



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 – 09.03.04 «Программная инженерия»  
Отделение школы (НОЦ) – Отделение информационных технологий

### ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Тема работы
<b>Разработка платформы платной подписки на контент</b>

УДК 004.774:379.825

Обучающийся

Группа	ФИО	Подпись	Дата
8К91	Семенов Олег Андреевич		

Руководитель ВКР

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

### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН УОД	Гасанов Магеррам Али оглы	д.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Мезенцева Ирина Леонидовна	–		

### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Чердынцев Евгений Сергеевич	к.т.н.		

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП/ОПОП

Код компетенции	Наименование компетенции
<b>Универсальные компетенции</b>	
УК(У)-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.
УК(У)-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений.
УК(У)-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде.
УК(У)-4	Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(-ых) языке(-ах).
УК(У)-5	Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах.
УК(У)-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни.
УК(У)-7	Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности.
УК(У)-8	Способен создавать и поддерживать безопасные условия жизнедеятельности, в том числе при возникновении чрезвычайных ситуаций.
УК(У)-9	Способен проявлять предприимчивость в практической деятельности, в т.ч. в рамках разработки коммерчески перспективного продукта на основе научно-технической идеи.
<b>Общепрофессиональные компетенции</b>	
ОПК(У)-1	Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности.
ОПК(У)-2	Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.
ОПК(У)-3	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.
ОПК(У)-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью.
ОПК(У)-5	Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем.
ОПК(У)-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения.
ОПК(У)-7	Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой.
ОПК(У)-8	Способен осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых

	технологий.
<b>Профессиональные компетенции</b>	
<b>ПК(У)-1</b>	Владение навыками разработки требований и проектирования программного обеспечения.
<b>ПК(У)-2</b>	Владение навыками разработки документов и стратегии тестирования программного обеспечения.
<b>ПК(У)-3</b>	Владение навыками моделирования, анализа и использования формальных методов конструирования программного обеспечения.
<b>ПК(У)-4</b>	Владение навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных.
<b>ПК(У)-5</b>	Владение концепциями и атрибутами качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли людей, процессов, методов, инструментов и технологий обеспечения качества.



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 – 09.03.04 «Программная инженерия»  
Отделение школы (НОЦ) – Отделение информационных технологий

УТВЕРЖДАЮ:

Руководитель ООП/ОПОП

\_\_\_\_\_ Чердынцев Е.С.  
(Подпись) (Дата) (ФИО)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

Обучающийся:

Группа	ФИО
8К91	Семенов Олег Андреевич

Тема работы:

Разработка платформы платной подписки на контент	
Утверждена приказом директора (дата, номер)	

Срок сдачи обучающимся выполненной работы:

--	--

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<p><b>Исходные данные к работе</b> (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 функционированию (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.)</p>	<p>Объектом проектирования в исследовательской работе является платформа платной подписки на контент, действующая на постоянной основе</p>
<p><b>Перечень разделов пояснительной записки подлежащих исследованию, проектированию и разработке</b> (аналитический обзор литературных источников 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)</p>	<ol style="list-style-type: none"> <li>1. Исследование платформ и сервисов платной подписки на контент</li> <li>2. Проектирование платформы платной подписки на контент</li> <li>3. Программная реализация</li> <li>4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</li> <li>5. Социальная ответственность</li> </ol>
<p><b>Перечень графического материала</b> (с точным указанием обязательных чертежей)</p>	<ol style="list-style-type: none"> <li>1. Диаграммы вариантов использования</li> <li>2. Диаграмма последовательности</li> <li>3. Диаграмма Исикавы</li> <li>4. Диаграмма компонентов</li> <li>5. Диаграмма развертывания</li> </ol>

	6. Блок-схема алгоритма 7. Диаграмма Ганта 8. Рисунки, демонстрирующие результаты работы
<b>Консультанты по разделам выпускной квалификационной работы</b> <i>(с указанием разделов)</i>	
<b>Раздел</b>	<b>Консультант</b>
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Гасанов Магеррам Али оглы
Социальная ответственность	Мезенцева Ирина Леонидовна

<b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b>	
---	--

**Задание выдал руководитель / консультант (при наличии):**

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

**Задание принял к исполнению обучающийся:**

Группа	ФИО	Подпись	Дата
8К91	Семенов Олег Андреевич		

Школа – Инженерная школа информационных технологий и робототехники  
 Направление подготовки – 09.03.04 «Программная инженерия»  
 Отделение школы (НОЦ) – Отделение информационных технологий  
 Период выполнения – весенний семестр 2023 учебного года

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН**  
**выполнения выпускной квалификационной работы**

Обучающийся:

Группа	ФИО
8К91	Семенов Олег Андреевич

Тема работы:

Разработка платформы платной подписки на контент
--

Срок сдачи обучающимся выполненной работы:

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
27.03.2023	Исследование платформ и сервисов платной подписки на контент	20
29.04.2023	Проектирование платформы платной подписки на контент	25
27.05.23	Программная реализация	25
02.05.2023	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
05.05.2023	Социальная ответственность	15

**СОСТАВИЛ:**

**Руководитель ВКР**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ ИШИТР	Саврасов Федор Витальевич	К.Т.Н.		

**СОГЛАСОВАНО:**

**Руководитель ООП/ОПОП**

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

**Обучающийся**

Группа	ФИО	Подпись	Дата
8К91	Семенов Олег Андреевич		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСООБЪЕДИНЕНИЕ И РЕСУРСОСБЕРЕЖЕНИЕ»

Студенту:

Группа	ФИО
8К91	Семенов Олег Андреевич

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

<b>Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:</b>	
1. Стоимость ресурсов научного исследования (НИ): <i>материально-технических, энергетических, финансовых, информационных и человеческих</i>	Оклад программиста – 40000
2. Нормы и нормативы расходования ресурсов	Премияльный коэффициент программиста 20%; Доплаты и надбавки программиста 30%; Коэффициент дополнительной заработной платы 12%; Накладные расходы 16%; Районный коэффициент 1,3.
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Коэффициент отчислений на уплату во внебюджетные фонды 30,2 %

<b>Перечень вопросов, подлежащих исследованию, проектированию и разработке:</b>	
1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	Определение потенциального потребителя результатов исследования, SWOT-анализ разработанной стратегии.
2. Планирование и формирование бюджета научных исследований	Определение структуры работы. Расчет трудоемкости выполнения работ. Подсчет бюджета исследования.
3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	Расчет показателей финансовой эффективности, ресурсоэффективности и эффективности исполнения.

<b>Перечень графического материала (с точным указанием обязательных чертежей):</b>	
1. Оценка конкурентоспособности технических решений	
2. Матрица SWOT	
3. График проведения и бюджета НИИ	
4. Оценка ресурсной, финансовой и экономической эффективности НИИ	

Дата выдачи задания для раздела по линейному графику	
--	--

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОСГН	Гасанов Магеррам Али оглы	д.э.н., профессор		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8К91	Семенов Олег Андреевич		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b> 8К91		<b>ФИО</b> Семенов Олег Андреевич	
<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение (НОЦ)</b>	<b>ОИТ</b>
<b>Уровень образования</b>	Бакалавриат	<b>Направление/специальность</b>	09.03.04 Программная инженерия

Тема ВКР:

<b>Разработка платформы платной подписки на контент</b>	
<b>Исходные данные к разделу «Социальная ответственность»:</b>	
<p><b>Введение</b></p> <ul style="list-style-type: none"> <li>– Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения.</li> <li>– Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации</li> </ul>	<p><i>Объект исследования:</i> онлайн-платформа, позволяющая пользователям создавать платные подписки и публиковать контент в них, а другим пользователям оформлять подписки и получать доступ к контенту</p> <p><i>Область применения:</i> авторы контента и их подписчики</p> <p><i>Рабочая зона:</i> офис</p> <p><i>Размеры помещения:</i> 4*7 м.</p> <p><i>Количество и наименование оборудования рабочей зоны:</i> ноутбук 1 шт.</p> <p><i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне:</i> Поиск и анализ информации, проектирование и разработка платформы</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p><b>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<p>ГОСТ 12.2.032-78. «Система стандартов безопасности труда». Рабочее место при выполнении работ сидя»; «Трудовой кодекс Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 19.12.2022, с изм. от 11.04.2023); ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения».</p>
<p><b>2. Производственная безопасность при разработке проектного решения:</b></p> <ul style="list-style-type: none"> <li>– Анализ выявленных вредных и опасных производственных факторов</li> </ul>	<p><b>Вредные факторы:</b></p> <ol style="list-style-type: none"> <li>1. Умственное перенапряжение, в том числе вызванное информационной нагрузкой;</li> <li>2. Эмоциональное перенапряжение;</li> <li>3. Отсутствие или недостаток необходимого искусственного освещения;</li> <li>4. Статические физические перегрузки, связанные с рабочей позой.</li> </ol> <p><b>Опасные факторы:</b></p> <ol style="list-style-type: none"> <li>1. Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий.</li> </ol>

	<b>Требуемые средства индивидуальной и коллективной защиты от выявленных факторов:</b> удобное рабочее место, устройства защитного отключения, изоляционные средства для электрооборудования, приборы искусственного освещения.
<b>3. Экологическая безопасность при разработке проектного решения</b>	Воздействие на селитебную зону, литосферу, гидросферу, атмосферу: неверный способ утилизации ПЭВМ, оргтехники, макулатуры и бытовых отходов.
<b>4. Безопасность в чрезвычайных ситуациях при разработке проектного решения</b>	<b>Возможные ЧС:</b> Природные катастрофы (наводнения, ураганы, лесные пожары и т.д.); Геологические воздействия (землетрясения, провалы территории и т.д.); Техногенные (пожары, аварии с выбросом химически/радиоактивно опасных веществ и т.д.); Биолого-социальные (пандемия) <b>Наиболее типичная ЧС:</b> пожар
<b>Дата выдачи задания для раздела по линейному графику</b>	

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна			

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8К91	Семенов Олег Андреевич		

## Реферат

Выпускная квалификационная работа содержит 97 страниц, 41 рисунок, 11 таблиц, 37 источников и 1 приложение.

Ключевые слова: веб-платформа, контент, автор, подписка, доступ, оплата, Django.

Объектом исследования является монетизации пользовательского контента с использованием специальных веб-платформ и сервисов.

Цель работы – проектирование и разработка платформы платной подписки на контент.

В ходе работы проводились исследования таких платформ и сервисов как Patreon, OnlyFans, Boosty, Sponsr, ЮКасса, Tinkoff, PayPal и Stripe.

В результате была спроектирована веб-платформа, которая позволяет монетизировать авторский контент по подписочной системе, с возможностью дополнительной настройки доступа внутри подписок.

Основные технологические характеристики: выполнено с использованием фреймворка Django и платежной системы Stripe.

Степень внедрения: в процессе.

Область применения: рынок пользовательского контента.

Экономическая значимость работы: разработанный проект позволит упростить процесс монетизации авторского контента.

В будущем планируется реализовать функции комментариев, «лайков» и личных сообщений.

## Содержание

Список терминов, сокращений и условных обозначений .....	15
Введение .....	16
1. Исследование платформ и сервисов платной подписки на авторский контент.....	18
1.1. Общая информация о подписной модели монетизации контента ....	18
1.2. Определение потребностей стейкхолдеров .....	18
1.3. Обзор конкурентных платформ .....	20
1.4. Результаты исследования предметной области.....	24
2. Проектирование платформы платной подписки на контент .....	26
2.1. Проектирование функций .....	26
2.1.1. Роли и варианты использования .....	26
2.1.2. Проектирование последовательности .....	28
2.2. Проектирование базы данных .....	29
2.3. Выбор инструментов для разработки .....	31
2.3.1. Выбор языка разработки.....	31
2.3.2. Выбор фреймворка .....	32
2.3.3. Выбор СУБД.....	34
2.3.4. Выбор среды разработки .....	35
2.3.5. Выбор платежной системы.....	36
2.4. Проектирование архитектуры платформы.....	37
2.4.1. Особенности архитектуры Django проекта.....	37
2.4.2. Выявление и организация компонентов платформы .....	38
2.4.3. Развертывание платформы .....	39
2.5. Результаты проектирования .....	40
3. Программная реализация онлайн-платформы .....	41

3.1. Файловая структура.....	41
3.2. Работа с БД.....	42
3.3. Реализация бизнес-логики .....	45
3.3.1. Реализация публикаций .....	46
3.3.2. Реализация подписок.....	48
3.3.3. Реализация настройки доступа к устаревшему контенту .....	49
3.4. Работа со Stripe .....	53
3.4.1. Оплата публикаций .....	53
3.4.2. Оплата подписок.....	56
3.4.3. Вывод средств.....	58
3.5. Обзор основных результатов разработки веб-платформы .....	59
3.5.1. Поиск авторов .....	59
3.5.2. Страница автора.....	60
3.5.3. Регистрация и авторизация.....	61
3.5.4. Панель администратора .....	64
4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение..	65
4.1. Введение .....	65
4.2. Оценка коммерческого потенциала и перспективности разработки	66
4.2.1. Потенциальные потребители результатов исследования .....	66
4.2.2. Анализ конкурентных технических решений.....	66
4.2.3. SWOT-анализ .....	67
4.3. Планирование работ по научно-техническому исследованию .....	69
4.3.1. Структура работ в рамках научного исследования.....	69
4.3.2. Определение трудоемкости выполнения работ.....	69
4.3.2.1. Разработка графика проведения научного исследования.....	70
4.4. Бюджет научно-технического исследования .....	72
4.4.1. Расчет материальных затрат НТИ.....	72

4.4.2. Расчет затрат на специальное оборудование для научных работ .....	72
4.4.3. Основная заработная плата исполнителя.....	72
4.4.4. Расчет дополнительной заработной платы.....	73
4.4.5. Отчисления во внебюджетные фонды.....	74
4.4.6. Накладные расходы.....	74
4.4.7. Формирование бюджета затрат научно-исследовательского проекта.....	75
4.5. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....	75
4.6. Вывод по разделу.....	78
5. Социальная ответственность.....	79
5.1. Введение.....	79
5.2. Правовые и организационные вопросы обеспечения безопасности	80
5.2.1. Правовые нормы трудового законодательства.....	80
5.2.2. Эргономические требования к правильному расположению и компоновке рабочей зоны.....	81
5.3. Производственная безопасность.....	82
5.3.1. Умственное перенапряжение, в том числе вызванное информационной нагрузкой и эмоциональное перенапряжение.....	83
5.3.2. Отсутствие или недостаток необходимого искусственного освещения.....	85
5.3.3. Статические физические перегрузки, связанные с рабочей позой .....	86
5.3.4. Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий.....	86
5.4. Экологическая безопасность.....	88

5.5. Безопасность в чрезвычайных ситуациях .....	88
5.6. Выводы по разделу .....	89
Заключение.....	91
Список использованных источников.....	92
Приложение А. Диаграмма вариантов использования «Пользователь» .....	97

## **Список терминов, сокращений и условных обозначений**

Онлайн-платформа (интернет-платформа, веб-платформа) – цифровой сервис, который облегчает взаимодействие между двумя или более отдельными взаимозависимыми группами пользователей (фирмы или отдельные лица), которые взаимодействуют через сервис посредством Интернета.

API (Application Programming Interface) – описание методов, с помощью которых одна компьютерная программа может взаимодействовать с другой.

HTTP (HyperText Transfer Protocol) – протокол прикладного уровня передачи данных.

HTTPS – расширение протокола HTTP для поддержки шифрования в целях повышения безопасности.

Рекуррентный платеж (автоплатеж) – регулярные и периодические списания определённой суммы с банковской карты без подтверждения её владельца (оплата по подписке).

SMTP (Simple Mail Transfer Protocol) – самый распространённый протокол передачи почты.

## **Введение**

В настоящее время число количество пользователей Интернета является очень большим. На начало 2023 года оно составляет 64,4% от общей численности населения всего мира, это более 5 миллиардов человек, а за 2022 год рост составил 1,9%. Вместе с этим за последние несколько лет возросло время, которое люди проводят в Интернете (в России это почти около 8 часов в сутки).

Согласно исследованиям британской компании GWI, больше 50% пользователей трудоспособного возраста посещают Интернет для поиска какой-либо информации [1]. Всю информацию в Интернете, в большинстве случаев, создают и формируют люди. Это могут быть как целые издания с большим штатом сотрудников и бюджетом, так и отдельные авторы, которые создают так называемый пользовательский контент.

Рынок пользовательского контента – это динамичная и быстрорастущая индустрия, которая предоставляет возможность для создания и распространения контента пользователями, что делает ее достаточно привлекательной для креативных людей, которые хотят делиться своими идеями и творчеством со всем миром. Площадками для этого выступают такие платформы, как социальные сети, форумы, блоги, видеохостинги, и так далее.

Однако, при всех положительных сторонах рынка пользовательского контента, он также имеет и свои недостатки. Проблемы некачественного, недостоверного и неприемлемого материала затрудняют развитие индустрии.

Если последнюю из обозначенных проблем можно частично нивелировать с помощью модерации платформ, на которых распространяется контент, то при решении остальных могут возникать сложности, так как они зависят напрямую от мотивации авторов. Многие из них создают контент, опираясь исключительно на свой энтузиазм и увлечение в интересующей их

теме, а постоянное производство качественного и актуального контента требует траты множества временных и денежных ресурсов.

Эту потребность закрывают специальные платформы, на которых авторы получают доход в зависимости от количества просмотров их контента или же напрямую через пожертвования от людей, которые решили их поддержать [2]. Также существуют сервисы, которые дают авторам возможность предоставлять доступ к своему творчеству через подписочную систему, что повышает уникальность работ авторов [3, 4].

За 2021 год российские авторы контента смогли заработать от 16 до 18 миллиардов рублей, что практически в два раза больше чем в 2019 году, и на 50-70% превышает доход за 2020 год [3].

Однако, в связи со сложившейся в настоящее время ситуацией в мире, большинство зарубежных платформ, позволяющих авторам получать деньги за свой труд, перестали работать с пользователями из России, или же были полностью заблокированы на территории РФ, вследствие чего доходы российских авторов значительно упали.

Конечно, на российском рынке также есть аналогичные сервисы, но их количество и популярность гораздо меньше, также они обладают ограниченным набором функций и не покрывают всех потребностей как авторов, так и обычных пользователей.

Таким образом, разработка собственной альтернативной платформы платной подписки на контент является актуальной задачей, подлежащей реализации в рамках данной работы.

# **1. Исследование платформ и сервисов платной подписки на авторский контент**

## **1.1. Общая информация о подписной модели монетизации контента**

Платформы подписки на авторский контент позволяют авторам монетизировать свое творчество, предлагая эксклюзивный доступ к своим работам по модели платной подписки. Авторы, создавая свои подписки и наполняя их каким-либо контентом, могут получать периодические платежи от своих читателей, оформивших эти подписки. Это дает создателям ряд преимуществ, включая повышенный контроль над своей работой, более тесные отношения со своей аудиторией, и относительно устойчивый источник дохода, что позволяет сохранять некоторую свободу творчества и независимость от рекламодателей. Популярность авторов зависит от них самих, в их обязанности входит продумывание системы подписок и их наполнения, чтобы удерживать читателей, оформивших подписку и привлекать новых подписчиков [3, 4, 5].

Теоретически, автором может выступать любой человек. Подписки авторов могут содержать в себе контент, который нигде более не распространяется, например, ранний доступ к видео или фото, эксклюзивные инструкции, советы, справочные или учебные материалы, и многое другое. Такой контент может быть мультимедийным, при этом важно, чтобы он не нарушал правила платформы и действующее законодательство.

## **1.2. Определение потребностей стейкхолдеров**

Самыми очевидными и заинтересованными сторонами в данной теме являются авторы и подписчики, поэтому, для их привлечения, платформа должна удовлетворять их потребностям, предоставляя следующие возможности:

1. Создание контента – платформа должна предоставлять авторам несложный в использовании интерфейс для публикации их контента и дальнейшего управления им. Это может включать инструменты для написания и форматирования текстовых статей, загрузки изображений и видео- или аудио-файлов, а также разбиения публикаций по категориям.

2. Создание подписок – платформа должна предоставлять авторам систему создания и настройки подписок, которая позволит устанавливать цены и регулировать доступ к контенту внутри подписок.

3. Общение – платформа должна предоставлять возможность авторам и читателям общаться друг с другом и давать обратную связь. Функционально это может быть реализовано с помощью комментариев, «лайков» и личных сообщений.

4. Отчетность и аналитика – платформа должна предоставлять возможность авторам отслеживать статистику своей страницы и доходов с помощью инструментов аналитики и создания отчетности.

5. Кастомизация – платформа должна позволять создателям контента настраивать свои страницы таким образом, чтобы они отражали их индивидуальный стиль.

6. Кроссплатформенность – платформа должна быть доступной для использования на различных устройствах, чтобы обслуживать пользователей, которые предпочитают получать доступ к контенту на своих смартфонах или планшетах.

7. Безопасность и конфиденциальность – платформа должна уделять приоритетное внимание функциям безопасности и конфиденциальности, таким как безопасная обработка платежей, шифрование данных и двухфакторная аутентификация, чтобы защитить как создателей контента, так и подписчиков.

В целом, реализация платформы платной подписки требует для создателей контента баланса между предоставлением необходимых инструментов для создания контента и управления подпиской, при одновременной приоритизации функций сообщества, аналитики, настройки, быстрого реагирования на различные устройства, безопасности и конфиденциальности. Используя эти элементы, создатели контента могут монетизировать свою работу, одновременно создавая лояльную аудиторию подписчиков.

### **1.3. Обзор конкурентных платформ**

На данный момент существует достаточное количество сервисов, позволяющих пользователям публиковать и монетизировать свой контент. Это могут быть как полноценные платформы, которые осуществляют работу исключительно в этом направлении, так и большие сервисы по типу социальных сетей или мессенджеров, предоставляющие пользователям эти функции, как дополнительные возможности. В обзоре не рассматриваются вопросы, касающиеся непосредственно самой монетизации, то есть возможные способы ввода и вывода средств и комиссии, взимаемые при этом платформами, так как это требует дополнительных маркетинговых исследований, выходящих за рамки решаемой проблемы.

Наиболее популярными сервисами, предоставляющими авторам функцию платной подписки на их контент, являются зарубежные Patreon [6] и OnlyFans [7], а также отечественные Boosty [8] и Sponsr [9].

#### **1. Patreon**

Запущенный в 2013 году сервис является одним из самых развитых и популярных (более 250 тысяч авторов) среди других аналогов [10]. Пользователям предоставляется большое количество функций и возможность подключать множество сторонних сервисов. В данный момент сервис заблокирован в России.

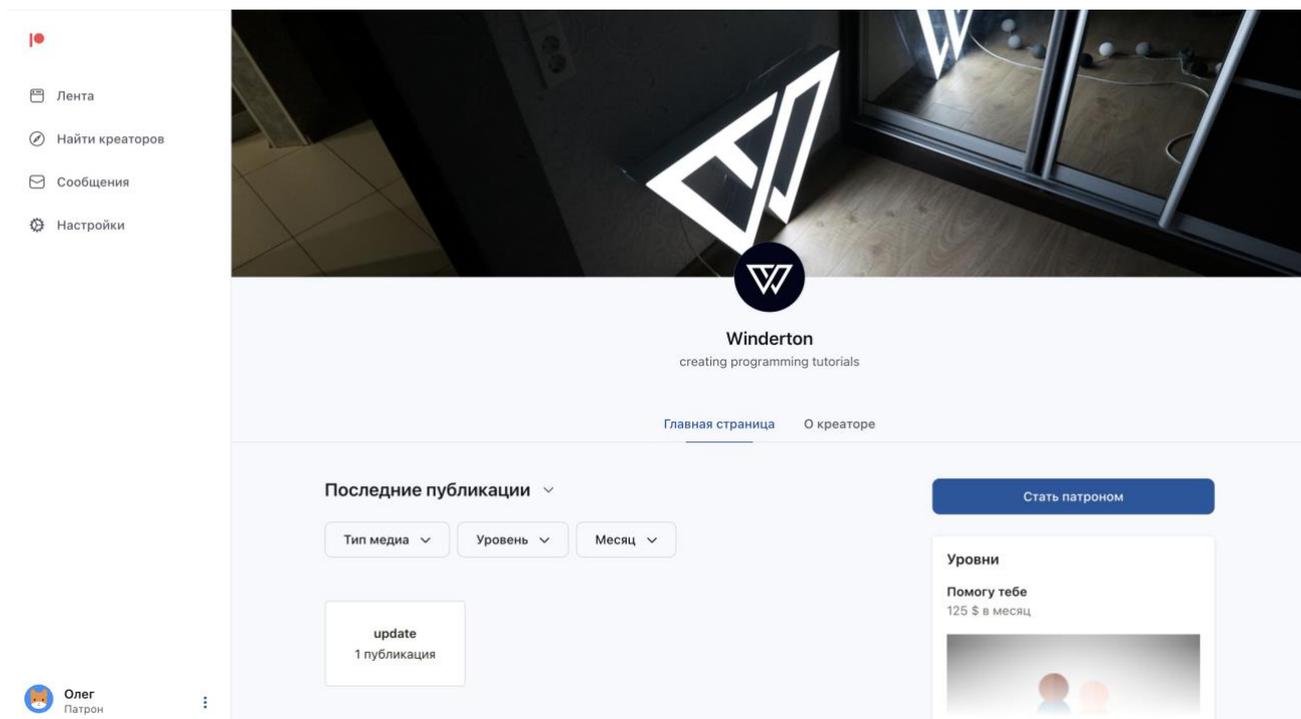


Рисунок 1 – Пример интерфейса страницы автора на Patreon

## 2. OnlyFans

Аналогичный, разработанный в 2015 году и имеющий еще большую популярность, сервис. На данный момент насчитывает более 170 миллионов зарегистрированных пользователей, из которых более 1,5 миллиона являются авторами [11]. OnlyFans получил свою известность из-за большого количества авторов, которые выкладывают на платформу контент для взрослых [4]. Имеет настраиваемую ленту постов, в которой располагается как контент от авторов, на которых вы подписались, так и контент, который предлагает пользователям сама платформа. Также есть список наиболее интересных авторов, но поиск по авторам не реализован. На данный момент сервис отключил возможности оплаты и вывода средств для пользователей из России [12].

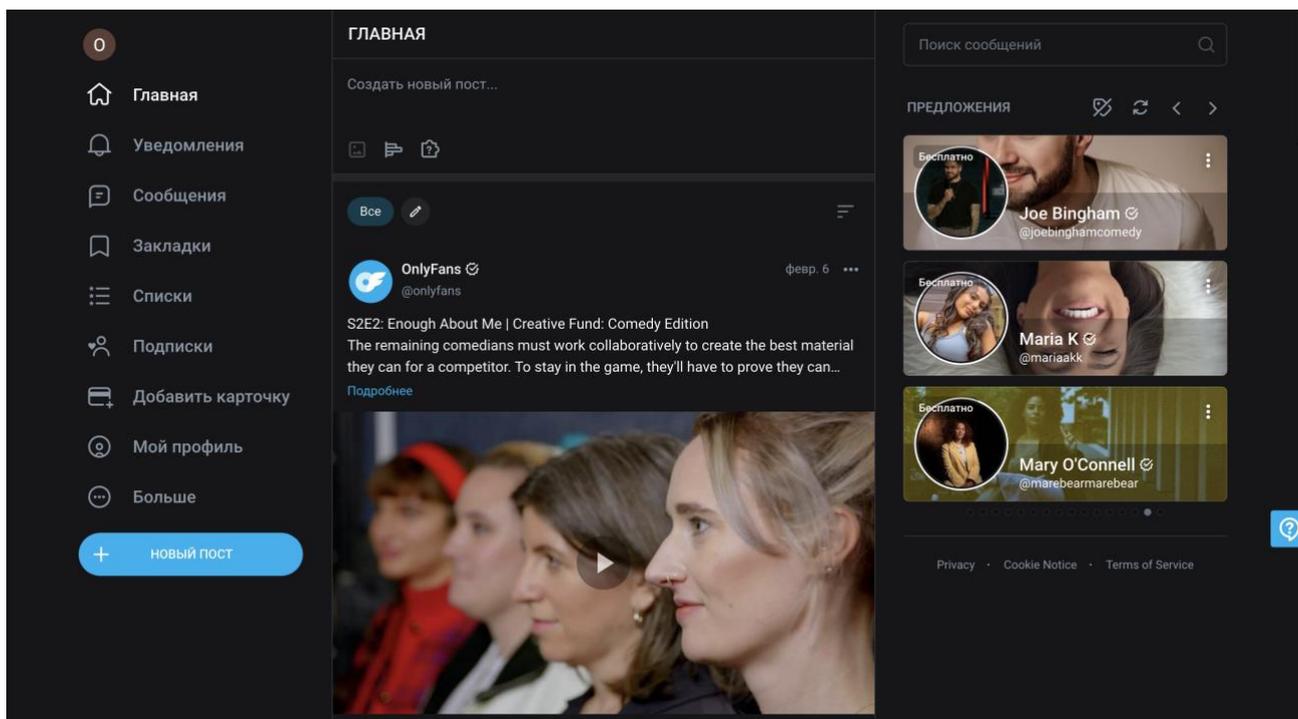


Рисунок 2 – Пример интерфейса главной страницы на OnlyFans

### 3. Boosty

Этот сервис начал свою работу в 2019 году и признан главным отечественным аналогом вышеупомянутых платформ [4, 13]. Популярность принесла Boosty в марте 2022 года прирост новых пользователей на 432% больше по сравнению с прошлым годом [5]. Главными недостатками можно выделить отсутствие таких функций, как поиск авторов на платформе, и настройка доступа к контенту в подписках.

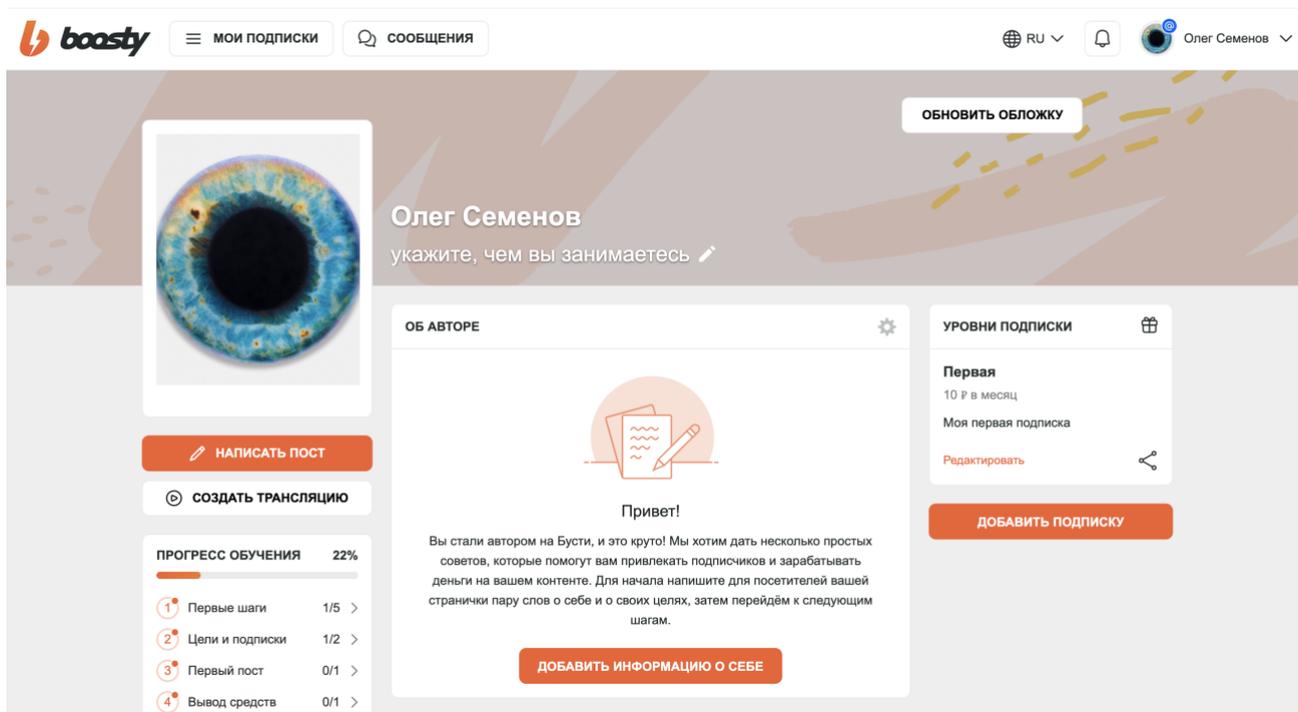


Рисунок 3 – Пример интерфейса личной страницы автора на Boosty

#### 4. Sponsr

Еще одна платформа российского происхождения с тем же основным принципом работы и также запущенная в 2019 году, но имеющая меньшую популярность. Она позиционируется как сервис только для зрелых авторов с интеллектуальным контентом, имеющих соответствующую аудиторию [5]. Вести авторскую деятельность здесь можно только по приглашению от другого автора или от самой команды платформы. На Sponsr, в отличие от других платформ, реализован поиск постов и подписок через специальную строку поиска, а также расширенные настройки доступа к контенту внутри подписки. Автор может выбрать вариант, чтобы все прошлые публикации были доступны для всех новых пользователей, или же ограничить доступ и выставить условия, что новые публикации доступны только пользователям, имеющим активную подписку в момент совершения публикации, или же чтобы все прошлые публикации становились доступны через указанный автором срок непрерывной подписки.



Рисунок 4 – Пример интерфейса страницы автора на Sponsr

Все рассматриваемые платформы имеют одинаковый принцип работы и набор основных функций, но у каждой из них есть отличительные черты.

Так как упомянутые зарубежные платформы полностью или частично ограничены для пользователей из России, далее они рассматриваться не будут.

Обзор конкурентов показал, что отечественные решения, существующие уже несколько лет, имеют большую аудиторию, а также обширный ряд функций. Однако, на рассматриваемых сервисах, он либо недостаточный, либо находится в ограниченном доступе. Поэтому, для того чтобы эффективно конкурировать с аналогами, необходимо реализовать в разрабатываемой платформе дополнительные возможности, а именно функции поиска авторов и гибкой настройки доступа к контенту внутри подписок.

#### 1.4. Результаты исследования предметной области

В результате исследования платформ и сервисов платной подписки на контент, для лучшего понимания и визуализации основной проблемы и ее

источников, была сформирована диаграмма Fishbone (диаграмма Исикавы) [14], представленная на рисунке 5. В качестве главной проблемы выделена необходимость монетизации контента авторов, к которой ведут четыре группы ее источников, в каждую группу включены отдельные конкретные источники.

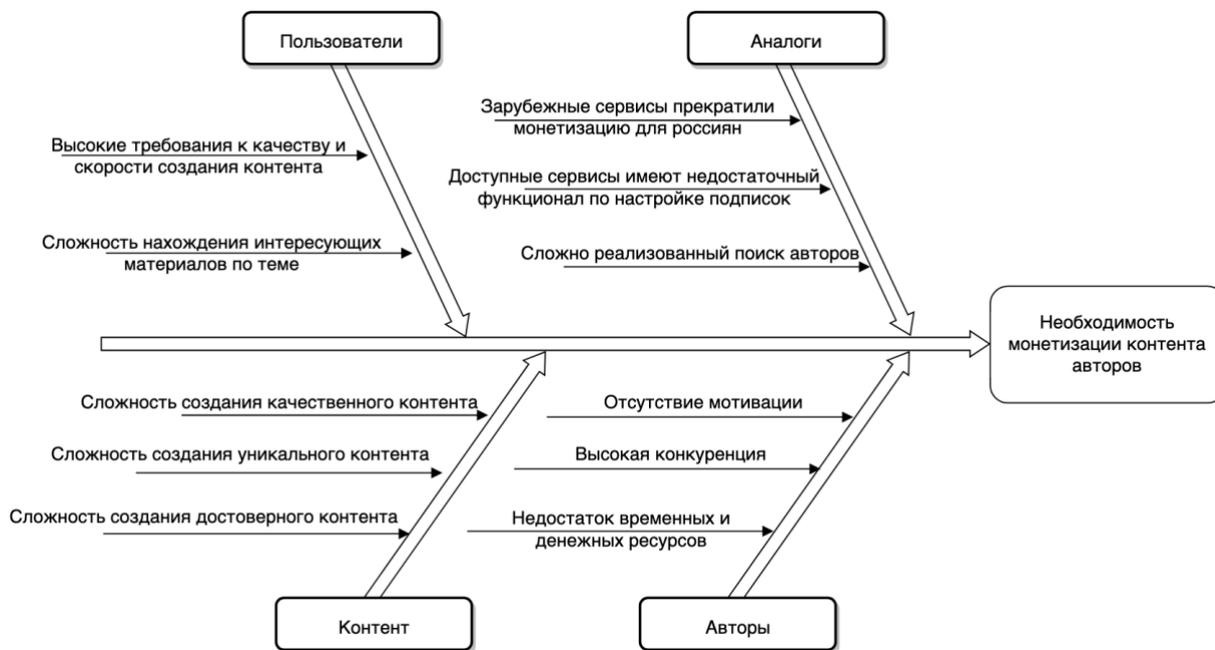


Рисунок 5 – Диаграмма Fishbone

## **2. Проектирование платформы платной подписки на контент**

### **2.1. Проектирование функций**

#### **2.1.1. Роли и варианты использования**

В ходе проектирования были определены следующие роли: неавторизованный пользователь, пользователь (он же подписчик), автор, и администратор. Для всех ролей были построены UML-диаграммы вариантов использования [15]. Диаграмма вариантов использования для роли пользователя представлена в приложении А.

Подписчик — это любой зарегистрированный пользователь на платформе. Большинство его действий возможно осуществить только при условии, что авторы на своей странице создали возможность для этого, например, выложили пост, который можно комментировать, создали подписку, и т.д. Пользователь в любой момент может стать автором, для этого ему необходимо совершить некоторые действия по настройке своего аккаунта. Если пользователь стал автором, то все прежние возможности у него остаются, и добавляются новые возможности, показанные на рисунке б.

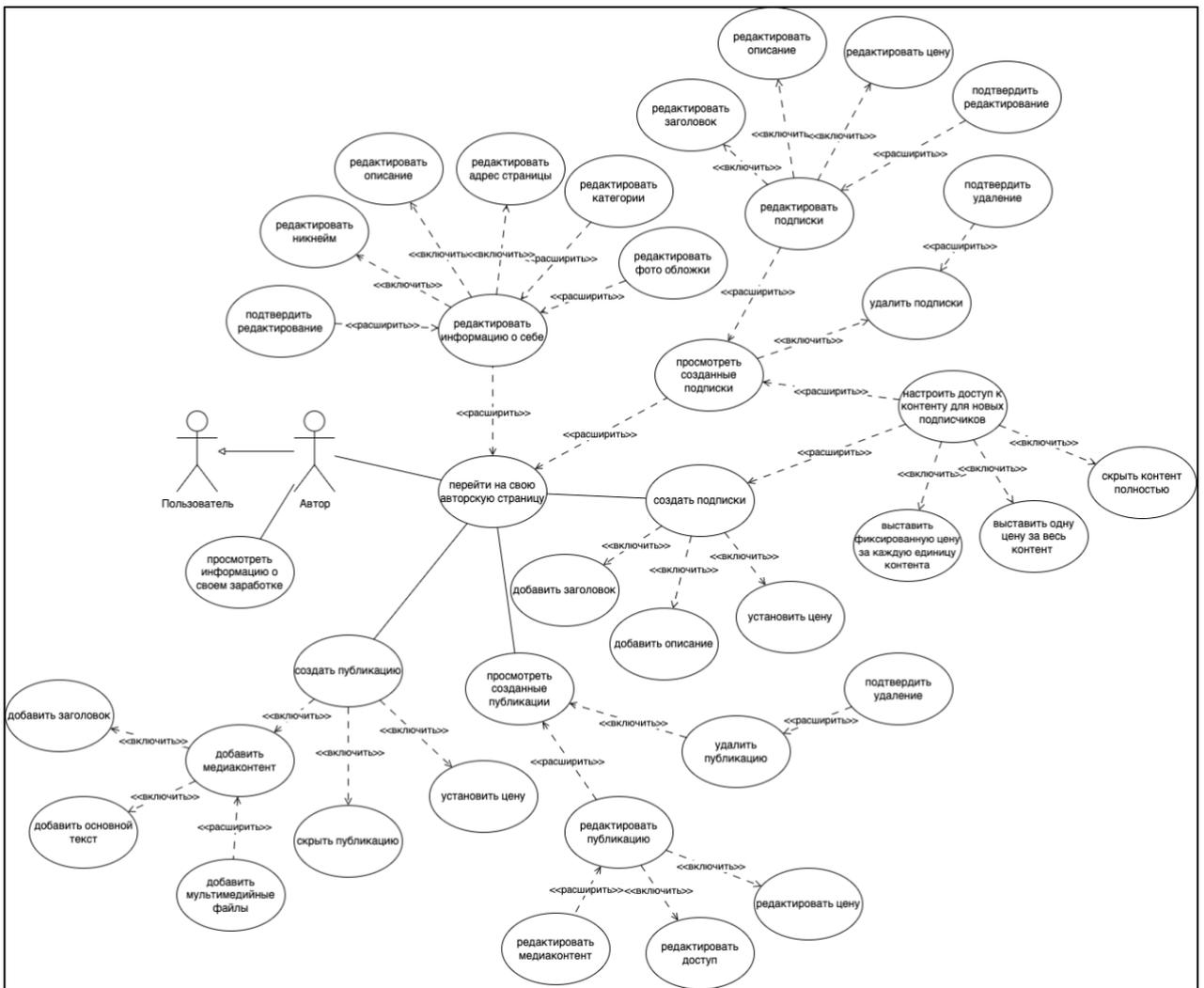


Рисунок 6 – Диаграмма вариантов использования «Автор»

Пользователь, обладающий правами администратора, создается во время разработки проекта Django, там же ему задаются объекты, которыми он будет управлять, и набор допустимых действий с этими объектами. Эти действия отображены на диаграмме (рисунок 7).



Рисунок 7 – Диаграмма вариантов использования «Администратор»

### 2.1.2. Проектирование последовательности

В связи с большим количеством вариантов использования, было принято решение построить (для демонстрации работы функций) UML-диаграмму последовательности, которая описывает одну из важных функций платформы «Оформление платной подписки» и затрагивает наибольшее

количество элементов [16]. Готовая диаграмма представлена на рисунке ниже:

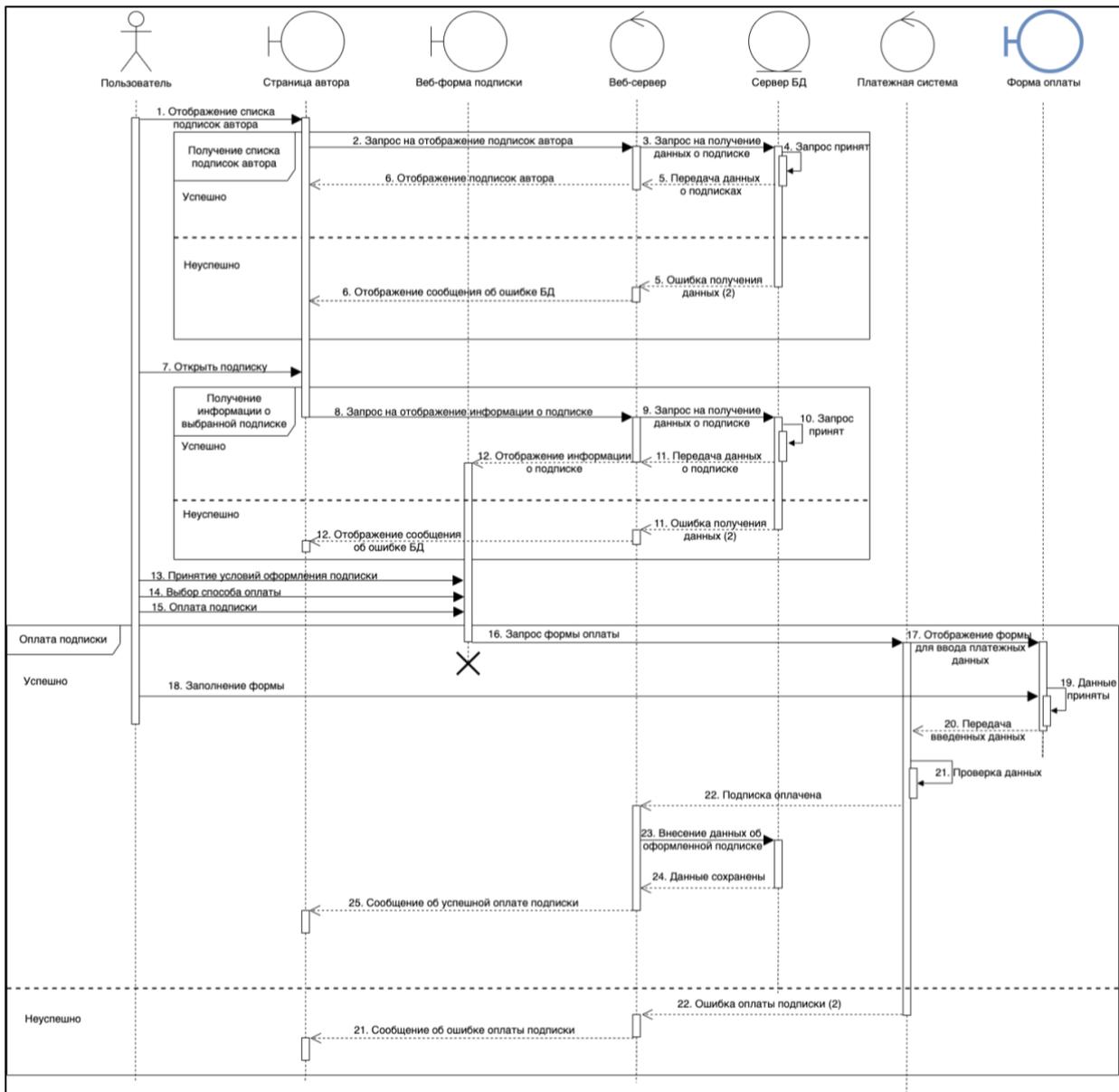


Рисунок 8 – Диаграмма последовательности «Оформление платной подписки»

## 2.2. Проектирование базы данных

Проектирование базы данных осуществлялось с использованием нотации «Crow's Foot» («вороньи лапки») [17]. Готовая концептуальная модель представлена на рисунке 9.

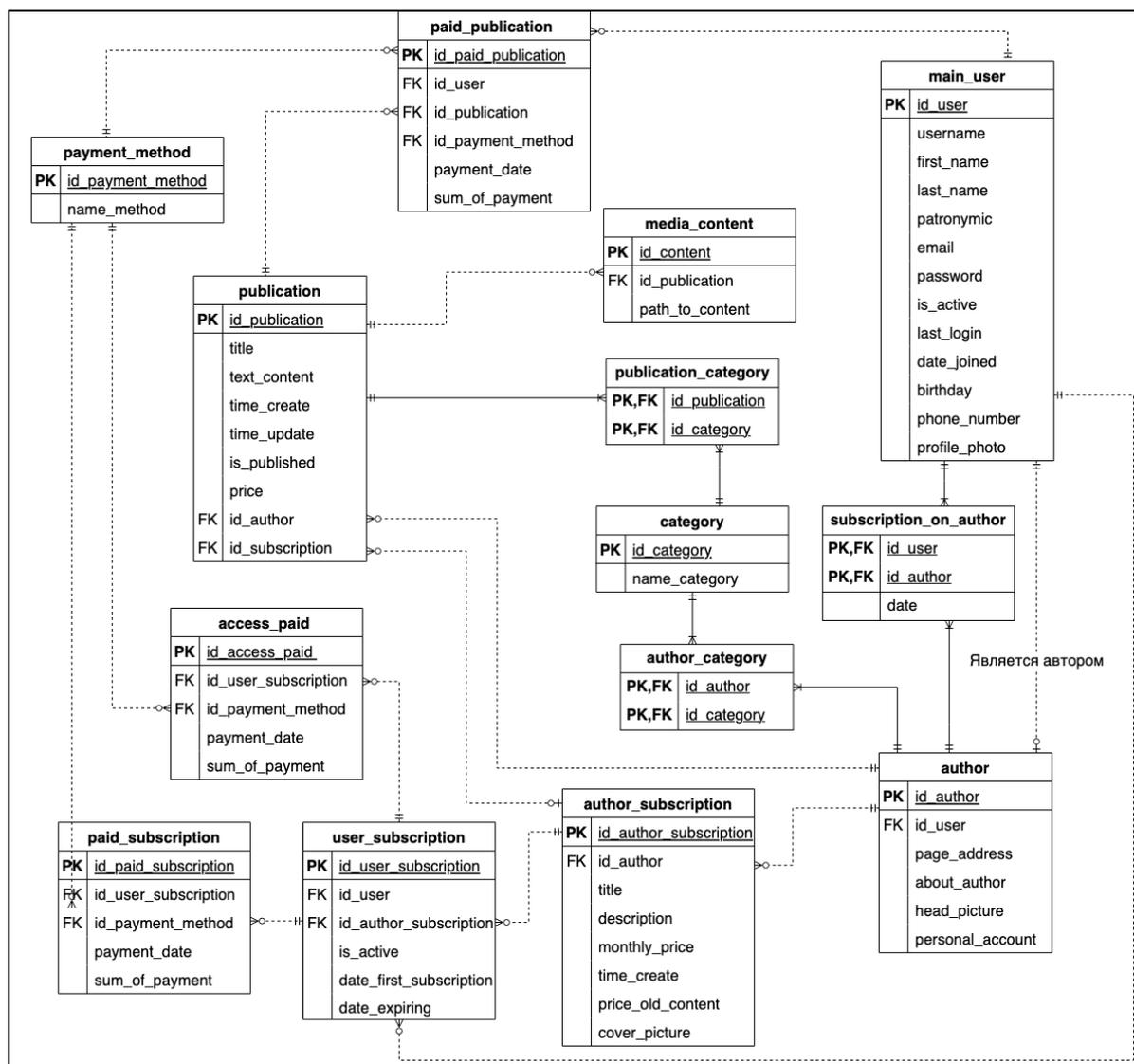


Рисунок 9 – Концептуальная модель базы данных

Главной является сущность «main\_user». Она хранит в себе данные, необходимые для регистрации и авторизации пользователей, а также некоторую дополнительную информацию о них.

Любой пользователь может стать автором, информация об этом будет храниться в сущности «author».

Таблица «subscription\_on\_author» необходима для реализации функции подписки пользователя на обновления автора.

В таблице «category» хранятся категории, которые необходимы для распределения авторов и публикаций. Чтобы автор и публикации могли

иметь более одной категории, между этими сущностями добавлены дополнительные таблицы для избегания связи «многие ко многим».

Автор может создавать публикации (сущность «publication»), которые будут располагаться на его личной странице либо внутри подписок. Публикация может содержать в себе медиа-контент, путь к которому указан в таблице «media\_content».

Если поле «price» в сущности публикации заполнено, доступ к публикации для пользователей доступен только после одноразовой оплаты за эту публикацию, данное условие проверяется в таблице «paid\_publications».

Автор может создавать подписки (сущность «author\_subscription»). Подписки содержат в себе публикации. Для доступа к ним пользователи должны оформлять подписку (сущность «user\_subscription»), то есть регулярно оплачивать ее. Имеет ли пользователь доступ к данной подписке, проверяется через таблицу «user\_subscription».

Автор может закрывать доступ к «старому» контенту в подписке для новых подписчиков. Если выставлена цена «price\_old\_content», то новому подписчику закрыт доступ к публикациям, если он не оплатил доступ к ним дополнительно (сущность «access\_paid»).

Все таблицы, связанные с оплатой, имеют связь с сущностью «payment\_method», которая хранит в себе методы оплаты.

## **2.3. Выбор инструментов для разработки**

### **2.3.1. Выбор языка разработки**

Для разработки платформы был выбран язык программирования Python по следующему ряду причин:

1. Простота освоения и использования: данный язык имеет относительно простой синтаксис, подробную документацию и большое сообщество. Все это в совокупности экономит время на написание кода и разбор сложных вопросов, а также дает возможность писать более чистый и понятный для чтения код.
2. Большой выбор библиотек: для Python написано большое количество различных библиотек, которые упрощают разработку больших проектов. Непосредственно для веб-разработки используются продвинутые фреймворки, такие как Django, Flask и Pyramid.
3. Масштабируемость: Python имеет высокую степень масштабируемости, что в свою очередь позволяет обрабатывать большие объемы данных и трафика. Такая особенность дает ему преимущество при разработке интернет-платформ, так как при росте пользователей и функционала растет и количество обрабатываемых данных, что приводит к необходимости масштабирования.
4. Высокая степень интеграции: многие технологии и инструменты можно без особой сложности интегрировать с Python, это могут быть как системы управления базами данных и API различных приложений, так и другие языки программирования. Это также позволяет сократить временные затраты и сложность разработки платформы [18].
5. Личный опыт работы с данным языком программирования.

### **2.3.2. Выбор фреймворка**

Для разработки платформы был выбран фреймворк Django [19]. Это фреймворк с открытым исходным кодом, который хорошо подходит для разработки интернет-платформы по ряду причин:

1. Популярность и развитость: Django был выпущен в 2005 году и с этого момента и до сих пор активно развивается. За это время он стал одним из лучших веб-фреймворков, написанных на Python. Соответственно Django, также, как и сам Python, имеет большое сообщество разработчиков, что облегчает поиск решений популярных проблем и ответов на часто возникающие вопросы. Также Django оснащен большим количеством собственных библиотек.
2. Готовые решения: Django предоставляет множество готового функционала, например, миграцию баз данных или обработку форм. Это очередное преимущество, позволяющее ускорить разработку и сократить время на исправления ошибок, которые могут возникать при самостоятельной реализации функций.
3. Готовая система безопасности: Django предоставляет встроенные функции безопасности, которые обеспечивают защиту от распространенных проблем безопасности, например, от «cross-site request forgery» (межсайтовая подделка запроса), «SQL injection» (внедрение SQL-кода), «cross-site scripting» (внедрение вредоносного кода на страницы) и «clickjacking» (размещение вызывающих какие-то действия невидимых элементов на сайте поверх видимых активных).
4. Готовая панель администратора: настраиваемая административная панель генерируется Django автоматически при создании приложения и имеет удобный интерфейс. С ее помощью можно управлять данными, пользователями, и настройками. Это означает, что не требуется тратить дополнительного времени на создание собственного интерфейса администрирования вручную.

5. Взаимодействие с базами данных: Django реализует объектно-реляционное отображение (ORM – Object-Relational Mapping), которое обеспечивает взаимодействие приложения с базами данных (автоматически передаёт данные из БД в объекты, которые используются в коде приложения).

### 2.3.3. Выбор СУБД

В проектах Django по умолчанию в качестве хранилища для файлов используется база данных SQLite, но она не подходит для больших и многопользовательских приложений. Поэтому для увеличения производительности было решено использовать PostgreSQL [20]. Это объектно-реляционная система управления базами данных с открытым исходным кодом, которая использует и расширяет язык SQL в сочетании со многими функциями, обеспечивающими безопасное хранение и масштабирование самых сложных рабочих нагрузок с данными.

Основные причины выбора PostgreSQL:

1. Совместимость с Django: PostgreSQL имеет отличную поддержку Django благодаря встроенному адаптеру, который обеспечивает интеграцию между этими технологиями. Это упрощает разработку интернет-платформ на Django с PostgreSQL.
2. Надежность: данная СУБД известна своей стабильностью работы, что является очень важным преимуществом для интернет-платформ, которые требуют высокую доступность и время безотказной работы. PostgreSQL предоставляет такие функции, как целостность данных, обработку транзакций, и автоматическое восстановление, что помогает обеспечить сохранность данных и бесперебойную работу интернет-платформы.

3. Производительность: PostgreSQL обладает высокой скоростью и эффективностью работы, он разработан таким образом, что может обрабатывать большие объемы данных и высокую нагрузку на трафик. В нем есть поддержка передовых методов индексации и оптимизации запросов, которые могут удерживать производительность интернет-платформы на высоком уровне.
4. Масштабируемость: PostgreSQL может легко масштабироваться по мере роста интернет-платформы. С ним можно производить как вертикальное, так и горизонтальное масштабирование. Можно добавлять ресурсы на сервер баз данных или использовать методы кластеризации и сегментирования для распределения данных по нескольким серверам.

#### **2.3.4. Выбор среды разработки**

Для разработки был выбран PyCharm [21]. Он представляет собой интегрированную среду разработки (IDE), которая поддерживает разработку на Django.

PyCharm предоставляет расширенные функции завершения кода, отладки и тестирования, которые помогают писать высококачественный код. Основными преимуществами PyCharm являются:

- удобный интерфейс;
- мощный и функциональный редактор кода с подсветкой синтаксиса, подсказками, авто-форматированием и авто-отступами;
- интеграция с системой управления версиями;
- большая коллекция плагинов;
- мощный инструмент рефакторинга кода, который позволяет быстро внести изменения и не допустить лишних ошибок;
- полнофункциональный графический отладчик.

Для более удобной работы с базой данных, было решено работать в специальном GUI (графический интерфейс пользователя). С его помощью гораздо проще обрабатывать и анализировать данные, чем это происходит в консоли. В качестве GUI был выбран DataGrip, так как он совместим с операционной системой macOS, на которой ведется разработка, и разработан той же компанией, что и PyCharm.

### **2.3.5. Выбор платежной системы**

При выборе платежной системы были выделены основные критерии, которыми она должна обладать:

- наличие API для Python;
- поддержка автоплатежей;
- возможность производить оплату между разными пользователями;
- обеспечение высокого уровня безопасности при оплате.

Были рассмотрены следующие платежные системы: «ЮКасса», «Tinkoff», «PayPal» и «Stripe».

Из представленных вариантов, выставленным критериям не удовлетворяет только платежная система «Tinkoff», она не обладает API для Python и не поддерживает настройку автоплатежей.

Для доступа к API «ЮКасса» необходимо быть зарегистрированным в сервисе, а при регистрации требуется указать ИНН организации (при наличии юридического лица) или собственный (при наличии ИП или самозанятости), в связи с чем данный вариант оказался неподходящим.

В связи с прекращением работы «PayPal» с российскими пользователями, интеграция данной платежной системы является невозможной.

Исходя из вышеупомянутых фактов, была выбрана платежная система «Stripe» – она удовлетворяет всем выделенным критериям и является относительно простой в интеграции [22].

## **2.4. Проектирование архитектуры платформы**

### **2.4.1. Особенности архитектуры Django проекта**

Так как Django имеет MVT архитектуру, вся платформа будет строиться согласно ей [23].

MVT – это набор из следующих компонентов: Model, View и Template (модель, представление и шаблон). Он схож с шаблоном MVC – Model, View и Controller (контроллер). Это способ организации кода, который предполагает выделение блоков, отвечающих за решение разных задач, что позволяет работать с визуальным представлением и бизнес-логикой приложения отдельно (один блок отвечает за данные приложения, другой отвечает за внешний вид, третий контролирует работу приложения). Django следует этому шаблону, при этом сохраняя свои собственные соглашения. В MVT Django нет отдельного контроллера, полное приложение основано на представлении, модели и шаблоне.

Model (модель) отвечает за взаимодействие с базой данных и выполнение обработки данных. Обычно одна модель указывает на одну таблицу в базе данных.

View (представление) получает запрос, выполняет бизнес-обработку и возвращает ответ. Представление получает данные от модели и предоставляет шаблонам доступ к этим данным с возможностью предварительной обработки.

Template (шаблоны) – это файлы, содержащие HTML-код, с помощью которого отображаются данные на странице. Содержимое файлов шаблонов может быть статическим и динамическим.

На рисунке 10 видно, что пользователь отправляет запрос к Django, а Django работает как контроллер и проверяет доступный ресурс в URL-адресе.

В случае, когда URL-адреса совпадают, идет обращение к представлению, которое, в свою очередь, обращается к модели и шаблону (отображает шаблон). Django отправляет шаблон пользователю в качестве ответа.

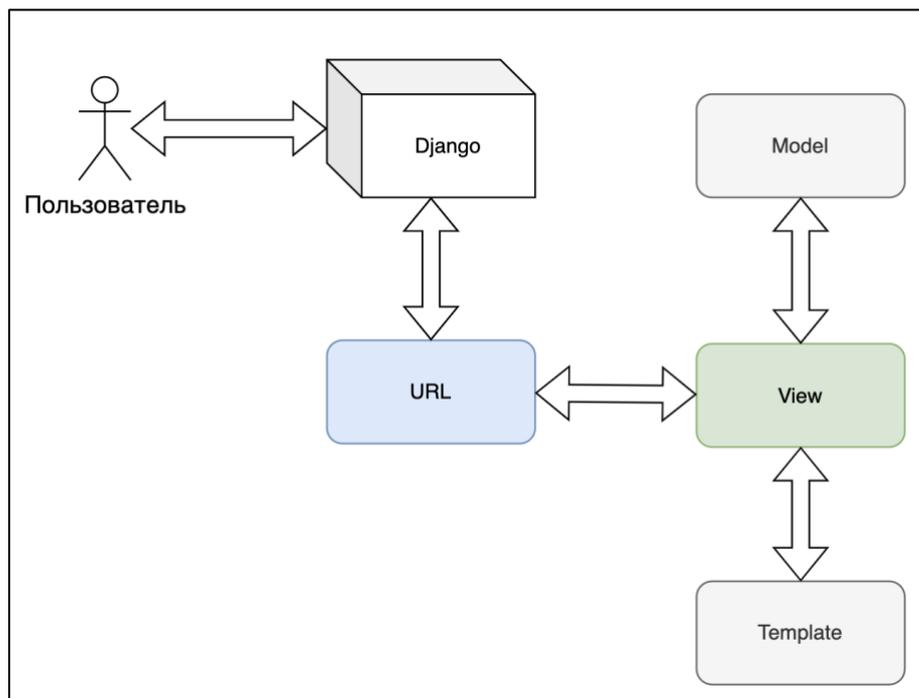


Рисунок 10 – Схема работы MVT

#### 2.4.2. Выявление и организация компонентов платформы

Для более четкого понимания структуры кода и проекта в целом была построена UML–диаграмма компонентов, представленная на рисунке 11.

Каждый компонент спроектирован с тем учетом, что он будет исполнен в качестве отдельного приложения в проекте Django. Все компоненты располагаются в блоке «Интерфейс», так как с каждым из них должен взаимодействовать пользователь, для них были разработаны шаблоны (templates). Сам Django на диаграмме представлен в качестве сервера приложений в блоке «Бэкэнд» [24].

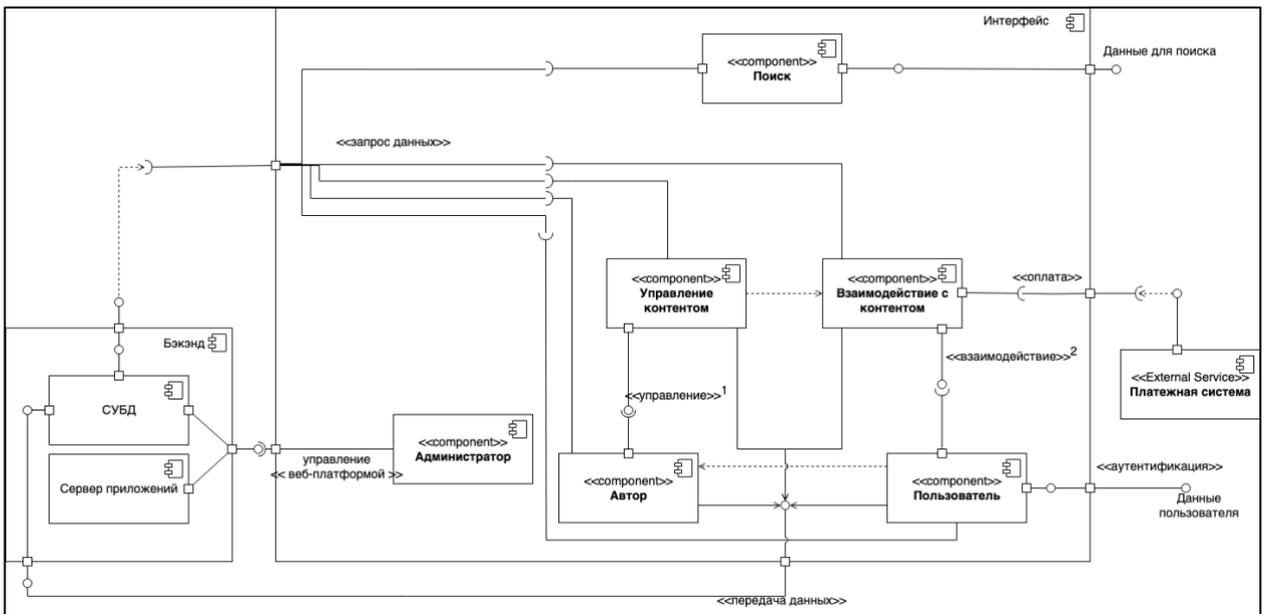


Рисунок 11 – Диаграмма компонентов

Сноски с рисунка:

1. Управление – автор может создавать, редактировать и удалять подписки и публикации.
2. Взаимодействие – пользователь может ставить контенту положительные либо отрицательные оценки, оставлять комментарии, оформлять подписки и оплачивать доступ.

### 2.4.3. Развертывание платформы

Для понимания архитектуры исполнения системы, включая такие узлы, как программные и аппаратные среды исполнения и соединяющее их промежуточное программное обеспечение, была построена UML-диаграмма развертывания, изображенная на рисунке 12 [25].

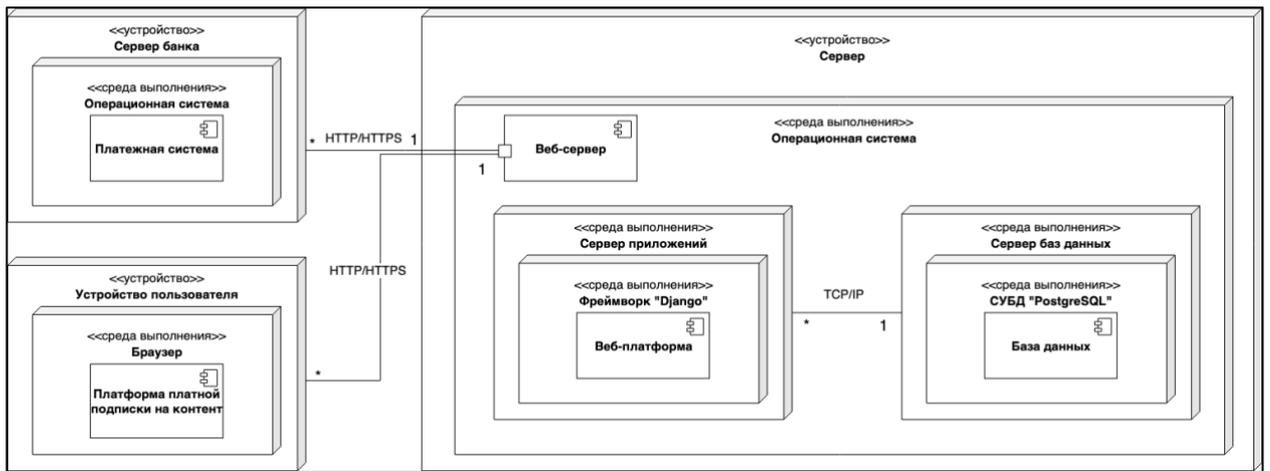


Рисунок 12 – Диаграмма развертывания

Из диаграммы видно, что имеется физический сервер, как устройство, на котором расположен веб-сервер, осуществляющий взаимодействие веб-платформы с внешними системами. Внешнее взаимодействие осуществляется с пользователем платформы, который может получить доступ к ней через браузер, используя личное устройство, а также со сторонней платежной системой. Так как пользователей и платежных систем может быть больше одного, их связь с веб-сервером отмечена как один ко многим.

Внутри самого веб-сервера сервер приложений осуществляет взаимодействие с сервером баз данных (Django взаимодействует с базами данных).

## 2.5. Результаты проектирования

В ходе написания данной главы был обоснован выбор конкретных инструментов разработки, определены основные роли и их функции, спроектированы база данных и архитектура системы. При проектировании, для улучшения проработки и визуализации, использовались UML-диаграммы.

### 3. Программная реализация онлайн-платформы

#### 3.1. Файловая структура

Для удобства разработки внутри проекта было произведено разделение на разные приложения, которые практически совпадают с компонентами диаграммы, представленной на рисунке 13. Были созданы следующие приложения:

- **author** – обеспечивает работу функций, обрабатывающих действия связанные с авторами;
- **content\_management** – включает функции по созданию, редактированию и просмотру контента;
- **main** – включает функции, необходимые для общей работы платформы.

Файловая структура проекта представлена на рисунке ниже.

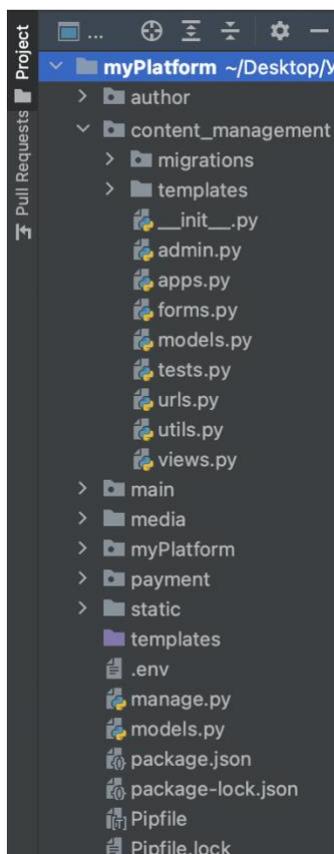


Рисунок 13. Корневая директория проекта

Каждое приложение имеет практически одинаковый набор вложенных файлов. На рисунке, в качестве примера, отображено наполнение директории приложения `content_management`. При разработке платформы, основная работа происходила со следующими файлами и папками:

- **templates** содержит html-файлы необходимые для отображения страниц платформы;
- **admin.py** содержит модели, к которым открывается доступ через админ-панель, также здесь прописываются ограничения при работе с ними;
- **forms.py** необходим для создания форм, с которыми взаимодействует пользователь;
- **models.py** содержит в себе модели, которые необходимы для создания сущностей в базе данных и последующей работы с ними;
- **urls.py** необходим для указания Django, какие страницы необходимо открывать по определенному запросу;
- **utils.py** был создан для содержания функций, которые не были включены во **views.py** и для создания самодельных **Mixin**-классов (подробная информация о них будет представлена далее).
- **views.py** содержит в себе функции и методы классов, необходимые для обработки данных и последующей их передачи в шаблоны (`templates`).

Непосредственно файлы **models.py**, **views.py** и html-файлы в папке `templates` осуществляют MVT-архитектуру, о которой говорилось ранее.

### 3.2. Работа с БД

К проекту Django была подключена база данных PostgreSQL. Подключение реализуется в файле **settings.py** путем изменения стандартных настроек (рисунок 14).

```
settings.py x
86 DATABASES = {
87     "default": {
88         'ENGINE': 'django.db.backends.postgresql_psycopg2',
89         'NAME': 'platform_db',
90         'USER': 'olegsemenov',
91         'PASSWORD': DB_PASSWORD,
92         'HOST': 'localhost',
93         'PORT': '5432',
94     }
95 }
```

Рисунок 14. Подключение БД к проекту Django

Согласно диаграмме базы данных, представленной на рисунке 9, были созданы модели Django. Пример созданной модели, описывающей сущность публикации, представлен на рисунке ниже.

```
content_management/models.py x main/models.py x
34 class Publication(models.Model):
35     id_publication = models.AutoField(primary_key=True)
36     id_author = models.ForeignKey(Author, models.DO_NOTHING, db_column='id_author', blank=True)
37     id_subscription = models.ForeignKey(AuthorSubscription, models.DO_NOTHING, db_column='id_author_subscription',
38         blank=True)
39     title = models.CharField(max_length=255, verbose_name="Заголовок")
40     text_content = models.TextField(blank=True, verbose_name="Текст статьи")
41     price = models.DecimalField(max_digits=10, decimal_places=2, blank=True, null=True)
42     time_create = models.DateTimeField(auto_now_add=True, verbose_name="Время создания", blank=True)
43     time_update = models.DateTimeField(auto_now=True, verbose_name="Время изменения", blank=True)
44     is_published = models.BooleanField(default=True, verbose_name="Публикация")
45
46     class Meta:
47         db_table = 'publication'
48
49     def __str__(self):
50         return self.title
51
52     def get_absolute_url(self):
53         return reverse('publication_page', kwargs={'id_publication': self.pk})
```

Рисунок 15 – Модель Publication в Django

После описания моделей и их миграции в базу данных, модели и последующие записи отображаются в БД, как на рисунках 16 и 17.

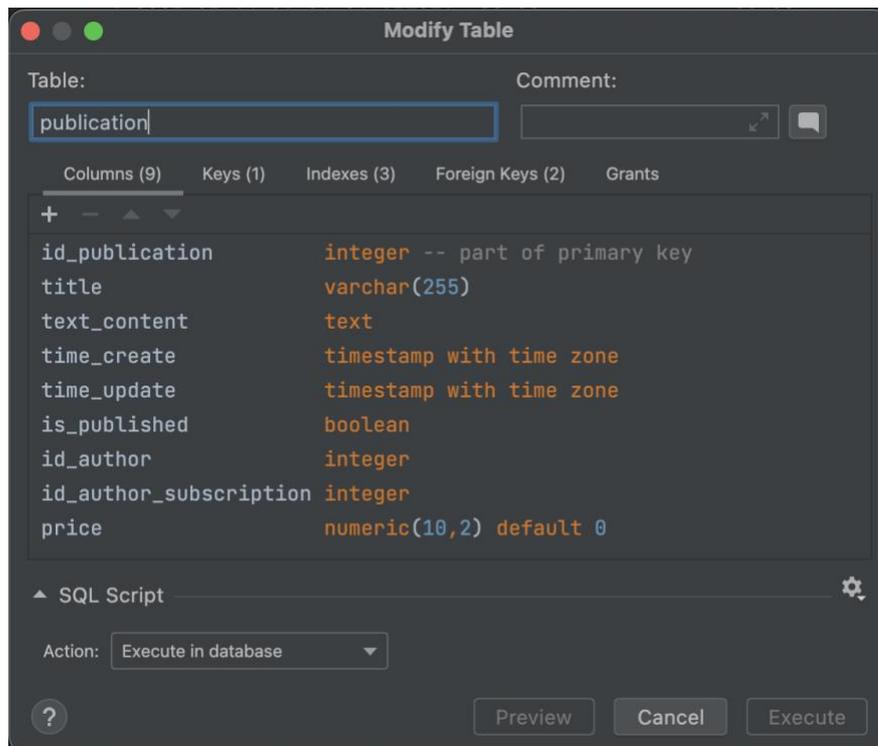


Рисунок 16 – Таблица Publication в PostgreSQL

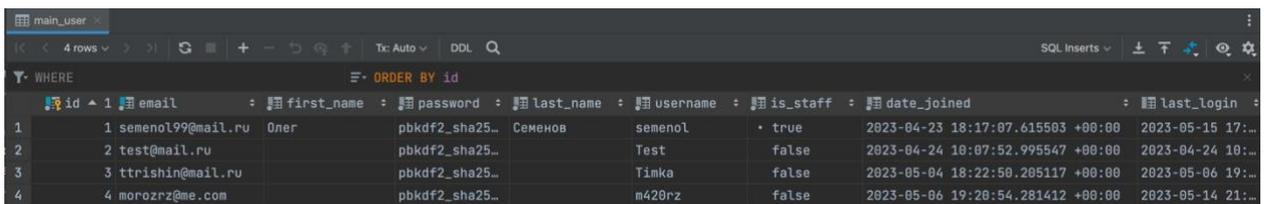


Рисунок 17 – Пример записей в таблице Publication в PostgreSQL

В Django встроена собственная пользовательская модель, для которой реализованы готовые функции регистрации и аутентификации, но она обладает неполным набором полей, поэтому ее необходимо было расширить.

Для расширения пользовательской модели есть несколько способов, в данной работе был выбран способ с наследованием класса `AbstractBaseUser`. Такой способ считается наиболее сложным в реализации, но он является наиболее подходящим для данной работы. `AbstractBaseUser` не имеет собственных полей и предоставляет только функцию аутентификации. Аутентификация в Django установлена по умолчанию по имени пользователя (`username`), при определении подкласса модели `MainUser` данная настройка была также изменена на аутентификацию по электронной почте

пользователя. Переопределенная модель пользователя представлена на рисунке ниже.

```
models.py x base_user.py x
19
20 class MainUser(AbstractBaseUser, PermissionsMixin):
21     id = models.AutoField(primary_key=True)
22     email = models.EmailField(_("email address"), unique=True)
23     first_name = models.CharField(max_length=150)
24     last_name = models.CharField(max_length=150)
25     patronymic = models.CharField(max_length=150, blank=True, null=True)
26     username = models.CharField(max_length=150)
27     is_staff = models.BooleanField(default=False)
28     is_active = models.BooleanField(default=True)
29     date_joined = models.DateTimeField(default=timezone.now)
30     phone_number = models.CharField(max_length=45, blank=True, null=True)
31     birthday = models.DateTimeField(blank=True, null=True)
32     profile_photo = models.ImageField(upload_to='avatar/%Y/%m/%d/', verbo
33
34     USERNAME_FIELD = "email"
35     REQUIRED_FIELDS = []
36
37     objects = CustomUserManager()
38
39     def __str__(self):
40         return self.email
41
42     class Meta:
43         db_table = 'main_user'
44
45     def get_absolute_url(self):
46         return reverse('id', kwargs={'slug': self.id})
```

Рисунок 18 – Переопределенная модель пользователя

Все запросы к базе данных происходят через созданные модели, используя Django ORM.

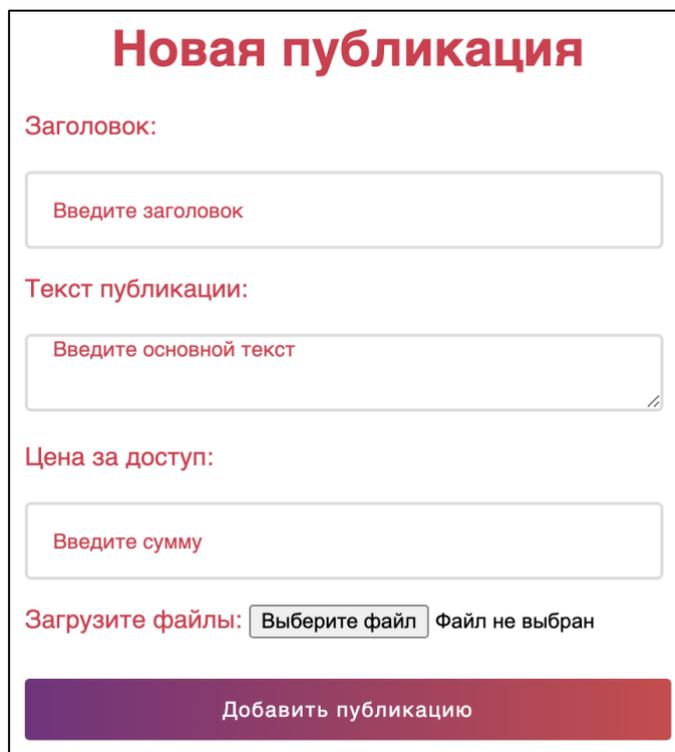
### 3.3. Реализация бизнес-логики

При разработке бизнес-логики данной платформы, были выделены следующие основные задачи:

1. реализация создания и доступа к публикациям;
2. реализация создания и доступа к подпискам;
3. реализация настройки подписок и доступа к устаревшему контенту в них.

### 3.3.1. Реализация публикаций

Автор может создать публикацию отдельно от подписки, тогда она будет просто располагаться на его странице. Автор может оставить публикацию доступной для любого аутентифицированного пользователя или сделать ее платной.



**Новая публикация**

Заголовок:

Введите заголовок

Текст публикации:

Введите основной текст

Цена за доступ:

Введите сумму

Загрузите файлы:  Файл не выбран

Добавить публикацию

Рисунок 19 – Форма создания публикации не входящей в подписку

Автор может создать публикацию внутри подписки, при этом он не может выставить цену за доступ к ней, и она будет доступна только для пользователей, оплативших данную подписку и удовлетворяющих настройкам доступа, которые выставил автор.

**Новая публикация**

**В подписке: «Тест»**

Заголовок:

Тестовая публикация

Текст публикации:

Публикация внутри подписки

Загрузите файлы: Выберите файл UM2F0117.jpg

Добавить публикацию

Рисунок 20 – Форма создания публикации внутри подписки

В результате, публикация при любом варианте создания будет выглядеть следующим образом:



Рисунок 21 – Отображение публикации

### 3.3.2. Реализация подписок

Автор может создавать подписки и выставять уровень доступа к устаревшему контенту. Пример формы для создания подписки представлен на рисунке ниже.

## Новая подписка

- При такой настройке нельзя выставить цену за доступ к старому контенту

**Заголовок:**

**Описание:**

**Цена за доступ к подписке:**

**Цена за доступ к старому контенту:**

**Фото обложки:**

**Выберите настройку доступа к старому контенту:**

- Не скрывать
- Скрыть для всех
- Выставить одну цену за доступ ко всему старому контенту
- Выставить цену за каждую единицу контента в подписке

Рисунок 22 – Форма создания подписки

Для демонстрации работы формы, на рисунке 22 приведен пример неверного заполнения формы, а именно заполнения поля «Цена за доступ к старому контенту» при выборе настройки доступа «Не скрывать», то есть автор не может выставить цену за доступ к устаревшему контенту, так как выбрал опцию открытого доступа.

При переходе к созданной доступной подписке, она будет отображена пользователю следующим образом:

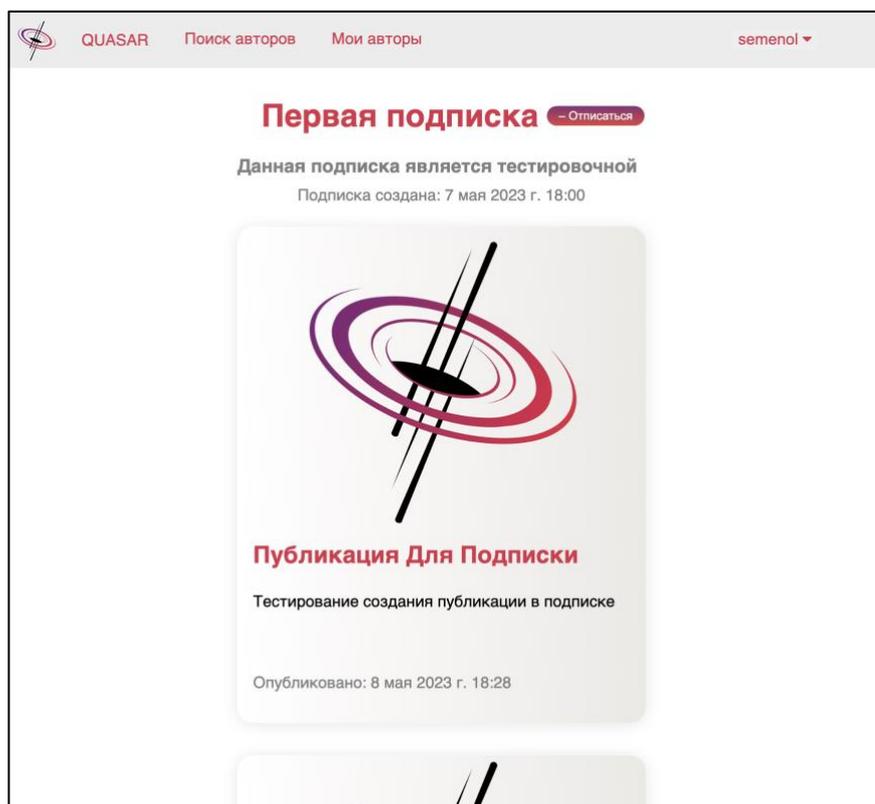


Рисунок 23 – Отображение подписки

### 3.3.3. Реализация настройки доступа к устаревшему контенту

Устаревшим контентом считаются публикации внутри подписок, с момента создания которых прошло более месяца.

Новым подписчиком является пользователь, который в первый раз оформил данную подписку на 30 и более дней позднее ее создания. Это условие проверяется по дате первого оформления пользователем конкретной подписки (таблица «user\_subscription», поле «date\_first\_subscription») и дате создания подписки (таблица «author\_subscription», поле «time\_create»).

Для настройки доступа к устаревшему контенту для новых подписчиков были выделены следующие варианты настройки подписок:

Вариант №1 – доступ открыт всем подписчикам;

Вариант №2 – доступ закрыт ко всем устаревшим публикациям;

Вариант №3 – доступ ко всем устаревшим публикациям предоставляется после единовременной оплаты;

Вариант №4 – доступ к каждой устаревшей публикации оплачивается отдельно.

При реализации описанных вариантов, было выявлено две проблемы.

Проблема №1: пользователь мог оформить подписку в начале ее существования на один месяц, а затем снова ее оформить через любое количество времени, например, через год. В этом случае он не будет считаться новым подписчиком и получит доступ ко всему контенту, который публиковал автор на протяжении всего года, заплатив только за 2 месяца.

Решение: при оформлении подписки производится проверка – если пользователь уже оплачивал подписку ранее, и с момента последней оплаты до настоящего времени прошло более трех месяцев, поле «date\_first\_subscription» перезаписывается на текущую дату, следовательно, пользователь, при попытке открыть устаревшую публикацию, определится как новый.

Проблема №2: автор может менять настройку доступа, из-за чего существующим или заплатившим за доступ новым пользователям может закрыться доступ к устаревшим публикациям. Например, автор изначально выставил 4 вариант доступа, новый пользователь оплатил доступ к публикации, и по прошествии времени автор решил полностью закрыть доступ к устаревшему контенту.

Решение: оплата за каждую публикацию или общий доступ записывается в отдельные таблицы, и при открытии каждой публикации производится проверка на наличие доступа к ней у текущего пользователя.

Факт оплаты за доступ к каждой отдельной публикации, вне зависимости от того, находится она в подписке или нет, записывается в таблицу «paid\_publication». То же самое правило принято и для оплаты подписки – каждый факт оплаты, даже одной и той же подписки, записывается в таблицу «paid\_subscription».

Для записи факта оплаты доступа к устаревшему контенту (при варианте доступа №3) было принято решение делать только одну запись в таблицу «access\_raid» и обновлять ее при повторной оплате доступа к устаревшему контенту в той же подписке. Такой вариант возможен, если пользователь сначала оплатил доступ к устаревшему контенту, а затем более трех месяцев не продлял подписку, но потом, подписавшись, решил снова оплатить доступ к контенту, который он пропустил.

Также стоит отметить, что публикации, созданные через месяц после того, как подписчик перестал продлевать подписку, но подписался снова в течение трех месяцев, все равно будут для него устаревшими, и доступ к ним будет предоставляться исходя из настроек доступа подписки. Публикации, созданные до его отписки, также останутся доступными.

Исходя из всех описанных выше условий, был разработан алгоритм проверки доступа к публикациям. Данный алгоритм может инициировать только зарегистрированный пользователь. Блок-схемы алгоритма представлены на рисунках ниже.

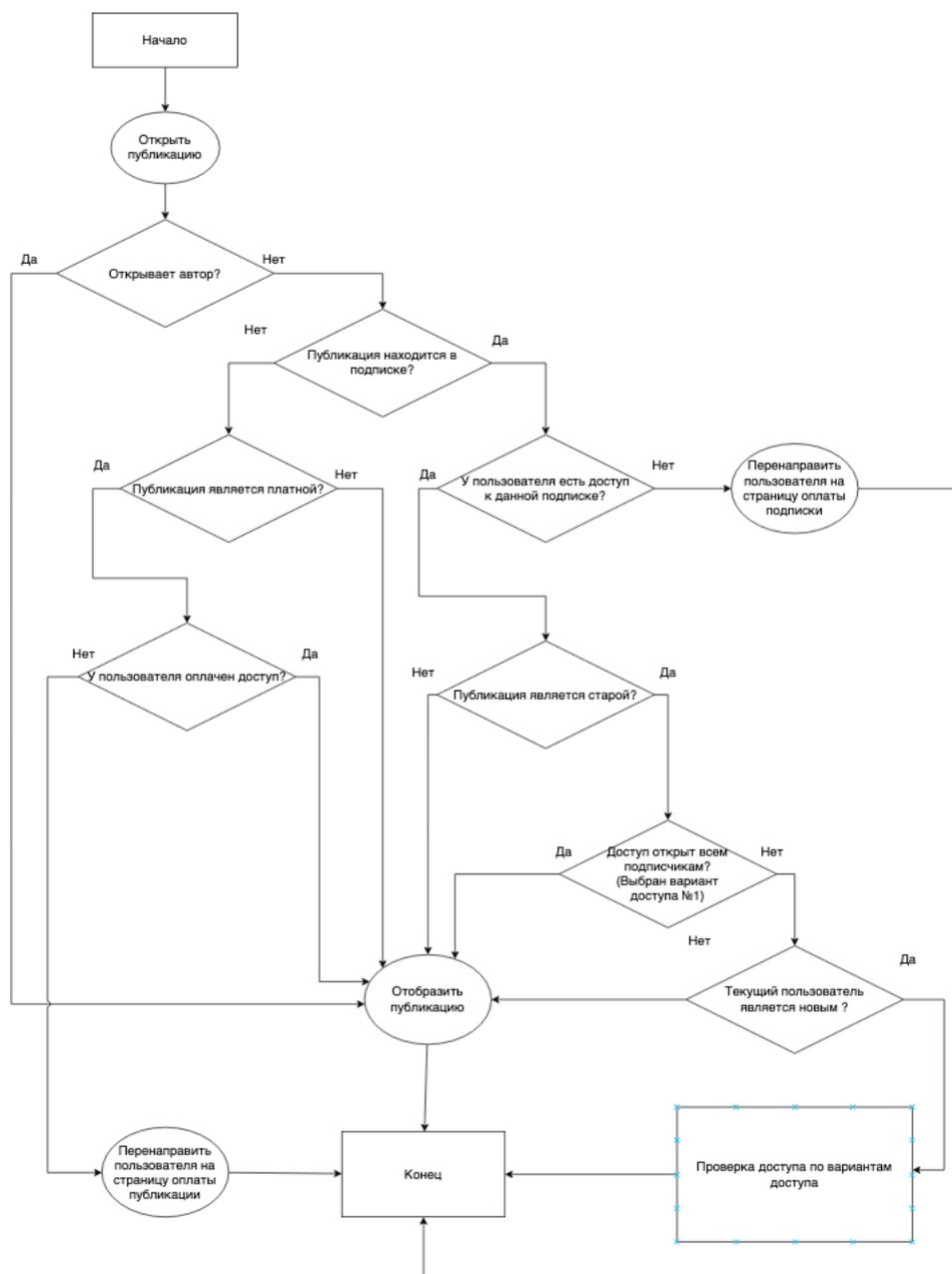


Рисунок 24 – Блок-схема алгоритма проверки доступа к публикациям

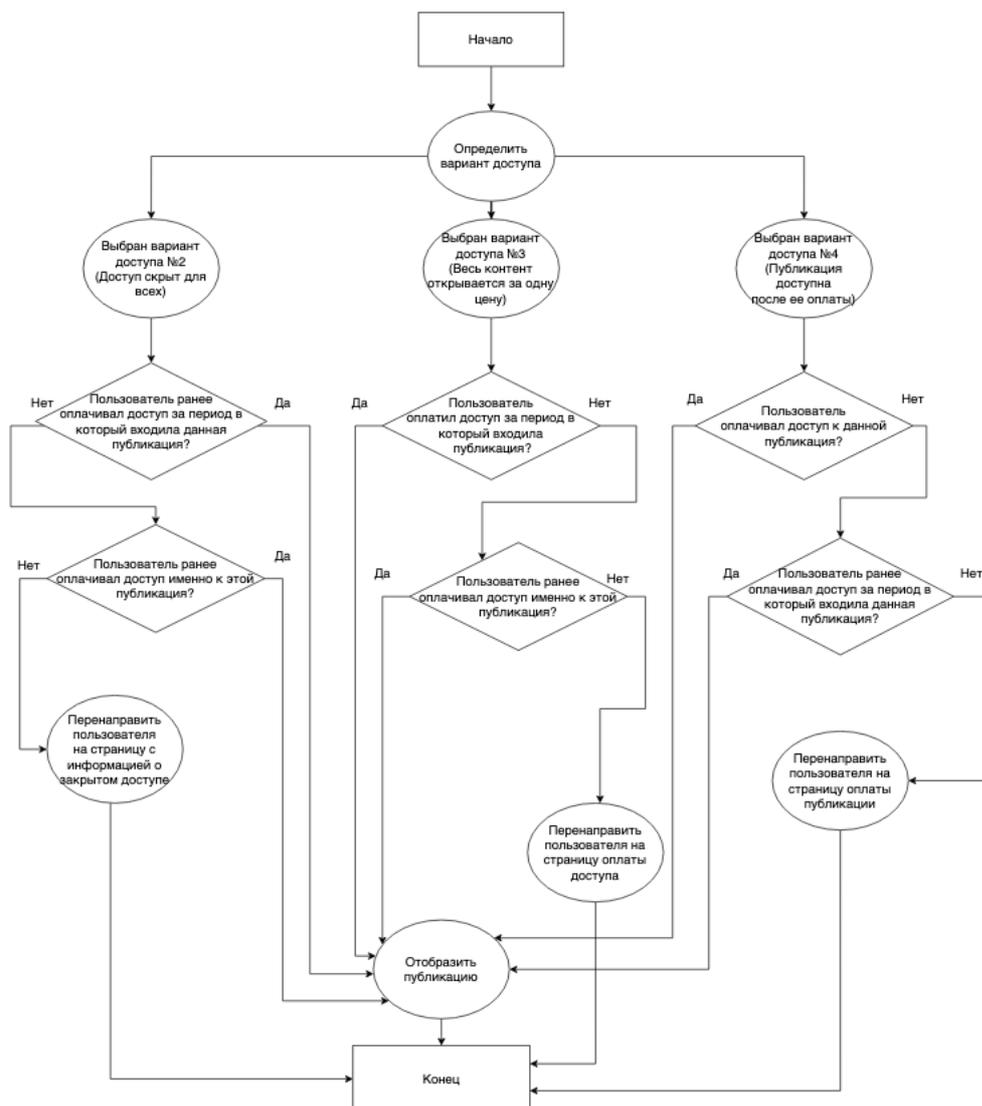


Рисунок 25 – Блок-схема алгоритма проверки доступа к публикациям (по вариантам доступа)

Этот алгоритм реализован в самодельных **Mixin**-классах, о которых упоминалось ранее, и срабатывает во время вызова класса представления для отображения публикации.

### 3.4. Работа со Stripe

#### 3.4.1. Оплата публикаций

Как видно из алгоритма, описанного на рисунках 24-25, если пользователь не оплатил доступ к контенту, он, в большинстве случаев, при

попытке взаимодействия с контентом, будет перенаправлен на страницу с информацией об оплате, как на рисунке ниже.

## Тестовая публикация

---

**Для просмотра данной публикации  
необходимо оплатить доступ**

**Стоимость публикации: 100,00**

**Способ оплаты: картой на платформе Stripe**

Я принимаю Условия получения доступа к публикации

---

[Перейти к оплате](#)

Рисунок 26 – Информация об оплате доступа к публикации

Далее, если пользователь принимает условия получения доступа к публикации, и переходит к оплате, он будет перенаправлен на страницу оплаты Stripe, представленной на рисунке 27. Так как эксплуатация платежной системы производится в тестовом режиме, все цены указываются в долларах.

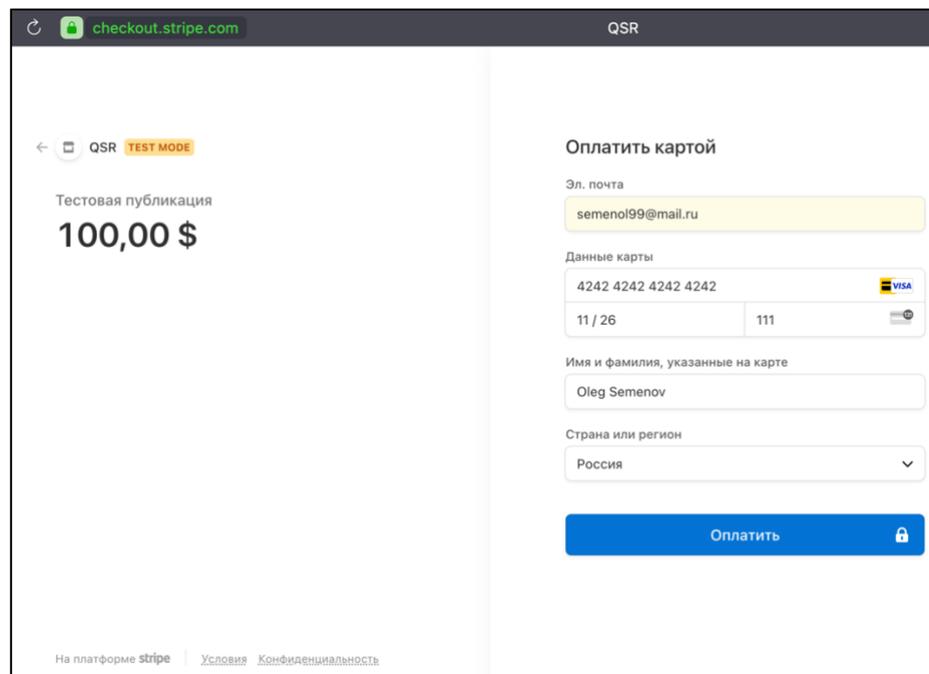


Рисунок 27 – Пример формы оплаты публикации на платформе Stripe

После ввода данных и оплаты, Stripe отправляет событие с информацией о завершенном сеансе оплаты. Перед обработкой полученных данных в проекте Django, необходимо произвести проверку, что событие пришло именно от нашей платежной системы. Библиотека Stripe имеет для этого встроенную верификацию событий. После проверки, чтобы обработать данные, необходимо произвести обработку события **checkout.session.completed**, в котором хранятся данные об оплаченной публикации и оплатившем пользователе. Для прослушивания данного события был создан специальный перехватчик (**webhook**). Полученные данные заносятся в базу данных, после чего пользователь перенаправляется на страницу с информацией об успешной оплате. Данные обо всех платежах можно просматривать в личном кабинете Stripe, как показано на рисунке ниже.

AMOUNT	DESCRIPTION	CUSTOMER	DATE
\$50.00 USD	Subscription payment	semenol99@mail.ru	May 22, 12:01 AM
\$100.00 USD	p1_3NA5zrC4FV3SAUMd1WvTqBn	semenol99@mail.ru	May 21, 1:44 PM
\$1,111.00 USD	p1_3N8dGBC4FV3SAUMd1m7RY9xc	hviyhivihv@me.com	May 17, 12:50 PM

Рисунок 28 – Информация о проведенных платежах на платформе Stripe

Для оплаты доступа ко всему устаревшему контенту внутри подписок (при третьем варианте доступа) и оплаты самих подписок созданы отдельные обработчики оплаты. Оплата доступа к устаревшему контенту отличается от оплаты публикации только данными, которые передаются и запрашиваются для заполнения в базу данных.

### 3.4.2. Оплата подписок

Для настройки рекуррентных платежей, по правилам Stripe, необходимо создать объект **Price** (предназначен для указания, сколько и как часто взимать плату), который назначается объекту **Product**. Исходя из этого, подписка, созданная автором, будет определяться как объект **Product**, а ее месячная стоимость («monthly\_price») как **Price**.

Сразу после того, как форма создания подписки проходит валидацию, и данные заносятся в базу, происходит создание продукта и цены на платформе Stripe. Пример создания подписки, представленной в форме на рисунке 22, показан на рисунках ниже.

id_author_subscrip...	title	description	monthly_price	time_create
21	Подписка - тес...	Подписка созданная для тестирования платформы Str...	50.00	2023-05-21 16:29:2

Рисунок 29 – Отображение созданной подписки в базе данных

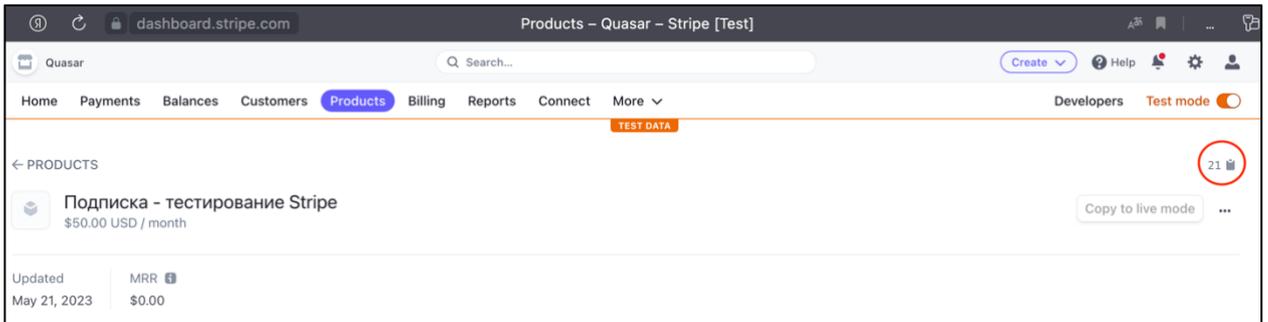


Рисунок 30 – Отображение созданной подписки на платформе Stripe

Для оформления созданной подписки, пользователи проходят тот же алгоритм, что и при оплате доступа к публикации. Но в данном случае Stripe «запоминает» данные пользователя, необходимые для оплаты, и через указанный период произведет списание средств самостоятельно. На рисунке ниже представлена форма оплаты Stripe.

Рисунок 31 – Пример формы оплаты подписки на платформе Stripe

После успешной оплаты, срабатывает тот же перехватчик, о котором упоминалось выше, и заносит данные в базу, а пользователь перенаправляется на страницу с сообщением об успешном оформлении подписки. Активность подписок у конкретных пользователей удобно отслеживать на самой платформе Stripe.

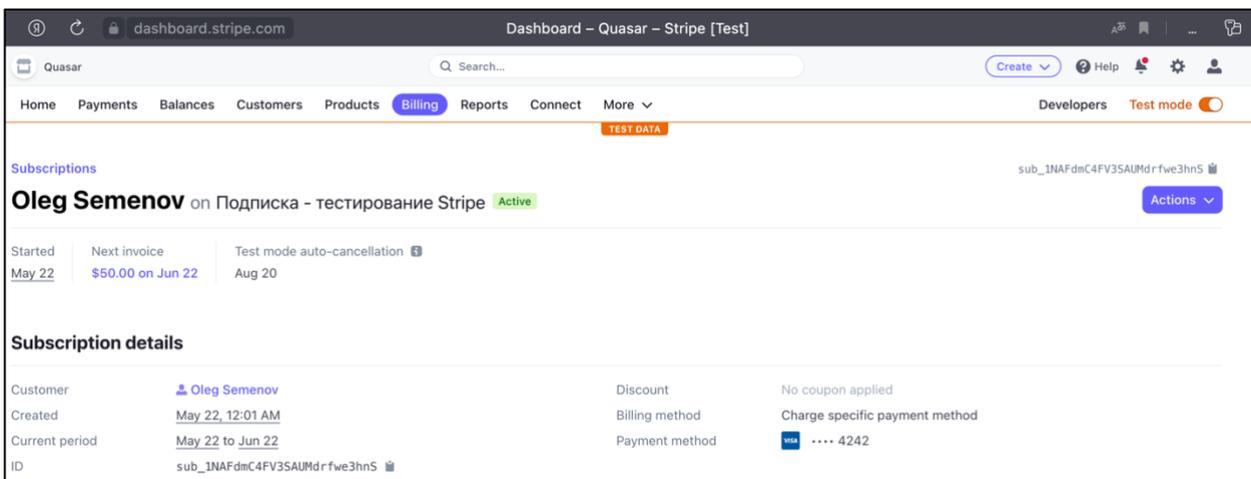


Рисунок 32 – Информация об активности подписки у конкретного пользователя

При проведении автоплатежей, платежная система отправляет уведомления о событиях на `webhook`, заранее прописанный в Django, как часть запроса POST с полезной нагрузкой JSON.

### 3.4.3. Вывод средств

Все средства пользователей, внесших оплату, поступают на один счет в Stripe, поэтому выплаты авторам на данном этапе можно осуществлять следующим образом: по итогам расчетного периода, с помощью таблиц, хранящих в себе информацию об оплате в созданной базе данных, и информации о платежах в Stripe, можно делать подсчет заработанных средств и затем рассылать их авторам на собственные счета.

Также был рассмотрен еще один вариант, однако он требует дополнительной реализации. Суть его заключается в том, что пользователям, при создании страницы автора, необходимо переходить на Stripe и привязывать уже существующий аккаунт или создавать новый для своей страницы автора, вследствие чего средства сразу могут поступать на личный счет автора в Stripe.

## 3.5. Обзор основных результатов разработки веб-платформы

### 3.5.1. Поиск авторов

На рисунке ниже представлена реализация поиска авторов. Поиск может производиться по категориям, которые выбрал себе автор, а также по никнейму или адресу страницы автора через поле ввода. Поиск является нечувствительным к регистру.

На странице также реализованы кнопки для подписки на интересующих авторов, чтобы их было проще найти в дальнейшем. Данные кнопки реализованы с помощью технологии AJAX, что позволяет, не перезагружая страницу менять отображение кнопок с «Подписаться» на «Отписаться» и наоборот, одновременно производя запись об этом в базу данных.

Также на странице реализована пагинация.

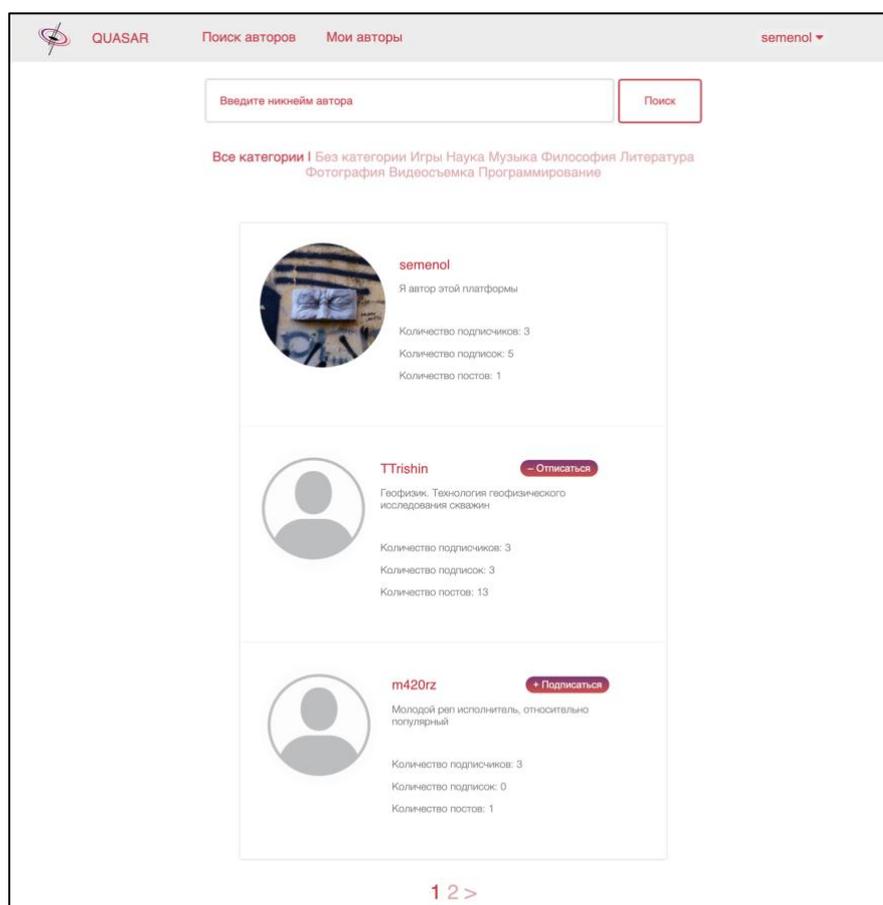


Рисунок 33 – Страница поиска авторов

### 3.5.2. Страница автора

На странице автора располагаются списки его публикаций и подписок, между ними реализовано переключение. На рисунке ниже показан пример страницы автора, открытой от лица самого автора. Если страницу автора открывает не сам автор, элементы для перехода к редактированию информации и добавлению контента для такого пользователя не отобразятся. Если неавторизованный пользователь попытается открыть публикацию или подписку, он будет автоматически перенаправлен на страницу ввода электронной почты и пароля.



Рисунок 34 – Пример отображения страницы автора для самого автора

### 3.5.3. Регистрация и авторизация

На рисунках ниже представлены формы для регистрации и авторизации. Как уже говорилось ранее, Django предоставляет готовые методы для проведения регистрации и авторизации, но так как модель пользователя была переопределена, необходимо было переопределить и формы, например, добавить метод проверки уникальности полей, что отображено на рисунке ниже.

**Регистрация**  
Создайте свою учетную запись

Test\_\_1

semenol99@mail.ru

- Такой Email уже зарегистрирован

.....

.....

**Зарегистрироваться**

Рисунок 35 – Форма регистрации

**Вход в личный кабинет**

Войдите в свою учетную запись, используя логин и пароль

semenol99@mail.ru

.....

[Забыл пароль](#)

Войти

Рисунок 36 – Форма авторизации

Если пользователь забыл пароль, он может его восстановить, нажав на элемент «Забыл пароль», после чего он будет перенаправлен на страницу, где ему будет предложено ввести свой адрес электронной почты – рисунок 35. Это представление также обрабатывается с помощью расширенного встроенного класса Django.

В этом представлении отображается форма, в которую пользователь вводит свой адрес электронной почты, который будет идентифицирован. После проверки генерируется временная ссылка для сброса пароля, которая будет использоваться только один раз для одного пользователя. А пользователь получает сообщение, отображенное на рисунке 36. Эта ссылка будет отправлена на адрес электронной почты пользователя, пример показан на рисунке 37.

Если на указанный адрес электронной почты нет зарегистрированного пользователя, никакого сообщения об этом не выводится, но письмо с ссылкой не отправляется. Это реализовано для того, чтобы предотвратить утечку информации.

Для отправки сообщений по электронной почте используется портативный сервер Google SMTP – один из компонентов Gmail.

## Забыли пароль?

Email

semenol99@mail.ru

Получить сообщение

Создать новый аккаунт

Вернуться к авторизации

Рисунок 37 – Форма заполнения email для пользователя, забывшего пароль



Рисунок 38 – Надпись, высвечивающаяся пользователям, после отправки запроса на сброс пароля

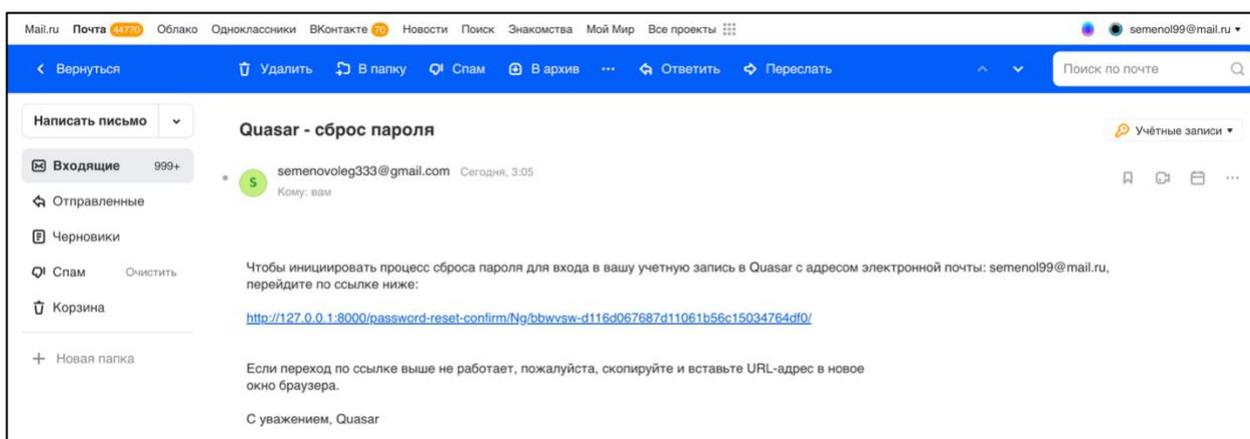


Рисунок 39 – Сообщение о сбросе пароля

### 3.5.4. Панель администратора

Как уже говорилось ранее, Django обладает готовой панелью администратора. Для реализации вариантов использования администратора, описанных в диаграмме на рисунке 7, в файлах **admin.py** в каждом из приложений были прописаны модели, которые должны быть доступны в панели администратора, а также ограничения на действия с ними. С помощью этого инструмента можно производить модерацию контента авторов.

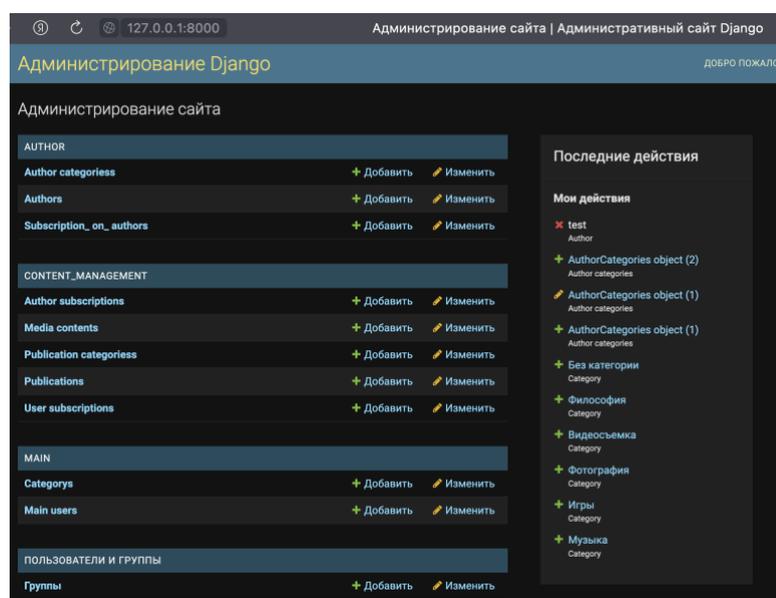


Рисунок 40 – Панель администратора

## **4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение**

### **4.1. Введение**

Главной идеей данной выпускной квалификационной работы является разработка онлайн-платформы платной подписки на контент. Платформа предназначена для предоставления сервиса, в котором пользователи могут становиться авторами и делиться с другими пользователями контентом, а заинтересованные пользователи могут следить за авторами и платить им за контент, оформляя подписки или оплачивая контент отдельно. Разрабатывается как веб-приложение с целью обеспечения кроссплатформенности. Проект разрабатывается одним студентом-разработчиком.

Целью данного раздела является определение перспективности и успешности, оценка эффективности, определение рисков и стратегий их устранения, формирование состава работ и бюджета проекта.

Для достижения поставленной цели были сформулированы следующие задачи:

1. Оценить коммерческий потенциал и перспективность разработки проекта.
2. Провести оценку научно-технического уровня исследования и оценку рисков.
3. Составить план работ по реализации проекта.
4. Рассчитать бюджет проекта.

## **4.2. Оценка коммерческого потенциала и перспективности разработки**

### **4.2.1. Потенциальные потребители результатов исследования**

Как уже было сказано выше, потенциальными потребителями платформы являются авторы контента и их подписчики. Авторов контента можно поделить на множество подкатегорий по типу контента, который они производят. Это могут быть различные художники, музыканты, фотографы, программисты, писатели, программисты и многие другие. При этом пользователи платформы в любой момент могут создать страницу автора и начать делиться своим контентом.

### **4.2.2. Анализ конкурентных технических решений**

В данный момент на рынке существуют несколько продуктов, как зарубежных, так и отечественных, предоставляющих такую же функциональность. Но, так как зарубежные сервисы полностью или частично ограничены для пользователей из России, в этом разделе они рассматриваться не будут. Из российских сервисов для анализа были выбраны два наиболее популярных платформы: Boosty и Sponsr [8, 9].

Для наглядной оценки конкурентоспособности разрабатываемой платформы, была построена карта сравнения конкурентных технических решений, которая представлена в таблице ниже. Индекс «ф» обозначает разрабатываемую платформу, индекс «к1» – платформу «Boosty», а индекс «к2» – платформу «Sponsr».

При сравнении конкурентных технических решений, рассматривались только функции, на которые был сделан упор при разработке данной платформы.

Таблица 1 – Оценочная карта для сравнения конкурентных технических решений

<b>Критерии оценки</b>	<b>Вес критерия</b>	<b>Баллы</b>	<b>Конкурентоспособность</b>
------------------------	---------------------	--------------	------------------------------

		Б <sub>ф</sub>	Б <sub>к1</sub>	Б <sub>к2</sub>	К <sub>ф</sub>	К <sub>к1</sub>	К <sub>к2</sub>
1	2	3	4	5	6	7	8
<b>Технические критерии оценки ресурсоэффективности</b>							
1. Доступность к ведению авторской деятельности	0.2	10	10	5	2	2	1
2. Понятность и удобство пользовательского интерфейса	0.2	6	9	6	1,2	1,8	1,2
3. Возможность общения с другими пользователями	0.1	5	3	3	0,5	0,3	0,3
4. Настройка доступа к контенту в подписках	0.2	8	0	6	1,6	0	1,2
5. Функция поиска авторов	0.1	7	1	8	0,7	0,1	0,8
<b>Экономические критерии оценки эффективности</b>							
1. Размер комиссии	0.2	7	6	5	1,4	1,2	1
<b>Итого</b>	<b>1</b>	<b>43</b>	<b>29</b>	<b>33</b>	<b>7,4</b>	<b>5,4</b>	<b>5,5</b>

Из данных оценочной карты можно сделать вывод, что разрабатываемая платформа является достаточно конкурентоспособной. В качестве сильных сторон платформы выделяются: легкий доступ к ведению авторской деятельности, возможность общения с другими пользователями, возможность настройки доступа к контенту внутри подписок и поиск авторов.

#### 4.2.3. SWOT-анализ

SWOT-анализ позволяет выделить сильные и слабые стороны продукта, определить его потенциал и возможные угрозы.

Процесс анализа состоит из двух этапов: первый, он же начальный, необходим для выявления достоинств и недостатков продукта, его возможностей и угроз, второй (завершающий) для определения соотношений всех фактов, выявленных на первом этапе.

Результаты анализа представлены в таблице ниже.

Таблица 2 – Матрица SWOT-анализ

	<b>Сильные стороны научно-исследовательского проекта:</b> С1. Возможность настройки доступа к устаревшему	<b>Слабые стороны научно-исследовательского проекта:</b> Сл1. Низкая скорость развития проекта (один
--	--	---

	контенту в подписке для новых подписчиков С2. Функция поиска авторов С3. Чат	разработчик). Сл2. Низкая производительность серверов в связи с отсутствием финансирования Сл3. Отсутствие коммерческого опыта реализации проекта
<b>Возможности:</b> В1. Привлечение российских пользователей с зарубежных платформ, которые ограничили им доступ В2. Рост производительности за счет использования высокопроизводительных серверов В3. Увеличение скорости развития проекта за счет найма разработчиков	Новых авторов заинтересуют уникальные функциональные возможности, позволяющие им более тонко настраивать свои подписки. Новые пользователи могут больше времени проводить на платформе, так как им не нужно переходить в другое приложение для общения, а также они могут легко искать интересующих их авторов или новых авторов по интересующей их тематике также не покидая платформы. С ростом количества пользователей, пропорционально будет расти заработок платформы, что позволит использовать более производительные сервера и нанять разработчиков.	Есть риск, что развитие платформы не будет успевать за ростом количества пользователей, что в дальнейшем оттолкнет всю аудиторию проекта и не будет привлекать новую.
<b>Угрозы:</b> У1. Развитие конкурирующих платформ У2. Появление превосходящего по большинству аспектов конкурентного продукта. У3. Снятие ограничений зарубежными платформами для российских пользователей	Функции, предлагаемые разрабатываемой платформой, могут привлечь пользователей и конкурировать даже с развитыми проектами.	Наибольшим риском является проигрыш конкурентной борьбы из-за недостаточно быстрого развития и большой имеющейся аудитории у конкурентов.

Исходя из результатов SWOT-анализа можно сделать вывод, что сильные стороны и возможности разрабатываемой платформы перевешивают слабые стороны и угрозы, что делает ее разработку целесообразной и актуальной на данный момент.

### 4.3. Планирование работ по научно-техническому исследованию

#### 4.3.1. Структура работ в рамках научного исследования

Перечень этапов и работ, распределение исполнителей по данным видам работ представлен в таблице ниже.

Таблица 3 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ раб	Содержание работ	Должность исполнителя
Исследование предметной области	1	Обзор конкурентных платформ	Семенов О. А.
	2	Определение потребностей стейкхолдеров	Семенов О. А.
Проектирование	3	Проектирование функций	Семенов О. А.
	4	Проектирование базы данных	Семенов О. А.
	5	Выбор инструментов для разработки	Семенов О. А.
	6	Проектирование архитектуры платформы	Семенов О. А.
Разработка	7	Разработка серверной части	Семенов О. А.
	8	Разработка клиентской части	Семенов О. А.
Тестирование	9	Проведение закрытого тестирования разработанной платформы	Семенов О. А.
Оформление отчета	10	Составление пояснительной записки	Семенов О. А.

#### 4.3.2. Определение трудоемкости выполнения работ

Трудоемкость выполнения научного исследования оценивается экспертным путем в человеко-днях и носит вероятностный характер, который зависит от множества трудно учитываемых факторов. Для определения ожидаемого (среднего) значения трудоемкости  $t_{ожі}$  используется следующая формула:

$$t_{ожi} = \frac{3t_{мини} + 2t_{маxi}}{5}, \quad (1)$$

где  $t_{ожi}$  – ожидаемая трудоемкость выполнения  $i$ -ой работы чел.-дн.;

$t_{мини}$  – минимально возможная трудоемкость выполнения заданной  $i$ -ой работы, чел.-дн.;

$t_{маxi}$  – максимально возможная трудоемкость выполнения заданной  $i$ -ой работы, чел.-дн.;

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях  $T_p$ , учитывающая параллельность выполнения работ по нескольким исполнителями.

$$T_{pi} = \frac{t_{ожi}}{Ч_i}, \quad (2)$$

где  $T_{pi}$  – продолжительность одной работы, раб.дн.;

$t_{ожi}$  – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$  – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

#### 4.3.2.1. Разработка графика проведения научного исследования

Согласно производственному календарю (для 6-дневной рабочей недели) в 2023 году 365 календарных дней, 247 рабочих дней, 118 выходных/праздничных дней. Таким образом, коэффициент календарности на 2023 год равен:

$$k_{кал} = \frac{T_{кал}}{T_{кал} - (T_{вых} + T_{пр})} = \frac{365}{365 - 118} = 1.48 \quad (3)$$

Используя полученный коэффициент календарности можно составить таблицу временных показателей проведения научного исследования, представленную ниже.

Таблица 4 – Временные показатели проведения научного исследования

Название работы	Трудоёмкость работ	Исполнители работы	Длительность работ в	Длительность работ в

	$T_{min}$	$T_{max,}$	$T_{ож}$		рабочих днях $T_{pi}$	календарных днях $T_{ki}$
Обзор конкурентных платформ	2	3	2,4	Семенов О. А.	3	4
Определение потребностей стейкхолдеров	1	2	1,6	Семенов О. А.	2	3
Проектирование функций	20	25	22	Семенов О. А.	22	33
Проектирование базы данных	6	8	6,8	Семенов О. А.	7	10
Выбор инструментов для разработки	2	3	2,4	Семенов О. А.	3	4
Проектирование архитектуры платформы	5	6	5,4	Семенов О. А.	6	9
Разработка серверной части	30	35	32	Семенов О. А.	32	47
Разработка клиентской части	20	25	22	Семенов О. А.	22	33
Проведение закрытого тестирования разработанной платформы	10	12	10,8	Семенов О. А.	11	16
Составление пояснительной записки	14	21	16,8	Семенов О. А.	17	25

С помощью показателей, приведенных в таблице выше, был разработан календарный план-график проведения научного исследования, для иллюстрации которого была использована диаграмма Ганта, представленная

ниже. Ориентировочные даты выполнения работы: с 9 января 2023 года по 16 мая 2023 года.

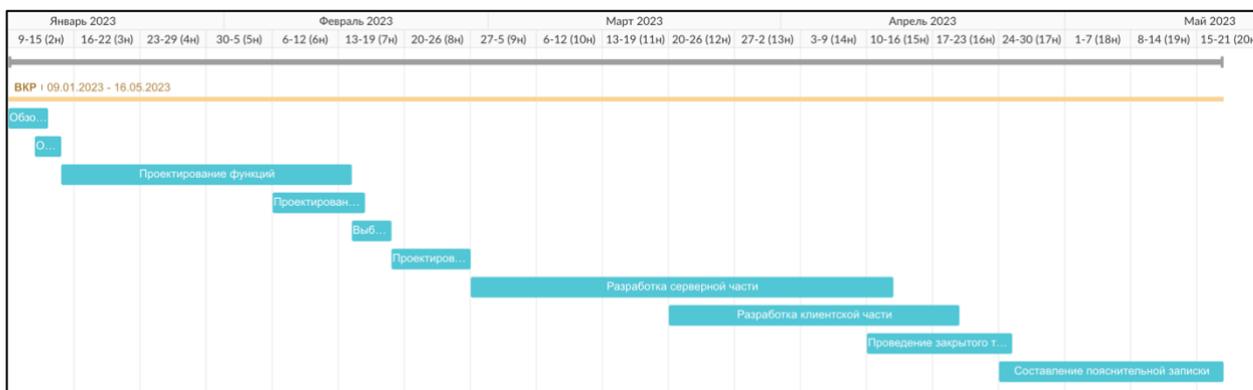


Рисунок 41 – Диаграмма Ганта

#### 4.4. Бюджет научно-технического исследования

##### 4.4.1. Расчет материальных затрат НТИ

Материальные затраты отсутствуют в связи проведения работ в офисе.

##### 4.4.2. Расчет затрат на специальное оборудование для научных работ

При выполнении работы использовался только один ноутбук стоимостью 53 тыс. руб.

##### 4.4.3. Основная заработная плата исполнителя

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{\text{дн}} = \frac{Z_{\text{м}} \cdot M}{F_{\text{д}}} \quad (4)$$

где  $Z_{\text{м}}$  – месячный должностной оклад работника, руб.;

$M$  – количество месяцев работы без отпуска в течение года:

при отпуске в 48 раб. дней  $M=10,4$  месяца, 6–дневная неделя;

$F_{\text{д}}$  – действительный годовой фонд рабочего времени научно–технического персонала, раб. дн.

Расчет баланса рабочего времени приведен в таблице ниже.

Таблица 5 – Баланс рабочего времени (для 6-тидневной рабочей недели)

Показатели рабочего времени	Количество дней
Календарное число дней	365
Количество нерабочих дней - выходные дни - праздничные дни	118
Потери рабочего времени - отпуск - невыходы по болезни	48 0
Действительный годовой фонд рабочего времени	199

Исходя из полученных данных, среднедневная заработная плата будет следующей:

$$Z_{\text{дн}} = \frac{40000 \cdot 10,4}{199} = 2090,45 \text{ руб.}$$

Для расчета основной заработной платы используется формула:

$$Z_{\text{м}} = Z_{\text{дн}} \cdot (1 + k_{\text{пр}} + k_{\text{д}}) \cdot k_{\text{р}} \cdot T_{\text{р}} \quad (5)$$

В таблице ниже приведен расчет основной заработной платы на период выполнения научно-исследовательского проекта.

Таблица 6 – Расчет основной заработной платы

Исполнители	$k_{\text{пр}}$	$k_{\text{д}}$	$k_{\text{р}}$	$Z_{\text{дн}}$ , руб.	$T_{\text{р}}$ , раб. дн.	$Z_{\text{осн}}$ , руб.
Семенов О. А.	0,3	0,2	1,3	2090,45	92	375026,73
Итого						375026,73

#### 4.4.4. Расчет дополнительной заработной платы

Дополнительная заработная плата учитывает величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда, а также выплат, связанных с обеспечением гарантий и компенсаций (при исполнении государственных и общественных обязанностей, при совмещении работы с обучением, при предоставлении

ежегодного оплачиваемого отпуска и т.д.). Величина надбавочного коэффициента  $k_{\text{доп}}$  принята за 0,12.

Расчет дополнительной заработной платы:

$$Z_{\text{доп}} = k_{\text{доп}} \cdot Z_{\text{осн}} = 0,12 \cdot 375026,73 = 45003,21 \quad (6)$$

#### **4.4.5. 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:

$$Z_{\text{внеб}} = k_{\text{внеб}} \cdot (Z_{\text{осн}} + Z_{\text{доп}}) \quad (7)$$

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

В соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30,2%.

Величина отчислений во внебюджетные фонды составляет:

$$Z_{\text{внеб}} = 0,302 \cdot (375026,73 + 45003,25) = 126849,05 \text{ руб.}$$

#### **4.4.6. Накладные расходы**

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов. Их величина определяется по формуле:

$$Z_{\text{накл}} = (\sum \text{статей}) \cdot k_{\text{нр}} \quad (8)$$

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

Величина коэффициента накладных расходов принимается равной 16%.

Накладные расходы для исполнения составили:

$$\begin{aligned} Z_{\text{накл}} &= (0 + 53000 + 375026,73 + 45003,21 + 126849,09) \cdot 0,16 \\ &= 95980,64 \text{ руб.} \end{aligned}$$

#### **4.4.7. Формирование бюджета затрат научно-исследовательского проекта**

Рассчитанная величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта. Определение бюджета затрат на научно-исследовательский проект приведено в таблице ниже.

Таблица 7 – Расчет бюджета затрат НИИ

Наименование статьи	Сумма, руб.	Примечание
Материальные затраты НИИ	0	Пункт 1.3.1
Затраты на специальное оборудование для научных работ	53000	Пункт 1.3.2
Затраты по основной заработной плате исполнителя темы	375026,73	Пункт 1.3.3
Затраты по дополнительной заработной плате исполнителя темы	45003,21	Пункт 1.3.4
Отчисления во внебюджетные фонды	126849,05	Пункт 1.3.5
Накладные расходы	95980,64	Пункт 1.3.6
Бюджет затрат НИИ	695859,63	

#### **4.5. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной 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. При расчете показателей эффективности оценивались 2 варианта исполнения системы:

1. Разработка с использованием фреймворка Django.
2. Разработка с использованием фреймворка React.

Стоимость разработки в обоих случаях будет являться практически равной, в связи с чем отдельный расчет для второго варианта не производился. Исходя из этого, интегральный финансовый показатель для каждого из вариантов будет принят равным единице.

Первый вариант является наиболее выгодным непосредственно для разработчика, так как требует знания языка программирования Python, в то время как для второго варианта исполнения необходимо знание языка программирования JavaScript, что в свою очередь потребует дополнительных временных трат от разработчика.

Расчет интегральных показателей ресурсоэффективности обоих вариантов исполнения приведен в таблице ниже.

Таблица 8 – Сравнительная оценка характеристик вариантов исполнения проекта

Критерии	Объект исследования	Весовой коэффициент параметра	Исп.1	Исп.2
1. Время необходимое на разработку		0,2	5	3
2. Безопасность		0,2	5	3
3. Простота изучения		0,15	3	4
4. Возможность разработки больших функциональных приложений		0,15	5	3
5. Развертывание		0,1	5	4
6. Масштабируемость		0,2	5	3
Итого		1	4,7	3,25

Интегральный показатель эффективности вариантов исполнения разработки ( $I_{испi}$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{испi} = \frac{I_{р-испi}}{I_{фин.р}^{испi}} \quad (9)$$

Так как интегральный финансовый показатель ( $I_{фин.р}^{испi}$ ) равен единице для обоих вариантов, то интегральные показатели эффективности будут иметь те же значения, что и интегральные показатели ресурсоэффективности ( $I_{р-испi}$ ).

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Сравнительная эффективность проекта ( $\mathcal{E}_{ср}$ ) определяется по формуле:

$$\mathcal{E}_{ср} = \frac{I_{исп2}}{I_{исп1}} \quad (10)$$

Расчет сравнительной эффективности проекта приведен в таблице ниже.

Таблица 9 – Сравнительная эффективность разработки

№	Показатели	Исп.1	Исп.2
1	Интегральный финансовый показатель разработки	1	1
2	Интегральный показатель ресурсоэффективности разработки	4,7	3,25
3	Интегральный показатель эффективности	4,7	3,25
4	Сравнительная эффективность вариантов исполнения	1	0,691

Сравнив значения интегральных показателей эффективности, можно сделать вывод, что реализация технологии в первом исполнении является более эффективным вариантом решения задачи, поставленной в данной работе с позиции финансовой и ресурсной эффективности.

#### **4.6. Вывод по разделу**

В ходе выполнения исследований в разделе финансового менеджмента, ресурсоэффективности и ресурсосбережения, была дана оценка коммерческого потенциала и перспективности разработки онлайн-платформы, сформирован график работ и бюджет затрат в следствии чего была определена эффективность исследования.

Был проведен SWOT-анализ, который показал, что разработка проекта является актуальной и целесообразной.

В результате планирования научно-исследовательских работ, для наглядной визуализации графика, построена диаграмма Ганта. Ориентировочные даты выполнения работы: с 09.01.2023 по 16.05.2023 года, что в сумме составляет 92 календарных дня.

Результатом расчета расходов стал сформированный бюджет затрат научного исследования, который составил 695859,63 руб.

При определении и сравнении интегральных показателей эффективности вариантов исполнения было выявлено, что наиболее выгодным является первый вариант исполнения, а именно разработка онлайн-платформы с использованием фреймворка Django, который и был реализован в данной работе.

## **5. Социальная ответственность**

### **5.1. Введение**

В рамках данной выпускной квалификационной работы была разработана онлайн-платформа платной подписки на контент. Платформа предназначена для предоставления сервиса, в котором пользователи могут становиться авторами и делиться с другими пользователями контентом, а заинтересованные пользователи могут следить за авторами и платить им за контент, оформляя подписки или оплачивая контент отдельно. Разрабатывается как веб-приложение с целью обеспечения кроссплатформенности.

Целью разработки платформы является решение проблемы необходимости монетизации контента авторов.

Основными факторами, подтверждающими актуальность разработки, являются: прекращение монетизации для российских пользователей зарубежными аналогичными платформами и недостаточный функционал или ограниченный доступ к нему у отечественных аналогов.

Пользователями разработанной платформы являются авторы контента (художники, фотографы, блогеры, научные деятели и т. д.) и их читатели.

Разработка платформы проводилась в ходе преддипломной практики в помещении с площадью 28 м<sup>2</sup>. В помещении расположена рабочая зона с ноутбуком, с помощью которого осуществлялся поиск и анализ информации, а также проектирование и разработка платформы.

## **5.2. Правовые и организационные вопросы обеспечения безопасности**

### **5.2.1. Правовые нормы трудового законодательства**

Трудовые отношения между работодателем и работником регулируются в соответствии с законодательным актом: «Трудовой кодекс Российской Федерации» от 30.12.2001 N 197-ФЗ (ред. от 19.12.2022, с изм. от 11.04.2023) [26]. Данный документ регламентирует права и обязанности работодателя и работника, режим труда и отдыха, оплату и нормирование труда, компенсации работникам, защиту персональных данных работника, урегулирование трудовых споров и вопросы организации труда. Ниже приведены наиболее важные нормы, применимые к условиям выполнения данной выпускной квалификационной работы:

Нормальная продолжительность рабочего времени не может превышать 40 часов в неделю.

- В течение рабочего дня (смены) работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут, который в рабочее время не включается. Правилами внутреннего трудового распорядка или трудовым договором может быть предусмотрено, что указанный перерыв может не предоставляться работнику, если установленная для него продолжительность ежедневной работы (смены) не превышает четырех часов. (в ред. Федерального закона от 18.06.2017 N 125-ФЗ).
- Всем работникам предоставляются выходные дни (еженедельный непрерывный отдых). При пятидневной рабочей неделе работникам предоставляются два выходных дня в неделю, при шестидневной рабочей неделе - один выходной день.

Работа инженера-программиста по виду трудовой деятельности относится к группе В – творческая работа в режиме диалога с ЭВМ, а по напряженности работы к III категории тяжести [27]. В следствии чего рекомендуется организовывать перерывы длительностью 10 – 15 минут через каждые 45-60 минут работы.

### **5.2.2. Эргономические требования к правильному расположению и компоновке рабочей зоны**

В качестве основных элементов рабочей зоны программиста, выступают: нерегулируемый рабочий стол, рабочее регулируемое кресло и ноутбук без отдельной клавиатуры и компьютерной мыши.

Согласно ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [28], ниже приведены наиболее важные требования применимые к условиям выполнения данной выпускной квалификационной работы:

- Рабочее место с дисплеем должно обеспечивать оператору возможность удобного выполнения работ в положении сидя и не создавать перегрузки костно-мышечной системы.
- Конструкция рабочего стола должна обеспечивать возможность размещения на рабочей поверхности необходимого комплекта оборудования и документов с учетом характера выполняемой работы.
- Высота рабочей поверхности стола при нерегулируемой высоте должна составлять 725 мм.
- Размеры рабочей поверхности стола должны быть: глубина - не менее 600 мм, ширина - не менее 1200 мм.
- Рабочий стул должен быть подъемно-поворотным и регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья.

- Дисплей на рабочем месте оператора должен располагаться так, чтобы изображение в любой его части было различимо без необходимости поднять или опустить голову.
- Дисплей на рабочем месте должен быть установлен ниже уровня глаз оператора. Угол наблюдения экрана оператором относительно горизонтальной линии взгляда не должен превышать 60°.

Элементы рабочей зоны, в которой выполняется выпускная квалификационная работа удовлетворяют вышеперечисленным требованиям.

### 5.3. Производственная безопасность

Условия труда, в которых выполнялась данная выпускная квалификационная работа, а также элементы рабочей зоны инженера-программиста могут спровоцировать появление вредных и опасных факторов производства. Все возможные выявленные факторы указаны в таблице ниже.

Таблица 10 – Возможные опасные и вредные производственные факторы на рабочем месте инженера-программиста

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
1. Умственное перенапряжение, в том числе вызванное информационной нагрузкой.	МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [29]
2. Эмоциональное перенапряжение.	ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [30]
3. Отсутствие или недостаток	СП 52.13330.2016 «Естественное и искусственное освещение» [31]

необходимого искусственного освещения	
4. Статические физические перегрузки, связанные с рабочей позой.	ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [30]
5. Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий	ГОСТ Р 12.1.019-2017 «Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты» [32]

Производственные факторы считаются вредными в тех случаях, когда их воздействие на человека, который работает в определенных условиях, приводят к снижению работоспособности или заболеванию. А опасными факторами считаются те факторы, которые приводят к травмам, резкому ухудшению здоровья или смерти работающего в определенных условиях человека.

### **5.3.1. Умственное перенапряжение, в том числе вызванное информационной нагрузкой и эмоциональное перенапряжение**

Работа программиста заключается в постоянном решении сложных и нестандартных задач. Часто, при разработке, встречаются проблемы, для разрешения, которых необходимо затрачивать много сил на поиск и анализ информации. Все это может приводить как к умственному, так и к эмоциональному перенапряжению (переутомлению). Такие перенапряжения могут привести к следующим негативным последствиям:

- головная боль;
- нарушение сна или сонливость;

- усталость, апатия;
- раздражительность, нервозность;
- медлительность;
- снижение внимательности и работоспособности;
- нарушения аппетита;
- приступы тошноты, обмороки.

Для снижения уровня умственного и эмоционального перенапряжений программиста следует принимать следующие меры:

- делать регулярные перерывы во время рабочего дня;
- нормализовать режим сна;
- нормализовать питание;
- больше времени проводить на свежем воздухе;
- уделять время физическим нагрузкам (разминке, тренировкам);
- разбивать сложные, объемные задачи на подзадачи;
- не заикливаться на сложных задачах.

Согласно МР 2.2.9.2311-07, меры профилактики стрессовых состояний предусматривают внедрение рациональных режимов труда и отдыха, комплекса оздоровительно-профилактических мероприятий для предупреждения воздействия стресс-факторов на организм работающих.

При постоянном взаимодействии с компьютером (набор текстов, ввод данных и т.п.), при исключении возможности периодического переключения на другие виды трудовой деятельности, не связанные с ПЭВМ, рекомендуется организация перерывов по 10 - 15 мин. через каждые 45 - 60 мин. работы.

Для повышения работоспособности, снижения зрительного утомления рекомендуется использовать очки защитные со спектральными фильтрами ЛС и НСФ, разрешенные Минздравом России для работы с ПЭВМ.

### 5.3.2. Отсутствие или недостаток необходимого искусственного освещения

Недостаточна освещенность рабочей зоны является вредным производственным фактором, который приводит к быстрому утомлению, снижению работоспособности, внимания, концентрации и ухудшению зрения работника.

Нормы освещения регламентируются документом: СП 52.13330.2016 «Естественное и искусственное освещение». Разработка программного обеспечения относится к работам высокой точности (наименьший или эквивалентный объект различения 0,30 – 0,50 мм), разряд Б, подразряд 1, относительная продолжительность зрительной работы при направлении зрения на рабочую поверхность не менее 70%.

В таблице ниже представлены требования к искусственному освещению рабочего помещения.

Таблица 11 – Требования к искусственному освещению рабочего помещения

Искусственное освещение				
Освещенность на рабочей поверхности системы общего освещения, лк	на от общего освещения, лк	Цилиндрическая освещенность, лк	Объединенный показатель UGR, не более	Коэффициент пульсации освещенности, Кп, %, не более
400		100	19	15

Искусственное освещение в помещениях для эксплуатации ПК должно осуществляться системой общего равномерного освещения. В случаях преимущественной работы с документами, следует применять системы комбинированного освещения (к общему освещению дополнительно устанавливаются светильники местного освещения).

Уровень освещения рабочего пространства должен приблизительно совпадать с яркостью дисплея, это необходимо для снижения влияния

вредного фактора недостаточной освещенности, так как яркий свет в зоне периферийного зрения заметно увеличивает напряжение глаз.

### **5.3.3. Статические физические перегрузки, связанные с рабочей позой**

Так как работа программиста является малоподвижной, что уже само по себе пагубно влияет на здоровье работника, то большое влияние на него также оказывают и параметры мебели, за которой он работает. Под параметрами мебели имеются ввиду нерациональная высота рабочей поверхности стола и сидения, отсутствие опорной спинки и подлокотников, неправильный угол наклона головы и экрана и т. д. Все эти эргономические неудобства приводят к нарушениям в работе опорно-двигательного аппарата и периферической нервной системы.

Длительное нахождение в условиях недостаточной физической активности также способствует преждевременному утомлению, снижению работоспособности и концентрации.

Для профилактики вышеописанных статических перегрузок следует предпринимать следующие меры:

- делать регулярные перерывы во время рабочего дня для разминок;
- увеличить количество физических активностей;
- правильно обустроить рабочее пространство.

### **5.3.4. Производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий**

Так как работа программиста всецело связана с компьютером, следовательно происходит в близости электрических сетей и приборов, от работника требуется осторожность в обращении с электричеством.

Поражение электрическим током является одним из наиболее опасных видов вредного воздействия из-за того, что органы чувств человека не могут обнаружить наличие электрического тока.

На вероятность поражения электрическим током влияют:

- высокая влажность воздуха в помещении (выше 75%);
- высокая температура воздуха и поверхностей (выше 35 °С);
- неправильное проектирование рабочего места;
- отсутствие защитных конструкций для проводов;
- наличие посторонних предметов на электроприборах;
- наличие токопроводящей пыли.

Даже при условии, что вышеперечисленные факторы не относятся к помещению, в котором выполняется работа, есть риск возникновения следующих опасностей:

- прикосновение к токоведущим частям (во время ремонта ПК);
- прикосновение к нетоковедущим частям, оказавшимся под напряжением (из-за нарушения изоляции);
- соприкосновение с полом или стенами, оказавшимся под напряжением (из-за нарушения электрической сети);
- короткое замыкание в высоковольтных блоках.

Место выполнения работы не относится к помещениям повышенной опасности поражения электричеством. В помещении используются стандартные бытовые приборы, потребляющие напряжение 220 В переменного тока с частотой 50 Гц.

Мерами защиты от воздействия электрического тока при неисправности изоляции являются: защитное заземление, зануление и использование устройств защитного отключения (УЗО) [32].

#### **5.4. Экологическая безопасность**

При разработке онлайн-платформы предполагаемым источником загрязнения выступает ноутбук, на котором производилась работа. При выходе из строя ПЭВМ подлежит утилизации 4 класса опасности. При утилизации компьютерного оборудования происходит обязательное извлечение компонентов, их сортировка и последующая отправка для повторного использования. При такой утилизации более 90% компонентов отправится на вторичную переработку и менее 10% будут отправлены на свалки. Такая утилизация обязательно должна производиться на специально оборудованных полигонах.

Процедура утилизации должна проводиться в соответствии с ГОСТ Р 53692-2009 «Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов» [33].

#### **5.5. Безопасность в чрезвычайных ситуациях**

Возможные ЧС:

- природные катастрофы (наводнения, ураганы, лесные пожары и т.д.);
- геологические воздействия (землетрясения, провалы территории и т.д.);
- техногенные (пожары, аварии с выбросом химически/радиоактивно опасных веществ и т.д.);
- биолого-социальные (пандемия)

При работе программного инженера пожар является наиболее вероятной чрезвычайной ситуацией. Наиболее частыми причинами возникновения пожара являются короткое замыкание, неисправность оборудования, перегрузка сетей с последующим нагревом токоведущих частей, искрение и статическое электричество.

Для предотвращения возникновения пожара необходимо:

- Соблюдать правила и нормы при монтаже электрических приборов и электрической проводки.
- Оборудовать рабочее помещение пожарной сигнализацией и средствами тушения пожара.
- Иметь в помещении план эвакуации и список действий, предпринимаемых при возникновении пожара.

Возможный пожар, на основании Федерального закона от 22.07.2008 N 123-ФЗ в зависимости от горючих объектов вероятнее будет соответствовать классам А1, А2 Е [34]. Первичными средствами тушения пожара являются: огнетушители порошковые переносные с порошками типа АВСЕ, огнетушители углекислотные.

В случае возникновения пожара необходимо:

- отключить электроэнергию;
- незамедлительно сообщить об этом по телефону в пожарную часть;
- принять меры для оповещения других людей в здании о пожаре;
- принять меры по эвакуации согласно плану эвакуации.

## **5.6. Выводы по разделу**

В ходе работы по разделу «Социальная ответственность» были выявлены основные особенности трудового законодательства применимые к данной выпускной квалификационной работе, а также эргономические требования к правильному расположению и компоновке рабочей зоны. Был проведен анализ вредных и опасных факторов, которые могут возникать при проведении работ программного инженера. Также были выявлены предполагаемые источники загрязнения окружающей среды, возникающие в результате разработки программного продукта и возможные чрезвычайные ситуации, которые могут возникнуть при разработке.

Исходя из исследований, перечисленных выше, можно сделать следующие выводы:

Согласно ПУЭ, помещение, в котором проводилась работа, можно классифицировать как помещение без повышенной опасности [27]. Согласно «Правилам по охране труда при эксплуатации электроустановок» персонал должен иметь первую группу по электробезопасности [36].

Работа в офисе относится к категории тяжести труда Ia– работы выполняются при оптимальных условиях внешней производственной среды и при оптимальной величине физической, умственной и нервно-эмоциональной нагрузки [37].

Рабочая зона, согласно СП 12.13130.2009, в связи с наличием горючих материалов и веществ в рабочем помещении, относится к категории В по взрывопожарной и пожарной опасности [35].

Элементы рабочей зоны, в которой выполняется выпускная квалификационная работа удовлетворяют требованиям, представленным в ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [28].

При разработке негативного воздействия на окружающую среду не производится, оно возможно только в случае утилизации ПЭВМ, на котором выполняется работа, но минимизируется при утилизации в соответствии с ГОСТ Р 53692-2009 [33].

При выполнении ВКР, нарушений по организации рабочего процесса не выявлено, а все необходимые нормы и требования безопасности соблюдены.

## Заключение

В результате выполнения выпускной квалификационной работы была спроектирована и разработана платформа платной подписки на контент, целью которой является решение проблемы монетизации контента авторов.

В процессе выполнения работы были решены следующие задачи:

1. произведено исследование существующих платформ и сервисов платной подписки на контент, в результате которого был проведен обзор конкурентных решений, определены основные стейкхолдеры и их потребности, а также выявлена основная проблема исследуемой области и ее источники;
2. осуществлено проектирование собственной платформы, в ходе которого были определены основные пользовательские роли и их функции, спроектирована база данных, определена архитектура системы, и выбраны программные инструменты для разработки;
3. осуществлена программная реализация веб-платформы в соответствии со спроектированными функциями (код реализации представлен по ссылке: [https://github.com/manyfaces-coder/platform\\_quasar](https://github.com/manyfaces-coder/platform_quasar)).

Также был произведен финансовый менеджмент, в ходе которого был определен план разработки, ее бюджет и коммерческий потенциал.

В главе, посвященной социальной ответственности, были рассмотрены организационно-правовые, производственные и экологические вопросы обеспечения безопасности, применимые к выполнению данной работы.

## Список использованных источников

1. Интернет и соцсети в начале 2023 года — главные цифры Global Digital 2023 [Электронный ресурс] // – Режим доступа: <https://vc.ru/marketing/596126-internet-i-socseti-v-nachale-2023-goda-glavnye-cifry-global-digital-2023> (дата обращения: 18.03.2023).
2. 10 сервисов для приёма донатов [Электронный ресурс] // – Режим доступа: <https://www.unisender.com/ru/blog/idei/servisy-dlya-priema-donatov/> (дата обращения: 18.03.2023).
3. 9 альтернативных сервисов для монетизации контента блогеров и инфлюенсеров в условиях санкций [Электронный ресурс] // – Режим доступа: <https://rb.ru/opinion/monetizirovat-kontent/> (дата обращения: 18.03.2023).
4. Как заработать на создание контента на OnlyFans, Sponsr.ru, Patreon и Boosty [Электронный ресурс] // – Режим доступа: [https://f.partnerkin.com/c/business\\_sozdanie\\_kontenta\\_na\\_onlyfans\\_spons\\_patreon\\_boosty](https://f.partnerkin.com/c/business_sozdanie_kontenta_na_onlyfans_spons_patreon_boosty) (дата обращения: 18.03.2023).
5. Чем заменить заблокированный Patreon? [Электронный ресурс] // – Режим доступа: <https://digitalocean.ru/n/zavedi-sebe-sponsora> (дата обращения: 18.03.2023).
6. Официальный сайт Patreon [Электронный ресурс] // – Режим доступа: <https://www.patreon.com> (дата обращения: 18.03.2023).
7. Официальный сайт Onlyfans [Электронный ресурс] // – Режим доступа: <https://onlyfans.com> (дата обращения: 18.03.2023).
8. Официальный сайт Boosty [Электронный ресурс] // – Режим доступа: <https://boosty.to> (дата обращения: 18.03.2023).
9. Официальный сайт Sponsr [Электронный ресурс] // – Режим доступа: <https://sponsr.ru> (дата обращения: 18.03.2023).

- 10.24 Curious Patreon Statistics to Show How Big It Is In 2023 [Электронный ресурс] // – Режим доступа: <https://techjury.net/blog/patreon-statistics/> (дата обращения: 18.03.2023).
- 11.18 Shocking OnlyFans Statistics to Show How Big It Is [2023] [Электронный ресурс] // – Режим доступа: <https://techjury.net/blog/onlyfans-statistics/> (дата обращения: 18.03.2023).
12. OnlyFans приостановила выплаты создателям контента из России [Электронный ресурс] // – Режим доступа: <https://habr.com/ru/news/t/662383/> (дата обращения: 18.03.2023).
13. OnlyFans vs Patreon vs Boosty: что лучше? [Электронный ресурс] // – Режим доступа: <https://storedigital.ru/2020/11/08/onlyfans-vs-patreon-vs-boosty-chto-luchshe/> (дата обращения: 18.03.2023).
14. Диаграмма Исикавы: зачем применять и как построить [Электронный ресурс] // – Режим доступа: <https://pmclub.pro/articles/diagramma-isikavy-zachem-primenyat-i-kak-postroit> (дата обращения: 18.03.2023).
15. Использование диаграммы вариантов использования UML при проектировании программного обеспечения [Электронный ресурс] // – Режим доступа: <https://habr.com/ru/articles/566218/> (дата обращения: 20.04.2023).
16. Учебное пособие по диаграммам последовательностей: полное руководство с примерами [Электронный ресурс] // – Режим доступа: <https://creately.com/blog/ru/диаграмма/учебное-пособие-по-последовательной/> (дата обращения: 20.04.2023).
17. Нотация Мартина (Crow's Foot) [Электронный ресурс] // – Режим доступа: [https://studme.org/77223/informatika/notatsiya\\_martina\\_crows\\_foot](https://studme.org/77223/informatika/notatsiya_martina_crows_foot) (дата обращения: 20.04.2023).
18. Документация Python [Электронный ресурс] // – Режим доступа: <https://www.python.org/doc/> (дата обращения: 20.04.2023).

19. Документация Django [Электронный ресурс] // – Режим доступа: <https://docs.djangoproject.com/en/4.1/> (дата обращения: 20.04.2023).
20. Документация PostgreSQL [Электронный ресурс] // – Режим доступа: <https://www.postgresql.org/docs/14/index.html> (дата обращения: 20.04.2023).
21. Возможности PyCharm [Электронный ресурс] // – Режим доступа: <https://www.jetbrains.com/ru-ru/pycharm/features/> (дата обращения: 20.04.2023).
22. Документация Stripe [Электронный ресурс] // – Режим доступа: <https://stripe.com/docs> (дата обращения: 20.04.2023).
23. Django MVT Architecture [Электронный ресурс] // – Режим доступа: <https://www.askpython.com/django/django-mvt-architecture> (дата обращения: 20.04.2023).
24. Простое руководство по диаграммам компонентов [Электронный ресурс] // – Режим доступа: <https://creately.com/blog/ru/uncategorized-ru/учебное-пособие-по-компонентной-диаг/> (дата обращения: 20.04.2023).
25. Простое руководство к диаграммам развертывания UML [Электронный ресурс] // – Режим доступа: <https://creately.com/blog/ru/uncategorized-ru/учебное-пособие-по-диаграмме-развёрт/> (дата обращения: 20.04.2023).
26. Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 25.02.2022) [Электронный ресурс] // – Режим доступа: [https://www.consultant.ru/document/cons\\_doc\\_LAW\\_34683/](https://www.consultant.ru/document/cons_doc_LAW_34683/) (дата обращения: 26.04.2023).
27. Правила устройства электроустановок [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200030216> (дата обращения: 26.04.2023).

- 28.ГОСТ Р 50923-96 «Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200025975> (дата обращения: 26.04.2023).
- 29.МР 2.2.9.2311-07 «Профилактика стрессового состояния работников при различных видах профессиональной деятельности» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200072234> (дата обращения: 26.04.2023).
- 30.ГОСТ 12.0.003-2015 «Система стандартов безопасности труда. Опасные и вредные факторы» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200136071> (дата обращения: 26.04.2023).
- 31.СП 52.13330.2016 «Естественное и искусственное освещение» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/456054197> (дата обращения: 26.04.2023).
- 32.ГОСТ Р 12.1.019-2017 «Система стандартов безопасности труда (ССБТ). Электробезопасность. Общие требования и номенклатура видов защиты» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200161238> (дата обращения: 26.04.2023).
- 33.ГОСТ Р 53692-2009 «Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200081740> (дата обращения: 26.04.2023).
- 34.Федеральный закон от 22.07.2008 N 123-ФЗ (ред. от 01.03.2023) «Технический регламент о требованиях пожарной безопасности» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/902111644> (дата обращения: 26.04.2023).

- 35.СП 12.13130.2009 «Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/1200071156> (дата обращения: 26.04.2023).
- 36.Приказ Минтруда России от 15 декабря 2020 года N 903н «Об утверждении Правил по охране труда при эксплуатации электроустановок» [Электронный ресурс] // – Режим доступа: <https://docs.cntd.ru/document/573264184> (дата обращения: 26.04.2023).
- 37.Приказ Минтруда России от 24.01.2014 N 33н (ред. от 27.04.2020) «Об утверждении Методики проведения специальной оценки условий труда, Классификатора вредных и (или) опасных производственных факторов, формы отчета о проведении специальной оценки условий труда и инструкции по ее заполнению» [Электронный ресурс] // – Режим доступа: [http://www.consultant.ru/document/cons\\_doc\\_LAW\\_158398/](http://www.consultant.ru/document/cons_doc_LAW_158398/) (дата обращения: 26.04.2023).

