

Школа: Инженерная школа информационных технологий и робототехники
 Направление подготовки: 09.03.04 «Программная инженерия»
 Отделение школы (НОЦ): Отделение информационных технологий

БАКАЛАВРСКАЯ РАБОТА

Тема работы Разработка системы администрирования веб-платформы для игр и приложений
--

УДК: 004.422.8:629.73-519

Студент

Группа	ФИО	Подпись	Дата
8К61	Венгер Артем Александрович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Саврасов Фёдор Витальевич	К.Т.Н.		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Спицына Любовь Юрьевна	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Белоенко Елена Владимировна	К.Т.Н.		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Чердынцев Евгений Сергеевич	К.Т.Н.		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения (выпускник должен быть готов)
P1	Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности.
P2	Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.
P3	Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей.
P4	Разрабатывать программные и аппаратные средства (системы, устройства, блоки, программы, базы данных и т. п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования.
P5	Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретация полученных данных, в области создания аппаратных и программных средств информационных и автоматизированных систем.
P6	Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.
P7	Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.
P8	Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.
P9	Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.
P10	Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.
P11	Демонстрировать способность к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии.

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа: Инженерная школа информационных технологий и робототехники
 Направление подготовки (специальность): 09.03.04 «Программная инженерия»
 Отделение школы (НОЦ): Отделение информационных технологий

УТВЕРЖДАЮ:

Руководитель ООП

_____ Чердынцев Е.С.
 (Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ на выполнение выпускной квалификационной работы

В форме:

Бакалаврской работы
(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8К61	Венгер Артему Александровичу

Тема работы:

Разработка системы администрирования веб-платформы для игр и приложений	
Утверждена приказом директора (дата, номер)	№153-51/с от 01.06.2020 г.

Срок сдачи студентом выполненной работы:	12.06.2020 г.
--	---------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Объектом проектирования в данной работе является система администрирования веб-платформы для игр и приложений, которая будет использоваться для управления играми, приложениями новостями и аккаунтами пользователей.</p>
<p>Перечень подлежащих исследованию, проектированию и разработке вопросов <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> 1. Проектирование системы администрирования 2. Выбор средств разработки 3. Разработка системы администрирования 4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение 5. Социальная ответственность

Перечень графического материала <i>(с точным указанием обязательных чертежей)</i>	<ol style="list-style-type: none"> 1. Диаграмма вариантов использования в нотации UML 2. Модель базы данных 3. Диаграммы последовательности 4. Рисунки, демонстрирующие результаты 5. Диаграмма Ганта
---	--

Консультанты по разделам выпускной квалификационной работы
(с указанием разделов)

Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Спицына Любовь Юрьевна
Социальная ответственность	Белоенко Елена Владимировна

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	27.02.2020 г.
---	---------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Саврасов Федор Витальевич	к.т.н		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К61	Венгер Артем Александрович		

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа: Инженерная школа информационных технологий и робототехники
 Направление подготовки (специальность): 09.03.04 «Программная инженерия»
 Уровень образования: Бакалавр
 Отделение школы (НОЦ): Отделение информационных технологий
 Период выполнения: осенний / весенний семестр 2019 / 2020 учебного года

Форма представления работы:

Бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	12.06.2020 г.
--	---------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
6.05.2020	Предметная область и бизнес-требования	15
13.05.2020	Проектирование системы	25
28.06.2020	Реализация системы	25
1.06.2020	Результат практического исследования	15
3.05.2020	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	10
5.05.2020	Социальная ответственность	10

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Саврасов Фёдор Витальевич	К.Т.Н.		

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Чердынцев Евгений Сергеевич	К.Т.Н.		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8К61	Венгер Артему Александровичу

Школа	ИШИТР	Отделение школы (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.04 Программная инженерия

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

<i>1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Материальные затраты – 462 руб. Затраты на специальное оборудование – 72797 руб. Затраты на заработную плату – 37561,2 руб. Затраты на отчисления во внебюджетные фонды – 10517,13 руб.
<i>2. Нормы и нормативы расходования ресурсов</i>	Бюджет проекта не более 150000 руб., в том числе затраты на оплату труда не более 50000 руб. Значение показателя интегральной ресурсоэффективности - не менее 2.5 баллов из 5.
<i>3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Районный коэффициент – 1,3 Коэффициент дополнительной заработной платы – 0,13 Коэффициент отчислений во внебюджетные фонды – 0,28 Коэффициент накладных расходов – 0,16

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<i>1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	1. Описание потребителей продукта 2. QuaD-анализ 3. SWOT-анализ
<i>2. Планирование и формирование бюджета научных исследований</i>	1. Описание структуры работ в рамках научного исследования. 2. Определение трудоемкости выполнения работ и разработка графика проведения научного исследования. 3. Подсчет бюджета проекта
<i>3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	Оценка сравнительной эффективности исследования

Перечень графического материала (с точным указанием обязательных чертежей):

<ol style="list-style-type: none"> 1. Матрица SWOT 2. График проведения и бюджет НИ 3. Оценка ресурсной, финансовой и экономической эффективности НИ 	
---	--

Дата выдачи задания для раздела по линейному графику	27.04.2020 г.
---	---------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Спицына Любовь Юрьевна	к.э.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К61	Венгер Артем Александрович		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА

«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8К61	Венгер Артему Александровичу

Школа	ИШИТР	Отделение (НОЦ)	ОИТ
Уровень образования	Бакалавриат	Направление/специальность	09.03.04 Программная инженерия

Тема ВКР:

Разработка системы администрирования веб-платформы для игр и приложений	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	Объект исследования – система администрирования веб-платформы для игр и приложений. Область применения – веб-платформа для игр и приложений. Рабочее место – рабочий стол с персональным компьютером в аудитории Кибернетического центра ТПУ.
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
1. Правовые и организационные вопросы обеспечения безопасности: – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны.	– Рабочее место при выполнении работ сидя регулируется ГОСТом 12.2.032-78 – Организация рабочих мест с электронно-вычислительными машинами регулируется СанПиНом 2.2.2/2.4.1340-03
2. Производственная безопасность: 2.1. Анализ выявленных вредных и опасных факторов 2.2. Обоснование мероприятий по снижению воздействия	– Отклонение показателей микроклимата; – Недостаточная освещенность рабочей зоны; – Повышенный уровень шума на рабочем месте;
3. Экологическая безопасность:	Анализ воздействия на литосферу: – Утилизация компьютеров, смартфонов, оргтехники и бумаги;
4. Безопасность в чрезвычайных ситуациях:	– Пожар;

Дата выдачи задания для раздела по линейному графику	27.04.2020 г.
--	---------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Белоенко Е.В.	к.т.н		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8К61	Венгер Артем Александрович		

Реферат

Выпускная квалификационная работа содержит 99 страниц, 32 рисунка, 21 таблица, 6 приложений и 11 литературных источников.

Ключевые слова: система администрирования, веб-платформа для игр и приложений.

Цель работы: создание многофункциональной системы администрирования веб-платформы для игр и приложений

В процессе выполнения работы использовалась среда выполнения NodeJS и СУБД MySQL.

В результате работы была реализована система администрирования веб-платформы для игры и приложений.

В первом разделе представлено описание предметной области, веб-платформы, и общее описание функций разрабатываемой системы.

Во втором разделе приводится подробное описание функций разрабатываемой системы и выбор средств разработки с обоснованием.

Третий раздел содержит информацию о способе реализации системы администрирования с приведением различных диаграмм, для более наглядного представления.

В четвертом разделе описывается разработанная система и иллюстрируются ее функции.

Пятый раздел представляет собой выполненное задание по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение».

В шестом разделе содержится выполненное задание по разделу «Социальная ответственность».

Определения, обозначения, сокращения, нормативные ссылки

SaaS – это модель продажи программного обеспечения, при которой поставщик разрабатывает веб-приложение и самостоятельно управляет им, предоставляя заказчикам доступ через интернет.

Стриминг – это потоковое онлайн-вещание.

Виртуальная машина – это полностью изолированная программная система, которая исполняя машинно-независимый или машинный код процессора, способна эмулировать операционную систему, приложения или устройства (например, компьютер).

Контент – это информационное содержание сайта (тексты, графическая, звуковая информация и др.).

БД – это хранилище для большого количества систематизированных данных, с которыми можно производить определённые действия.

WYSIWYG – это свойство прикладных программ, в которых содержание отображается в процессе его редактирования и выглядит максимально близко к конечному результату.

Ник – псевдоним, используемый пользователем в Интернете, обычно в местах общения.

Среда выполнения – вычислительное окружение, необходимое для выполнения компьютерной программы и доступное во время выполнения компьютерной программы.

Front-end – клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса.

Back-end – программно-аппаратная часть сервиса.

SQL – (Structured Query Language) – язык, используемый для работы с базами данных.

СУБД – специализированная программа, предназначенная для организации и ведения базы данных.

ORM – программное решение, позволяющее работать с сущностями базы данных как с объектами.

Оглавление

Реферат	10
Определения, обозначения, сокращения, нормативные ссылки.....	11
Введение.....	16
1 Предметная область и бизнес-требования	17
1.1 Описание разрабатываемой веб-платформы.....	17
1.2 Постановка задачи	20
1.3 Общее описание функций системы администрирования	20
2 Проектирование системы.....	23
2.1 Проектирование функций системы.....	23
2.1.1 Функции разработчика	23
2.1.2 Функции модератора	26
2.1.3 Функции администратора	29
2.2 Выбор средств разработки	32
2.2.1 Node.js	32
2.2.2 MySQL.....	33
3 Реализация системы.....	33
3.1 Архитектурный шаблон MVC	33
3.1.1 Слой представлений	34
3.1.2 Слой данных	36
3.1.3 Управляющий слой.....	39
3.2 Реализация БД	40
4 Результат практического исследования.....	41
4.1.1 Изменение вида страниц в зависимости от роли.....	41
4.1.2 Работа с заявками в системе администрирования.....	42

4.1.3	Редактирование игры или приложения	47
4.1.4	Просмотр списка всех пользователей	49
4.1.5	Просмотр и редактирование информации о пользователе	49
4.1.6	Создание и редактирование новости	50
4.1.7	Управление ролями.....	53
5	Финансовый менеджмент, ресурсоэффективность	55
5.1	Оценка коммерческого потенциала и	55
5.1.1	Потенциальные потребители результатов исследования	55
5.1.2	Анализ конкурентных технических решений.....	56
5.1.3	QuaD-анализ	57
5.1.4	SWOT-анализ.....	59
5.2	Определение возможных альтернатив.....	62
5.3	Планирование научно-исследовательских работ	63
5.3.1	Структура работ в рамках научного исследования	63
5.3.2	Определение трудоемкости выполнения работ	64
5.4	Бюджет научно-технического исследования (НТИ)	65
5.4.1	Расчет материальных затрат НТИ.....	65
5.4.2	Расчет амортизационных затрат.....	65
5.4.3	Основная заработная плата исполнителей темы	67
5.4.4	Дополнительная заработная плата исполнителей темы	69
5.4.5	Отчисления во внебюджетные фонды.....	69
5.4.6	Накладные расходы	70
5.4.7	Формирование бюджета затрат	71
5.5	Определение ресурсной (ресурсосберегающей),.....	72
	Заключение	74

6 Социальная ответственность	75
Введение.....	75
6.1 Правовые и организационные вопросы.....	75
6.2 Производственная безопасность	77
6.2.1 Отклонение показателей микроклимата.....	78
6.2.2 Недостаточная освещенность рабочей зоны.....	80
6.2.3 Повышенный уровень шума на рабочем месте	81
6.3 Экологическая безопасность	83
6.4 Безопасность в чрезвычайных ситуациях	84
Вывод по разделу	84
Заключение	85
Список использованных источников	86
Приложение А	88
Приложение Б.....	89
Приложение В	90
Приложение Г	93
Приложение Д	96
Приложение Е.....	98

Введение

В наше время происходит непрерывная разработка новых программных продуктов различного уровня сложности и функциональности. При этом, для комфортной и стабильной работы основной функциональности может быть недостаточно, особенно при росте сложности проекта. Для удобного управления приложениями необходима разработка специальных систем администрирования, функционал которых может быть недоступен обычным пользователям.

Системы администрирования крайне важны, так как они позволяют привлекать к управлению программными продуктами даже тех людей, которые не занимаются программированием профессионально. Соответственно, их интерфейс должен быть понятным даже не IT-специалистам. Также эти системы должны предоставлять широкий набор функций для решения проблем, возникающих при функционировании программного продукта и взаимодействии с ним пользователей.

Целью данной работы является создание многофункциональной системы администрирования веб-платформы для игр и приложений, которая должна позволять управлять расположенными на веб-платформе программными продуктами, новостными и рекламными оповещениями, и аккаунтами пользователей.

Исходя из поставленной цели, были выдвинуты следующие задачи:

1. Анализ предметной области и выявление основных требований к системе.
2. Проектирование функций системы.
3. Выбор средств разработки.
4. Реализация системы.

1 Предметная область и бизнес-требования

1.1 Описание разрабатываемой веб-платформы.

В настоящее время существует множество приложений, которые, в большинстве случаев, необходимо устанавливать на персональный компьютер пользователя. Однако это не всегда бывает удобно, так как установка может требовать много времени, доступ к программному обеспечению имеется только с компьютера, где оно установлено, а постоянные обновления могут мешать работе. Запуск приложений в браузере позволит решить описанные выше проблемы.

Решение пользоваться программным обеспечением через браузер сейчас набирает всё большую популярность, как можно увидеть по объему мирового рынка SaaS-решений, прибыль которого последние 10 лет ежегодно увеличивается в среднем на 39 % [1]. Игры в браузере также являются очень популярными, в веб-версии социальной сети «ВКонтакте» в 2017 году их запускало 55 миллионов человек [2].

В связи с растущей популярностью представленных выше направлений, группой студентов ТПУ было принято решение о разработке веб-платформы для игр и приложений. При первоначальном анализе, были рассмотрены различные конкурентные продукты на российском рынке, позволяющие запускать игры и приложения.

В первую очередь, таковым является классическое клиент-серверное приложение «World Of Tanks», в котором игровой клиент устанавливается на персональный компьютер пользователя. Главный плюс такого подхода – небольшое время отклика, что делает игру плавной и комфортной. Минус заключается в объеме устанавливаемого приложения, для «World of Tanks» его значение равно примерно 45-70 Гб, в зависимости от качества текстур [3].

Второй конкурент – это SaaS решения, наподобие Google-документов. Однако следует учитывать, что разрабатываемая веб-платформа будет не SaaS-решением, а именно платформой, на которую начинающие разработчики и компании, желающие попробовать себя в новой сфере, смогут выкладывать свои продукты с целью тестирования их широкой аудиторией и получения прибыли.

Также к числу облачных сервисов относятся стриминговые сервисы, которые запускают приложение на виртуальной машине, а затем отправляют видеопоток с нее пользователю. Примером такой системы является «PlayKey». Но, в отличие от нее и ей подобных, клиентская часть будущей веб-платформы, находящаяся в браузере, будет самостоятельно рендерить изображение приложений.

При определении состава примерного функционала разрабатываемой платформы был подробно изучен сервис приложений, представленный в социальной сети «ВКонтакте». При его запуске открывается стартовая страница с информацией о достижениях друзей, популярных играх, и играх пользователя (рисунок 1). С главной страницы можно перейти на другие, с целью поиска интересующего продукта. Когда пользователь заинтересуется какой-либо игрой, он запускает ее. Пример запущенной игры можно увидеть на рисунке 2.

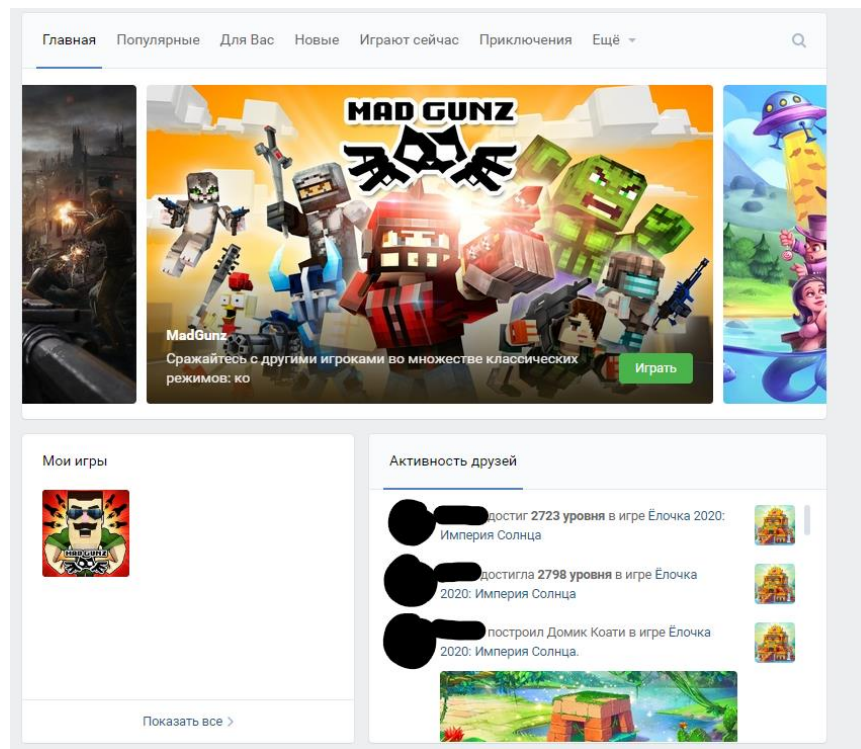


Рисунок 1 – Стартовая страница игрового сервиса социальной сети «ВКонтакте»

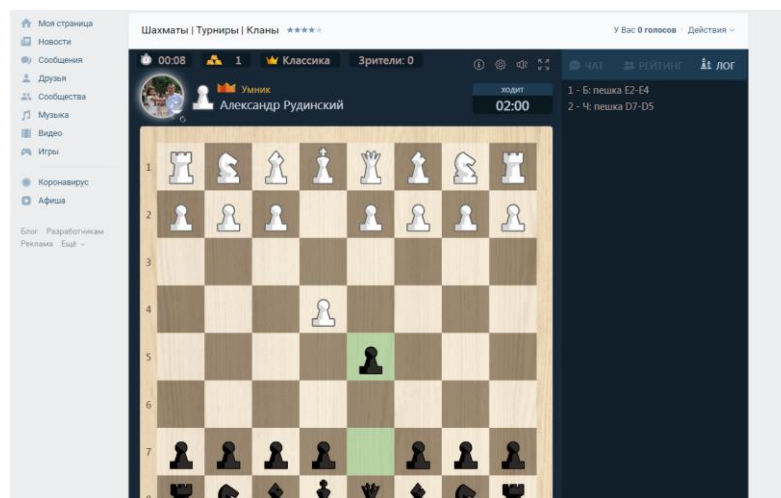


Рисунок 2 – Пример игры «Шахматы | Турниры | Клань» в игровом сервисе социальной сети «ВКонтакте»

Также в социальной сети «ВКонтакте» имеется возможность запускать прикладные приложения. Одно из них изображено на рисунке 3.

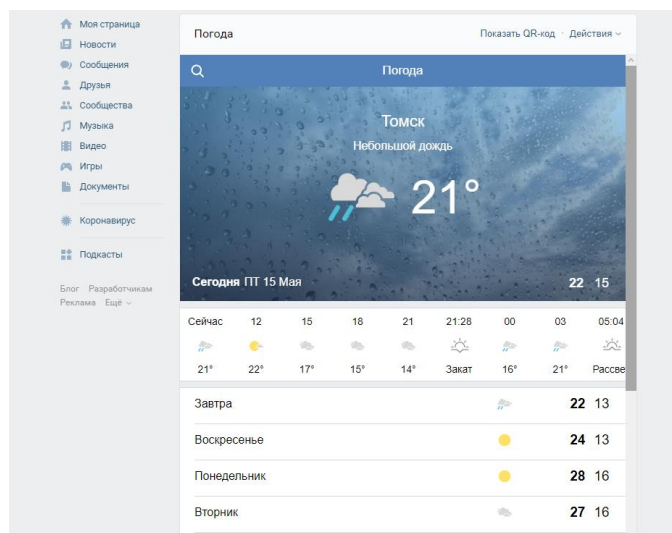


Рисунок 3 – Приложение «Погода»

1.2 Постановка задачи

Разрабатываемая веб-платформа является многокомпонентной и сложной системой, предполагающей наличие большого числа игр, прикладных приложений и пользователей. Чтобы она была работоспособной, ей необходима качественная и многофункциональная система администрирования, позволяющая управлять играми и приложениями, правами пользователей, новостями сообщества, осуществлять связь между администраторами веб-платформы и разработчиками приложений.

Данная работа будет направлена на решение обозначенной выше задачи путем создания системы администрирования веб-платформы для игр и приложений.

1.3 Общее описание функций системы администрирования

В первую очередь, необходимо реализовать разделение пользователей на роли, чтобы то, что могут выполнять администраторы, не могли выполнять обычные пользователи. В результате проектирования были определены следующие роли: неавторизованный пользователь, авторизованный пользователь, забаненный пользователь, разработчик, модератор,

администратор. Все эти роли и их возможные действия изображены на диаграмме вариантов использования, представленной в приложении А.

В данной работе будет рассматриваться функционал только для трех ролей: разработчик, модератор, администратор.

Разработчик управляет своими играми. Он будет иметь возможность подать заявку на добавление на платформу нового приложения, редактировать уже существующее приложение, или удалить его с веб-платформы. Также на веб-платформе будет предусмотрена реклама приложений, и разработчик сможет отплатить ее для продвижения своего продукта. Реклама будет располагаться в двух местах, и в одном случае будет выглядеть как небольшое прямоугольное изображение, а в другом – изображение, вытянутое по горизонтали.

Модератор является помощником администратора, и помогает ему управлять работой веб-платформы. Одной из его обязанностей является тестирование приложений, которые разработчики хотят добавить на веб-платформу, и описание результатов проведенных тестов. Данное тестирование нужно для проверки работоспособности добавляемого программного обеспечения, и его проверки на отсутствие ненормативного контента.

Также модератор ответственен за наполнение новостной ленты веб-платформы контентом. Он может создавать новости и редактировать их при необходимости. Редактирование страницы «О нас» тоже входит в обязанности модератора.

Самой главной ролью, обладающей самыми большими правами, является администратор. Ему доступны все действия, которые может совершать модератор, и еще дополнительные, доступные только для него. Администратор принимает окончательное решение одобрить или отклонить заявку на публикацию новой игры или нового приложения. Он сможет редактировать информацию о любой игре и любом приложении, и даже удалить их. Ему будет

доступна возможность добавить права на совершение какого-либо действия к роли или забрать их у нее. Администратор в случае необходимости сможет редактировать информацию о любом пользователе или удалить его.

Описанных выше ролей и их функций будет достаточно для обеспечения эффективного управления разрабатываемой веб-платформой.

2 Проектирование системы

2.1 Проектирование функций системы

В данном разделе будут более подробно описаны основные функции системы администрирования веб-платформы для игр и приложений.

2.1.1 Функции разработчика

2.1.1.1 Отправка заявки на публикацию приложения

Отправка заявки будет проходить через страницу «Кабинет разработчика». На этой странице разработчику необходимо будет выбрать пункт меню «Подать заявку на добавление игры или приложения», после чего заполнить следующие поля формы: название, тип приложения, описание, тип подключения, IP-адрес, порт, цена, пиктограмма, главное изображение, дополнительные изображения.

Под типом приложения понимается игра или программа, по IP-адресу и порту веб-платформа будет связываться с сервером игры или приложения разработчика, главное и дополнительные изображения будут отображаться на странице игры или приложения.

После заполнения всех полей формы и нажатия кнопки «Подать заявку» будет сформирована и сохранена новая заявка на публикацию игрового или прикладного приложения, ее смогут просмотреть как администратор, так и модератор.

2.1.1.2 Отправка заявки на публикацию рекламы

На страницу подачи заявки на публикацию рекламы можно будет перейти через страницу «Кабинет разработчика», нажав на пункт меню «Подать заявку на публикацию рекламы». Эта страница будет содержать форму с полями: игра

или приложение, тип подписки, малое рекламное изображение, большое рекламное изображение.

Поле «Игра или приложение» будет представлено выпадающим списком, содержащим названия всех продуктов разработчика, опубликованных на веб-платформе. Типы подписки устанавливаются владельцами сайта и отличаются друг от друга частотой появления рекламы, периодом отображения рекламы на веб-платформе, и ценой. Изображения предоставляются для отображения их в специальных рекламных местах, и имеют требования к размеру.

Для отправки заявки на рассмотрение, после заполнения всех полей, необходимо нажать на кнопку «Подать заявку».

2.1.1.3 Просмотр опубликованных приложений

Для просмотра опубликованных приложений разработчику будет необходимо зайти на страницу «Кабинет разработчика» и выбрать там пункт меню «Мои приложения». После этого система отправит запрос на сервер и загрузит все приложения разработчика. Это будет возможно благодаря связи между таблицами приложений и пользователя в БД.

2.1.1.4 Оправка заявки на удаление своего приложения

Удаление приложений с веб-платформы необходимо проводить очень аккуратно, так как пользователи могли уже заплатить за продукт, и, если разработчик по ошибке произведет удаление, это может вызвать много проблем. Удаление должно проходить по согласованию с администратором веб-платформы. Для удаления своего продукта с веб-платформы разработчик должен будет оставить заявку с указанием причины желая удалить его. Эту заявку сможет рассмотреть только администратор.

Заявку на удаление можно заполнить, нажав в кабинете разработчика на элемент меню «Подать заявку на удаление игры или приложения».

2.1.1.5 Просмотр списка отправленных заявок

У разработчика имеется возможность просмотра списка отправленных заявок. Для этого ему необходимо на странице «Кабинет разработчика» нажать на элемент меню «Список отправленных заявок», после чего откроется страница со всеми отправленными разработчиком заявками.

Каждая заявка будет состоять из следующих полей: название приложения, тип, статус, дата отправки, и ссылка на страницу заявки. В типе заявки будет указано «добавление», «удаление» или «реклама». Статус может быть одним из четырех: не рассмотрена, проверена, одобрена, отклонена.

На этой странице можно будет осуществить фильтрацию заявок по типу, статусу, и дате отправки. Фильтрация по дате будет производиться путем указания интервала дат, в пределах которого следует производить поиск.

2.1.1.6 Просмотр отправленной заявки

При нажатии на ссылку в ячейке «Страница заявки» в списке отправленных заявок, будет осуществлен переход на страницу заявки, где будет находиться вся информация (название, тип приложения, описание, тип подключения, IP-адрес, порт, цена, пиктограмма, главное изображение, дополнительные изображения, ник разработчика, тип заявки, статус заявки и комментарии модератора или администратора).

2.1.1.7 Редактирование информации об опубликованном приложении

Имеется два варианта реализации этой функции в системе администрирования. Под информацией о приложении в данном случае понимается массив данных, в котором находятся название, тип приложения, описание, цена, пиктограмма, главное изображение, дополнительные изображения.

Первый вариант предполагает предоставление разработчику большей свободы в управлении своим продуктом и наделение его правом самому, без согласования с модератором или администратором веб-платформы, изменять информацию об опубликованном приложении. Это снимет часть нагрузки с модераторов или администраторов и упростит разработчику возможности устанавливать акции на свой продукт, изменять оформление в связи с обновлением, и т.д.

Второй вариант (более формальный), предполагает согласование всех изменений с модератором или администратором веб-платформы. Это увеличит нагрузку на администрацию веб-платформы, замедлит процесс обновления информации, однако при этом повысит контроль над приложениями на веб-платформе.

Для описываемой в данной работе системы администрирования был выбран первый вариант, предоставляющий большую свободу разработчику, что позволит придать ему стимул совершенствовать свой продукт как можно чаще, без ожидания принятия каждого изменения продолжительное время.

Для редактирования информации об игре или приложении разработчику необходимо будет зайти на страницу своего приложения и нажать на соответствующую кнопку. После этого осуществляется переход на форму редактирования, имеющую схожий внешний вид с той, которая отображалась при создании заявки на публикацию, только с уже заполненными полями, и без возможности изменения типа подключения, IP-адреса и порта. После окончания редактирования, необходимо будет нажать кнопку «Сохранить изменения».

2.1.2 Функции модератора

2.1.2.1 Просмотр списка заявок на публикацию игры, приложения или рекламы

После того как разработчик отправит заявку на публикацию игры, приложения или рекламы, она станет доступна для просмотра модераторам и администраторам веб-платформы. Просмотреть все заявки можно будет на странице «Администрирование», выбрав элемент меню «Заявки от разработчиков». После этого откроется список всех заявок.

Будет предусмотрена возможность просматривать выборку заявок по следующим фильтрам: «Все», «Не рассмотренные», «Проверенные», «Одобренные», «Отклонённые», «Закрытые», «Добавление», «Реклама». Фильтр «Все» должен производить отбор всех заявок. Фильтр «Не рассмотренные» должен производить отбор всех заявок, с которыми ни модератор, ни администратор еще не взаимодействовали (не проверяли, не одобряли и не отклоняли). Фильтр «Проверенные» должен производить отбор только тех заявок, которые были проверены на правильность заполнения и отсутствие ненормативного контента. Фильтры «Одобренные» и «Отклоненные» должны производить отбор только тех заявок, которые были соответственно одобрены или отклонены администратором веб-платформы. Фильтр «Закрытые» должен производить отбор всех одобренных и отклоненных заявок. Фильтр «Добавление» должен отобразить заявки на добавление игры или приложения. Фильтр «Реклама» должен показать все заявки на рекламу. Также будет возможность указания интервала дат для выбора.

2.1.2.2 Проверка заявки на публикацию игры, приложения или рекламы

При выборе одной из заявок, модератор переходит на ее страницу. Если это заявка на публикацию игры или приложения, то на этой странице будет указан IP-адрес и порт, перейдя по которому модератор тестирует игру или приложение. Если игра или приложение работает корректно, не содержит ненормативного контента, и все поля в заявке оформлены правильно, то модератор указывает, что проверка пройдена, и пишет комментарий к проверке.

Если заявка не прошла проверку, модератор указывает, что проверка не пройдена, и пишет комментарий, в котором описывает причины провала проверки.

После того, как заявка проверена, администратор может вынести решение, одобрить или отклонить заявку.

2.1.2.3 Создание и редактирование новости

Для создания новости модератору будет необходимо на странице «Администрирование» выбрать пункт меню «Создать новость» и заполнить форму новости.

Форма новости будет состоять из поля «Заголовок» и поля «Содержание», представленного WYSIWYG-редактором.

После заполнения всех полей и нажатия на кнопку «Опубликовать», новость будет отображена на странице новостей.

Для редактирования новости модератору необходимо будет на странице «Администрирование» выбрать пункт меню «Редактировать новость». После этого будет отображен список новостей, состоящий из их заголовков, и упорядоченный по дате. При необходимости можно будет указать интервал дат, и выбрать только те новости, которые были созданы в этот интервал дат.

Из появившегося списка можно будет выбрать нужную новость, кликнув по ее заголовку, после чего откроется форма, как при создании новости, с уже заполненными полями. Модератору останется только внести исправления в нужных местах и нажать на кнопку «Обновить новость».

2.1.2.4 Просмотр списка всех пользователей

Для просмотра списка всех пользователей модератору необходимо на странице «Администрирование» выбрать пункт меню «Список всех пользователей». После этого отобразится список всех пользователей,

упорядоченный по алфавиту. В этом списке можно будет увидеть ник пользователя и его роль. Для просмотра профиля любого пользователя из списка необходимо будет кликнуть на его ник.

2.1.2.5 Редактирование страницы «О нас»

Для редактирования страницы «О нас» модератору необходимо на странице «Администрирование» выбрать пункт меню «Редактирование страницы «О нас»». После этого будет открыта форма с полем «Содержание страницы», в котором будет содержаться текст текущей страницы «О нас». Это поле будет представлено WYSIWYG-редактором. Модератору будет необходимо редактировать текст, представленный в поле «Содержание страницы», после чего нажать кнопку «Обновить содержимое страницы».

2.1.3 Функции администратора

2.1.3.1 Одобрение или отклонение заявки на публикацию игры, приложения или рекламы

Просмотреть все заявки можно будет на странице «Администрирование», выбрав элемент меню «Заявки от разработчиков». После этого откроется список всех заявок, описанный в пункте 2.1.2.1.

Администратор, как и модератор, может просматривать заявки на публикацию игры, приложения или рекламы. Но, в отличие от модератора, только он сможет принять окончательное решение, одобрить заявку или отклонить.

Для этого он просматривает результат проверки модератора, и после этого либо еще раз сам производит проверку, либо сразу принимает окончательное решение. При желании, администратор может при принятии решения оставить свой комментарий к этой заявке. Для принятия решения будут две кнопки, «Одобрить» и «Отклонить». При отклонении, статус заявки

измениться на «Отклонена», при одобрении, статус заявки изменится на «Одобрена», и игра, приложение или реклама будет автоматически добавлена на веб-платформу.

2.1.3.2 Одобрение или отклонение заявки на удаление приложения

В отличие от модератора, администратор, помимо заявок на публикацию, может видеть заявки на удаление. Также он сможет выбрать, какие заявки отображать: на добавление, на удаление, рекламные, либо все.

При рассмотрении новой заявки, администратор должен будет ознакомиться с причиной желая удалить игру или приложение, связаться с разработчиком, уточнить, действительно ли он хочет это сделать, и, если это не ошибка, одобрить заявку с помощью специальной кнопки. После одобрения статус заявки изменится, и приложение автоматически скроется из каталога. Если удаляемый продукт был платным, то он удаляется только из общего каталога, и им могут воспользоваться те пользователи, которые приобрели его до удаления. Если удаляемый продукт был бесплатным, то он удалится с веб-платформы полностью. Если же заявка на удаление была отправлена по ошибке, то администратор отклонит ее с помощью специальной кнопки.

2.1.3.3 Редактирование информации о приложении

Редактирование информации о приложении для администратора по функционалу напоминает действия, описываемые в пункте 2.1.1.7, но с тем отличием, что администратор сможет редактировать информацию о любом продукте на веб-платформе. Для начала редактирования необходимо будет зайти на страницу приложения и нажать на кнопку «Редактировать». Также администратор сможет изменить абсолютно все поля при редактировании. После внесения изменений необходимо будет нажать на кнопку «Сохранить изменения».

2.1.3.4 Удаление приложения из каталога

Удаление любой игры или любого приложения из каталога не позволит каким-либо образом найти его и посетить его страницу обычным пользователям, но если пользователь хотя бы раз запускал его до удаления, то возможность запуска сохранится; модератор, администратор и разработчик продукта все еще смогут посетить страницу этого продукта.

Удаление из каталога происходит на странице приложения с помощью кнопки «Удалить из каталога».

2.1.3.5 Полное удаление приложения

Полное удаление приложения удаляет его из каталога для абсолютно всех пользователей, и запрещает его запуск.

Полное удаление будет происходить на странице продукта с помощью кнопки «Удалить полностью».

2.1.3.6 Удаление пользователя

Удаление любого пользователя — это функция не для постоянного использования. Применять ее возможно только в случае крайней необходимости. Она полностью удаляет пользователя с веб-платформы, вместе с записью о нем в БД.

Удалить пользователя возможно на странице его профиля, нажав на кнопку «Удалить пользователя».

2.1.3.7 Редактирование информации о пользователе

Редактировать информацию о любом пользователе возможно на странице его профиля, при нажатии на кнопку «Редактировать основную информацию». Администратор сможет редактировать все поля в профиле, также ему доступно поле «Роль», где он может изменить роль пользователя на «Пользователь»,

«Разработчик», «Модератор», «Администратор» или «Забаненный пользователь». В зависимости от указанной роли будут меняться и права доступа пользователя.

2.1.3.8 Добавление и удаление прав к ролям

На странице «Администрирование» при нажатии на элемент меню «Управление ролями», администратор перейдет на страницу управления ролями, где возможно будет выбрать роль и увидеть все ее права, а также добавить к роли новое право или удалить уже существующее.

2.2 Выбор средств разработки

2.2.1 Node.js

Для написания серверной части системы администрирования была выбрана среда выполнения JavaScript Node.js.

По данным опроса пользователей ресурса «StackOverflow» в 2018 году [4], Node.js стал самой популярной технологией в категории «Frameworks, Libraries, and Tools», обогнав .NET Core, Spring и Django . В 2019 году его перенесли в категорию «Other Frameworks, Libraries, and Tools», где он также занял первое место [5], что говорит о его популярности и признании в IT-сообществе. Такие ресурсы как Netflix, PayPal, eBay написаны на Node.js [6].

Одна из ключевых причин, по которой был выбран Node.js – возможность использовать для серверной части web-приложения JavaScript, т.к. ситуация, когда и front-end, и back-end часть пишется на одном и том же языке, является весьма удобной для разработки и поддержки. Для малых команд и разработчиков-одиночек это может значительно повысить скорость разработки продукта.

Еще одним основанием для выбора Node.js послужило наличие неблокирующего ввода/вывода, благодаря которому программа не

приостанавливает свою работу во время ожидания ответа от СУБД или записи данных в файл. Предполагается, что это повысит скорость функционирования системы администрирования, которая должна будет часто взаимодействовать с БД.

2.2.2 MySQL

При выборе СУБД для системы администрирования рассматривались три варианта: MySQL, PostgreSQL, SQL Server 2019 Express. Все представленные продукты являются бесплатными, а MySQL и PostgreSQL имеют открытый исходный код.

PostgreSQL полностью поддерживает все новые возможности SQL и является наиболее профессиональной из все перечисленных вариантов, но это делает ее тяжеловесной и медленной, при выполнении простых запросов. Так как разрабатываемая система администрирования в основном будет использовать простые запросы, был сделан вывод, что данная СУБД не подходит для нее.

MySQL и SQL Server 2019 Express являются достаточно безопасными, а их отличия не являются значительными для разрабатываемого проекта. Определяющим фактором стала поддержка и активность сообщества MySQL, которое выпустило огромное количество статей и накопило множество решений проблем, возникающих при работе с MySQL.

3 Реализация системы

3.1 Архитектурный шаблон MVC

Для реализации системы администрирования был выбран архитектурный шаблон MVC (Model-View-Controller). Основное его преимущество заключается в разделении структуры приложения на три компонента: модель, представление и контроллер. Модель содержит данные и методы взаимодействия с ними. Представление организует оформление и вывод полученных из модели данных

в удобном для анализа виде, а также распознавание действий пользователя и передачу информации о них контроллеру. Контроллер является управляющим слоем, позволяющим связать уровни данных и представления. В итоге, при использовании такой архитектуры, то, как данные представлены в системе, напрямую не зависит от того, как их видит пользователь, что дает большую гибкость в разработке и возможность несколько раз использовать одни и те же участки кода.

3.1.1 Слой представлений

Для разработки слоя представлений помимо стандартных HTML и CSS был использован шаблонизатор Pug, позволяющий составлять веб-страницы из различных блоков и вписывать в HTML-разметку JavaScript код. Структура папок этого слоя представлена на рисунке 4.

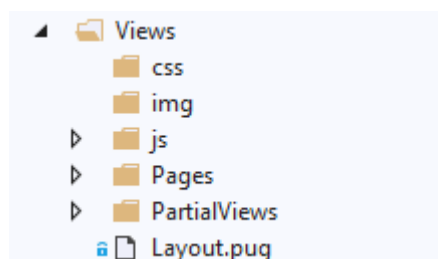


Рисунок 4 – Структура папок слоя представлений

Далее приводится описание содержимого каждой папки:

- `css` – таблицы стилей;
- `img` – изображения, используемые на страницах;
- `js` – скрипты, клиентской части;
- `Pages` – pug-файлы с содержимым страницы;
- `PartialViews` – Pug-файлы, содержимое которых встраивается в файлы из папки `Pages`;

Во Views, помимо папок, еще содержится файл Layout.pug, который является так называемой мастер-страницей, позволяющей отделить общие блоки разметки от остальных представлений (листинг 1).

```
<!DOCTYPE html>

<html>

<head>

  block title

  block scripts

  <meta charset="utf-8" />

</head>

<body>

  block content

</body>

</html>
```

Листинг 1– Содержание файла Layout.pug

В качестве примера страницы, сделанной с помощью шаблонизатора Pug, представлена страница авторизации (листинг 2).

```
extends Layout.pug

block title

  <title>#{title}</title>

block content

  <h1>Страница авторизации</h1>

  <a href="/">Домой</a><br><br>

  <h1>Введите данные</h1>

  <form action="/login" method="post">
```

```
<label>Логин</label><br>  
<input type="text" name="login" /><br><br>  
<label>Пароль</label><br>  
<input type="password" name="password" /><br><br>  
<input type="submit" value="Отправить" />  
</form>
```

Листинг 2 – Разметка страницы авторизации

3.1.2 Слой данных

Для реализации слоя данных была использована ORM-библиотека Sequelize, позволяющая работать с данными из БД как с объектами. Ее использование позволило избавиться от необходимости писать большое количества SQL-запросов и сделало работу с данными более наглядной. Так как в системе администрирования отсутствуют большие и сложные SQL-запросы, потеря производительности, возникающая при использовании ORM-технологии, будет несущественной.

В результате анализа требуемых от системы функций, была спроектирована модель БД, представленная в приложении Б. Реализация БД проходила с использованием подхода CodeFirst, который требует только описание используемых моделей в коде приложения, а затем самостоятельно создает сущности в БД. Список моделей, разрабатываемой системы администрирования, изображен на рисунке 5.

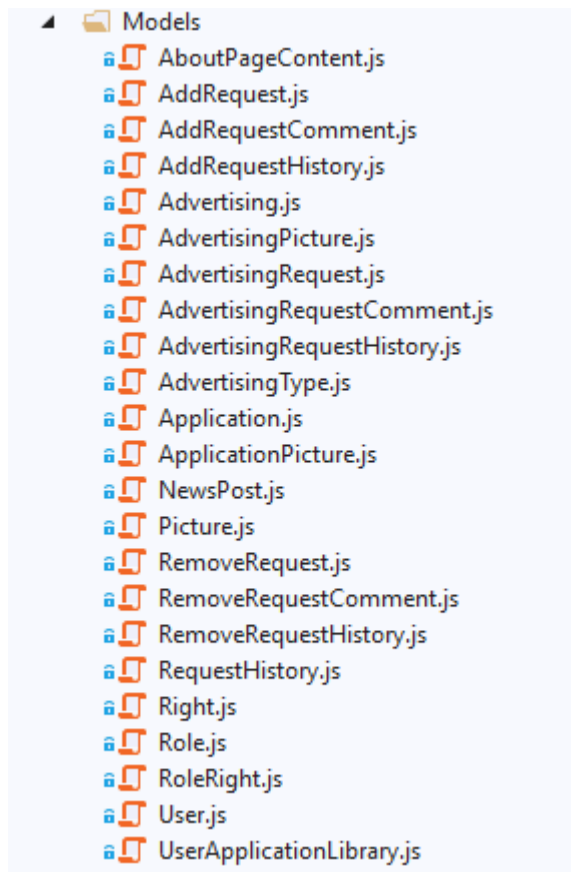


Рисунок 5 – Список моделей системы администрирования

Все представленные на рисунке 5 файлы имеют схожую структуру. Для примера рассмотрим содержимое файла AdvertisingRequestComment.js (листинг 3). В нем содержится одна функция CreateModel, содержащая описание полей модели и ее связей. Эта функция нужна для определения модели в системе. Сам процесс определения модели представлен на рисунке 6 в виде диаграммы последовательности.

```
const sequelize = require("sequelize");

module.exports.CreateModel = function (sequelizeInstance, modelName) {

  let advertisingRequestComment = sequelizeInstance.define(modelName, {

    Id: {

      type: sequelize.INTEGER,

      autoIncrement: true,

      primaryKey: true,
```

```
allowNull: false

},

Text: {

type: sequelize.STRING,

allowNull: true

}

}

);

let advertisingRequestModel = sequelizeInstance.model('AdvertisingRequest');

let userModel = sequelizeInstance.model('User');

advertisingRequestModel.hasMany(advertisingRequestComment);

userModel.hasMany(advertisingRequestComment);

return advertisingRequestComment;

}
```

Листинг 3 – Содержимое файла AdvertisingRequestComment.js

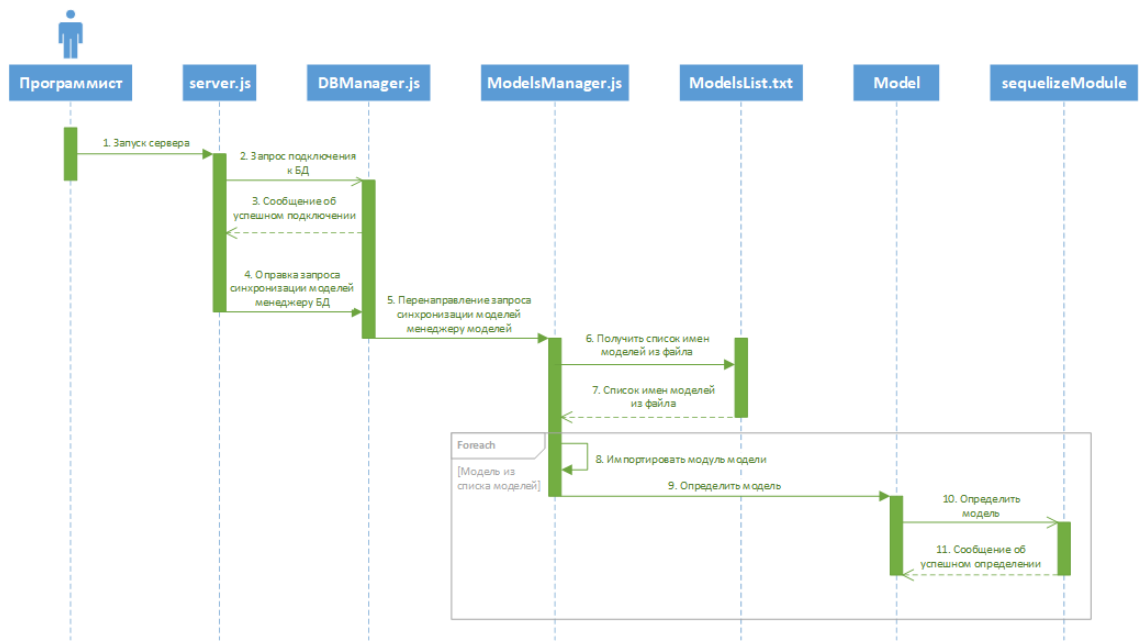


Рисунок 6 – Диаграмма последовательности для процесса определения модели

3.1.3 Управляющий слой

Управляющий слой представлен контроллерами, обрабатывающими запросы и руководящими процессом их обработки. В проекте используется фреймворк Express, который отвечает за маршрутизацию и, в случае запроса, передает управление соответствующему контроллеру. Работу контроллера AccountsController.js можно увидеть на представленной на рисунке 7 диаграмме последовательности.

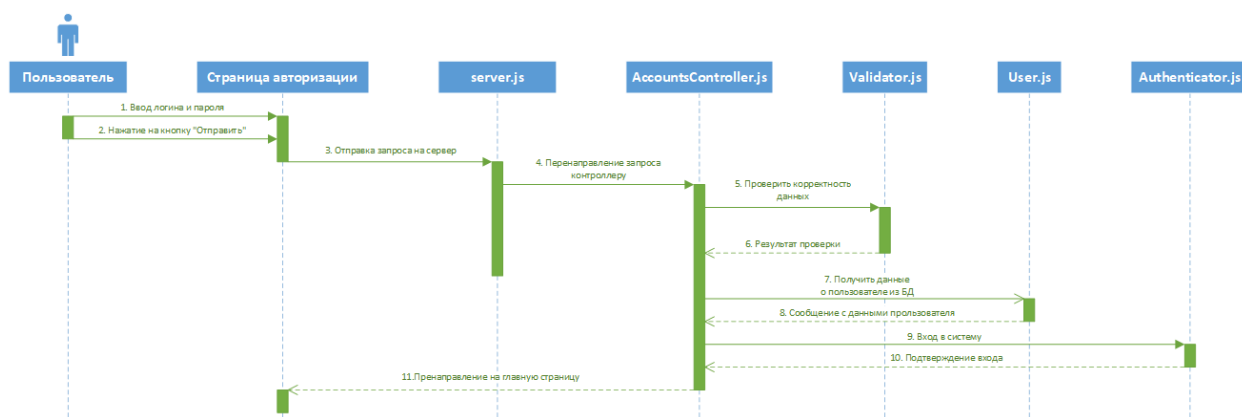


Рисунок 7 – Диаграмма последовательности для процесса авторизации

Ниже будут описаны основные контроллеры, созданные для системы администрирования:

- `RequestsController` – контроллер для работы с запросами на создание, удаление приложений, а также на размещение рекламы.
- `ApplicationsController` – контроллер для работы с приложениями.
- `NewsController` – контроллер для работы с новостями.
- `AccountsController` – контроллер для работы с аккаунтами пользователей.
- `RolesController` – контроллер управления ролями пользователей.

3.2 Реализация БД

В приложении Б приведена модель БД для разрабатываемой системы администрирования. Таблицы данной БД и их назначение приведено в приложении В.

4 Результат практического исследования

разработанной платформы

4.1.1 Изменение вида страниц в зависимости от роли

Для работы с системой в первую очередь необходимо пройти авторизацию. Неавторизованному пользователю на данный момент доступно только четыре действия: вход, регистрация, просмотр страницы «Новости», просмотр страницы «О нас» (рисунок 8). Для демонстрации было создано 3 аккаунта, имеющих разные роли: admin, developer и User. Как можно увидеть из рисунков 9-11, вид навигационной панели для главной страницы для различных ролей отличается.

Вы вошли как гость.

[Вход](#) [Регистрация](#) [Новости](#) [О нас](#)

Рисунок 8 – Навигационная панель для неавторизованного пользователя

Вы вошли как admin.

[Выход](#) [Профиль](#) [Администрирование](#) [Новости](#) [О нас](#)

Рисунок 9 – Навигационная панель для администратора

Вы вошли как developer.

[Выход](#) [Профиль](#) [Кабинет разработчика](#) [Новости](#) [О нас](#)

Рисунок 10 – Навигационная панель для разработчика

Вы вошли как User.

[Выход](#) [Профиль](#) [Новости](#) [О нас](#)

Рисунок 11 – Навигационная панель для пользователя

4.1.2 Работа с заявками в системе администрирования

Разработанная система администрирования позволяет разработчиками добавлять, удалять и рекламировать свои продукты, данный функционал поддерживается благодаря заявкам, отправляемым на рассмотрение модераторам и администраторам веб-платформы. Далее описывается прохождение пути заявки от отправки до одобрения на примере заявки на добавление приложения (как типовой).

Войдя в систему в качестве разработчика (рисунок 10) и перейдя в кабинет разработчика (рисунок 12), необходимо нажать на ссылку «Подать заявку на добавление игры или приложения». Эти действия перенаправляют на страницу подачи заявки (рисунок 13). Заполнив все поля и нажав на кнопку «Подать заявку», пользователь отправит форму на сервер, и, если сохранение заявки прошло удачно, мы сможем увидеть ее на странице «Список отправленных заявок» (рисунок 14).

Кабинет разработчика

[Домой](#)

[Подать заявку на добавление игры или приложения](#)

[Подать заявку на удаление игры или приложения](#)

[Подать заявку на публикацию рекламы](#)

[Мои игры и приложения](#)

[Список отправленных заявок](#)

Рисунок 12 – Навигационная панель кабинета разработчика

Страница подачи заявки на добавление игры или приложения

[Домой](#)

Введите данные

Название

Тип

Описание

Тип подключения

IP-адрес

Порт

Цена

Иконка
 Icon.png

Главное изображение
 MainPicture.png

Дополнительные изображения
 Число файлов: 2

Рисунок 13 – Страница подачи заявки на добавление приложения

Список отправленных заявок

[Домой](#)

Тип заявки:

Статус заявки:

Дата: с по

Название приложения	Тип заявки	Статус заявки	Дата отправки	Страница заявки
Игра 1	Добавление	Не рассмотрена	03.06.2020	Перейти

Рисунок 14 – Страница со списком отправленных заявок

Таким образом, отправленную заявку смогут увидеть модератор и администратор. Далее можно зайти в систему в качестве администратора и

перейти на страницу администрирования (рисунок 15), где нужно выбрать пункт меню «Заявки от разработчиков». Эти действия позволяют осуществить переход на соответствующую страницу (рисунок 16).

Страница администрирования

[Домой](#)

[Заявки от разработчиков](#)

[Создать новость](#)

[Редактировать новость](#)

[Список всех пользователей](#)

[Редактирование страницы «О нас»](#)

[Управление ролями](#)

Рисунок 15 – Навигационная панель страницы администрирования

Заявки от разработчиков

[Домой](#)

Тип заявки:

Статус заявки:

Дата: с по

Название приложения	Тип заявки	Статус заявки	Дата отправки	Страница заявки
Игра 1	Добавление	Не рассмотрена	03.06.2020	Перейти

Рисунок 16 – Страница «Заявки от разработчиков»

Прежде чем одобрить заявку, ее необходимо проверить. Для этого, осуществляется переход на страницу заявки (рисунок 17), и проверяются все поля на корректность их заполнения. После этого, при условии, что все поля содержат корректные данные, пишется комментарий, и проверка завершается (рисунок 18).

Страница заявки

[Домой](#)

Название:

Тип:

Описание:

Тип подключения:

IP-адрес:

Порт:

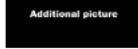
Цена:

Иконка:



Рисунок 17 – Страница заявки

Дополнительные изображения:



Ник разработчика:

Тип заявки:

Статус заявки:

Комментарии к заявке

Комментарий:

Рисунок 18 – Завершение проверки заявки

Когда приложение проверено, можно одобрить заявку на ее добавление, нажав соответствующую кнопку (рисунок 19). После этого, статус заявки изменится, и приложение будет считаться добавленным на веб-платформу. Разработчик может увидеть новый статус своей заявки на странице «Список отправленных заявок» (рисунок 20). Также, одобренное приложение теперь отображается на странице «Мои приложения», которая доступна из кабинета разработчика (рисунок 21).

Ник разработчика:

Тип заявки:

Статус заявки:

Комментарии к заявке

Автор комментария:

Комментарий:

Комментарий:

Рисунок 19 – Одобрение заявки

Список отправленных заявок

[Домой](#)

Тип заявки:

Статус заявки:

Дата: с по

Название приложения	Тип заявки	Статус заявки	Дата отправки	Страница заявки
Игра 1	Добавление	Одобрена	03.06.2020	Перейти

Рисунок 20 – Новый статус заявки

Мои игры и приложения

[Домой](#)

Название	Тип	Описание	Тип подключения	IP-адрес	Порт	Цена	Иконка	Главное изображение	Дополнительное изображение
Игра 1	Игра	Лучшая игра	TCP	127.0.0.1	3001	150	Иконка	Главное изображение	Дополнительное изображение 1 Дополнительное изображение 2

Рисунок 21 – Страница «Мои игры и приложения»

4.1.3 Редактирование игры или приложения

Вход на страницу редактирования осуществляется через страницу приложения (рисунок 22), попасть на которую возможно только авторизовавшись как разработчик и кликнув на название приложения на странице «Мои игры и приложения» (рисунок 21). При нажатии на кнопку «Редактировать», будет осуществлён переход на страницу редактирования (рисунок 23), где можно изменить все поля, кроме типа подключения, IP-адреса и порта. Для сохранения изменений необходимо нажать кнопку «Сохранить изменения».

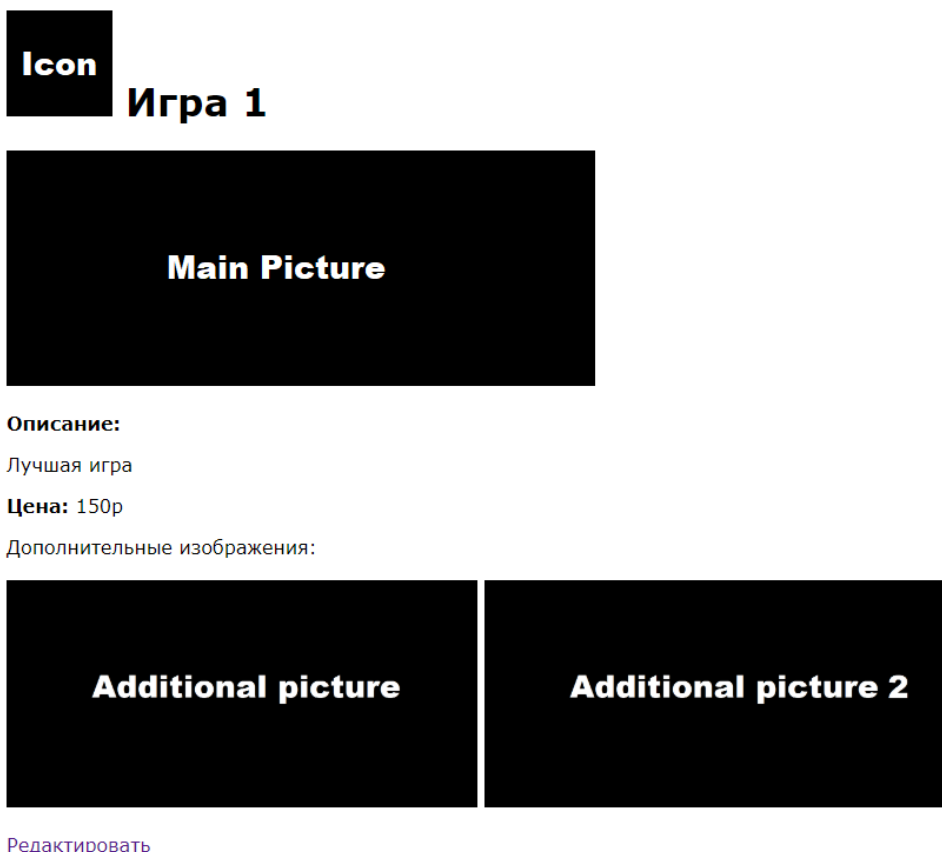


Рисунок 22 – Страница приложения

Страница редактирования игры или приложения

[Домой](#)

Название

Тип

Описание

Тип подключения

IP-адресс

Порт

Цена

Иконка
 Icon.png

Главное изображение
 MainPicture.png

Дополнительные изображения
 Число файлов: 2

Рисунок 23 – Страница редактирования приложения

4.1.4 Просмотр списка всех пользователей

Перейти на страницу просмотра всех пользователей можно со страницы администрирования. На ней будут отображены все пользователи, зарегистрированные на данный момент на веб-платформе, и их роль (рисунок 24).

Список всех пользователей

[Домой](#)

Ник пользователя	Роль
admin	Администратор
developer	Разработчик
User	Пользователь

Рисунок 24 – Список всех пользователей

4.1.5 Просмотр и редактирование информации о пользователе

Находясь на странице «Список всех пользователей» (рисунок 24), можно перейти в профиль любого пользователя, нажав на его ник (рисунок 25). Администратор может изменить информацию о пользователе, перейдя по ссылке «Редактировать основную информацию», или удалить пользователя из системы нажав «Удалить пользователя».

Профиль

[Домой](#)

Ник:

User

Email:

111@xmail.com

[Редактировать основную информацию](#)

[Удалить пользователя](#)

Рисунок 25 – Профиль пользователя

4.1.6 Создание и редактирование новости

Перейти на страницу создания новости можно со страницы администрирования. На новой странице можно увидеть два поля, одно для заголовка, а второе для основной части (рисунок 26). Благодаря использованию TinyMCE, к тексту новости можно применить множество стилей. После заполнения всех полей, необходимо нажать на кнопку «Опубликовать», и новая запись появится на странице новостей (рисунок 27).

Создать новость

[Домой](#)

Заголовок

Игра 1 теперь доступна на

Содержание

File	Edit	View	Format								
↶	↷	Paragraph	▼	B	<i>I</i>	≡	≡	≡	≡	≡	≡
<p>Уважаемые пользователи нашей веб-платформы, мы рады сообщить вам, что только что Игра 1 была добавлена в каталог игр. Приятной вам игры!</p>											
Р											
<input type="button" value="Опубликовать"/>											

Рисунок 26 – Создание новой новости

Новости веб-платформы

[Домой](#)

<p>Заголовок: Игра 1 теперь доступна на веб-платформе!</p>
<p>Содержание:</p> <p>Уважаемые пользователи нашей веб-платформы, мы рады сообщить вам, что только что Игра 1 была добавлена в каталог игр. Приятной вам игры!</p>

Рисунок 27 – Страница новостей

Для перехода на страницу редактирования новости необходимо сделать следующее: на странице администрирования нажать на элемент меню «Редактировать новость», на открывшейся странице (рисунок 28) выбрать

интересующую пользователя новость, и кликнуть по ее заголовку. Эти действия позволят открыть страницу редактирования новости (рисунок 29).

Список новостей для редактирования

[Домой](#)

Дата: с по

Заголовок	Дата создания
Игра 1 теперь доступна на веб-платформе!	03.06.2020

Рисунок 28 – Список новостей для редактирования

Редактирование новости

[Домой](#)

Заголовок

Содержание

File	Edit	View	Format
		Bold	
	<p>Уважаемые пользователи нашей веб-платформы, мы рады сообщить вам, что только что Игра 1 была добавлена в каталог игр. Приятной вам игры!</p>		
P » EM » STRONG			

Рисунок 29 – Редактирование новости

После завершения редактирования, для сохранения изменений, необходимо нажать на кнопку «Обновить новость». Отредактированную новость теперь можно увидеть на странице новостей (рисунок 30).

Новости веб-платформы

[Домой](#)

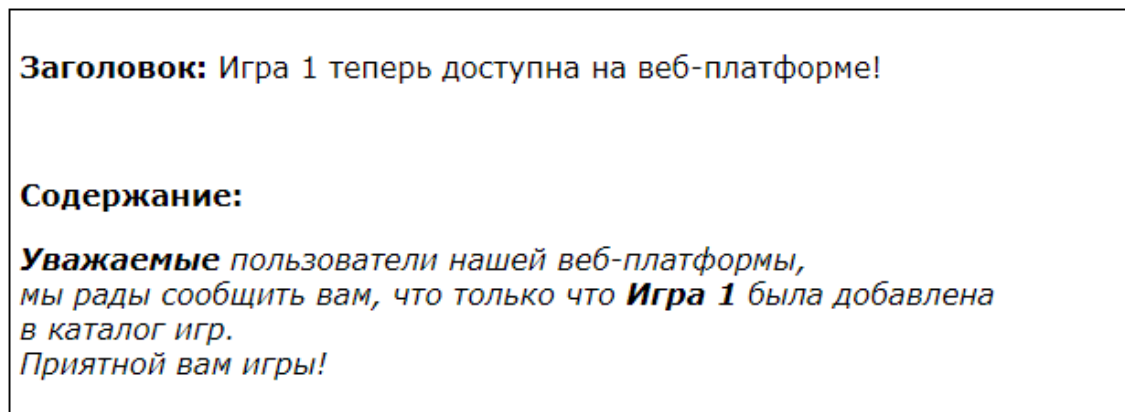


Рисунок 30 – Обновленная новость

4.1.7 Управление ролями

Каждый пользователь в системе имеет одну из 5 ролей: пользователь, забаненный пользователь, разработчик, модератор и администратор. В зависимости от роли они видят различные отображения одних и тех же страниц (рисунок 9 – 11), а также имеют права на определенные действия. Например, если обычный пользователь попытается зайти на страницу администрирования, то результат будет как на рисунке 31.

< > C ☰ | VPN 🌐 localhost:3000/administration

User, у вас недостаточно прав для доступа к этой странице.

[Домой](#)

Рисунок 31 – Попытка входа на страницу администрирования

Администратор имеет возможность управлять правами доступа различных ролей. Эти действия осуществляются на странице управления ролями (рисунок 32). Для изменения прав роли необходимо установить или убрать метку рядом с описанием предоставляемых возможностей, а после окончания редактирования нажать кнопку сохранения изменений.

Управление ролями

[Домой](#)

Пользователь

- Права авторизованного пользователя
- Доступ к функциям кабинета разработчика
- Доступ к странице администрирования
- Проверка заявок от разработчиков
- Создание и редактирование новостей
- Редактирование страницы "О нас"
- Просмотр списка всех пользователей
- Редактирование информации о любом пользователе
- Управление ролями
- Одобрение или отклонение заявок от разработчиков
- Удаление игры или приложения из каталога
- Полное удаление игры или приложения с веб-платформы
- Редактирование информации о любой игре или любом приложении
- Удаление любого пользователя

Разработчик

- Права авторизованного пользователя
- Доступ к функциям кабинета разработчика
- Доступ к странице администрирования
- Проверка заявок от разработчиков
- Создание и редактирование новостей
- Редактирование страницы "О нас"
- Просмотр списка всех пользователей
- Редактирование информации о любом пользователе
- Управление ролями
- Одобрение или отклонение заявок от разработчиков
- Удаление игры или приложения из каталога
- Полное удаление игры или приложения с веб-платформы
- Редактирование информации о любой игре или любом приложении
- Удаление любого пользователя

Рисунок 32 – Страница управления ролями

5 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

5.1 Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения

5.1.1 Потенциальные потребители результатов исследования

В данной НИР рассматривается разработка системы администрирования веб-платформы для игр и приложений, которая не будет продаваться отдельно. Конечным же продуктом, который будет приносить прибыль разработчикам является именно веб-платформа для игр и приложений, над которой параллельно работают другие люди. В связи с этим, предлагаю рассмотреть потенциальных потребителей относительно нее.

Эта система разрабатывается для того, чтобы неуверенные разработчики и компании, желающие попробовать себя в новом направлении, могли протестировать свои продукты на реальных пользователях. Благодаря тому, что игры и приложения с веб-платформы не нужно будет устанавливать себе на устройство, пользователи с большим желанием будут ими пользоваться и пробовать что-то новое.

Разрабатываемая система пока что поддерживает только русский язык, поэтому основные потребители будут из России. Проведем сегментирование рынка по критериям вид клиента, способ взаимодействия с сервером (таблица 1).

Таблица 1 – Карта сегментирования рынка игр и приложений в России

		Тип решения			
		Облачный сервис	Веб-приложение	Клиент-серверное приложение	Клиент-сервер-серверное приложение
вид клиента	Веб-клиент				
	Десктоп-клиент				

В таблице 1 каждый цвет соответствует определенному продукту: синий – PlayKey; зеленый – игры и приложения социальной сети «В Контакте»; красный – World Of Tanks; желтый – Google документы. Более подробно конкуренты будут рассмотрены в подразделе 1.2.

Из таблицы 1 видно, что есть четыре свободные ниши. Создаваемая веб-платформа ориентируется на рынок клиент-сервер-серверных приложений с веб-клиентом.

5.1.2 Анализ конкурентных технических решений

Системы администрирования не являются готовым решением, которое можно купить и встроить в имеющийся программный продукт, и должны разрабатываться индивидуально, в соответствии с нуждами конкретного программного обеспечения. Как следствие, среди них нет конкуренции, поэтому предлагаю рассмотреть конкурентов веб-платформы для игр и приложений, в состав которой будет входить разрабатываемая в рамках данной НИР система администрирования.

В подпункте 5.1.1 был выявлен ряд продуктов, занимающих соседние ниши с рассматриваемой веб-платформой. Предлагаю ниже рассмотреть их более подробно.

Клиент-серверное приложение «World Of Tanks» требует установки одной из своих частей на персональный компьютер пользователя. Их клиентская часть очень тяжелая и является требовательной к ресурсам, поэтому перенести ее в браузер, с сохранением быстродействия, необходимого для комфортной игры, не представляется возможным.

Отличие от приложений в социальной сети «В Контакте» заключается во взаимодействии разрабатываемой веб-платформы с сервером приложения разработчика.

Игровой стриминговый сервиса «PlayKey» запускает приложение на виртуальной машине и предоставляет пользователю только поток изображения с нее. Принятый видеопоток на специальном клиенте отображается преобразуется в картинку. Это совершенно другая технология, и бояться конкуренции этого продукта не стоит.

Что касается Google документов, то это конечное программное обеспечение, распространяемое как услуга, разрабатываемое и поддерживаемое компанией Google. Наша веб-платформа, является именно платформой, для приложений наподобие Google документов, а не конечным продуктом.

В заключение можно сделать вывод, что веб-платформа для игр и приложений, в состав которой будет входить разрабатываемая в рамках данной НИР система администрирования, прямых конкурентов не имеет.

5.1.3 QuaD-анализ

Для оценки качества и перспективности разрабатываемой системы администрирования был проведен анализ по технологии QuaD (таблица 2).

Таблица 2 – Анализ по технологии QuaD

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Средневзвешенное значение (*100)
1	2	3	4	5	
Показатели оценки качества разработки					
1.Потребность в ресурсах памяти	0,1	70	100	0,7	7
2.Функциональность	0,1	75	100	0,75	7,5
3.Простота эксплуатации	0,1	90	100	0,9	9
4.Скорость работы	0,15	70	100	0,7	10,5
5.Надежность	0,15	80	100	0,8	12
6.Удобство эксплуатации	0,1	80	100	0,8	8
7.Точность	0,2	90	100	0,9	18
Показатели оценки коммерческого потенциала разработки					
8.Срок выхода на рынок	0,05	90	100	0,9	4,5
9.Сопровождение продукта после публикации	0,05	95	100	0,95	4,75
Итого	1	750	900	7,5	81,25

Средневзвешенное значение показателя качества и перспективности разрабатываемой системы равна 81,25. Таким образом, данная разработка считается перспективной (находится в пределах от 80 до 100).

5.1.4 SWOT-анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT-анализ применяют для исследования внешней и внутренней среды проекта. Сильные и слабые стороны являются факторами внутренней среды объекта анализа, возможности и угрозы являются факторами внешней среды.

Результат проведения первого этапа SWOT-анализа приведен в таблице 3.

Таблица 3 – Матрица SWOT

	<p>Сильные стороны научно-исследовательского проекта:</p> <p>С1. Простой в эксплуатации</p> <p>С2. Точные и почти безошибочные выполнение необходимых операций</p> <p>С3. Удобен в использовании</p> <p>С4. Высокий уровень безопасности</p>	<p>Слабые стороны научно-исследовательского проекта:</p> <p>Сл1. Не очень высокая скорость работы</p> <p>Сл2. Проект требователен к ресурсам сервера</p> <p>Сл3. Функциональность проекта достаточная, но не идеальная</p>
--	--	--

Продолжение

<p>Возможности:</p> <p>В1. Появление более быстрых библиотек, чем используются сейчас.</p>		
<p>Угрозы:</p> <p>У1. Появления уязвимостей в используемых в разрабатываемой системе технологиях.</p>		

Чтобы более детально рассмотреть степень соответствия возможностей и угроз сильным и слабым сторонам проекта, построим две интерактивные матрицы (таблица 4 – 5).

Таблица 4 – Интерактивная матрица проекта (сильные стороны)

Сильные стороны проекта					
		С1	С2	С3	С4
Возможности проекта	В1	-	-	+	-
Угрозы проекта	У1	-	-	-	+

Таблица 5 – Интерактивная матрица проекта (слабые стороны)

Слабые стороны проекта				
		Сл1	Сл2	Сл3
Возможности проекта	V1	+	+	-
Угрозы проекта	У1	+	-	-

В результате работ над данным подпунктом была составлена итоговая матрица SWAT-анализа (таблица 6), выявлены сильные и слабые стороны проекта и возможные направления дальнейших улучшений системы.

Таблица 6 – Итоговая матрица SWOT

	<p>Сильные стороны научно-исследовательского проекта:</p> <p>С1. Простой в эксплуатации</p> <p>С2. Точные и почти безошибочные выполнение необходимых операций</p> <p>С3. Удобен в использовании</p> <p>С4. Высокий уровень безопасности</p>	<p>Слабые стороны научно-исследовательского проекта:</p> <p>Сл1. Не очень высокая скорость работы</p> <p>Сл2. Проект требователен к ресурсам сервера</p> <p>Сл3. Функциональность проекта достаточная, но не идеальная</p>
--	--	--

Продолжение

<p>Возможности:</p> <p>В1. Появление более быстрых библиотек, чем используются сейчас.</p>	<p>Улучшение скорости работы разрабатываемой системы сделает ее более удобной для использования.</p>	<p>Необходимо следить за выходом новых библиотек, которые помогут увеличить скорость работы и снизить потребность в ресурсах.</p>
<p>Угрозы:</p> <p>У1. Появления уязвимостей в используемых в разрабатываемой системе технологиях.</p>	<p>Появление уязвимостей сделает разрабатываемую систему менее безопасной, а безопасность, одно из ключевых свойств систем администрирования.</p>	<p>Использование злоумышленником уязвимостей может сильно замедлить систему или даже остановить ее работу.</p>

5.2 Определение возможных альтернатив проведения научных исследований

Так как разработка системы находится на стадии создания конечного продукта, предлагаю рассмотреть несколько вариантов совершенствования процесса разработки или направлений научного исследования.

Одним из вариантов является исследование возможности использования микросервисов в разрабатываемой системе. Они помогут сделать систему более стабильной и управляемой, а также обеспечат хорошую масштабируемость.

Также будет полезным изучить быстродействие различных библиотек для работы с базами данных, чтобы выявить наиболее быструю и удобную из них.

Если в результате исследования библиотек для работы с базами данных окажется, что ни одна из библиотек не даст сильного прироста в скорости работы, то необходимо рассмотреть возможность изменения процесса разработки для написания собственной библиотеки.

5.3 Планирование научно-исследовательских работ

5.3.1 Структура работ в рамках научного исследования

Перечень этапов работы и распределение исполнителей представлен в таблице 7.

Таблица 7 – Структура работ

№ п/п	Этапы работы	Исполнители
1	Выбор научного руководителя бакалаврской работы	Студент
2	Составление и утверждение темы бакалаврской работы	Студент, Научный руководитель
3	Постановка целей и задач	Студент, Научный руководитель
4	Подбор и изучение материалов по тематике	Студент
5	Разработка календарного плана	Студент, Научный руководитель
6	Проведение анализа предметной области	Студент
7	Проектирование	Студент, Научный руководитель
8	Разработка	Студент
9	Тестирование	Студент

Продолжение

10	Отладка	Студент
11	Выполнение других частей работы (фин. менеджмент, соц. Ответственность)	Студент
12	Подведение итогов, оформление работы	Студент, Научный руководитель

Научный руководитель – Саврасов Фёдор Витальевич;

Студент – Венгер Артем Александрович;

5.3.2 Определение трудоемкости выполнения работ и разработка графика проведения научного исследования

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, т.к. зависит от множества трудно учитываемых факторов.

Для расчета длительности работ в календарных днях использовался коэффициент календарности равный 1,48, что соответствует 2020 календарному году.

Рассчитываемое время для каждого из этапов проекта представлено в приложении Г.

На основе временных показателей был построен календарный план-график научного исследовательского проекта. В графике использованы максимальные по длительности исполнения работы из работ студента и научного руководителя. Календарный план графика в виде диаграммы Ганта представлен в Приложении Е.

5.4 Бюджет научно-технического исследования (НТИ)

Для планирования бюджета НТИ, необходимо отразить все расходы, связанные с его выполнением. В последующих пунктах будет подробно рассмотрена каждая из статей расходов.

5.4.1 Расчет материальных затрат НТИ

В материальных затратах учтены только расходы на канцелярские принадлежности и картриджи для принтера, так как все необходимые для работы над проектом материалы имелись в распоряжении исполнителей. Материалы, необходимые для выполнения данной работы, и расчет материальных затрат представлены в таблице 8.

Таблица 8 – Материальные затраты

Наименование	Единица измерения	Количество	Цена за ед., руб.
Бумага для принтера, А4	уп.	1	350,00
Ручка шариковая	шт.	2	56,00
Итого:			462,00

5.4.2 Расчет амортизационных затрат

В специальное оборудование входят оборудование для рабочего места и ПО, необходимое для реализации программного обеспечения. Затраты на специальное оборудование приведены в таблице 9.

Таблица 9 – Затраты на специальное оборудование

Наименование оборудования	Количество единиц оборудования	Цена за 1 ед. оборудования	Затраты, руб.
Персональный компьютер	1	40000	40000
Монитор	1	10499	10499
Клавиатура	1	599	599
Компьютерная мышь	1	199	199
Windows 10 Pro	1	21500	21500
Итого:			72797

Амортизационные отчисления для рассматриваемого проекта включают в себя амортизацию используемого оборудования за время выполнения работы. Амортизационные отчисления рассчитываются по времени использования компьютера по формуле:

$$C_{AM} = \frac{N_A * C_{OB}}{F_D} * t_{рф} * n, \quad (1)$$

где N_A – годовая норма амортизации;

C_{OB} – цена оборудования;

F_D – действительный годовой фонд рабочего времени;

$t_{рф}$ – время работы вычислительной техники;

n – число задействованных единиц оборудования, $n = 1$.

Годовая амортизация H_A определяется как величина, обратная сроку амортизации оборудования C_A , который определяется согласно постановлению правительства РФ «О классификации основных средств, включенных в амортизационные группы». Для компьютера и периферийного оборудования, используемого с ним, примем $C_A = 3$ года, тогда $H_A = 0,33$.

Расчет затрат на амортизационные отчисления представлен в таблице 10.

Таблица 10 – Затраты на амортизационные отчисления

Наименование оборудования	Норма амортиз. Оборуд., H_A	Стоим. Оборуд., Цоб, руб.	Факт. р/вр. Оборуд., $t_{рф}$, ч	Действ. Год. Фонд р/вр., F_d , ч.	Аморт. Отчисл., $C_{ам}$, руб.
Персональный компьютер	0,33	40000	240	1976	1603,24
Монитор	0,33	10499	240	1976	420,81
Windows 10 Pro	0,33	21500	240	1976	861,74
Итого:					2885,79

5.4.3 Основная заработная плата исполнителей темы

Планирование основной заработной платы производилось из расчета того, что основная заработная плата научного руководителя составляет 1500 руб/день, студента – 900 руб/день.

Планирование основной заработной платы приведено в приложении Д.

Таблица 11 – Баланс рабочего времени

Показатели рабочего времени	Студент	Научный руководитель
Календарное число дней	366	366
Количество нерабочих дней – выходные дни – праздничные дни	118	118
Потери рабочего времени – отпуск – невыходы по болезни	30	30
Действительный годовой фонд рабочего времени	218	218

Таблица 12 – Расчет основной заработной платы

Исполнитель	Разряд	З _{тс} , руб.	к _{пр}	к _д	к _р	З _м , руб	З _{дн} , руб	Т _р , раб. Дн	З _{осн} , руб
Студент	Младший программист	20769	-	-	1,3	27000	900	30,1	27090
Научный руководитель	Ведущий программист	34615	-	-	1,3	45000	1500	4,1	6150
Итого:									33240

5.4.4 Дополнительная заработная плата исполнителей темы

Расчет дополнительной заработной платы ведется по формуле:

$$З_{\text{доп}} = k_{\text{доп}} * З_{\text{осн}}, \quad (2)$$

где $З_{\text{доп}}$ – затраты по дополнительной заработной плате исполнителей темы;

$k_{\text{доп}}$ – коэффициент дополнительной заработной платы (на стадии проектирования принимается равным 0,12 – 0,15);

$З_{\text{осн}}$ – затраты по основной заработной плате исполнителей темы.

В данном случае коэффициент дополнительной заработной платы будет взят равным 0,13. Таким образом, затраты на дополнительную заработную плату можно считать равными 4321,2 рублей.

5.4.5 Отчисления во внебюджетные фонды

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений во внебюджетные фонды определяется исходя из формулы:

$$З_{\text{внеб}} = k_{\text{внеб}} * (З_{\text{осн}} + З_{\text{доп}}), \quad (3)$$

где $З_{\text{внеб}}$ – затраты на отчисления во внебюджетные фонды;

$k_{\text{внеб}}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.);

$З_{\text{осн}}$ – затраты по основной заработной плате исполнителей темы;

$З_{\text{доп}}$ – затраты по дополнительной заработной плате исполнителей темы.

Для учреждений, осуществляющих научную и образовательную деятельность, коэффициент отчислений берётся равным 28%. Результаты расчета приведены в таблице 13.

Таблица 13 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб	Дополнительная заработная плата, руб
Студент	27090	3521,7
Научный руководитель	7650	799,5
Коэффициент отчислений во внебюджетные фонды	0,28	
Итого		
Студент	8571,27	
Научный руководитель	1945,86	

5.4.6 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = (\text{сумма статей } 1 \div 5) * k_{\text{нр}}, \quad (4)$$

где $Z_{\text{накл}}$ – затраты на накладные расходы;

$k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы.

Величину коэффициента накладных расходов можно взять в размере 16%.

Таким образом, накладные расходы для максимального по длительности исполнения работ можно считать равными 19875,7 рублей.

5.4.7 Формирование бюджета затрат научно-исследовательского проекта

По итогам всех рассчитанных статей для каждого из вариантов исполнения приведён расчёт бюджета затрат НИИ (таблица 14).

Таблица 14 – Расчет бюджета затрат НИИ

Наименование статьи	Сумма, руб		Примечание
	Студент	Научный руководитель	
Материальные затраты	462	0	4.1
Затраты на специальное оборудование для научных работ	72797	0	4.2
Амортизационные затраты	2885,79	0	4.2
Основная заработная плата исполнителей	27090	6150	4.3
Дополнительная заработная плата исполнителей	3521,7	799,5	4.4

Продолжение

Отчисления во внебюджетные фонды	8571,27	1945,86	4.5
Накладные расходы	18452,44	1423,26	4.6
Бюджет затрат НИИ	144098,82		4.7

5.5 Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальное и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный финансовый показатель разработки определяется как:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{ri}}{\Phi_{\text{max}}}, \quad (5)$$

где $I_{\text{финр}}$ – интегральный финансовый показатель разработки;

Φ_{ri} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в разгах (значение больше единицы), либо соответствующее численное

удешевление стоимости разработки в разгах (значение меньше единицы, но больше нуля).

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i * b_i, \quad (6)$$

где I_{pi} – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

n – число параметров сравнения.

Результаты расчета интегрального показателя ресурсоэффективности приведены в таблице 15.

Таблица 15 – Расчет интегрального показателя ресурсоэффективности

Объект исследования/Критерии	Весовой коэффициент параметра	Исп.1
1.Потребность в ресурсах памяти	0,1	5
2.Функциональность	0,1	3
3.Простота эксплуатации	0,1	3
4.Скорость работы	0,15	5
5.Надежность	0,15	4
6.Удобство эксплуатации	0,2	3
7.Точность	0,2	4
I_p		2,7

В таблице 16 приведены результаты расчета интегрального показателя эффективности разработки.

Таблица 16 – Результаты расчета интегрального показателя эффективности разработки

№ п/п	Показатели	Исп.1
1	Интегральный финансовый показатель разработки	1
2	Интегральный показатель ресурсоэффективности разработки	2,7
3	Интегральный показатель эффективности	2,7

Из полученной таблицы видно, что разрабатываемая в рамках данной НИР система, имеет показатель эффективности немного выше среднего.

Заключение

В результате работы по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» был рассмотрен проект, выполняемый в рамках научно-исследовательской работы, определены его слабые и сильные стороны, конкуренты и эффективность. Также был посчитан бюджет затрат, который составил 144098,82 руб., и срок выполнения работы, равный 53 календарным дням.

6 Социальная ответственность

Введение

В наши дни, большой популярностью пользуются облачные SaaS решения, с которыми пользователи могут работать в браузере, ничего не устанавливая себе на компьютер или смартфон. Одним из примеров таких решений являются веб-платформы для игр и приложений. Такие проекты не могут существовать без качественной системы администрирования. Научно-исследовательская работа заключается в написании системы администрирования веб-платформы для игр и приложений. Она разрабатывается для людей, также называемых администраторами, которые будут осуществлять управление веб-платформой и контроль за ней. Чтобы администраторы веб-платформы могли удобно и эффективно выполнять возложенные на них обязанности, система будет поддерживать различный функционал, такой как: функционал взаимодействия с аккаунтами пользователей, функционал управления играми и приложениями, функционал управления правами доступа, функционал управления новостями.

Данная система разрабатывалась во время прохождения производственной практики в ТПУ, ИШИТР, ОИТ в 418 аудитории кибернетического центра ТПУ.

6.1 Правовые и организационные вопросы

обеспечения безопасности

Рабочее место должно быть организовано с учетом требований ГОСТ 12.2.032-78 «Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования» [7] и СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» [8].

При организации работы с ЭВМ, согласно указанным выше требованиям, должны быть соблюдены следующие условия:

- площадь на одно рабочее место пользователя с ПК должна составлять не менее 6 м²;
- конструкция рабочей мебели должна обеспечивать возможность индивидуальной регулировки соответственно росту пользователя и создавать удобную позу для работы;
- ПК и, соответственно, рабочее место должно располагаться так, чтобы свет падал сбоку;
- расстояние от ПК до стен должно быть не менее 1 м, по возможности следует избегать расположения рабочих мест в углах помещения либо лицом к стене;
- Рабочее место должно быть установлено так, чтобы, подняв глаза от экрана, можно было увидеть какой-нибудь удаленный предмет в помещении или на улице, таким образом, предоставляя эффективный способ разгрузки зрительного аппарата;
- окна в помещениях с ПК должны быть оборудованы регулируемыми устройствами – жалюзи, занавески, внешние козырьки;
- монитор, клавиатура и корпус компьютера должны находиться прямо перед работником;
- высота рабочего стола с клавиатурой должна составлять 680 - 800 мм над уровнем стола;
- высота экрана над полом – 900-1280 мм, монитор должен находиться на расстоянии 600-700 мм от работника на 20 градусов ниже уровня глаз;
- рабочее кресло должно иметь мягкое сиденье и спинку, с регулировкой сиденья по высоте с удобной опорой для поясницы;
- положение тела пользователя относительно монитора должно соответствовать направлению просмотра под прямым углом 90 градусов или под углом 75 градусов;

На рабочем месте, предоставленном для разработки системы администрирования веб-платформы для игр и приложений, были соблюдены все требования по организации труда с ЭВМ.

6.2 Производственная безопасность

Разработчики программного обеспечения на своих рабочих местах подвергаются воздействию различных вредных и опасных факторов. Все выявленные факторы представлены в Таблице 17, вместе с нормативными документами, относящимся к ним и этапами работ, во время которых работники могут столкнуться с их влиянием.

Таблица 17 – Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ			Нормативные документы
	Разработка	Внедрение	Эксплуатация	
Отклонение показателей микроклимата	+	+	+	СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений» [9]
Недостаточная освещенность рабочей зоны	+	+	+	СП 52.13330.2016 «Естественное и искусственное освещение» [10]
Повышенный уровень шума на рабочем месте	+	+	+	СН 2.2.4/2.1.8.562-96 «Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки» [11]

Исходя из данной таблицы можно сделать вывод, что на разработчиков программного обеспечения в ходе их деятельности воздействуют только физические и психологические факторы, а химические и биологические факторы отсутствуют.

6.2.1 Отклонение показателей микроклимата

Использование персональных компьютеров может привести к повышению температуры и снижению относительной влажности в рабочем помещении, то есть к изменению микроклимата. Отклонение показателей микроклимата от комфортных может повлиять на здоровье работников. Понижение температуры и повышение скорости движения воздуха способствуют усилению конвективного теплообмена и процесса теплоотдачи при испарении пота, что может привести к переохлаждению организма. Недостаточная влажность приводит к интенсивному испарению влаги со слизистых оболочек, что приводит к их пересыханию, растрескиванию, а затем и заражению болезнетворными микробами.

Нормативные показатели микроклимата регламентируются СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений». Санитарные нормы устанавливают оптимальные и допустимые значения показателей в рабочей зоне, что позволяет создавать благоприятные условия работы, соответствующие физиологическим потребностям организма человека.

Работа, выполняемая разработчиком системы администрирования веб-платформы для игр и приложений, относится к категории Ia, так как она является мало подвижной и мало интенсивной, выполняется в положении сидя с минимальными физическими напряжениями. В таблицах 18 и 19 представлены оптимальные и допустимые значения показателей микроклимата на рабочих местах для данной категории.

Таблица 18 – Оптимальные величины показателей микроклимата на рабочих местах

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	22-24	21-25	60-40	0,1
Теплый	23-25	22-26	60-40	0,1

Таблица 19 – Допустимые величины показателей микроклимата на рабочих местах

Период года	Температура воздуха, °С		Температура поверхностей, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
	диапазон ниже оптимальных величин	диапазон выше оптимальных величин			для диапазона температур воздуха ниже оптимальных величин, не более	для диапазона температур воздуха выше оптимальных величин, не более
Холодный	20,0	24,1	19,0	15	0,1	0,1
	-	-	-	-		
Теплый	21,9	25,0	26,0	75	0,1	0,2
	-	-	-	-		
Теплый	21,0	25,1	20,0	15	0,1	0,2
	22,9	28,0	29,0	75		

Для минимизации воздействия отклонений показателей микроклимата в рабочем помещении необходимо использовать кондиционеры, обогреватели и увлажнители воздуха, которые помогают привести эти показатели к оптимальным значениям.

6.2.2 Недостаточная освещенность рабочей зоны

Отсутствие достаточного количества источников освещения в рабочей зоне является причиной такого вредного фактора, как недостаточная освещенность. Недостаточная освещенность приводит к понижению работоспособности, а также может вызвать проблемы со здоровьем, а именно может повлиять на качество зрения работников.

Согласно СП 52.13330.2016 зрительную работу разработчика программного обеспечения можно характеризовать как работу разряда Б – высокой точности (наименьший эквивалентный размер объекта различения составляет 0,3-0,5 мм), подразряда 1 (относительная продолжительность зрительной работы при направлении зрения на рабочую поверхность не менее 70%). В таблице 20 представлены требования к освещению рабочего помещения для указанного разряда.

Таблица 20 – Требования к освещению рабочего помещения для разряда Б1

Искусственное освещение				Естественное освещение	
Освещенность на рабочей поверхности от системы общего освещения, лк	Цилиндрическая освещенность, лк	Объединенный показатель дискомфорта, не более	Коэффициент пульсации освещенности, Кл, %, не более	Коэффициент естественной освещенности, %, при	
				верхнем или комбинированном	боковом
300	100	21	15	3	1

Для снижения влияния фактора недостаточной освещенности необходимо, чтобы уровень естественного освещения рабочего места и яркость дисплея персонального компьютера были приблизительно одинаковыми, так как яркий свет в зоне периферийного зрения заметно увеличивает глазное напряжение, что приводит к их быстрой утомляемости. При недостаточной освещенности помещения может помочь расширение оконного проема кабинета и установка качественных источников искусственного освещения.

6.2.3 Повышенный уровень шума на рабочем месте

Использование персональных компьютеров, наличие центральной системы вентиляции и кондиционирования воздуха приводят к повышению уровня шума на рабочем месте. Повышенный уровень шума затрудняет разборчивость речи, снижает работоспособность, повышает утомляемость.

В таблице 21 представлены предельно допустимые уровни звукового давления, уровни звука и эквивалентные уровни звука для разработчиков программного обеспечения, описанные в СН 2.2.4/2.1.8.562-96.

Таблица 21 – Предельно допустимые уровни звукового давления, уровни звука и эквивалентные уровни звука для инженера-программиста

Вид трудовой деятельности, рабочее место	Уровни звукового давления, дБ в октавных полосах со среднегеометрическими частотами, Гц						
	31,5	63	125	250	500	1000	2000
Творческая деятельность, руководящая работа с повышенными требованиями, научная деятельность, конструирование и проектирование, программирование, преподавание и обучение, врачебная деятельность. Рабочие места в помещениях дирекции, проектно-конструкторских бюро, расчетчиков, программистов вычислительных машин, в лабораториях для теоретических работ и обработки данных, приема больных в здравпунктах	86	71	61	54	49	45	42

Для уменьшения воздействий шума можно использовать следующие методы:

1. экранирование рабочих мест, то есть установка перегородок между рабочими местами;
2. установка оборудования, производящего минимальный шум.

Также, любое оборудование при загрязнении увеличивает уровень шума, поэтому рекомендуется регулярно проводить его техническое обслуживание: чистка от пыли, замена смазывающих веществ.

6.3 Экологическая безопасность

Разрабатываемое программное обеспечение, система администрирования веб-платформы для игр и приложений, не оказывает влияния на окружающую среду, так как разрабатывается и используется внутри персональных компьютеров. Тем не менее, неправильная утилизация компьютеров, смартфонов, оргтехники и бумаги может стать источником загрязнения литосферы.

Утилизация компьютерной и организационной техники ограничена законодательно, так как в производстве такой техники используется большое количество материалов, способных нанести большой вред окружающей среде. Утилизация компьютерного оборудования происходит через обязательное извлечение компонент, их сортировку и последующую отправку для повторного использования. Такая утилизация обязательно производится на оборудованных полигонах с привлечением квалифицированного персонала.

Утилизация мусорных отходов, таких как бумажная макулатура, отходы от канцелярских принадлежностей, отходы от продуктов питания, личной гигиены, производится через сбор, обязательную сортировку и утилизацию. Отходы, которые можно использовать повторно, например, макулатуру, после сортировки отправляют на переработку через компании, занимающиеся сбором таких отходов.

Используя такую систему утилизации отходов работы, можно уменьшить свое воздействие на окружающую среду, а также на собственное здоровье, так как качественная утилизация отходов исключает отравление опасными веществами и попадание тяжелых металлов в организмы.

6.4 Безопасность в чрезвычайных ситуациях

Разработка системы администрирования веб-платформы для игр и приложений подразумевает постоянное использование электрического тока. При несоблюдении правил электробезопасности возможна такая чрезвычайная ситуация, как пожар.

При использовании электрооборудования причиной пожара является искра, источником которой является либо короткое замыкание, обусловленное использованием неисправного оборудования, либо нагрев участка электросети вследствие больших переходных сопротивлений или перегрузок.

Чтобы не допустить возникновения данной чрезвычайной ситуации, необходима периодическая, своевременная диагностика по обнаружению неисправностей, а также соблюдение персоналом норм пожарной безопасности.

Для минимизации последствий потенциального пожара рабочие помещения должны быть оборудованы средствами пожаротушения и пожарным оборудованием, готовым к эксплуатации. Кроме того, необходим инструктаж сотрудников по плану действий в случае возникновения чрезвычайной ситуации.

Вывод по разделу

В результате работы по разделу «Социальная ответственность» были выявлены основные нормативные акты для обеспечения безопасности жизнедеятельности на рабочем месте, рассмотрены наиболее значимые опасные и вредные факторы, возникающие при работе над системой администрирования веб-платформы для игр и приложений, описано влияние процесса разработки системы администрирования веб-платформы для игр и приложений на окружающую среду и меры, необходимые для уменьшения влияния вредных и опасных факторов на организм человека и для сокращения негативного влияния процесса разработки программного обеспечения на окружающую среду.

Заключение

В результате выполнения работы были исследованы популярные веб-платформы для игр и приложений и определены основные функции разрабатываемой системы. Это позволило получить широкое представление о предметной области и составить «use-case» диаграмму. Полученные знания были использованы для составления подробного описания функций системы администрирования и способов взаимодействия пользователей внутри нее.

Также была проведена работа по изучению существующих на рынке средств разработки и произведен выбор наиболее подходящих из них для данной работы.

В процессе реализации системы администрирования, были закреплены и углублены теоретические знания и практические навыки по проектированию баз данных, проектированию программных систем, работе с архитектурным шаблоном MVC и языком программирования JavaScript.

Основным результатом выполнения выпускной квалификационной работы стала созданная система администрирования веб-платформы для игр и приложений.

Список использованных источников

1. The Decade's Megatrends in Numbers – Part 2 // Synergy Research Group [Электронный ресурс]. 2020. URL: <https://www.srgresearch.com/articles/the-decades-megatrends-in-numbers-part-2> (дата обращения: 2.05.2020)
2. «В контакте» впервые сообщила, сколько платит разработчикам игр // «Ведомости» – деловое издание [Электронный ресурс]. 2020. URL: <https://www.vedomosti.ru/technology/articles/2018/12/25/790388-kontakte-igr> (дата обращения: 02.05.2020)
3. Системные требования для World of Tanks // World of Tanks [Электронный ресурс]. 2020. URL: <https://worldoftanks.ru/ru/content/docs/download/> (дата обращения: 02.05.2020)
4. Developer Survey Results 2018 // Stack Overflow Insights [Электронный ресурс]. 2020. URL: <https://insights.stackoverflow.com/survey/2018#technology> (дата обращения: 03.05.2020)
5. Developer Survey Results 2019 // Stack Overflow Insights [Электронный ресурс]. 2020. URL: <https://insights.stackoverflow.com/survey/2019> (дата обращения: 03.05.2020)
6. 12 Top Applications Written in Node.js // Netguru [Электронный ресурс]. 2020. URL: <https://www.netguru.com/blog/top-companies-used-nodejs-production> (дата обращения: 03.05.2020)
7. ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования // Электронный фонд правовой и нормативно-технической документации [Электронный ресурс]. 2020. URL: <http://docs.cntd.ru/document/1200003913> (дата обращения: 06.05.2020)
8. СанПиН 2.2.2/2.4.1340-03 Гигиенические требования к персональным электронно-вычислительным машинам и организации работы //

Электронный фонд правовой и нормативно-технической документации [Электронный ресурс]. 2020. URL: <http://docs.cntd.ru/document/901865498> (дата обращения: 06.05.2020)

9. СанПиН 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений // Электронный фонд правовой и нормативно-технической документации [Электронный ресурс]. 2020. URL: <http://docs.cntd.ru/document/901704046> (дата обращения: 06.05.2020)

10. СП 52.13330.2016 «Естественное и искусственное освещение» // Электронный фонд правовой и нормативно-технической документации [Электронный ресурс]. 2020. URL: <http://docs.cntd.ru/document/456054197> (дата обращения: 06.05.2020)

11. СН 2.2.4/2.1.8.562-96 «Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки» // Электронный фонд правовой и нормативно-технической документации [Электронный ресурс]. 2020. URL: <http://docs.cntd.ru/document/901703278> (дата обращения: 06.05.2020)

Приложение А

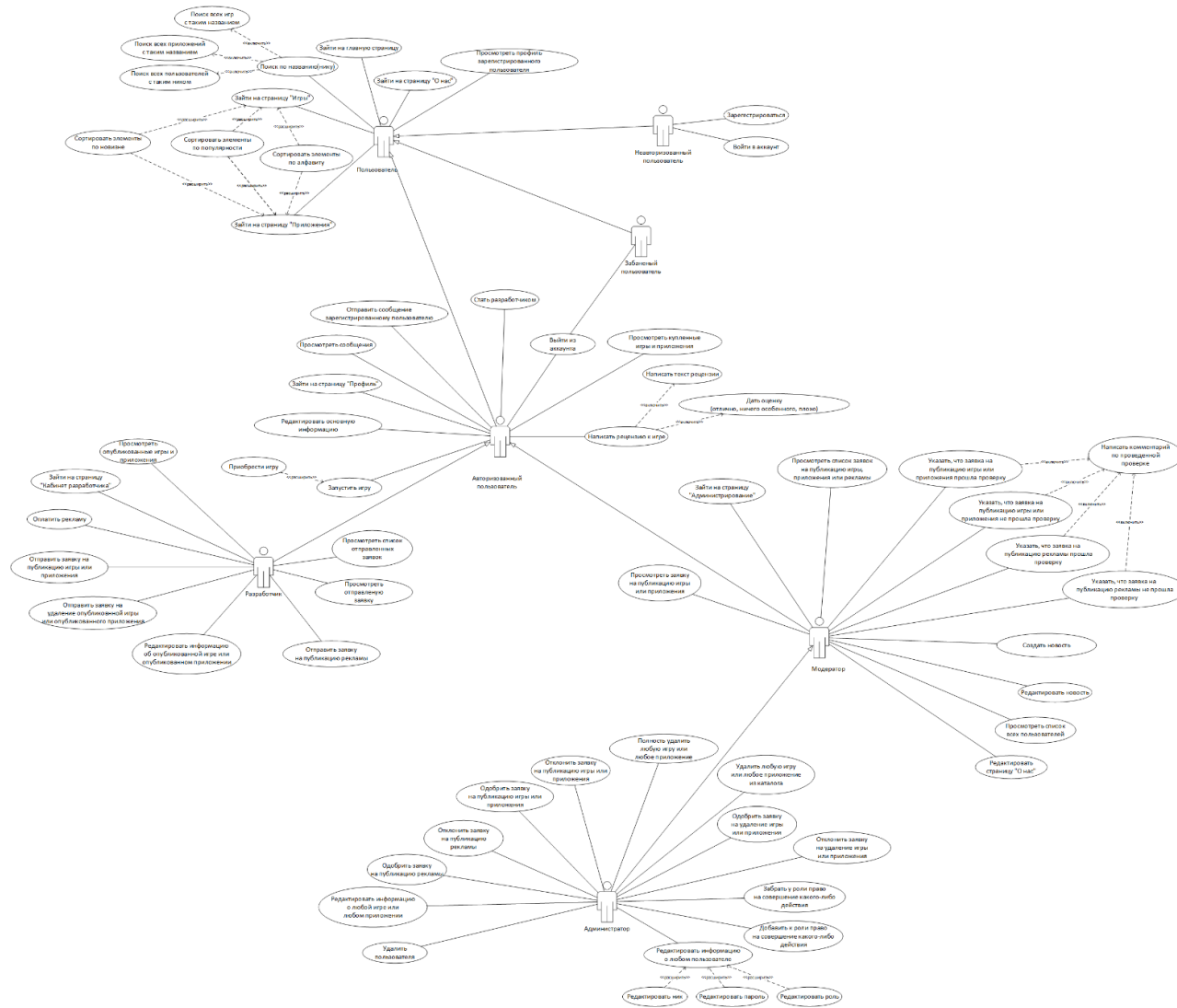


Рисунок А.1 – Use-case диаграмма для ролей пользователей

Приложение Б

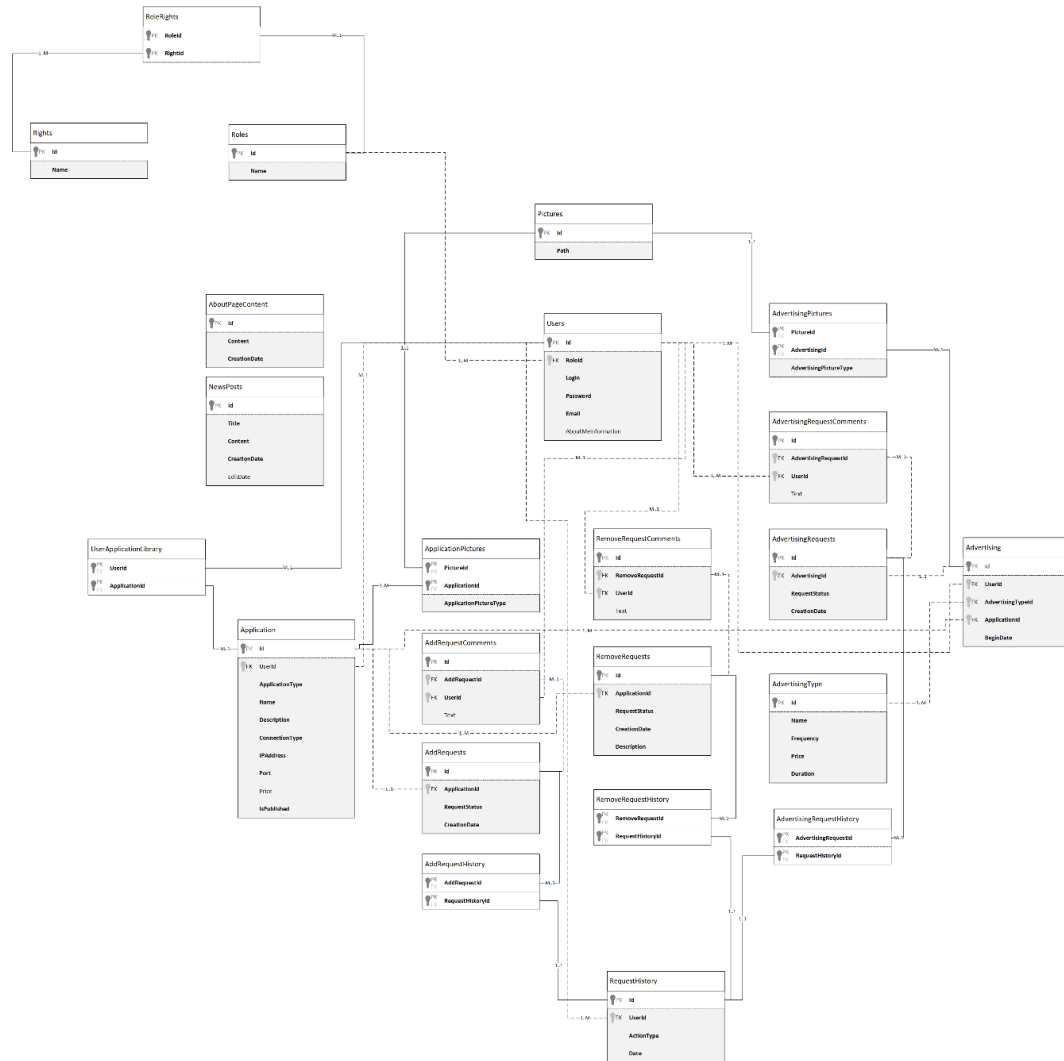


Рисунок Б.1 – Модель БД

Приложение В

Таблица В.1 – Описание таблицы базы данных

Название таблицы	Назначение
RoleRights	Сопоставление прав (возможностей действия) с ролью
Roles	Хранение названий и идентификаторов ролей
Rights	Хранение названий и идентификаторов прав доступа
Users	Хранение информации о пользователе, его правах
AboutPageContent	Хранение информации о странице веб-платформы «About»
NewsPosts	Хранение информации о новостях веб-платформы
Pictures	Хранение путей и идентификаторов используемых сайтом изображений
AdvertisingPictures	Связь между таблицами «Pictures» и «Advertising»
AdvertisingRequestComments	Хранение информации о комментарии модератора к запросу о добавлении рекламы
AdvertisingRequests	Хранение запросов на добавление рекламы

Продолжение

Advertising	Хранение информации о рекламе
AdvertisingType	Хранение типов рекламы
AdvertisingRequestHistory	Хранение истории запросов на добавление новостей (связь между таблицами «RequestHistory» и «AdvertisingRequests»)
RequestHistory	Хранение истории запросов добавления данных
RemoveRequestHistory	Хранение истории запросов удаления данных
RemoveRequests	Хранение запросов на удаление данных
RemoveRequestComments	Хранение комментариев к запросам удаления данных
ApplicationPictures	Связь между таблицами «Pictures» и «Application»
AddRequestComments	Хранение комментариев к запросам на добавление объектов
AddRequests	Хранение запросов на добавление объектов
AddRequestHistory	Хранение истории запросов на добавление объектов

Продолжение

UserApplicationLibrary	Связь между таблицами «User» и «Application» (приложения, доступные пользователю)
Application	Хранение информации о приложении

Приложение Г

Таблица Г.1 – Временные показатели проведения научного исследования

Название работы	Трудоёмкость работ						Исполнители		Длительность работ в рабочих днях T_{pi}		Длительность работ в календарных днях T_{ki}	
	t_{min} , чел-дни		t_{max} , чел-дни		$t_{ож}$, чел-дни							
	Исп. 1	Исп. 2	Исп. 1	Исп. 2	Исп. 1	Исп. 2	Исп. 1	Исп. 2	Исп. 1	Исп. 2	Исп. 1	Исп. 2
Выбор научно-руководителя бакалаврской работы	1	0	1	0	1	0	С.	Н.р.	0,5	0	0,74	0
Составление и утверждение темы бакалаврской работы	1	1	1	1	1	1	С.	Н.р.	0,5	0,5	0,74	0,74

Продолжение

Постановка целей и задач	1	1	1	1	1	1	С.	Н.р.	0,5	0,5	0,74	0,74
Подбор и изучение материалов по тематике	3	0	8	0	5	0	С.	Н.р.	5	0	7,4	0
Разработка календарного плана	1	1	1	1	1	1	С.	Н.р.	0,5	0,5	0,74	0,74
Проведение анализа предметной области	1	0	1	0	1	0	С.	Н.р.	0,5	0	0,74	0
Проектирование	2	2	3	3	2,4	2,4	С.	Н.р.	1,2	1,2	1,78	1,78
Разработка	7	0	14	0	9,8	0	С.	Н.р.	9,8	0	14,5	0
Тестирование	2	0	3	0	2,4	0	С.	Н.р.	2,4	0	3,55	0
Отладка	2	0	3	0	2,4	0	С.	Н.р.	2,4	0	3,55	0
Выполнение других частей работы (фин. менеджмент, соц. Ответственность)	3	0	4	0	3,4	0	С.	Н.р.	3,4	0	5,03	0

Продолжение

Подведение итогов, оформление работы	3	1	4	2	3,4	1,4	С.	Н.р.	3,4	1,4	5,03	2,07
Итого	Студент								30,1		44,54	
	Научный руководитель								4,1		6,07	
	Всего								35,2		52,09	

Приложение Д

Таблица Д.1 – Расчет основной заработной платы

Название работы	Трудоёмкость работы		Зарботная плата, приходящаяся на один чел.-день, руб.		Всего заработная плата по тарифу (окладам)	
	Студент	Научный руководитель	Студент	Научный руководитель	Студент	Научный руководитель
Выбор научного руководителя бакалаврской работы	1	0	900	1500	450	0
Составление и утверждение темы бакалаврской работы	1	1	900	1500	450	750
Постановка целей и задач	1	1	900	1500	450	750
Подбор и изучение материалов по тематике	5	0	900	1500	4500	0
Разработка календарного плана	1	1	900	1500	450	750

Продолжение

Проведение анализа предметной области	1	0	900	1500	450	0
Проектирование	2,4	2,4	900	1500	1080	1800
Разработка	9,8	0	900	1500	8820	0
Тестирование	2,4	0	900	1500	2160	0
Отладка	2,4	0	900	1500	2160	0
Выполнение других частей работы (фин. менеджмент, соц. Ответственность)	3,4	0	900	1500	3060	0
Подведение итогов, оформление работы	3,4	1,4	900	1500	3060	2100
Итого:					27090	6150

Приложение Е

Таблица Е.1 – Диаграмма Ганта

№ работ	Вид работ	Исполнитель	Т _{кi} , кал.дн.	Продолжительность выполнения работ						
				Февраль		Апрель		Май		Июнь
				2	3	1	2	3	1	
1	Выбор научного руководителя бакалаврской работы	Студент	0,74	█						
2	Составление и утверждение темы бакалаврской работы	Студент, научный руководитель	0,74	█ █						
3	Постановка целей и задач	Студент, научный руководитель	0,74		█ █					
4	Подбор и изучение материалов по тематике	Студент	7,4		█████████████████████					
5	Разработка календарного плана	Студент, научный руководитель	0,74			█ █				
6	Проведение анализа предметной области	Студент	0,74			█				

Продолжение

7	Проектирование	Студент, научный руководитель	1,78											
8	Разработка	Студент	14,5											
9	Тестирование	Студент	3,55											
10	Отладка	Студент	3,55											
11	Выполнение других частей работы (фин. менеджмент, соц. Ответственность)	Студент	5,03											
12	Подведение итогов, оформление работы	Студент, научный руководитель	С. – 5,03 Н. р. – 2,07											

Студент – зеленый цвет; Научный руководитель – фиолетовый цвет