

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук  
Кафедра программирования и информационных технологий

Техническое задание  
на разработку мобильного и веб-приложений  
«Онлайн-платформа для дистанционного управления учебным процессом  
Ratingus».

Исполнители

\_\_\_\_\_ Д.Г. Шлыков

\_\_\_\_\_ П.А. Сапегин

\_\_\_\_\_ Е.А. Саков

Заказчик

\_\_\_\_\_ В.С. Тарасов

Воронеж 2024

## Содержание

Титульный лист .....	1
Содержание.....	2
1 Термины, обозначения и аббревиатуры .....	5
2 Общие положения .....	9
2.1 Название автоматизированной системы.....	9
2.2 Разработчики и заказчик .....	9
2.3 Перечень документов, на основании которых создается платформа .....	9
2.4 Порядок оформления и предъявления заказчику результатов работ по созданию приложения .....	10
3 Состав и содержание работ по созданию платформы.....	11
4 Назначение и цель создания приложения .....	12
4.1 Цели создания платформы .....	12
4.2 Назначение платформы .....	12
5 Требования к АС .....	13
5.1 Требования к структуре АС в целом.....	13
5.1.1 Структура АС и взаимосвязи её частей.....	13
5.1.2 Перспективы развития, модернизации АС .....	13
5.2 Требования к функциям (задачам), выполняемым АС .....	14
5.2.1 Требования к ролям пользователей в системе сервиса.....	14
5.2.2 Требования к функциональности роли «Неавторизованный пользователь» .....	15
5.2.3 Требования к функциональности роли «Авторизованный пользователь» .....	15

5.2.4 Внесение сведений об оценках и посещаемости учеников учителем.....	15
5.2.5 Выдача домашнего задания учителем .....	16
5.2.6 Создание расписания локальным администратором .....	16
5.2.7 Просмотр оценок и посещаемости учеником .....	16
5.2.8 Просмотр домашнего задания учеником .....	17
5.2.9 Просмотр расписания учеником .....	17
5.2.10 Запись домашнего задания учеником (создание заметок) .....	17
5.2.11 Создание и удаление объявлений .....	17
5.2.12 Просмотр объявлений .....	17
5.2.13 Администрирование виртуальной учебной организации..	18
5.2.14 Создание виртуальной учебной организации .....	18
5.3 Требования к видам обеспечения АС .....	19
5.3.1 Информационное обеспечение.....	19
5.3.2 Требования к численности и квалификации персонала и пользователей АС .....	19
5.4 Техническое обеспечение .....	19
5.5 Общие технические требования .....	20
5.5.1 Требования к программному обеспечению платформы .....	20
5.5.2 Требования к эргономике и технической эстетике .....	21
5.5.3 Требования совместимости.....	22
5.5.4 Требования безопасности .....	22
5.6 Требования к дизайну, интерфейсу и расположению элементов на экранах мобильного приложения и сайта платформы .....	22
5.6.1 Навигация .....	23
5.6.2 Лендинг .....	24
5.6.3 Регистрация в сервисе .....	27
5.6.4 Вход в сервис.....	28
5.6.5 Раздел «Профиль».....	30

5.6.6 Раздел «Дневник» .....	31
5.6.7 Раздел «Информация о предмете» .....	34
5.6.8 Раздел «Расписание» .....	36
5.6.9 Создание и просмотр объявлений .....	37
5.6.10 Полученные заявки на создание виртуальной учебной организации .....	39
5.6.11 Добавление пользователей в виртуальную учебную организацию .....	41
5.6.12 Журнал .....	44
6 Порядок контроля и приемки работ .....	46
7 Реквизиты и подписи сторон .....	47
Приложение А .....	48

## 1 Термины, обозначения и аббревиатуры

В техническом задании используются следующие термины, сокращения и аббревиатуры:

**Авторизация** – предоставление определённому лицу или группе лиц прав на выполнение определённых действий, а также процесс проверки данных прав при попытке выполнения этих действий.

**Аккаунт** – хранимая в компьютерной системе совокупность данных о пользователе, необходимая для его опознавания и предоставления доступа к его личным данным и настройкам.

**АС** – автоматизированная система.

**Аутентификация** – процедура проверки подлинности пользователя путём сравнения введённого им пароля с паролем, сохранённым в базе данных пользовательских логинов.

**БД** – база данных.

**Валидный** – соответствующий требованиям, достоверный.

**Виртуальная учебная организация** – наименование представления учебной организации внутри платформы.

**Класс** – термин, имеющий значение обычного школьного класса как группы людей. Также под понятием «класс» подразумевается категория пользователей, имеющих общие школьные активности.

**Клиент** – аппаратный или программный компонент вычислительной системы, посылающий запросы серверу (в контексте настоящего документа клиент – это браузер или мобильное приложение пользователя сервиса).

**Код приглашения** – то же, что уникальный одноразовый код.

**Лендинг** – страница сайта или экран приложения, которые призывают пользователя что-то сделать.

**Логин** – имя пользователя, которое выступает в качестве идентификатора пользователя (или идентификатора учётной записи).

**Локальный администратор** – одна из возможных ролей, которой может обладать пользователь отдельно в рамках каждой виртуальной организации. Пользователь с данной ролью занимается управлением и настройкой виртуальной учебной организации.

**Менеджер платформы** – одна из пользовательских ролей на разрабатываемой платформе. Пользователь с данной ролью занимается созданием виртуальных учебных организаций.

**Паттерн проектирования** – часто встречающееся решение определённой проблемы при проектировании архитектуры программ.

**Платформа** – то же, что сервис.

**ПО** – программное обеспечение.

**Пользователь** – человек, который использует платформу.

**Роль** (пользователя) – абстракция, принятая для деления пользователей на группы по уровням доступа и возможностей.

**Сервер** – программный компонент вычислительной системы, выполняющий сервисные (обслуживающие) функции по запросу клиента, предоставляя ему доступ к определённым ресурсам или услугам.

**Сервис** – разрабатываемый программный продукт, который рассматривается с точки зрения функций и возможностей, предоставляемых пользователю.

**Таск-менеджер** – сервис для планирования и управления задачами.

**Уникальный одноразовый код** – уникальный код, используемый внутри платформы. Представляет из себя последовательность символов и генерируется менеджером платформы или локальным администратором. С помощью ввода кода пользователь может подключиться к виртуальной учебной организации. После первого использования код становится недействительным.

**Учебный процесс** – целенаправленное взаимодействие преподавателя и учащихся, в ходе которого решаются задачи образования, развития и воспитания учащихся.

**Ученик** – одна из возможных ролей, которой может обладать пользователь отдельно в рамках каждой виртуальной организации. Пользователь с данной ролью может просматривать информацию об оценках, посещаемости, выданном домашнем задании, а также расписание в рамках виртуальной учебной организации, к которой имеет доступ.

**Учитель** – одна из возможных ролей, которой может обладать пользователь отдельно в рамках каждой виртуальной организации. Пользователь с данной ролью может в рамках виртуальной учебной организации, к которой имеет доступ, заполнять информацию об оценках и посещаемости своих учеников, а также выдавать им домашнее задание.

**API** (Application Programming Interface) – описание способов взаимодействия одной компьютерной программы с другими.

**FSD** (Feature-Sliced Design) – это архитектурная методология для проектирования клиентской части приложений.

**CPU** (Central Processing Unit) – центральный процессор.

**MVVM** (Model-View-ViewModel) – это паттерн разработки, позволяющий разделить приложение на три функциональные части:

- **Model** – основная логика программы (работа с данными, вычисления, запросы и так далее);
- **View** – вид или представление (пользовательский интерфейс);
- **ViewModel** – модель представления, которая служит прослойкой между View и Model.

**MVC** (Model-View-Controller) – схема разделения данных приложения и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо.

**NVMe** (Non-Volatile Memory Express)– это протокол хранения данных, разработанный специально для твердотельных накопителей. «NVM» в названии спецификации обозначает энергонезависимую память.

**RAM** (Random Access Memory) – оперативная память.



## **2 Общие положения**

### **2.1 Название автоматизированной системы**

Полное наименование: «Онлайн-платформа для дистанционного управления учебным процессом Ratingus».

Условное обозначение платформы: «Ratingus».

### **2.2 Разработчики и заказчик**

Заказчик: Старший Преподаватель Тарасов Вячеслав Сергеевич, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Представитель заказчика: Ассистент Проскуряков Егор Дмитриевич, Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Разработчик: «7.7» команда группы «7»

Состав команды разработчика:

- Шлыков Данила Георгиевич;
- Сапегин Павел Александрович;
- Саков Евгений Андреевич.

### **2.3 Перечень документов, на основании которых создается платформа**

Проект разрабатывается на основании следующих документов:

- федерального закона от 29.12.2012 N 273-ФЗ (ред. от 25.12.2023) «Об образовании в Российской Федерации»;
- закона РФ от 07.02.1992 N 2300-1 (ред. от 11.06.2021) «О защите прав потребителей»;
- федерального закона «О персональных данных» от 27.07.2006 N 152-ФЗ.

## **2.4 Порядок оформления и предъявления заказчику результатов работ по созданию приложения**

Согласование и корректировка направления работы над проектом будет проводиться раз в две недели в формате очной встречи. Отчёты по работе над проектом будут проводиться во время рубежных аттестаций:

- 1 аттестация (середина марта 2024): предоставлена ссылка на репозиторий проекта на GitHub, где хранятся все необходимые ссылки на документы и сервисы (ссылка на проект Figma с первоначальным приблизительным дизайном приложения и сайта; ссылка на проект в YouTrack, который является хранилищем документации и таск-менеджером); распределены роли в команде; предоставлены промежуточные результаты по курсовому проекту и готовое техническое задание;
- 2 аттестация (конец апреля 2024): написана основополагающая часть программного кода приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка программного кода, проведено тестирование по работе системы;
- 3 аттестация (конец мая 2024): разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

### 3 Состав и содержание работ по созданию платформы

Состав и содержание работ по созданию платформы включают в себя следующие этапы (см. таблицу):

Таблица 1 – Сроки работ

Срок	Этап, работы
16.02.24 – 01.03.24	Сбор необходимой информации, постановка целей, задач, которые в будущем должны быть реализованы. Анализ аналогов и первоначальное проектирование системы.
01.03.24 – 14.03.24	Анализ имеющейся информации. Построение структуры требований, ведущих к решению поставленных задач и целей. Написание и согласование технического задания. Продумывание структуры разрабатываемой платформы.
14.03.24 – 01.04.24	Построение модели программы, описание спецификаций данных, определение связей между сущностями, разработка модели БД. Проработка архитектуры платформы. Подготовка к разработке.
01.04.24 – 01.05.24	Разработка рабочего проекта, состоящего из написания программного кода, отладки и корректировки кода программы. Первоначальное тестирование.
01.05.24 – 10.06.24	Проведение тестирования программного обеспечения. Внесение корректировок в программный код и документацию. Подготовка к защите проекта.
10.06.24	Сдача проекта

## **4 Назначение и цель создания приложения**

### **4.1 Цели создания платформы**

Целями создания платформы являются:

- систематизация учебных процессов в школе (включая выставление оценок, отметку посещаемости, хранение расписания, выдачу домашних заданий);
- упрощение доступа учащихся к учебной информации (включая оценки и посещаемость, расписание, домашние задания).

Критерий достижения целей создания платформы: используя разрабатываемую платформу, учащийся может получить доступ к такой информации, как:

- его оценки;
- его посещаемость;
- его домашние задания;
- его расписание.

Вопрос соответствия критериям рассматривается при условии, что учителя и локальный администратор конкретной виртуальной учебной организации должным образом загрузили в систему всю имеющуюся у них информацию об оценках, посещаемости, домашнем задании и расписании.

### **4.2 Назначение платформы**

Платформа предназначена для удобного доступа учащихся к:

- оценкам;
- посещаемости;
- выданному домашнему заданию;
- расписанию уроков.

Также платформа призвана систематизировать хранение домашнего задания с помощью реализации функции записи заметок к предметам.

## **5 Требования к АС**

### **5.1 Требования к структуре АС в целом**

#### **5.1.1 Структура АС и взаимосвязи её частей**

Разрабатываемый сервис будет иметь следующую структуру.

Сайт и мобильное приложение будут взаимодействовать с единым сервером для обмена данными в качестве клиентов.

Вся загружаемая на сервер информация будет храниться в базе данных, подключённой к нему.

Сайт предназначен для пользователей всех ролей. Пройдя аутентификацию, учителя могут в конкретной виртуальной учебной организации заполнять информацию об учебных процессах; локальные администраторы – настраивать виртуальную учебную организацию; менеджеры платформ – создавать виртуальные учебные организации.

Мобильное приложение предназначено для учащихся. После аутентификации учащиеся могут просматривать информацию в виртуальных учебных организациях, заполненную учителями и локальным администратором.

#### **5.1.2 Перспективы развития, модернизации АС**

Далее приведено несколько идей для развития и модернизации разрабатываемой АС. В АС в перспективе можно добавить:

- возможность платы за использование сервиса виртуальной учебной организацией;
- возможность использования платформы любой учебной организацией, не только школой;
- возможность формирования расписания с учётом чётности недель (числитель/знаменатель);
- возможность вести каждым авторизованным пользователем личный дневник, для которого он сам составляет расписание, а также может

подгружать его туда из всех виртуальных учебных организаций, в которых состоит, чтобы отслеживать всю свою занятость;

— роль «Родитель».

## **5.2 Требования к функциям (задачам), выполняемым АС**

В приложении А содержатся диаграммы вариантов использования (Use Case):

- Диаграмма вариантов использования для неавторизованного и авторизованного пользователей;
- Диаграмма вариантов использования для ученика и учителя;
- Диаграмма вариантов использования для локального администратора;
- Диаграмма вариантов использования для менеджера платформы.

Они отражают основные возможности пользователей различных ролей в рамках платформы.

### **5.2.1 Требования к ролям пользователей в системе сервиса**

В рамках сервиса пользователь должен иметь одну из следующих ролей:

- неавторизованный пользователь;
- авторизованный пользователь;
- менеджер платформы.

В рамках виртуальных учебных организаций пользователь будет иметь одну из следующих ролей:

- ученик;
- учитель;
- локальный администратор.

Пользователи разных ролей должны иметь разные права доступа и возможности.

Роли пользователей могут изменяться.

### **5.2.2 Требования к функциональности роли «Неавторизованный пользователь»**

Неавторизованный пользователь может:

- просматривать лендинг;
- войти в аккаунт, введя логин и пароль;
- зарегистрироваться в сервисе (создать аккаунт), задав логин и пароль, а также указав дату рождения и ФИО.

### **5.2.3 Требования к функциональности роли «Авторизованный пользователь»**

Авторизованный пользователь должен обладать следующими возможностями в рамках платформы:

- изменять личные данные в разделе «Профиль» (дату рождения и ФИО);
- вводить в разделе «Профиль» уникальный одноразовый код для подключения виртуальной учебной организации с определённой ролью;
- отправлять заявку на создание новой виртуальной учебной организации.

### **5.2.4 Внесение сведений об оценках и посещаемости учеников учителем**

Учитель в рамках конкретной организации должен иметь возможности (в разделе «Журнал»):

- ставить оценки ученикам по пятибалльной шкале («1», «2», «3», «4», «5»);
- редактировать поставленные оценки;
- отмечать их посещаемость уроков;
- редактировать отмеченную посещаемость.

В качестве вариантов для отметки посещаемости должны быть предусмотрены следующие варианты:

- «отсутствовал по уважительной причине»;
- «отсутствовал по неуважительной причине».

Ставить оценки и отмечать посещаемость учитель должен только своим ученикам по своим предметам.

Как результат, выставленные оценки и посещаемость должны сохраняться в системе платформы. Впоследствии они должны быть доступны к просмотру пользователям соответствующих ролей.

### **5.2.5 Выдача домашнего задания учителем**

Учитель в рамках конкретной организации должен иметь возможности (в разделе «Журнал»):

- выдавать домашнее задание ученикам по тем предметам, которые он у них ведёт;
- редактировать выданное домашнее задание.

Как результат, домашнее задание должно сохраняться в системе платформы. Впоследствии оно должно быть доступно к просмотру пользователям всех ролей с соответствующими правами доступа.

### **5.2.6 Создание расписания локальным администратором**

Локальный администратор в рамках конкретной организации должен обладать возможностями:

- с помощью предоставляемого графического интерфейса создавать расписание в электронном формате;
- редактировать составленное расписание.

Как результат, расписание должно сохраняться в системе платформы. Впоследствии оно должно быть доступно к просмотру пользователям всех ролей с соответствующими правами доступа.

### **5.2.7 Просмотр оценок и посещаемости учеником**

Ученик в рамках конкретной организации должен иметь возможности:



- просмотреть свои оценки по предметам;
- просмотреть свою посещаемость.

### **5.2.8 Просмотр домашнего задания учеником**

Ученик в рамках конкретной организации должен иметь возможность:

- просмотреть выданное учителем домашнее задание.

### **5.2.9 Просмотр расписания учеником**

Ученик в рамках конкретной организации должен иметь возможность:

- просмотреть своё расписание.

### **5.2.10 Запись домашнего задания учеником (создание заметок)**

Ученик в рамках конкретной организации должен иметь возможности:

- создавать заметки, видимые только ему, отдельно по каждому предмету;
- просматривать созданные заметки;
- редактировать созданные заметки.

Пример диаграммы последовательности процесса создания и редактирования заметки учеником представлен в приложении А.

Пример диаграммы состояний заметки содержится в приложении А.

### **5.2.11 Создание и удаление объявлений**

Учитель и локальный администратор должны обладать возможностью создания и публикации текстовых объявлений в разделе «Объявления», указав, для какого класса они должны отображаться.

Учитель и локальный администратор должны обладать возможностью удаления объявлений, написанных ими, в разделе «Объявления» в рамках виртуальной учебной организации. Локальный администратор также должен иметь возможность удалить объявление, опубликованное учителем.

### **5.2.12 Просмотр объявлений**

В рамках каждой виртуальной учебной организации любой пользователь должен иметь возможность просматривать опубликованные объявления:

- ученики: опубликованные для тех классов, в которых они состоят;
- учителя и локальные администраторы: все опубликованные объявления.

### **5.2.13 Администрирование виртуальной учебной организации**

Локальный администратор в рамках конкретной организации должен иметь возможности:

- генерировать уникальные одноразовые коды для пользователей, дающие доступ к виртуальной учебной организации в роли ученика или учителя. При этом предварительно заполняются поля «Роль», «ФИО» и, если добавляется ученик, «Класс», под которыми пользователь будет выполнять действия в рамках виртуальной учебной организации;
- редактировать роли пользователей, состоящих в виртуальной учебной организации (ученик или учитель);
- редактировать информацию об учебной организации (адрес, контакты (электронная почта, номер телефона), название);
- создавать классы (задавая название);
- подписывать учеников на предметы;
- создавать предметы (задавая название);
- редактировать предметы (название).

### **5.2.14 Создание виртуальной учебной организации**

Менеджер платформы должен обладать возможностями:

- создать виртуальную учебную организацию в системе;
- сгенерировать уникальный одноразовый ключ для получения роли локального администратора в созданной виртуальной учебной организации.

Диаграмма активности для отображения процесса создания виртуальной учебной организации и получения в ней пользователем роли локального администратора представлена в приложении А.

## **5.3 Требования к видам обеспечения АС**

### **5.3.1 Информационное обеспечение**

Для работы сервиса необходимо, чтобы учебная информация постоянно обновлялась. За внесение всех актуальных данных виртуальной учебной организации (оценки, посещаемость, расписание, домашнее задание, информация об учебном заведении) отвечают пользователи с соответствующими ролями этой виртуальной учебной организации (учитель и локальный администратор).

Процесс создания виртуальных учебных организаций в сервисе не автоматизирован. Для создания виртуальных учебных организаций и выдачи в них роли локального администратора необходим пользователь с ролью менеджера платформы, который будет делать описанные манипуляции вручную.

### **5.3.2 Требования к численности и квалификации персонала и пользователей АС**

Для обеспечения работоспособности сервиса необходим один пользователь с ролью менеджера платформы.

Также для каждой виртуальной учебной организации как минимум требуется пользователь с ролью локального администратора. Лучший вариант использования виртуальной учебной организации достигается тогда, когда каждый реальный учитель имеет доступ к платформе и загружает туда необходимую информацию.

Специальной квалификации для работы с платформой не требуется.

## **5.4 Техническое обеспечение**

Для стабильной работы серверного ПО должны выполняться следующие аппаратные и системные требования:

- операционная система: Ubuntu 22.04;
- минимальный CPU: 1 x 3.3 ГГц;
- минимальный объём RAM: 1 ГБ;

- минимальный размер твердотельного накопителя NVMe: 15ГБ;
- минимальное количество серверов: 1;
- минимальная сеть: 100Мбит/с.

## **5.5 Общие технические требования**

### **5.5.1 Требования к программному обеспечению платформы**

Сайт должен иметь архитектуру, соответствующую модели Клиент-Серверного взаимодействия на основе REST API.

Для реализации серверной части будут использоваться следующие средства:

- язык программирования Java 21;
- фреймворк Spring Boot 3.2.3;
- база данных PostgreSQL 14.

Основной паттерн для проектирования сервера – MVC (Model-View-Controller).

Для реализации клиентской части сайта будут использоваться следующие средства:

- язык программирования JavaScript ES6;
- язык программирования TypeScript 5.0.4;
- среда выполнения для JavaScript Node.js 18.18.2;
- фреймворк Next.js 14.1.1.

При проектировании клиентской части сайта будет использоваться архитектурная методология FSD.

Для реализации клиентской части мобильного приложения будут использоваться следующие средства:

- язык программирования Dart 3.3.0;
- фреймворк Flutter 3.19.2.

Основной паттерн для проектирования клиентской части мобильного устройства – MVVM (Model-View-ViewModel).

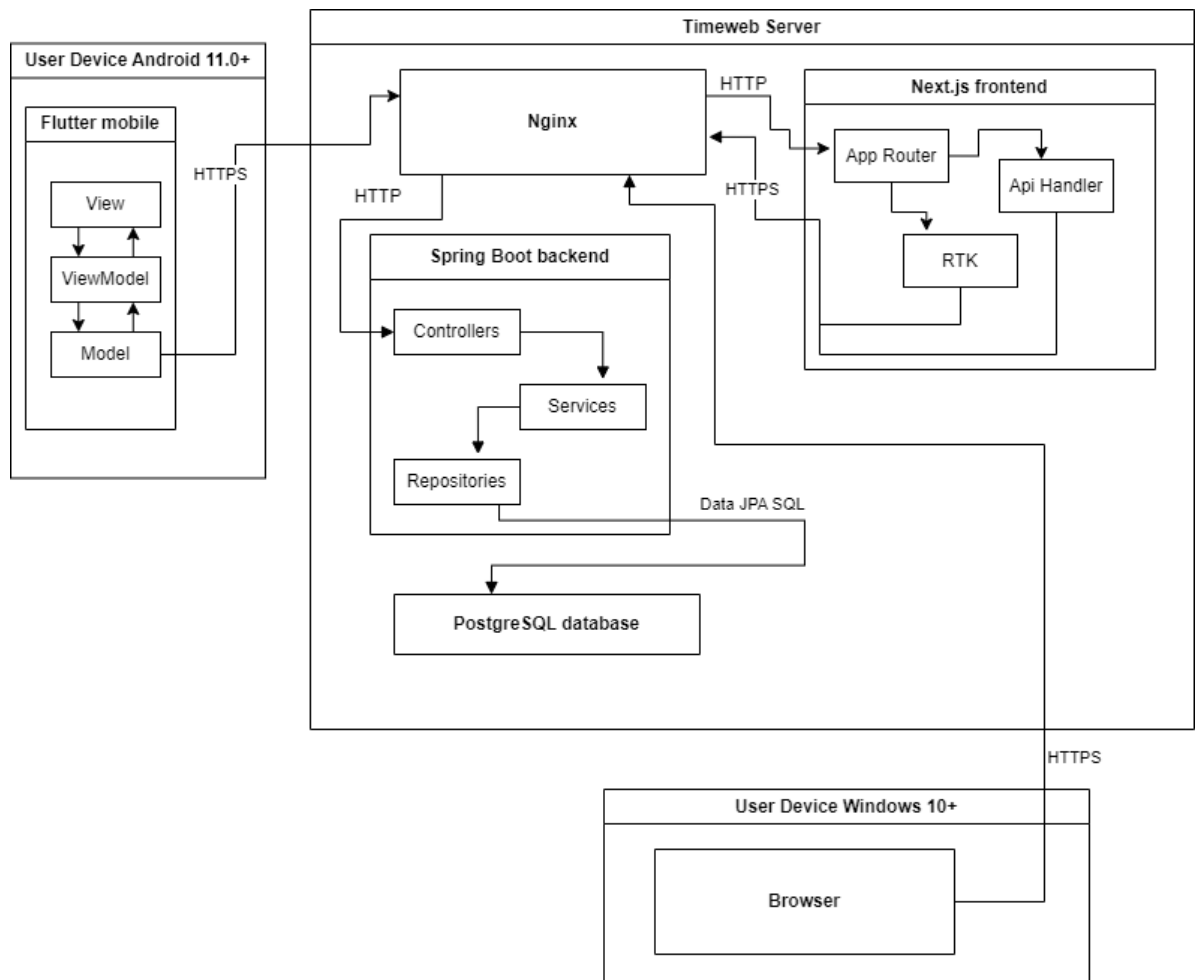


Рисунок 1 - Архитектура и структура реализуемого сервиса

### 5.5.2 Требования к эргономике и технической эстетике

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы, должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами.

Стиль оформления страниц сайта и экранов приложения должен быть единообразным. Нужно придерживаться одной цветовой палитры и использовать ограниченный набор шрифтов.

### **5.5.3 Требования совместимости**

Веб-приложение должно корректно работать на операционной системе Windows 10 и выше в следующих версиях браузеров:

- Google Chrome 117.0.5938.89 и выше;
- Яндекс.Браузер 22.1.0.2510 и выше.

Мобильное приложение должно корректно работать на устройствах, использующих операционную систему Android 11.0 и выше.

### **5.5.4 Требования безопасности**

Платформа имеет следующие требования к безопасности:

- все компоненты системы, взаимодействующие внутри сервера, должны передавать данные по протоколу HTTP, однако взаимодействие между сервером и клиентом должно осуществляться только через протокол HTTPS;
- система должна предоставлять конфиденциальные данные только после успешной авторизации пользователя через JWT, который должен передаваться в заголовке Authorization;
- все пароли в системе должны храниться в хешированном виде. Применяемый алгоритм хеширования должен быть pbkdf2-sha512 из пакета `org.springframework.security.crypto.password`;
- при работе с внешними сервисами или API система должна использовать безопасные методы аутентификации и авторизации, такие как OAuth или API ключи.

## **5.6 Требования к дизайну, интерфейсу и расположению элементов на экранах мобильного приложения и сайта платформы**

Далее представлены описания основных экранов мобильного приложения и сайта. Рисунки отображают примерные дизайн и расположение элементов. Реальный интерфейс может отличаться, не изменяя основное наполнение экрана и функций основных элементов.

Следующие подразделы будут описывать экраны и страницы для основных сценариев использования.

### 5.6.1 Навигация

Для веб-приложения есть header (шапка) экрана. На ней слева расположен логотип, который перенаправляет на экран лендинга, а справа – иконки разделов (или кнопка «Войти»).



Рисунок 2 - Шапка для неавторизованного пользователя



Рисунок 3 - Шапка для авторизованного пользователя



Рисунок 4 - Шапка для пользователя с ролью «Ученик». Разделы слева направо: «Объявления», «Дневник», «Расписание», «Профиль»



Рисунок 5 - Шапка для пользователя с ролью «Учитель». Разделы слева направо: «Объявления», «Расписание», «Журнал», «Профиль»



Рисунок 6 - Шапка для пользователя с ролью «Локальный администратор». Разделы слева направо: «Объявления», «Расписание», «Журнал», «Админ-панель», «Профиль»



Рисунок 7 - Шапка для пользователя с ролью «Менеджер платформы». Разделы слева направо: «Админ-панель», «Профиль»

Мобильное приложение предназначено для учеников, поэтому его навигационная панель содержит только четыре иконки, соответствующие навигационной панели в веб приложении для роли «Ученик».

В мобильном приложении навигационная панель находится внизу экрана.



Рисунок 8 - Навигационная панель мобильного приложения. Разделы слева направо: «Объявления», «Дневник», «Расписание», «Профиль»

### 5.6.2 Лендинг

Экран показывает вводную информацию о проекте, почему стоит выбрать именно наш сервис. Экран можно листать. В конце находится кнопка «Присоединяйтесь к нам!» для добавления своей организации в сервис.



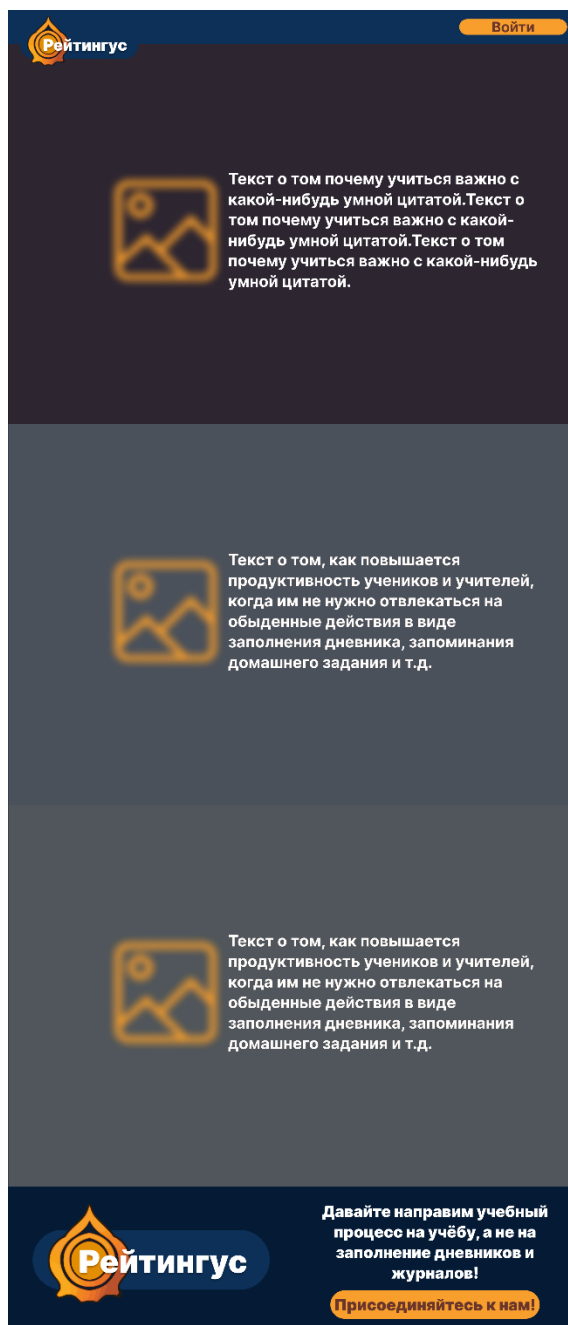


Рисунок 9 - Развёртка экрана лендинга

При нажатии на кнопку «Присоединяйтесь к нам!» происходят следующие действия:

- если пользователь не авторизован: открывается экран входа в сервис;
- если пользователь авторизован: открывается панель для регистрации организации.

На панели для регистрации организации находятся поля с заголовками «Электронная почта организации», «Название организации» и «Адрес организации». Затем идёт флаговая кнопка, куда нужно установить галочку для подписания «Пользовательского соглашения». Внизу панели – кнопка «Отправить заявку».

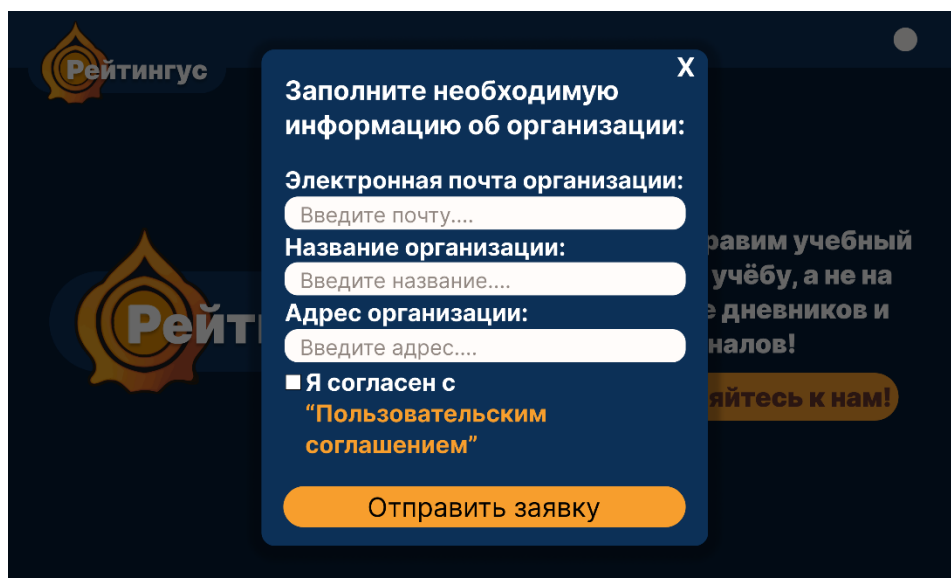
The image shows a dark-themed mobile application interface. In the top left corner, there is a logo with a flame and the word "Рейтингус". A modal form is displayed in the center. The form has a title "Заполните необходимую информацию об организации:" followed by three input fields: "Электронная почта организации:" with placeholder "Введите почту...", "Название организации:" with placeholder "Введите название...", and "Адрес организации:" with placeholder "Введите адрес...". Below these fields is a checkbox labeled "Я согласен с 'Пользовательским соглашением'". At the bottom of the form is an orange button labeled "Отправить заявку". In the background, there is a large, semi-transparent "Рейт" logo and some text about a learning diary.

Рисунок 10 - Отображение панели для регистрации организации

После заполнения формы и нажатия на кнопку «Отправить заявку» появляется окно с информацией о том, что заявка принята в обработку.

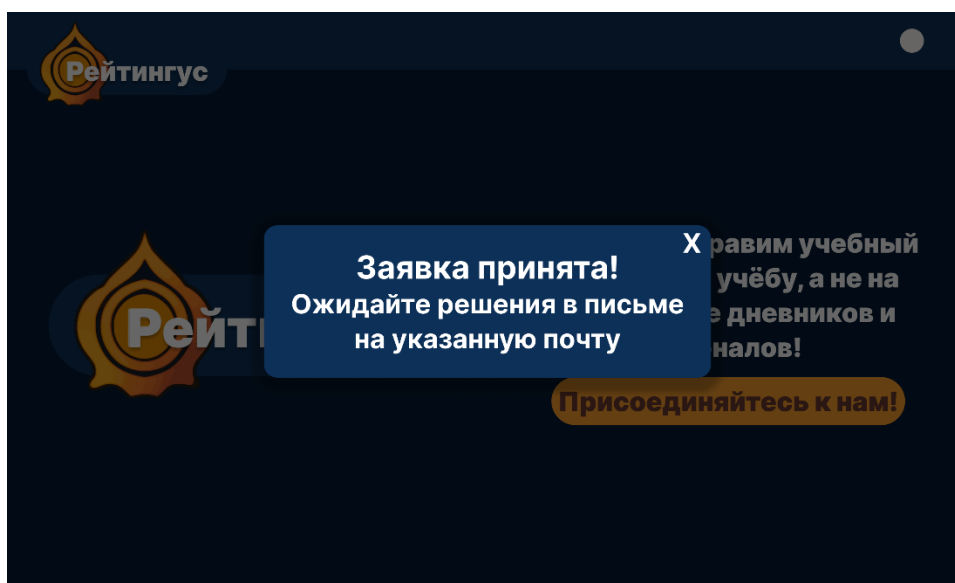


Рисунок 11 - Отображение окна с информацией о том, что заявка принята в обработку

### 5.6.3 Регистрация в сервисе

На экране регистрации отображаются поля для ввода логина, пароля, ФИО и даты рождения. Также рядом с ними находится картинка профиля, где отображаются первые символы указанных фамилии и имени. Далее расположены кнопки «Регистрация» и «Вход».

Для регистрации пользователю нужно заполнить поля и нажать кнопку «Регистрация». Также он может перейти на экран входа, нажав кнопку «Вход».

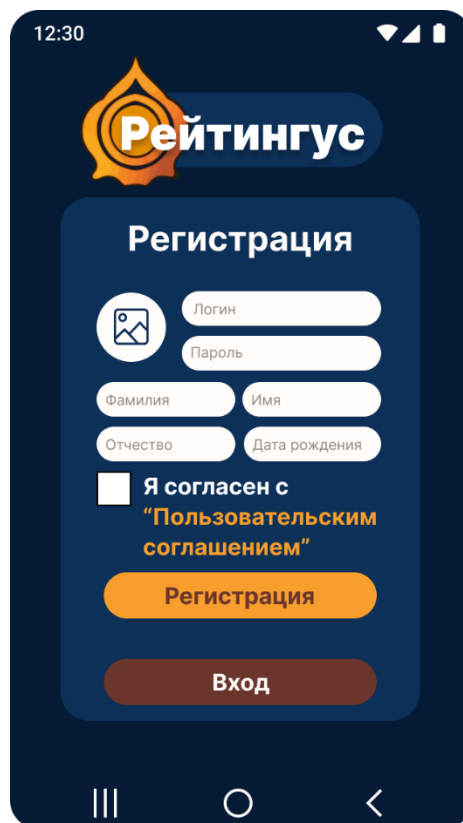


Рисунок 12 - Экран регистрации в мобильном приложении

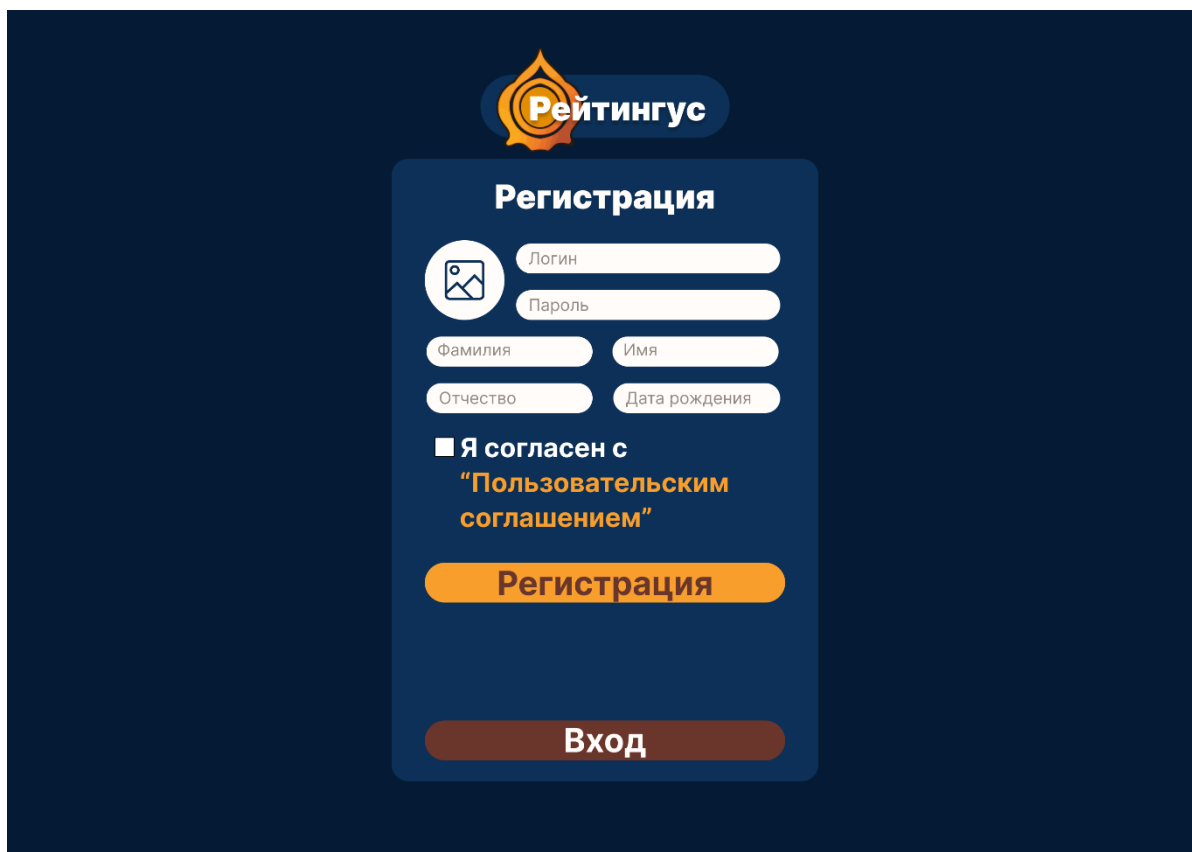


Рисунок 13 - Экран регистрации на сайте

#### 5.6.4 Вход в сервис

На экране входа отображаются поля для ввода логина и пароля. Далее находятся кнопки «Вход» и «Регистрация».

Для входа пользователю нужно заполнить поля и нажать кнопку «Вход». Также он может перейти на экран регистрации, нажав кнопку «Регистрация».

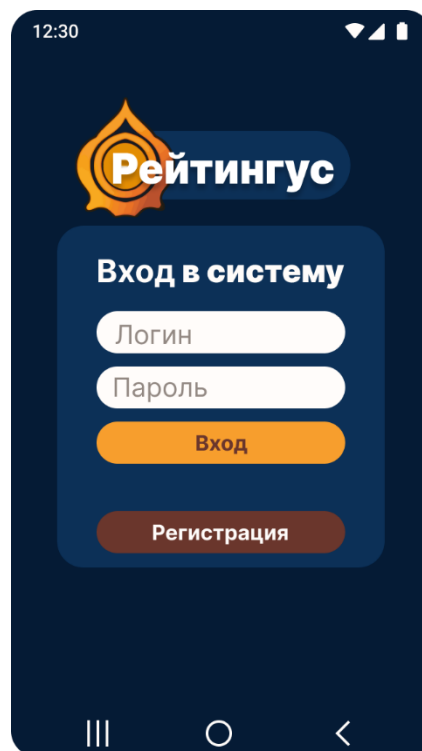


Рисунок 14 - Экран входа в мобильном приложении

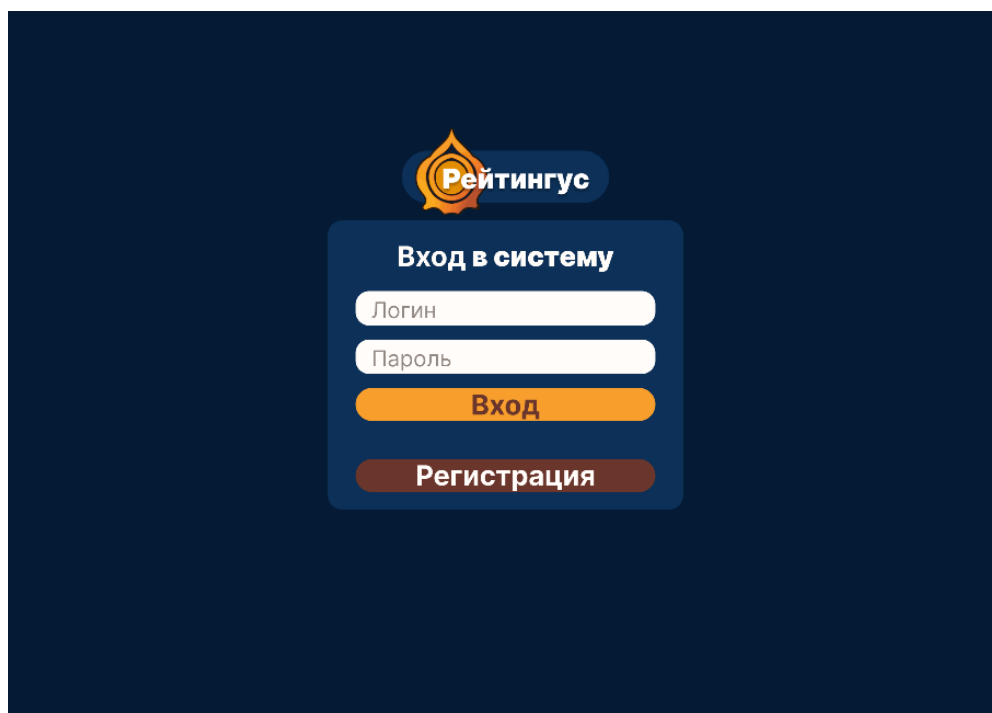


Рисунок 15 - Экран входа на сайте

### 5.6.5 Раздел «Профиль»

На этом экране пользователь видит всю информацию о себе: логин, ФИО, дату рождения и виртуальные учебные организации, в которых он состоит, с его ролью в ней и классами.

Вверху присутствует картинка профиля, слева – кнопка для ввода кода приглашения (уникального одноразового кода), справа – кнопка для редактирования информации профиля.

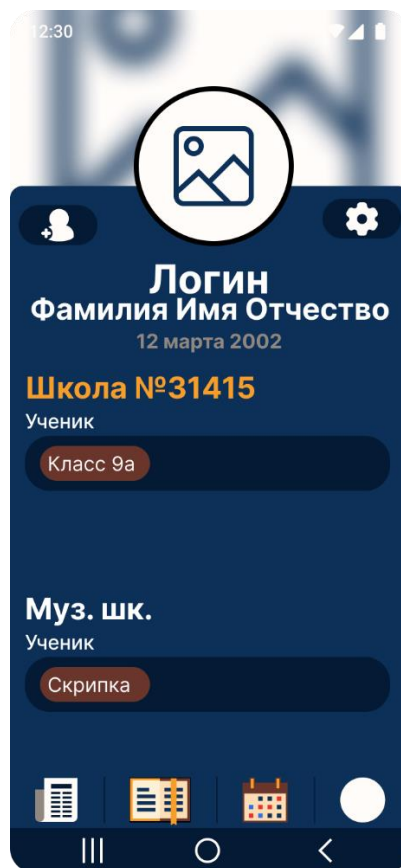


Рисунок 16 - Экран раздела «Профиль» для мобильного приложения

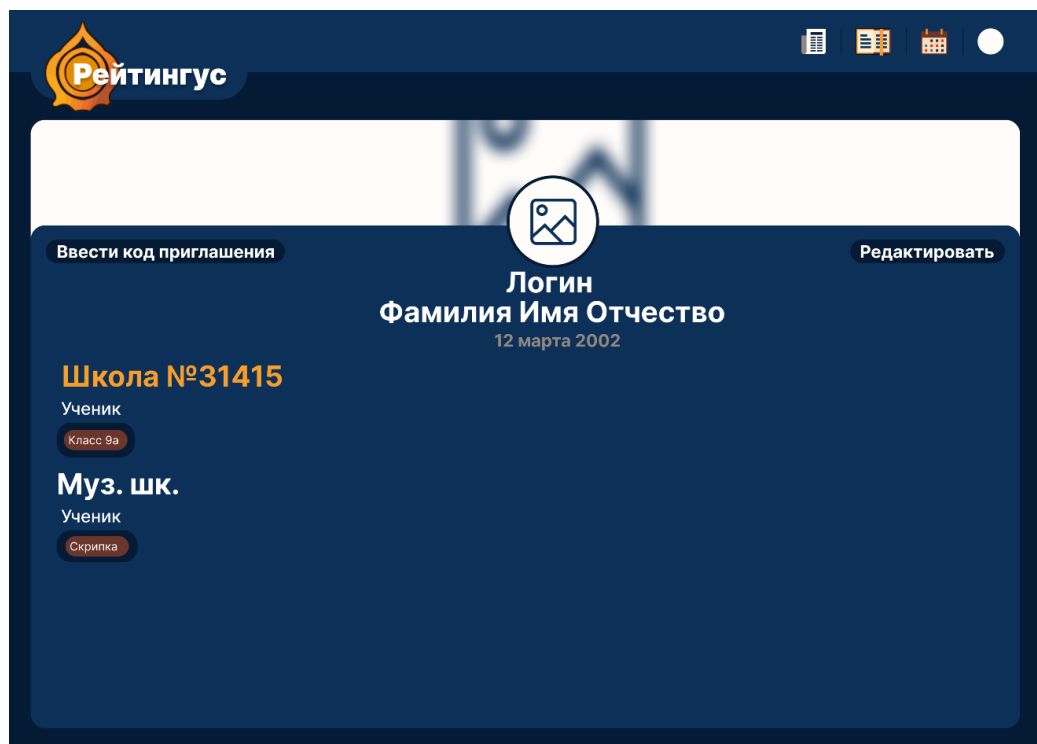


Рисунок 17 - Экран раздела «Профиль» для сайта

#### 5.6.6 Раздел «Дневник»

Вверху раздела можно переключаться между неделями с помощью кнопок.

Далее идёт кликабельный список всех дней недели с расписаниями, а также оценками и посещаемостью.

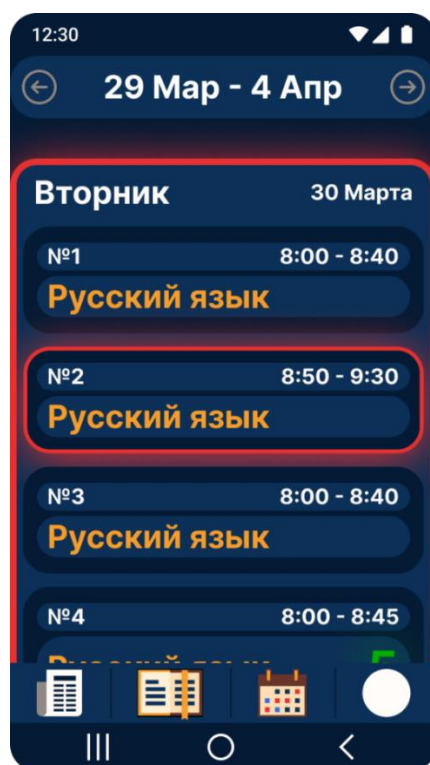


Рисунок 18 - Экран раздела «Дневник» в мобильном приложении

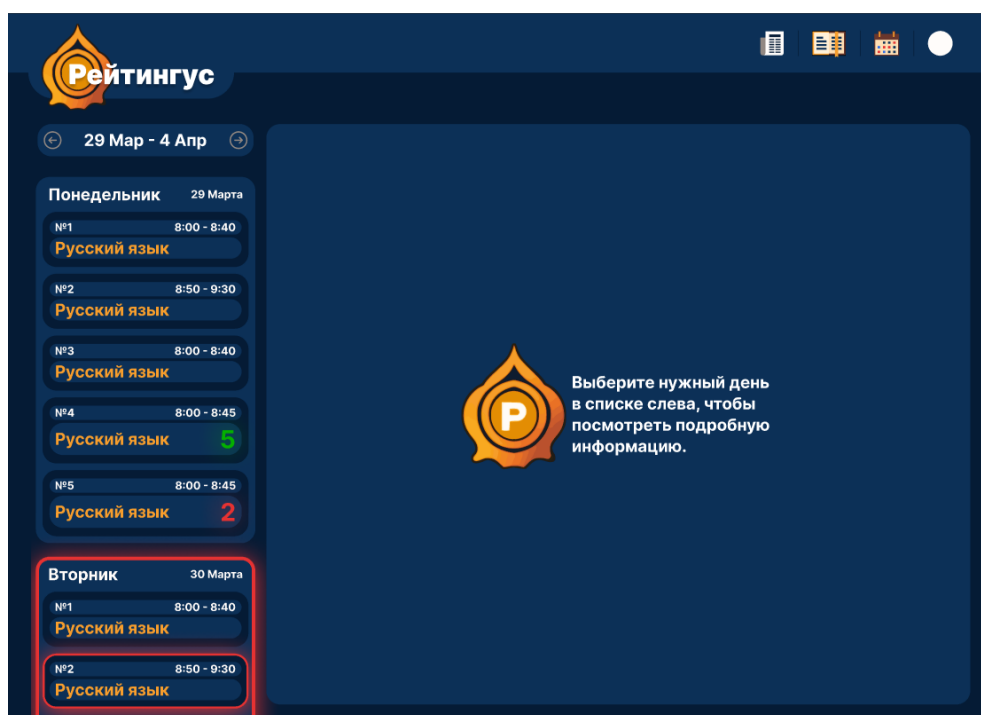


Рисунок 19 - Экран раздела «Дневник» на сайте



Если нажать на день недели, то откроется просмотр конкретного дня, в котором будут отображены оценки, посещаемость, домашние задания и заметки для каждого дня (возможно, кратко).

Вверху находятся кнопки, чтобы листать дни.

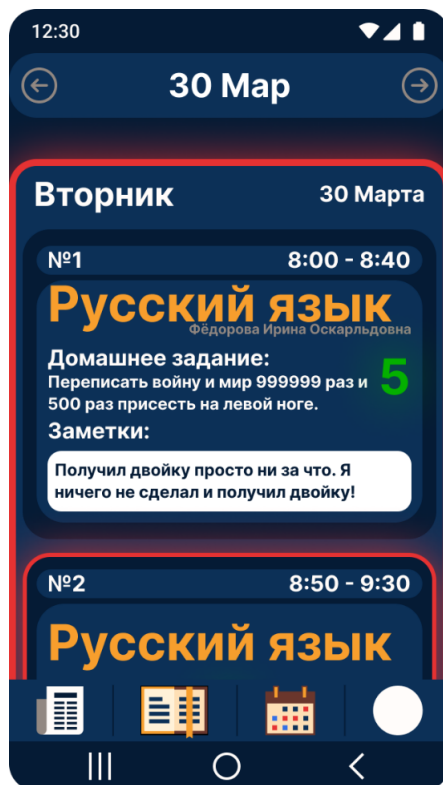


Рисунок 20 - Экран раздела «Дневник» при просмотре конкретного дня в мобильном приложении

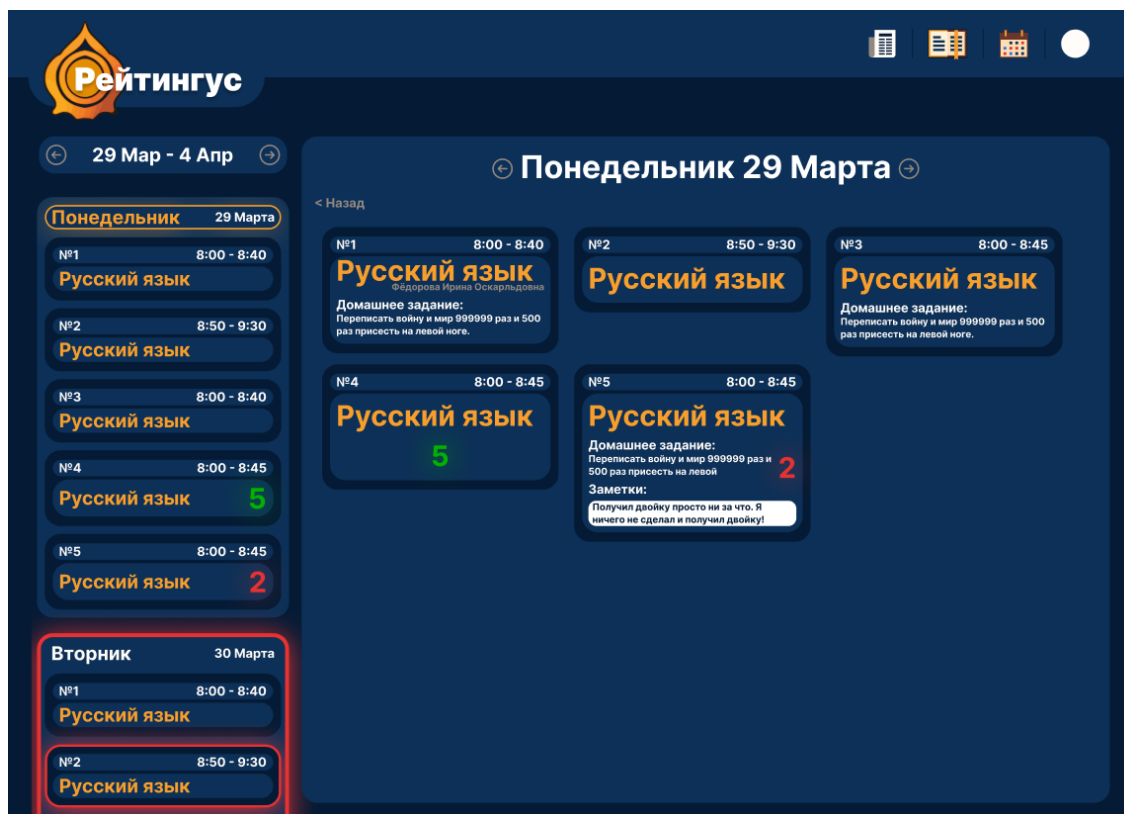


Рисунок 21 - Экран раздела «Дневник» при просмотре конкретного дня на сайте

Если нажать на карточку предмета, то откроется раздел «Информация о предмете».

### 5.6.7 Раздел «Информация о предмете»

В этом разделе отображаются оценки, посещаемость, домашние задания, заметки по этому предмету, а также редактируемая заметка.

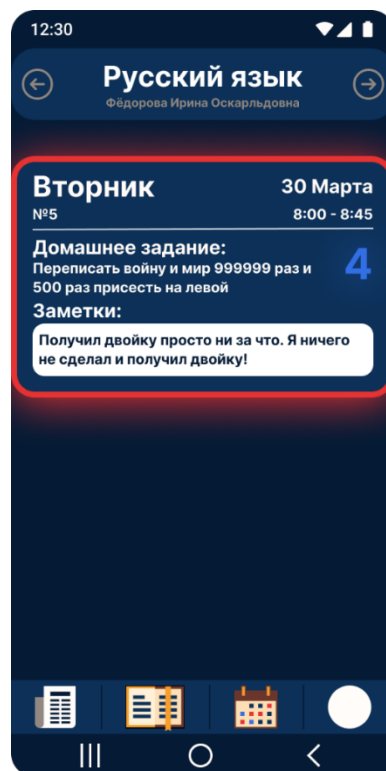


Рисунок 22 - Экран раздела «Информация о предмете» в мобильном приложении

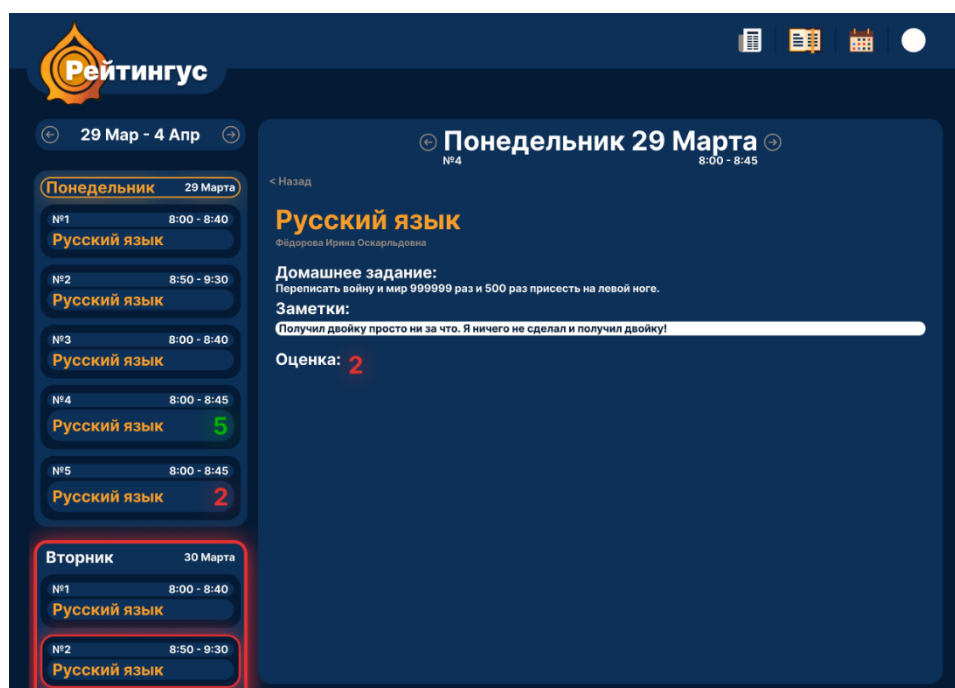


Рисунок 23 - Экран раздела «Информация о предмете» на сайте

### 5.6.8 Раздел «Расписание»

Вверху экрана находится выпадающий список для выбора класса, расписание которого нужно отобразить. По умолчанию выбран тот класс, к которому принадлежит ученик.

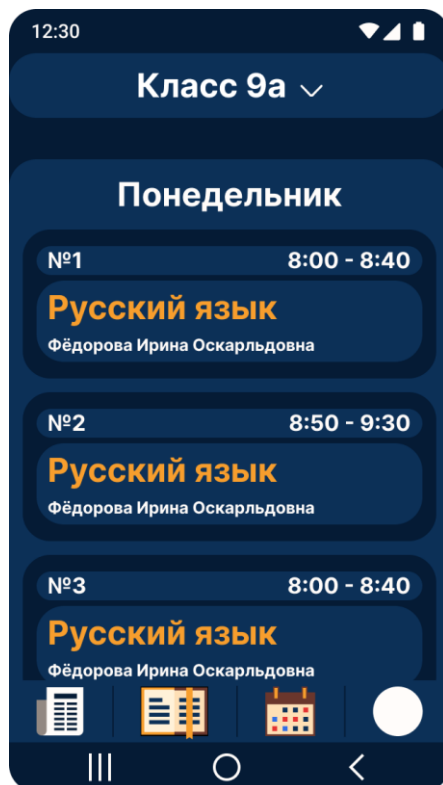


Рисунок 24 - Экран раздела «Расписание» в мобильном приложении

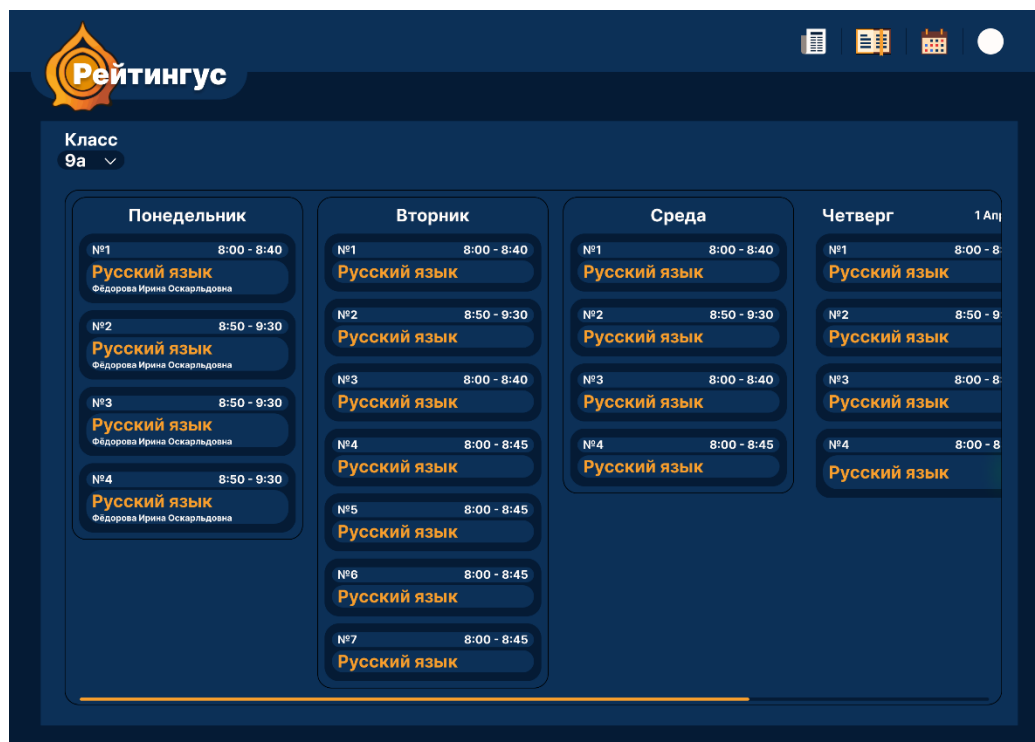


Рисунок 25 - Экран раздела «Расписание» на сайте

### 5.6.9 Создание и просмотр объявлений

Лента объявлений представляет из себя список записей, которые содержат:

- заголовок;
- текст;
- автора;
- дату публикации;
- упоминание классов, которым объявление адресовано;
- количество просмотров записи.

Вверху отображены вкладки «Все объявления», где показаны объявления для всей школы и «Класс», где показаны объявления, затрагивающие только конкретный класс, в котором состоит ученик.

На сайте есть также вкладка «+», через которую можно перейти в интерфейс создания объявления (вкладка не отображается у учеников).

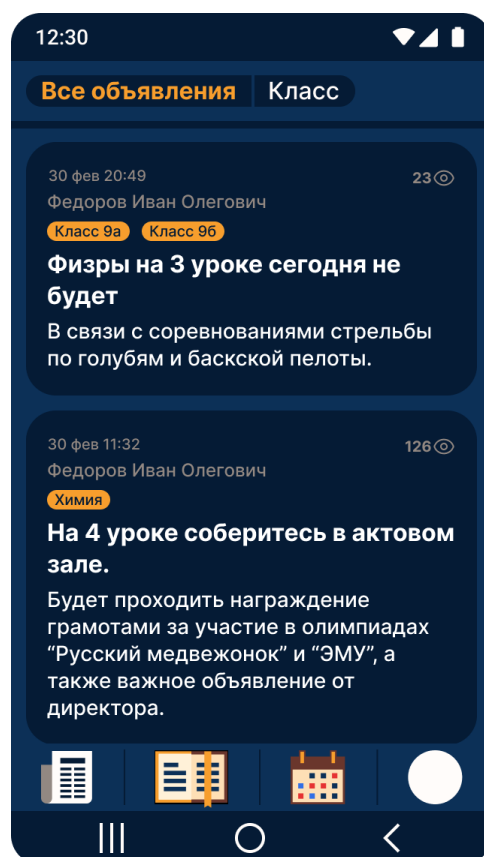


Рисунок 26 - Экран раздела «Объявления» в мобильном приложении

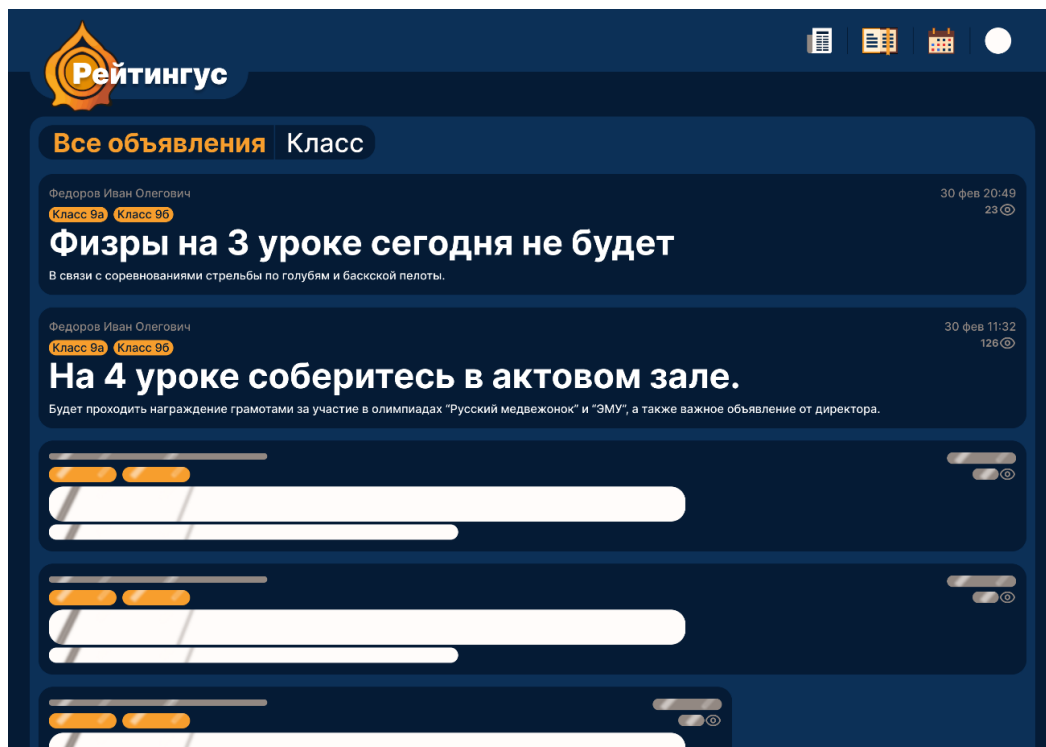


Рисунок 27 - Экран раздела «Объявления» на сайте (для учеников)

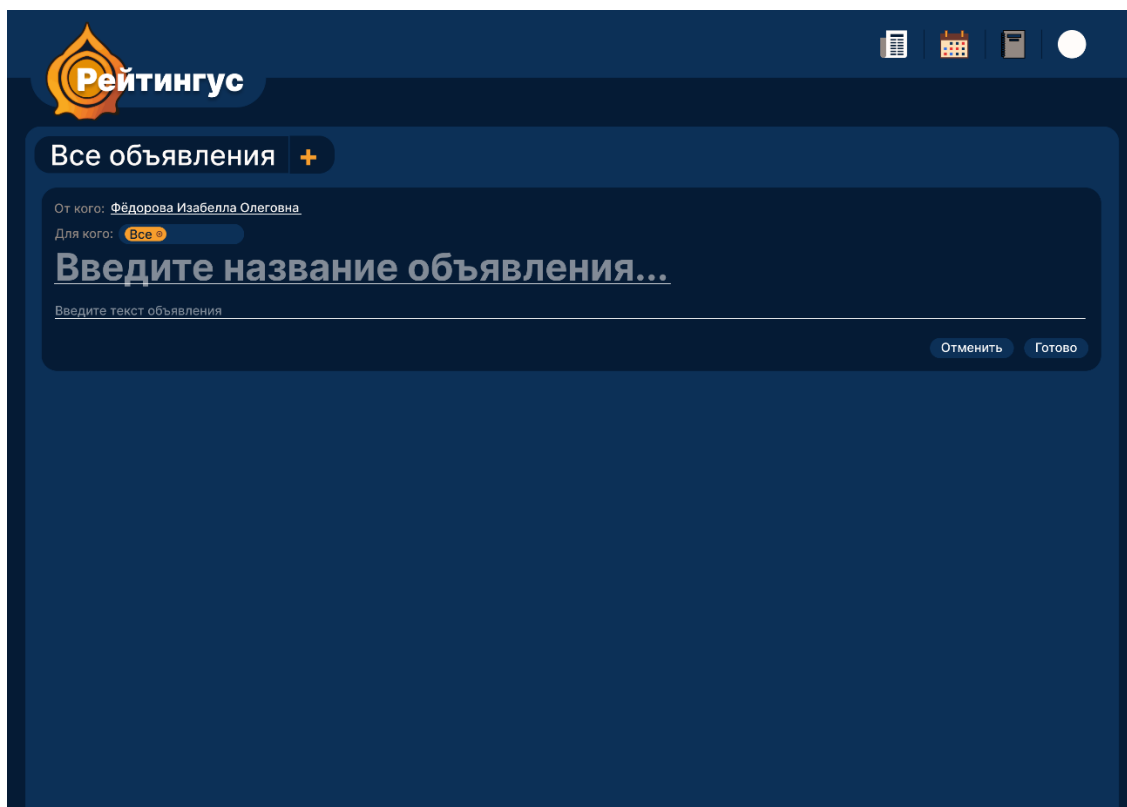


Рисунок 28 - Интерфейс создания объявления

#### **5.6.10 Полученные заявки на создание виртуальной учебной организации**

Этот экран доступен только «Менеджеру платформы». Здесь он может просматривать и одобрять заявки на создание новых виртуальных учебных организаций.

Рядом с каждой заявкой есть кнопка «Одобрить», после нажатия на которую появляется панель для создания «Локального администратора» для создаваемой организации.

Панель содержит кнопку «Добавить» и три поля, которые необходимо заполнить: фамилия, имя и отчество. После заполнения и нажатия на кнопку происходит создание виртуальной учебной организации, а рядом с одобренной заявкой отображается код приглашения для «Локального администратора».

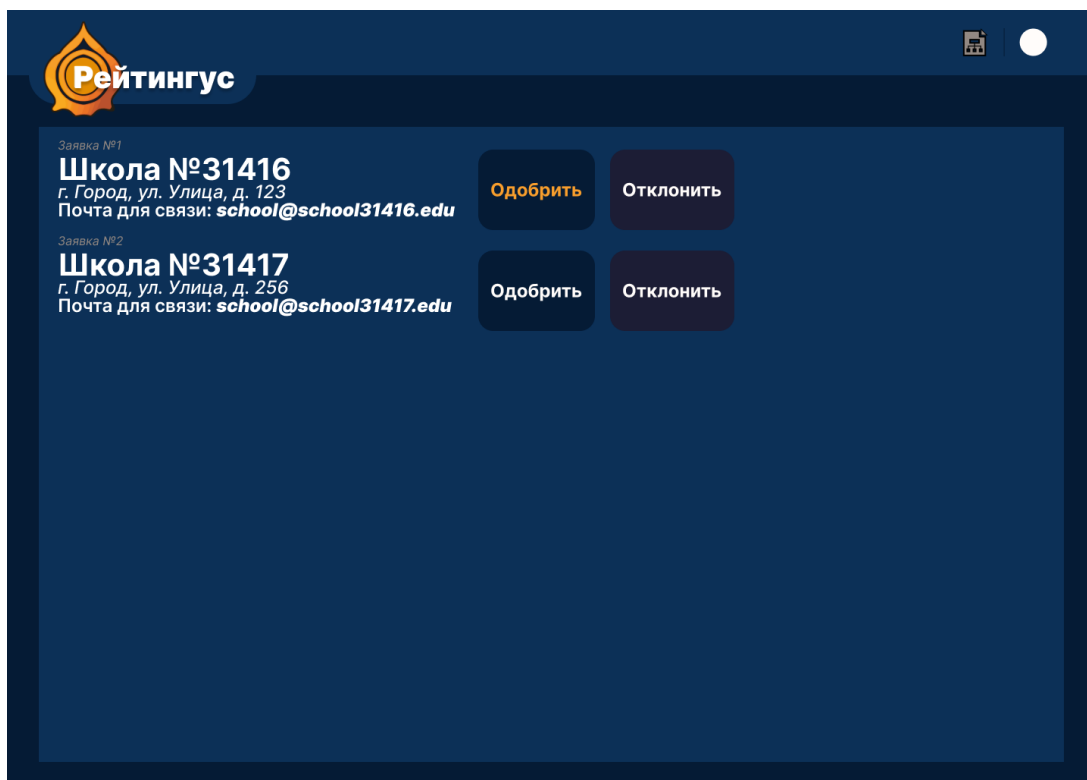


Рисунок 29 - Экран раздела просмотра заявок на создание виртуальных учебных организаций на сайте

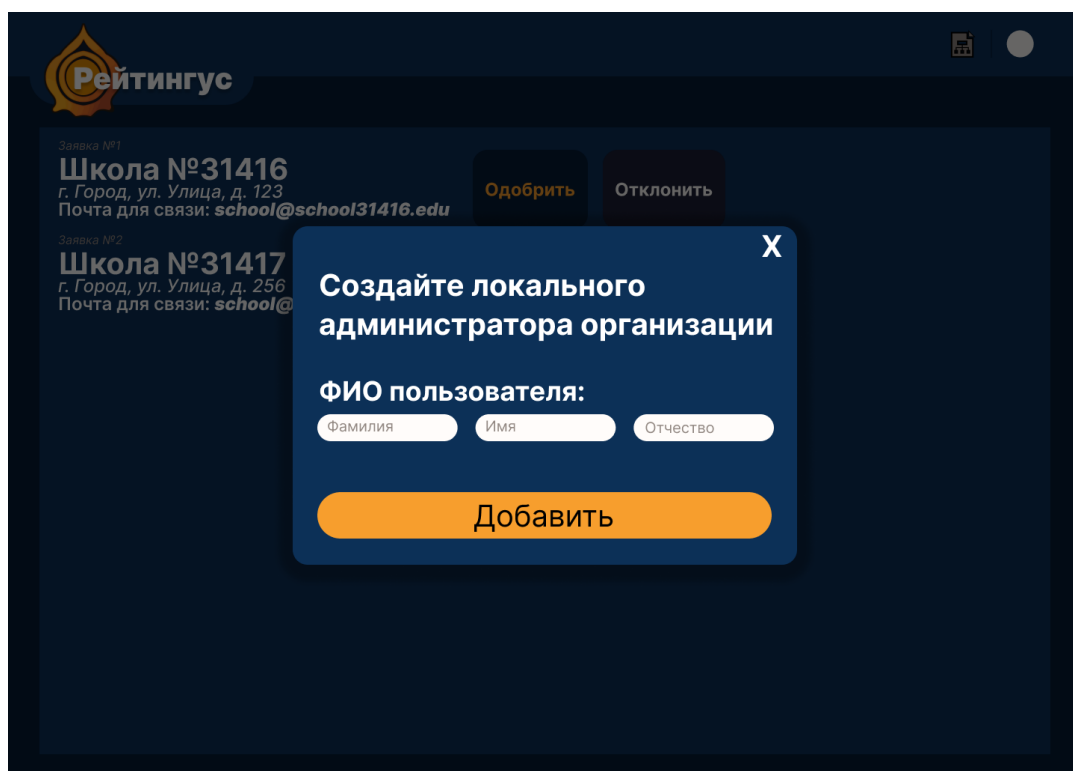


Рисунок 30 - Отображение панели для создания локального администратора на сайте



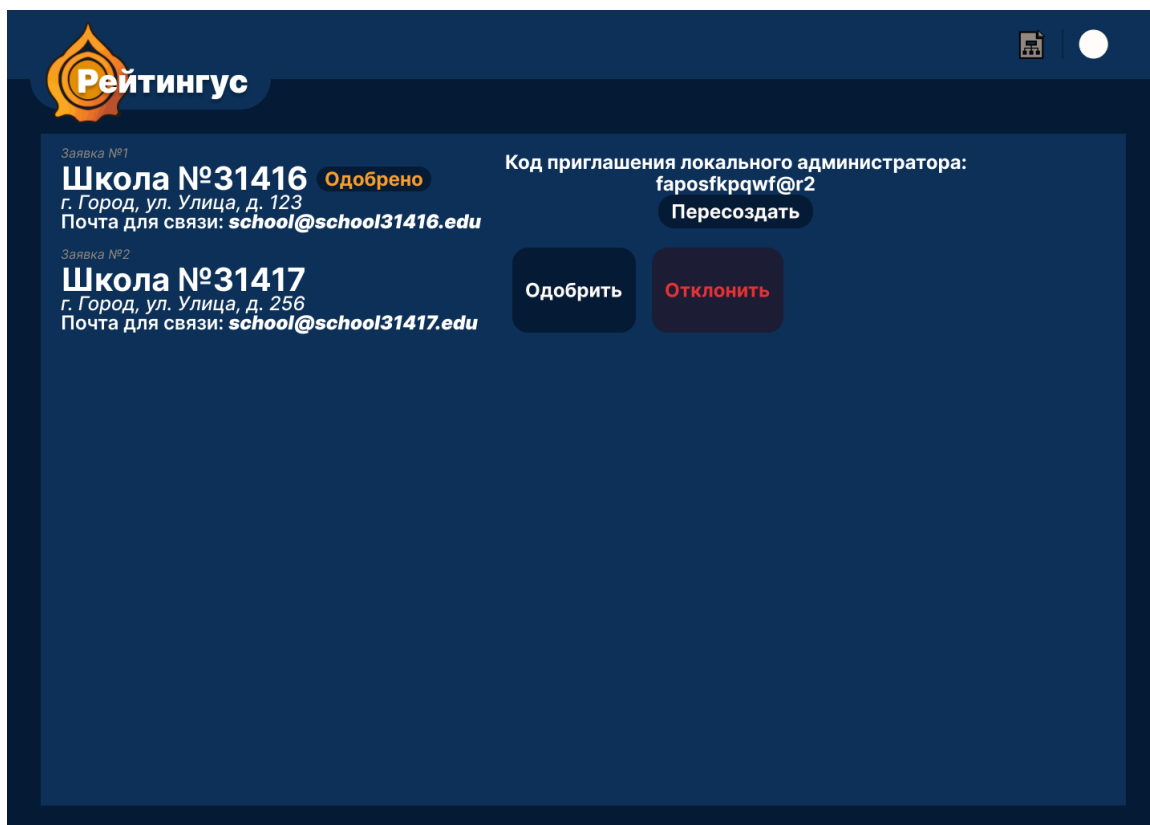


Рисунок 31 - Одобренная заявка и код приглашения для локального администратора на сайте

#### 5.6.11 Добавление пользователей в виртуальную учебную организацию

Этот экран виден только локальному администратору. Здесь он может искать и просматривать профили пользователей в учебной организации.

Есть поиск по логинам и ФИО, список пользователей, а также кнопка (или активная ссылка) для добавления пользователей.

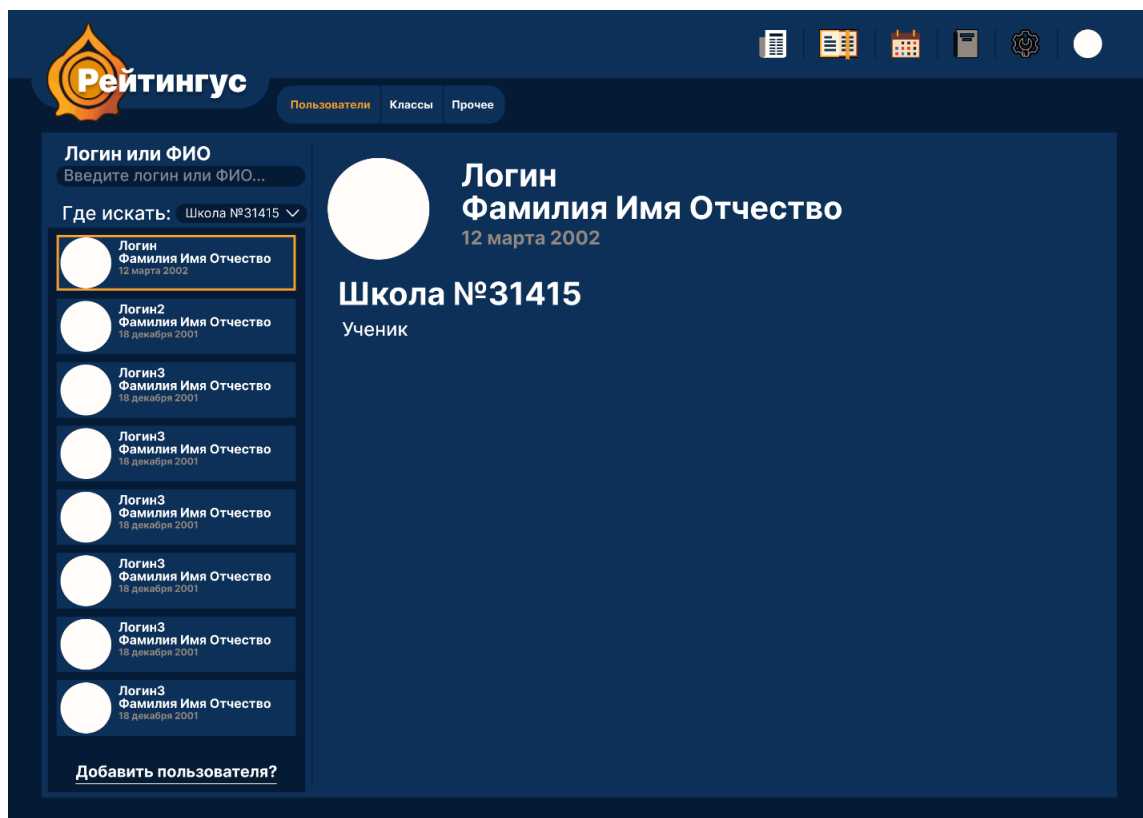


Рисунок 32 - Экран просмотра пользователей виртуальной учебной организации на сайте

Для добавления пользователей нужно заполнить поля фамилия, имя, отчество, а также через выпадающие списки указать роль добавляемого пользователя (и класс, если выбранная роль – «Ученик»).

Ниже форм находится кнопка «Добавить».

Ниже кнопки находятся сгенерированные уникальные одноразовые коды.

The screenshot shows the 'Рейтингус' website interface. At the top, there is a logo and navigation tabs: 'Пользователи', 'Классы', and 'Прочее'. The main section is titled 'Логин или ФИО' with a text input field 'Введите логин или ФИО...'. Below this is a dropdown menu 'Где искать:' and a list of existing users, each with a profile picture, login, and full name. A 'Добавить пользователя?' link is at the bottom of the list.

To the right, the 'ФИО пользователя:' section contains three input fields for 'Фамилия', 'Имя', and 'Отчество'. Below these is a 'Роль:' dropdown menu currently set to 'Ученик', and a 'Класс:' dropdown menu set to '9a'. A 'Добавить' button is located below the role and class selection. Further down, the 'Коды приглашения:' section displays three codes: 'Фамилия Имя Отчество, Класс 9a - favof8129fhaxf', 'Фамилия Имя Отчество, Класс 9a - fasfqw@FSaxz', and 'Фамилия Имя Отчество, Класс 11 - PXFOMXzfq29'.

Рисунок 33 - Пример процесса добавления пользователя с ролью «Ученик» на сайте

This screenshot shows the same 'Рейтингус' website interface as Figure 33, but with the 'Роль:' dropdown menu set to 'Учитель'. The 'Класс:' dropdown menu is not visible in this view. The 'Коды приглашения:' section now displays three different codes: 'Фамилия Имя Отчество - fosapo2FPoasf0f', 'Фамилия Имя Отчество - czxmqw20fas9fd', and 'Фамилия Имя Отчество - FPxozfwq1490sf'. The rest of the interface, including the user list and navigation tabs, remains the same.

Рисунок 34 - Пример процесса добавления пользователя с ролью «Учитель» на сайте

### 5.6.12 Журнал

Вверху раздела есть переключатель, где можно выбрать «Ученики» или «Занятия».

Ниже отображаются кнопка «Редактировать» и два выпадающих списка для выбора класса и предмета.

Кнопка «Редактировать» активно только в том случае, если учитель ведёт данный предмет у данного класса.

При выбранном режиме «Ученики» ниже находится таблица всех учеников, где можно ставить оценки или отметить посещаемость.

При выбранном режиме «Занятия» ниже находится таблица с датами, темами и домашними заданиями. Через неё можно выдавать домашнее задание ученикам.

Рисунок 35 - Экран раздела «Журнал» с выбранной вкладкой «Ученики» на сайте



## **6 Порядок контроля и приемки работ**

Контроль разработки системы осуществляется посредством выполнения тестов написанного программного кода и запланированных встреч между исполнителями данного проекта и заказчиком. Готовая система с полной документацией будет представлена заказчику в запланированную согласно данному документу дату. Заказчик определит соответствие системы его требованиям и осуществит её приём. Вся документация должна быть подготовлена и передана, как в печатном, так и в электронном виде (в форматах docx и pdf), а также размещена на GitHub.

## **7 Реквизиты и подписи сторон**

Заказчик:

\_\_\_\_\_ (Тарасов В.С.)

Исполнители:

\_\_\_\_\_ (Шлыков Д.Г.)

\_\_\_\_\_ (Сапегин П.А.)

\_\_\_\_\_ (Саков Е.А.)

## Приложение А

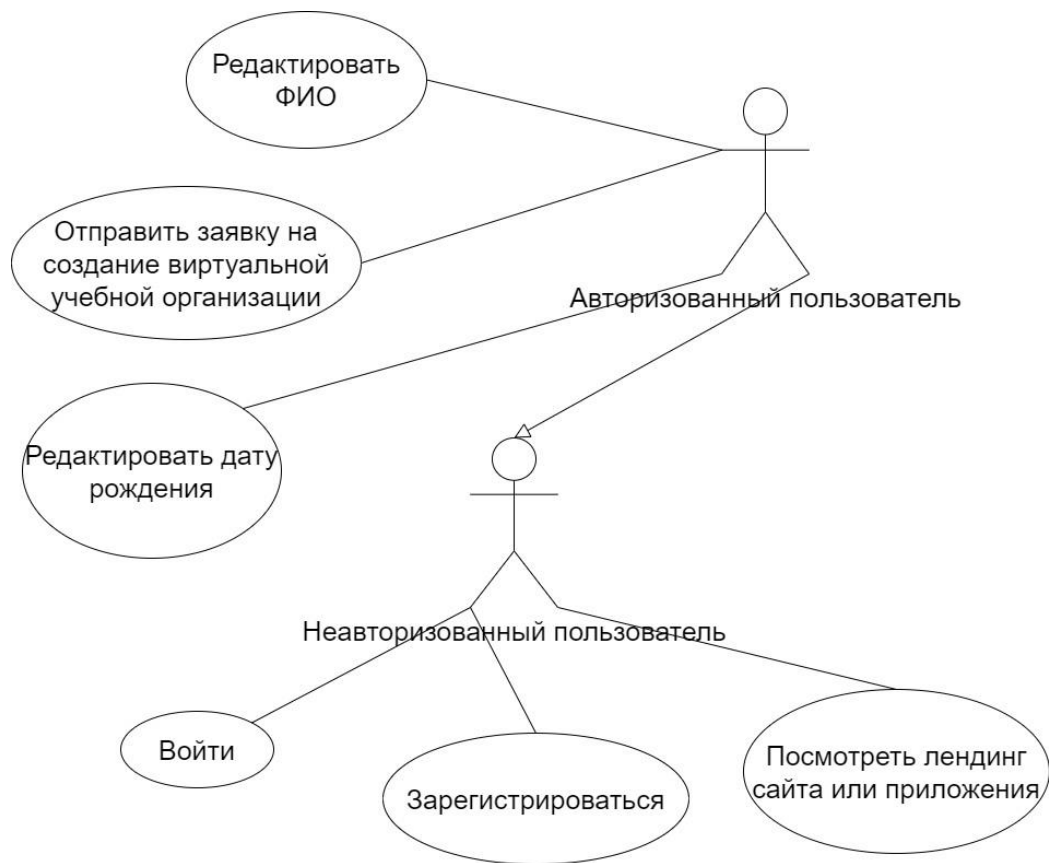


Рисунок А.1 - Диаграмма вариантов использования для неавторизованного и авторизованного пользователей





Рисунок А.2 - Диаграмма вариантов использования для ученика и учителя

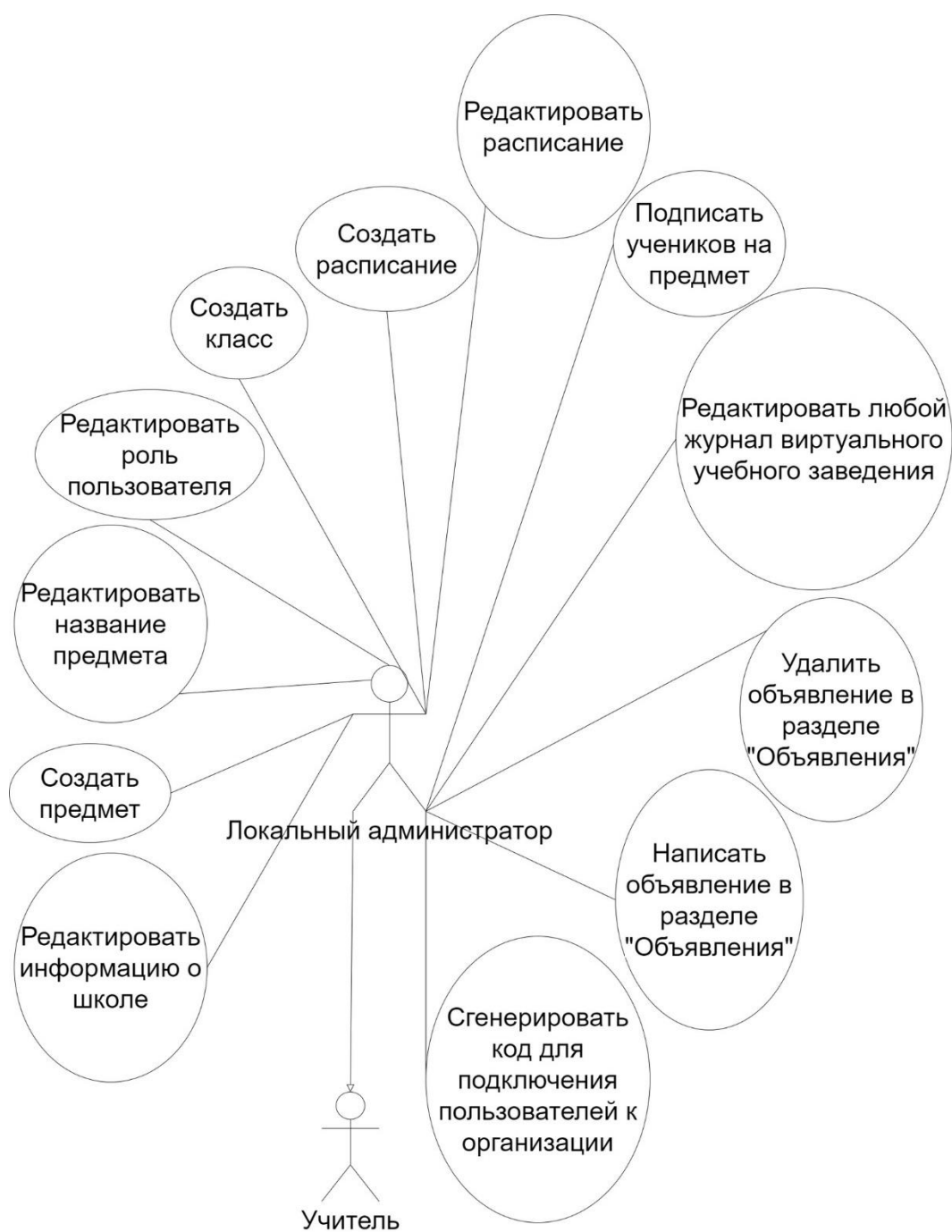


Рисунок А.3 - Диаграмма вариантов использования для локального администратора

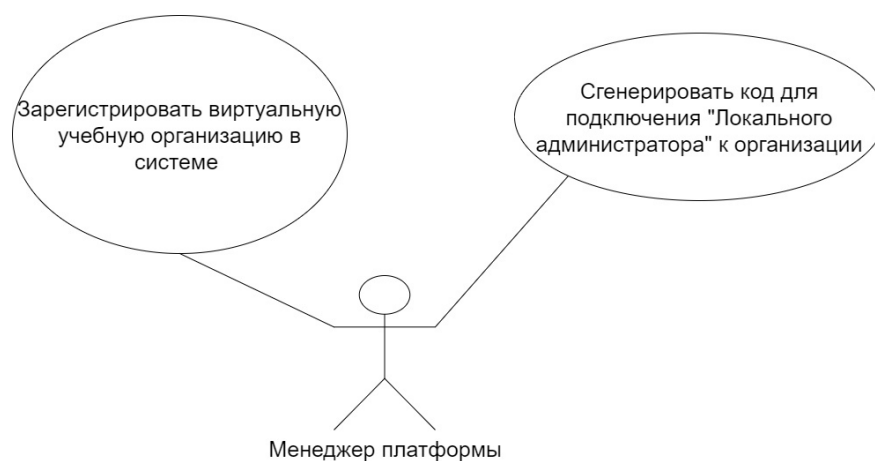


Рисунок А.4 - Диаграмма вариантов использования для менеджера платформы

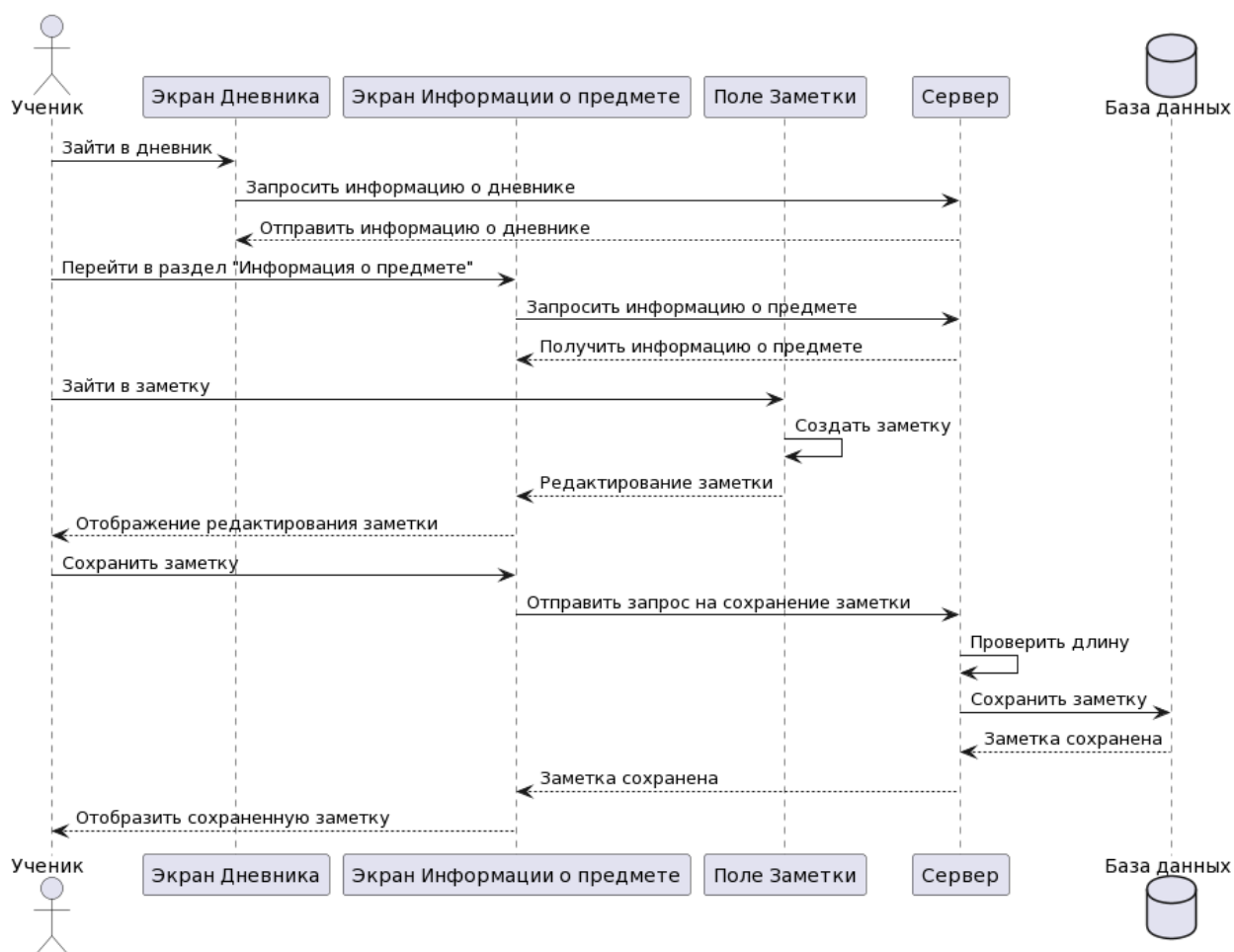


Рисунок А.5 - Пример диаграммы последовательности процесса создания и редактирования заметки учеником

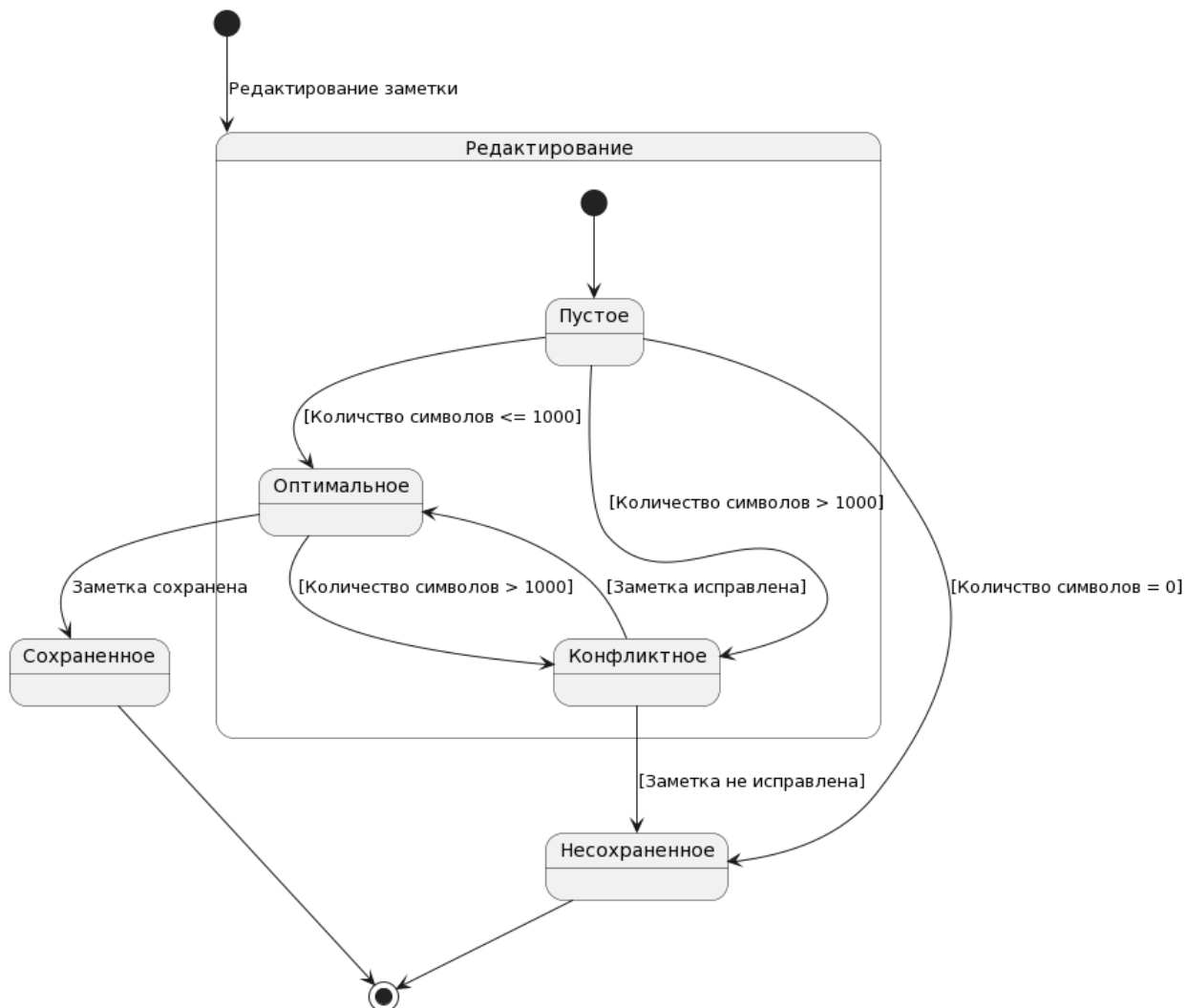


Рисунок А.6 - Пример диаграммы состояний заметки

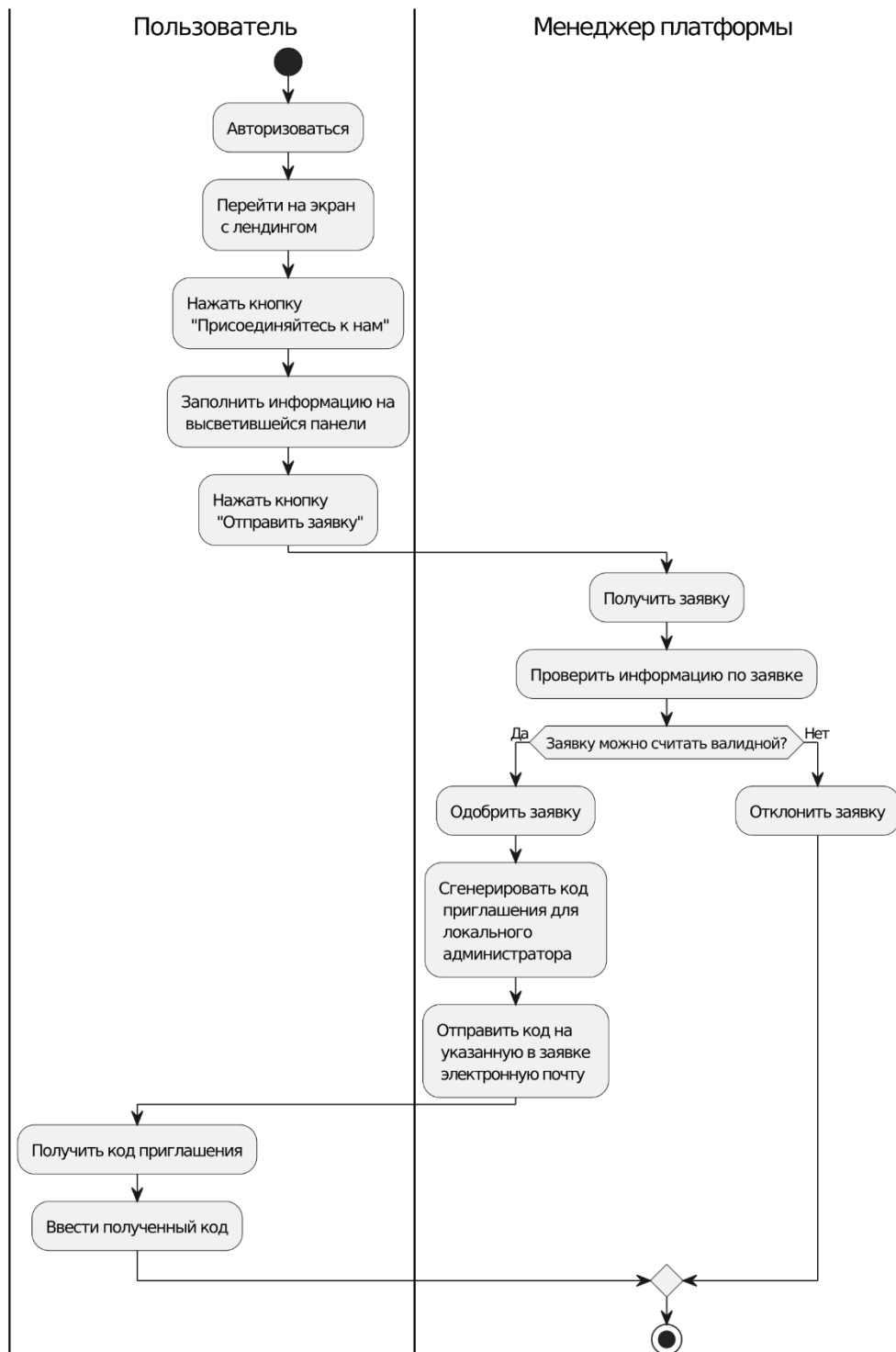


Рисунок А.7 - Диаграмма активности для отображения процесса создания виртуальной учебной организации и получения в ней пользователем роли локального администратора