



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования

«Дальневосточный федеральный университет»

(ДВФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

**Кафедра информатики, математического и компьютерного
моделирования**

«Кубок Йети - соревнование по построению машин Голдберга»

КУРСОВОЙ ПРОЕКТ

По направлению подготовки 01.03.02 Прикладная математика и информатика,
профиль «Системное программирование»

Выполнили студенты

гр. Б8119-01.03.02систпро

Середа Семен, Нагорнов Станислав

Пищиков Александр, Коренькин Семён

Руководитель курсового проекта

(должность, уч. звание)

Сущенко Андрей Андреевич

(ФИО)

(подпись)

« » _____ 20 г.

г. Владивосток
2021

Оглавление

1.	Введение	3
2.	Глоссарий	4
3.	Цели и задачи	5
3.1	Цель проекта	5
3.2	Задачи проекта	5
4.	Анализ предметной области	6
4.1	«Event.Rocks»	6
4.2	«Лига Голдберга»	7
4.3	«Eventum»	7
5.	Анализ требований к системе	9
5.1	Требования к стилю оформления кода.	9
5.2	Требования к архитектуре приложения.	9
5.3	Требования к пользовательскому интерфейсу	9
5.4	Принципы сохранения на сервере git.....	10
5.5	Требования к системе	10
5.5.1	Функциональные требования мобильного приложения	10
5.5.2	Функциональные требования оконного приложения	14
5.6	Проектирование API для мобильного приложения и тонкого клиента.	15
5.7	Требования к аппаратному и программному обеспечению	16
5.8	Требования к реализации.....	16
5.9	Заключение	18
6.	Проектирование программного средства.....	19
6.1	Введение	19
6.2	Архитектурно-контекстная диаграмма (АКД)	19
6.3	Архитектурная диаграмма потоков	20
6.4	Диаграмма прецедентов (вариантов использования).....	22
6.5	Заключение.	23
7.	Проектирование базы данных	24
7.1	Введение	24
7.2	Инфологическое проектирование	24
7.3	Взаимосвязи информационных объектов	33
8.	Заключение	35
9.	Список литературы	37
10.	Приложение	38

1. Введение

В наше время существует множество различных соревнований и каждый год появляется немало новых. Они проводятся по разным направлениям и правилам. В соревнованиях с небольшим количеством участников нет необходимости в особом программном обеспечении, потому что все можно зарегистрировать на бумажном носителе или в электронной таблице.

Однако, со временем соревнования приобретают огромную популярность и выходят различные региональные, государственные или даже международный уровни, и тогда без специального программного обеспечения не обойтись.

Цель нашего проекта - создать программное обеспечение, необходимое для улучшения качества и удобства проведения соревнования «Кубок Йети», задачей которого является построение машин Голдберга среди инженеров и изобретателей.

Машина Голдберга – это устройство, предназначенное для решения различного рода простых задач наиболее сложным путём.

В рамках соревнования «Кубок Йети», командам участников необходимо построить несколько взаимосвязанных элементов машины Голдберга из подручных средств, доступных на площадке соревнования. Первое такое соревнование прошло в 1949 году в университете Пердью (Индиана, США) между двумя студенческими группами. Оно проходило ежегодно, вплоть до 1956, когда временно прекратилось, но вновь возродилось уже в 1983 году и, с тех пор, проходит каждый год в марте, и уже давно вышло за пределы одного университета.

2. Глоссарий

- 1) **Back-end** — программно-аппаратная часть сервиса. Это набор средств, с помощью которых происходит реализация логики веб-сайта.
- 2) **Front-end** — клиентская сторона пользовательского интерфейса по отношению к программно-аппаратной части сервиса.
- 3) **UI (user interface)** – структура, отвечающая за перенос разработки продукта из исследований и макетов в привлекательный, понятный и отзывчивый для пользователей опыт.
- 4) **UX (user experience)** – процесс определения проблем пользователя и решение этих проблем с целью повышения его удовлетворённости.
- 5) **Машина Голдберга** – устройство, которое выполняет очень простое действие чрезвычайно сложным образом — как правило, посредством длинной последовательности взаимодействий по «принципу домино».

3. Цели и задачи

3.1 Цель проекта

Целями курсового проекта являются: разработка пакета приложений - сервисов, предназначенных для автоматизации и администрирования соревнования с предоставлением удобного сервиса участникам соревнования «Кубок Йети», с проектированием базы данных, которая будет привязана ко всем приложениям.

В пакет приложений – сервисов входят:

1. Информационная система YetiCup – мобильное приложение, демонстрирующее профиль участника мероприятия и новостную ленту, имеющее функцию обработки оценок в чемпионате.
2. Информационное приложение, предназначенное для ретрансляции результатов чемпионата на большие телевизионные экраны.
3. Оконное приложение – панель управления и мониторинга соревнований, которое запускается с персонального компьютера под операционной системы Windows, позволяющее добавлять участников и печатать результаты соревнований.

3.2 Задачи проекта

1. Изучить похожие популярные решения, выделить их сильные и слабые стороны;
2. Определить требования к новому решению;
3. Разработать архитектуру проекта - определить, необходимые приложения и их взаимосвязи;
4. Разработать новые приложения.

4. Анализ предметной области

Для работы потребуется разработать десктопное приложение для судей с функционалом вывода информации до 8 экранов, мобильное приложение для регистрации участников и отслеживания их прогресса на соревновании.

Лидирующими компаниями, у которых существуют достаточно известные приложения с таким же функционалом являются «Event.Rocks», «Eventum», «Лига Голдберга». Был проведён анализ данных приложений.

Функционал приложений у всех имеют одинаковые возможности:

- просмотр информации по марафону;
- регистрация на мероприятие;
- подать заявку на участие в мероприятии;
- стать организатором.

4.1 «Event.Rocks»

Приложение для конференций, выставок и других событий, с возможностью персонализации “под себя”, брендирования, управления и захватом внимания и привлечения публики к данному мероприятию. Существует возможность списка участников (paywall) или включения любого зрителя после обязательной регистрации.

4.1.1 Плюсы

1. Приложение, поддерживающее мероприятия разного формата;
2. Возможность общения с другими участниками через приложение;
3. Оповещение о мероприятии через push-уведомления;
4. Возможность транслировать мероприятие;

4.1.2. Минусы

1. Только онлайн-мероприятия;
2. Вход на мероприятие осуществляется только при помощи его ID;
3. Невозможно посмотреть функционал приложения, не участвуя в мероприятии - при запуске отображается только поиск мероприятия;

4. Отсутствует реализация десктоп – приложения для судей.

4.2 «Лига Голдберга»

Первая организация, начавшая проводить соревнования по построению машин Голдберга на всероссийском, региональном уровне. Первые Всероссийские студенческие инженерные командные соревнования по созданию машин Голдберга прошли осенью 2017 года.

4.2.1 Плюсы

1. Соревнование, подходящее по теме работы;
2. Существует сопутствующий курс по построению машин Голдберга;
3. Отборочные этапы к соревнованию проводятся по всей России.

4.2.2 Минусы

1. Отсутствие десктоп и мобильного приложения для участников и жури;
2. Отсутствие онлайн - регистрации на мероприятие.

4.3 «Eventum»

Приложение от Eventum Premo доступно для устройств на базе iOS и Android. Множество возможностей и функций: от подбора сценария и площадки до расчёта меню и количества обслуживающего персонала. Удобная русскоязычная программа с довольно продуманными функциями поможет в организации любых мероприятий.

4.2.3 Плюсы

1. Поддерживает мероприятия разного формата;
2. Организация мероприятия «под ключ»;
3. Работа с крупными «игроками» рынка.

4.2.4 Минусы

1. Отсутствует десктоп приложения.

Рассмотрев на практике три сервиса, изучив их основные характеристики, можно сформировать пользовательские требования к

будущему программному продукту.

На основании всего вышеизложенное можно прийти к выводу, что данные сервисы хороши по отдельности, но ни один из них полностью не покрывает всего объёма программных требований к разрабатываемому продукту.

5. Анализ требований к системе

5.1 Требования к стилю оформления кода.

Код будет придерживаться Google Java Style.

Именование названий переменных, методов, классов и прочих ресурсов на английском языке. Использование транслитераций не допускается.

5.2 Требования к архитектуре приложения.

Использовать паттерны проектирования при создании программного обеспечения, такие как Builder, Factory, Singleton. Использовать методологию MVC при проектировании модулей информационной системы и системы в целом.

Локализация приложений на русском языке. Необходима поддержка кириллических символов в информационной системе.

5.3 Требования к пользовательскому интерфейсу

При реализации приложения следует придерживаться адаптивной вёрстки и адаптивного дизайна пользовательского интерфейса.

Для мобильных приложений необходимо предусмотрена возможность работы в вертикальной ориентации экрана.

Требования к пользовательским интерфейсам настольных и мобильных систем:

- Количество действий на экране не должно превышать 8.
- Для каждого сложного действия создаётся свой экран.
- Пользовательский интерфейс должен быть ориентирован на возможность работы с экранами большего разрешения.
- При размещении используется одинаковые отступы для всех элементов одного типа на протяжении всего приложения.

5.4 Принципы сохранения на сервере git

Актуальная версия разработанной информационной системы должна быть в ветке «main». Полное комментирование сделанной работы проведено в «Pull request».

5.5 Требования к системе

5.5.1 Функциональные требования мобильного приложения

Функциональность мобильного приложения проекта представлена следующими пунктами:

1. Регистрация и авторизация участников и организаторов соревнований.
2. Просмотр результатов соревнований.
3. Редактирование личного профиля участника.
4. Реализация ленты будущих или прошлых мероприятий.
5. Просмотр истории и достижений пользователя.
6. Создание и поиск команды.
7. Выставление экспертных оценок для участника соревнований.

Стартовый экран приложения

Макет стартового экрана приведён в приложении №1.

Стартовый экран приложения автоматически появляется при запуске приложения. Он отображает логотип мероприятия.

Авторизация пользователей

Макет экрана авторизации пользователей приведён в приложении №2.

Экран авторизации пользователей появляется после стартового экрана. Экран авторизации пользователей позволяет ввести адрес электронной почты в

качестве идентификатора пользователя. В зависимости от вводимого значения по мере введения символов появляется маска электронной почты.

Поле пароль по умолчанию скрывается. При вводе пароля каждый вводимый символ отображается в виде «•».

После ввода пары логин-пароль происходит валидация пары. При ошибке валидации появляется сообщение для пользователя, при прохождении - переход на экран ленты мероприятий, с запоминанием состояния авторизации. При следующем запуске экран авторизации пропускается автоматически.

Лента событий мероприятия

Макет экрана ленты событий мероприятия приведён в приложении №3.

Экран событий мероприятия выводит списком график проведения мероприятия. Каждая строка из списка состоит из эмблемы с ассоциацией с событием или графического изображения, названия события и краткого его описания (до 64 символов). Текущее событие находится сверху списка и выделяется рамкой. Список событий мероприятия обладает свойством прокрутки в случае, если количество событий, не входит в экран.

Профиль пользователя

Макет экрана профиля пользователя приведён в приложении №4.

Экран профиля пользователя позволяет просматривать следующую информацию:

- графический аватар,
- фамилию, имя, отчество пользователя,
- дату рождения,
- место жительства
- адрес электронной почты
- номер телефона.

На экране профиля содержится функциональная возможность выхода из своего аккаунта информационной системы.

В случае если значение текстового поля больше его видимых размеров, то текст в конце видимой строки урезается, а полный текст появляется в текстовой подсказке к полю.

Из экрана профиля пользователя возможен переход к экранам «история и достижения», лента событий и авторизация.

История соревнований и достижения участника

Макет «История соревнований и достижения» участника приведён в приложении №5.

Экран «История и достижения» содержит список мероприятий, в которых пользователь участвовал в качестве участника, эксперта, модератора или организатора.

Если пользователь принимал участие в «Кубке Йети» в качестве участника, то выводится место участника.

Авторизация модераторов и экспертов

После авторизации участник, данной роли, должен увидеть стартовый экран, вид которого будет зависеть от того какая у него роль. Всего таких экранов 2:

1. Для экспертов. Должен представлять собой быструю систему поиска участников и команд для последующего ввода оценок.
2. Для участников. Должен отображать ленту событий.

Детектор участников – функция, необходимая для проверки профиля пользователя по его QR коду: пользователь делает фото кода, используя фронтальную камеру; затем QR код сразу же отсылается на дешифровку. После дешифровки отображаются персональные данные в виде ФИО и роли пользователя на соревнованиях.

Экран «История и достижения» должен быть доступен через вкладку «профиль пользователя», чтобы не загромождать навигацию между основными экранами.

Сборка произвольной команды

Макет создание произвольной команды приведён в приложениях №6.1, 6.2, 6.3.

В распоряжении каждого участника мероприятия для конструирования команды есть инструмент “генератор команды”. Для её создания пользователю необходимо ввести название самой команды и нажать кнопку “создать”. Если зарегистрированное количество участников в данной команде равно 1, то создатель может присоединиться к другой команде, а созданная им будет автоматически удалена.

Участники имеют возможность добавлять новых членов команды в свои ряды. Для этого необходимо совершить два действия: отправить приглашение человеку, а приглашённому её принять.

Ввод экспертных оценок

Макет ввода экспертных оценок приведён в приложениях №7.1, 7.2, 7.3.

Оценка действий участников - является важной задачей экспертов, контролирующих ход мероприятия. Поэтому необходимо реализовать быстрый и дружелюбный пользователю функционал, позволяющий вносить оценки участников в систему.

Существуют два типа оценивания – командная и индивидуальная, которые различаются по ряду критериев.

Командный тип - требует приложения фотографии итоговой машины Голдберга, собранной участниками какой-либо команды, а также заполнения

нескольких числовых полей, данных в макете. Проставляемый балл – от 0 до 10. Бал может быть только целочисленным.

При индивидуальном же типе пропускается критерий оценивания.

5.5.2 Функциональные требования оконного приложения

Функциональные возможности настольных систем:

1. Авторизация и регистрация в информационной системе для организаторов.
2. Окно со списком участников чемпионата и их баллы.
3. Окно со списком команд чемпионата и их баллы.
4. Окно трансляции информационной панели чемпионата.

Необходимо разработать несколько приложений для организаторов чемпионата. Первое – информационное, которое может быть запущено на больших телевизионных экранах. Второе приложение – панель управления и мониторинга соревнований. Оно запускается с персонального компьютера из-под операционной системы Windows и позволяет добавлять участников и печатать результаты соревнований.

Организаторы чемпионата должны проходить авторизацию в информационной системе. При авторизации вносится идентификатор пользователя, название чемпионата и пароль. После авторизации необходимо отобразить информационную панель по чемпионату. На информационной панели располагается следующая информация:

- список участников чемпионата и их баллы,
- список команд чемпионата и их баллы,
- перечень информационных панелей, подключённых к чемпионату.

Информационные панели - это большие вертикальные мониторы,

располагающиеся в разных местах площадки чемпионата. Количество панелей может быть от 1 до 8,

- программа трансляции информационной панели чемпионата.

Организаторы могут просматривать всех пользователей в информационной системе. Организаторы должны иметь права на использование функций CRUD.

Профиль пользователя информационной системы должен включать в себя:

- фамилию, имя, отчество;
- возраст или дату рождения;
- место жительства;
- фотографию;
- email;
- телефон.

Организаторы должны иметь возможность добавить пользователя системы на чемпионат в качестве одной из ролей: участник, эксперт или модератор. После добавления нового пользователя на чемпионат, он может авторизоваться через мобильное приложение.

5.6 Проектирование API для мобильного приложения и тонкого клиента.

Информационная система будет построена по классической схеме:

сервер базы данных ↔ сервер приложений ↔ клиентское приложение.

Для работы мобильного клиента информационной системы требуется реализовать сервер приложений.

При реализации сервера приложений учитывались следующие требования:

- однозначная идентификация пользователя в запросе к серверу приложений;
- обработка корректности выполнения запросов к серверу приложений; некорректный запрос должен отправлять сообщение об ошибке с текстом «Некорректный запрос»;
- поддержка кириллических символов.

Результатом выполненной работы является сервер приложений документированное API.

5.7 Требования к аппаратному и программному обеспечению

1) Системные требования для оконного приложения:

Операционная система: Windows 10.

Оперативная память: 512 МБ.

Память: 1 Гб

2) Системные требования для мобильного приложения:

Операционная система: Android 6.0.

Оперативная память: 256 МБ.

Память: 150 Мб

5.8 Требования к реализации

Для разработки оконных приложений существует множество разнообразных инструментов. Для этого проекта были отобраны популярные Swing, JavaFX, Electron, Qt C++, PyQt. В итоге был выбран фреймворк PyQt по следующим факторам:

1. Фреймворк Swing достаточно известен, однако, его можно отнести к устаревающим. Данный фреймворк больше не поддерживается,

благодаря этому он не получает новых возможностей. Так же он не обладает широким функционалом по сравнению с другими технологиями на рынке.

2. Фреймворк JavaFX не имеет большого сообщества, поэтому при возникновении проблем в разработке поиск решения может занять много времени вплоть до остановки разработки.
3. Electron позволяет создавать оконные приложения на основе языка JavaScript. Его недостаток - работа с языком программирования, в силу его особенностей. Могут возникнуть проблемы при использовании приложения - есть большая вероятность найти ошибку даже в уже протестированном приложении, чем при использовании других технологий.
4. Qt C++ позволяет быстро создать оконное приложение, однако, в силу сложности C++, разработка может затянуться из-за многих особенностей данного языка.

Для разработки мобильного приложения был выбран строго типизированный объектно-ориентированный язык программирования Java и Kotlin, среда разработки - Android Studio. Данный выбор пал по нескольким причинам:

1. Удобная документация от Google.
2. Удобный инструмент IDE Android Studio.
3. Не требуется много времени на разработку простого приложения.
4. Имеется большое сообщество разработчиков.
5. Язык является родным для разработки под Android

Разработка дизайна проводилась при помощи Adobe Photoshop, Adobe XD.

Для организации командной работы был выбран Git - распределённая система управления версиями.

5.9 Заключение

В ходе выявления требований к будущей эко системе был сформирован ряд чётких условий, функций, аппаратных аспектов необходимых для ее реализации. На основании проведённого анализа можно приступать к проектированию программного средства.

6. Проектирование программного средства

6.1 Введение

Проектирование программного средства – это последний этап разработки программного обеспечения, который контролируется заказчиком. В нем утверждаются все подсистемы системы, потоки данных между подсистемами, основные функции и подфункции пользователей, их тестирование. Прогнозируется результат ответа системы в любой момент работы. Описывается связь объектов системы. Процесс проектирования сопровождается построением разных диаграмм, отражающих пользовательское восприятие системы, и поясняющих разработчику поставленные перед ним задачи.

На основании анализа предметной области и маркетингового исследования была построена архитектурно-контекстную диаграмму системы, диаграмму прецедентов, диаграмму потоков данных. Также необходимо протестировать все пользовательские функции, выявленные в ходе построения диаграммы прецедентов.

6.2 Архитектурно-контекстная диаграмма (АКД)

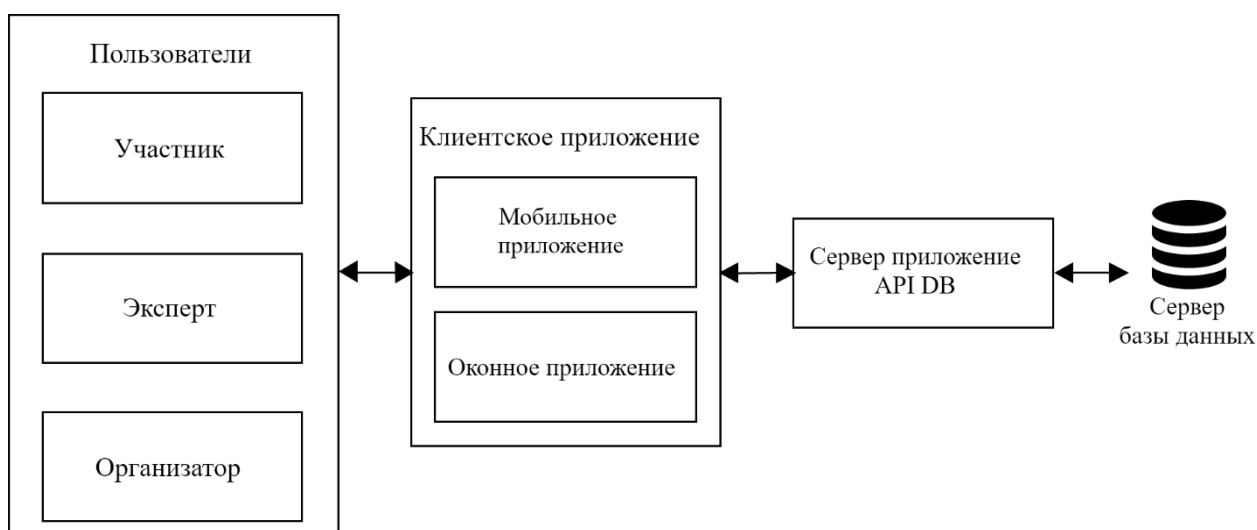


Рисунок. Архитектурно-контекстная диаграмма.

- 1) Оконное приложение – система, обеспечивающая управление соревнованием, вывод результатов, управление учётными записями участника.
- 2) Мобильное приложение – система, позволяющая участникам создавать свои учётные записи, просматривать и регистрироваться на соревнования.
- 3) Участники – люди, участвующие в мероприятии и создающие событийность.
- 4) Эксперты – люди, занимающиеся экспертной оценкой результатов работы участников в индивидуальном и командном зачёте.
- 5) Организаторы – люди, которые занимаются настройкой информационного оборудования и проведением соревновательной части мероприятия. Организаторы отвечают за ведение документального обеспечения соревнования.

6.3 Архитектурная диаграмма потоков

Архитектурная диаграмма потоков приведена в приложении №8.

Основными пользователями мобильного приложения будут участники, эксперты и модераторы. Основными функциями будут просмотр ленты событий мероприятия, профиля пользователя, истории событий и достижений участника, регистрация в мероприятиях и ввод экспертных оценок. Перед началом работы пользователь обязан зарегистрироваться или авторизоваться в приложении.

При запуске приложения перед пользователем появляется стартовый экран. На данном экране пользователь выбирает мероприятие, в котором он участвует. Если пользователь выбрал мероприятие, то при повторном запуске производится автоматический переход в окно авторизации.

В окне авторизации пользователь вводит логин и пароль. После попадает в свой профиль, где отображаются его личные данные: аватар, ФИО, дата рождения, место жительства, адрес электронной почты и номер телефона. Из экрана профиля пользователя возможен переход к экранам «история и достижения», лента событий и авторизация.

В экране «история и достижения» отображаются мероприятия, в которых пользователь принимал участие в качестве участника, эксперта, модератора или организатора. Если пользователь состоял участником в мероприятии, отображается место участника.

Экран ленты событий отображает график проведения мероприятия. Каждое событие состоит из эмблемы, которая ассоциируется с событием, названия события и краткого его описания.

Основными пользователями оконного приложения будут организаторы. Основными функциями будут вывод результатов на экран, просмотр, добавление, изменение и удаление аккаунтов участников. Перед началом работы пользователь обязан зарегистрироваться или авторизоваться в приложении.

После авторизации необходимо отобразить информационную панель по чемпионату. На информационной панели располагается следующая информация:

- список участников чемпионата и их баллы,
 - список команд чемпионата и их баллы,
 - перечень информационных панелей, подключённых к чемпионату.
- Информационные панели представляют большие вертикальные мониторы, располагаемые в разных местах площадки чемпионата. Количество панелей может быть от 1 до 8,
- программа трансляции информационной панели чемпионата.

6.4 Диаграмма прецедентов (вариантов использования)

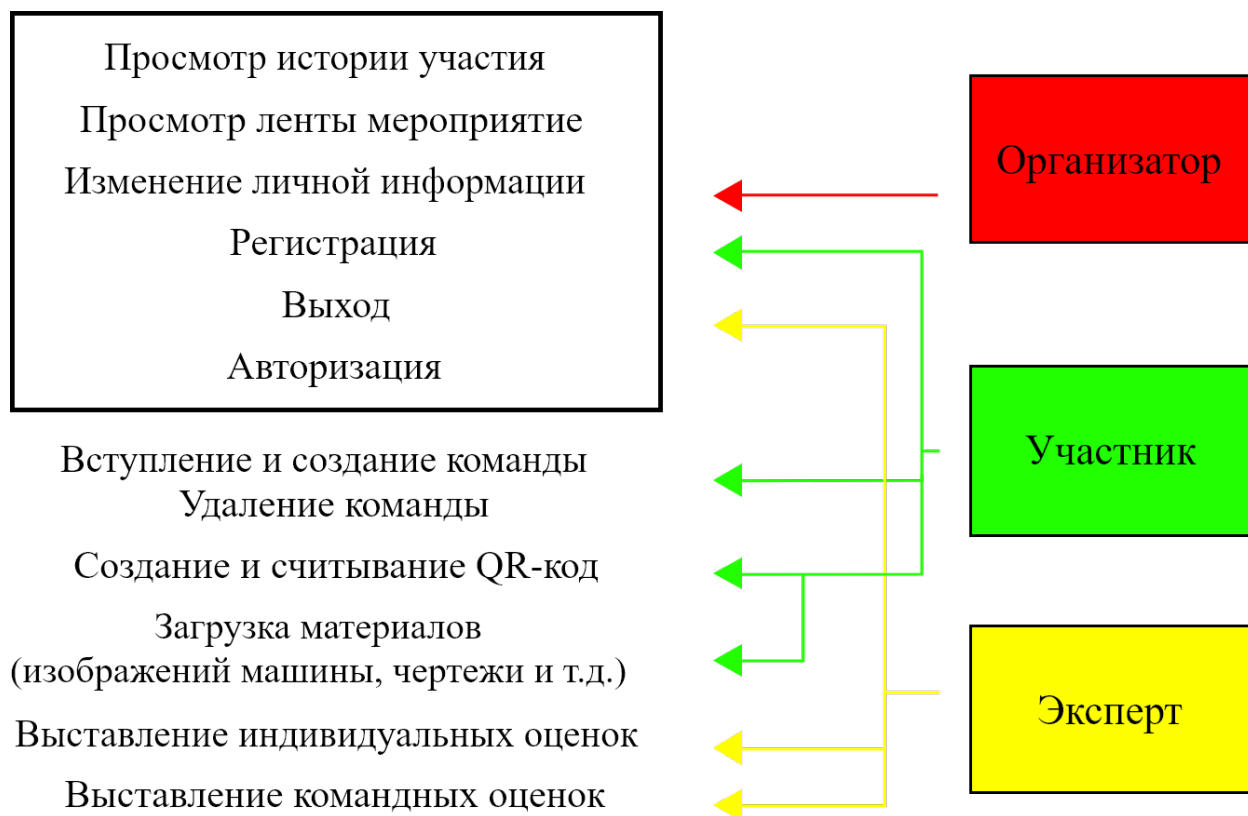


Рисунок. *Диаграмма прецедентов.*

- 1) Участник, эксперт, модератор, организатор - пользователи системы с разными правами
- 3) Сервер - сервер, на котором хранятся данные
- 4) БД - база данных файлов, хранящихся на сервере.
- 5) Создание аккаунта - добавление информации о новом пользователе в базу данных организатором
- 6) Изменение аккаунта - изменение информации о пользователе в базе данных
- 7) Удаление аккаунта - удаление информации об участнике из базы данных
- 8) Показ контента - вывод информации о соревновании на телевизионные экраны
- 9) Управление информационной панелью - управление телевизионными экранами для показа контента
- 10) Ввод оценок - изменение экспертом информации об оценках участника
- 11) Просмотр соревнований - показ информации о текущих соревнованиях

12) Регистрация в соревновании - добавление участника в список участников соревнования

13) Регистрация аккаунта - добавление информации о новом пользователе в базу данных через мобильное приложение

13) Загрузка данных – процесс передачи информации об участниках и соревнованиях в приложения

14) Приложения - специальные программы для участников, экспертов, организаторов для общения с базой данных участников или соревнований

6.5 Заключение.

На основе выявленных требований из предыдущей главы, был сформирован проект программного средства, позволяющий пользователю и разработчику окончательно договориться об однозначном понимании системы. Проект позволяет разработчику приступить непосредственно к реализации. Но на данном этапе необходимо провести тестирование всех пользовательских функций.

7. Проектирование базы данных

7.1 Введение

Для успешной работы комплекса приложений требуется использовать СУБД для хранения данных, выбор был сделан в пользу свободной объектно-реляционной системы управления базами данных SQLite. Для быстрой и удобного управления системой.

7.2 Инфологическое проектирование

Описание информационных объектов:

7.2.1. Users (пользователи)

Таблица, содержащая профили пользователей. Является первичной по отношению к таблице «Participants».

1) Id

- a. Отвечает за однозначную идентификацию каждого пользователя.
- b. Тип данных – целочисленный.
- c. Является уникальным.
- d. Генерируется автоматически СУБД при создании нового пользователя.
- e. Изменение данного поля с учетом уникальности других полей того же типа может производить системный администратор.
- f. Степень важности – высокая.
- g. Является обязательным для заполнения.

2) Name

- a. Содержит фамилию, имя и отчество пользователя.
- b. Тип данных – текст с фиксированной длиной.
- c. Ограничение длины – 50 символов.
- d. Не является уникальным.
- e. Изменение данного поля может производить как пользователь, так и администратор.
- f. Степень важности – средняя.
- g. Является обязательным для заполнения.

3) Login

- a. Содержит логин пользователя, необходимый для регистрации и дальнейшей авторизации.
- b. Тип данных – текст с фиксированной длиной.
- c. Ограничение длины – 50 символов.
- d. Является уникальным.
- e. Изменение данного поля может производить только пользователь.
- f. Степень важности – высокая.
- g. Является обязательным для заполнения.

4) Password

- a. Содержит пароль пользователя, необходимый для регистрации и дальнейшей авторизации.
- b. Тип данных – текст с ограниченной длиной.
- c. Ограничение длины – 100 символов.
- d. Не является уникальным.
- e. Изменение данного поля может производить только пользователь.
- f. Степень важности – высокая.
- g. Является обязательным к заполнению.

5) Mail

- a. Содержит электронную почту пользователя.
- b. Тип данных – текст с ограниченной длиной.
- c. Ограничение длины – 150 символов.
- d. Является уникальным.
- e. Изменение данного поля может производить только пользователь.
- f. Степень важности – низкая.
- g. Не является обязательным к заполнению.

6) Address

- a. Содержит адрес проживания пользователя.
- b. Тип данных – текст с ограниченной длиной.
- c. Ограничение длины – 100 символов.

- d. Не является уникальным.
- e. Изменение данного поля может производить только пользователь.
- f. Степень важности – низкая.
- g. Не является обязательным к заполнению.

7) Phone

- a. Содержит номер телефона пользователя.
- b. Тип данных – текст с ограниченной длиной.
- c. Ограничение длины – 12 символов.
- d. Является уникальным.
- e. Изменение данного поля может производить только пользователь.
- f. Степень важности – средняя.
- g. Не является обязательным к заполнению.

8) Photo

- a. Содержит номер телефона пользователя.
- b. Тип данных – фото файл
- c. Не является уникальным.
- d. Изменение данного поля может производить как пользователь, так и администратор.
- e. Степень важности – низкая.
- f. Не является обязательным к заполнению.

9) Date

- a. Содержит дату регистрации пользователя.
- b. Тип данных – дата формат.
- c. Не является уникальным.
- d. Изменение данного поля может производить только администратор.
- e. Степень важности – низкая.
- f. Заполняется автоматически СУБД при создании профиля пользователя.

7.2.2 Competitions (соревнования).

Таблица, содержащая информацию обо всех соревнованиях. Является первичной по отношению к таблице «Participants»

1) Year

- a. Содержит год проведения соревнования.
- b. Является первичным ключом
- c. Тип данных – целочисленный.
- d. Является уникальным.
- e. Изменение данного поля может производить только администратор.
- f. Степень важности – высокая.
- g. Является обязательным к заполнению

2) Comp_name

- a. Содержит название соревнования.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 50 символов.
- d. Является уникальным.
- e. Изменение данного поля может производить только администратор.
- f. Степень важности – высокая.
- g. Является обязательным к заполнению

3) Address

- a. Содержит адрес проведения соревнования.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 100 символов.
- d. Не является уникальным.
- e. Изменение данного поля может производить только администратор.
- f. Степень важности – высокая.
- g. Является обязательным к заполнению.

4) Date

- a. Содержит дату проведения соревнования.
- b. Тип данных – дата формат.
- c. Является уникальным
- d. Изменение данного поля может производить только администратор.
- e. Степень важности – высокая.

f. Является обязательным к заполнению.

5) Users_number

- a. Содержит количество участников на соревновании.
- b. Тип данных – целочисленный.
- c. Не является уникальным
- d. Изменение данного поля происходит при добавлении или удалении участников, связанных с этим соревнованием
- e. Степень важности – средняя.
- f. Заполняется значением по умолчанию, равным нулю.

7.2.3 Participants (участники)

Таблица, содержащая информацию об участниках за все время. Является вторичной по отношению к таблицам «Users», «Competitions» и «Teams».

1) User_id

- a. Является внешним ключом. Ссылается на таблицу «Users», поле id.
- b. Является частью составного первичного ключа
- c. Тип данных – целочисленный.
- d. Не является уникальным.
- e. Изменение данного поля производится только администратором.
- f. Степень важности – высокая.
- g. Заполняется значением, на которое ссылается в первичной таблице.

2) Year

- a. Является внешним ключом. Ссылается на таблицу «Competitions», поле year.
- b. Является частью составного первичного ключа
- c. Тип данных – целочисленный.
- d. Не является уникальным.
- e. Изменение данного поля производится только администратором.
- f. Степень важности – высокая.
- g. Заполняется значением, на которое ссылается в первичной таблице.

3) Score (счёт)

- a. Содержит индивидуальную оценку участника.
- b. Тип данных – целочисленный.
- c. Не является уникальным.
- d. Изменение данного поля производится только экспертом.
- e. Степень важности – высокая.
- f. Заполняется значением по умолчанию, равным нулю.

4) Role

- a. Содержит роль участника.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 20 символов.
- d. Не является уникальным.
- e. Изменение данного поля производится только администратором.
- f. Степень важности – высокая.
- g. Является обязательным к заполнению.

5) Team_id

- a. Является внешним ключом. Ссылается на таблицу «Teams», на поле id.
- b. Тип данных – целочисленный.
- c. Не является уникальным.
- d. Изменение данного поля производится только экспертом (админом?).
- e. Степень важности – средняя.
- f. Не является обязательным к заполнению.
- g. При удалении связанной строки, значение становится NULL.

7.2.4 Teams (команды)

Таблица, содержащая информацию обо всех командах. Является первичной по отношению к таблицам «Participants», «Images» и «Blueprints».

1) Id

- a. Содержит уникальный идентификатор команды.
- b. Является первичным ключом.
- c. Тип данных – целочисленный.
- d. Является уникальным.

- e. Изменение данного поля производится только админом.
- f. Степень важности – высокая.
- g. Заполняется автоматически СУБД при создании строки.

2) Team_name (название команды)

- a. Содержит название команды.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 25 символов.
- d. Является уникальным.
- e. Изменение данного поля производится капитаном команды?
- f. Степень важности – высокая.
- g. Является обязательным к заполнению.

3) Score (счёт)

- a. Содержит общую оценку команды.
- b. Тип данных – целочисленный.
- c. Не является уникальным.
- d. Изменение данного поля производится экспертом.
- e. Степень важности – высокая.
- f. Заполняется значением по умолчанию, равным нулю.

4) Video_path (путь к видео)

- a. Содержит видеоотчёт команды – работу машины.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 255 символов.
- d. Является уникальным.
- e. Изменение данного поля производится капитаном команды?
- f. Степень важности – средняя.
- g. Является обязательным к заполнению.

5) Info

- a. Содержит информацию о команде.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 255 символов.

- d. Является уникальным.
- e. Изменение данного поля производится капитаном команды?
- f. Степень важности – средняя.
- g. Не является обязательным к заполнению.

6) Place

- a. Содержит место, занятое командой на соревновании.
- b. Тип данных – целочисленный.
- c. Не является уникальным.
- d. Изменение данного поля производится экспертом.
- e. Степень важности – высокая.
- f. Заполняется значением по умолчанию, равным нулю.

7.2.5 Images (фотографии)

Таблица, содержащая фотографии результатов команд и дополнительную информацию. Является вторичной по отношению к таблице «Teams».

7) Id

- a. Содержит уникальный идентификатор.
- b. Тип данных – целочисленный.
- c. Является уникальным.
- d. Изменение данного поля производится капитаном команды.
- e. Степень важности – высокая.
- f. Заполняется автоматически СУБД при создании строки.

8) Photo (фотография)

- a. Содержит фотографию машины.
- b. Тип данных – фото файл.
- c. Является уникальным.
- d. Изменение данного поля производится капитаном команды.
- e. Степень важности – высокая.
- f. Является обязательным к заполнению.

9) Comment (комментарий)

- a. Содержит комментарий к фотографии.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 255 символов.
- d. Не является уникальным.
- e. Изменение данного поля производится капитаном команды.
- f. Степень важности – низкая.
- g. Не является обязательным к заполнению.

10) Team_id (id команды)

- a. Является внешним ключом. Указывает на таблицу «Teams», на поле id.
- b. Тип данных – целочисленный.
- c. Не является уникальным.
- d. Изменение данного поля производится капитаном команды?
- e. Степень важности – высокая.
- f. Является обязательным к заполнению.

7.2.6 Blueprints (чертежи)

Таблица, содержащая чертежи команд и дополнительную информацию.
Является вторичной по отношению к таблице «Teams».

1) Id

- a. Содержит уникальный идентификатор.
- b. Тип данных – целочисленный.
- c. Является уникальным.
- d. Изменение данного поля производится капитаном команды.
- e. Степень важности – высокая.
- f. Заполняется автоматически СУБД при создании строки.

2) Blueprint (чертеж)

- a. Содержит фотографию чертежа.
- b. Тип данных – фото файл.
- c. Является уникальным.
- d. Изменение данного поля производится капитаном команды.
- e. Степень важности – высокая.

f. Является обязательным к заполнению.

3) Info (информация)

- a. Содержит информацию о чертеже.
- b. Тип данных – текст ограниченной длины.
- c. Ограничение длины – 255 символов.
- d. Не является уникальным.
- e. Изменение данного поля производится капитаном команды.
- f. Степень важности – низкая.
- g. Не является обязательным к заполнению.

4) Team_id (id команды)

- a. Является внешним ключом. Указывает на таблицу «Teams», на поле id.
- b. Тип данных – целочисленный.
- c. Не является уникальным.
- d. Изменение данного поля производится капитаном команды?
- e. Степень важности – высокая.
- f. Является обязательным к заполнению.

7.3 Взаимосвязи информационных объектов

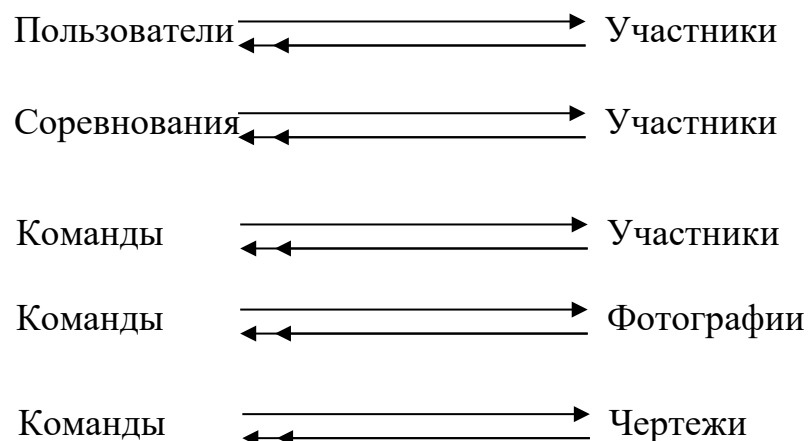


Диаграмма. Взаимосвязи информационных объектов.

Данная диаграмма отражает взаимосвязи информационных объектов предметной области.

Заключение

На основе данных, полученных при анализе предметной области, было сформировано чёткое представление базы данных необходимой для создания связи между мобильным и оконным клиентом. Было составлено детальное описание объектов предметной области. Для каждого объекта был определен набор реквизитов и соответствующий тип данных. Были установлены взаимосвязи между объектами. Составлена схема БД и приведена к 3-ей нормальной форме. В качестве СУБД была выбрана бесплатная СУБД SQLite.

8. Заключение

Таким образом, в рамках курсовой работы были выполнены все поставленные задачи. Был проведён обзор существующих конкурентов «Event.Rocks», «Eventum», «Лига Голдберга», на основе которого была сформирована абстрактная модель разрабатываемого программного средства.

На основе анализа предметной области спроектированы взаимосвязи системы с внешними и внутренними факторами. Взаимосвязи были отражены при помощи диаграмм, каждая из которых показывала определенную точку зрения пользователя и разработчика на проект. Архитектурно-контекстная диаграмма выявила все объекты, окружающие систему и подобъекты, составляющие программное средство. Диаграмма прецедентов выявила все пользовательские функции. Диаграмма потоков данных, ставшая симбиозом предыдущих, позволила разработчику совершить переход к более точному пониманию представления пользователя о будущем программном средстве.

В итоге, благодаря чёткому описанию абстрактного проекта, был произведён переход к конкретной реализации.

Таким образом, в результате данной курсовой работы были выполнены следующие задачи:

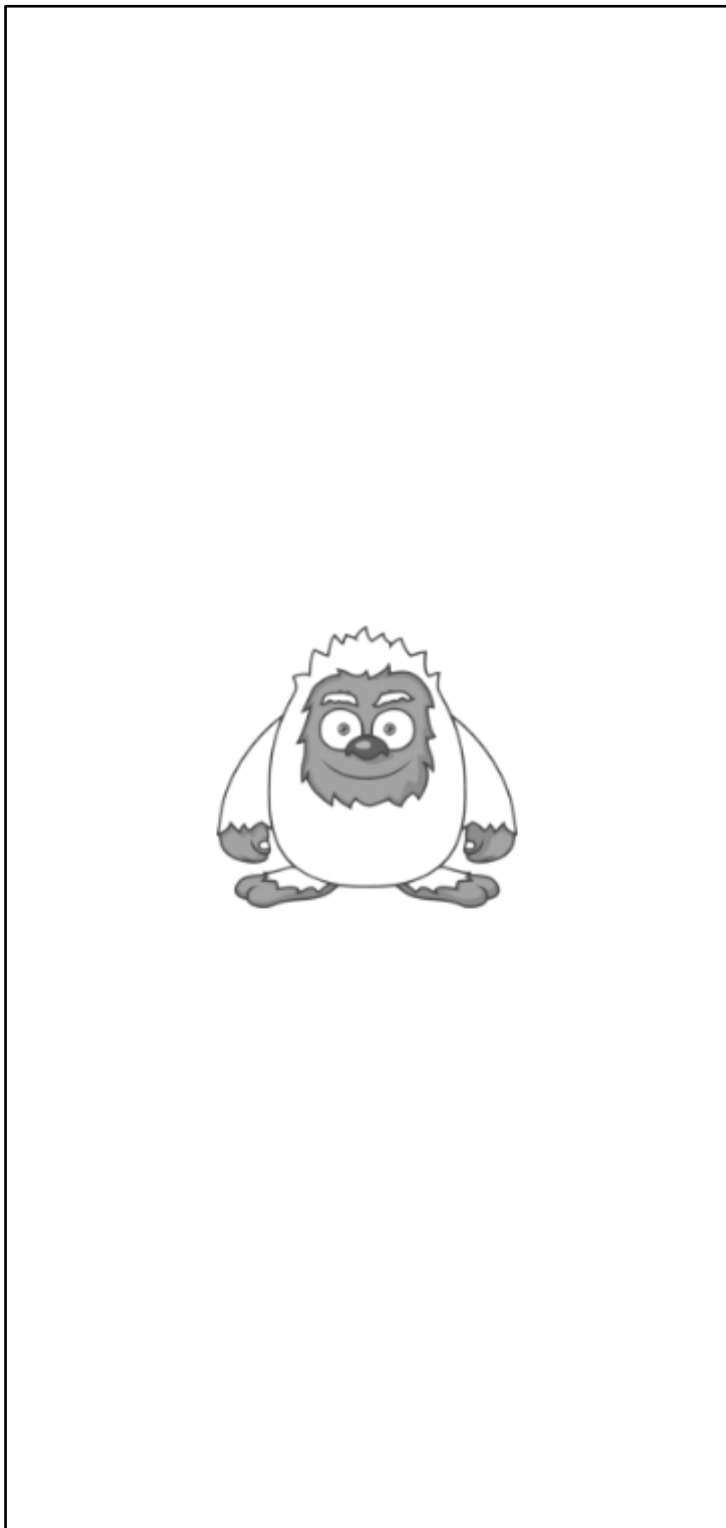
- 1) Проведён обзор наиболее популярных сервисов создания мероприятий.
- 2) Сформулированы требования к программному средству, позволяющее пользователям просматривать и регистрироваться на мероприятия.
- 3) Разработан проект программного средства, пригодный для дальнейшей реализации.
- 4) Разработан программный продукт – Сервис администрирования конкурса по построение машин «Голдберга»

В результате была выполнена поставленная цель курсовой работы - спроектировать оконное и мобильное приложение для автоматизации сервисов и информационной поддержки мероприятий по «Кубку Йети».

9. Список литературы

1. Event.Rocks: [Электронный ресурс]. – URL: <https://event.rocks/>
2. Лига Голдберга: [Электронный ресурс]. – URL: <http://goldbergclub.ru/>

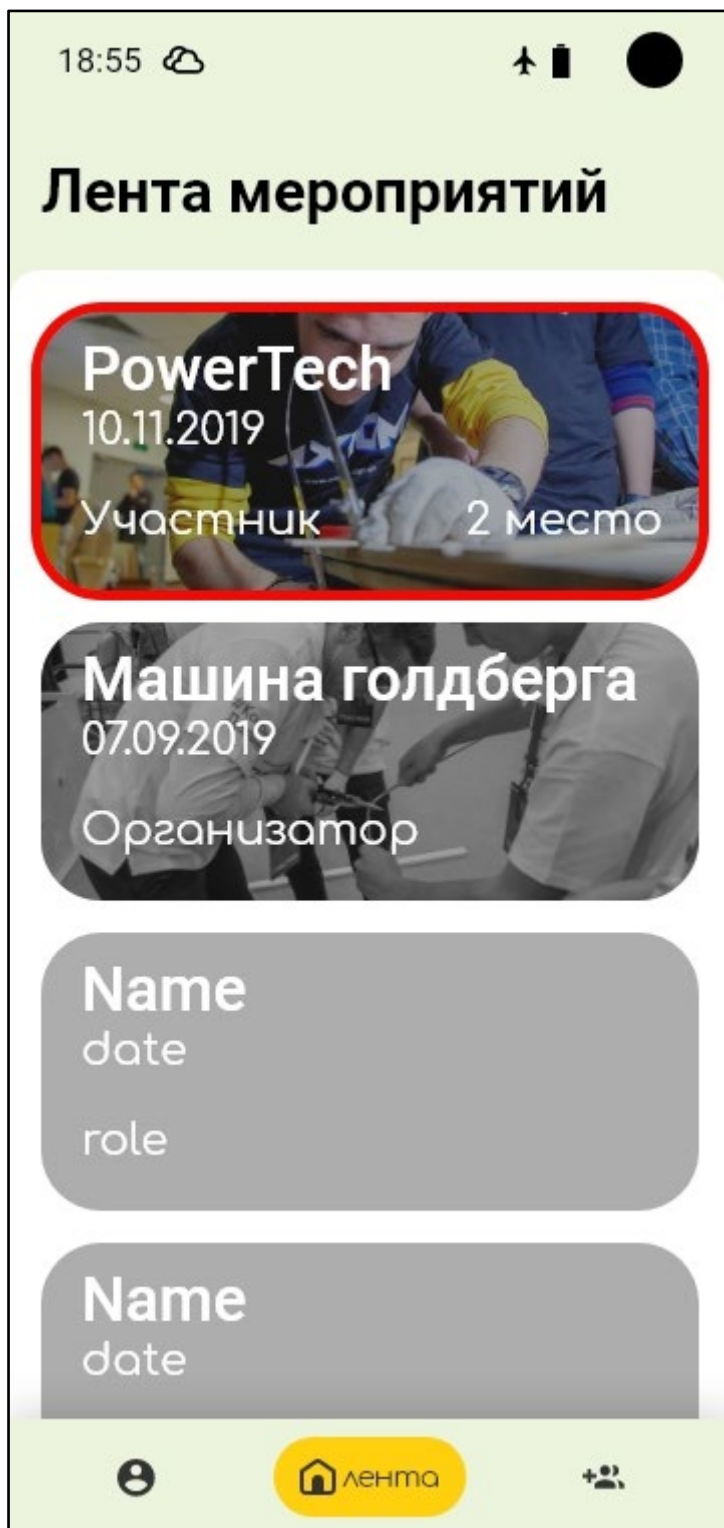
10. Приложение



Приложение № 1. Макет «Стартовый экран»



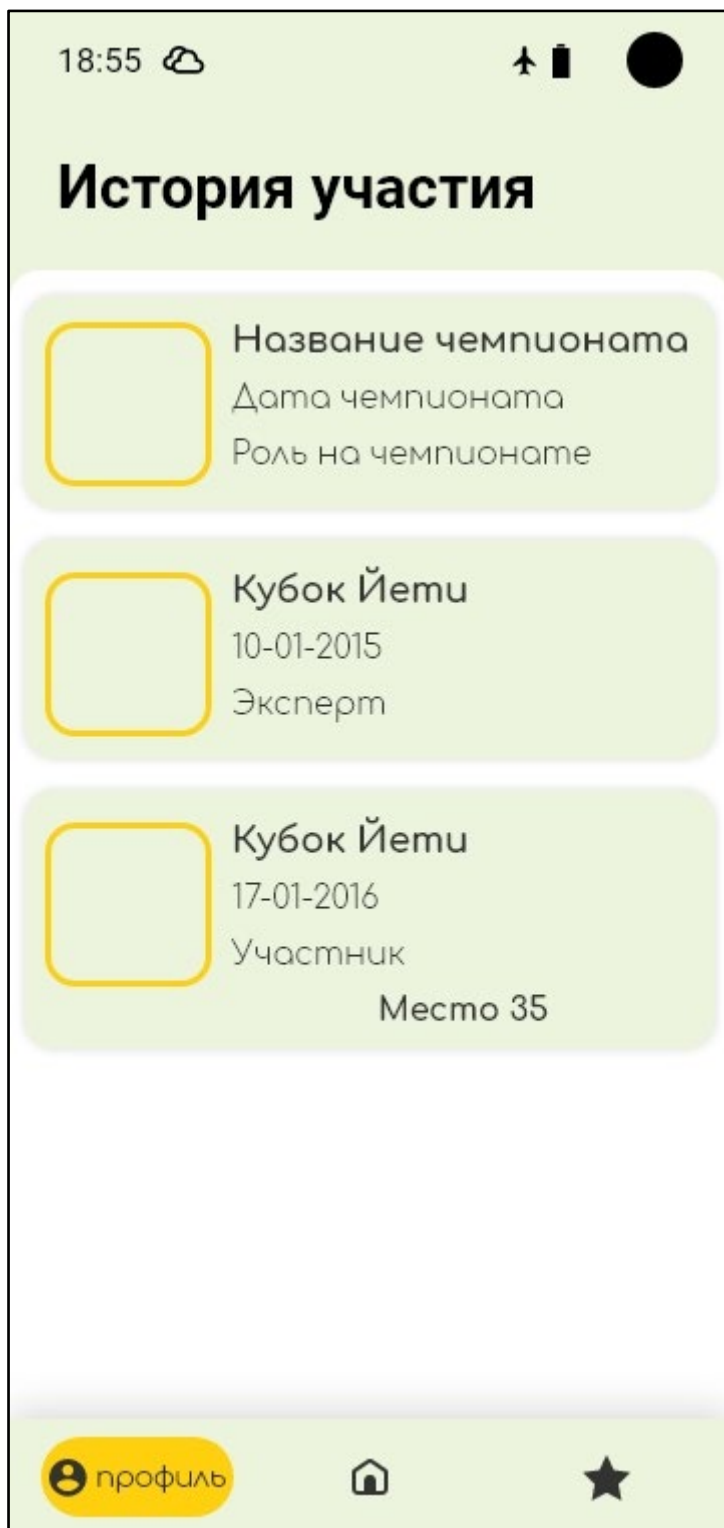
Приложение №2. Макет «Авторизация пользователя»



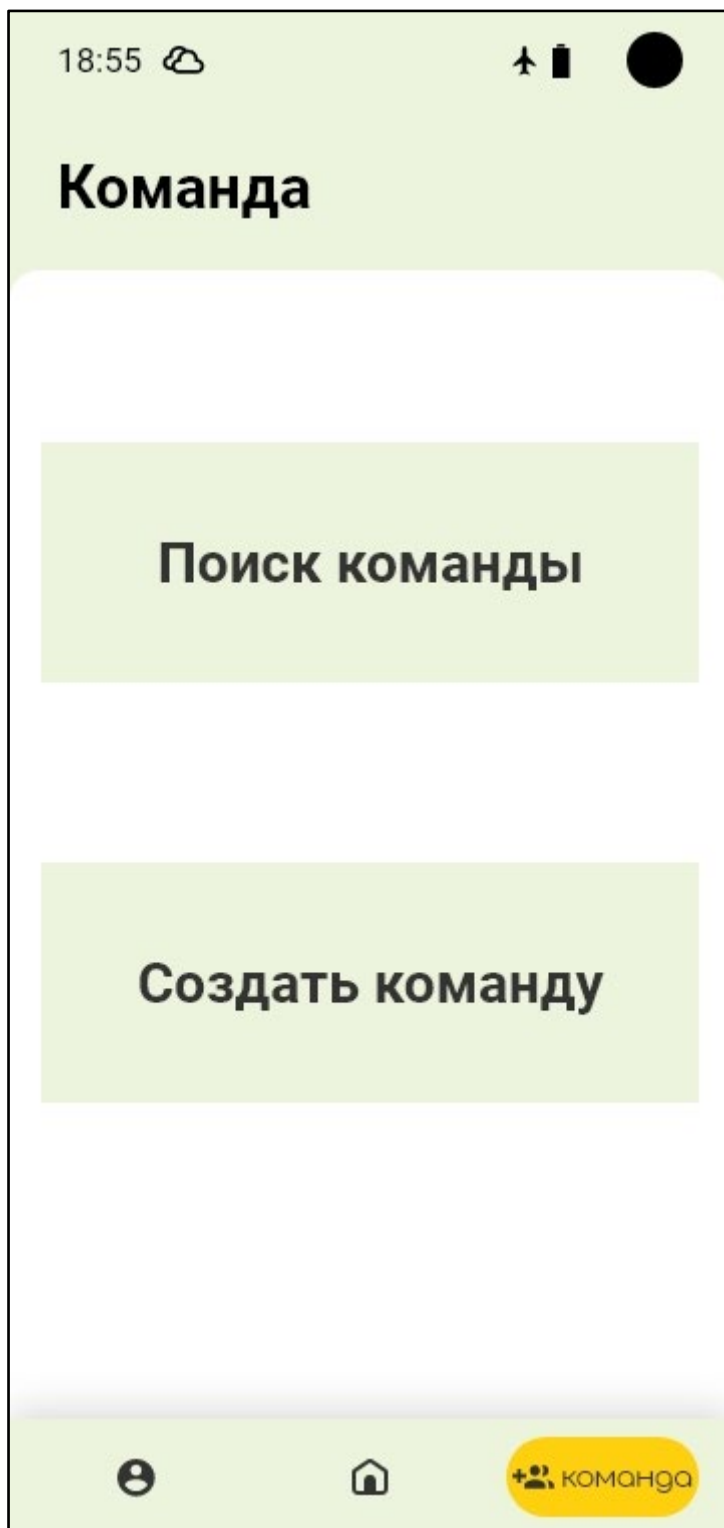
Приложение №3. Макет «лента мероприятий»



Приложение №4. Макет «Профиль участника»



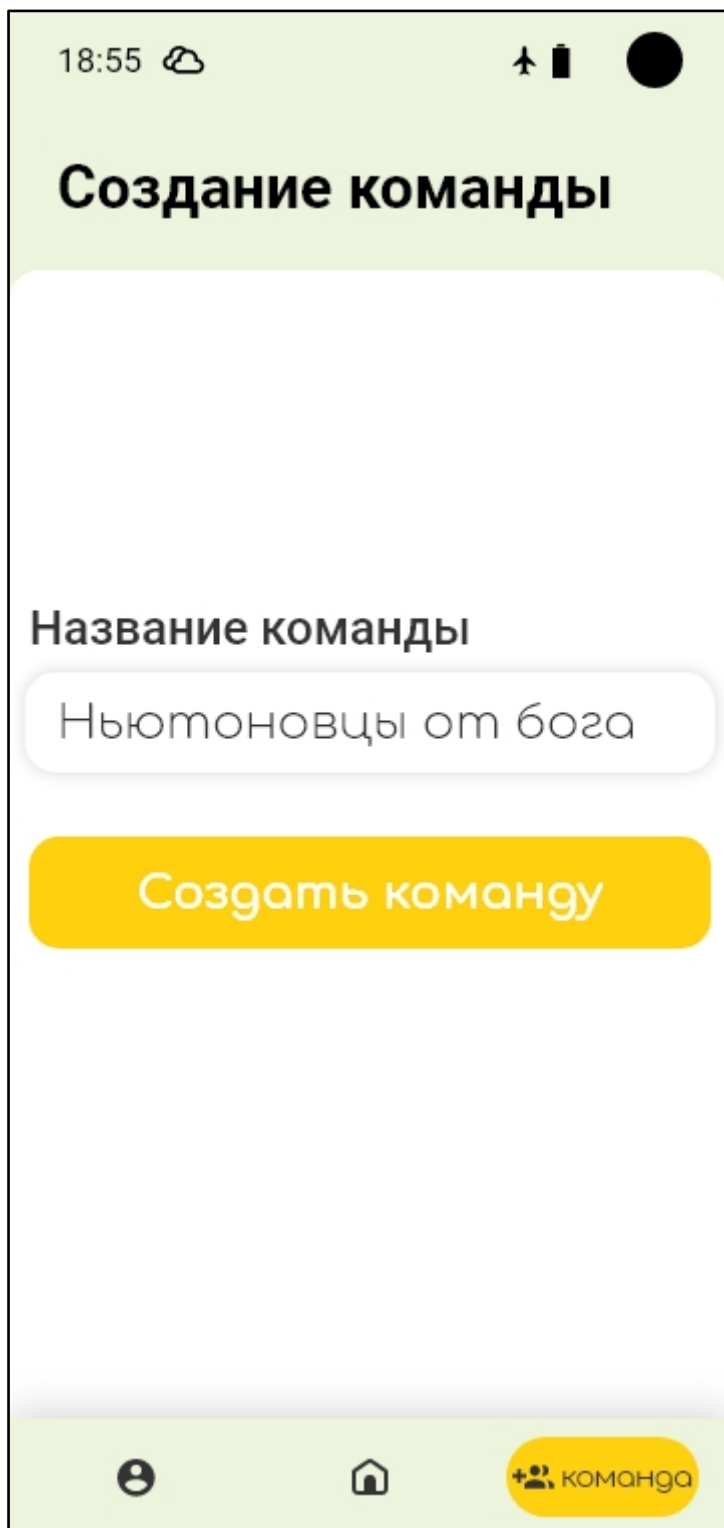
Приложение №5. Макет «История соревнований и достижения участника»



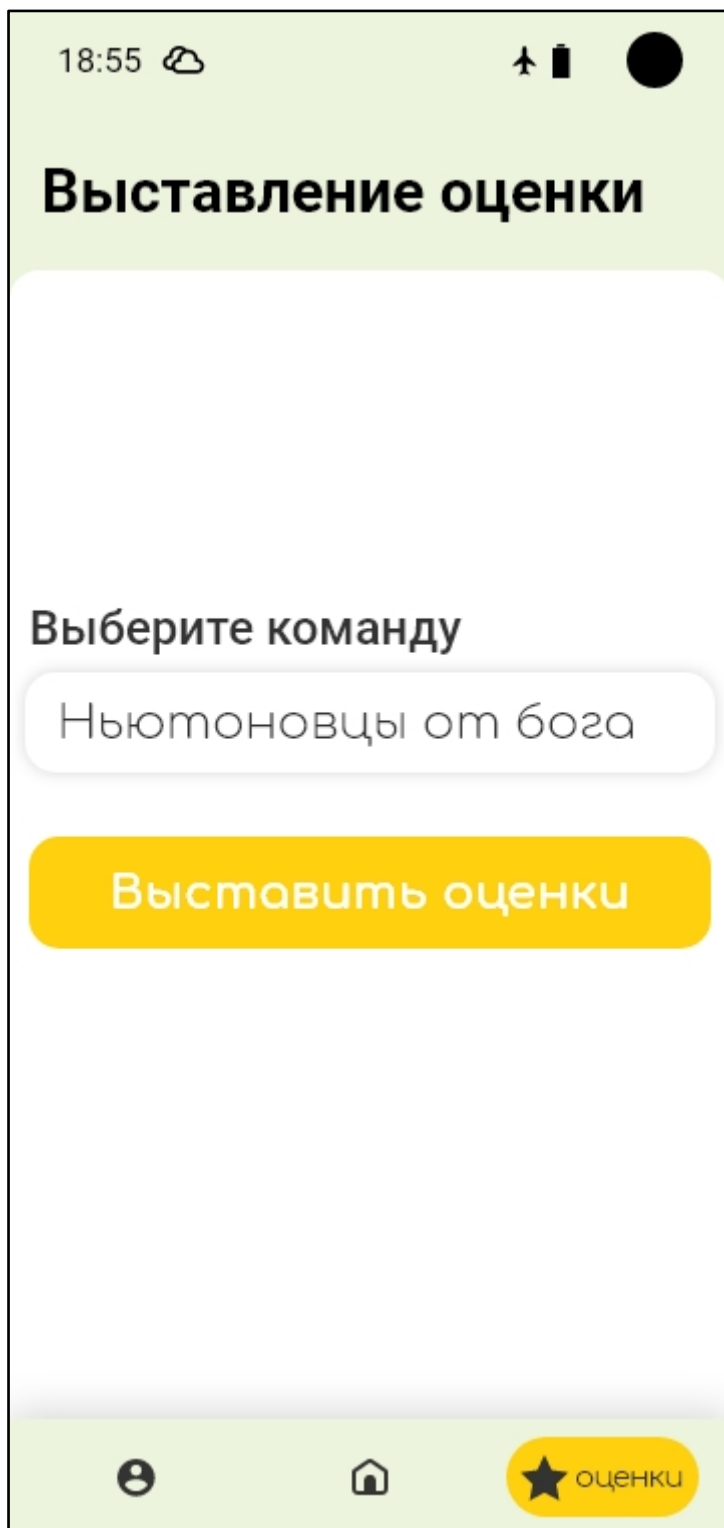
Приложение №6.1. Макет «Поиск или создание команды»



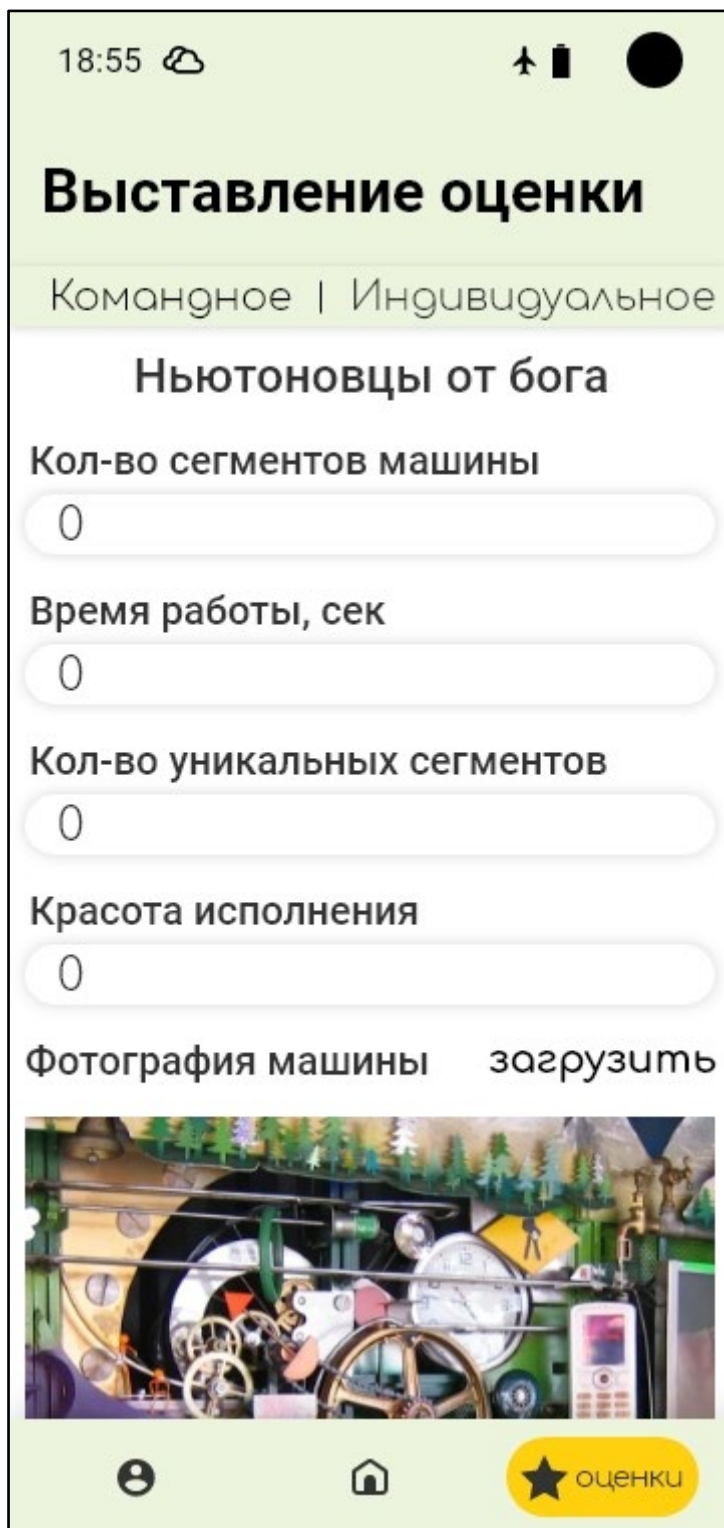
Приложение №6.2. Макет «Поиск команды»



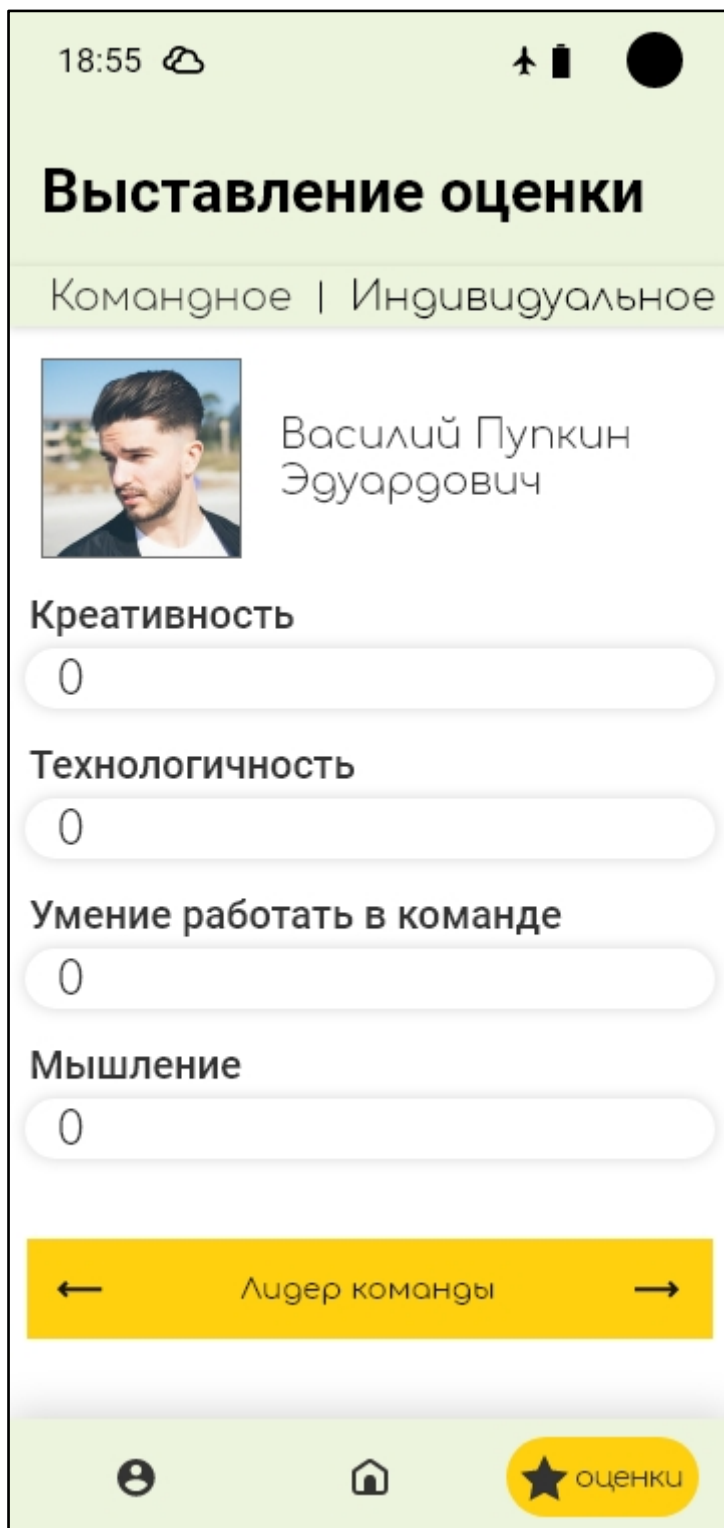
Приложение №6.3. Макет «Создание команды»



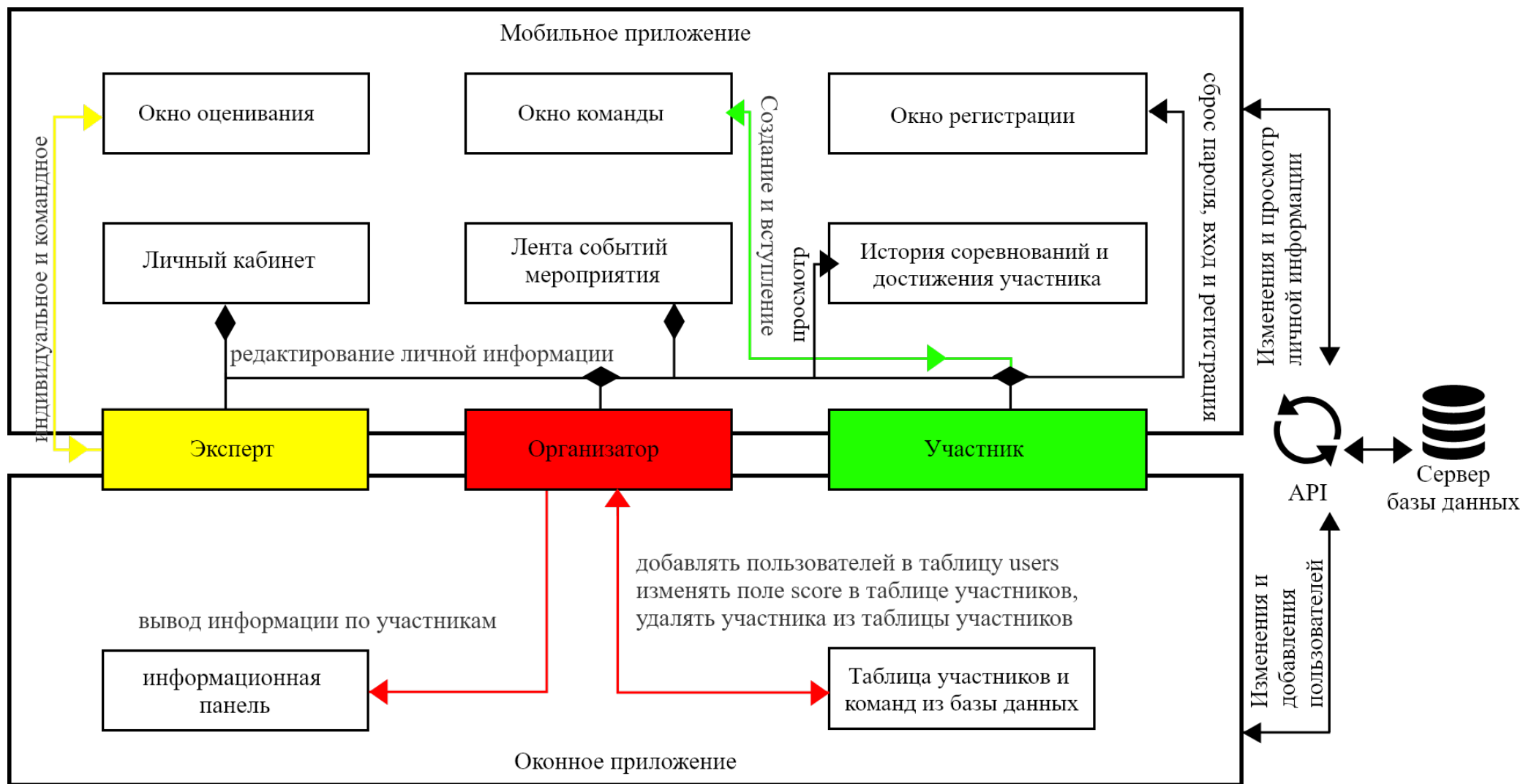
Приложение №7.1. Макет «поиск команды для выставления оценок»



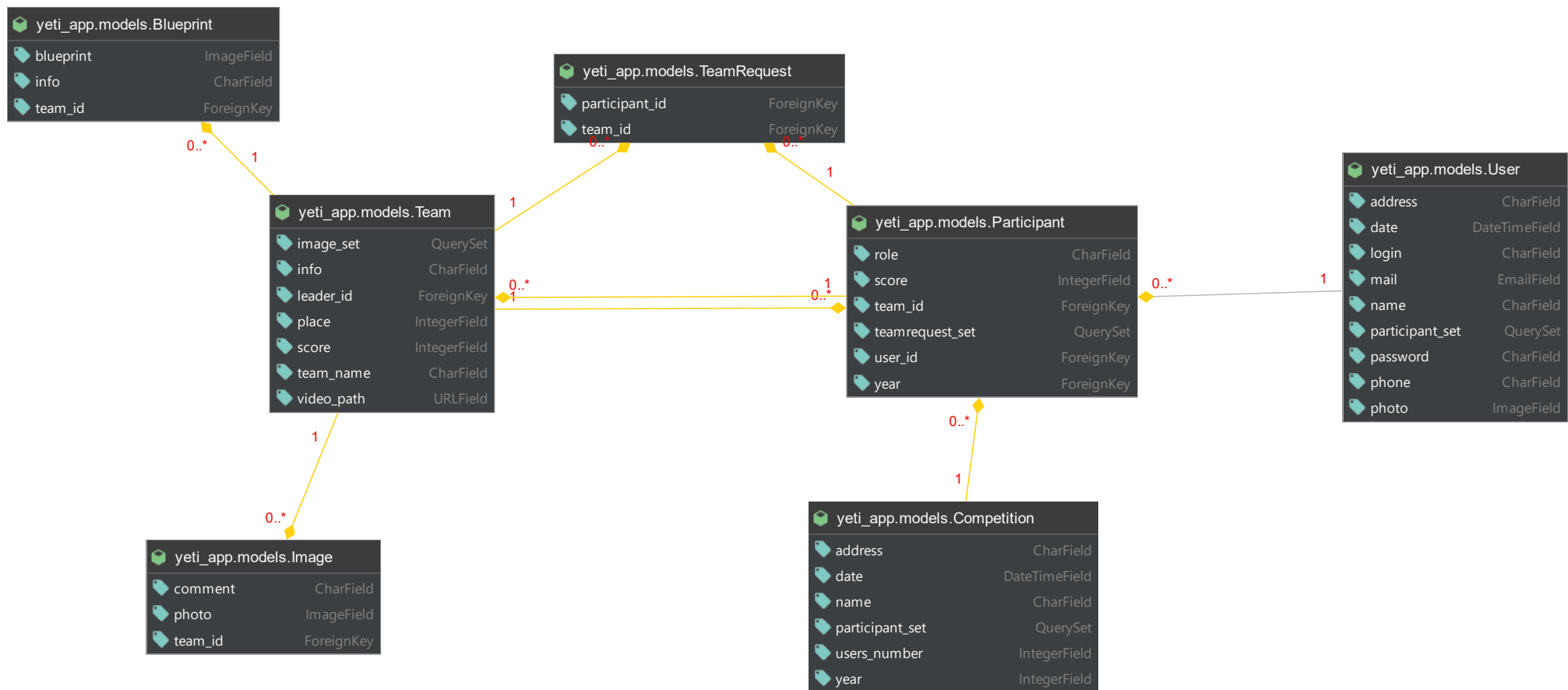
Приложение №7.2. Макет «Выставление оценок команде»



Приложение №7.3. Макет «Выставление оценок участникам команды»



Приложение №8. Архитектурная диаграмма потоков



Приложение №9. ER диаграмма базы данных