

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)
Научно-образовательный центр «Высшая ИТ школа»

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК
Руководитель ООП

д-р. физ.-мат. наук, профессор

О.А. Змеев

подпись

« 11 » июня 2024 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

РАЗРАБОТКА ПЛАТФОРМЫ ДЛЯ СОЗДАНИЯ
И ПРОВЕДЕНИЯ КВИЗОВ

по направлению подготовки 09.03.04 Программная инженерия
направленность (профиль) «Программная инженерия»

Назарова Мария Олеговна

Руководитель ВКР
старший преподаватель
НОЦ «Высшая ИТ школа»

Д.А. Соколов

подпись

« 11 » июня 2024 г.

Автор работы
студент группы № 972002

М.О. Назарова

подпись

« 11 » июня 2024 г.

Министерство науки и высшего образования Российской Федерации.
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)
НОЦ «Высшая ИТ школа»

УТВЕРЖДАЮ
руководитель ООП
д-р.физ.-мат. наук, профессор
О.А.Змеев
« 16 » февраля 2024 г.

ЗАДАНИЕ

по выполнению выпускной квалификационной работы бакалавра обучающемуся
Назаровой Марии Олеговне

(Ф.И.О. обучающегося)

по направлению подготовки Программная инженерия, направленность «Программная
инженерия»

1. Тема выпускной квалификационной работы бакалавра
Разработка платформы для создания и проведения квизов

2. Срок сдачи обучающимся выполненной выпускной квалификационной работы:
а) в учебный офис – « 11 » февраля 2024 г.
б) в ГЭК – « 14 » февраля 2024 г.

3. Исходные данные к работе:

Цель работы: разработка платформы для создания и проведения квизов.

Задачи:

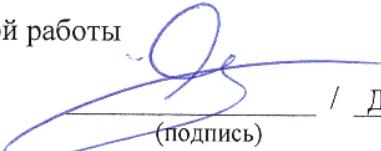
- 1) анализ требований;
- 2) определение необходимого набора инструментов;
- 3) проектирование и разработка серверной части платформы;
- 4) проектирование и разработка клиентского приложения для организаторов;
- 5) проектирование и разработка клиентского приложения для участников;

Ожидаемые результаты: проанализированы требования, определены инструменты для разработки, спроектированы и разработаны серверная часть платформы и клиентские приложения для участников и организаторов.

цели и задачи ВКР, ожидаемые результаты

Организация, по тематике которой выполняется работа

Руководитель выпускной квалификационной работы
старший преподаватель
НОЦ «Высшая ИТ школа»
(должность, место работы)


(подпись) / Д.А.Соколов

(И.О. Фамилия)

Задание принял к исполнению
« 13 » февраля 2024 г
(дата)


(подпись) / М.О.Назарова

(И.О. Фамилия)

АННОТАЦИЯ

Выпускная квалификационная работа: 45 стр., 50 рис., 10 источников.

REACT.JS, NODE.JS, ВЕБ-ПРИЛОЖЕНИЕ, JAVASCRIPT, REDUX, ANT DESIGN,
РАЗРАБОТКА.

Цель работы – разработка платформы для создания и проведения квизов.

Результаты работы – проведен анализ требований, определен необходимый набор инструментов, спроектированы и разработаны серверная часть платформы и клиентские приложения для организаторов и участников.

ОГЛАВЛЕНИЕ

Перечень условных обозначений, символов, сокращений, терминов	4
Введение	5
1 Анализ требований.....	6
1.1 Пользователи системы.....	6
1.2 Функциональные требования	6
1.2.1 Функциональные требования приложения для организаторов.....	6
1.2.2 Функциональные требования приложения для участников.....	6
1.3 Варианты использования приложения.....	7
1.3.1 Вариант использования «Создать квиз»	8
1.3.2 Вариант использования «Запустить игру».....	8
1.3.3 Вариант использования «Принять участие в игре»	9
1.4 Модель предметной области	10
2 Описание технологий и инструментов разработки	12
2.1 Node.js	12
2.2 Express.js.....	12
2.3 Socket.io.....	12
2.4 MongoDB	13
2.5 TypeScript	13
2.6 React.....	13
2.7 Redux	14
2.8 Ant Design.....	15
3 Реализация.....	16
3.1 Реализация клиентского приложения для организаторов	16
3.1.1 Авторизация в приложении.....	17
3.1.2 Реализация конструктора квизов.....	17
3.1.3 Реализация проведения квиза.....	30

3.2 Реализация клиентского приложения для участников	35
3.2.1 Реализация расписания квизов	35
3.2.2 Реализация участия в игре	38
3.3 Реализация серверной части платформы	41
3.3.1 Реализация запросов	42
3.3.2 Реализация проверки ответов	42
3.3.3 Реализация сортировки итоговых результатов	43
Заключение.....	44
Список использованных источников и литературы	45

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ, СИМВОЛОВ, СОКРАЩЕНИЙ, ТЕРМИНОВ

Квиз – командная интеллектуальная игра, в которой участники за ограниченное время отвечают на вопросы на определенную тему.

CRUD (Create, Read, Update, Delete) – четыре базовые операции над сущностями: создание, чтение, редактирование, удаление.

DOM (Document Object Model) – структурированное представление веб-страницы в виде дерева объектов.

ВИ – варианты использования.

АЗВИ – архитектурно значимые варианты использования.

Компонент – переиспользуемый блок кода.

Node JS – серверная среда выполнения JavaScript.

Express – фреймворк для создания веб-приложений на Node.js.

MongoDB – документо-ориентированная база данных.

WebSocket – протокол, который обеспечивает двустороннюю связь между веб-браузером и сервером для обмена данными в режиме реального времени.

JavaScript – высокоуровневый, интерпретируемый язык программирования, широко используемый для создания динамических веб-страниц и веб-приложений.

React – библиотека JavaScript для создания пользовательских интерфейсов.

Redux – библиотека JavaScript для управления состоянием приложения.

TypeScript – расширение языка JavaScript, добавляющее статическую типизацию.

Ant Design – библиотека готовых компонентов и стилей для создания пользовательского интерфейса.

HTML – язык разметки для создания веб-страниц.

Хуки – функции React, позволяющие использовать состояние и другие функциональности React в компонентах.

API – интерфейс, который позволяет взаимодействовать с другим программным обеспечением.

HTTP – протокол передачи гипертекста.

ВВЕДЕНИЕ

В Высшей ИТ-школе часто проводятся квизы на различные тематики, для организации которых необходима значительная подготовка. Процесс организации состоит из большого количества этапов. Необходимо составить задания, сделать презентацию, форму регистрации, бланки для ответов и таблицу для проверки. В настоящее время все это делается вручную.

Каждая презентация – это около трехсот слайдов, на составление которых уходит не менее двух часов. На слайдах необходимо вставить вопросы, настроить таймеры, анимации и переходы. Бланки для ответов также делаются вручную, что допускает шанс ошибки, например, в количестве вопросов. Бланки необходимо распечатать на каждую команду, разрезать и раздать, что требует временных и финансовых затрат и повышает шанс ошибиться с количеством бланков.

Процесс проведения квиза также включает множество действий и ролей. Ведущий переключает слайды с вопросами, участники записывают ответы на бланках, затем бланки собираются, проверяются и выставляются в таблицу. Поскольку ответы проверяются и проставляются вручную, это занимает большое количество времени, которое прямо пропорционально количеству команд. Из-за этого приходится делать большие перерывы, чтобы успеть всё проверить. Также ручная проверка допускает возможность случайной ошибки.

В связи с этими проблемами возникла потребность в платформе, которая позволит автоматизировать часть действий при подготовке и проведении квиза, тем самым значительно сократит затрачиваемое время и упростит процессы.

Целью работы является разработка платформы для создания и проведения квизов.

Для достижения поставленной цели было необходимо решить следующие задачи:

- 1) анализ требований;
- 2) определение необходимого набора инструментов;
- 3) проектирование и разработка серверной части платформы;
- 4) проектирование и разработка клиентского приложения для организаторов;
- 5) проектирование и разработка клиентского приложения для участников.

1 Анализ требований

1.1 Пользователи системы

Для разрабатываемой платформы были выделены следующие роли:

1. Организатор – пользователь, осуществляющий управление созданием и проведением квизов.
2. Участник – пользователь, принимающий участие в квизе.

В зависимости от роли пользователю будет доступен различный функционал.

1.2 Функциональные требования

Для разрабатываемой платформы были выявлены следующие функциональные требования.

1.2.1 Функциональные требования приложения для организаторов

1. Возможность создать, редактировать, удалить квиз, посмотреть его данные.
2. Возможность создать, редактировать, удалить раунд, посмотреть его данные.
3. Возможность создать, редактировать, удалить задание, посмотреть его данные.
4. Возможность посмотреть список существующих квизов.
5. Возможность посмотреть список раундов квиза.
6. Возможность посмотреть список вопросов раунда.
7. Возможность создать, редактировать, удалить игру по составленному квизу.
8. Возможность запустить игру.
9. Возможность добавить игру в расписание.
10. Возможность удалить игру из расписания.
11. Возможность скачать обложку квиза.
12. Возможность посмотреть список зарегистрированных команд.
13. Возможность посмотреть ответы участников.
14. Возможность проверить ответы участников.
15. Возможность посмотреть таблицу результатов.
16. Возможность опубликовать таблицу результатов.

1.2.2 Функциональные требования приложения для участников

1. Возможность посмотреть расписание игр.
2. Возможность посмотреть краткую информацию о предстоящей игре.

3. Возможность зарегистрироваться на игру.
4. Возможность присоединиться к игре.
5. Возможность отправить ответы.
6. Возможность посмотреть результаты прошедших игр.

1.3 Варианты использования приложения

На основе указанных ранее функциональных требований и ролей системы были сформулированы варианты использования платформы. На рисунке 1 представлены ВИ для роли Организатор, на рисунке 2 представлены ВИ для роли Участник.

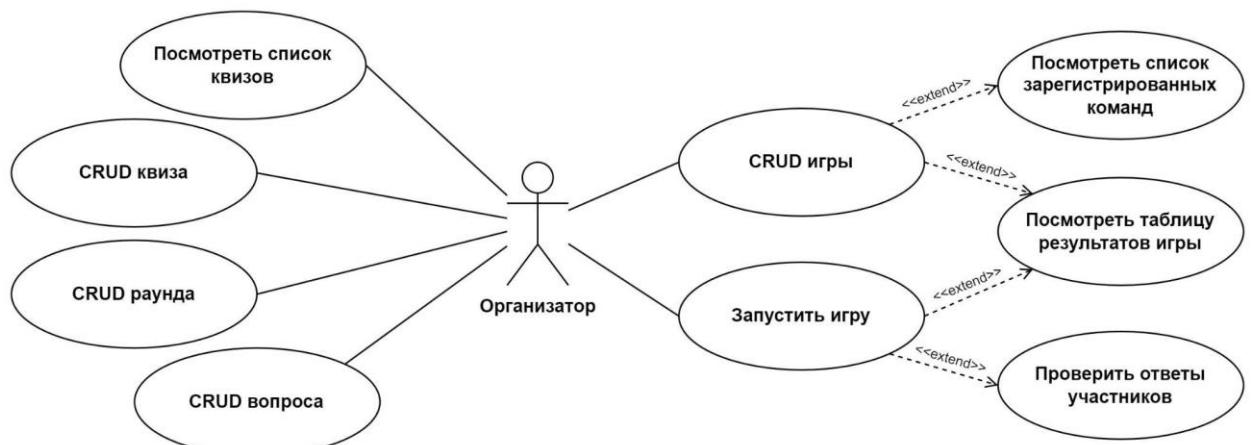


Рисунок 1 – Варианты использования для роли Организатор

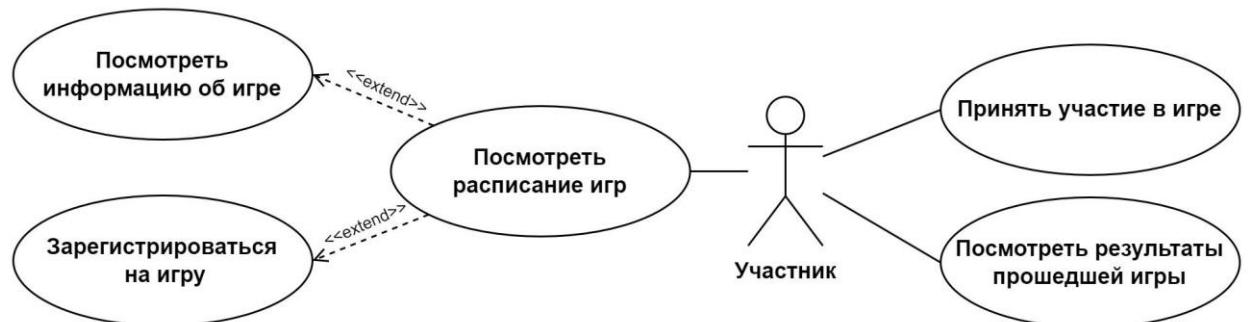


Рисунок 2 – Варианты использования для роли Участник

На основе имеющихся вариантов использования были выделены АЗВИ – наиболее важные с точки зрения реализации архитектуры варианты использования:

- 1) создать квиз;
- 2) запустить игру;
- 3) принять участие в игре.

Рассмотрим их более подробно.

1.3.1 Вариант использования «Создать квиз»

При создании квиза и дальнейшем создании раундов и вопросов пользователю будет доступен предпросмотр слайдов, которые он создаёт. При создании самого квиза, слайд – это его обложка, при создании раунда – слайд с правилами, при создании вопроса – слайды с вопросом и ответом. Пользователю необходимо ввести необходимые данные в форму, при каждом изменении происходит обновление слайда. На рисунке 2 представлен процесс создания слайда в виде диаграммы активности.

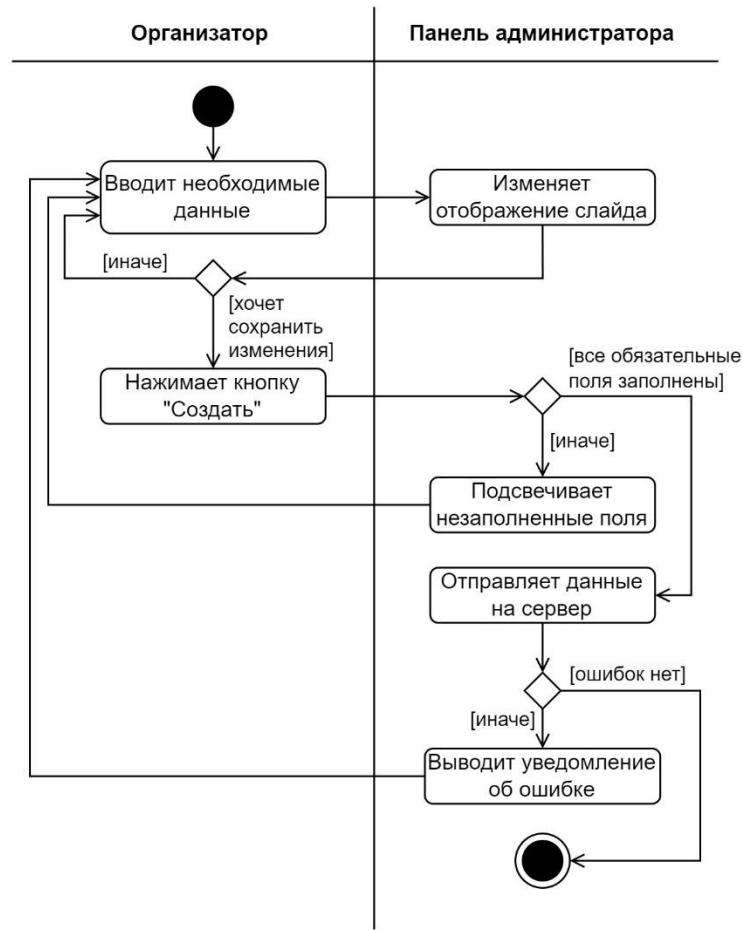


Рисунок 3 – Процесс создания слайда

1.3.2 Вариант использования «Запустить игру»

При запуске демонстрации игры пользователю открывается страница показа слайдов на полный экран. Слайды отрисовываются поочередно, когда пользователь нажимает кнопку «Далее». Сначала показывается обложка, затем слайды с названиями команд-участников, правила квиза. После системы получает данные раундов, и по очереди для каждого раунда отрисовывает вопросы, повтор вопросов (если он есть) и ответы. Если после раунда есть перерыв, то показывается такой слайд. В конце после финального раунда

выводится слайд с результатами. Когда слайды закончились, организатор выходит из демонстрации.

На рисунке 4 представлен процесс запуска демонстрации игры в виде диаграммы активности.

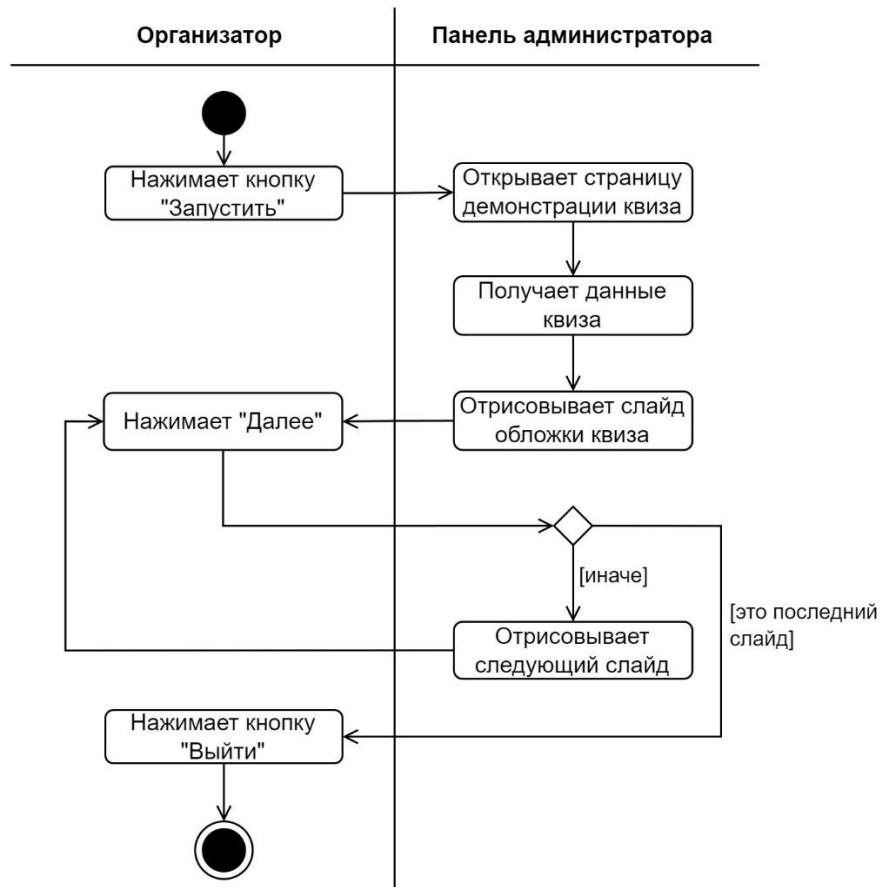


Рисунок 4 – Процесс запуска демонстрации игры

1.3.3 Вариант использования «Принять участие в игре»

После запуска игры Организатором участники могут присоединиться к ней. На слайде квиза отображается код, который участнику необходимо ввести. Затем при переключении организатором слайдов у участника также меняется отображение страницы. На слайдах с вопросами он может ввести ответ и отправить их на сервер, который проверяет ответы и отправляет их администратору. На рисунке 5 представлен процесс игры в виде диаграммы активности.

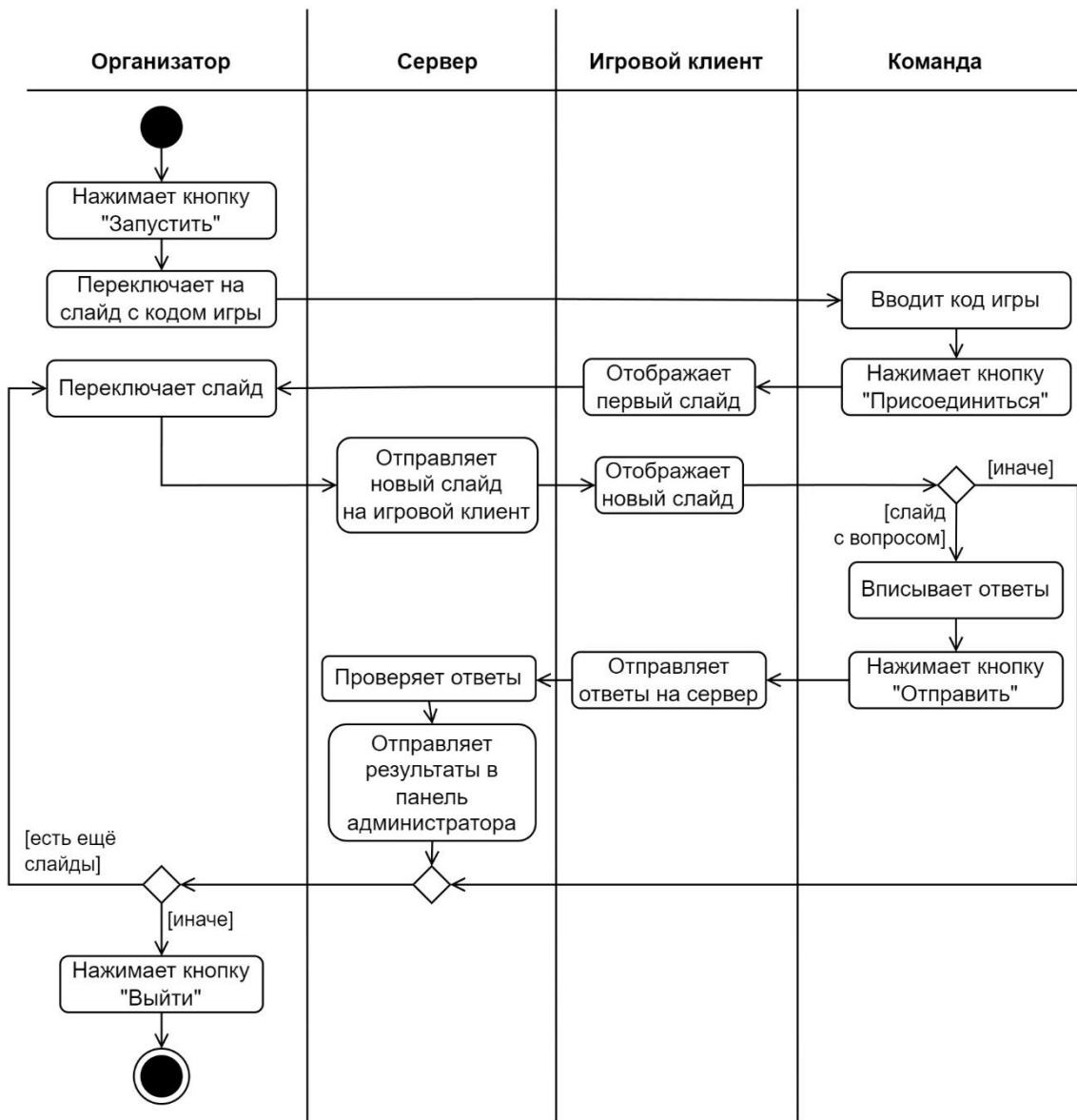


Рисунок 5 – Процесс игры

1.4 Модель предметной области

В ходе анализа были выделены основные сущности системы и спроектирована модель предметной области, представленная на рисунке 6.

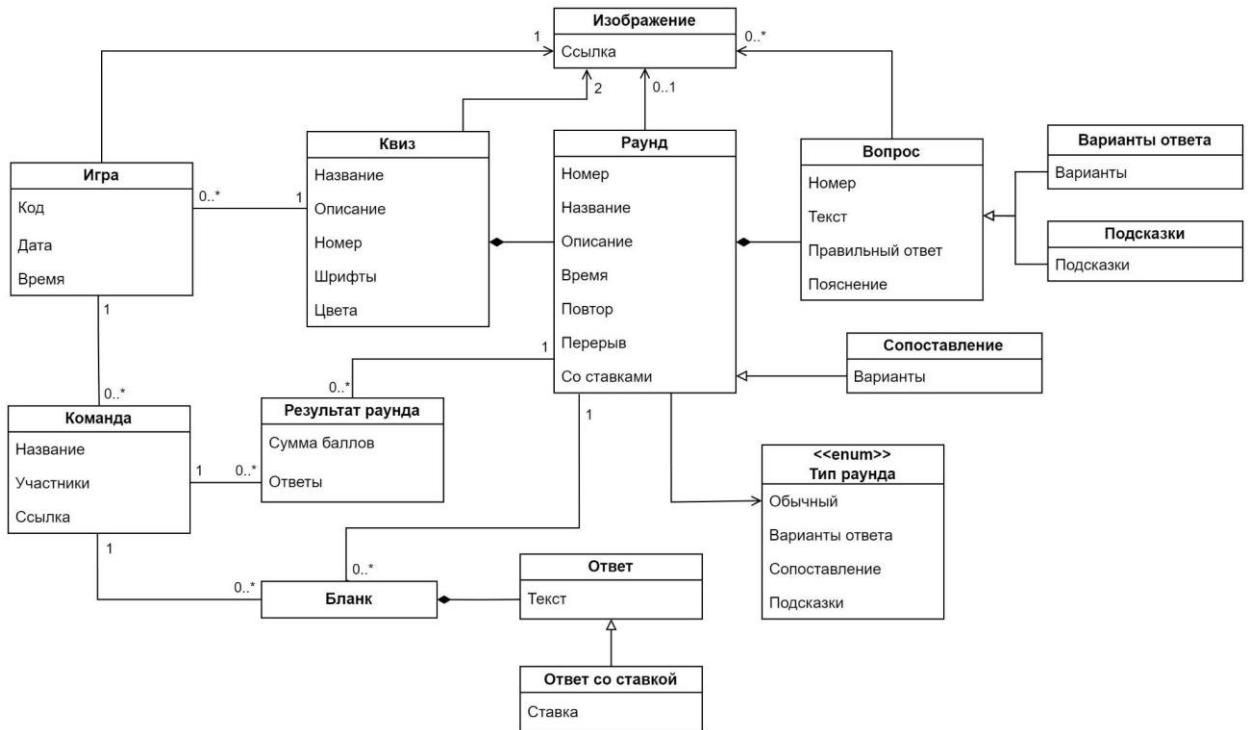


Рисунок 6 – Модель предметной области

Пояснения к диаграмме.

1. Квиз – сущность, содержащая основную информацию о квизе.
2. Раунд – сущность, содержащая информацию о раунде квиза.
3. Тип раунда – перечисление с фиксированным типом полей, определяет, какие вопросы будут в раунде.
4. Вопрос – сущность, содержащая информацию о вопросе раунда.
5. Игра – сущность, содержащая информацию об игре квиза.
6. Изображение – сущность, содержащая информацию о файле квиза/раунда/вопроса/игры.
7. Команда – сущность, содержащая информацию о команде, зарегистрированной на игру.
8. Бланк – сущность, содержащая ответы команды на раунд.
9. Ответ – сущность, содержащая ответ команды.
10. Результат раунда – сущность, содержащая результат команды в раунде.

2 Описание технологий и инструментов разработки

2.1 Node.js

Для реализации серверной части платформы был выбран Node.js – это среда выполнения JavaScript, позволяющая исполнять его на серверной стороне.

Основные преимущества Node.js:

1. Благодаря асинхронной модели выполнения, Node.js способен обрабатывать большое количество запросов одновременно, что делает его идеальным для реализации масштабируемых систем.
2. Позволяет использовать один и тот же язык программирования (JavaScript) как на клиентской стороне, так и на серверной стороне, что упрощает разработку и поддержку кода.
3. Имеет большое количество библиотек для упрощения разработки приложений.

2.2 Express.js

Для создания API был использован фреймворк Express.js. Он является одним из самых популярных фреймворков для Node.js благодаря своей простоте и удобству использования.

Основные преимущества Express:

1. Облегчает создание маршрутов для веб-приложения, позволяя определять обработчики для различных запросов (GET, POST, PUT, DELETE и другие.).
2. Использует концепцию middleware, которая позволяет добавлять слои обработки между получением запроса и отправкой ответа. Это упрощает создание сложных приложений и обеспечивает гибкость в управлении потоком выполнения.
3. Легко интегрируется с другими популярными библиотеками и фреймворками для Node.js, такими как MongoDB (для работы с базами данных) и Socket.io (для работы с WebSockets), которые также используются в проекте.

2.3 Socket.io

Для реализации веб-сокетов была выбрана библиотека Socket.io.

Основные преимущества:

1. Позволяет отправлять сообщения всем подключенными клиентам с помощью одной операции.
2. Поддерживает автоматическое переподключение при разрыве соединения.

2.4 MongoDB

В качестве базы данных для хранения была выбрана MongoDB. Это документо-ориентированная NoSQL база данных, которая позволяет хранить данные в формате документов, используя JSON-подобные структуры.

Основные преимущества MongoDB:

1. Не требует строгой схемы данных, что позволяет хранить различные типы данных в одной коллекции без необходимости определения заранее всех полей и их типов, что важно при работе с данными, которые имеют разную структуру.
2. Возможность быстро и легко масштабировать систему. Если информация в базе должна измениться, например, планируется добавить новое поле, не нужно радикально перекраивать структуру всей БД.

2.5 TypeScript

В качестве языка программирования был выбран TypeScript. Это язык программирования, который является оберткой над JavaScript, добавляющей статическую типизацию и дополнительные возможности. TypeScript компилируется в обычный JavaScript и может работать на любой платформе, где поддерживается JavaScript.

Основные отличия TypeScript от JavaScript:

1. Позволяет явно указывать типы данных для переменных, функций, параметров и возвращаемых значений, что помогает выявлять ошибки на этапе разработки и улучшает читаемость кода.
2. Поддерживает объектно-ориентированное программирование, включая принципы наследования, интерфейсы, абстрактные классы и другие концепции, что делает его более мощным и гибким по сравнению с JavaScript.

2.6 React

С появлением более сложных и интерактивных веб-приложений, разработчики сталкиваются с необходимостью эффективно управлять состоянием приложения, его компонентами и взаимодействием с пользователем. Библиотека React стала одним из самых популярных инструментов для разработки динамических пользовательских интерфейсов, поэтому для разработки приложения была выбрана она.

Основными преимуществами React являются

1. Компонентный подход.

React разделяет пользовательский интерфейс на независимые и переиспользуемые компоненты, каждый из которых имеет свое состояние (state) и входные данные (props). Родительский компонент передает props в дочерний компонент, тем самым влияет на его состояние. При изменении состояния происходит перерисовка (re-rendering) компонента.

2. Виртуальный DOM.

React использует виртуальное представление DOM (document object model) для оптимизации процесса обновления пользовательского интерфейса. При изменении состояния компонента React создает виртуальный DOM, сравнивает его с реальным и находит самый оптимальный способ изменения страницы. Это позволяет уменьшить время на перерисовку и повысить производительность приложения.

3. JSX/TSX.

JSX – расширение синтаксиса JavaScript, позволяющие описывать компоненты в виде комбинации JavaScript и HTML-подобного кода, что делает разработку более удобной и интуитивной. Аналогично для TypeScript есть расширение TSX.

4. Хуки.

Хуки – специальные функции, позволяющие получать доступ к состоянию React без создания классов. Основные хуки, которые используются в приложении:

- 1) useState – позволяет управлять состоянием элементов;
- 2) useEffect – принимает функцию, которая запускается при каждом рендеринге;
- 3) useMemo – принимает функцию и массив зависимостей, возвращает memoизированное значение, которое изменяется только тогда, когда значение какой-либо из зависимостей изменилось;
- 4) useLocation – позволяет получить информацию о текущей странице;
- 5) useCallback – принимает на вход колбэк и массив зависимостей, возвращает memoизированную версию колбэка, который изменяется только при изменении значения одной из зависимостей.

2.7 Redux

Для работы с данными был выбран Redux. Это библиотека управления состоянием, основанная на принципах одностороннего потока данных. Redux позволяет эффективно управлять состоянием приложения, упрощая разработку и отладку кода.

Основные преимущества Redux

1. Благодаря одностороннему потоку данных и четкому управлению состоянием, Redux делает поведение приложения предсказуемым и легко воспроизводимым.

2. Состояние приложения хранится в одном месте - в хранилище (store), что делает управление данными более простым и эффективным.
3. Благодаря принципам Redux, логика управления состоянием может быть легко переиспользована в различных частях приложения.
4. Позволяет легко отслеживать изменения состояния приложения, что упрощает процесс отладки и устранение ошибок.
5. Благодаря архитектуре Redux, приложение можно легко расширять и добавлять новые функциональности, не нарушая целостность кода.

Основными компонентами Redux являются

1. Хранилище (store): хранит состояние приложения.
2. Действия (actions): набор информации, который исходит от приложения к хранилищу и указывает, что именно нужно сделать.
3. Создатели действий (action creators): функции, которые создают действия.
4. Reducer: функция, которая получает действие и в соответствии с этим действием изменяет состояние хранилища.

Из компонентов React действия передается в Reducers, которые изменяют состояние.

Затем компоненты React применяют обновленное состояние.

2.8 Ant Design

В качестве библиотеки компонентов был выбран Ant Design. Он предлагает широкий выбор готовых компонентов, которые могут быть легко интегрированы в веб-приложения.

Основными его преимуществами являются:

1. Стильный и современный дизайн: компоненты Ant Design имеют профессиональный и современный вид, что способствует созданию приятного пользовательского интерфейса.
2. Обширный набор компонентов: Ant Design предлагает широкий выбор готовых компонентов для различных случаев использования, что ускоряет процесс разработки.
3. Удобство в использовании: компоненты Ant Design легко настраиваемы, имеют простой и интуитивно понятный API, что делает их удобными в использовании для разработчиков.

3 Реализация

На рисунке 7 представлены основные компоненты разработанной платформы. Она состоит из серверного приложения и двух клиентских приложений: игровой клиент – приложение для участников, панель администратора – приложение для организаторов. Взаимодействие сервера и клиентских приложений осуществляется через протокол HTTP и веб-сокеты. Веб-сокеты необходимы для организации обмена данными между панелью администратора, сервером и игровым клиентом в режиме реального времени.

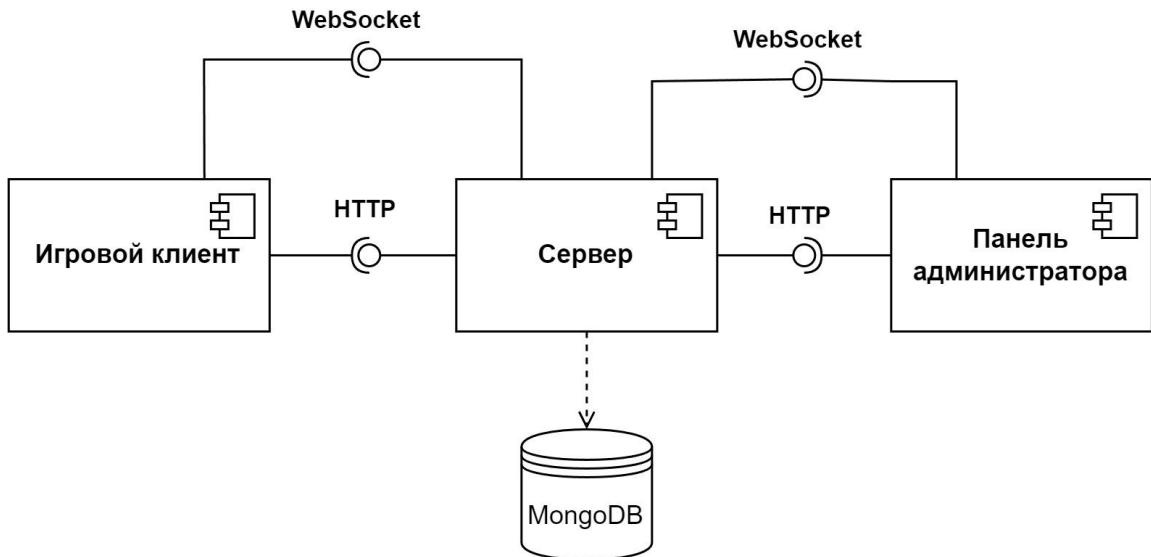


Рисунок 7 – Компоненты платформы

3.1 Реализация клиентского приложения для организаторов

Верхнеуровневая структура приложения для организаторов, содержащая основные пакеты и связи между ними, приведена на рисунке 8 в виде диаграммы пакетов.

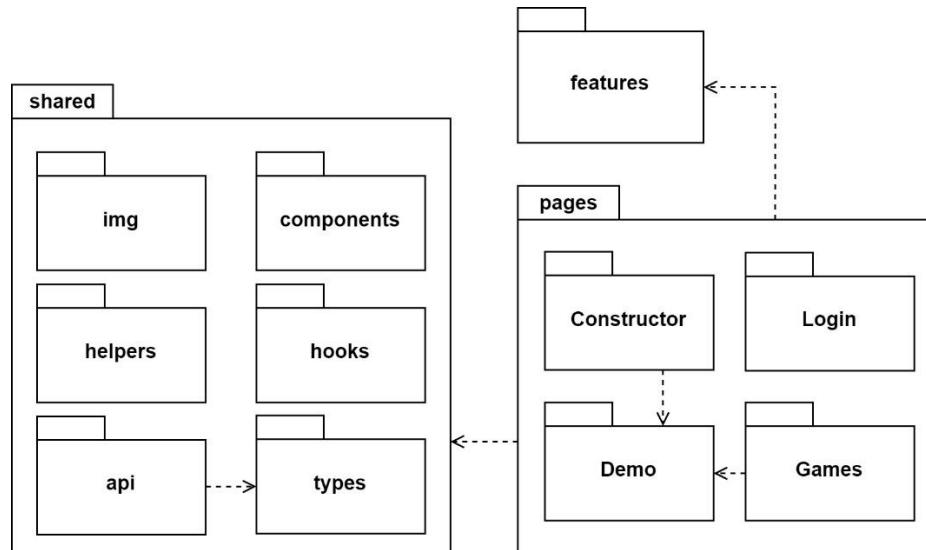


Рисунок 8 – Диаграмма пакетов приложения для организаторов

Рассмотрим подробнее, что содержат пакеты:

1. features – глобальные компоненты, используемые на всех страницах приложения;
2. shared – объединяет все методы, компоненты и константы, которые используются неоднократно в различных частях приложения;
 - 2.1. api – методы для выполнения запросов к серверу;
 - 2.2. components – компоненты;
 - 2.3. helpers – методы;
 - 2.4. hooks – кастомные хуки;
 - 2.5. img – изображения;
 - 2.6. types – интерфейсы;
3. pages – объединяет все компоненты страниц приложения;
 - 3.1. Constructor – компоненты страницы «Конструктор»;
 - 3.2. Login – компоненты страницы авторизации;
 - 3.3. Games – компоненты страницы «Игры»;
 - 3.4. Demo – компоненты страницы демонстрации квиза.

3.1.1 Авторизация в приложении

При открытии приложения пользователь видит страницу авторизации. Авторизация обязательна, без неё функциональные возможности приложения не доступны. На рисунке 9 представлен вид страницы авторизации.



Рисунок 9 – Интерфейс страницы авторизации

3.1.2 Реализация конструктора квизов

При открытии страницы «Конструктор» пользователь видит список всех созданных квизов. На рисунке 10 представлена диаграмма компонентов страницы.

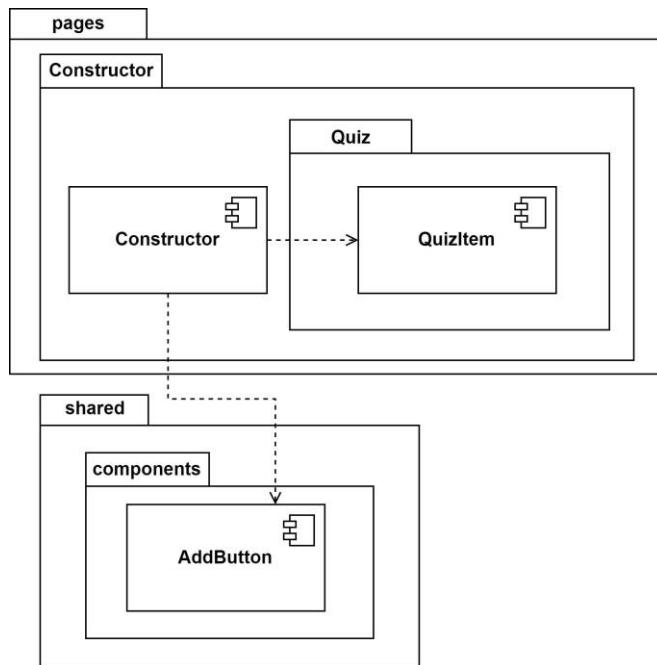


Рисунок 10 – Диаграмма компонентов страницы «Конструктор»

«Constructor» – основная компонента страницы, которая включает в себя компоненту карточки квиза «QuizItem» из папки «Quiz», и компоненту кнопки добавления «AddButton» из папки «shared». Итоговый вид страницы представлен на рисунке 11.

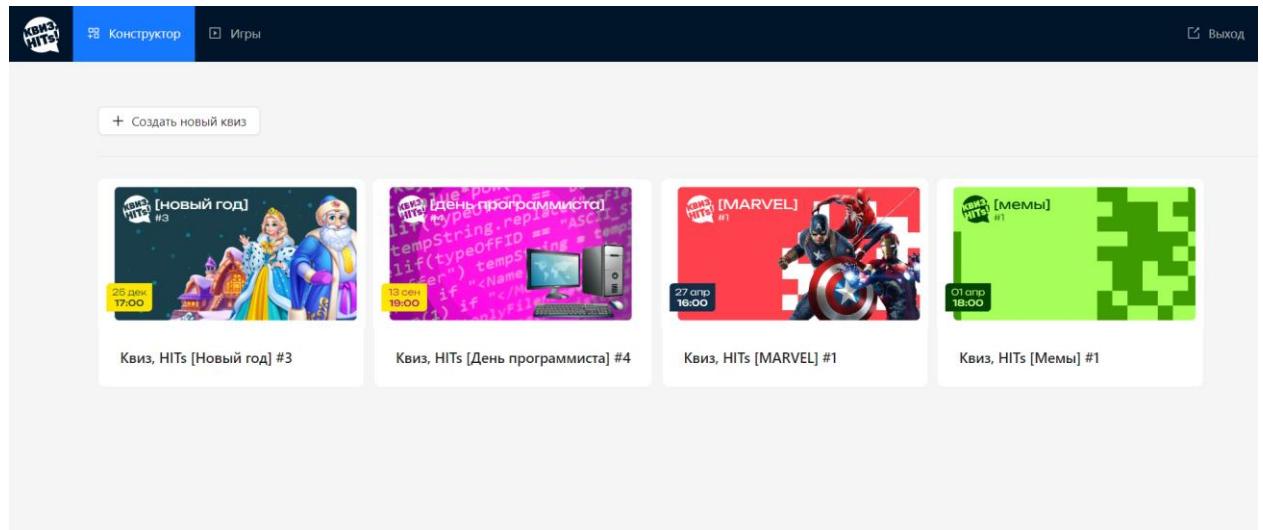


Рисунок 11 – Интерфейс страницы «Конструктор»

При нажатии кнопки «Создать новый квиз» открывается страница создания квиза. При нажатии на любой квиз открывается страница с детальной информацией о данном квизе.

Для создания квиза пользователю необходимо заполнить форму и ввести необходимые данные. Справа от формы выводится предпросмотр обложки квиза, которая меняется автоматически при изменении данных формы. При нажатии кнопки «Создать» данные квиза передаются на сервер, создаётся новый квиз и открывается страница деталей квиза. На рисунке 12 представлена диаграмма компонентов страницы создания квиза.

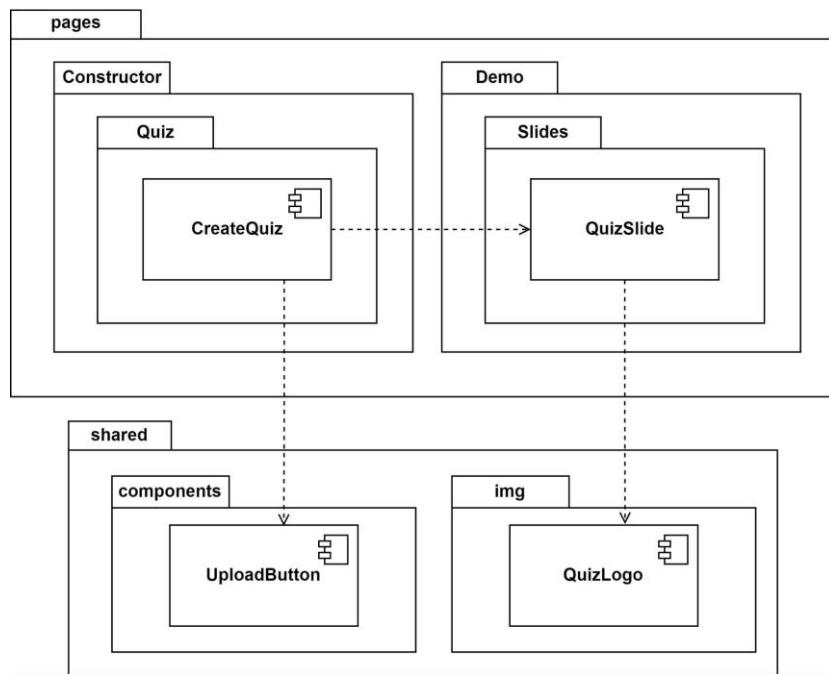


Рисунок 12 – Диаграмма компонентов страницы создания квиза

«CreateQuiz» – основная компонента страницы, которая содержит компоненту кнопки загрузки файла «UploadButton» из папки «shared» и компонент слайда обложки квиза «QuizSlide» из папки «Demo».

Для более понятного представления взаимодействия компонентов на рисунке 13 представлен процесс создания квиза организатором.

Итоговый вид страницы создания квиза представлен на рисунке 14.

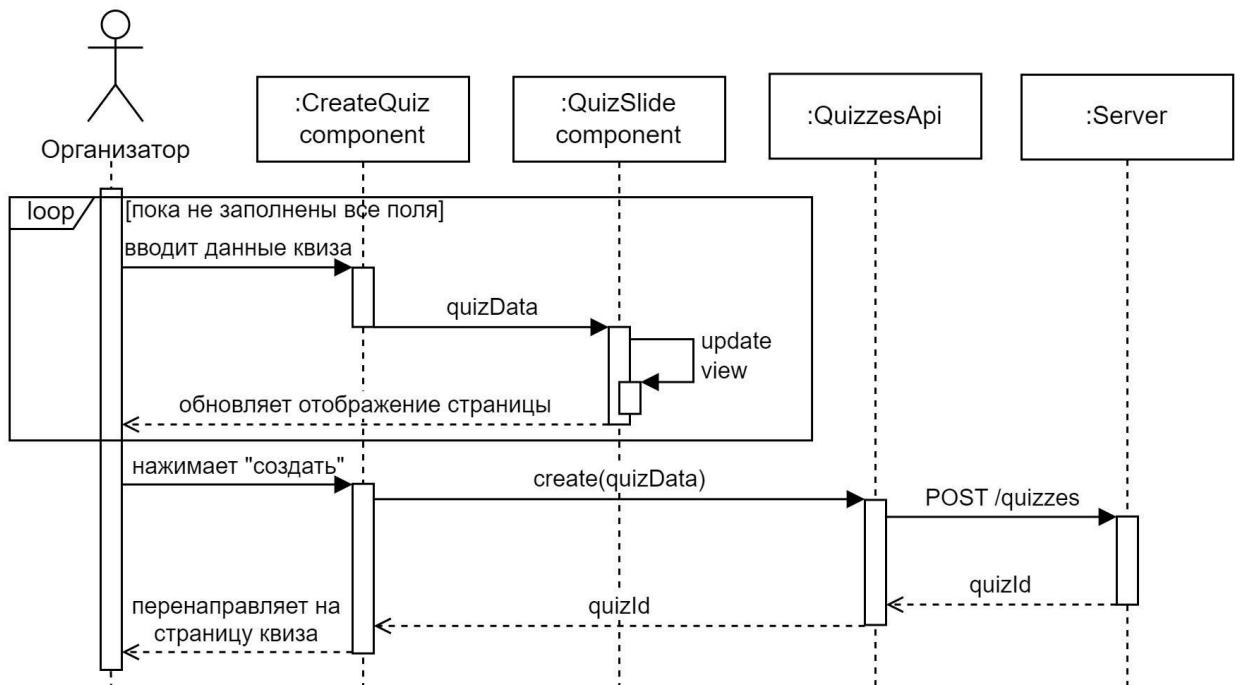


Рисунок 13 – Процесс создания квиза

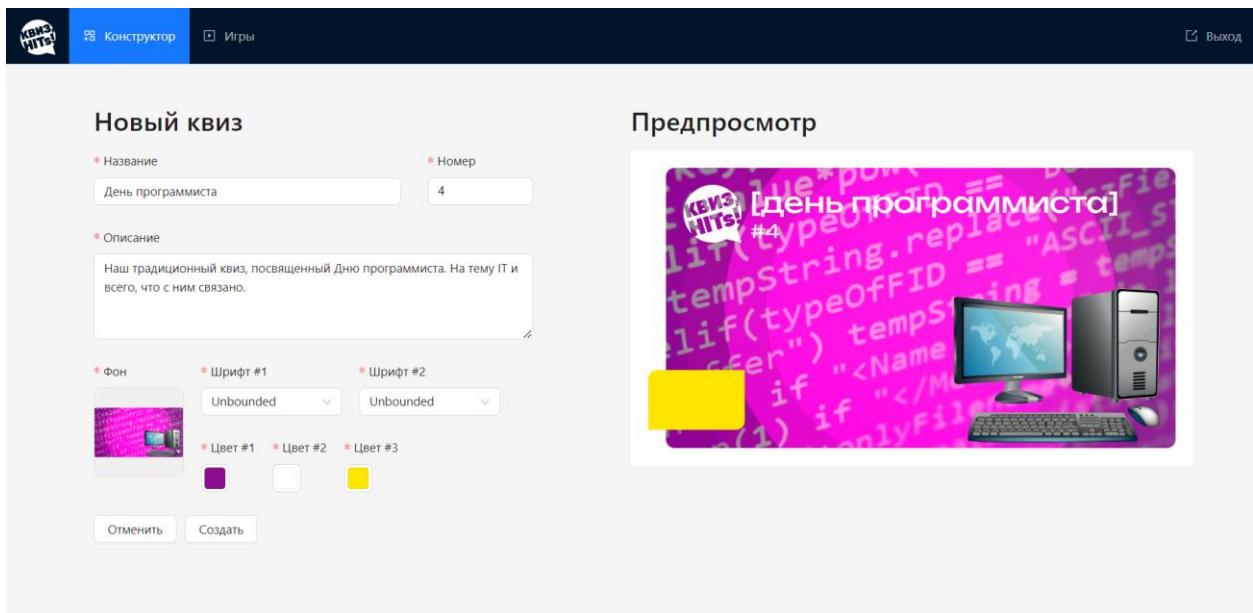


Рисунок 14 – Интерфейс страницы создания квиза

На детальной странице квиза пользователь может редактировать, удалить квиз, добавить раунды, запустить предпросмотр слайдов, а также создать новую игру. При нажатии «Добавить раунд» открывается страница создания раунда. При нажатии на любой раунд открывается страница с его детальной информацией. На рисунке 15 представлена диаграмма компонентов страницы деталей квиза.

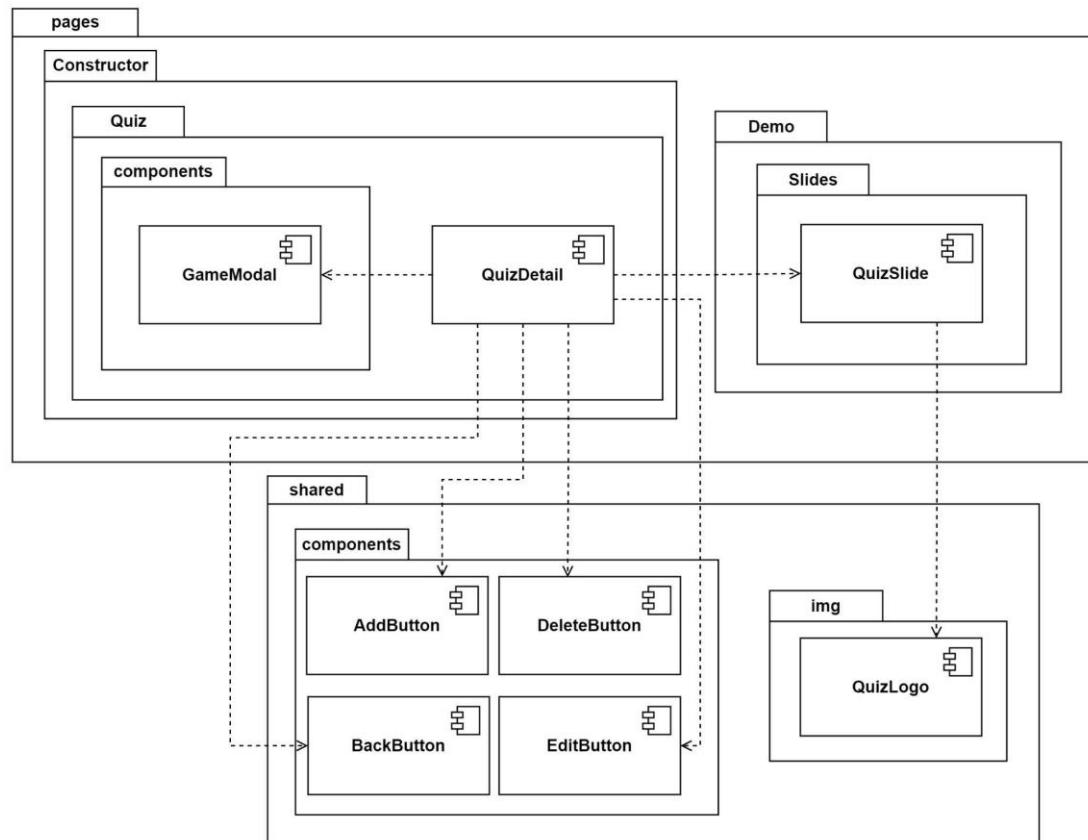


Рисунок 15 – Диаграмма компонентов страницы деталей квиза

«QuizDetail» – основная компонента страницы, которая включает в себя компоненту модального окна «QuizGame», компоненту слайда обложки квиза «QuizSlide» из папки «Demo» и компоненты кнопок из папки «shared»: кнопку добавления «AddButton», кнопку редактирования «EditButton», кнопку удаления «DeleteButton» и кнопку шага назад «BackButton». Итоговый вид страницы представлен на рисунке 16.

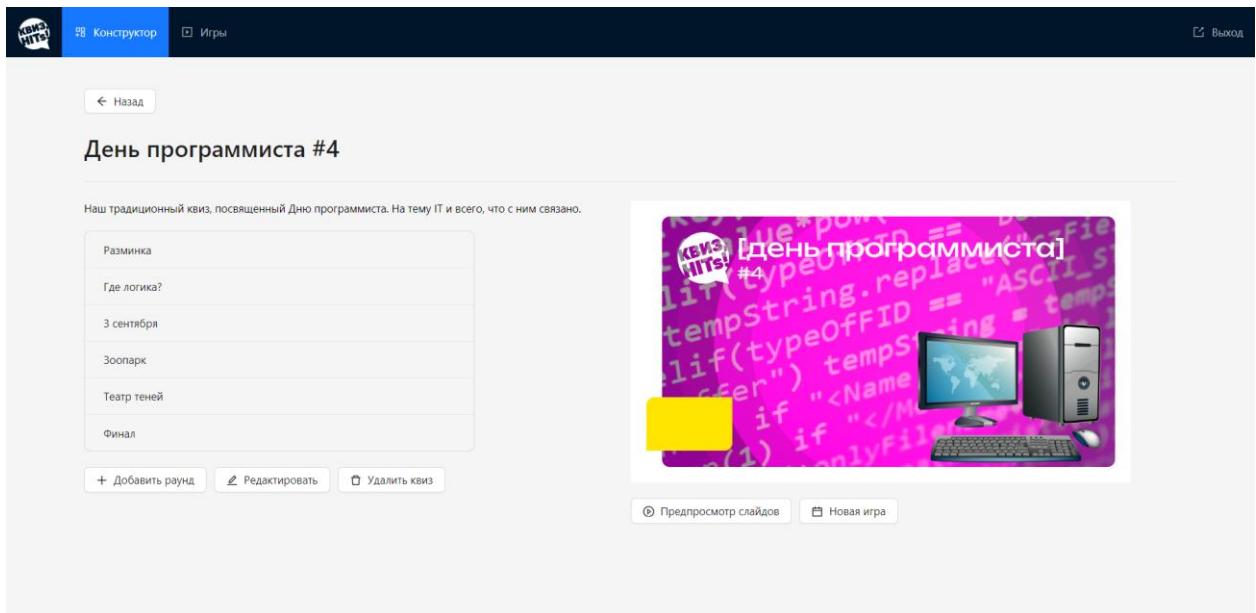


Рисунок 16 – Интерфейс страницы деталей квиза

При нажатии кнопки «Предпросмотр слайдов» открывается демонстрация квиза. При нажатии кнопки «Новая игра» открывается модальное окно создания игры, в котором можно выбрать дату и время игры, которые также появляются на обложке слайда. После сохранения квиз добавляется в расписание. На рисунке 17 представлен интерфейс модального окна создания игры.



Рисунок 17 – Модальное окно создания игры

Для создания раунда необходимо также заполнить форму и ввести необходимые данные. Справа от формы выводится предпросмотр обложки раунда, которая меняется автоматически при изменении введенных данных. При нажатии кнопки «Создать» данные раунда передаются на сервер, добавляется новый раунд и открывается страница деталей раунда. На рисунке 18 представлена диаграмма компонентов страницы создания раунда.

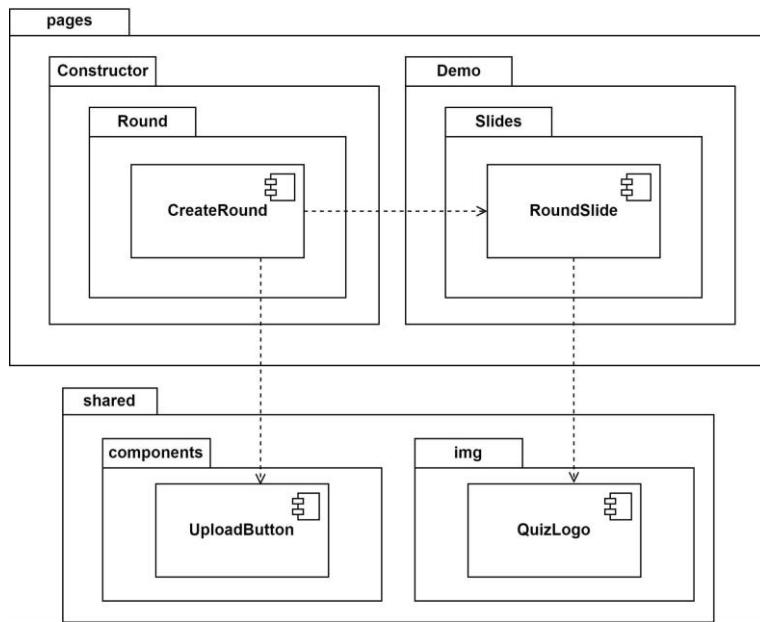


Рисунок 18 – Диаграмма компонентов страницы создания раунда

«CreateRound» – основная компонента страницы, которая включает в себя компоненту слайда обложки раунда «RoundSlide» из папки «Demo» и компоненту кнопки загрузки файла из папки «shared».

Для более понятного представления взаимодействия компонентов на рисунке 19 представлен процесс создания раунда организатором.

Итоговый вид страницы создания раунда представлен на рисунке 20.

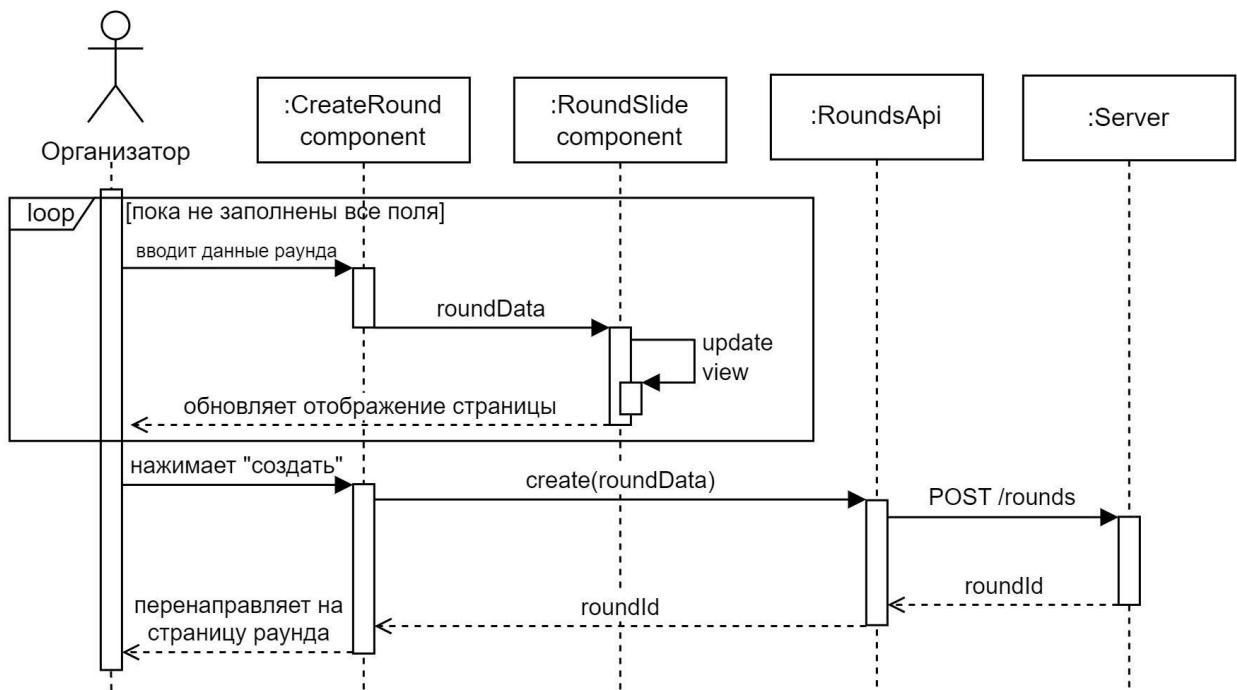


Рисунок 19 – Процесс создания раунда

Рисунок 20 – Интерфейс страницы создания раунда

В форме создания есть выпадающий список – «Тип раунда», который содержит фиксированные значения. При выборе значения «Сопоставление» в форму добавляется ещё одно поле ввода – «Варианты». На рисунке 21 представлен вид выпадающего списка, на рисунке 22 – появляющееся поле «Варианты».

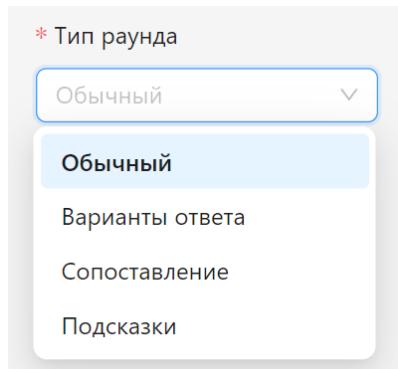


Рисунок 21 – Выпадающий список «Тип раунда»

Рисунок 22 – Поле ввода «Варианты»

На детальной странице раунда пользователь может удалить и редактировать раунд, а также добавить вопросы. При нажатии кнопки «Добавить вопрос» открывается страница создания вопроса, при нажатии на любой вопрос открывается его детальная страница. На рисунке 23 представлена диаграмма компонентов страницы деталей раунда.

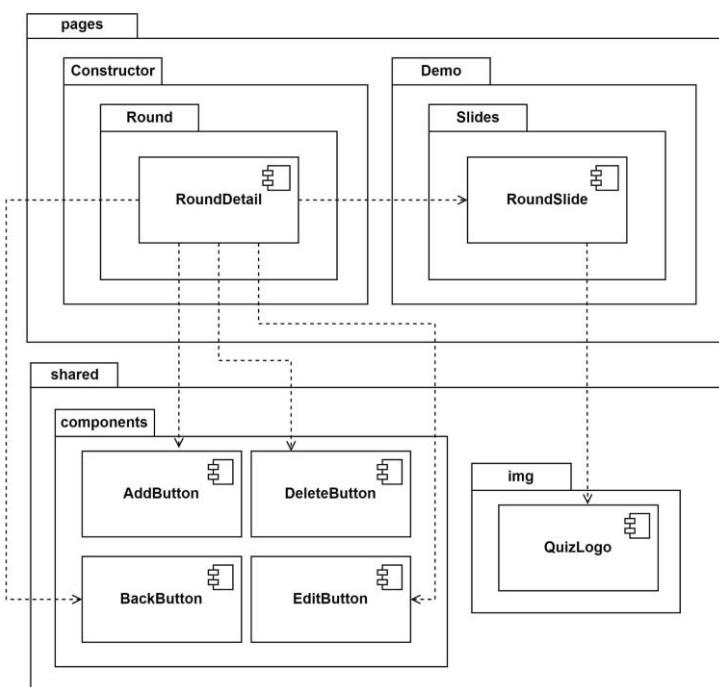


Рисунок 23 – Диаграмма компонентов страницы деталей раунда

«RoundDetail» – основная компонента страницы, которая включает в себя компоненту слайда обложки раунда «RoundSlide» из папки «Demo» и компоненты кнопок

из папки «shared»: кнопку добавления, кнопку редактирования, кнопку удаления и кнопку шага назад. Итоговый вид страницы представлен на рисунке 24.

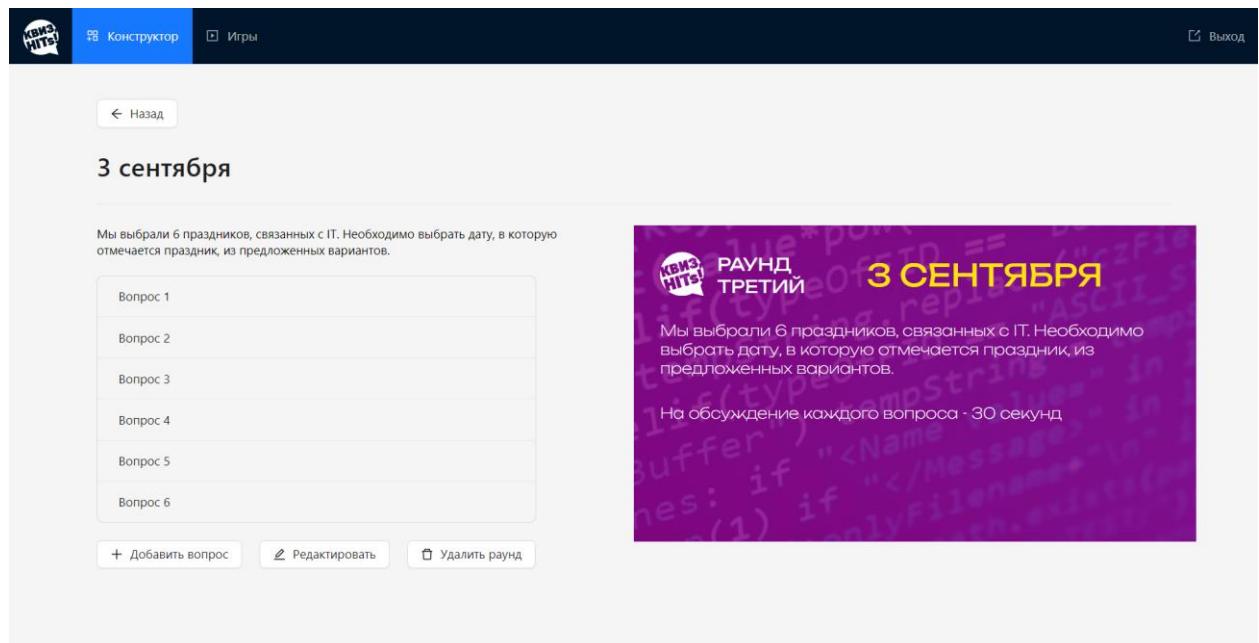


Рисунок 24 – Интерфейс страницы деталей раунда

Для создания вопроса пользователю также необходимо заполнить форму и ввести необходимые данные. Помимо вопроса пользователь вводит данные ответа, поэтому справа от формы выводится предпросмотр слайда с вопросом и предпросмотр слайда с ответом, которые изменяются автоматически при изменении введенных данных. При нажатии кнопки «Создать» данные вопроса передаются на сервер, добавляется новый вопрос и открывается страница деталей вопроса.

Для каждого типа вопросов форма создания разная. Перед отрисовкой страницы создания вопроса система проверяет тип раунда, в котором создаётся вопрос, и в зависимости от него изменяет вид страницы.

На рисунке 25 представлена диаграмма компонентов страницы создания вопроса.

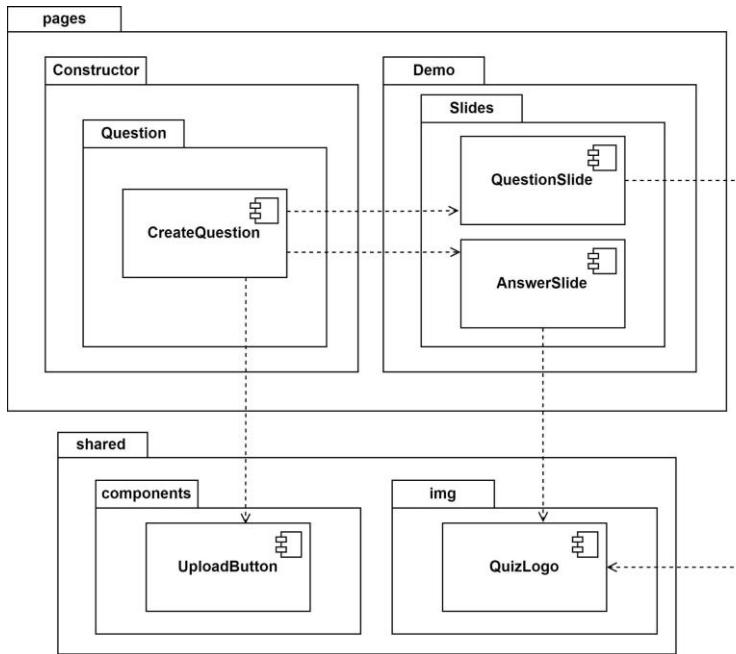


Рисунок 25 – Диаграмма компонентов страницы создания вопроса

«CreateQuestion» – основная компонента страницы, которая включает в себя компоненту кнопки загрузки файла из папки «shared» и компоненты из папки «Demo»: компонент слайда вопроса «QuestionSlide» и компонент слайда ответа «AnswerSlide».

Для более понятного представления взаимодействия компонентов на рисунке 26 представлен сценарий создания вопроса организатором.

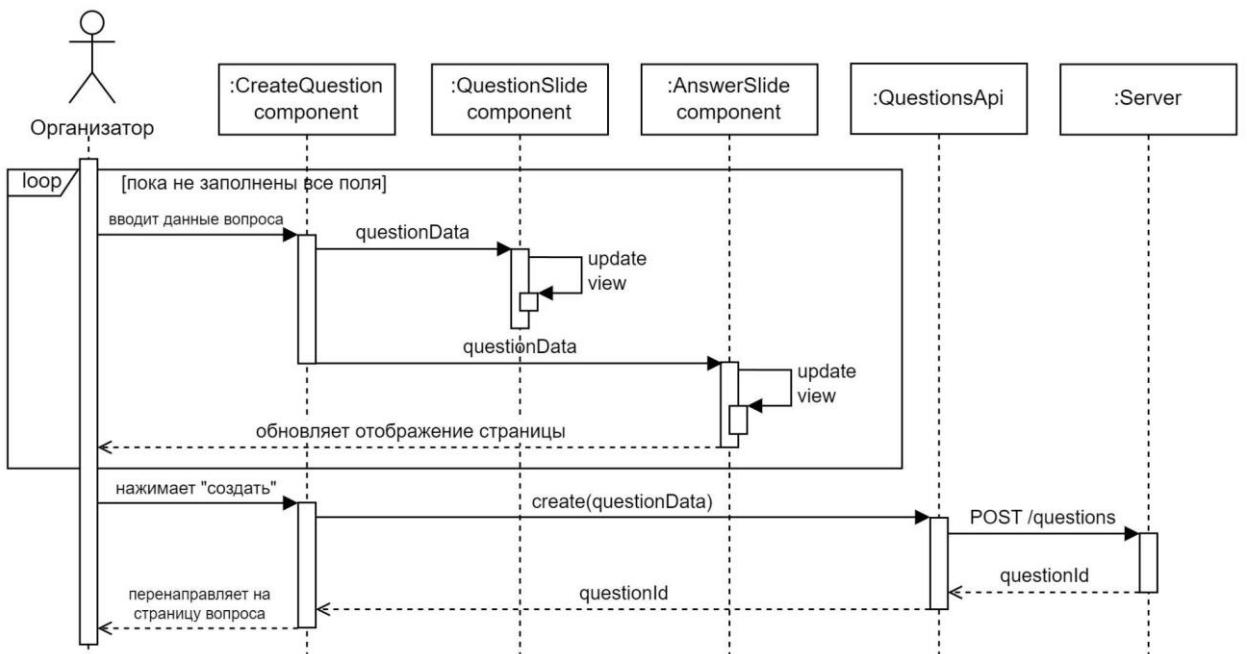


Рисунок 26 – Процесс создания вопроса

Рассмотрим основные типы вопросов. На рисунке 27 представлен итоговый вид страницы создания вопроса для типа «Варианты ответа».

Рисунок 27 – Интерфейс страницы создания вопроса для типа «Варианты ответа»

При типе «Варианты ответа» пользователь может ввести текст вопроса и/или картинку, и сами варианты ответа. Затем выбрать верный ответ из этих четырёх и по желанию пояснение и/или картинку ответа. Правильный ответ выделяется рамкой. Если введено пояснение к ответу, то на слайде ответа отображается оно, иначе – текст вопроса. Если загружена картинка ответа, то на слайде ответа отображается она, иначе – картинка вопроса.

На рисунке 28 представлен итоговый вид страницы создания вопроса для типа «Сопоставление». При этом типе на слайдах отображаются варианты, введенные при создании раунда. Можно также добавить текст и картинку вопроса, ввести ответ, пояснение и картинку ответа. Правильный ответ выделяется рамкой.

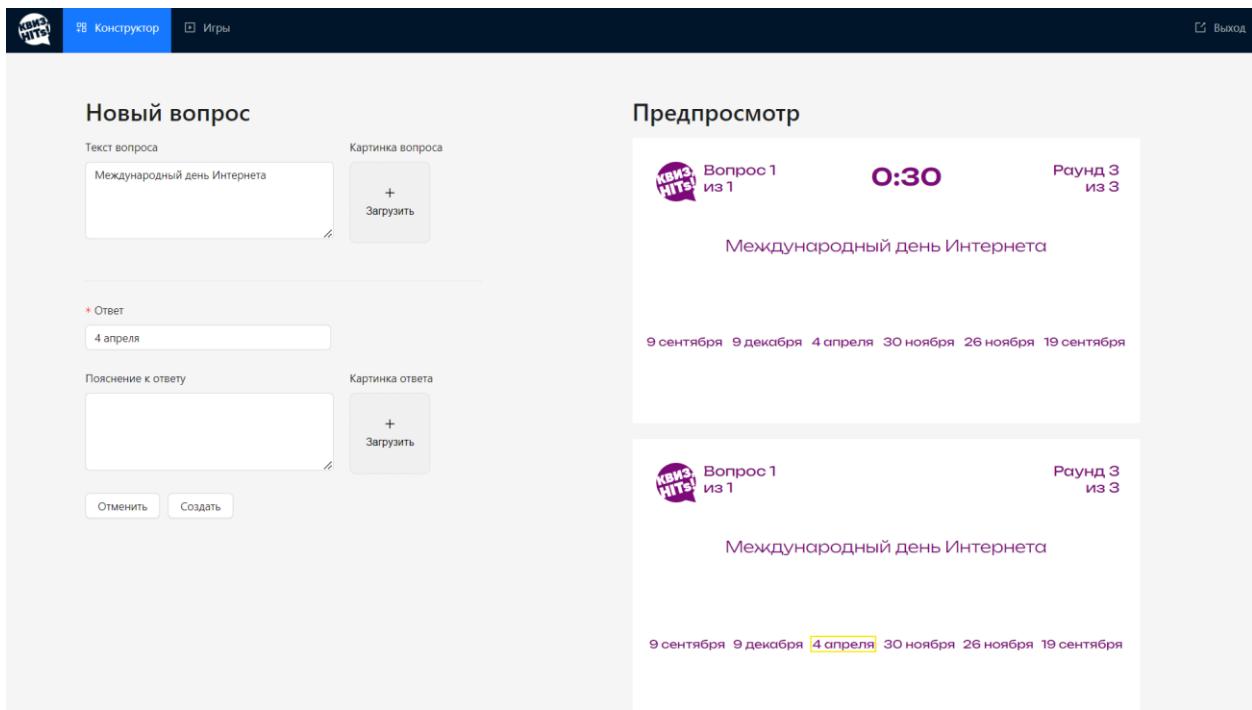


Рисунок 28 – Интерфейс страницы создания вопроса для типа «Сопоставление»

При выборе типа «Подсказки» на странице отображается три слайда – по одному для каждой из подсказок. Пользователь может ввести текст и/или картинку для каждой из трех подсказок и правильный ответ. На слайде ответа отображается ответ и пояснение/картинка, если они были добавлены. Итоговый вид страницы создания вопроса для этого типа представлен на рисунке 29.

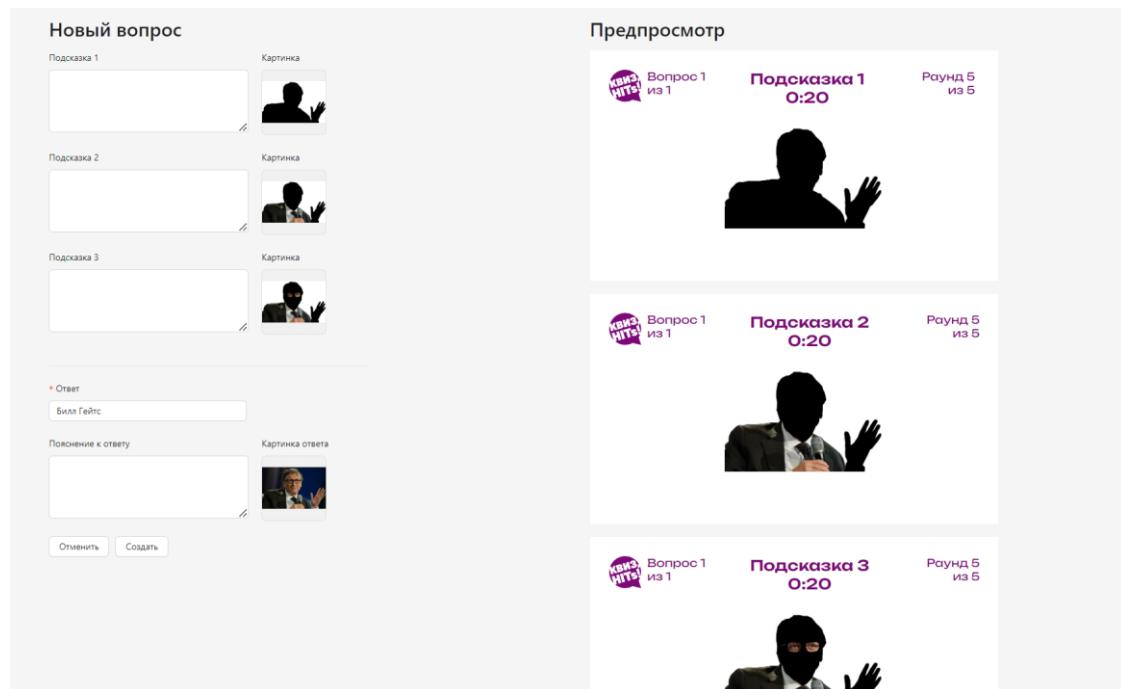


Рисунок 29 – Интерфейс страницы создания вопроса для типа «Подсказки»

На детальной странице вопроса пользователь может удалить и редактировать вопрос. На ней также отображается слайд и вопроса, и ответа. Диаграмма компонентов страницы деталей вопроса представлена на рисунке 30.

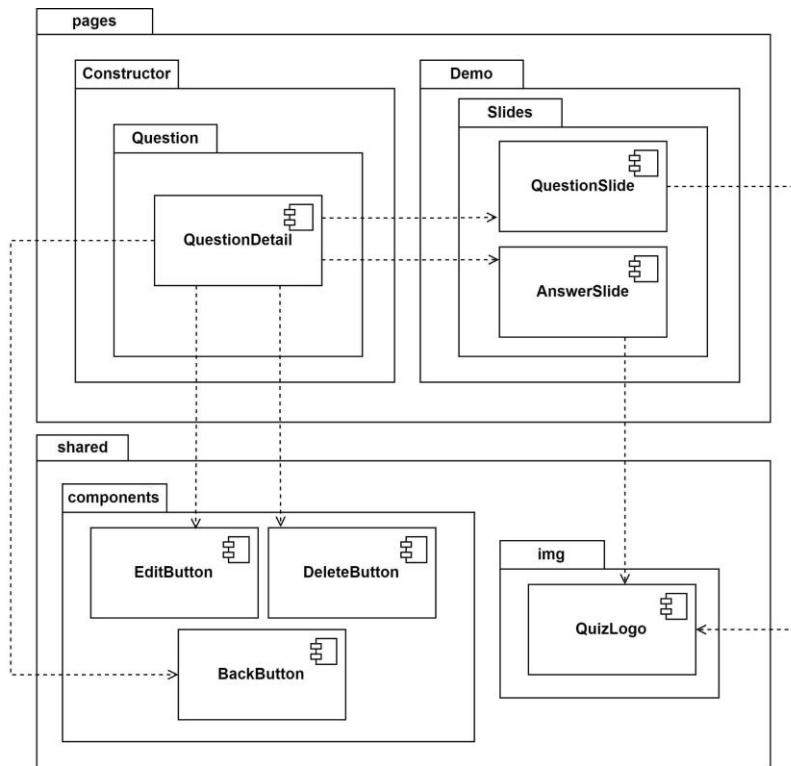


Рисунок 30 – Диаграмма компонентов страницы деталей вопроса

«QuestionDetail» – основная компонента страницы, которая включает в себя компоненты из папки «Demo»: компонент слайда вопроса «QuestionSlide» и компонент слайда ответа «AnswerSlide», а также компоненты кнопок из папки «shared»: кнопку редактирования, кнопку удаления и кнопку шага назад. На рисунке 31 представлен итоговый вид страницы.

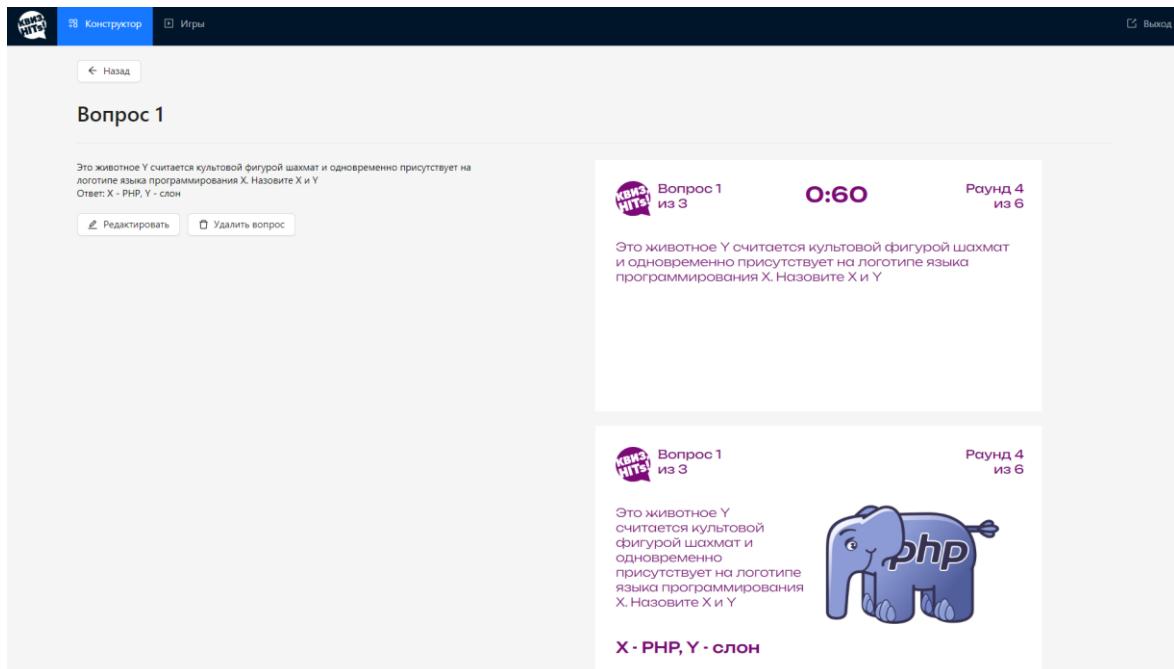


Рисунок 31 – Интерфейс страницы деталей вопроса

3.1.3 Реализация проведения квиза

При открытии страницы «Игры» пользователь видит список всех прошедших и предстоящих игр, добавленных в расписание. При нажатии на карточку игры открывается страница с подробной информацией. Итоговый вид страницы «Игры» представлен на рисунке 32.

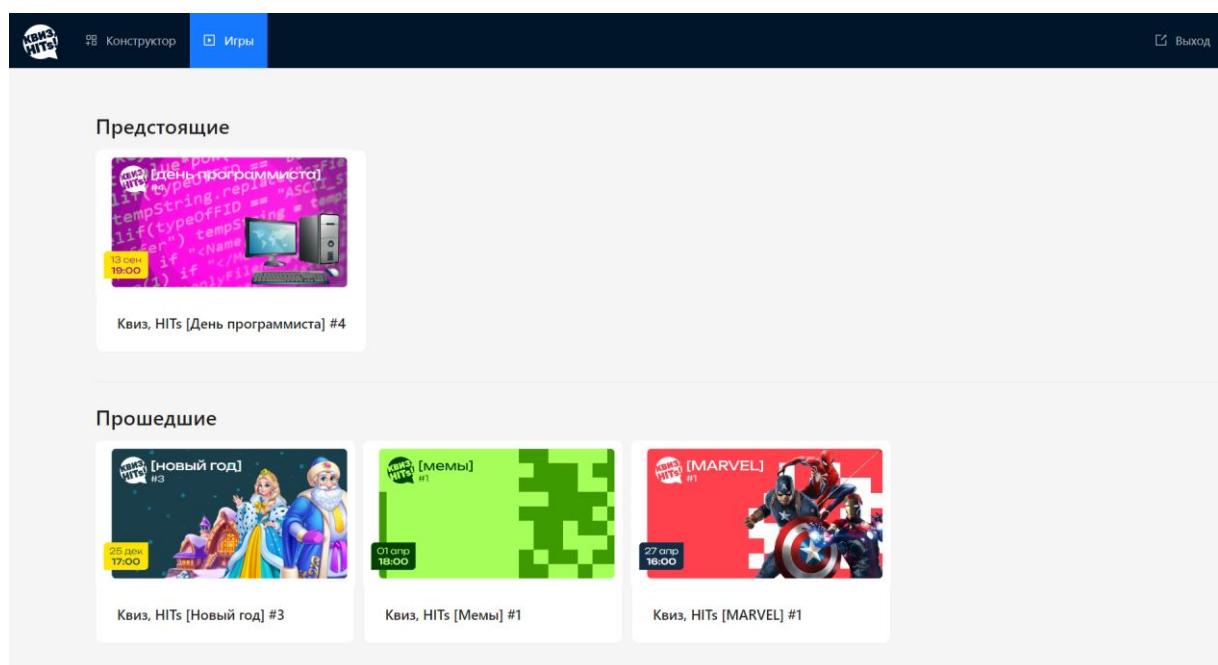


Рисунок 32 – Интерфейс страницы «Игры»

На странице деталей игры пользователь может начать игру, удалить её из расписания, а также посмотреть список зарегистрированных команд. Список команд можно редактировать. На рисунке 33 представлена диаграмма компонентов страницы игры, итоговый вид страницы представлен на рисунке 34.

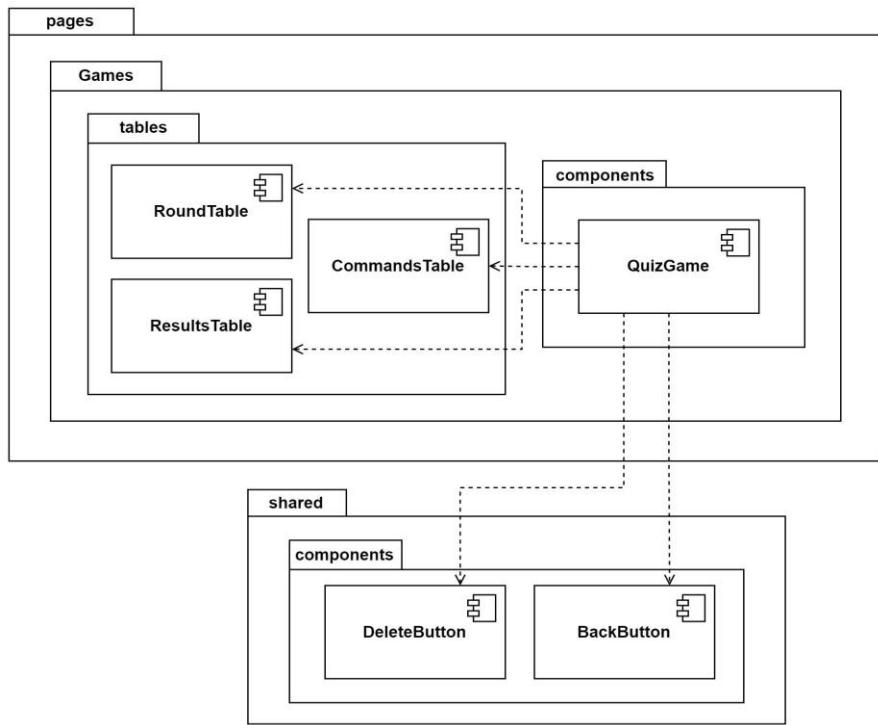


Рисунок 33 – Диаграмма компонентов страницы деталей игры

«QuizGame» – основная компонента страницы, которая включает в себя компоненты из папки «tables»: компонент таблицы списка команд «CommandsTable», компонент таблицы результатов раунда «RoundTable», компонент таблицы итоговых результатов «ResultsTable», а также компоненты кнопок из папки «shared»: кнопку удаления и кнопку шага назад.

Название	Участники	Ссылка на капитана	
Мем-школа	Назарова Мария Нетаев Владислав Потапов Степан Неверовский Максим Ситникова Елена Ткачев Максим Веретнов Тимофей	vk.com/macoolri	Сообщить Удалить
Чемпионы	Тюделекова Наталья Смирнов Виталий Павлов Леонид Миллер Игорь	vk.com/qwerty	Сообщить Удалить
Знатоки	Иванов Иван Петров Пётр Семёнов Семён	vk.com/ivanov	Сообщить Удалить

Рисунок 34 – Интерфейс страницы деталей игры

При нажатии кнопки «Начать игру» открывается страница демонстрации квиза. На рисунке 35 представлена диаграмма компонентов страницы.

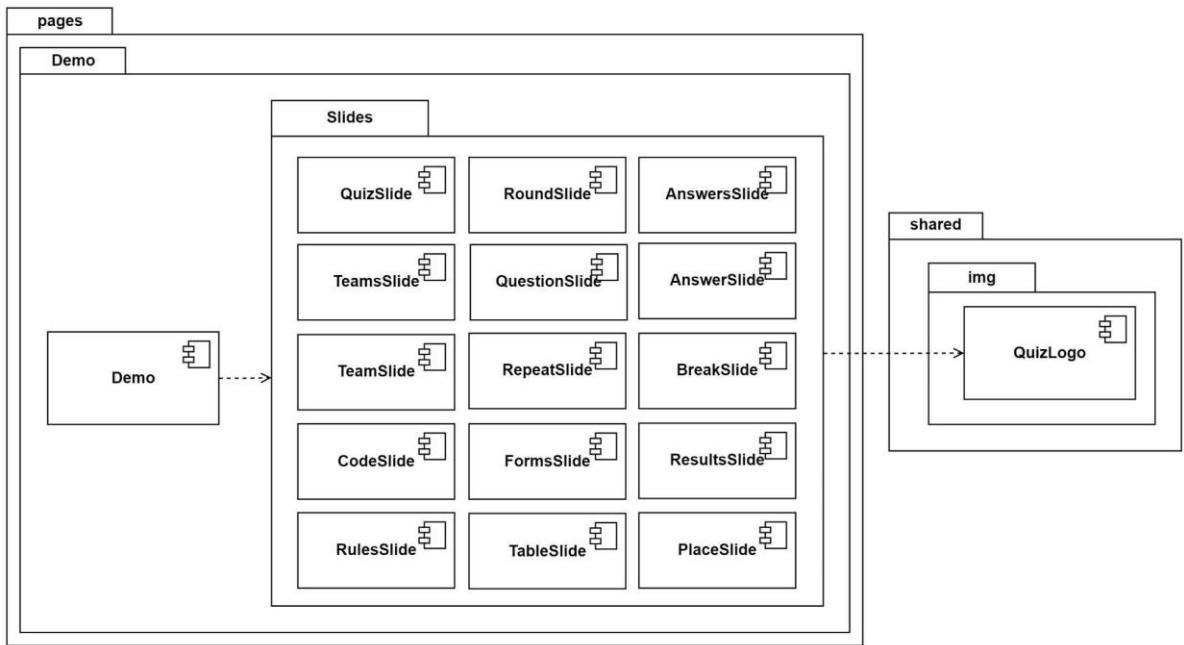


Рисунок 35 – Диаграмма компонентов страницы демонстрации квиза

«Demo» – основной компонент страницы, который использует компоненты слайдов из папки «Slides». Также слайды используют компонент «QuizLogo» из папки «shared».

Описание компонентов слайдов:

1. QuizSlide – слайд обложки квиза;
2. CodeSlide – слайд с кодом игры;
3. TeamsSlide – слайд «Участники»;
4. TeamSlide – слайд с названием команды;
5. RulesSlide – слайд с правилами квиза;
6. RoundSlide – слайд обложки раунда;
7. QuestionSlide – слайд вопроса;
8. RepeatSlide – слайд «Повторим вопросы»;
9. FormsSlide – слайд «Отправляем ответы»;
10. AnswersSlide – слайд «Проверим ответы»;
11. AnswerSlide – слайд ответа;
12. BreakSlide – слайд «Перерыв»;
13. ResultsSlide – слайд «Результаты»;
14. PlaceSlide – слайд с командой-призером;
15. TableSlide – слайд с таблицей результатов.

Страница демонстрации открывается на весь экран. Первым слайдом показывается обложка квиза. Затем, когда пользователь нажимает на экран, показывается следующий слайд. Поочередно показываются слайды раундов, вопросов, ответов. В компонент слайда передаются данные квиза, его раундов и вопросов, и отрисовываются в соответствии с этими данными. Также есть промежуточные слайды, такие как «Участники», «Перерыв», «Результаты». На рисунке 36 представлен порядок показа слайдов в виде диаграммы состояний.

При нажатии на клавиатуре клавиши «Escape» происходит выход из полноэкранного режима и открывается страница игры.

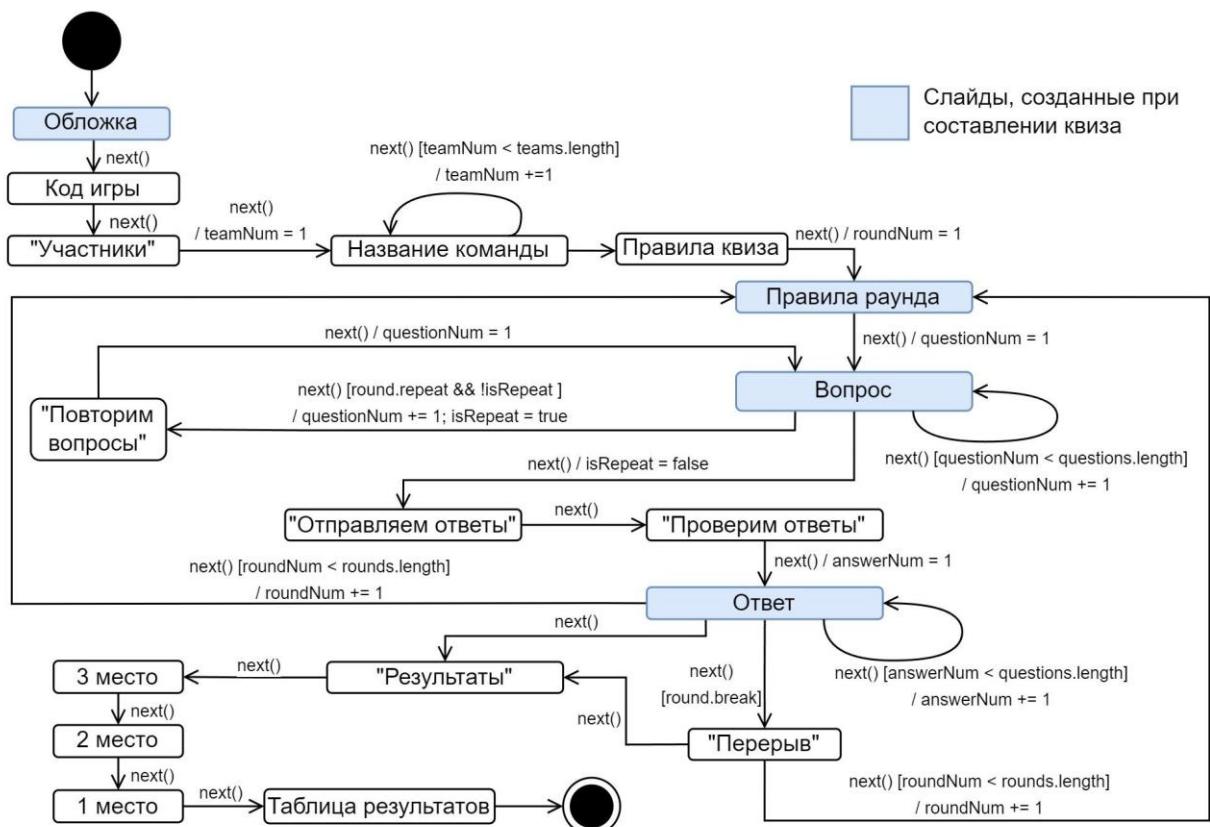


Рисунок 36 – Порядок показа слайдов

После начала игры пользователю доступен просмотр результатов квиза. После окончания каждого раунда с сервера приходят результаты, таблица обновляется и в ней появляются ответы каждой команды с отметкой верно/неверно. При ошибке автоматической проверки Организатор может дополнительно изменять таблицу результатов, отмечая ответы самостоятельно. Также ему доступна таблица итоговых результатов игры, в которой команды отсортированы по количеству набранных баллов. После окончания игры пользователь может опубликовать результаты, чтобы участники могли их посмотреть. Итоговый вид таблицы результатов раунда представлен на рисунке 37, таблица итогов игры представлена на рисунке 38.

← Назад

День программиста #4



13 сен
19:00

Игра запущена Удалить из расписания

Результаты

Общие **Раунд 1** Раунд 2 Раунд 3 Раунд 4 Раунд 5 Раунд 6

	1	2	3	4	5	6
	B	A	B	B	Г	A
Мем школа	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Чемпионы	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Знатоки	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Рисунок 37 – Таблица результатов раунда

← Назад

День программиста #4



13 сен
19:00

Игра закончена Удалить из расписания

Результаты

Общие **Раунд 1** Раунд 2 Раунд 3 Раунд 4 Раунд 5 Раунд 6

Место	Команда	Разминка	Где логика?	3 сентября	Зоопарк	Театр теней	Финал	Итог
1	Чемпионы	5	8	12	8	6	17	56
2	Мем школа	4	10	11	6	6	14	51
3	Знатоки	3	5	9	8	8	12	45

Опубликовать результаты

Рисунок 38 – Таблица итоговых результатов

3.2 Реализация клиентского приложения для участников

Верхнеуровневая структура приложения для участников, содержащая основные пакеты и связи между ними, приведена на рисунке 39 в виде диаграммы пакетов.

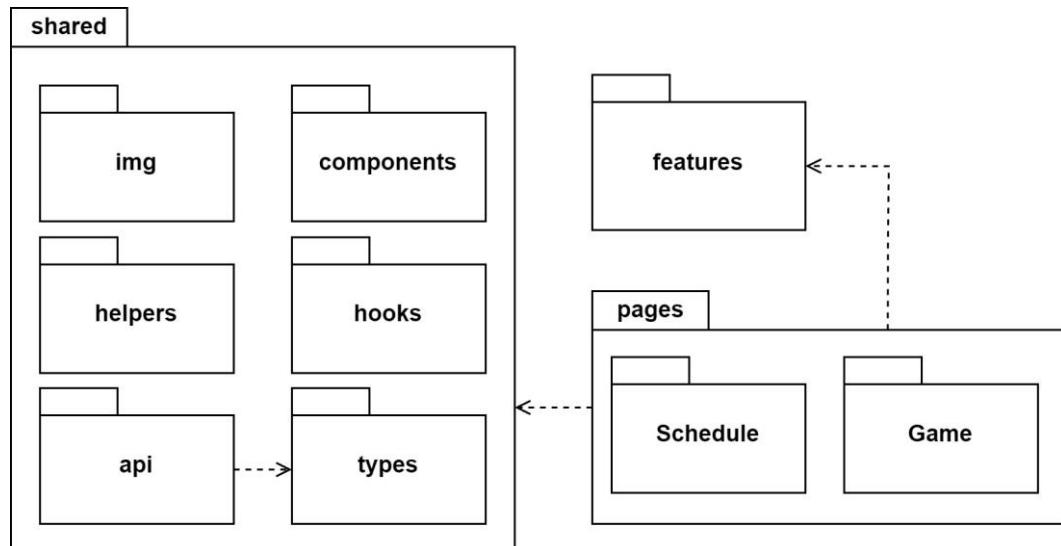


Рисунок 39 – Диаграмма пакетов приложения для участников

Структура проекта аналогична приложению для организаторов, рассмотренного на Рисунке 8. Отличия состоят в пакете pages:

1. Schedule – объединяет компоненты страницы «Расписание»;
2. Game – объединяет компоненты страницы «Играть».

3.2.1 Реализация расписания квизов

При открытии страницы «Расписание» пользователь видит список всех прошедших и предстоящих игр. На рисунке 40 представлена диаграмма компонентов страницы.

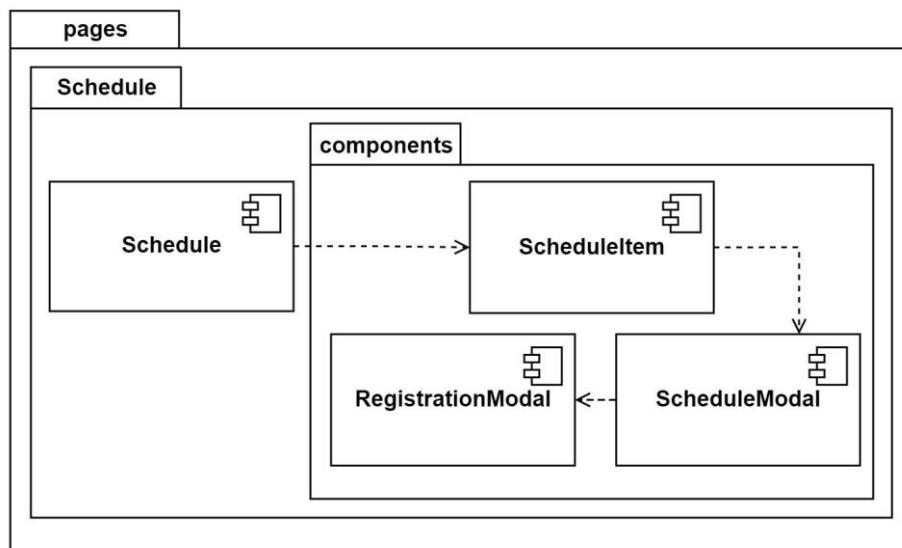


Рисунок 40 – Диаграмма компонентов страницы «Расписание»

«Schedule» – основная компонента страницы, которая включает в себя компоненту карточки квиза «ScheduleItem». Карточка содержит компоненту модального окна с информацией о квизе «ScheduleModal», содержащую в свою очередь компоненту модального окна регистрации «RegistrationModal».

Итоговый вид страницы расписания представлен на рисунке 41. Все игры отсортированы по дате проведения и разделены на прошедшие и предстоящие. При нажатии на карточку квиза открывается модальное окно, внутри которого пользователь может посмотреть информацию о предстоящей игре и зарегистрироваться на неё. При нажатии на кнопку открывается модальное окно регистрации, в котором нужно ввести необходимые данные. Вид модального окна игры представлен на рисунке 42, вид модального окна регистрации – на рисунке 43.

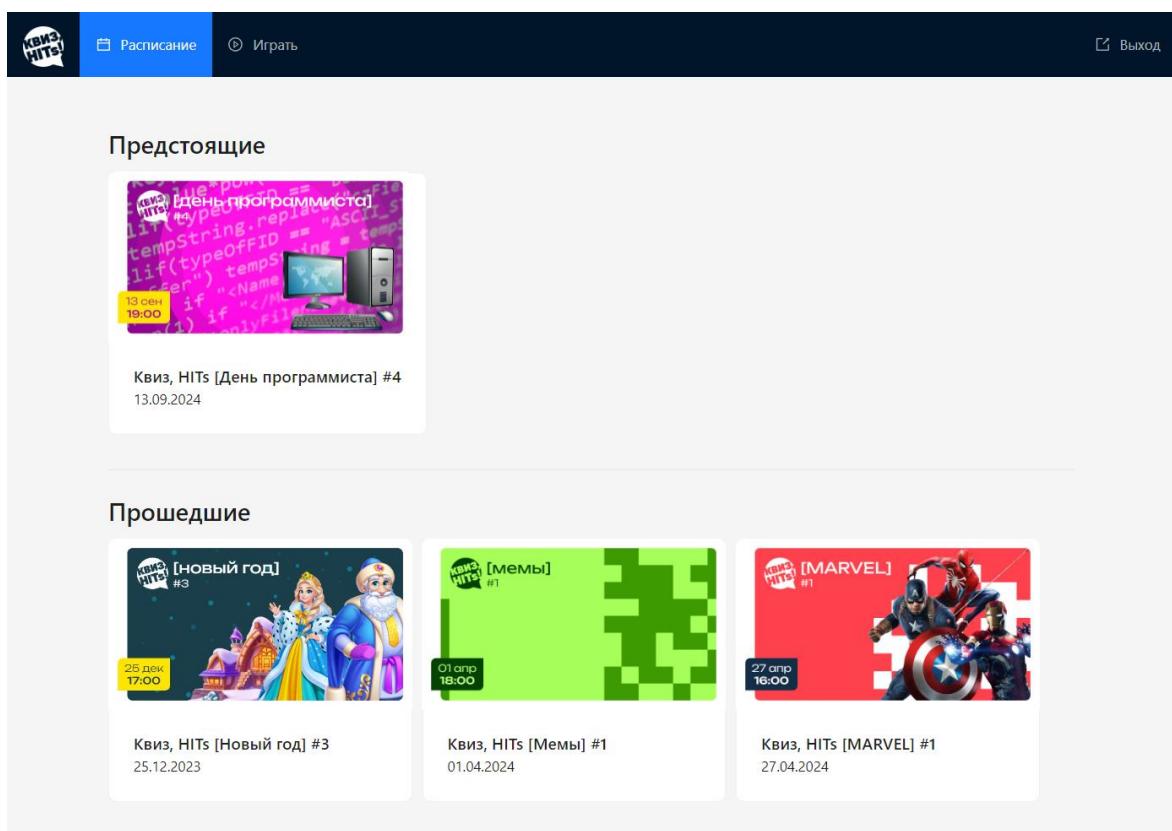


Рисунок 41 – Интерфейс страницы «Расписание»

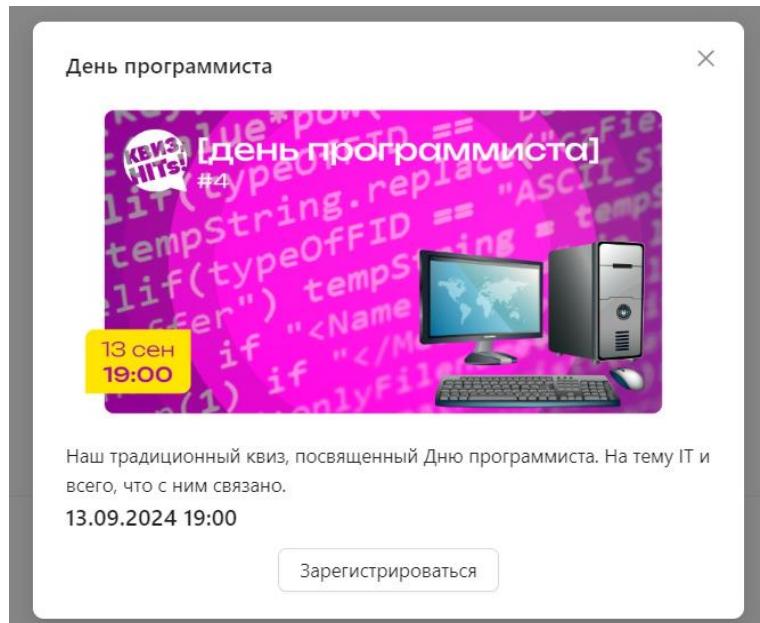


Рисунок 42 – Модальное окно предстоящей игры

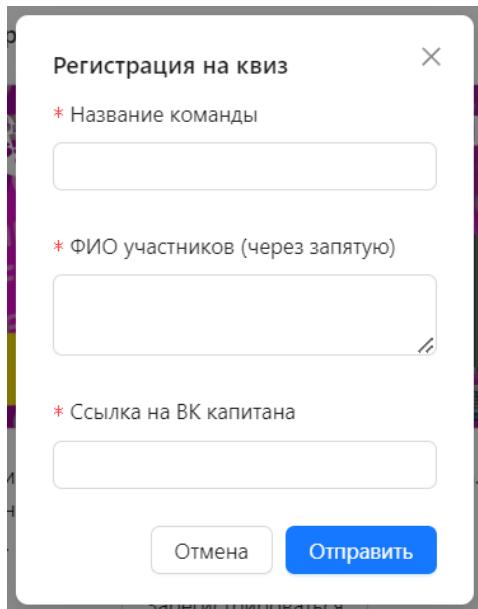


Рисунок 43 – Модальное окно регистрации на квиз

Если игра уже прошла, пользователь может посмотреть результаты. На рисунке 44 представлен вид модального окна прошедшей игры. При нажатии на кнопку «Посмотреть результаты» открывается страница игры.

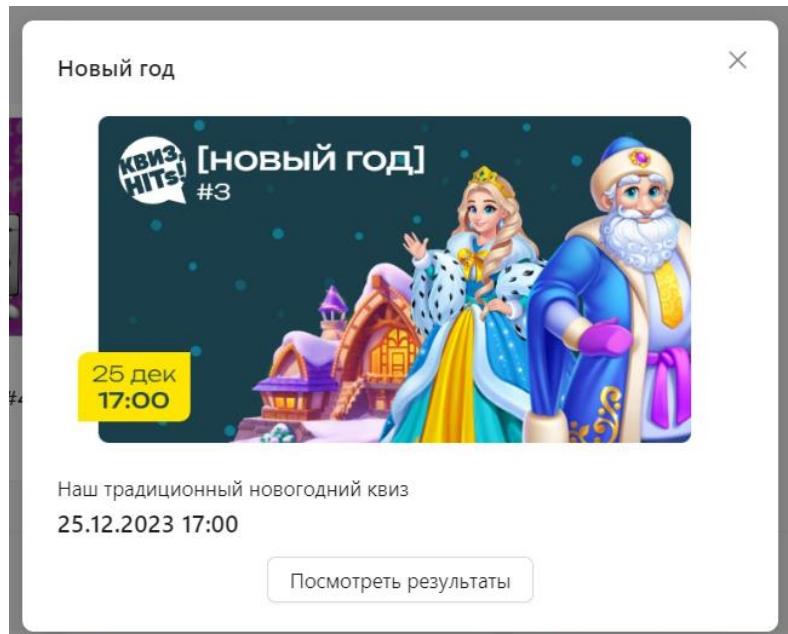


Рисунок 44 – Модальное окно прошедшей игры

На странице прошедшей игры пользователь может посмотреть общие результаты и результаты каждого раунда, если они были опубликованы организатором. На рисунке 45 представлен итоговый вид страницы с результатами игры.

Место	Команда	Разминка	Где логика?	Чей год?	Подарочки	Традиции	Финал	Итог
1	Мем-школа	5	8	12	8	6	17	56
2	Снегурочки	4	10	11	6	6	14	51
3	Снежинки	3	5	9	8	8	12	45

Рисунок 45 – Интерфейс страницы результатов игры

3.2.2 Реализация участия в игре

При переходе на страницу «Играть» пользователь может присоединиться к текущей игре. Участнику необходимо ввести название команды и код, после этого открывается страница игры. На рисунке 46 представлен итоговый вид страницы.

Присоединиться к игре

* Название команды
Команда

* Код игры
1 2 3 4

Начать

Рисунок 46 – Вид страницы присоединения к игре

При успешном присоединении открывается страница игры. На рисунке 47 представлена диаграмма компонентов страницы.

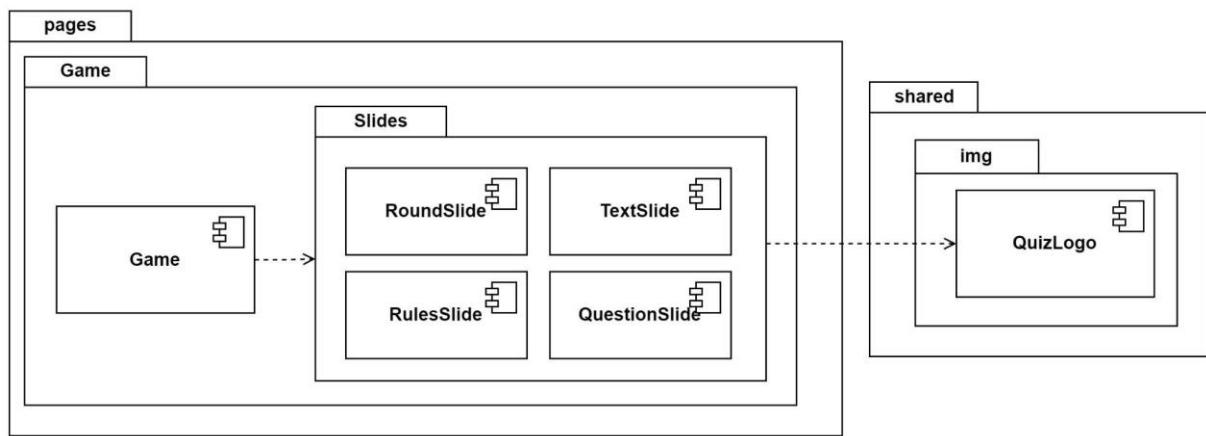


Рисунок 47 – Диаграмма компонентов страницы игры

«Game» – основной компонент страницы, который использует компоненты слайдов из папки «Slides». Также слайды используют компонент «QuizLogo» из папки «shared».

Описание компонентов слайдов:

1. TextSlide – слайд, дублирующий текст на слайде квиза;
2. RulesSlide – слайд с правилами квиза;
3. RoundSlide – слайд с правилами раунда;
4. QuestionSlide – слайд с формой для ввода ответов.

При открытии страницы игры происходит подключение к веб-сокетам, и при переключении организатором слайда, у участника также автоматически переключается слайд. Для более понятного взаимодействия компонентов на рисунке 48 представлена диаграмма проведения одного раунда.

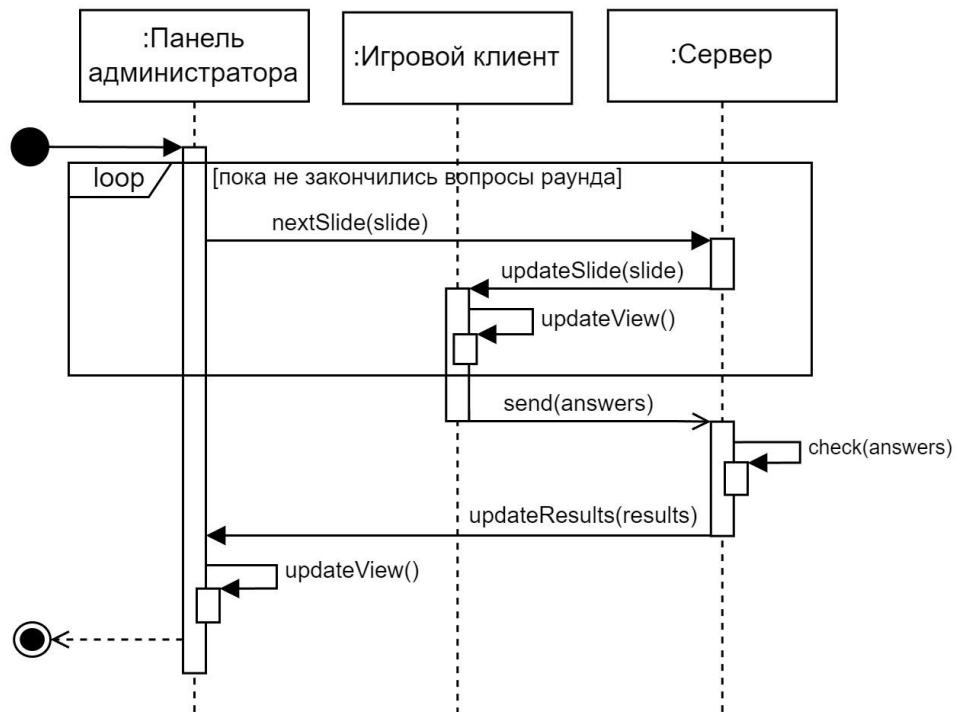


Рисунок 48 – Проведение раунда

Когда показывается любой из промежуточных слайдов, у участника он просто дублируется. Во время показа раунда ему доступна форма для ввода ответов. При типе раунда «Варианты ответа» ему доступен выбор из четырех вариантов для каждого вопроса, при остальных типах ему доступно поле для ввода текстового ответа. При раунде со ставками ему также доступна кнопка «+1» для ставки. На рисунке 49 представлен интерфейс страницы демонстрации игры у организатора (слева) и участника (справа) при раунде с вариантами ответа. На рисунке 50 – интерфейс при раунде со ставками.

Квиз HITs! Вопрос 1 из 4

О:19

Раунд 1 из 6

Это изображение — фрагмент кода, выводящий фразу «Hello, World!». На каком языке программирования это было написано?

A. VisSim
B. Piet

Б. Scratch
Г. HiAsm

[день программиста] #4
Разминка

Вопрос 1
А Б В Г

Вопрос 2
А Б В Г

Вопрос 3
А Б В Г

Вопрос 4
А Б В Г

Отправить

Рисунок 49 – Демонстрация игры при раунде с типом «Варианты ответа»

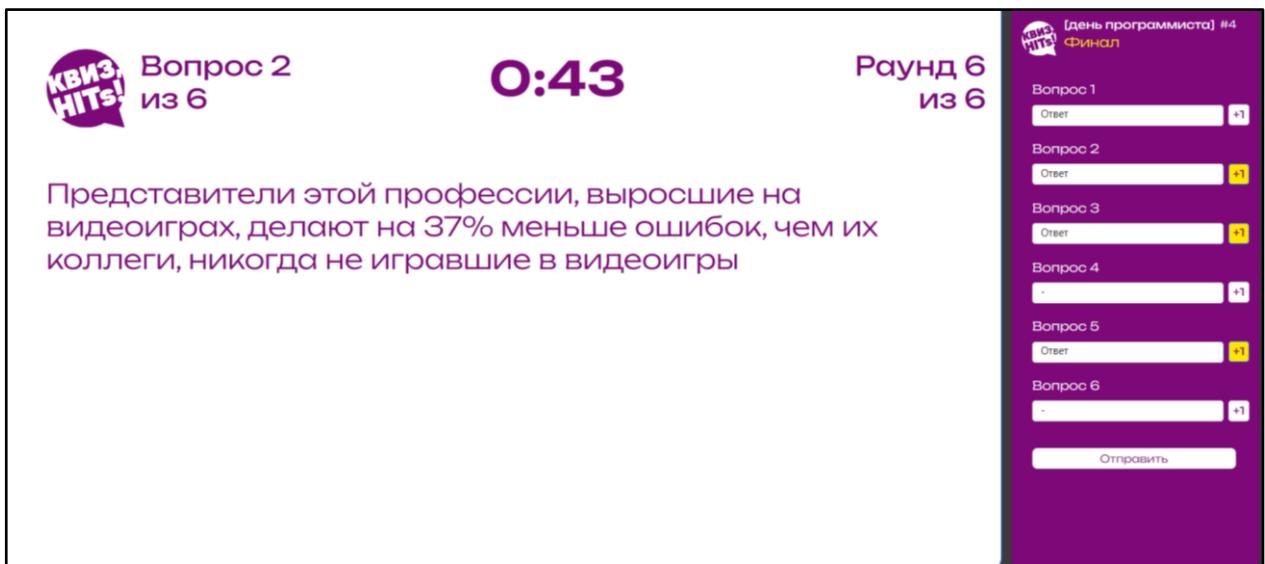


Рисунок 50 – Демонстрация игры при раунде со ставками

3.3 Реализация серверной части платформы

Верхнеуровневая структура серверной части, содержащая основные пакеты и связи между ними, приведена на рисунке в виде диаграммы пакетов.

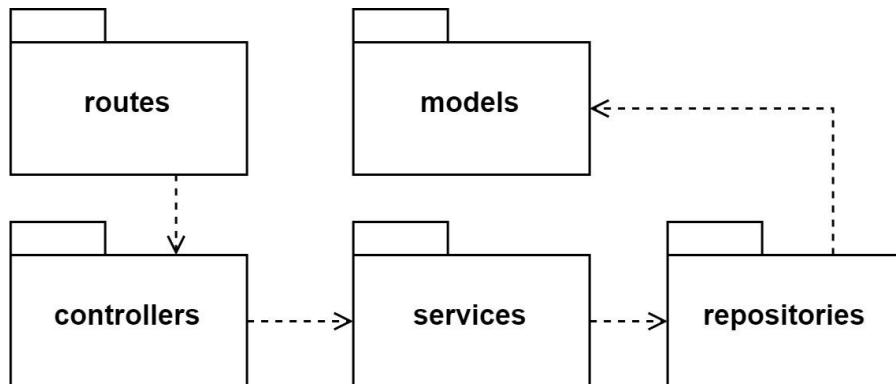


Рисунок 47 – Диаграмма пакетов серверной части платформы

Рассмотрим подробнее, что содержат пакеты:

1. routes – содержит определения маршрутов для обработки запросов;
2. controllers – содержит логику обработки маршрутов;
3. services – содержит функции и методы для выполнения бизнес-логики приложения;
4. repositories – содержит методы для работы с базой данных;
5. models – содержит определения моделей данных.

3.3.1 Реализация запросов

На рисунке 48 представлен типовой набор классов для реализации одного запроса.

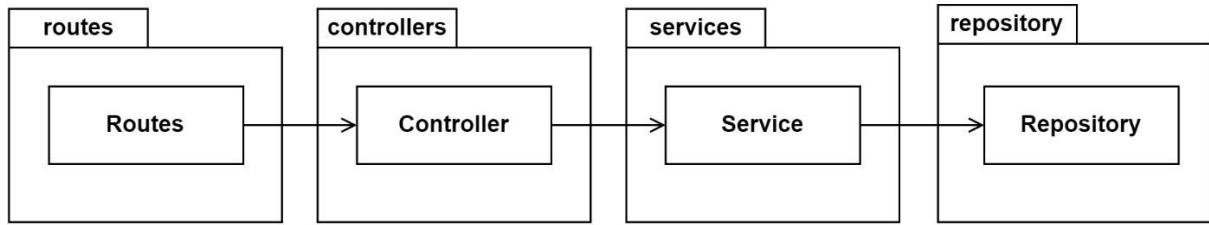


Рисунок 48 – Типовой набор классов для реализации запроса

3.3.2 Реализация проверки ответов

На рисунке 49 представлен процесс автоматического подсчета сервером результатов раунда одной команды. Сервер сравнивает каждый ответ с правильным, прибавляет или отнимает баллы, учитывая вопросы со ставками, считает результаты раунда и отправляет их в панель администратора.

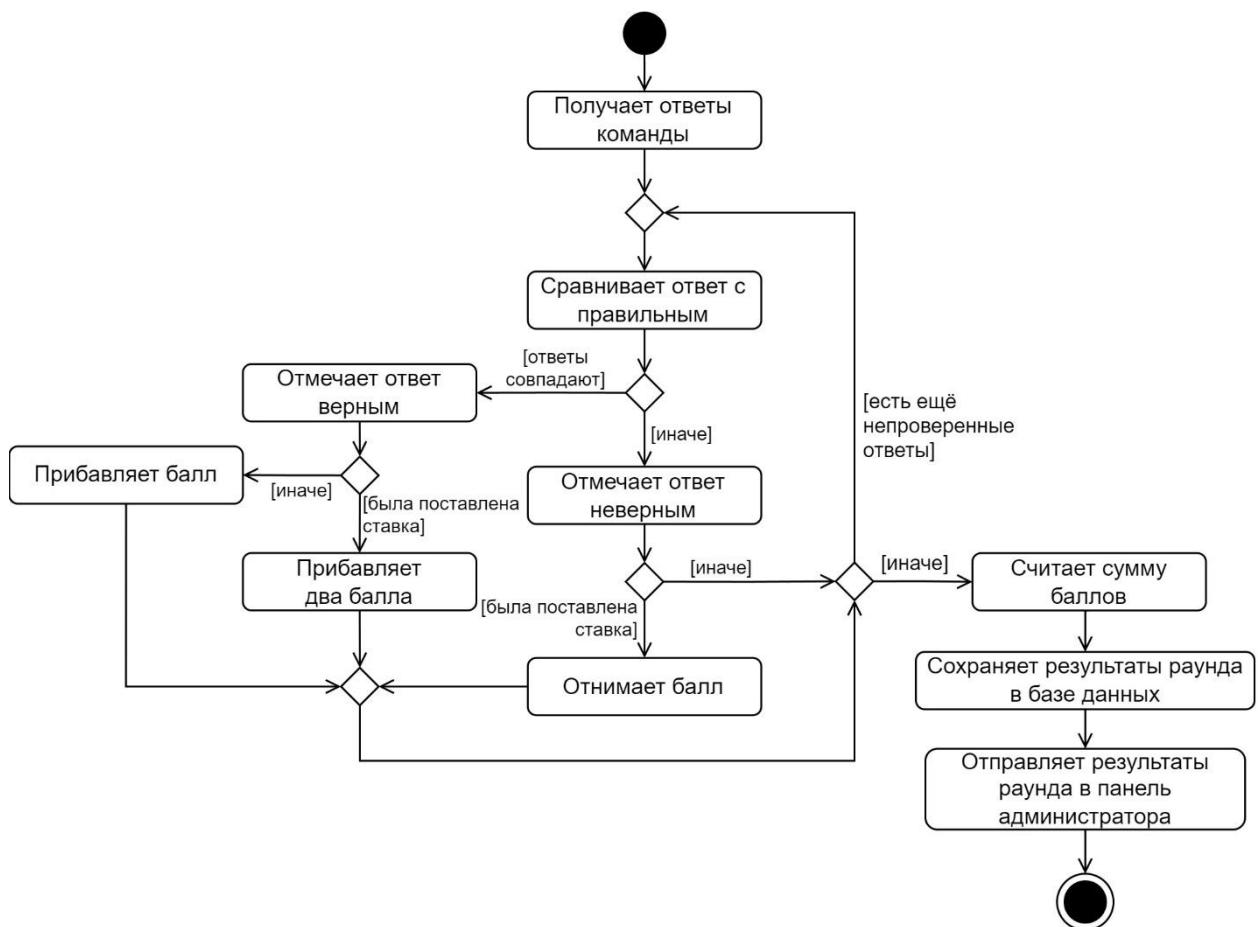


Рисунок 49 – Процесс подсчёта результатов раунда

3.3.3 Реализация сортировки итоговых результатов

На рисунке 50 представлена последовательность сортировки сервером итоговых результатов игры. Команды сортируются по убыванию итоговых баллов. При равенстве баллов у нескольких команд происходит дополнительная сортировка по последнему раунду, и так вплоть до первого. В конце отсортированные результаты отправляются в панель администратора.

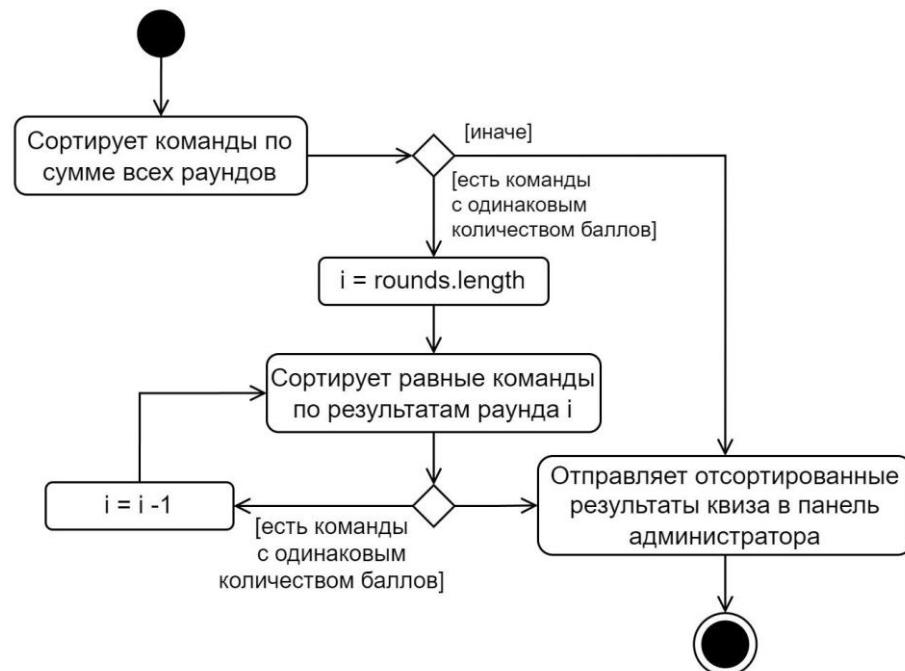


Рисунок 50 – Процесс сортировки итоговых результатов

ЗАКЛЮЧЕНИЕ

Таким образом, в рамках выпускной работы была разработана платформа для создания и проведения квизов. В процессе работы были успешно выполнены все поставленные задачи:

- 1) проанализированы и сформулированы требования;
- 2) определен необходимый набор инструментов;
- 3) спроектирована и разработана серверная часть веб-приложения;
- 4) спроектировано и разработано клиентское приложение для организаторов;
- 5) спроектировано и разработано клиентское приложение для участников.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. Документация React [Электронный ресурс]. URL: <https://react.dev/> (дата обращения: 15.02.2024).
2. Документация Redux [Электронный ресурс]. URL: <https://redux.js.org/> (дата обращения: 28.02.2024).
3. Документация TypeScript [Электронный ресурс]. URL: <https://www.typescriptlang.org/> (дата обращения: 12.03.2023).
4. Документация Ant Design [Электронный ресурс]. URL: <https://ant.design/> (дата обращения: 15.02.2024).
5. Руководство React [Электронный ресурс] URL: <https://metanit.com/web/react/> (дата обращения: 21.03.2024).
6. Справочник API Хуков [Электронный ресурс] URL: <https://lenarin.github.io/docs/hooks-reference.html/> (дата обращения: 22.03.2024).
7. Документация Node.js [Электронный ресурс]. URL: <https://nodejs.org/api/all.html/> (дата обращения: 07.04.2024).
8. Учебное пособие Express.js [Электронный ресурс]. URL: <https://www.geeksforgeeks.org/express-js/> (дата обращения: 15.04.2024).
9. Документация Socket.io [Электронный ресурс]. URL: <https://socket.io/docs/v4/> (дата обращения: 28.03.2024).
10. Руководство MongoDB [Электронный ресурс]. URL: <https://metanit.com/nosql/mongodb/> (дата обращения: 03.05.2024).



Отчет о проверке

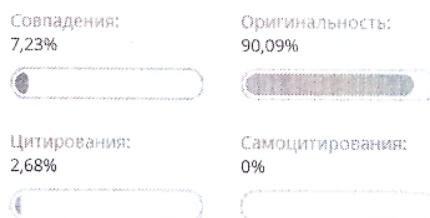
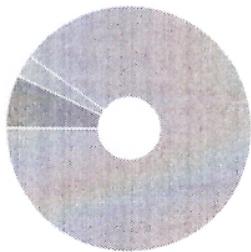
Автор: Назарова Мария Олеговна

Название документа: UKR_Nazarova (1)

Проверяющий: Политова Анастасия Михайловна

Организация: Томский Государственный Университет

РЕЗУЛЬТАТЫ ПРОВЕРКИ



«Совпадения», «Цитирования», «Самоцитирования», «Оригинальность» являются отдельными показателями, отображаются в процентах и в сумме дают 100%, что соответствует проверенному тексту документа.

- **Совпадения** — фрагменты проверяемого текста, полностью или частично сходные с найденными источниками, за исключением фрагментов, которые система отнесла к цитированию или самоцитированию. Показатель «Совпадения» – это доля фрагментов проверяемого текста, отнесенных к совпадениям, в общем объеме текста.
- **Самоцитирования** — фрагменты проверяемого текста, совпадающие или почти совпадающие с фрагментом текста источника, автором или соавтором которого является автор проверяемого документа. Показатель «Самоцитирования» – это доля фрагментов текста, отнесенных к самоцитированию, в общем объеме текста.
- **Цитирования** — фрагменты проверяемого текста, которые не являются авторскими, но которые система отнесла к корректно оформленным. К цитированием относятся также шаблонные фразы: библиография; фрагменты текста, найденные модулем поиска «СПС Гарант: нормативно-правовая документация». Показатель «Цитирования» – это доля фрагментов проверяемого текста, отнесенных к цитированию, в общем объеме текста.
- **Текстовое пересечение** — фрагмент текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника.
- **Источник** — документ, проиндексированный в системе и содержащийся в модуле поиска, по которому проводится проверка.
- **Оригинальный текст** — фрагменты проверяемого текста, не обнаруженные ни в одном источнике и не отмеченные ни одним из модулей поиска. Показатель «Оригинальность» – это доля фрагментов проверяемого текста, отнесенных к оригинальному тексту, в общем объеме текста.

Обращаем Ваше внимание, что система находит текстовые совпадения проверяемого документа с проиндексированными в системе источниками. При этом система является вспомогательным инструментом, определение корректности и правомерности совпадений или цитирований, а также авторства текстовых фрагментов проверяемого документа остается в компетенции проверяющего.

ИНФОРМАЦИЯ О ДОКУМЕНТЕ

Номер документа: 201

Количество страниц: 46

Тип документа: Выпускная квалификационная работа

Символов в тексте: 42091

Дата проверки: 04.06.2024 13:28:44

Слов в тексте: 4833

Дата корректировки: Нет

Число предложений: 3917

Комментарий: не указано

ПАРАМЕТРЫ ПРОВЕРКИ

Выполнена проверка с учетом редактирования: Да

Выполнено распознавание текста (OCR): Нет

Выполнена проверка с учетом структуры: Нет

Модули поиска: ИПС Адилет, Цитирование, СПС ГАРАНТ: нормативно-правовая документация, Перефразированные заимствования по коллекции Интернет в английском сегменте, Публикации РГБ, Патенты СССР, РФ, СНГ, Перефразирования по коллекции издательства Wiley, Публикации eLIBRARY (переводы и перефразирования), Коллекция НБУ, Библиография, Переводные заимствования*, Шаблонные фразы, Перефразирования по коллекции IEEE, СПС ГАРАНТ: аналитика, Диссертации НББ, Издательство Wiley, Кольцо вузов, Публикации eLIBRARY, СМИ России и СНГ, Медицина, Сводная коллекция ЭБС, IEEE, Перефразированные заимствования по коллекции Интернет в русском сегменте, Перефразирования по СПС ГАРАНТ: аналитика, Переводные заимствования IEEE, Перефразирования по Интернету (EN), Переводные заимствования по коллекции Интернет в русском сегменте, Перефразирования по коллекции Интернет в английском сегменте, Перефразирования по Интернету, Переводные заимствования издательства Wiley, Переводные заимствования по Интернету (EnRu), Кольцо вузов (переводы и перефразирования), Перефразирования по коллекции Гарант: аналитика, Переводные заимствования (RuEn), Интернет Плюс*, Собственная коллекция компании

ИСТОЧНИКИ

№	Доля в тексте	Доля в отчете	Источник	Актуален на	Модуль поиска	Комментарий
[01]	2,68%	2,68%	не указано	29 Сен 2022	Библиография	
[02]	1,8%	0,18%	1521791 Козлов	06 Июн 2023	Кольцо вузов	
[03]	1,48%	0%	2020BKP430316УЛЯШЕВ	02 Июн 2020	Кольцо вузов	
[04]	1,43%	0,22%	RODIN-VKR	27 Июн 2021	Кольцо вузов	
[05]	1,4%	1,4%	Диплом (full).pdf	31 Мая 2024	Кольцо вузов (переводы и перефразирования)	
[06]	1,21%	0,24%	Дубинский. 2антiplагиат	10 Июн 2022	Кольцо вузов	
[07]	1,06%	0%	VKR_SibagatullinRR_09-936	15 Июн 2021	Кольцо вузов	
[08]	1,05%	1,05%	отчет для антиплагиата	13 Янв 2024	Кольцо вузов (переводы и перефразирования)	
[09]	1%	0,19%	%D0%92%D1%8B%D0%BF%D1%8... https://stolypin-vestnik.ru	30 Апр 2024	Интернет Плюс*	
[10]	0,92%	0,6%	191468_b1-ifst41_2023_1	12 Окт 2023	Кольцо вузов	
[11]	0,89%	0,89%	Материалы одним файлом https://belstu.by	31 Окт 2022	Перефразированные заимствования по коллекции Интернет в русском сегменте	
[12]	0,89%	0%	ФИТ https://belstu.by	28 Фев 2023	Перефразированные заимствования по коллекции Интернет в русском сегменте	
[13]	0,88%	0,88%	Курсовая Рамазанов Родион 09-0...	13 Янв 2024	Кольцо вузов (переводы и перефразирования)	
[14]	0,86%	0%	Абдухосимов Абдукарим 09-032	13 Янв 2024	Кольцо вузов (переводы и перефразирования)	Источник исключен. Причина: Маленький процент пересечения.
[15]	0,86%	0%	ИССЛЕДОВАНИЕ СОВРЕМЕННОГО... https://cyberleninka.ru	25 Дек 2023	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[16]	0,84%	0%	Tyurin_VKR_14_06	21 Июн 2021	Кольцо вузов	
[17]	0,84%	0%	Разработка мобильного кросспл... http://elibrary.ru	27 Мая 2020	Публикации eLIBRARY	
[18]	0,83%	0%	Разработка локального многопол... https://elibrary.ru	20 Дек 2022	Публикации eLIBRARY	Источник исключен. Причина: Маленький процент пересечения.
[19]	0,81%	0,81%	Отчет по курсовой работе - Бала...	13 Янв 2024	Кольцо вузов (переводы и перефразирования)	
[20]	0,77%	0,77%	Fynbgkfubfn2	28 Дек 2020	Кольцо вузов	
[21]	0,77%	0%	BKP_АнисимоваO.O.pdf	11 Июн 2020	Собственная коллекция компании	

[22]	0,73%	0%	Разработка гибридного мобильн... https://vital.lib.tsu.ru	16 Фев 2024	Перефразированные заимствования по коллекции Интернет в русском сегменте	Источник исключен. Причина: Маленький процент пересечения.
[23]	0,69%	0%	Кузьмина Карина ВКР	15 Июн 2019	Собственная коллекция компании	Источник исключен. Причина: Маленький процент пересечения.
[24]	0,63%	0%	ОСОБЕННОСТИ РАЗРАБОТКИ ПР... http://elibrary.ru	01 Янв 2020	Публикации eLIBRARY	Источник исключен. Причина: Маленький процент пересечения.
[25]	0,62%	0%	https://hits.tsu.ru/sites/default/fil... https://hits.tsu.ru	03 Июн 2024	Интернет Плюс*	
[26]	0,61%	0%	8-APINO%2019.%20%D0%A2.1.... http://sut.ru	25 Mar 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[27]	0,61%	0%	Разработка web-приложения ин... https://vital.lib.tsu.ru	07 Mar 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[28]	0,56%	0%	https://www.muctr.ru/upload/iblo... https://muctr.ru	27 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[29]	0,54%	0%	Diplomnaya_Rabota_revise_7.docx	08 Июн 2023	Собственная коллекция компании	
[30]	0,53%	0%	Разработка TCP-чата с использо... https://moluch.ru	28 Апр 2023	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[31]	0,52%	0%	Разработка обучающей системы... https://moluch.ru	02 Июн 2023	Кольцо вузов	Источник исключен. Причина: Маленький процент пересечения.
[32]	0,48%	0%	Калугин_Никита_Юрьевич_ПИиК...	30 Мая 2024	Кольцо вузов (переводы и перефразирования)	Источник исключен. Причина: Маленький процент пересечения.
[33]	0,47%	0%	Разработка пользовательского и...	31 Мая 2024	Кольцо вузов (переводы и перефразирования)	Источник исключен. Причина: Маленький процент пересечения.
[34]	0,44%	0%	MongoDB - что это за СУБД, NoS... https://blog.skillfactory.ru	23 Мая 2022	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[35]	0,44%	0%	MongoDB - что это за СУБД, NoS... https://blog.skillfactory.ru	27 Мая 2022	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[36]	0,44%	0%	MongoDB - что это за СУБД, NoS... https://blog.skillfactory.ru	29 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[37]	0,43%	0%	React Введение в Redux https://metanit.com	27 Мая 2021	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[38]	0,43%	0%	React Введение в Redux https://metanit.com	03 Июн 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[39]	0,42%	0%	https://elar.urfu.ru/bitstream/109... https://elar.urfu.ru	31 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[40]	0,42%	0%	https://elar.urfu.ru/bitstream/109... https://elar.urfu.ru	31 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[41]	0,4%	0%	Копия ВКР	29 Мая 2024	Кольцо вузов (переводы и перефразирования)	Источник исключен. Причина: Маленький процент пересечения.
[42]	0,4%	0%	https://www.tsu.ru/upload/medial... https://tsu.ru	03 Июн 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[43]	0,38%	0%	000001FIT-pervij-shag-SNTK-2019.... http://belstu.by	04 Июн 2021	Перефразированные заимствования по коллекции Интернет в русском сегменте	Источник исключен. Причина: Маленький процент пересечения.
[44]	0,37%	0%	не указано	29 Сен 2022	Шаблонные фразы	Источник исключен. Причина: Маленький процент пересечения.
[45]	0,37%	0%	Инновационная активность и на... http://ivo.garant.ru	29 Окт 2022	СПС ГАРАНТ: аналитика	Источник исключен. Причина: Маленький процент пересечения.
[46]	0,32%	0%	Фроленко диплом.docx	07 Июн 2019	Собственная коллекция компании	Источник исключен. Причина: Маленький процент пересечения.
[47]	0,32%	0%	BKP Скворцовой М.Н.	15 Июн 2019	Собственная коллекция компании	Источник исключен. Причина: Маленький процент пересечения.
[48]	0,32%	0%	практика	20 Июн 2019	Собственная коллекция компании	Источник исключен. Причина: Маленький процент пересечения.
[49]	0,3%	0%	15 лекций по термодинамике с у... https://ibooks.ru	01 Янв 2021	Сводная коллекция ЭБС	Источник исключен. Причина: Маленький процент пересечения.
[50]	0,3%	0%	Динамическое взаимодействие ... https://ibooks.ru	01 Янв 2019	Сводная коллекция ЭБС	Источник исключен. Причина: Маленький процент пересечения.
[51]	0,3%	0%	Атмосферные процессы в облач... https://ibooks.ru	01 Янв 2021	Сводная коллекция ЭБС	Источник исключен. Причина: Маленький процент пересечения.
[52]	0,3%	0%	Психология стресса: системный ... https://ibooks.ru	01 Янв 2019	Сводная коллекция ЭБС	Источник исключен. Причина: Маленький процент пересечения.
[53]	0,27%	0%	https://rguk.ru/upload/medialibra... https://rguk.ru	15 Фев 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[54]	0,25%	0%	РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ... http://elibrary.ru	01 Янв 2023	Публикации eLIBRARY	Источник исключен. Причина: Маленький процент пересечения.
[55]	0,25%	0%	pdf https://vvsu.ru	25 Мая 2023	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.

[56]	0,25%	0%	http://sveden.ugrasu.ru/sveden/fi... http://sveden.ugrasu.ru	07 Mar 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[57]	0,24%	0%	https://gstou.ru/sveden/files/35.B... https://gstou.ru	31 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[58]	0,21%	0%	GitHub - FatB0YY/CLOUD-STORAG... https://github.com	25 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[59]	0,2%	0%	Оптимизация режимов работы с ... http://diss.natlib.uz	29 Авг 2014	Коллекция НБУ	Источник исключен. Причина: Маленький процент пересечения.
[60]	0,2%	0%	Преодоление фонетической инт... http://diss.natlib.uz	02 Сен 2014	Коллекция НБУ	Источник исключен. Причина: Маленький процент пересечения.
[61]	0,2%	0%	Очистка семян зерновых культу... http://dep.nlb.by	11 Ноя 2016	Диссертации НББ	Источник исключен. Причина: Маленький процент пересечения.
[62]	0,2%	0%	Самофлюсирующиеся композицио... http://dep.nlb.by	11 Ноя 2016	Диссертации НББ	Источник исключен. Причина: Маленький процент пересечения.
[63]	0,2%	0%	Разработка игрового виртуально... https://e.lanbook.com	22 Янв 2020	Сводная коллекция ЭБС	Источник исключен. Причина: Маленький процент пересечения.
[64]	0,18%	0%	Сравнение SQL Server и MongoDB... https://nauchniestati.ru	20 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[65]	0,17%	0%	Контроль механических напряж... http://dep.nlb.by	01 Янв 2023	Диссертации НББ	Источник исключен. Причина: Маленький процент пересечения.
[66]	0,17%	0%	Кто такой PHP разработчик, скол... https://dtf.ru	03 Июн 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.
[67]	0,16%	0%	Разработка платформенных реш... http://elibrary.ru	01 Янв 2023	Публикации eLIBRARY	Источник исключен. Причина: Маленький процент пересечения.
[68]	0,13%	0%	React и TypeScript: Практическое... https://kartaslov.ru	18 Мая 2024	Интернет Плюс*	Источник исключен. Причина: Маленький процент пересечения.