данные
4	Введен логин не существует, пароль	Введен пароль	Сообщение об ошибке	Сообщение о необходимости ввести данные
5	Введен логин, ничего не выбрано	Введен некорректный пароль	Сообщение об ошибке	Сообщение о необходимости ввести данные
6	Введен логин, ничего не выбрано	Введен корректный пароль	Успешная авторизация	Успешная авторизация и переход на главный экран

Тестирование экрана авторизации.

При тестировании основных форм информационной системы стратегией черного ящика не было выявлено никаких критических недостатков.

Критерий	Итого
Интерфейс	100%
Читаемость текста	100%
Размер шрифта	75%
Расположение полей ввода и кнопок	75%
Цветовая гамма	50%
Понятность назначения кнопок	100%

Тестирование удобства эксплуатации

Исходя из результатов тестирования можно сказать, что система удобна большинству пользователей, которые проходили тестирование.

№	ФФД	Роль	Е-mail	Путь	Наблюдатель не проан.	Ожидаемый результат	Полученный результат
1	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Сообщение об ошибке	Сообщение о необходимости ввести данные
2	Ничего не выбрано	Ничего не выбрано	Введен логин	Ничего не выбрано	Ничего не выбрано	Сообщение об ошибке	Сообщение о необходимости ввести данные
3	Ничего не выбрано	Ничего не выбрано	Введен логин	Ничего не выбрано	Ничего не выбрано	Сообщение об ошибке	Сообщение о необходимости ввести данные
4	Ничего не выбрано	Ничего не выбрано	Введен логин	Введен некорректный пароль	Введен некорректный пароль	Сообщение об ошибке	Сообщение о необходимости ввести данные
5	Ничего не выбрано	Ничего не выбрано	Введен логин	Введен некорректный пароль	Введен некорректный пароль	Сообщение об ошибке	Сообщение о необходимости ввести данные
6	Ничего не выбрано	Введен логин	Введен логин	Введен логин	Введен логин	Сообщение об ошибке	Сообщение о необходимости ввести данные
7	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Успешная авторизация	Успешная авторизация и переход на главный экран

Тестирование формы регистрации

Критерий	Результат тестирования
Тестирование удобства использования	Сервис полностью соответствует требованиям заказчика, выполнены все заданные функции.
Тестирование конфигурации	Проверка была осуществлена на 2 устройствах и в 3 браузерах. При наличии подключения к Интернету, сервис хорошо работает на таких браузерах, как Google Chrome, Internet Explorer 11, Mozilla Firefox.

Тестирование конфигурации оборудования

Результаты тестирования конфигурации также показали соответствием требованиям указанным в техническом задании.

№	ФФД	Возраст	Рост	Пол	Пол	Пол	Пол	Пол	Пол
1	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано
2	Введен логин	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано
3	Введен логин	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано	Ничего не выбрано
4	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин
5	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин
6	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин	Введен логин

Тестирование формы добавления узла

Выполнено функциональное тестирование «Черным ящиком», а также выполнено оценочное тестирование удобства эксплуатации и конфигурации. По итогам тестирования сделан вывод о соответствии разработанного программного продукта всем функциям и конфигурационным требованиям, указанным в техническом задании.

Выполнено функциональное тестирование				Система полностью соответствует требованиям			
№	Исход	Путь	Результат	№	Исход	Путь	Результат
1	Ничего не выбрано	Ничего не выбрано	Сообщение об ошибке	1	Ничего не выбрано	Ничего не выбрано	Сообщение об ошибке
2	Введен логин	Ничего не выбрано	Сообщение об ошибке	2	Введен логин	Ничего не выбрано	Сообщение об ошибке
3	Введен логин	Ничего не выбрано	Сообщение об ошибке	3	Введен логин	Ничего не выбрано	Сообщение об ошибке
4	Введен логин	Ничего не выбрано	Сообщение об ошибке	4	Введен логин	Ничего не выбрано	Сообщение об ошибке
5	Введен логин	Ничего не выбрано	Сообщение об ошибке	5	Введен логин	Ничего не выбрано	Сообщение об ошибке
6	Введен логин	Ничего не выбрано	Сообщение об ошибке	6	Введен логин	Ничего не выбрано	Сообщение об ошибке

## Формы интерфейса

Вход в систему

## Добавление игроков

**ФИО**

---

**Возраст**

---

**Рост**

---

**Номер**

---

**Позиция**

---

**Команда**

---

Сохранить

Отмена

## Добавление узлоков в систему

Гравци					Матчи	
Команда	Стран	Дата	Време			
ЛСРД - Спартак	БЕЛ Арена	21.04.2020	18:30			
Ран Мадри - Чехия	Премия	07.04.2020	14:15			
Бавария - Борусия	Алианс Арена	19.03.2020	17:30			
Спартак - Динабург	Зенит	13.03.2020	15:00			
Зенит - Ливорно	РСКД Арена	09.04.2020	20:30			
Рубин - Спартак	Казань Арена	10.04.2020	18:45			
Спартак - Ахмат	Мордовия	15.03.2020	19:30			
Бавария - Манч	Зенит да Рун	04.04.2020	18:30			
Бейер - Бавар	Бай Арена	24.03.2020	12:30			
Бавария - Зенитбург	Динамо	09.03.2020	18:00			
Турция - Бавария	Дарм	18.03.2020	21:00			

Список сформированных матчей турнира

Город	Россия		Израиль		Израиль		Израиль		Израиль	
	Имя	Возраст	Возраст	Имя	Имя	Имя	Имя	Имя	Имя	Имя
Москва	Александр	185	86.0	9	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Санкт-Петербург	Александр	181	72.0	10	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Новосибирск	Александр	178	80.0	99	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Казань	Александр	175	70.0	11	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Иркутск	Александр	170	72.0	4	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Астрахань	Александр	185	79.0	10	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Екатеринбург	Александр	168	69.0	2	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Иркутск	Александр	189	82.0	32	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Новосибирск	Александр	188	86.0	23	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Иркутск	Александр	189	85.0	43	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль
Иркутск	Александр	178	72.0	34	Израиль	Израиль	Израиль	Израиль	Израиль	Израиль

Просмотр всех узлоков системы

## Диагностическое окно

Категория		Результат	
Матч	Результат	Против и составный состав	ЛСРКА
Колумбия vs Украина	3-1	Ривалдо, Мелка	
Англия vs Украина	1-0	Белков	
Колумбия - Бразилия	0-2	Панателас	
США - Франция	0-1	Ривалдо	
Россия - Чехия	1-2	Семак	
Украина - Аргентина	2-3	Акинфеев	
Бразилия - Италия	1-2	Мендес	
Бразилия - Италия	0-1	Белков	
Бразилия - Австралия	2-1	Белков	
Россия - Франция	1-3	Белков	

## Результаты сыгранных матчей

[illegible]