

Министерство науки и высшего образования Российской Федерации  
**Муромский институт (филиал)**  
федерального государственного бюджетного образовательного учреждения высшего образования  
**«Владимирский государственный университет  
Имени Александра Григорьевича и Николая Григорьевича Столетовых»  
(МИВлГУ)**

Факультет \_\_\_\_\_ ИТР \_\_\_\_\_

Кафедра \_\_\_\_\_ ПИИ \_\_\_\_\_

УТВЕРЖДАЮ

Зав. кафедрой

\_\_\_\_\_ А.Л. Жизняков  
(подпись)

« \_\_\_\_\_ » \_\_\_\_\_ 2023г

## БАКАЛАВРСКАЯ РАБОТА

Тема \_\_\_\_\_ Разработка распределенной информационной системы

\_\_\_\_\_ "Волейбольный помощник"

\_\_\_\_\_ МИВУ.09.03.04-12.000 ВКР

Руководитель

Привезенцев Д.Г.  
(фамилия, инициалы)

\_\_\_\_\_ (подпись) \_\_\_\_\_ (дата)

Студент \_\_\_\_\_ ПИИ-119  
(группа)

Львова К.Ю.  
(фамилия, инициалы)

\_\_\_\_\_ (подпись) \_\_\_\_\_ (дата)

Муром 2023



В данной выпускной квалификационной работе разработана распределенная информационная система «Волейбольный помощник». В ходе выполнения работы произведен анализ и сбор требований к проектируемой системе. На этапе разработки структуры классов выделены основные сущности, их атрибуты и взаимосвязи. Система состоит из таких компонентов как Android-приложение, Web-сервер и Web-сайт. На заключительном этапе работы произведено тестирование разработанного продукта.

In this final qualifying work, a distributed information system "Volleyball Assistant" was designed and developed. In the course of this work, an analysis and collection of requirements for the system being designed was carried out. At the stage of developing the class structure, the main entities, their attributes and relationships are identified. The system consists of such components as an Android application, a Web site, and a Web service. At the final stage of the work, the developed product was tested.

## Содержание

Введение .....	7
1 Анализ технического задания .....	9
1.1 Описание предметной области .....	9
1.2 Извлечение требований .....	14
1.3 Анализ требований .....	15
2 Разработка алгоритмов .....	18
2.1 Процессы системы .....	18
2.2 Моделирование данных .....	20
2.2.1 Моделирование серверной части .....	21
2.2.2 Моделирование локальной базы данных .....	25
2.3 Основные действия и сценарии использования .....	27
2.3.1 Выявление основных действий .....	27
2.3.2 Основные сценарии использования .....	31
3 Реализация системы .....	34
3.1 Выбор и основание средств разработки .....	34
3.2 Руководство программиста .....	36
3.2.1 Мобильное приложение .....	36
3.2.2 Web-сервис .....	44
4 Руководство пользователя .....	48
4.1 Мобильное приложение .....	48
4.1.1 Не авторизированный пользователь .....	48
4.1.2 Авторизированный пользователь .....	56
4.2 WEB-сайт .....	57
4.2.1 Гость сайта .....	57

					МИВУ 09.03.04-012.000 ПЗ										
Изм.	Лист	№докум.	Подп.	Дата	Разработка распределенной информационной системы «Волейбольный помощник»					Лит.	Лист	Листов			
Разраб.		Львова К.Ю.								у			5	89	
Пров.										МИ ВлГУ ПИН-119					
Н.контр.															
Утв.															

4.2.2	Авторизированный пользователь .....	57
4.2.3	Администратор.....	60
5	Тестирование .....	61
	Заключение .....	70
	Список использованных источников .....	71
	Приложение 1. Код программы .....	73
	Приложение 2. Интервью с заказчиком.....	74
	Приложение 3. Скриншоты программы .....	76

## Введение

С развитием спорта внедрение информационных технологий приобретает все большую актуальность. Особенно для тренировочного процесса волейболистов требующего постоянного общения и обмена опытом.

Цифровые видео, табло и проекционная техника, а также различные измерительные системы широко используются на всех крупных соревнованиях.

Волейбол - один из самых распространенных видов спорта. Так же это игра не сложная по правилам, как раз подходящая для отдыха, поэтому она увлекает многих. Проще говоря - вид спорта для всех.

Секции волейбола есть почти в каждом учебном заведении не только в нашей стране, но и в целом мире. Поэтому существует необходимость небольшой автоматизации тренировочных процессов и организации официальных матчей.

Используя опыт игры в волейбол как в общеобразовательной и спортивной школах, институте и участия в городских и областных соревнованиях, можно заметить, что автоматизация игрового процесса необходима при ведении счета, определения переходов при забитом мяче, контроле взятия таймаутов и замен тренерами, создании протокола. Для облегчения работы судьи и/или секретаря матча необходимо разработать программу для помощи в ведении игрового процесса.

Целью работы является разработка распределенной информационной системы «Волейбольный помощник» для упрощения работы судей на матче и предоставления любителям волейбола возможности самим вести официальные матчи и делится ими с другими пользователями.

Задачами данной работы являются создание мобильного приложения и сайта для реализация следующего функционала:

1) Контроль игрового процесса:

- ведение счета очков,

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		7

- проведение замен и таймаутов.

2) Два типа проведения матчей, первый для тренировок, второй для официальных матчей.

3) Автоматическое составление протокола матча по его окончании.

4) Транслирование матчей на сайт.

5) Администрирование пользователей на сайте.

6) Планирование матчей на определенную дату.

					МИВУ.09.03.04-12.000 ПЗ	Лист
						8
Изм	Лист	№ докум.	Подп.	Дата		



## 1. Анализ технического задания

### 1.1 Описание предметной области

Волейбол - один из самых массовых и зрелищных видов спорта, распространенный на всех континентах. Это игра для отдыха и поддержания формы, которая увлекает многих. Это вид спорта для всех.

Волейбол доступен как для детей, так и для людей пожилого возраста. Он популярен в школах и на спортивных площадках, поддерживается многими институтами и организациями, привлекает молодежь, которая хочет соревноваться на высоком уровне.

Для автоматизации игрового и тренировочного процесса уже существует несколько приложений. Одно из них — это приложение "Оценка по волейболу простая"[1].

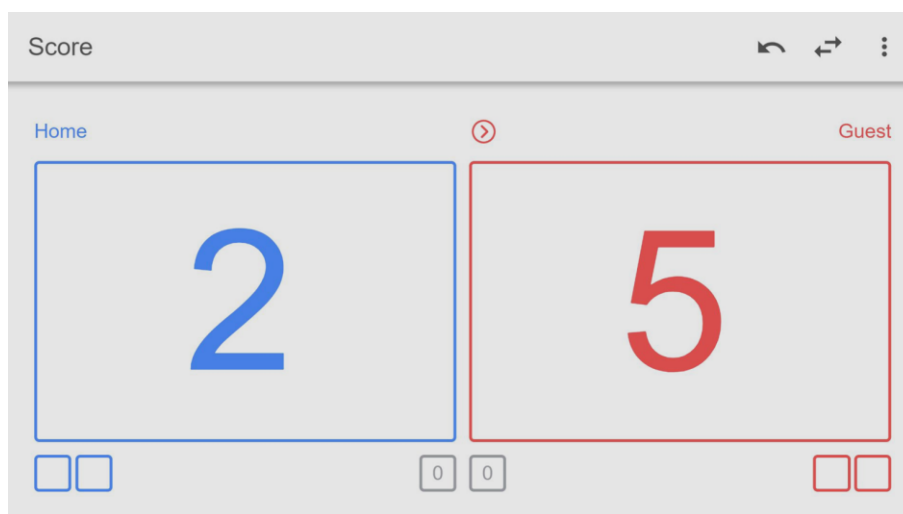


Рисунок 1 - аналог приложение "Оценка по волейболу простая"

Это простое и понятное приложение для подсчета очков, позволяющее подсчитывать ваши очки и отслеживать ваш результат в волейболе. Это совершенно бесплатно и не содержит рекламы.

Особенности:

- увеличить количество очков можно нажатием на большие поля.

- кнопкой отмены можно уменьшить количество очков.
- значок круга указывает, кто подает следующим. Нажатием на один из значков круга перед первой точкой определяется, кто начинает подачу.
- нажатием на названия команд можно изменять их.
- нажатием на одну из маленьких ячеек, чтобы использовать тайм-аут.
- используя меню синхронизации, можно синхронизировать табло с другими устройствами или веб-приложением.

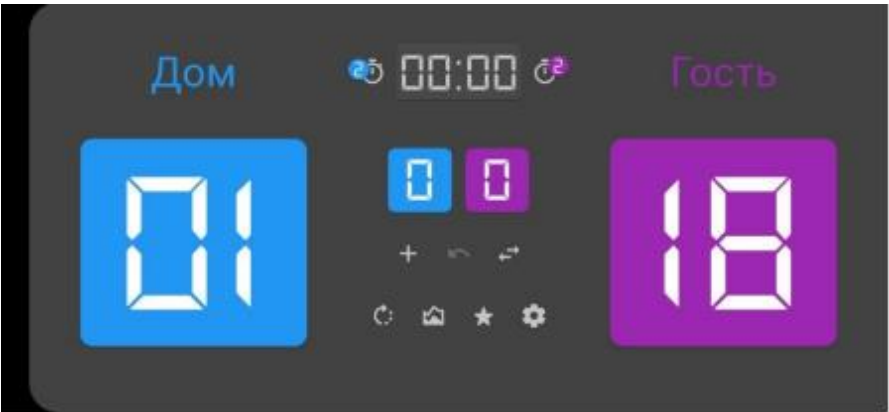
Можно делиться текущим счетом в любое время или результатами после окончания игры.

Минусами данной программы является:

- скучный интерфейс
- при назначении тайм-аутов можно изменять счет
- текущий счет или результат игры для отправки сообщением – просто текст с названием команд и счетом по партиям и т.д.

Volleyball scoreboard [2] - табло волейбол, простое, настраиваемое и с хронологией очков и ротаций. Данное приложение реализует почти все функции необходимые для комфортного проведения тренировок, но не официальных матчей. К основным функциям можно отнести:

- ведение счета текущей партии,
- ведение счета партий,
- проведение таймаутов,
- ведение хронологии очков,
- хранение данных о матчах.



А)



Б)

Рисунок 2 - скриншоты приложения «Volleyball scoreboard».

Плюсы:

- удобный интерфейс,
- ведение хронологии очков,
- возможность настройки матча.

Минусы:

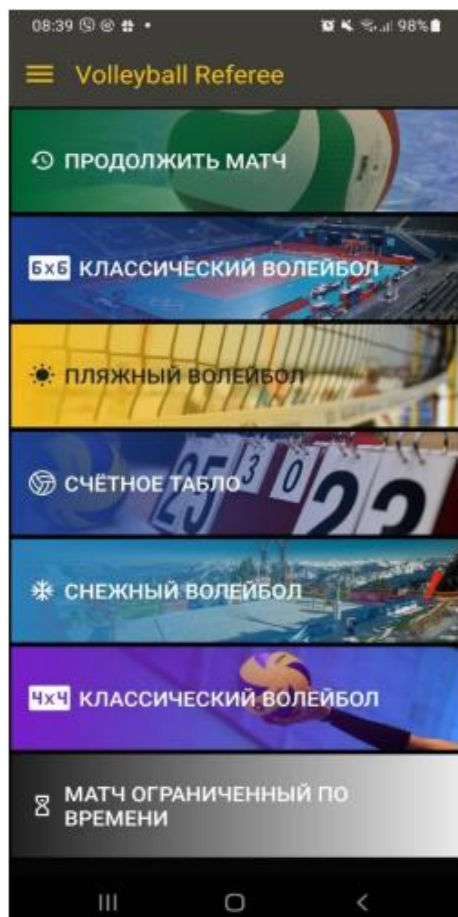
- при взятии таймаута пользователь может изменять очки матча,
- один вид матча,
- никак не отслеживается проведение замены,
- после окончания партии нет автоматической смены сторон.

Volleyball Referee - бесплатное приложение Volleyball Referee очень удобно для подсчитывания очков и ведения матчей по классическому и пляжному волейболу [3]. Приложение может быть полезно для волейбольных судей, тренеров, а также для ведения счёта во время любительских турниров.

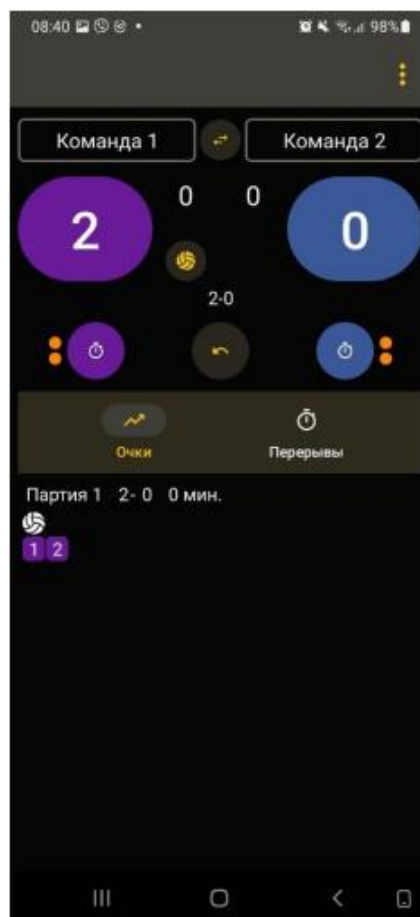
Данное приложение реализует почти все функции необходимые для комфортного проведения как официальных матчей, так и тренировочных. К основным функциям можно отнести:

- ведение счета текущей партии,
- ведение счета партий,
- проведение таймаутов,
- проведение замен,
- ведение хронологии очков,
- хранение данных о матчах,
- хранение данных о командах.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		12



А)



Б)

Рисунок 3 - скриншоты приложения «Volleyball Referee».

Плюсы:

- удобный интерфейс,
- большое количество типов матчей,
- возможность настройки матча,
- возможность ведения трансляции матча на сайте,
- ведение статистики матча,
- составление документа с результатами матча.

Минусы:

- нет ограничения по количеству игроков в команде,
- при взятии таймаута или замены пользователь может изменять очки матча,
- обязательный критерий для проведения официального матча – номера игроков.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		13

## 1.2 Извлечение требований

Для составления требований к разрабатываемой системе необходимо провести интервью с заказчиком (см. Приложение 2).

На основании интервью с заказчиком был составлен набор функциональных требований и нефункциональных требований.

Таблица 1 - функциональные и нефункциональные требования

Требования	
1.	Функциональные требования
1.1	Бизнес-требования
1.1.1	Разрабатываемая система призвана помочь судье в проведении матча
1.1.2	Организация возможности назначения матчей на определенные дни
1.1.3	Предусматривать возможность транслировать матч
1.2	Пользовательские требования
1.2.1	Организация официальных/тренировочных матчей
1.2.2	Назначение и проведение запланированных матчей на определенную дату
1.2.3	Получение готового протокола матча по его окончанию
1.3	Функциональные требования
1.3.1	Создание, редактирование и удаление матчей и команд
1.3.2	Предварительное завершение трансляции
1.3.3	Удаление пользователей Администратором
1.3.4	создание трансляции матча и комментариев
2	Нефункциональные
2.1	Интерфейс приложения и сайта должен быть интуитивно понятным, минималистичным
2.2	Периодичность обращения к серверу из приложения при транслировании матча – 5 минут

2.3	Создание тренировочного/официального матча из приложения возможно и без подключения к интернету
2.4	Начать запланированный матч можно только с подключением к интернету
2.5	Организация локальной базы данных для корректной работы при отсутствии подключения к интернету (периодичность обращения к серверу – при запуске приложения/авторизации)
2.6	Правила проведения матчей (тайм-аутов, замен) должны отвечать общепринятым правилам
2.7	Системные требования
2.7.1	Для работы с приложением: <ul style="list-style-type: none"> <li>- Операционная система: Android 5.1 и выше</li> <li>- Размер ОЗУ: от 2 ГБ</li> <li>- Подключение к интернету (рекомендуемое)</li> <li>- Размер встроенной памяти: от 8 ГБ</li> </ul>
2.7.2	Для работы с сайтом: <ul style="list-style-type: none"> <li>- Подключение к интернету</li> <li>- Устройства ввода/вывода</li> </ul>

### 1.3 Анализ требований

Проанализировав требования и предметную область, можно составить диаграмму прецедентов (см. рисунок 4), показывающая общий функционал разрабатываемой программы.

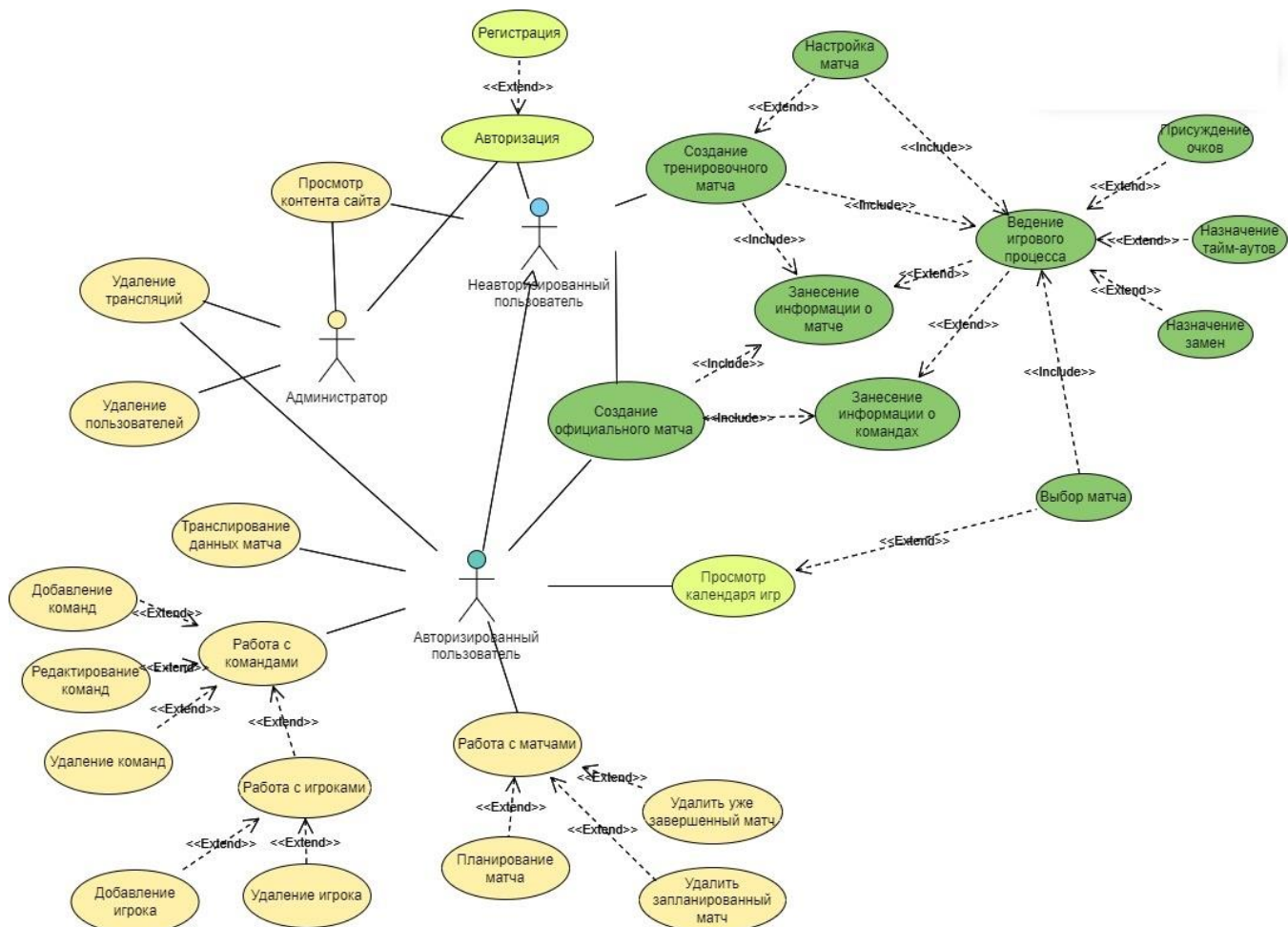


Рисунок 4 - Диаграмма прецедентов

В разрабатываемой системе предусмотрено несколько видов пользователей.

Актер «Не авторизируемый пользователь» - пользователь, не имеющий подключения к интернету или не авторизованный в системе. Данный актер может:

- создать тренировочный матч. Здесь пользователь вводит настройки матча и названия команд;
- создать официальный матч. Здесь пользователь вносит информацию о матче и командах;
- просматривать на сайте прямые трансляции.

Актер «Авторизируемый пользователь» - пользователь, имеющий подключения к интернету и авторизованный в системе. Данный актер имеет тот



же функционал что и «Не авторизируемый пользователь», но более расширенный, а именно:

- создавать официальный матч с заранее созданными командами на сайте;
- начинать заранее запланированные матчи;
- создание команд и матчей через сайт;
- транслирование матчей.

Актер «Администратор» просматривает трансляций и пользователей с возможностью их удаления.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		17

## 2. Разработка алгоритмов

### 2.1 Процессы системы

После того как мы определили основные типы данных необходимо составить диаграмму DFD (см. рисунок 5), позволяющую визуально показать все процессы, проходимые в разрабатываемой системе с точки зрения данных.

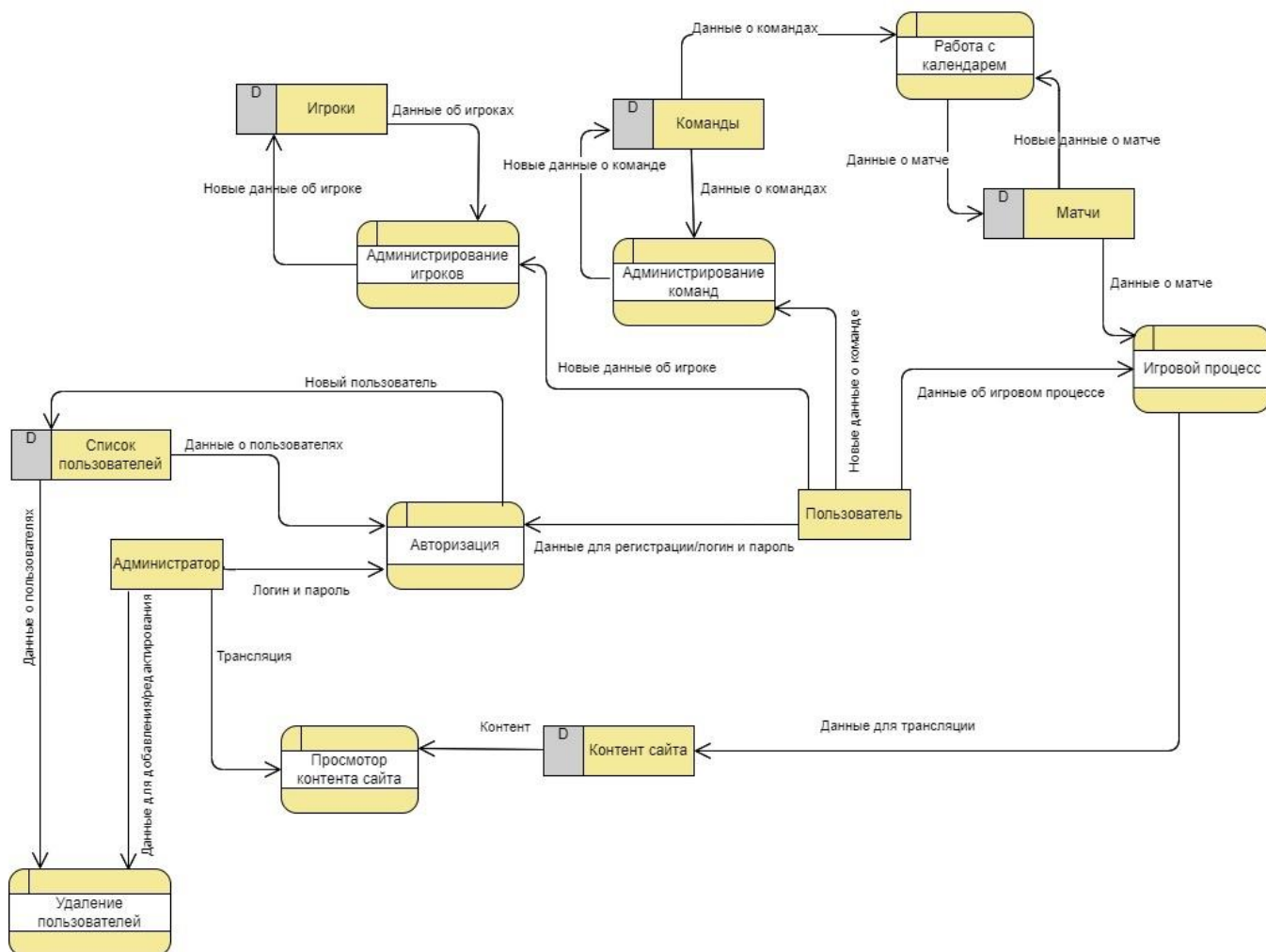


Рисунок 5 - диаграмма DFD

В процесс «Администрирования пользователей» удаление Администратором данных пользователей и трансляций.

Процесс «Просмотр контента сайта» осуществляет показ трансляций всем пользователям.

Под процессом «Администрирование команд» следует понимать добавление, изменение, удаление данных о командах и их игроках.

Процесс «Работа с календарем» включает в себя добавление матчей на определенный день и их отмену. В Игровом процессе происходит ведение матча, а именно присуждение очков командам, назначение тайм-аутов и замен.

После того как мы определили предметную область и основные процессы, выполняемые в разрабатываемой системе, необходимо определить из каких компонентов будет состоять разрабатываемая ИС.

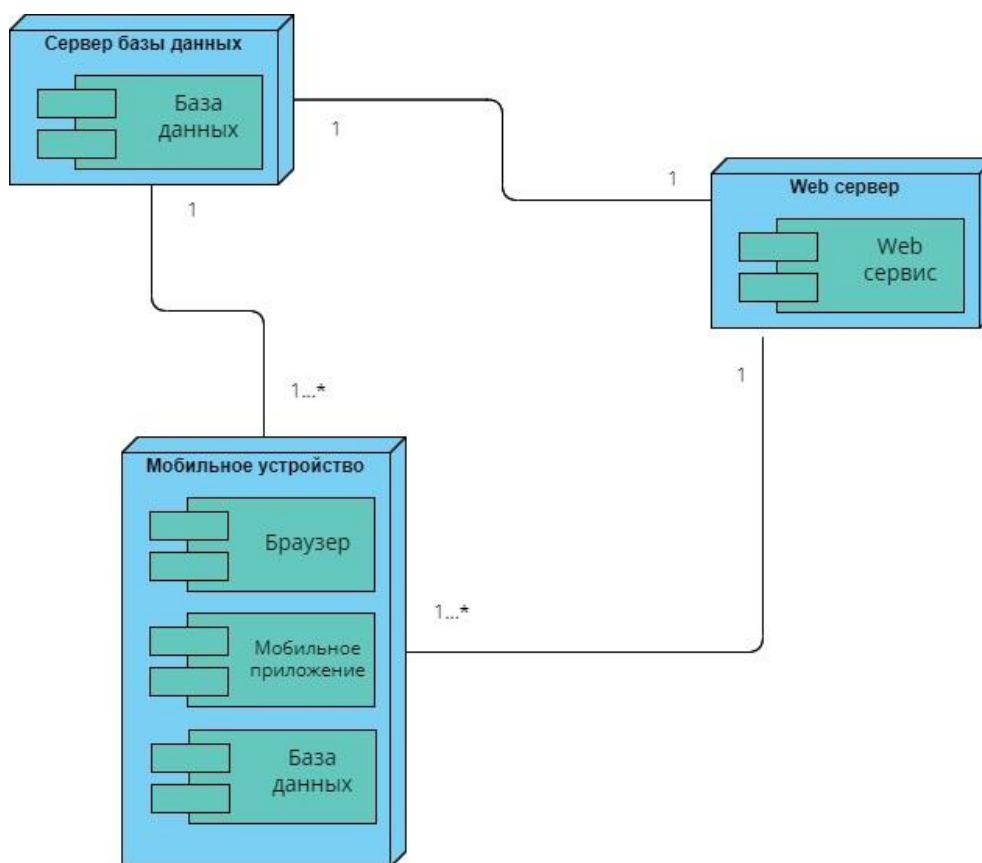


Рисунок 6 - диаграмма развертывания

На диаграмме, показанной на рисунке 6 представлены:

- сервер базы данных, где будут храниться все данные о пользователях, матчах и т.д.;

- Web сервер, предназначенный для организации взаимодействия сервера и пользовательского интерфейса для сайта, принимая запрос от пользователя отправляет его серверу и принимает данные по этому запросу;

- Мобильное устройство, которое может содержать в себе «Браузер» для работы с сайтом, само мобильное приложение и «Локальная база данных», необходимый для корректной работы приложения при отсутствии подключения к Интернету.

## 2.2 Моделирование данных

Моделирование данных — это создание визуального представления о всей информационной системе либо ее части. Цель в том, чтобы проиллюстрировать типы данных, которые используются и хранятся в системе, отношения между этими типами данных, способы группировки и организации данных, их форматы и атрибуты [4].

Концептуальная модель — это модель, представленная множеством понятий и связей между ними, определяющих смысловую структуру рассматриваемой предметной области или её конкретного объекта [5].

На рисунке 7 показана общая модель структуры системы.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		20

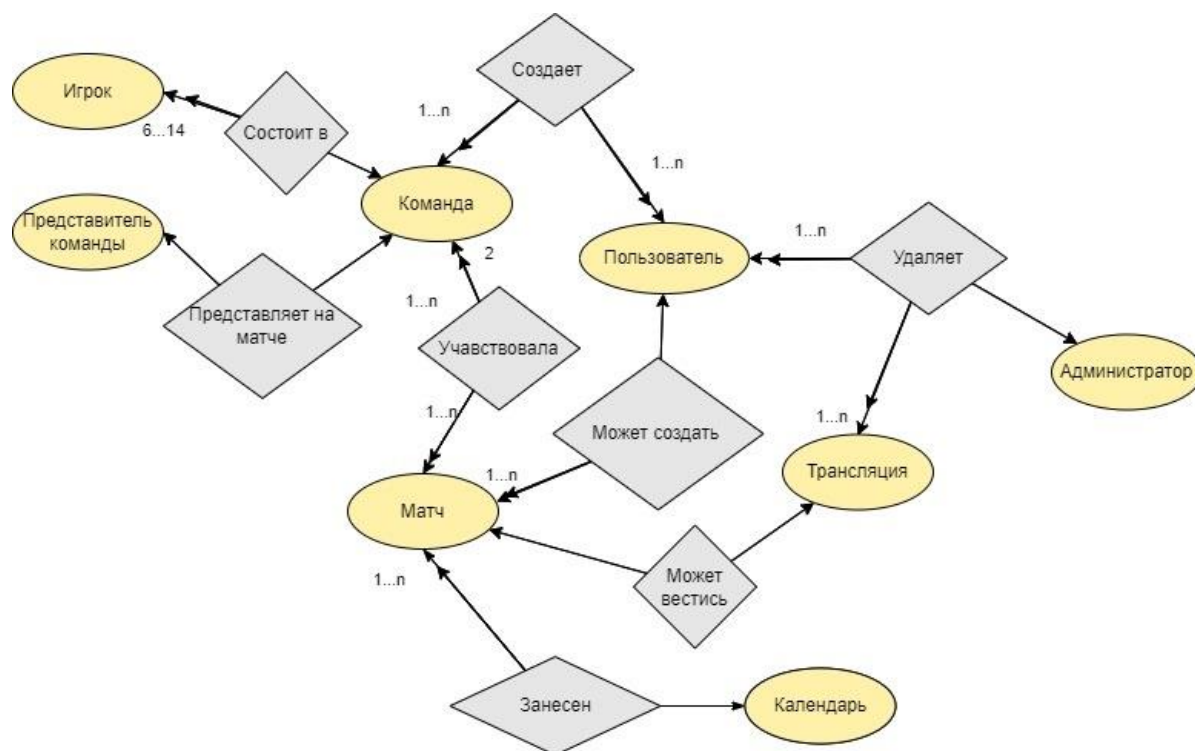


Рисунок 7 - Концептуальная модель

В разрабатываемой системе Администратор контролирует транслируемые матчи и регистрируемых пользователей.

Пользователь создает команды и их игроков. Может запланировать матч на определенную дату. При желании при ведении матча может транслировать его данные на сайт.

В команде для начала матча должно быть от 6 до 14 игроков считая Либера.

Для того что бы определить какие модели какие данные должны содержать необходимо составить логическую модель.

Логическая модель – графическое представление структуры базы данных с учетом принимаемой модели данных (иерархической, сетевой, реляционной и т.д.), независимое от конечной реализации базы данных и аппаратной платформы.

Иными словами, она показывает, что хранится в базе данных (объекты предметной области, их атрибуты и связи между ними), но не отвечает на вопрос как [6].

### 2.2.1 Моделирование серверной части

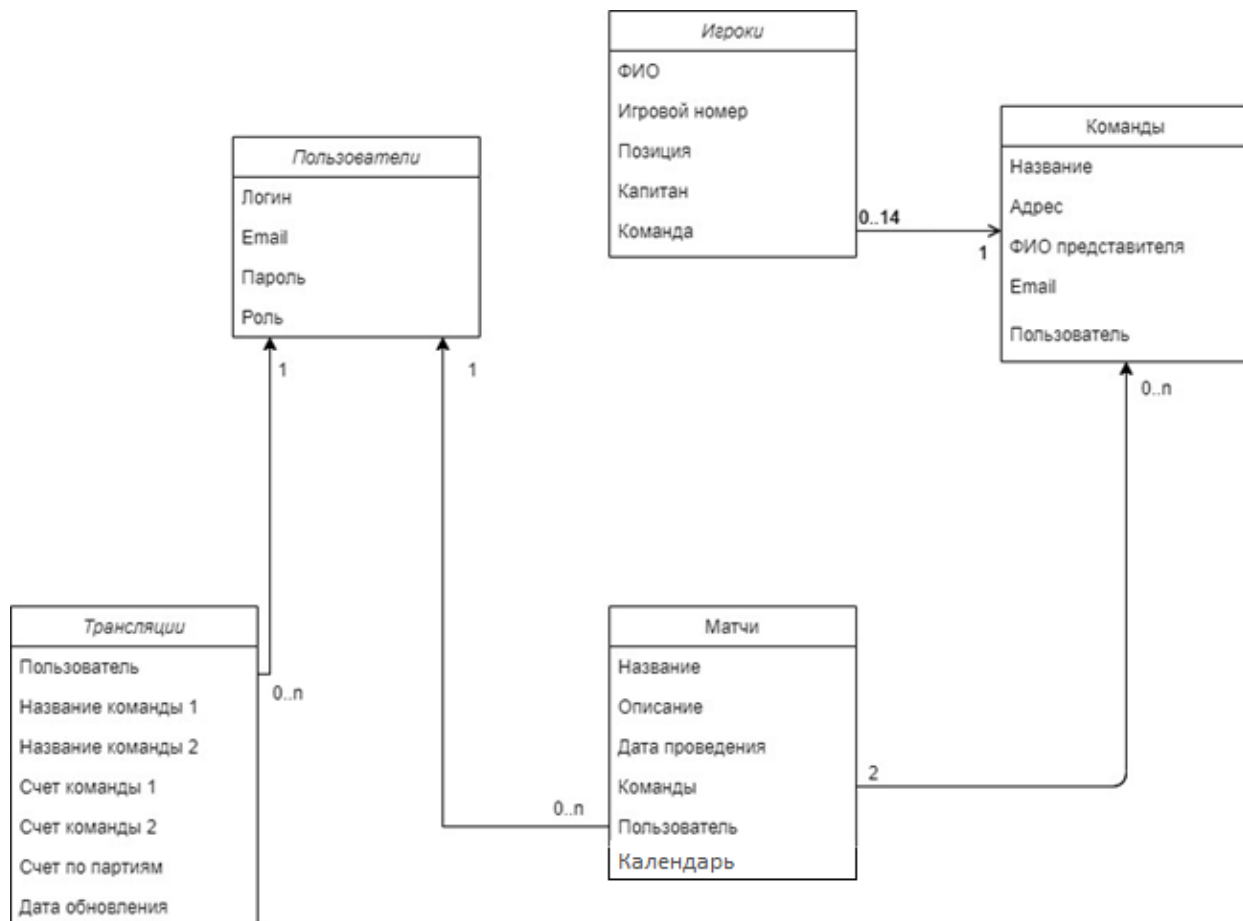


Рисунок 8 - логическая модель базы данных

На рисунке 8 показана логическая модель базы данных для сервера.

На основе нее создана физическая модель, содержащая все детали необходимые для создания таблиц (см. рисунок 9). В данной базе данных хранятся все данные пользователей, их команды, матчи и трансляции. Эта информация выводится на сайт.

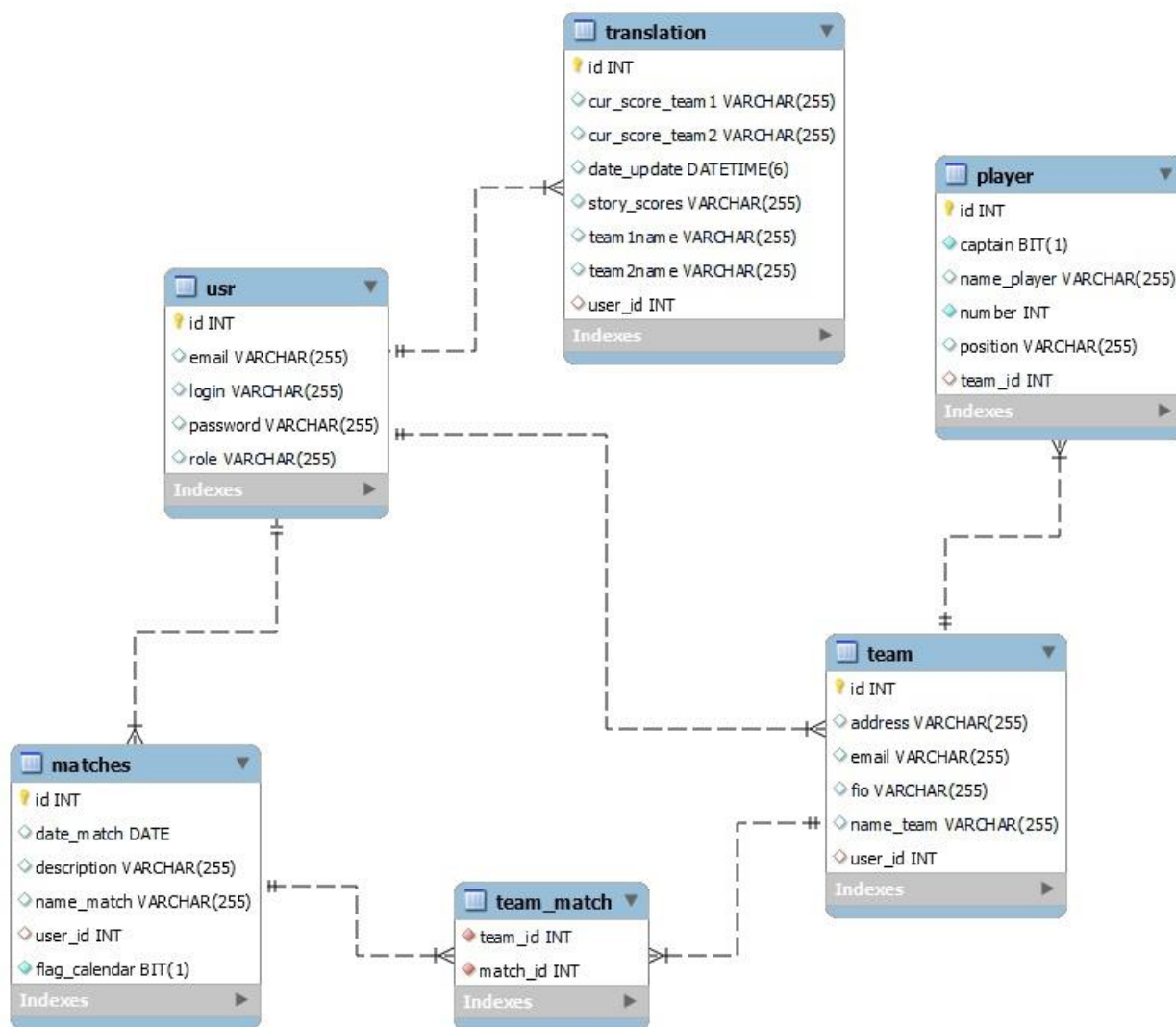


Рисунок 9 - физическая модель данных

Была создана физическая модель, для явного представления таблиц в базе данных. Значения полей и их типы данных по таблицам показаны в Таблицах 2-7.

Таблица 2 - данные пользователя

Поле	Значение поля	Тип данных
Id	Идентификатор	Int
Login	логин для доступа	varchar
password	пароль для доступа	Varchar
Email	электронная почта	Varchar
Role	роль пользователя	Varchar

Таблица 3 - данные о трансляциях

Поле	Значение поля	Тип данных
Id	Идентификатор	Int
Cur_score_team1	Очки команды 1 в текущей партии	Varchar
Cur_score_team1	Очки команды 2 в текущей партии	Varchar
Date_update	Дата последнего обновления	Varchar
Story_scores	История партий	Varchar
Team1name	Название команды 1	Varchar
Team2name	Название команды 2	Varchar
User_id	Пользователь, который ведет матч	Int

Таблица 4 - данные о командах

Поле	Значение поля	Тип данных
id	Идентификатор	Int
Address	Адрес команды	Varchar
Email	Электронная почта для связи	Varchar
Fio	Дата последнего обновления	varchar
Name_team	Название	Varchar
User_id	Пользователь, который добавил команду	Int

Таблица 5 - данные об игроках

Поле	Значение поля	Тип данных
Id	Идентификатор	Int
Captain	Флаг, является игрок капитаном или нет	Bit
Name_player	Имя игрока	Varchar
Number	Игровой номер	Int
Position	Позиция (Игрок, Либеро)	Varchar
Team_id	Идентификатор команды, в которой находится игрок	Int



Таблица 6 - данные о матче

Поле	Значение поля	Тип данных
Id	Идентификатор	Int
Date_match	Дата проведения матча	Date
Description	Описание	Varchar
Name_match	Название	Varchar
User_id	Пользователь, который ведет/будет вести матч	Int
Flag_calendar	Метка является ли матч запланированным (1) или сыгранным (0)	Bit

Таблица 7 – связь команд и матчей

Поле	Значение поля	Тип данных
Match_id	Идентификатор матча	Int
Team_id	Идентификатор команды	Int

### 2.2.2 Моделирование локальной базы данных



Рисунок 10 - Логическая модель для локальной базы данных

На рисунке 10 показана логическая модель для локальной базы данных на мобильном устройстве.

На основе нее создана физическая модель, содержащая все детали необходимые для создания таблиц (см. рисунок 11).

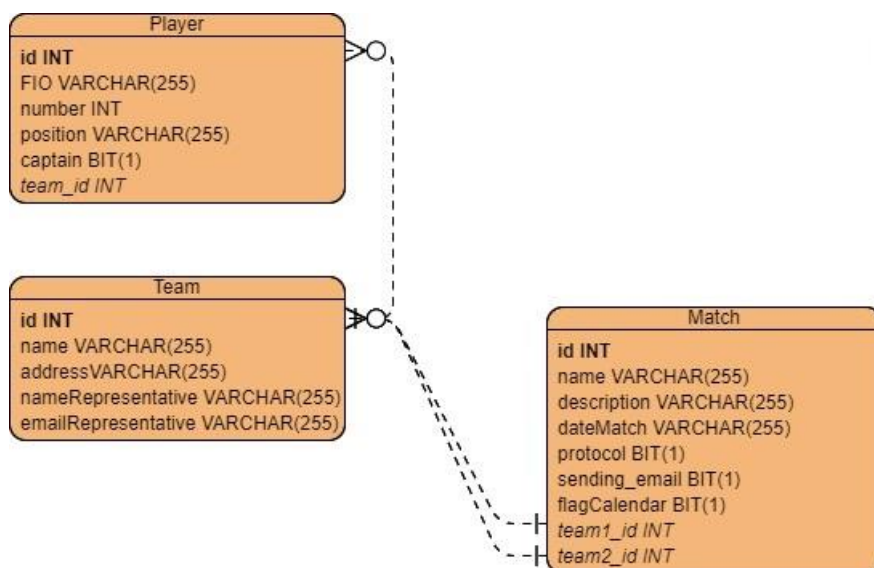


Рисунок 11 - физическая модель локальной базы данных

Была создана физическая модель, для явного представления таблиц в локальной базе данных для мобильного приложения.

В локальной БД хранятся данные пользователя мобильного приложения. А именно данные о командах, их игроках и матчах.

Таблица Team хранит данные о командах.

Поле id – идентификатор команды.

Поле name – наименование.

Поле address – адрес.

Поле nameRepresentative – имя представителя.

Поле emailRepresentative – электронная почта для связи.

Таблица Player хранит данные об игроках.

Поле id – идентификатор игрока.

Поле FIO – имя.

Поле number – игровой номер.

Поле position – позиция игрока.

Поле captain – флаг на капитана (является данный игрок им или нет).

Поле team\_id – идентификатор команды.

Таблица Match хранит данные о матчах.

Поле id – идентификатор игрока.

Поле name – название.

Поле description – описание.

Поле dateMatch – дата проведения.

Поле protocole – флаг на составление протокола (будет составляться протокол матча или нет).

Поле send\_email – флаг на отправку протокола представителям команд.

Поле flagCalendar – флаг на отправку протокола представителям команд.

Поле team1\_id – идентификатор первой команды.

Поле team2\_id – идентификатор второй команды.

## 2.3 Основные действия и сценарии использования

### 2.3.1 Выявление основных действий

Основными действиями пользователя в системе является:

#### 1) Проведение тренировочного матча

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		27

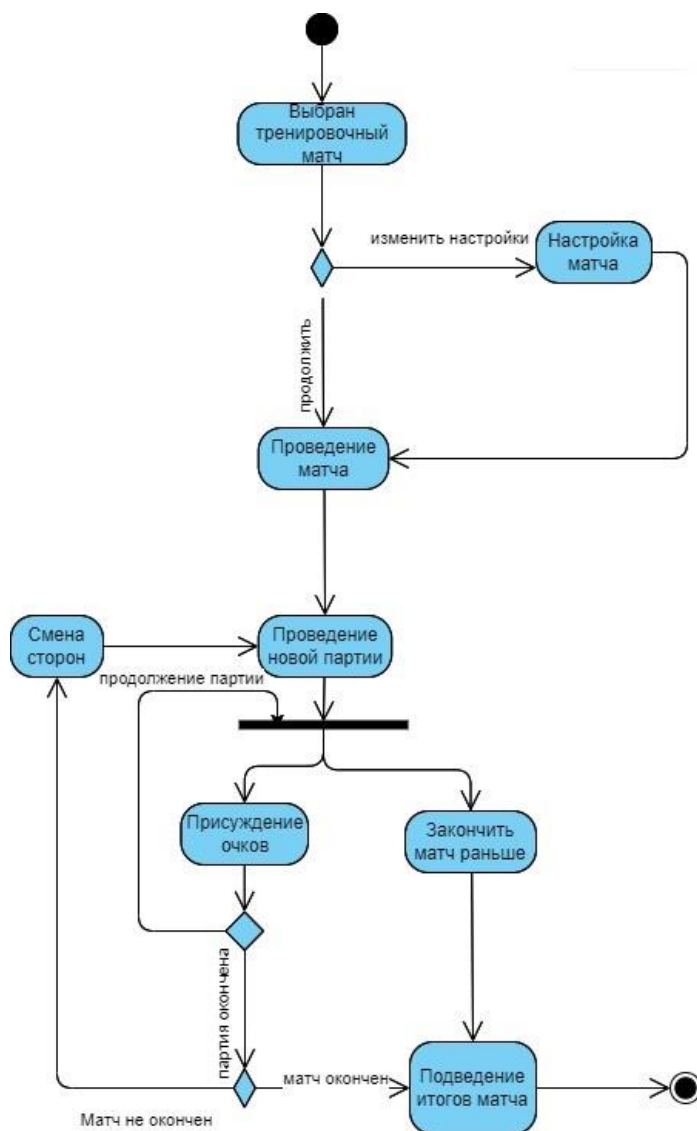


Рисунок 12 - диаграмма активности "Тренировочный матч"

Проведение тренировочных матчей начинается с его настройки. Если настройки, выбранные ранее, пользователя устраивают, то он переходит к началу матча, иначе переход в соответствующее меню и после сохранения настроек пользователь переходит к началу матча. Каждая новая партия сопровождается сменой сторон команд, записью результатов предыдущей партии и обновлением тайм-аутов и счета. Так же пользователь может в любое время заранее завершить матч, система подведет итоги матча (выберет победителя) и перейдет в главное меню.

## 2) Проведение официального матча

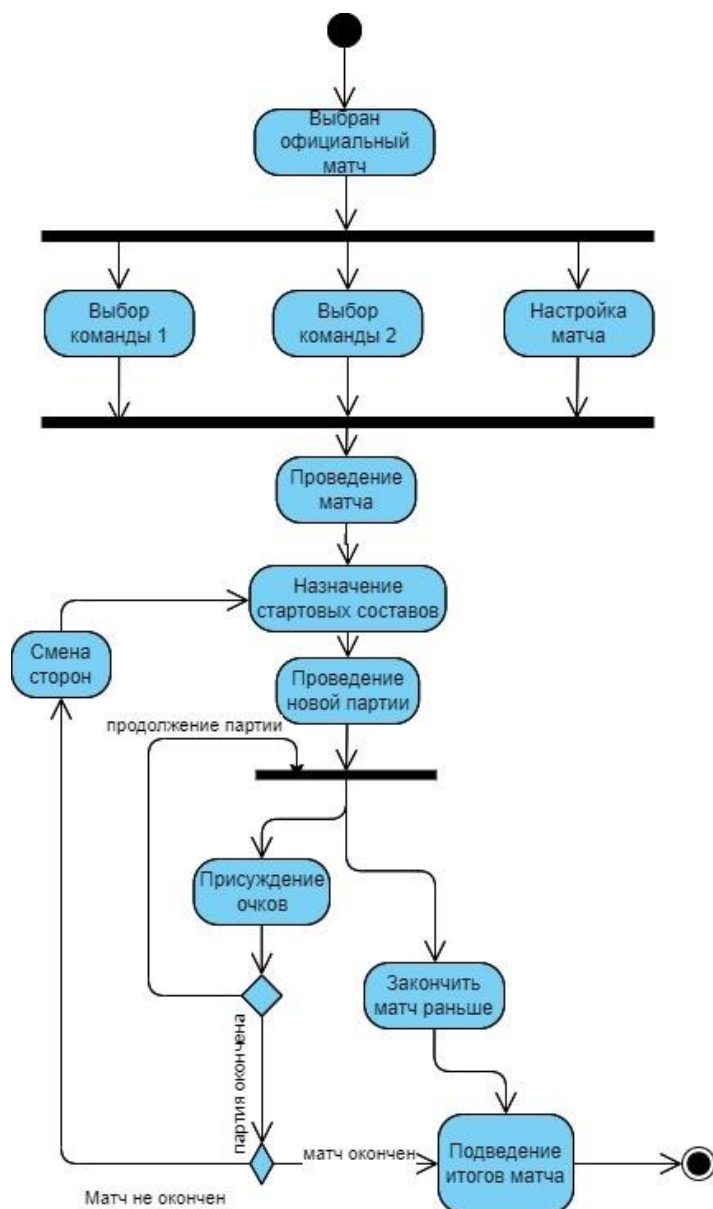


Рисунок 13 - диаграмма активности "Официальный матч"

Проведение официального матча начинается с выбора команд из уже имеющихся или добавлением новых и заполнением настроек матча. Только после этого система перейдет к проведению матча. Каждая новая партия сопровождается сменой сторон команд, записью стартовых составов, результатов предыдущей партии и обновлением тайм-аутов, замен и счета. Так же пользователь может в любое время заранее завершить матч, система подведет итоги матча и перейдет в главное меню. Данные матча будут сохранены в истории матчей пользователя.

### 3) Проведение запланированного матча

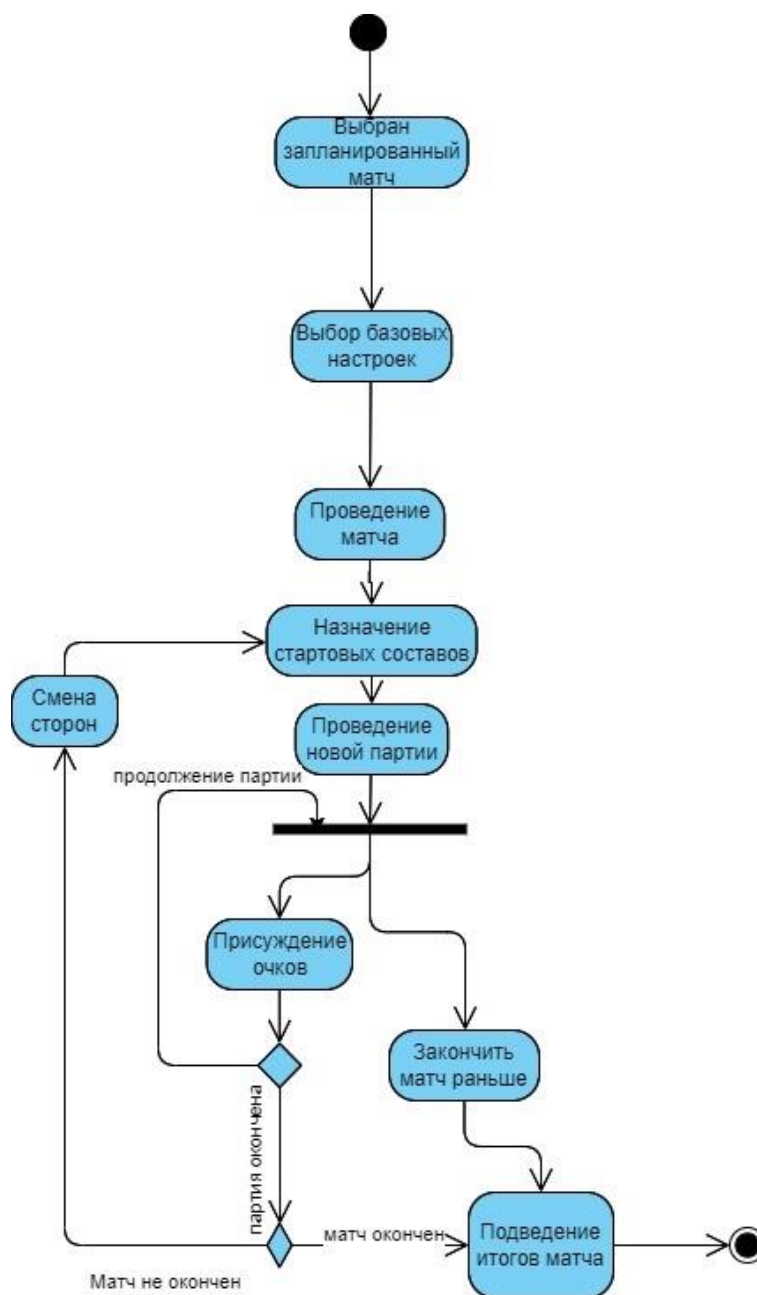


Рисунок 14 - диаграмма активности "Запланированный матч"

Проведение запланированного матча начинается заполнения базовый настроек (выбор количества партий, составлять ли протокол, вести ли трансляцию). Только после этого система перейдет к проведению матча. Каждая новая партия сопровождается сменой сторон команд, записью стартовых составов, результатов

предыдущей партии и обновлением тайм-аутов, замен и счета. Так же пользователь может в любое время заранее завершить матч, система подведет итоги матча и перейдет в главное меню. Данные матча будут сохранены в истории матчей пользователя.

#### 4) Проведение замен

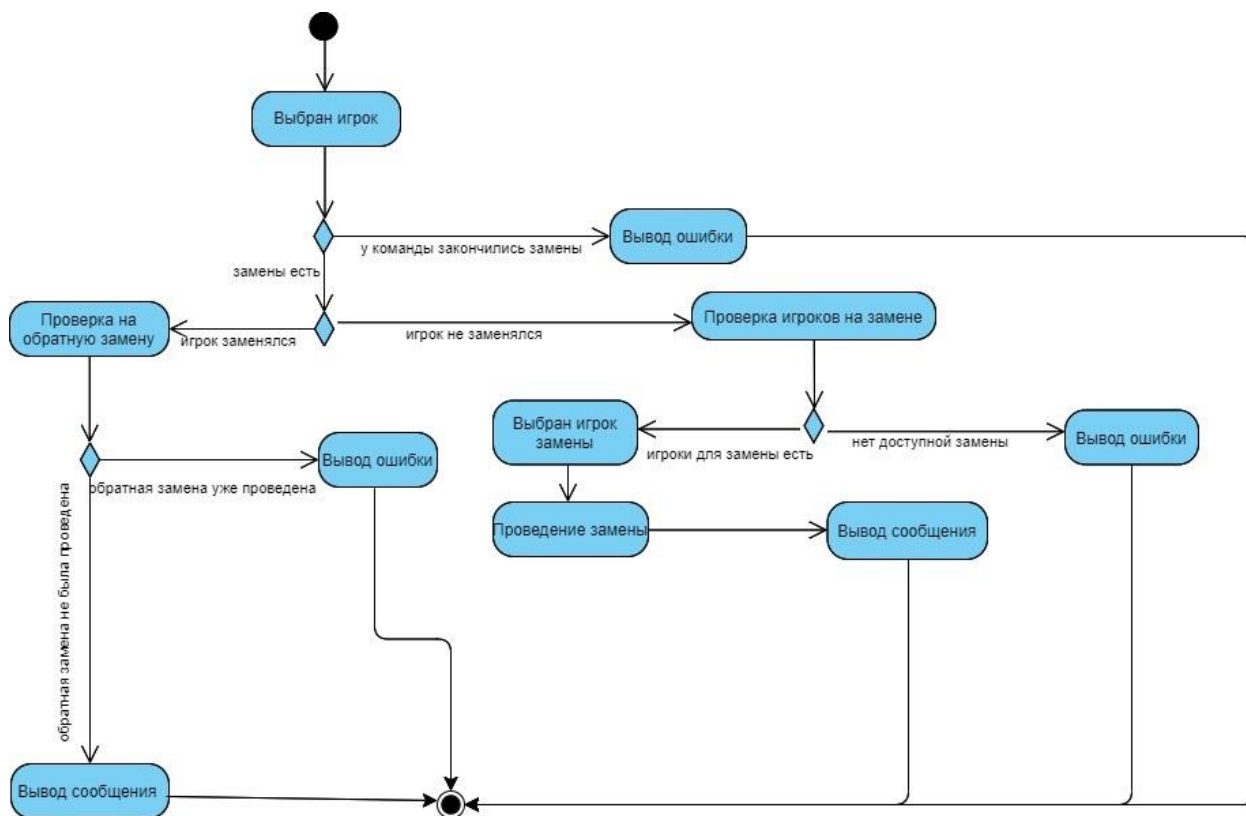


Рисунок 15 - диаграмма активности "Проведение замены"

Проведение замены требует выбора игрока, который будет заменяться. Проводятся необходимые проверки и по результату выводится либо ошибка замены игрока, либо сообщение об успешной замене.

#### 2.3.2 Основные сценарии использования

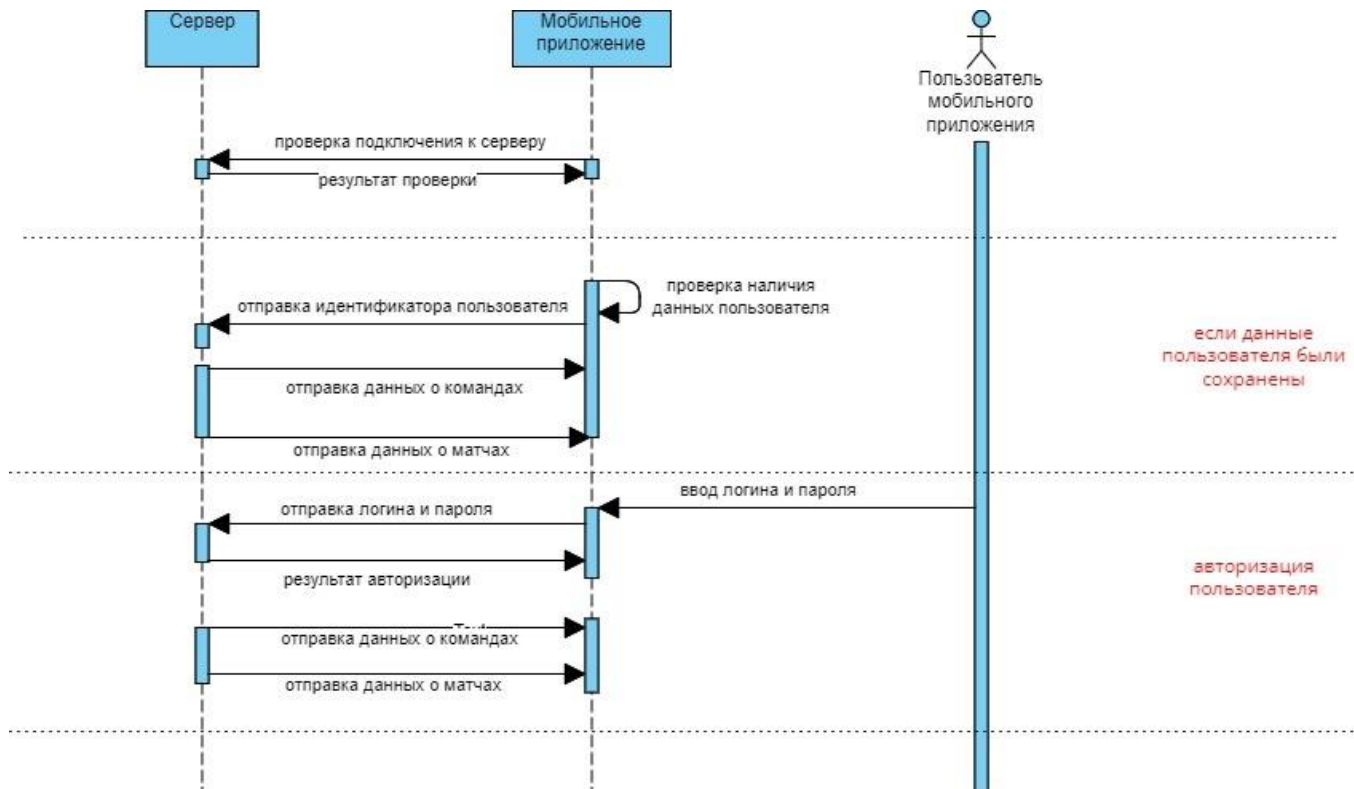


Рисунок 16 - диаграмма последовательности загрузки данных

Одним из основных сценариев использования мобильного приложения является получение данных о командах и матчах с сервера. При первом заходе в приложение система проверяет есть подключение к серверу и не был ли пользователь авторизован ранее. Если пользователь уже авторизовался ранее в системе, то загрузка данных происходит автоматически, иначе для их получения необходимо войти в систему.



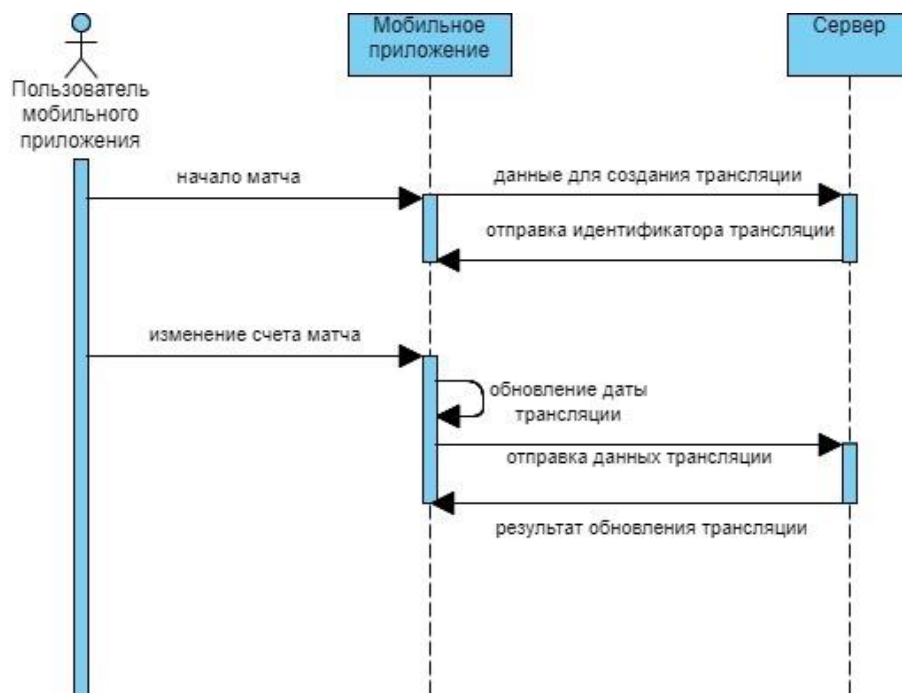


Рисунок 17 - диаграмма последовательности отправки данных матча

Возможным сценарием использования является передача данных о матче на сервер для их публикации, то есть транслирование матча.

### 3 Реализация системы

#### 3.1 Выбор и основание средств разработки

Для разработки мобильного приложения используется одна из наиболее популярных встраиваемых СУБД SQLite. Ее основные преимущества:

- открытость
- бесплатность
- отсутствие потребности в администрировании
- простая процедура подключения
- высокая производительность, при достаточном объеме ОЗУ
- возможность шифрования данных [7].

В качестве основной интегрированной среды разработки - Android Studio и язык программирования Kotlin.

Преимущества Android Studio:

- Приятный дизайнер пользовательских интерфейсов, позволяющий облегчить визуальное проектирование приложения
- Удобный XML редактор
- Поддержка системы контроля версий
- Эмуляция устройств
- Обширная база примеров проектирования
- Возможность проводить тестирование и анализ кода
- Скорость сборки приложения [8].

Язык программирования Kotlin имеет такие преимущества и предоставляет следующие возможности:

- коды открытые, внедрения бесплатные
- программы применяют фреймворки, библиотеки Джава
- доступны компиляции в байткод JVM, JavaScript
- процесс изучения простой
- уровень безопасности высокий
- преобразование Java в Kotlin происходит автоматически [9].

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		34

Для разработки WEB-сайта и WEB-сервера используется такой популярный фреймворк для создания веб-приложений с использованием Java, как Spring.

Преимуществами данного фреймворка является:

- быстрая и легкая разработка приложений на основе Spring.
- автоконфигурация всех компонентов для приложения Spring производственного уровня.
- HTTP end-points, позволяющие вводить внутренние функции приложения, такие как показатели, состояние здоровья и другие.
- легкий доступ к базам данных и службам очередей, таким как MySQL, Oracle и другим.

Недостатками:

- отсутствие контроля. Spring Boot создает множество неиспользуемых зависимостей, что приводит к большому размеру файла развертывания.
- сложный и трудоемкий процесс преобразования устаревшего или существующего проекта Spring в приложения Spring Boot [10].

В качестве СУБД используется одна из наиболее популярных - MySQL. Ее основные преимущества:

- Легко использовать,
- Предоставляет множество функций, связанных с базой данных,
- Легко масштабируется и подходит для больших баз данных,
- Обеспечивает хорошую скорость и производительность [11].

В качестве средства для разработки базы данных будет использоваться MySQL, а основная интегрированная среда разработки - IntelliJ IDEA и язык программирования Java.

Преимущества IntelliJ IDEA:

- Кроссплатформенность;
- Построение синтаксического дерева на лету;
- Мощный редактор кода [12].

Язык программирования Java имеет такие преимущества как:

- Независимый код.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		35

- Строгая статистическая типизация.
- Высокая функциональность.
- Синтаксис средней сложности [13].

## 3.2 Руководство программиста

### 3.2.1 Мобильное приложение

При разработке Android-приложения использовался такой шаблон как Model View View-Model, где Model – модель данных, View – представление, View-Model - модель представления, служащая прослойкой между View и Model.

#### 1) Работа с локальной базой данных

Для работы с локальной БД используется библиотека Room.

Такие классы как Match, Team, Player и User являются моделями таблиц локальной базы данных. TeamWithPlayers – модель объединяющая модель команды с ее игроками.

Dao – интерфейс, содержащий в себе все методы для работы с базой данных.

MainDB – основной класс по работе с БД наследуемый от RoomDatabase.

MyDBRepository – класс являющийся прослойкой между Dao и ViewAllData.

ViewAllData – класс реализующий логику работы с моделями базы данных при загрузке новых данных с сервера.

ViewTeam - класс реализующий логику работы с моделями базы данных во время работы приложения.

#### 2) Работа с сервером

Для реализации взаимодействия приложения с сервером используется библиотека Retrofit.

Класс RetrofitService – класс, реализующий получение и отправку данных через web-сервис.

Интерфейс MyApi – запросы на получение или отправку данных на сервер.

Такие классы как Translation, MatchToServer, TeamToServer, PlayerToServer и UserToServer являются моделями данных аналогичные отправляемым с сервера JSON.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		36

Класс DataModelAuth – содержит статус работы приложения, а именно результат соединения с сервером. Если соединение не установлено или установлено, но пользователь не авторизован, пользователь может только проводить тренировочные матчи и официальные, но без команд, хранящихся на сервере. Если соединение установлено, пользователь может авторизоваться в системе и получить данные с сервера.

### 3) Адаптеры

Класс MyPagerAdapter – реализация многооконного режима для создания матча. 1 окно – работа с командой 1, 2 – работа с командой 2, 3 – работа с данными о матче.

Класс MatchAdapter – организация вывода данных о матчах пользователя. Наследуется от RecyclerView.Adapter позволяющий вывести список объектов.

MatchForCalendarAdapter – организация вывода запланированных матчей пользователя с возможностью их проведения заполнив необходимые данные для начала матча.

PlayerAdapter - организация вывода игроков для выбора капитана команды.

TeamsAdapter - организация вывода команд пользователя и просмотра данных о них.

### 4) Обработка логики ведение матчей

TrainMatch – класс для ведения тренировочного матча.

- pointsToWin – числовое поле для максимального количества очков в партии
- setsToWin - числовое поле для максимального количества партий в матче
- setsAccounting – флаг на учет количества партий (если true, то ведется учет количества партий, иначе партий может быть не ограниченное количество)
- addPointTA – добавление очка для первой команды.
- deletePointTA – удаление очка для первой команды.
- addPointTB – добавление очка для второй команды.
- deletePointTB – удаление очка для второй команды.
- checkEndPart – проверка партии на окончание.
- checkEndMatch – проверка матча на окончание.

OfficialMatch – класс для контроля ведения официального матча.

- setsToWin - числовое поле для максимального количества партий в матче.

- setsAccounting – флаг на учет количества партий.

- timeBrakeAccounting – флаг на наличие тайм-брейка (специальная укороченная партия или дополнительный матч, для выявления победителя при ничейном счёте).

- pointsForTimeBrake – числовое поле для обозначения количества очков в тайм-брейке.

- teamTransition – флаг переход команды на площадке.

- flagTranslation – флаг на ведение трансляции.

- liveTranslation – экземпляр класса Translation для хранения данных трансляции матча для передачи на сервер.

- storyScoreSets – список результатов сыгранных партий.

- idTran – числовое поле для хранения результата транслирования матча, если результат не соответствует идентификатору трансляции, то транслирование матча прекращается.

- teamA – экземпляр класса TeamForTrainMatch для хранения данных о первой команде в матче.

- teamB – экземпляр класса TeamForTrainMatch для хранения данных о второй команде в матче.

- currentSetsChronPoint – экземпляр класса ChronologyPoints для хранения хронологии очков по текущей партии.

- chronologyPoints – список экземпляров ChronologyPoints для хранения хронологий очков по партиям.

- numSet – числовое поле для номера текущей партии.

- addPointTA – добавление очка для первой команды.

- deletePointTA – удаление очка для первой команды.

- addPointTB – добавление очка для второй команды.

- deletePointTB – удаление очка для второй команды.

- checkEndPart – проверка партии на окончание.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		38

- checkEndMatch – проверка матча на окончание.
- nameTeamForSides – метод для получения площадки по названия команды.
- getChronologyPoints – метод для получения списка хронологии очков без текущей партии.

- insertLiveTranslation – метод для создания трансляции на сервере.
- setLiveTranslation – метод для актуализации трансляции на сервере.

ChronologyPoints – класс для хранения хронологии очков в партии.

- chronologyPoint – список имен команд по хронологии заработанных очков.
- numSet – номер партии.

#### 5) Activity-и и Fragment-ы

CalendarFragment – фрагмент для вывода данных о запланированных матчах.

DataMatchModel – класс унаследованный от ViewModel для хранения данных о командах и матче для текущего матча.

TeamForMatchFragment – фрагмент для обработки и вывода выбранной команды для матча.

- db – экземпляр класса MainDB для создания связи с локальной базой данных.
- repository – экземпляр класса MyDBRepository для работы с локальной БД.
- viewTeam - экземпляр класса ViewTeam для работы с локальной БД.
- listTeamWithPlayers – список экземпляров класса TeamWithPlayers для хранения команд пользователя с их игроками.

- team - экземпляр класса Team для хранения выбранной команды
- namesTeams – список названий команд для выпадающего списка
- currentTeam - экземпляр класса TeamWithPlayers для хранения выбранной команды и ее игроков

- selectedTeam - экземпляр класса TeamWithPlayers для хранения выбранной команды из списка.

- dataModel - экземпляр класса DataMatchModel для хранения данных о команде в рамках Activity.

- numTeam – номер команды (1 – первая, 2 - вторая).

- positions – список позиций игроков (Игрок, Либеро).

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		39

- spinnerAdapter - экземпляр класса SpinnerAdapter для выпадающего списка.
- playerAdapter - экземпляр класса PlayerAdapter для вывода списка игроков для выбора капитана.

- recyclerViewForPlayers - экземпляр класса RecyclerView для вывода списка игроков.

InfAboutMatchFragment – фрагмент для вывода и занесения данных о матче.

- status - статус работы (регистрация/профиль).

При инициализации по статусу работы меняется функционал.

- checkAllFields - проверка вводимых данных.

AutorizationFragment - фрагмент для авторизации пользователя в приложении.

- userToServer - экземпляр класса UserToServer для отправки данных на сервер.

- teamsToServer - список экземпляров класса TeamToServer для получения данных о командах пользователя.

- matchToServer - список экземпляров класса MatchToServer для получения данных о командах пользователя.

- login - отправка логина и пароля на сервер и получение результата авторизации.

- getApiUser - получение данных пользователя с сервера по идентификатору.

- startWork - загрузка другого Activity.

- loadDataInDB - загрузка данных в локальную БД.

- loadTeams - загрузка команд пользователя с сервера в локальную БД.

- loadMatches - загрузка матчей пользователя с сервера в локальную БД.

- loadPlayers - загрузка игроков пользователя с сервера в локальную БД.

- checkDataUser - проверка ввода данных для авторизации.

LoginActivity - страница для авторизации пользователя.

Main2Activity - главная страница приложения.

- connectToServer - проверка подключения к серверу.

- connectToServer1 - второй этап проверка подключения к серверу.

- loginUser - авторизация пользователя.

- loadFragment - загрузка боковой панели.

					МИВУ.09.03.04-12.000 ПЗ	Лист
						40
Изм	Лист	№ докум.	Подп.	Дата		



- loadUser - загрузка данных пользователя.
  - loadTeams - загрузка команд пользователя с сервера в локальную БД.
  - loadMatches - загрузка матчей пользователя с сервера в локальную БД.
  - loadPlayers - загрузка игроков пользователя с сервера в локальную БД.
- MainPageFragment - фрагмент главной страницы.
- goToTrainMatch - обработка нажатия кнопки для перехода в тренировочный матч.
  - loadSettings - загрузка диалогового окна с настройками данных для тренировочного матча.
  - startVisibleFields - метод задает видимые поля.
  - startValueFields - стартовые значения для полей.
  - setVisibleForSets - метод задает видимые поля, относящиеся к партиям.
  - setVisibleForTimeBrake - метод задает видимые поля, относящиеся к тайм-брейкам.
  - setVisibleForTimeOut - метод задает видимые поля, относящиеся к тайм-аутам.
- MatchActivity - страница для создания матча.
- match - экземпляр класса Match для хранения данных о матче.
  - listPlayerTeam1 - список экземпляров класса Player для хранения данных об игроках первой команды текущего матча.
  - team1 - экземпляр класса Team для хранения данных о первой команде.
  - listPlayerTeam2 - список экземпляров класса Player для хранения данных об игроках второй команды текущего матча.
  - team2 - экземпляр класса Team для хранения данных о второй команде.
  - arr - список вариантов количества партий.
  - countSets - количество партий.
  - transition - флаг на ведение трансляции.
  - choosingTeamServeBall - загрузка диалогового окна для выбора команды, которая начнет подавать.
  - startMatch - загрузка страницы для ведения матча.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		41

- checkDataForMatch - проверка данных о матче.
- checkDataForTeam - проверка данных о команде.
- checkCountPlayers - проверка данных об игроках.
- StartMatchActivity - страница для ведения официального матча.
- observer - загрузка данных о матче и командах с игроками.
- checkDetailsCurMatch - проверка деталей матча.
- startWork - загрузка фрагментов для ведения матча.
- onTransitionBack - переход назад.
- onReplacementPlayer - обработка нажатия кнопки замены выбранного игрока.
- replacementPlayer - определение команды, в которой проводится замена.
- loadDialogviewForReplacement - загрузка диалогового окна для замены игрока.
- appointmentStartingLineUp - занесение стартовых составов игроков.
- loadDialogview - загрузка диалогового окна для стартовых составов.
- appointmentPlayerToZone - загрузка составов занен по командам.
- addBtnForAppointmentPlayer - добавление кнопки обозначающей игрока из замены.
- teamTransition - переход (смена зон игроков) команды.
- teamTransitionBack - переход назад.
- initPlayerInPlace - заполнение полей игроков.
- checkEndPart - проверка конца партии.
- endMatch - конец матча.
- createProcole - создание протокола.
- sendEmails - отправка протокола матча на почту.
- changeOfSides - смена сторон.
- changeOfSidesPlayers - смена сторон по зонам.
- goToNewFragment - загрузка фрагментов по номеру партии.
- imageBall - выбор какой мяч показывать на экране.
- startTime - время матча.
- onSvist - обработка кнопки для запуска аудиозаписи свистка.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		42

- interrupting\_match - предварительное окончание матча.

TeamInMatchFragment - фрагмент для ведения счета и тайм-аутов для команды для официального матча.

- numTeam - номер команды.

- nameTeam - название команды.

- countsets - количество партий.

- onClick - обработка нажатия кнопок.

- updateModel - обновление данных о команде.

- managmentTimeOut - загрузка диалогового окна для тайм-аута.

- startTimer - запуск таймера для тайм-аута.

TMTeamAFragment - фрагмент для ведения счета и тайм-аутов для команды для тренировочного матча.

TrainMatchActivity - страница для ведения тренировочного матча.

- listPointsSetsTeamA - список очков для первой команды.

- listPointsSetsTeamB - список очков для второй команды.

- pointTeamA - текущие очки для первой команды.

- pointTeamB - текущие очки для первой команды.

- flagTimeBrakeTwo - флаг на переход в тайм-брейке.

- visibleBtnTimeOut - ограничение тайм-аутов.

- imageBall - выбор какой мяч показывать на экране.

- startTime - время матча.

- onSvist - обработка кнопки для запуска аудиозаписи свистка.

- goToNewFragment - загрузка фрагментов.

- onTimeOutTeam - обработка нажатия кнопок для тайм-аутов.

- onScoreTeam - обработка нажатия кнопок для добавления очков команд.

- onMinusScoreTeam - обработка нажатия кнопок для отката очков команд.

- startTimer - запуск таймера для тайм-аута.

- changeOfSides - смена сторон.

- endMatch - конец матча.

- updateBtnTimeOut - обновление кнопок тайм-аутов.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		43

- columnsForChronologyPoints - добавление колонок для вывода хронологии очков.

- onChronologyPoints - обработка нажатия кнопки для вывода хронологии очков.

- createFieldsGridLayout - создание полей для очков.

- addGridLayout - вывод панели хронологии очков.

### 3.2.2 Web-сервис

#### 1) Авторизация

Для организации аутентификации и авторизации использовалась такая библиотека как Spring Security [14].

CustomUserDetailsService - класс унаследованный от UserDetailsService для проверки пользователя.

- loadUserByUsername - поиск пользователя по логину.

WebSecurityConfiguration — класс, расширяющий WebSecurityConfigurerAdapter для авторизации пользователя в системе.

- publicPages - список страниц доступных не авторизируемым пользователям.

- passwordEncoder - метод для кодирования пароля пользователя.

- userDetailsService - экземпляр класса CustomUserDetailsService для проверки пользователя.

- configure(HttpSecurity http) - реализация логики авторизации пользователей и доступа пользователей к страницам.

- configure(WebSecurity web) - реализация игнорирования определенных страниц для всех пользователей.

- configure(AuthenticationManagerBuilder auth) - метод для настройки userDetails.

Класс User унаследованный от UserDetails

- setUser - заполнение полей для обычного пользователя.

- setAdmin - заполнение полей для администратора пользователя.

Role - перечисляемый тип ролей пользователей.

#### 2) Модели

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		44

Такие классы как Translation, Matches, Player, Representative, Team аналогичны таблицам в базе данных.

MatchRepository – интерфейс, расширяющий CrudRepository для работы с матчами.

PlayerRepository – интерфейс, расширяющий CrudRepository для работы с игроками.

TeamRepository – интерфейс, расширяющий CrudRepository для работы с командами.

TransactionRepository – интерфейс, расширяющий CrudRepository для работы с трансляциями.

UserRepository – интерфейс, расширяющий CrudRepository для работы с пользователями.

- checkLogin - проверка индивидуальности логина.

- checkEmail - проверка индивидуальности email-ла.

UserService - класс сервиса для работы с пользователями.

UserTeamsService - класс сервиса для работы с данными о командах, игроках, матчах и трансляциях.

### 3) Контроллеры

AdminController - класс-контроллер для страниц доступных Администратору.

- users - вывод пользователей системы.

- delete - удаление выбранного пользователя.

PublicController - класс-контроллер для страниц доступных всех пользователей.

- home - вывод главной страницы.

- profile - вывод\изменение данных пользователя.

- registration - регистрация пользователя.

- translation - вывод всех трансляции.

ServerController - класс-контроллер для работы с мобильными приложениями.

UserController - класс-контроллер для страниц доступных обычным пользователям.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		45

- setModelsForTeams - метод для получения модели для вывода на страницы.
- getrlayers - перепись списка игроков.
- translationDelete - удаление выбранной трансляции.
- teams - вывод команд пользователя.
- team - вывод данных по выбранной команде.
- teamSave - сохранение данных команды.
- addTeam - добавление команды.
- deleteTeam - удаление команды.
- deletePlayer - удаление игрока.
- addPlayer - добавление игрока.
- addMatch – добавление матча.

#### 4) Вывод данных

MatchDto - класс для добавления запланированного матча.

- id - идентификатор матча.
- nameMatch - название матча.
- description - описание матча.
- dateMatch - дата матча.
- now - текущая дата.
- team1id - идентификатор первой команды.
- team2id - идентификатор второй команды.
- user - экземпляр класса User для хранения данных пользователя.

ProfileDto - класс для работы с профилем.

- id - идентификатор пользователя.
- login - логин.
- old\_password - старый пароль.
- email - электронная почта.
- new\_password - новый пароль.

TeamsDto - класс для выпадающего списка команд.

- name - название команды.
- id - идентификатор.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		46

PositionsPlayers - класс для выпадающего списка позиций игроков.

#### 5) Планировщик

CalendarMatch - интерфейс для проверки запланированных матчей.

CalendarMatchImpl - сервис унаследованный от CalendarMatch для проверки актуальности запланированных матчей.

DeleteTranslation - интерфейс для проверки трансляций.

DeleteTranslationImpl - сервис унаследованный от DeleteTranslation для проверки актуальности трансляций.

TranslationsScheduler - класс планирования проверок актуальностей трансляций и запланированных матчей.

- checkTranslations - проверка актуальности трансляций каждые 5 минут.

- calendarMatch - проверка актуальности запланированных матчей каждые 12 часов.

#### 6) HTML-файлы

users.html - вывод пользователей.

footer.html - нижняя часть html-страницы.

head.html - верхняя часть html-страницы.

login.html - авторизация пользователей.

add.html - добавление команды.

infteam.html - вывод информации о команде.

teams.html - команды пользователя.

translations.html - трансляции.

about.html - html-страница "О нас".

home.html - html-страница "Главная".

profile.html - данные пользователя.

registration.html - регистрация пользователя.

					МИВУ.09.03.04-12.000 ПЗ	Лист
						47
Изм	Лист	№ докум.	Подп.	Дата		

## 4 Руководство пользователя

### 4.1 Мобильное приложение

#### 4.1.1 Не авторизованный пользователь

Если пользователь видит сообщение «Нет доступа к серверу или подключения к интернету», то приложение находится в так называемом “автономном” режиме. Пользователь может проводить тренировочные или официальные матчи нажав на соответствующие кнопки главного экрана (см. Приложение 3 рисунок 1).

##### 1) Тренировочный матч

Перед тем как начать тренировочный матч на экране появится окно с настройками (см. рисунок 18).

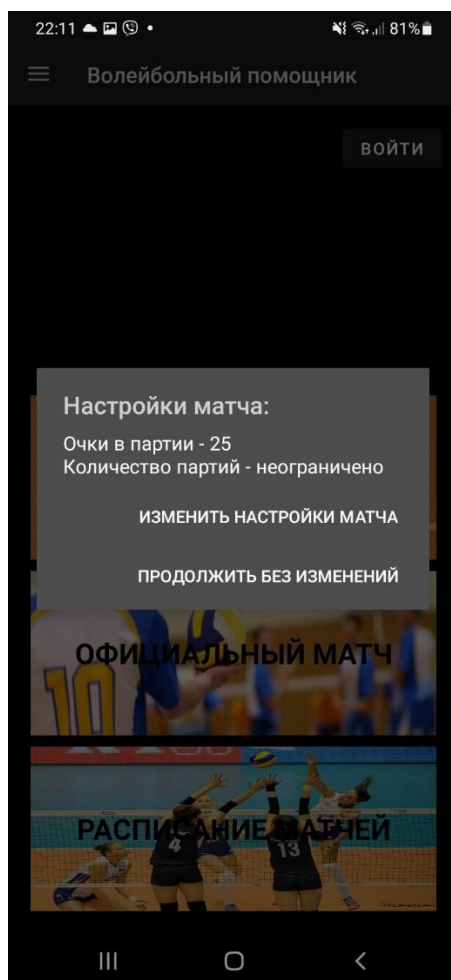


Рисунок 18 - окно проверки настроек тренировочного матча

					МИВУ.09.03.04-12.000 ПЗ	Лист
						48
Изм	Лист	№ докум.	Подп.	Дата		



Для того что бы изменить их необходимо нажать кнопку «Изменить настройки матча». При нажатии кнопки «Продолжить без изменений» начнется тренировочный матч (см. Приложение 3 рисунок 2).

Для добавления очков необходимо нажать большую кнопку в соответствующей стороне команды (см. Приложение 3 рисунок 3).

Для удаления очков необходимо нажать кнопку «-1» в соответствующей стороне команды (см. Приложение 3 рисунок 4).

Для назначения тайм-аутов необходимо нажать кнопку «Т» в соответствующей стороне команды (см. Приложение 3 рисунок 5).

Для просмотра хронологии очков необходимо нажать кнопку «Хронология очков» (см. Приложение 3 рисунок 6-7).

На экране показано меню в верхнем правом углу (см. Приложение 3 рисунок 8). Для того что бы провести жеребьевку в меню нужно выбрать соответствующий пункт. Если жеребьевка проводится в начале партии, то появится сообщение с названием подающей команды. Если нет, появится соответствующее сообщение (см. Приложение 3 рисунок 9).

## 2) Официальный матч

При выборе данного матча необходимо создать команды и заполнить необходимые данные матча.

Для создания новой команды необходимо в вкладке «Хозяева» / «Гости» заполнить название, имя представителя команды и больше 6, но меньше 14 игроков (Либеро – игроком не считается) (см. рисунок 19).

Рисунок 19 - форма для занесения данных о команде

Если для какой-либо команды это условие не выполнено, то система выдаст сообщение об соответствующей ошибке. Пример - Приложение 3 рисунок 10.

Для добавления игрока необходимо нажать кнопку находящаяся под полем адреса (см. Приложение 3 рисунок 11). Появится окно, где необходимо задать имя игрока, позицию и индивидуальный игровой номер (см. Приложение 3 рисунок 12).

Так же необходимо назначить капитана команды нажав на соответствующую кнопку. Появится окно со списком игроков, при нажатии на маркер выбранного игрока он будет назначен капитаном. Пример окна показан на рисунке 20.

					МИВУ.09.03.04-12.000 ПЗ	Лист
						50
Изм	Лист	№ докум.	Подп.	Дата		



Рисунок 20 - форма выбора капитана

После занесения данных о командах необходимо ввести название матча, выбрать количество партий (по умолчанию 3 партии), выбрать будет ли создаваться протокол, и будет ли вестись трансляция матча.

Ведение официального матча похоже на тренировочный, только названия команд уже занесено и нет возможности их менять. Так же добавляется возможность слежения за расстановками команд и назначения стартовых составов и замен.

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		51

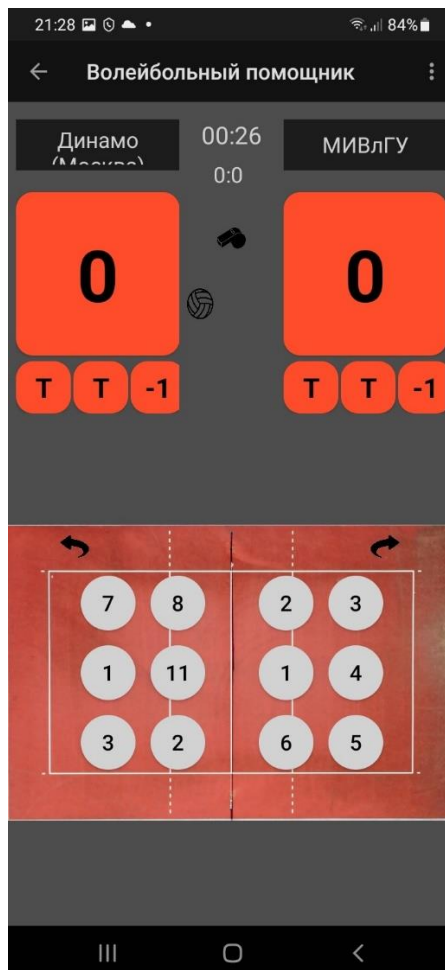


Рисунок 21 - форма проведения официального матча

Перед тем как перейти к ведению матча необходимо заполнить стартовые составы команд. Форма для занесения стартового состава команды показала на рисунке 22.

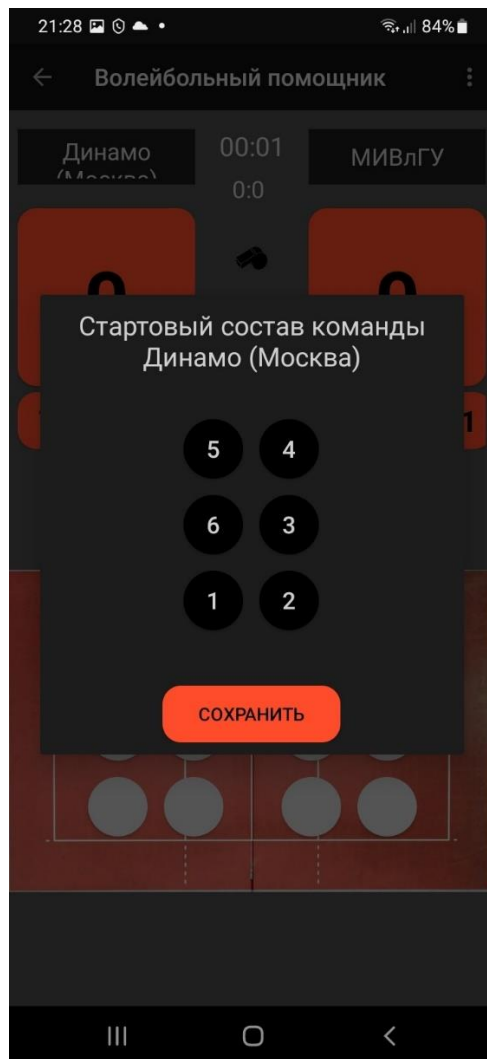


Рисунок 22 - форма занесения стартового состава

При нажатии зоны появится окно для выбора игрока (см. рисунок 23) команды (кнопки белого цвета – игроки доступные для выбора).

					МИВУ.09.03.04-12.000 ПЗ	Лист
						53
Изм	Лист	№ докум.	Подп.	Дата		

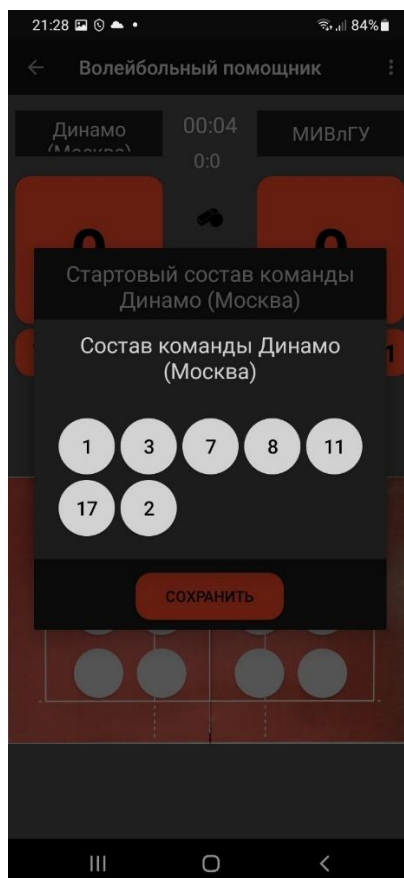


Рисунок 23 - форма выбора игрока из состава

Для утверждения стартового состава надо нажать кнопку «Сохранить». Если хотя бы одна зона не заполнена система выдаст соответствующее сообщение, иначе текущее окно закрывается и открывается окно для занесения состава для второй команды. Если стартовый состав второй команды занесен, то приложение переходит к ведению матча.

Для проведения замены необходимо выбрать игрока на поле, далее появится окно для проведения замены (заменой считается игрок, не находящийся на поле) (см. рисунок 24). Если замен нет, то система выдаст сообщение об этом.

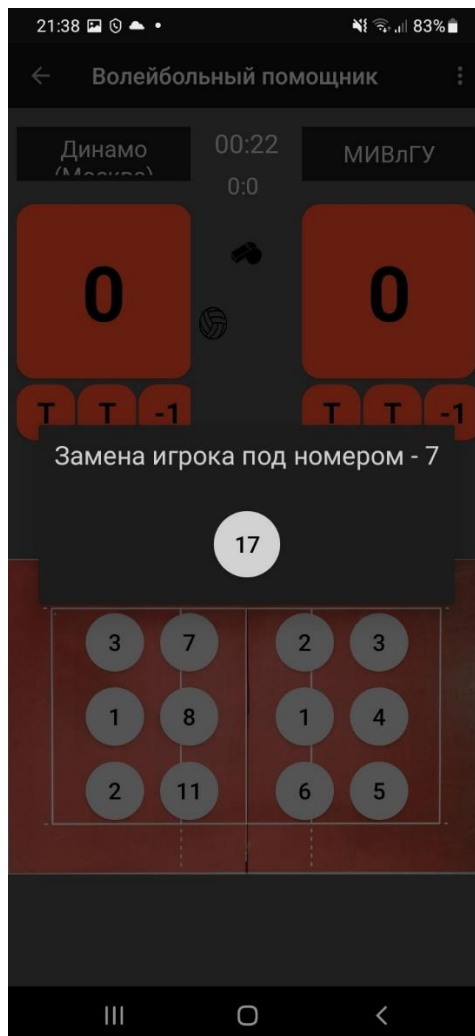


Рисунок 24 - форма выбора игрока замены

Если игрок выбран, система выдает сообщение об удачной замене игрока.

Для того что бы провести обратную замену нужно выбрать игрока и на форме подтверждения обратной замены нажать кнопку «Да» (см. Приложение 3 рисунок 13).

В конце матча выводится сообщение об окончании партии (см. рисунок 25).

					МИВУ.09.03.04-12.000 ПЗ	Лист
						55
Изм	Лист	№ докум.	Подп.	Дата		

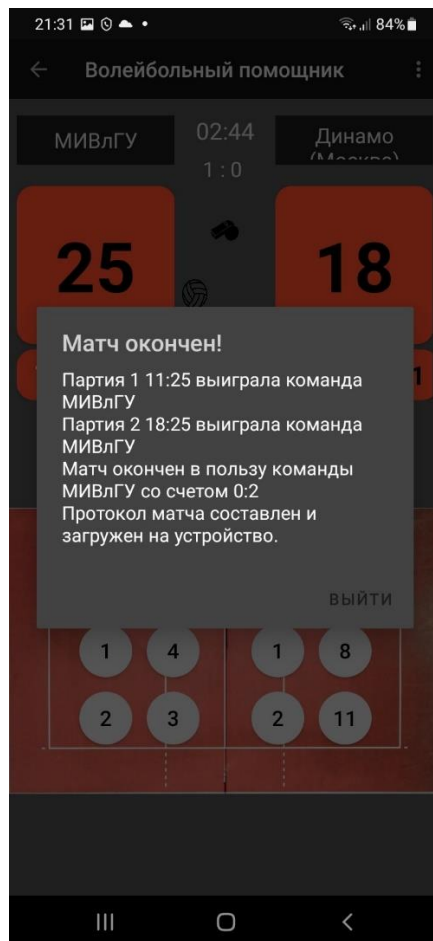


Рисунок 25 - сообщение о результате матча

Если при создании матча был выбран пункт составления протокола, то он будет загружен на устройство (пример представлен в Приложении 3 рисунок 14).

#### 4.1.2 Авторизированный пользователь

Если есть подключение к серверу, то в правом верхнем углу есть кнопка авторизации (форма авторизации и регистрации показаны в Приложении 3 рисунок 15-16).

После авторизации главная страница изменится (см. Приложение 3 рисунок 17). Появляется боковая панель для работы с профилем (изменение логина, электронной почты), просмотр команд и матчей (см. Приложение 3 рисунок 18-19).



## 4.2 WEB-сайт

Главная страница сайта показана в приложении 2 рисунок 20.

### 4.2.1 Гость сайта

Для гостя сайта доступны трансляции матчей пользователей по пункту меню «Трансляции» (см. Приложение 3 рисунок 21).

По пункту меню «Авторизация» пользователь может авторизоваться или зарегистрироваться (см. Приложение 3 рисунок 22-23).

### 4.2.2 Авторизированный пользователь

Для работы с данными пользователя необходимо перейти в «Профиль» (см. Приложение 3 рисунок 24). Для редактирования данных профиля необходимо ввести изменения и нажать кнопку «Изменить данные профиля». Если данные введены не верно у поля, в котором совершена ошибка появиться сообщение.

Для работы со своими трансляциями необходимо в профиле найти раздел «Трансляции». Для из завершения надо нажать соответствующую кнопку.

Для работы с командами и матчами в профиле необходимо нажать соответствующую кнопку.

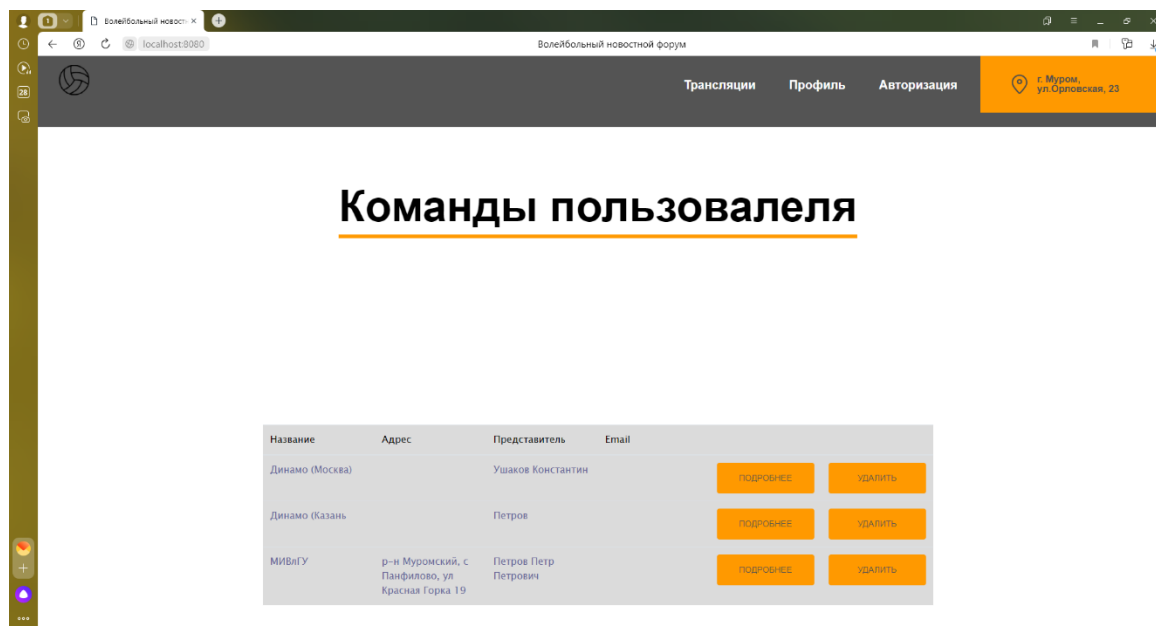


Рисунок 26 - команды пользователя



Волейбольный новостной форум

Трансляции Профиль Авторизация

г. Муром, ул. Орловская, 23

Название: МИБВГУ

Адрес: р-н Муромский, с. Панфилово, ул. Красная Горка 19

**Представитель команды**

ФИО: Петров Петр Петрович

Email:

СОХРАНИТЬ ИЗМЕНЕНИЯ

**Список игроков**

Номер	ФИО	Позиция
<b>Добавление игрока</b> Игровой номер (№) <input type="text" value="0"/> Позиция <input type="text" value="Игрок"/> ФИО <input type="text"/> Капитан: <input type="checkbox"/> <input type="button" value="Добавить"/> <input type="button" value="Очистить"/>		

Рисунок 28 - страницы изменения данных команды

Для добавления матча на определенную дату необходимо на странице команд и матчей пользователя на панели под таблицей/заголовком «Запланированные матчи» (см. рисунок 29) занести данные для запланированного матча и нажать кнопку «Добавить».

Волейбольный новостной форум

2023-06-10

УДАЛИТЬ

**Запланированные матчи**

Дата проведения	Название	Описание	Команды
2023-06-25	Товарищеская встреча	Товарищеская встреча между командами Динамо(Москва) и МИБВГУ(Муром)	МИБВГУ - Динамо (Москва)

УДАЛИТЬ

Название   
 Описание   
 Дата и время   
 Команда 1   
 Команда 2

Рисунок 29 - планирование матча

### 4.2.3 Администратор

Для удаления пользователя необходимо перейти в пункт меню «Пользователи» (см. рисунок 30) и в таблице нажать кнопку «Удалить» по выбранному пользователю.

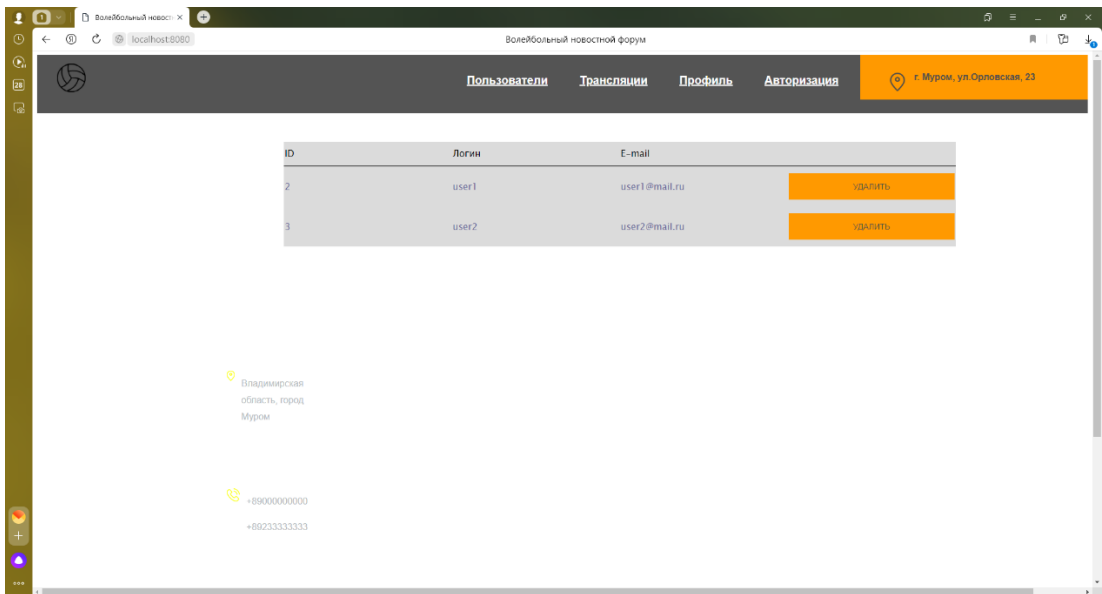


Рисунок 30 - удаление пользователей

Для удаления трансляции необходимо перейти в пункт меню «Трансляции» (см. рисунок 31) и нажать кнопку «Удалить» по выбранной трансляции.

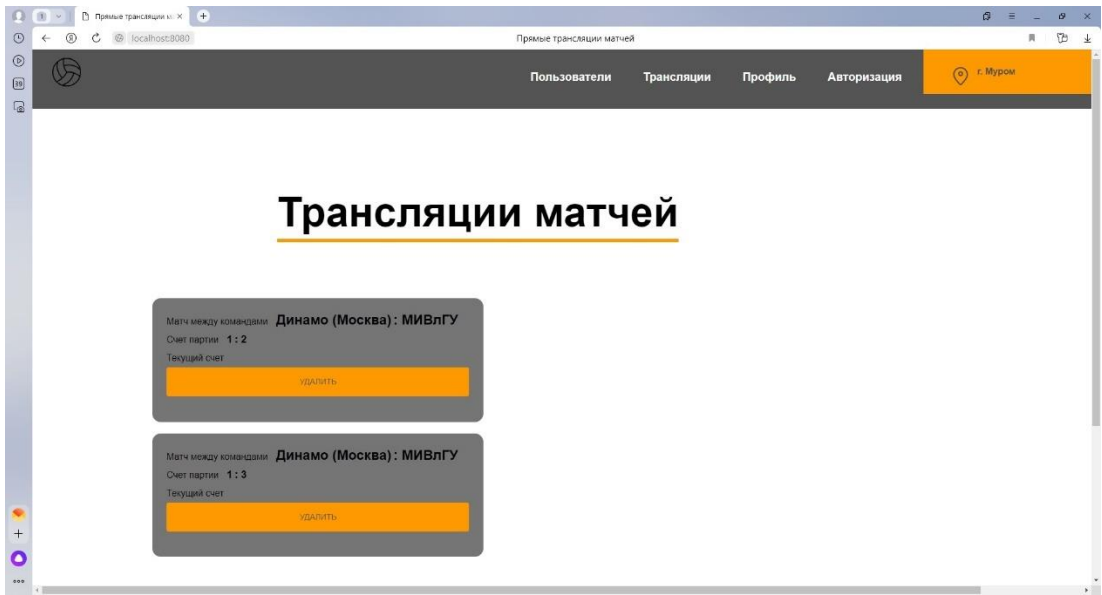


Рисунок 31 - удаление трансляции

## 5. Тестирование

При тестировании разрабатываемой системы использовалось ручное тестирование, так как оно позволяет быстро получить обратную связь.

Таблица 8 - тестирование мобильного приложения

Тест	Ожидаемый результат	Полученный результат
Ввод данных о пользователе при регистрации		
Не введены данные о почте.	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод электронной почты, которая уже занята другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Логин и/или Email заняты другим пользователем».
Корректный ввод электронной почты.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Не введен логин	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод логина, который уже занят другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Логин и/или Email заняты другим пользователем».
Логин введен, который не занят другим пользователем.	Ввод корректных данных.	Принятие корректного ввода логина и продолжение проверки других данных.
Ввод пароля меньше 8 или больше 20 символов	Вывод ошибки	Вывод сообщения об ошибке «Пароль должен быть больше 8, но не больше 20»
Ввод корректного пароля	Ввод корректных данных.	Принятие корректного ввода логина и продолжение проверки других данных.
Авторизация пользователя		

Ввод данных несоответствующих ни одному пользователю в системе.	Ошибка авторизации пользователя.	Вывод сообщение об ошибке: «Ошибка авторизации».
Ввод данных соответствующих пользователю в системе.	Успешная авторизация пользователя.	Успешная авторизация пользователя.
Не введен логин	Вывод ошибки	Вывод сообщения об ошибке «Обязательное поле».
Не введен пароль	Вывод ошибки	Вывод сообщения об ошибке «Обязательное поле».
Ввод данных для редактирования профиля		
Корректный ввод электронной почты.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Не введен логин	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Корректный ввод электронной почты.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Не введен логин	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод логина, который уже занят другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Логин и/или Email заняты другим пользователем».
Логин введен, который не занят другим пользователем.	Ввод корректных данных.	Принятие корректного ввода логина и продолжение проверки других данных.
Ввод данных об игроке.		
Выбор позиции игрока как Либеро, когда в команде	Вывод ошибки.	Вывод сообщения об ошибке «Нельзя назначить больше двух либеро».

уже назначены 2 и более игрока с такой позицией.		
Выбор позиции игрока как Либера, когда в команде назначены меньше 2 игроков с такой позицией.	Позиция игрока – Либера.	Позиция игрока – Либера.
Выбор позиции игрока как Игрок.	Позиция игрока – Игрок.	Позиция игрока – Игрок.
Ввод игрового номера игрока, который уже занят другим игроком данной команды.	Вывод ошибки.	Вывод сообщения об ошибке «Игровые номера индивидуальны».
Ввод игрового номера игрока, который еще не занят другим игроком данной команды.	Назначение игроку данный игровой номер.	Назначение игроку данный игровой номер.
Корректный ввод имени игрока.	Ввод корректных данных.	Принятие корректного ввода имени игрока и продолжение проверки других данных.
Не введено имя игрока	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод данных о команде		
Название команды не заполнено.	Вывод ошибки.	Вывод сообщения об ошибке «Заполните название команды».
Ввод названия команды.	Ввод корректных данных.	Принятие корректного ввода названия команды и продолжение проверки других данных.
Ввод адреса команды.	Ввод корректных данных.	Принятие корректного ввода адреса команды и продолжение проверки других данных.
Данные об имени представителя команды не занесены.	Вывод ошибки.	Вывод сообщения об ошибке «Заполните ФИО представителя команды».

Данные о представителе команды занесены.	Ввод корректных данных.	Принятие корректного ввода ФИО представителя команды и продолжение проверки других данных.
Выбор 2 одинаковых команд	Вывод ошибки.	Вывод сообщения об ошибке «Выбраны одинаковые команды».
Выбор 2 разных команд	Ввод корректных данных.	Принятие корректного ввода команд и продолжение проверки других данных.
Количество игроков в одной из команд меньше 6 игроков (игроками считаются только с позицией «Игрок»)	Вывод ошибки.	Вывод сообщения об ошибке «У команды меньше 6 игроков».
Количество игроков в одной из команд больше 6, но меньше 14 игроков	Ввод корректных данных.	Принятие корректного ввода количества игроков команды и продолжение проверки других данных.
Занесение стартовых составов команд		
Заполнены не все зоны.	Вывод ошибки.	Вывод сообщения об ошибке «Заполните стартовый состав команды».
Все зоны заполнены игроками.	Ввод корректных данных.	Принятие корректного ввода стартового состава и продолжение проверки других данных.
Проверка настроек официального матча		
Название матча не введено	Вывод ошибки	Вывод ошибки «Заполните название матча»
Название матча введено	Ввод корректных данных.	Принятие корректного ввода названия матча и продолжение проверки других данных.
Поставлен маркер на создание протокола	В конце матча создан протокол.	В конце матча создан протокол.



Поставлен маркер на транслирование матча	В процессе ведения матча на сервер отправляются данные.	В процессе ведения матча на сервер отправляются данные.
Введено описание матча	Ввод корректных данных.	Принятие корректного ввода описания матча и продолжение проверки других данных.
Проведение замен		
Выбор игрока, когда у команды закончились замены	Вывод ошибки.	Вывод сообщения об ошибке «Превышено количество замен в партии».
Выбор игрока, который уже был заменен	Вывод ошибки.	Вывод сообщения об ошибке «Выбранного игрока нельзя заменить».
Попытка замены игрока при отсутствии заменяющих игроков	Вывод ошибки.	Вывод сообщения об ошибке «Нет игроков для замены».
Замена игрока с соблюдением всех условий	Замена игрока	Вывод сообщения об успешном проведении замены.

Таблица 8 - тестирование сайта

Тест	Ожидаемый результат	Полученный результат
Ввод данных о пользователе при регистрации		
Некорректный ввод электронной почты или не введены данные о почте.	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод электронной почты, которая уже занята другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Email занят другим пользователем».
Корректный ввод электронной почты.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Не введен логин	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».

Ввод логина, который уже занят другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Логин занят другим пользователем».
Логин введен и не занят другим пользователем.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Ввод пароля меньше 8 или больше 20 символов	Вывод ошибки	Вывод сообщения об ошибке «Пароль должен быть больше 8, но не больше 20»
Ввод корректного пароля	Ввод корректных данных.	Принятие корректного ввода пароля и продолжение проверки других данных.
Авторизация пользователя		
Ввод данных несоответствующих ни одному пользователю в системе.	Ошибка авторизации пользователя.	Пользователь не авторизован.
Ввод данных, соответствующих пользователю в системе.	Успешная авторизация пользователя.	Успешная авторизация пользователя.
Ввод данных о пользователе при изменении профиля		
Не введены данные об электронной почте.	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод электронной почты, которая уже занята другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Email занят другим пользователем».
Корректный ввод электронной почты.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Не введен логин	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».

Ввод логина, который уже занят другим пользователем.	Вывод ошибки.	Вывод сообщения об ошибке «Логин занят другим пользователем».
Логин введен и не занят другим пользователем.	Ввод корректных данных.	Принятие корректного ввода электронной почты и продолжение проверки других данных.
Ввод неправильного старого.	Вывод ошибки	Вывод сообщения об ошибке «Неверно введен старый пароль»
Ввод правильного старого, но введенный новый пароль соответствует старому.	Вывод ошибки	Вывод сообщения об ошибке «Новый пароль не может соответствовать старому»
Ввод нового пароля меньше 8 или больше 20 символов и правильного старого.	Вывод ошибки	Вывод сообщения об ошибке «Пароль должен быть больше 8, но не больше 20»
Ввод корректного нового пароля и правильного старого	Ввод корректных данных.	Принятие корректного ввода пароля и продолжение проверки других данных.
Ввод данных об игроке.		
Выбор позиции игрока как Либера, когда в команде уже назначены 2 и более игрока с такой позицией.	Вывод ошибки.	Вывод сообщения об ошибке «Нельзя назначить больше двух либеро».
Выбор позиции игрока как Либера, когда в команде назначены меньше 2 игроков с такой позицией.	Позиция игрока – Либера.	Позиция игрока – Либера.
Выбор позиции игрока как Игрок.	Позиция игрока – Игрок.	Позиция игрока – Игрок.
Ввод игрового номера игрока, который уже занят другим игроком данной команды.	Вывод ошибки.	Вывод сообщения об ошибке «Номер игрока занят».

Ввод игрового номера игрока, который еще не занят другим игроком данной команды.	Назначение игроку данный игровой номер.	Назначение игроку данный игровой номер.
Не введено имя игрока	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Введено имя игрока	Ввод корректных данных.	Принятие корректного ввода имени игрока и продолжение проверки других данных.
Ввод данных о команде		
Название команды не заполнено.	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Ввод названия команды.	Ввод корректных данных.	Принятие корректного ввода названия команды и продолжение проверки других данных.
Ввод адреса команды.	Ввод корректных данных.	Принятие корректного ввода адреса команды и продолжение проверки других данных.
Ввод имени представителя команды не занесены.	Вывод ошибки.	Вывод сообщения об ошибке «Обязательное поле».
Данные о представителе команды занесены.	Ввод корректных данных.	Принятие корректного ввода имени представителя команды продолжение проверки других данных.
Проверка планирования матча		
Выбор 2 одинаковых команд	Вывод ошибки.	Вывод сообщения об ошибке «Выбраны одинаковые команды».
Выбор 2 разных команд	Ввод корректных данных.	Принятие корректного ввода команд и продолжение проверки других данных.
Название матча не введено	Вывод ошибки	Вывод ошибки «Обязательное поле»

Название матча введено	Ввод корректных данных.	Принятие корректного ввода названия матча и продолжение проверки других данных.
Введена дата меньше сегодняшней	Вывод ошибки	Вывод ошибки «Ошибка ввода даты»
Введена дата равная или больше сегодняшней	Ввод корректных данных.	Принятие корректного ввода даты матча и продолжение проверки других данных.
Удаление данных		
Удаление матчи	Успешное удаление данных	Удаление матча и обновление данных на странице
Удаление игрока	Успешное удаление данных	Удаление игрока и обновление данных на странице
Удаление трансляции	Успешное удаление данных	Удаление трансляции и обновление данных на странице
Удаление команды, не имеющей матчей	Успешное удаление данных	Удаление игроков и команды и обновление данных на странице
Удаление пользователя	Успешное удаление данных	Удаление всех данных пользователя (матчей, трансляций, команд, их игроков) и обновление данных на странице

Скриншоты вывода ошибок показаны в Приложении 3.

## Заключение

В данной выпускной квалификационной работе была разработана распределенная информационная система «Волейбольный помощник», отвечающее всем требованиям технического задания и заказчика.

Были реализованы такие функции как:

1. Контроль игрового процесса, включающего в себя ведение счета очков, замен и таймаутов.
2. Ведение трансляций матча по выбору пользователя.
3. 2 уровнями доступа, первый для тренировок, второй для матчей.
4. Две роли пользователей (авторизированный и не авторизированный) с разными уровнями доступа.
5. Добавление, редактирование команд пользователя, редактирование составов команд на текущий матч.
6. Добавление и редактирование данных о представителях команд.
7. Добавление, редактирование и удаление команд на сайте.
8. Планирование матчей на определенную дату.
9. Автоматическое создание протокола матча по его окончании.

В ходе разработки мобильного приложения использовалась база данных SQLite, язык программирования Kotlin и среда разработки Android Studio.

Для разработки Web-сайта использовалась база данных MySQL, язык программирования Java и среда разработки IntelliJ IDEA.

В дальнейшем планируется улучшение разработанной системы и добавления в нее такого функционала как ведение матча несколькими людьми (судья, второй судья и секретарь), публикацией статей о матчах, назначение санкций во время ведения матча.

					МИВУ.09.03.04-12.000 ПЗ	Лист
						70
Изм	Лист	№ докум.	Подп.	Дата		

## Список использованных источников

1. Оценка по волейболу простая [Электронный ресурс].URL: [https://translated.turbopages.org/proxy\\_u/en-ru.ru.88346903-634e59ac-628b5d4d-74722d776562/https/m.apkpure.com/volleyball-score-simple/com.wespiapps.volleyballscore](https://translated.turbopages.org/proxy_u/en-ru.ru.88346903-634e59ac-628b5d4d-74722d776562/https/m.apkpure.com/volleyball-score-simple/com.wespiapps.volleyballscore) (Дата обращения 18.10.2022)
2. volleyball scoreboard [Электронный ресурс]: URL: <https://apkpure.com/ru/volleyball-scoreboard/com.libelulasoftware.volleypoints/download/29-APK> (Дата обращения 18.10.2022)
3. Volleyball Referee [Электронный ресурс]: URL: <https://apkpure.com/ru/volleyball-referee/com.tonkar.volleyballreferee>
4. Моделирование данных: зачем нужно и как реализовать [Электронный ресурс]: URL: <https://habr.com/ru/post/554388/> (Дата обращения: 20.10.2022)
5. Концептуальная модель [Электронный ресурс]: URL: [http://wikiredia.ru/wiki/Концептуальная\\_модель#:~:text=Концептуальная%20модель%20\(англ.%20conceptual%20model\),модель%20—%20модель%20предметной%20области](http://wikiredia.ru/wiki/Концептуальная_модель#:~:text=Концептуальная%20модель%20(англ.%20conceptual%20model),модель%20—%20модель%20предметной%20области) (Дата обращения: 20.10.2022)
6. Логическая модель [Электронный ресурс]: URL: <https://teletype.in/@reddo/SJsmWPnpV#:~:text=Логическая%20модель%20—%20графическое%20представление,базы%20данных%20и%20аппаратной%20платформы> (Дата обращения: 20.10.2022)
7. SQLite позволяет стать ближе к данным [Электронный ресурс]: URL: <https://loginom.ru/blog/sqlite> (Дата обращения: 10.11.2022)
8. Android Studio 3.1.2.173.4720617 [Электронный ресурс]: URL: <https://lumpics.ru/android-studio/> (Дата обращения: 1.11.2022)
9. Kotlin vs Java: что лучше учить Android разработчику? [Электронный ресурс]: URL: <https://lampalampa.net/ua/kotlin-vs-java-chto-luchshe-dlya-android-razrabotki/> (Дата обращения: 1.11.2022)

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		71

10. Преимущества и недостатки использования Spring Boot [Электронный ресурс]: URL: <https://javarush.com/groups/posts/3380-kofe-breyk-75-preimujshestva-i-nedostatki-ispoljhzovanija-spring-boot-funkcii-dlja-strok-v-java> (Дата обращения 12.12.2022)

11. Основы работы с MySQL [Электронный ресурс]: URL: <https://help.reg.ru/support/hosting/bazy-dannykh/osnovy-raboty-s-mysql> (Дата обращения 12.12.2022)

12. IntelliJ что это за программа [Электронный ресурс]: URL: <https://eft-soft.ru/intellij-cto-eto-za-programma> (Дата обращения 12.12.2022)

13. Для чего нужен Java и в чем он принципиально хорош [Электронный ресурс]: URL: [https://codernet.ru/articles/drugoe/dlya\\_chego\\_nuzhen\\_java\\_i\\_v\\_chem\\_on\\_princzipialno\\_xorosh/](https://codernet.ru/articles/drugoe/dlya_chego_nuzhen_java_i_v_chem_on_princzipialno_xorosh/) (Дата обращения 12.12.2022)

14. Spring Security [Электронный ресурс]: URL: <https://spring.io/projects/spring-security> (Дата обращения 13.01.2023)

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		72



## Приложение 1. Код программы

Код программы предоставлен в репозитории по ссылке:

<https://github.com/student12-Lvova/courseWork.git>

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		73

## Приложение 2. Интервью с заказчиком

<b>УПРАВЛЕНИЕ ДОКУМЕНТОМ</b>		
<b>Автор</b>	Львова К.Ю.	
<b>Дата создания</b>	06.02.23	
<b>ИНФОРМАЦИЯ О ВСТРЕЧЕ</b>		
<b>Время и дата</b>	14:00	
<b>Порядковый номер</b>	1	
<b>Адрес/ место</b>	МИВлГУ	
<b>УЧАСТНИКИ ВСТРЕЧИ</b>		
<b>Со стороны заказчика</b>	Привезенцев Д.Г.	
<b>Со стороны исполнителя</b>	Львова К.Ю.	
<b>РЕЗУЛЬТАТЫ ОБСУЖДЕНИЯ</b>		
<b>Пункт повестки/ вопрос</b>	<b>Информация, полученная от заказчика</b>	<b>Предварительные результаты обсуждения</b>
1. Количество режимов матчей (тренировочный, любительский, официальный)	Реализовать календарь матчей, с заранее заполненными данными о командах через сайт.	Предусмотреть возможность создания через сайт календаря матчей товарищеских встреч. В такие матчи данные заносятся заранее
2. Трансляция данных матча на сайт	Да, трансляция счета матча должна проводиться онлайн по выбору пользователя	Трансляция счета матча по партиям и текущей партии
3. Реализация проведения замен	По выбору игрока на площадке провести замену	Сделать удобный и понятный функционал для реализации замен
4. Количество типов матчей	Классический	Классический
5. Составление протокола по окончании матча	По окончанию матча по выбору пользователя	По окончанию матча по выбору пользователя
6. Дополнительные функции сайта	Календари	Календари
7. Авторизация пользователей	Авторизация пользователя для связи с сервером, если пользователь не авторизирован, то только тренировочные матчи	Авторизация пользователя для связи с сервером, если пользователь не авторизирован, то только тренировочные матчи

8. Платный контент	Не предусмотрен	Не предусмотрен
9. Отслеживание расстановок команд на площадке	Автоматическая смена расстановки. Не начинать матч если не забиты стартовые составы.	Автоматическая смена расстановки. Не начинать матч если не забиты стартовые составы.
10. Статистика матча	Хронология забитых мячей в тренировочном режиме.	Хронология забитых мячей в тренировочном режиме.
11. Настройки матча	Для создаваемых в приложении тренировочных матчей. К ним отнести: - кол-во партий - очки в партии - очки на тайм-брейке - очки при переходе на тайм брейке - использование санкций - просмотр расстановки	Настройка тренировочного матча: - кол-во партий - очки в партии - очки на тайм-брейке - очки при переходе на тайм брейке - использование санкций - просмотр расстановки

#### СТАТУС ПРОТОКОЛА

<b>Согласовано</b>	<i>Привезенцев Д.Г., заказчик</i>
<b>Утверждено</b>	<i>Привезенцев Д.Г., заказчик</i>

#### ИНФОРМАЦИЯ О СЛЕДУЮЩЕЙ ВСТРЕЧЕ

<b>Время/ Дата</b>	17.02.23 12:30	17.02.23 12:30
<b>Место</b>	МИВлГУ, ауд. 401/2	МИВлГУ, ауд. 401/2

### Приложение 3. Скриншоты программы

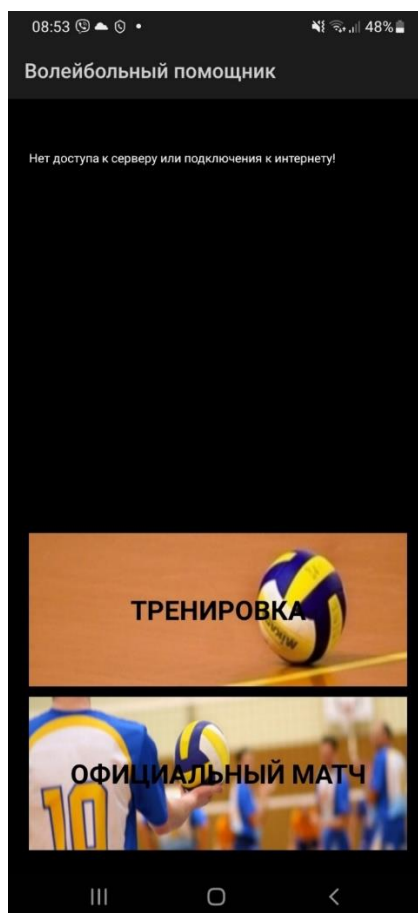


Рисунок 1 – главный экран в «автономном» режиме

					МИВУ.09.03.04-12.000 ПЗ	Лист
						76
Изм	Лист	№ докум.	Подп.	Дата		

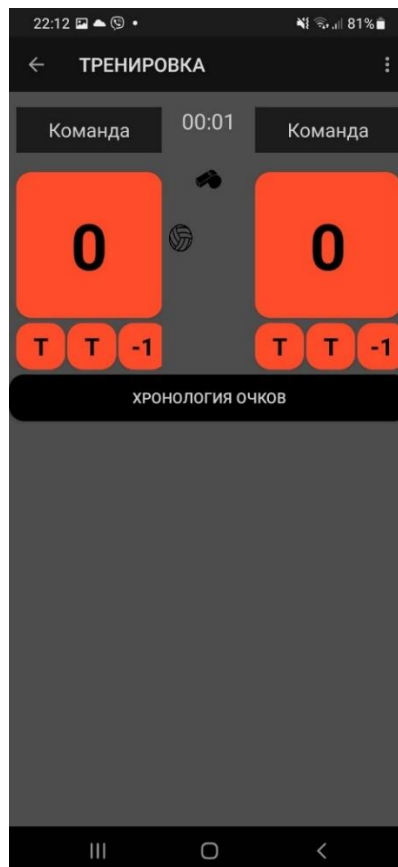


Рисунок 2 – начало тренировочного матча

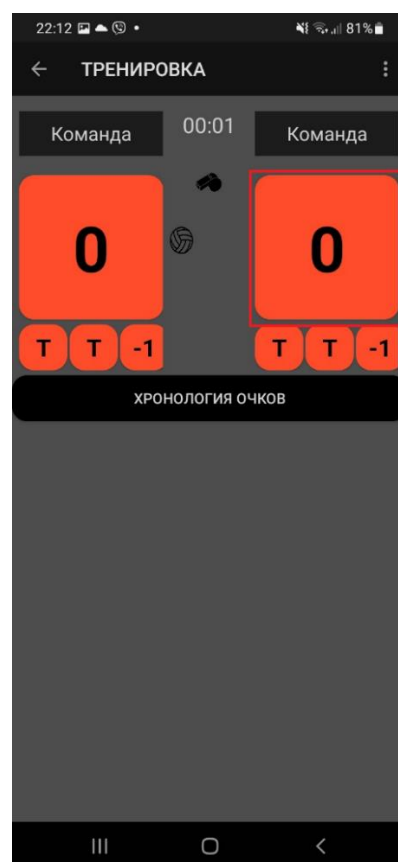


Рисунок 3 – добавление очков

					МИВУ.09.03.04-12.000 ПЗ	Лист
						77
Изм	Лист	№ докум.	Подп.	Дата		



Рисунок 4 – вычитание очков

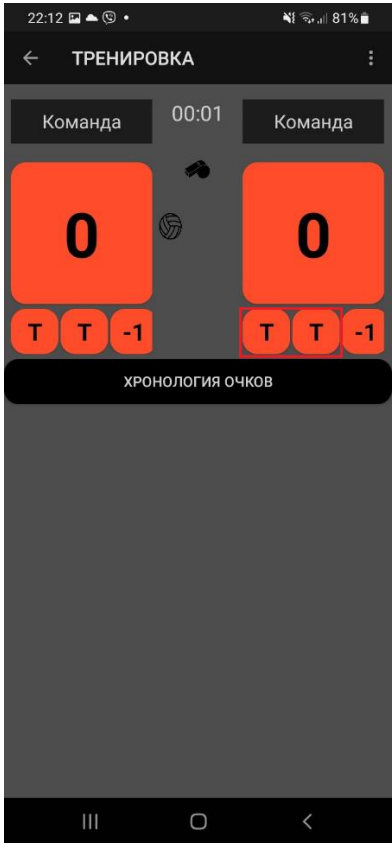


Рисунок 5 – ошибка ввода старого пароля



Рисунок 6 – кнопка «Хронология очков»

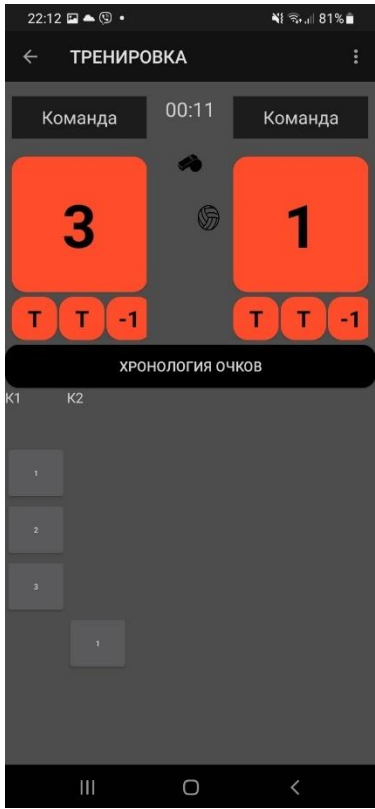


Рисунок 7– нажата кнопка «Хронология очков»



Рисунок 8 – меню матча



Рисунок 9 – ошибка жеребьёвки



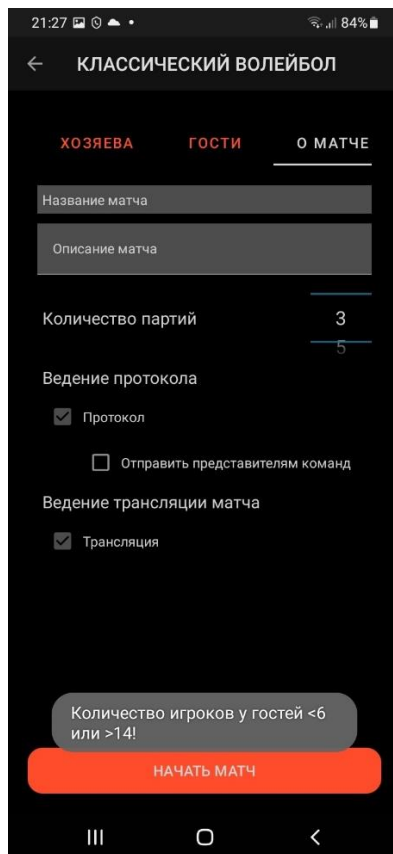


Рисунок 10 – ошибка занесения данных команды

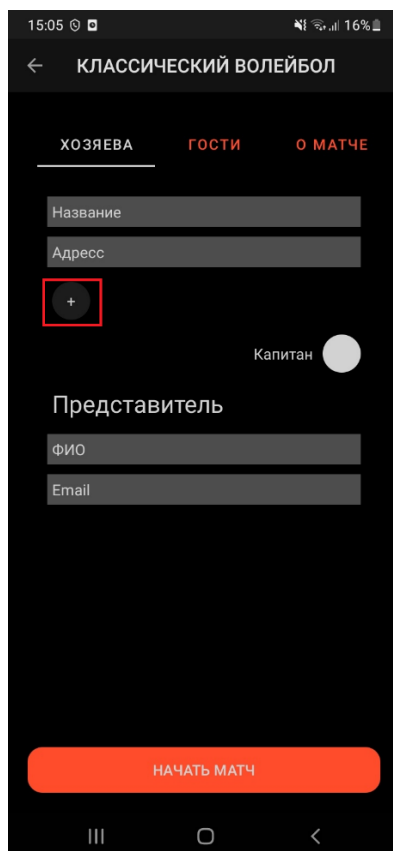


Рисунок 11 - кнопка для добавления игрока

					МИВУ.09.03.04-12.000 ПЗ	Лист
						81
Изм	Лист	№ докум.	Подп.	Дата		

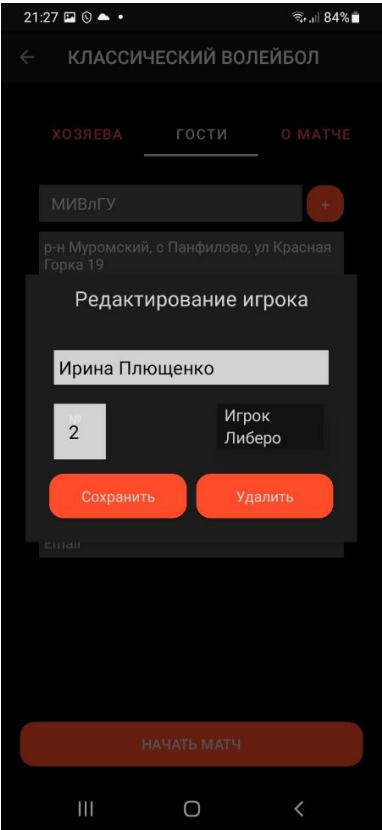


Рисунок 12 – форма добавления/редактирования игрока

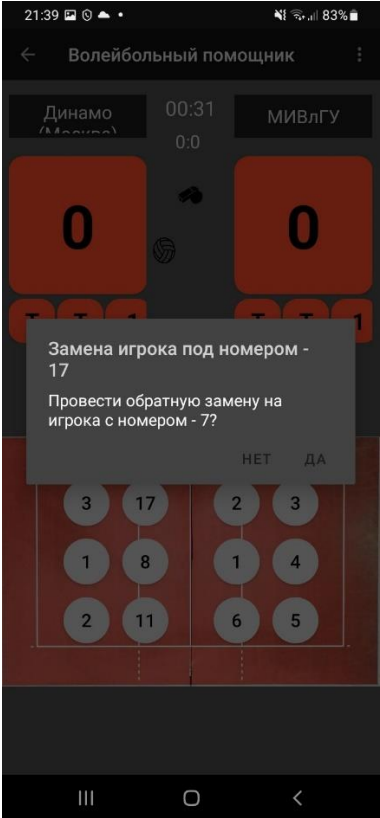


Рисунок 13 – форма подтверждения обратной замены

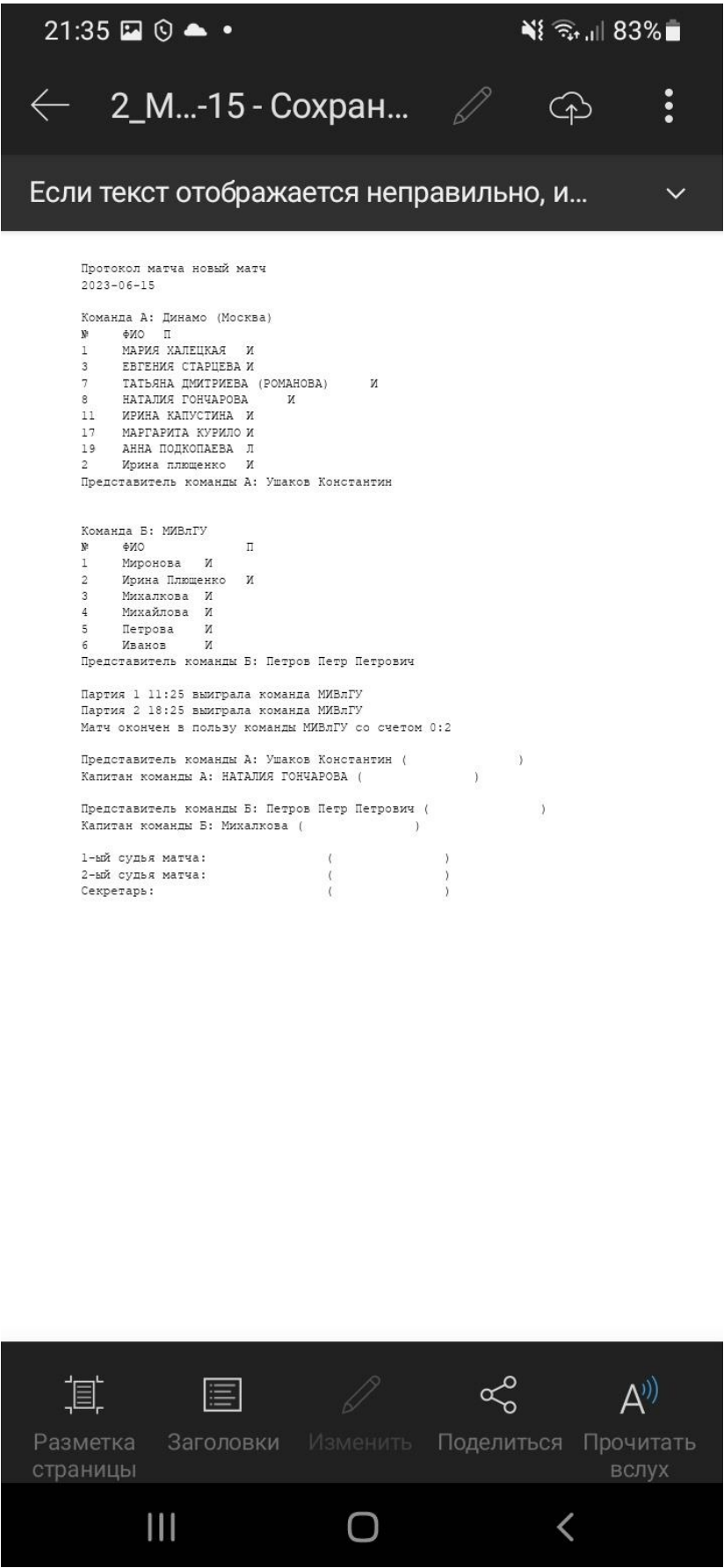


Рисунок 14 – пример протокола матча



Рисунок 15 - форма авторизации

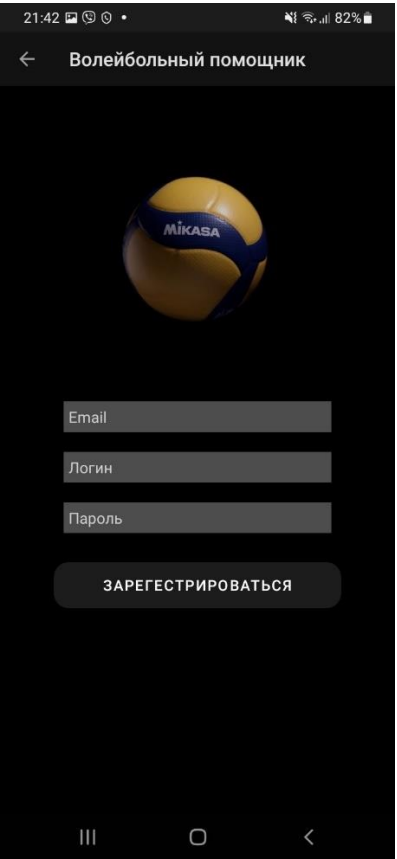


Рисунок 16 - форма регистрации



Рисунок 17 - форма главной страницы авторизованного пользователя

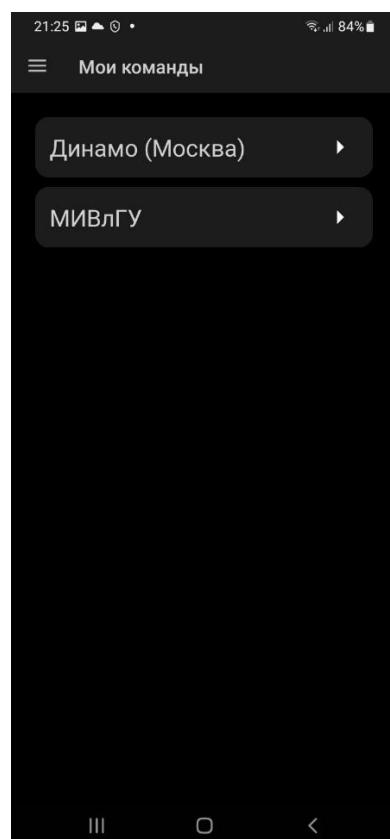


Рисунок 18 – команды пользователя

					МИВУ.09.03.04-12.000 ПЗ	Лист
						85
Изм	Лист	№ докум.	Подп.	Дата		

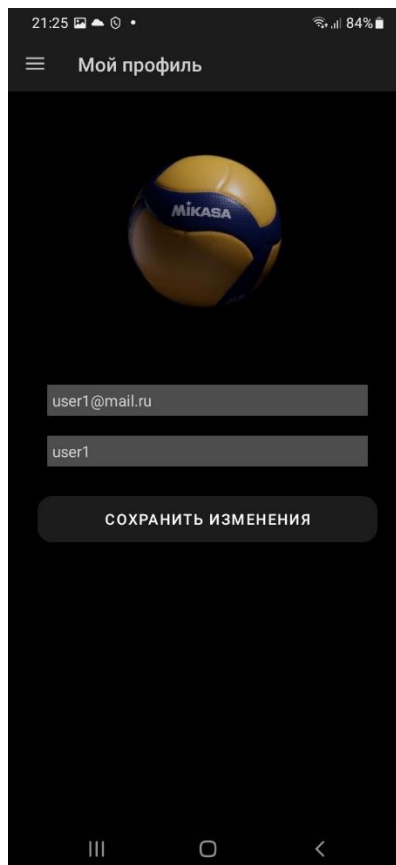


Рисунок 19 – профиль пользователя

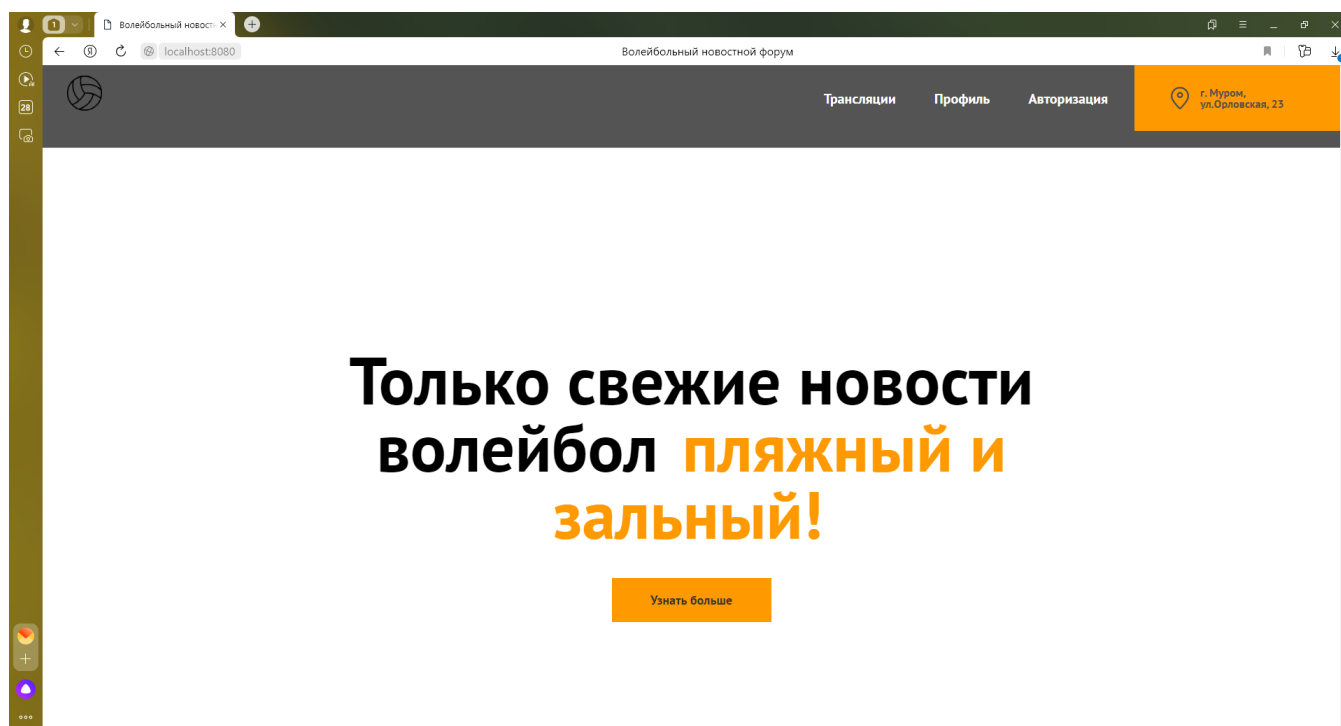


Рисунок 20 – главная страница сайта

					МИВУ.09.03.04-12.000 ПЗ	Лист
Изм	Лист	№ докум.	Подп.	Дата		86

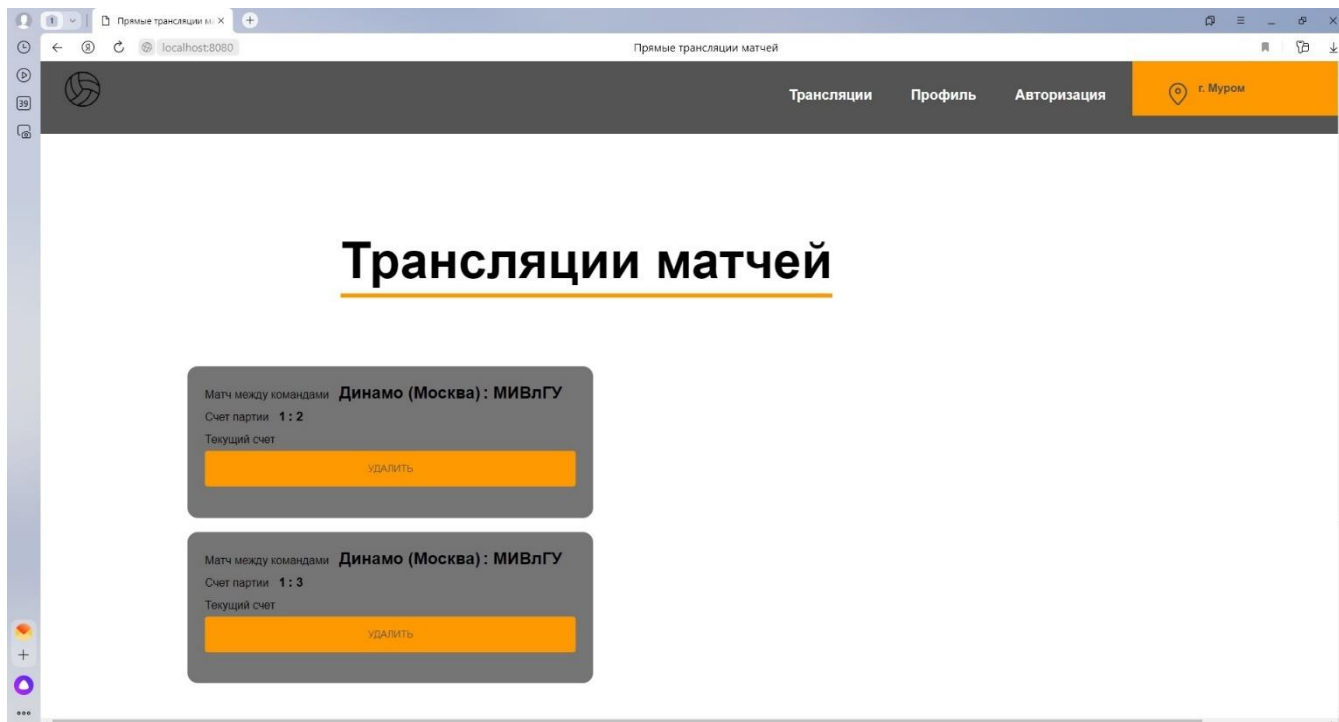


Рисунок 21 – страница «Трансляции»

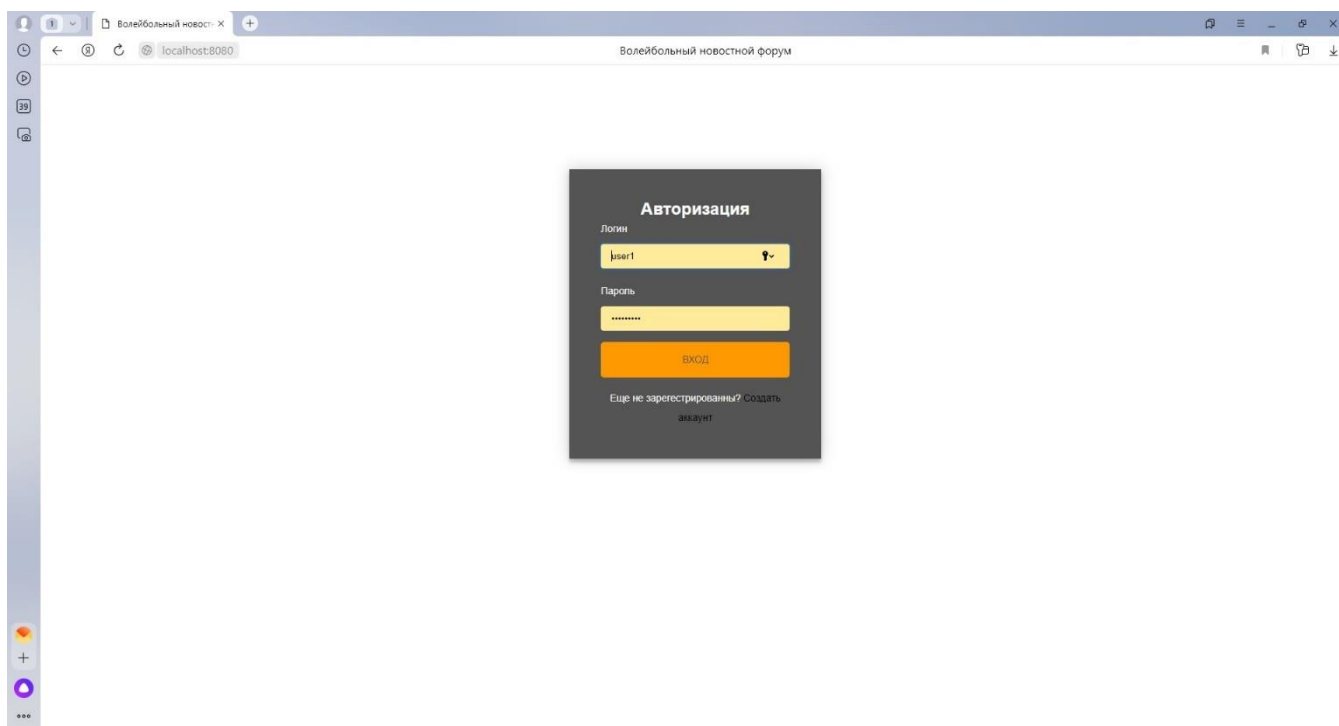


Рисунок 22 – страница «Авторизация»

Волейбольный новостной форум

## Регистрация

Логин:  
user1

Пароль:  
.....

E-mail:

ЗАРЕГИСТРИРОВАТЬСЯ

ОЧИСТИТЬ

Рисунок 23 – страница «Регистрация»

Профиль

Логин:  
user1

Старый пароль:  
.....

Новый пароль:

E-mail:  
user1@mail.ru

Изменить данные профиля

Очистить

### Трансляции

Матч между командами **Динамо (Москва)** : **МИВЛГУ**  
Счет партии **1 : 2**  
Текущий счет

ЗАВЕРШИТЬ

КОМАНДЫ И МАТЧИ

Рисунок 24 – страница «Профиль»



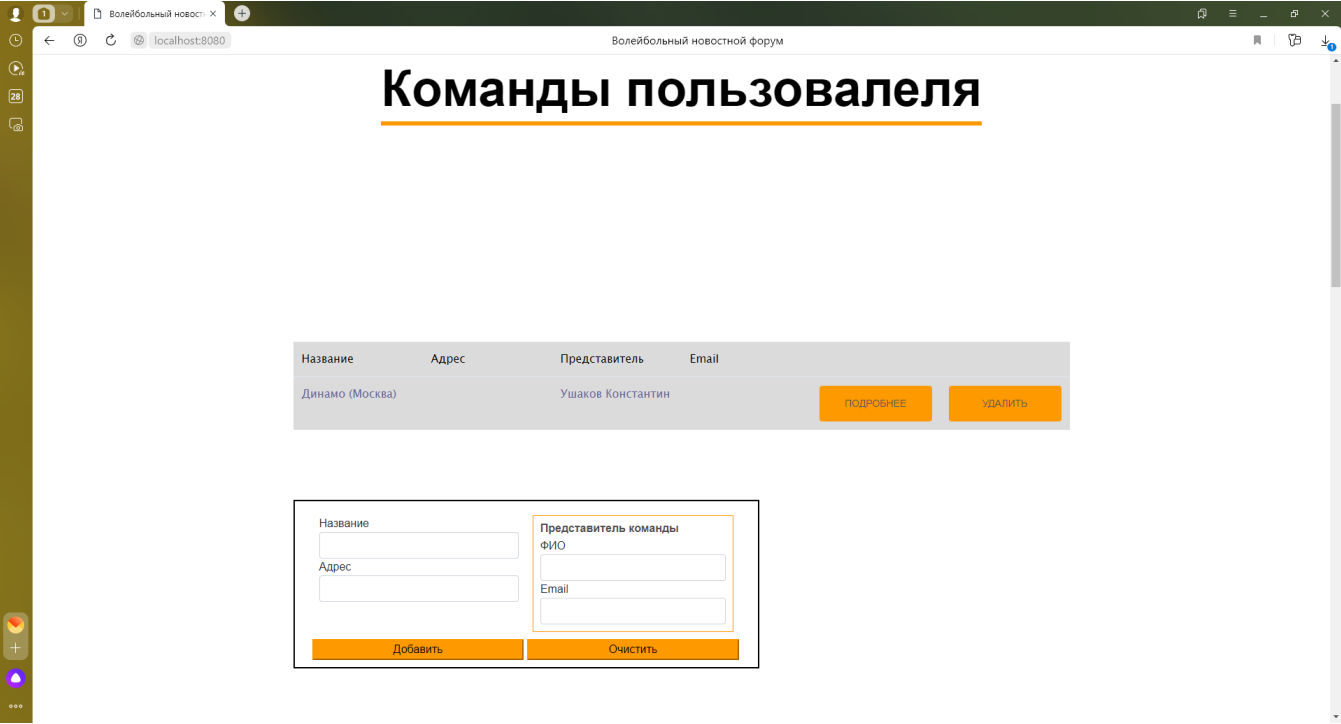


Рисунок 25 – добавление команды