

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего  
образования  
«Национальный исследовательский университет ИТМО»  
Факультет Программной инженерии и компьютерной техники

Лабораторная работа №1  
по дисциплине «Основы программной инженерии»

Вариант: **322004**

**Преподаватель:**  
Егошин Алексей Васильевич

**Выполнила:**  
Гафурова Фарангиз Фуркатовна  
**Группа:** Р3220

Санкт-Петербург, 2025

## Оглавление

<b>ЗАДАНИЕ</b> .....	3
<b>SOFTWARE REQUIREMENTS SPECIFICATION</b> .....	4
1. Introduction (Введение) .....	4
1.1 Purpose (Цель) .....	4
1.2 Project scope .....	4
1.3 Definitions, Acronyms and Abbreviations (Определения и аббревиатуры) .....	4
1.4 Intended Audience and Reading Suggestions (Целевая аудитория) .....	6
1.5 References (Ссылки) .....	6
1.6 Overview (Обзор документа) .....	6
2. Overall Description (Общее описание) .....	6
2.1 Product features (Функционал продукта) .....	6
2.2 User characteristics (Описание пользователей) .....	6
2.3 Operating environment (Операционная среда) .....	7
2.4 Design and implementation constraints .....	7
2.5 Assumptions and dependencies (Влияющие факторы и зависимости) .....	7
2.6 Constrains (Ограничения) .....	8
3. System features (Особенности системы) .....	8
3.1 Functional requirements (Функциональные требования) .....	8
4. Nonfunctional requirements (Нефункциональные требования) .....	12
4.1 Usability requirements (Требования к удобству использования) .....	12
4.2 Reliability requirements (Требования к надежности) .....	13
4.3 Performance requirements (Требования к производительности) .....	13
4.4 Licensing requirements (Лицензионные требования) .....	16
4.5 Safety requirements .....	16
4.6 Security requirements .....	16
5. External interface requirements (Требования к внешнему интерфейсу) .....	16
5.1 User Interfaces (Пользовательские интерфейсы) .....	16
5.2 Software Interfaces (Программные интерфейсы) .....	17
5.3 Communications Interfaces (Коммуникационные интерфейсы) .....	17
6. Requirements Attributes (Атрибуты требований) .....	17
7. Precedents (Прецеденты) .....	21
<b>APPENDIX A: ANALYSIS MODELS</b> .....	29
1. Use-Case-диаграммы прецедентов использования, реализующих функциональные требования. .....	29
<b>ВЫВОД</b> .....	32

# ЗАДАНИЕ

## Лабораторные работы

### Лабораторная работа #1

Вариант 322004

**Вариант №322004:** Rutube - лучшее видео) - <https://www.rutube.ru/>

Составить список требований, предъявляемых к разрабатываемому веб-сайту (в соответствии с вариантом). Требования должны делиться на следующие категории:

- Функциональные.
  - Требования пользователей сайта.
  - Требования владельцев сайта.
- Нефункциональные.

Требования необходимо оформить в соответствии с шаблонами RUP (документ SRS - Software Requirements Specification). Для каждого из требований нужно указать его атрибуты (в соответствии с методологией RUP), а также оценить и аргументировать приблизительное количество часов, требующихся на реализацию этого требования.

Для функциональных требований нужно составить UML UseCase-диаграммы, описывающие реализующие их прецеденты использования.

**Отчёт по лабораторной работе должен содержать:**

1. Документ Software Requirements Specification, содержащий список требований к сайту.
2. UseCase-диаграммы прецедентов использования, реализующих функциональные требования.
3. Выводы по работе.

**Вопросы к защите лабораторной работы:**

1. Методологии разработки ПО. Унифицированный процесс.
2. Требования и их категоризация. Атрибуты требований.
3. Язык UML.
4. Прецеденты использования. UseCase-диаграммы - состав, виды связей.

# SOFTWARE REQUIREMENTS SPECIFICATION

## 1. Introduction (Введение)

### 1.1 Purpose (Цель)

Цель этого документа – формулирование функциональных и нефункциональных требований, описание прецедентов, а также выделение актеров, требующихся для разработки системы. Документ можно рассматривать как соглашение между заказчиком и исполнителем о продукте, который должен быть разработан.

### 1.2 Project scope

Документ относится к разрабатываемому веб-сайту <https://www.rutube.ru/> – социальному веб-сайту для потоковой передачи видео. Цель RuTube – предоставлять пользователям видео по их поиску и рекомендации по тематике. Описанная в этом документе система позволит пользователям использовать сайт для просмотра, оценивания и публикации видеороликов.

### 1.3 Definitions, Acronyms and Abbreviations (Определения и аббревиатуры)

1. RuTube – название обозреваемого веб-сайта <https://www.rutube.ru/>.
2. Видеоролик (Видео, Video) – электронная технология записи, обработки, передачи, хранения и воспроизведения подвижного изображения (визуальной и аудиовизуальной информации), представленной в форме цифрового потока видеоданных, хранимая, обрабатываемая и воспроизводимая на RuTube .
3. Плейлист (Playlist) – упорядоченный набор видеороликов, созданный определенным пользователем и обладающий названием.
4. Канал (Channel) – канал на RuTube является домашней страницей для аккаунта пользователя и необходим для загрузки видео, добавления комментариев или составления плейлистов.
5. Подписка (Subscribe) – это функция на RuTube , которая уведомляет субъекта подписки в случае определенных действий объекта подписки (выкладывание нового видеоролика, выход в прямой эфир и т. д.). Как только пользователь подпишется на канал, все новые видео, опубликованные на канале, будут отображаться в его ленте подписок. Подписка дает объекту подписки +1 к количеству подписчиков, помогая ему выглядеть более популярным.
6. RuTube Adds – платный сервис по подписке для RuTube , предоставляющий просмотр видео без рекламы, фоновое воспроизведение на мобильных устройствах или офлайн-воспроизведение.
7. Бизнес-риски – потеря репутации при поломке.
8. Ресурсные риски – организационные, финансовые.
9. Прямые и косвенные риски – можно управлять, контролировать риск или нет.
10. Технические риски – границ проекта, технологические, внешних проектных зависимостей.
11. Политические риски – сферы влияния менеджеров.
12. Ресурсы на выполнение требования – количество денег и времени на выполнение требования.

13. Технические требования – требуемая квалификация людей для выполнения требования.
14. API – Application Programming Interface – Интерфейс программного приложения
15. TCP/IP - сетевая модель передачи данных, представленных в цифровом виде.
16. HTML – Hypertext Markup language – стандартизованный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере
17. CSS – Cascading Style Sheets – формальный язык описания внешнего вида документа, написанного с использованием языка разметки.
18. JavaScript – язык для программного доступа к объектам приложений.
19. Spfjs – легкий JS-фреймворк для быстрой навигации и обновления страниц.
20. HTTP – HyperText Transfer Protocol – сетевой протокол прикладного уровня.
21. HTTPS – Hypertext Transfer Protocol Secure – расширение протокола HTTP для поддержки шифрования в целях повышения безопасности.
22. Linux – семейство Unix-подобных операционных систем на базе ядра Linux.
23. Apache – Apache HTTP Server – свободный кроссплатформенный веб-сервер.
24. MySQL – свободная реляционная система управления базами данных.
25. Vitess – система кластеризации баз данных для горизонтального масштабирования MySQL.
26. Python – высокоуровневый язык программирования общего назначения.
27. Spitfire – высокопроизводительный язык шаблонов Python, разработанный RuTube.
28. Cobalt – высокопроизводительная и компактная платформа, реализующая подмножество HTML5/CSS/JS для запуска приложений, включая приложение RuTube TV.
29. Rcat – Randomized Counter-Abuse Tokens Explainer – средство обнаружения злоупотреблений вовлечением встроенного стороннего веб-контента без использования файлов cookie.
30. Google Analytics – сервис, предоставляемый Google для создания детальной статистики посетителей веб-сайтов.
31. Google Modular Data center – дата-центр Google исполняет функции обработки, хранения и распространения информации.
32. CDN – Content Delivery Network – географически распределённая сетевая инфраструктура, способствующая увеличению скорости загрузки интернет-пользователями аудио-, видео-, программного и других видов цифрового содержимого в точках присутствия сети CDN.
33. GGC – Google Global Cache – программно-аппаратный комплекс компании Google, предназначенный для оптимизации нагрузки на мощности Интернет-провайдера при работе с сервисами Google (один из элементов Google CDN).
34. RPS – Requests Per Second – количество HTTP запросов к веб-сайту за одну секунду.
35. UML – Unified Modeling Language – Язык графического описания для объектного моделирования в области разработки программного обеспечения, для моделирования бизнес-процессов, системного проектирования и отображения организационных структур.
36. Use Case Diagram – диаграмма вариантов использования. Диаграмма, отражающая отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

37. Frontend – презентационная часть информационной или программной системы, её пользовательский интерфейс и связанные с ним компоненты.

38. Backend – это внутренняя часть продукта, которая находится на сервере и скрыта от пользователей.

#### 1.4 Intended Audience and Reading Suggestions (Целевая аудитория)

Данный документ предназначен для:

- Руководителя проекта;
- Разработчиков;
- Тестировщиков;
- Разработчиков BI-решений;
- Сотрудников отдела маркетинга;
- Бизнес-пользователей;
- Технических писателей;

#### 1.5 References (Ссылки)

- Разрабатываемый веб-сайт – <https://rutube.ru/>

#### 1.6 Overview (Обзор документа)

Следующий раздел, Overall Description, представит общее описание разрабатываемого продукта. В частности, опишет предполагаемый функционал продукта, различные ограничения и зависимости на высоком уровне.

Третий раздел, Specific Requirements, содержит конкретное описание различных функций, требований к удобству, безопасности и надежности, необходимых при разработке системы. Также в нем указаны технологии, системы разработки и прочие инструменты, предназначенные непосредственно для разработчиков.

## 2. Overall Description (Общее описание)

### 2.1 Product features (Функционал продукта)

Разрабатываемая система должна предоставлять пользователям возможность просматривать и искать видеоролики и комментарии на них, делиться видеороликами в социальных сетях и на других площадках, просматривать популярные видеоролики. Для зарегистрированных пользователей должны быть доступно комментирование видеороликов, оценивание и публикация видеороликов.

### 2.2 User characteristics (Описание пользователей)

Система должна поддерживать три класса пользователей:

- Неавторизованный пользователь. Имеет доступ к просмотру видеороликов на ресурсе, а также к комментариям, оценкам и просмотрам на них, может

просматривать каналы, делиться видеороликами в социальных сетях и на других площадках, управлять настройками просмотра (изменение громкости, приостановка, перематывание, включение субтитров, изменение качества и скорости воспроизведения, включение мини-проигрывателя), осуществлять поиск по названию ролика, названию канала или по жанру. Кроме того, регистрироваться на сайте, читать информацию о сайте и просматривать ленту рекомендованных и популярных видеороликов, а также подавать жалобы на действия пользователей, нарушающие правила сайта.

- Авторизованный пользователь – пользователь, прошедший регистрацию. Имеет все полномочия неавторизованного пользователя, а также имеет возможность публиковать свои видео, оставлять комментарии на видео каналы других авторов, подписываться на каналы, оценивать видеоролики и комментарии (ставить лайки/дизлайки), создавать плейлисты, загружать собственные видео, участвовать в обсуждениях в сообществах каналов.
- Модератор. Имеет права на удаление комментариев (оскорбительные, спамовые и т. д.), блокировку каналов, мониторинг активности. Также администратор может просматривать, отвечать на обращения пользователей и сотрудничать с создателями контента, проверять потенциально недопустимые сообщения, включать и выключать чат на трансляции, останавливать трансляцию, применять к видео возрастные ограничения.

## 2.3 Operating environment (Операционная среда)

RuTube может работать в любой системе, имеющей современный браузер и рекомендуемое подключение к Интернету (более 500 Кбит/с). Например – Windows, Mac, Linux, Android, IOS. У пользователя должно быть устройство, которое, по крайней мере, способно воспроизводить видео с разрешением 144p.

## 2.4 Design and implementation constraints

Требования к реализации веб-сайта:

- Frontend: реализована при помощи HTML, CSS, JavaScript, для навигации и обновления страниц используется `spfjs`, для сбора данных о пользователе – Google Analytics, использует Cobalt как среду для видеоплеера.
- Backend: основные сервисы реализованы на Python, данные хранятся в MySQL, предоставляет некоторое открытое API для сторонних разработчиков,
- Другое: сервера, на которых работает RuTube, должны работать под управлением Linux, использовать Apache в качестве веб-сервера, находиться в нескольких географически различных Data center и более высокой скорости передачи контента и меньшему времени доступа.

## 2.5 Assumptions and dependencies (Влияющие факторы и зависимости)

2.5.1 Обмен данными между клиентом и сервером происходит посредством протокола HTTPS.

2.5.2 Система предполагает, что пользователи имеют доступ к современному браузеру с интернет-соединением и могут создавать и управлять учетными записями RuTube .

2.5.3 Сайт предполагает, что пользователи прочитали и согласились с условиями использования, политикой конфиденциальности и политикой возврата до совершения покупки.

2.5.4 Сайт предполагает, что пользователи имеют базовые знания работы в браузере и умеет пользоваться компьютером или смартфоном.

## 2.6 Constrains (Ограничения)

2.6.1 Веб-приложение должно отображаться корректно во всех современных браузерах. Пользователь любого современного браузера должен иметь полный доступ ко всем возможностям сайта в соответствии с его ролью.

2.6.2 Сайт должен соответствовать всем соответствующим законам и правилам, связанным с онлайн-торговлей, защитой данных пользователей и конфиденциальностью.

2.6.3 Сайт должен учитывать лицензии сторонних сервисов, используемых им, и подчиняться этим лицензиям.

2.6.4 Сайт должен быть разработан с учетом возможности дальнейшего масштабирования.

2.6.5 Ограничения на доступность функциональности сайта в определенных регионах, ограничений на приобретения товаров и услуг.

2.6.6 Ограничения на доступность функциональности сайта для пользователей, не зарегистрированных на сайте.

2.6.7 Невозможность авторизации без наличия браузера, поддерживающего JavaScript.

2.6.8 Приложение должно безопасно хранить конфиденциальные данные пользователей и запрещать несанкционированный доступ к данным.

2.6.9 Приложение должно регулярно совершать резервное копирование во избежание потери данных пользователей.

## 3. System features (Особенности системы)

### 3.1 Functional requirements (Функциональные требования)

Требования пользователя сайта:

#### **Видеоплеер и просмотр:**

3.1.1 Система должна предоставлять возможность просмотра плейлистов, видеороликов и каналов их авторов.



3.1.2 Система должна предоставлять возможность просмотра каналов и видеороликов по тегированным спискам (например, Все, Видеоигры, Музыка, Джеммы, Сейчас в эфире, Недавно опубликованные, Новые для вас и так далее) в зависимости от предпочтений в темах и жанрах отдельного пользователя.

3.1.4 Система должна предоставлять возможность просмотра комментариев видеороликов, их оценок и ответов.

3.1.9 Система должна предоставлять возможность изменять громкость видеоролика.

3.1.10 Система должна предоставлять возможность поставить видеоролик на паузу, быстро промотать на 10 секунд вперед или назад, или перейти к любому моменту видеоролика.

3.1.11 Система должна предоставлять возможность перейти к следующему видеоролику в плейлисте или в списке рекомендованных со страницы плеера.

3.1.12 Система должна предоставлять возможность включения и выключения субтитров.

3.1.13 Система должна предоставлять возможность включения мини-проигрывателя.

3.1.14 Система должна предоставлять возможность открытия видеоролика на весь экран.

3.1.15 Система должна предоставлять возможность изменять скорость воспроизведения видеоролика (в 0.25, 0.5, 0.75, 1.25, 1.5, 2 раза, а также выбору из промежутка 0.25х-2х) и переходить к обычной скорости.

3.1.16 Система должна предоставлять возможность читать описание к видеороликам.

3.1.35 Система должна предоставлять возможность изменять качество видеоролика во время просмотра в зависимости от качества, в котором оно было выложено каналом.

3.1.36 Система должна предоставлять возможность автонастройки качества видеоролика на основе скорости интернет-соединения пользователя.

3.1.37 Система должна предоставлять возможность рекомендовать пользователям видеороликов персонализированный контент на основе их предпочтений и поведения на платформе.

### **Плейлисты:**

3.1.17 Система должна предоставлять возможность создавать плейлисты из видеороликов.

3.1.31 Система должна предоставлять возможность добавлять видеоролики в существующие плейлисты пользователя.

### **Подписки:**

3.1.20 Система должна предоставлять возможность подписаться на канал со страниц канала и страницы видеоплеера для текущего канала для получения уведомлений о новых видеороликах, пользователи должны получать уведомления о новых видеороликах, комментариях, подписках и других событиях на платформе через электронную почту или мобильные устройства.

3.1.21 Система должна предоставлять возможность отписаться от канала со страницы канала или страницы видеоплеера для текущего канала.

### **Комментирование:**

3.1.5 Система должна предоставлять возможность комментирования видеороликов и сообщений в сообществах канала.

3.1.6 Система должна предоставлять возможность отвечать на комментарии к видеоролику.

3.1.23 Система должна предоставлять возможность написанию комментариев к другим видеороликам.

#### **Оценивание:**

3.1.3 Система должна предоставлять возможность просмотра оценок (лайков и дизлайков) и количества просмотров видеороликов.

3.1.7 Система должна предоставлять возможность оценивать видеоролики.

3.1.8 Система должна предоставлять возможность оценивать комментарии.

#### **Навигация:**

3.1.19 Система должна предоставлять возможность быстро перейти к каналу пользователя, выложившего видеоролик и посмотреть количество его подписчиков.

3.1.22 Система должна предоставлять возможность поиска видеороликов и каналов по названию и теме, а также быстрому выбору из списка поиска пользователя по времени.

3.1.29 Система должна предоставлять возможность просмотра истории поиска пользователя.

3.1.30 Система должна предоставлять возможность пользователю увидеть его историю просмотров.

#### **Уведомления:**

3.1.32 Система должна предоставлять возможность уведомлять пользователя о новых видео из подписок.

3.1.33 Система должна предоставлять возможность пользователю изменять настройки уведомлений.

#### **Каналы:**

3.1.24 Система должна предоставлять возможность подписки на сервис RuTube, используя платежные системы для приема оплаты.

3.1.25 Система должна предоставлять возможность пользователю создать аккаунт на сайте и возможность авторизации под ним.

3.1.28 Система должна предоставлять возможность выкладывать собственные видеоролики в свой канал.

3.1.38 Система должна предоставлять возможность управления профилем, пользователи должны иметь возможность редактировать свой профиль, изменять настройки конфиденциальности и управлять своими подписками.

- 3.1.39 Система должна предоставлять возможность монетизации контента, создатели контента должны иметь возможность монетизировать свои видеоролики через рекламу, спонсорские контракты или другие методы.
- 3.1.34 Система должна предоставлять возможность автору канала удалять выложенные им видео и комментарии.
- 3.1.42 Система должна предоставлять возможность автоматической защиты авторских прав, RuTube должен обеспечивать защиту авторских прав и предоставлять инструменты для обнаружения и удаления незаконного контента, нарушающего авторские права.
- 3.1.45 Система должна предоставлять возможность поддержки партнерских программ, RuTube должен предлагать партнерские программы для создателей контента, которые позволяют им получать дополнительные преимущества, поддержку и инструменты для развития своего канала.

#### **Другое:**

- 3.1.18 Система должна предоставлять возможность поделиться видеороликом в социальных сетях и на других площадках (ВКонтакте, Facebook, Reddit, X, WhatsApp, ОК, электронная почта, goo, Pinterest, Blogger, Tumblr, LinkedIn, Mix) с использованием временной метки.
- 3.1.26 Система должна предоставлять возможность общения со службой поддержки. Пользователи должны иметь возможность обращаться в службу поддержки RuTube для решения проблем, задавать вопросы и предлагать свои идеи по улучшению платформы.
- 3.1.27 Система должна предоставлять возможность просмотра информации о сайте.
- 3.1.40 Система должна предоставлять возможность аналитики, пользователи и создатели контента должны иметь доступ к инструментам аналитики, позволяющим отслеживать статистику просмотров, лайков, комментариев и других метрик для оптимизации своего контента.
- 3.1.44 Система должна предоставлять возможность мультиязычности, RuTube должен предоставлять возможность просмотра и загрузки контента на различных языках, а также поддерживать субтитры и переводы для улучшения доступности для пользователей из разных стран.
- 3.1.46 Система должна предоставлять возможность поддержки различных устройств, RuTube должен быть доступен на различных устройствах, включая компьютеры, мобильные телефоны, смарт-телевизоры, игровые консоли и другие платформы, чтобы обеспечить максимальную доступность для пользователей.

#### **Требования владельца сайта:**

- 3.1.47 Система должна предоставлять возможность доступа к добавлению и удалению видеороликов.
- 3.1.48 Система должна предоставлять возможность доступа к изменению текущих рекомендованных видео, добавлению и удалению комментариев к видеороликам, ограничению доступности видеороликов по возрастному ограничению.
- 3.1.49 Система должна предоставлять возможность блокировать пользователей.
- 3.1.50 Система должна предоставлять возможность учета посещаемости сайта.

3.1.51 Система должна предоставлять возможность добавления и изменения справочной и контактной информации.

3.1.52 Система должна предоставлять возможность отвечать на обращения пользователей.

## 4. Nonfunctional requirements (Нефункциональные требования)

### 4.1 Usability requirements (Требования к удобству использования)

4.1.1. Система должна иметь адаптивный дизайн, который позволяет корректно отображаться на различных устройствах и экранах, включая мобильные устройства. Поддержка всех разрешений экрана пользователя (для телефонов: 360x640, 375x667, 414x896, 360x780, 360x760, 375x812, 360x720, 414x736, 412x846, 412x892, для компьютеров: 1366x768, 1920x1080, 1536x864, 1440x900, 1280x720, 1600x900, 1280x800, 1280x1024, 1024x768, 768x1024).

4.1.2 Система должна отображать сайт с полностью работающим функционалом и без нарушения дизайна в современных популярных браузерах: Chrome 79+, Safari 11+, Mozilla 70+, Яндекс Браузер 21+.

4.1.3 В системе должны быть реализована удобная навигация, которая позволяет пользователям быстро (менее 20 секунд на визуальный поиск нужного элемента интерфейса) и легко (основные правила UX выполняются) находить нужную информацию.

4.1.4. Система должна иметь понятный и простой интерфейс: время поиска нужного видеоролика для опытного пользователя (уже использовавшего сайт) не должно превышать минуту, для неопытного 5 минут, глубина дерева переходов по сайту не должна превышать 10.

4.1.5 В системе должны быть документации и обучающие материалы.

4.1.6 Главные разделы сайта должны быть вынесены в шапку профиля.

4.1.7 Доступность сайта для людей с ограниченными возможностями, в соответствии с рекомендациями WAI (Web Accessibility Initiative) и WCAG (Web Content Accessibility Guidelines). Это включает в себя разработку с учетом навигации с помощью клавиатуры, альтернативных текстов для изображений, достаточного контраста текста и фона и других аспектов, делающих сайт доступным для как можно более широкой аудитории.

4.1.8 Требования к поддержке нескольких языков могут быть критически важны для сайтов, ориентированных на международную аудиторию. Описание того, как система будет поддерживать локализацию и интернационализацию, включая управление переводами контента и интерфейса, может значительно повысить доступность и удобство сайта для пользователей из разных стран.

4.1.9 Важно упомянуть требования к интеграции с внешними сервисами и API, такими как социальные сети, системы оплаты, аналитические инструменты и другие внешние сервисы, необходимо использовать соответствующие технологии, такие как RESTful API,

JSON и другие. API должен поддерживать стандартные методы HTTP, такие как GET, POST, PUT и DELETE, а также обеспечивать возможность передачи параметров и данных в формате JSON или XML.

#### 4.2 Reliability requirements (Требования к надежности)

4.2.1 Доступность система должна составлять не менее 99.999% (время простоя – 5 минут 16 секунд в год) из-за огромной пользовательской базы. Кроме того, желательно полная доступность системы, критические ошибки не должны сильно влиять на качество использования продукта – допустима только временное ограничение некоторого функционала.

4.2.2 Среднее время на устранение проблем: 10 минут. В случае возникновения критического сбоя в системе необходимо возобновить его работу в течение 1 часа.

4.2.3 Отказоустойчивость: создать систему, способную продолжать работу даже при отказе отдельных компонентов.

#### 4.3 Performance requirements (Требования к производительности)

##### 4.3.1 Assessment of required resources (Оценка требуемых ресурсов)

Предполагаемые вводные данные:

- Количество пользователей – 1 000 000.
- Количество активных пользователей в день – 500 000.
- Количество авторов, которые публикуют видео – 100 000.
- Среднее количество публикаций – 2 видео в день.
- Каждый пользователь в среднем просматривает 20 видео в день.
- Максимальный размер видео 256 Гб.
- Средний размер видео – 1 Гб.

Хранилище

- 100 000 авторов публикуют по 2 видео в день, что составляет 200 000 видео ежедневно.
- Если максимальный размер видео составляет 256 Гб, предположим, что средний размер видео составляет 1 Гб (это зависит от форматов и качества видео).
- Таким образом, каждый день будет загружаться 200 000 Гб или 20 000 Тб видео.

**Нагрузка (RPS – requests per second)**

Загрузка/просмотр видео:

- Активных пользователей: 500 000.
- Предположим, что пользователи просматривают 20 видео в день. Это означает 10 000 000 просмотров в день.
- Также учитываем загрузку видео, которая составляет 200 000 в день.
- Общее количество запросов в день: 10 200 000.

Лайки/дизлайки:

- Предположим, что пользователи ставят оценку (лайк или дизлайк) для 5 из 20 просмотренных видео в день.
- Это составляет  $5 * 500\,000 = 2\,500\,000$  лайков/дизлайков в день.

Комментарии:

- Предположим, что пользователи оставляют комментарии для 2 из 20 просмотренных видео в день.
- Это составляет  $2 * 500\,000 = 1\,000\,000$  комментариев в день.

Подписки и отписки

- Предположим, что пользователи в среднем подписываются и отписываются от 3 каналов в день.
- Это составляет  $3 * 500\,000 = 1\,500\,000$  подписок/отписок в день.

Резюмируем расчеты:

Общее количество запросов (просмотры + загрузки + лайки/дизлайки + комментарии + подписки/отписки) в день:  $13,700,000 + 1,500,000 = 15,200,000$

Чтобы получить RPS, нужно поделить общее количество запросов на количество секунд в день (86 400 секунд):  $RPS = 15,200,000 / 86,400 = 175,926$

Таким образом, примерная оценка RPS с учетом всех активностей, включая подписки и отписки, составляет около 175 926 запросов в секунду.

Веб-сайт гарантировать высокую доступность системы, чтобы избежать простоев и обеспечить непрерывное обслуживание пользователей. В 95% случаев запросов максимальное время ответа от сервера не должно превышать 1с.

#### 4.3.2 Bandwidth (Пропускная способность)

- Просмотры: 10,000,000 видео в день, средний размер видео 1 ГБ: 10,000,000 ГБ данных для передачи.
- Загрузки: 200,000 видео в день, средний размер видео 1 ГБ: 200 000 ГБ данных для передачи.
- Общий объем данных для передачи в день: 10 200 000 ГБ
- Чтобы получить пропускную способность в ГБ/сек, нужно поделить общий объем данных на количество секунд в день (86 400 секунд):  $Bandwidth = 10\,200\,000\text{ ГБ} / 86,400\text{ сек} = 118\,056\text{ ГБ/сек}$

Таким образом, примерная оценка пропускной способности составляет около 118 056 ГБ/сек или  $118,056 * 8 = 944\,448$  Гбит/сек (гигабит в секунду).

#### 4.3.3 Bottlenecks (Узкие места)

При проектировании системы, такой как RuTube, потенциальные узкие места (bottlenecks) могут возникнуть из-за множества факторов. Вот некоторые из них:

#### **4.3.3.1 Пропускная способность и масштабирование**

Большое количество пользовательских запросов и передача видео между серверами и пользователями может привести к проблемам с пропускной способностью. Оптимизация и горизонтальное масштабирование сетевой инфраструктуры является важным аспектом в решении этой проблемы для обслуживания миллионов пользователей и хранения огромного количества видео.

#### **4.3.3.2 Вычислительная мощность серверов**

Обработка запросов, транскодирование видео и монетизация требуют значительных вычислительных ресурсов. Важно предусмотреть автомасштабирование серверов для обработки пиковой нагрузки и предотвращения задержек или сбоев.

#### **4.3.3.3 Базы данных и кеширование**

Множество запросов на чтение и запись в базу данных могут привести к узким местам. Оптимизация производительности базы данных, использование шардирования и кеширования может снизить нагрузку на базу данных и улучшить производительность системы.

#### **4.3.3.4 Сеть доставки контента (CDN)**

Без использования CDN, большое количество запросов на видео может перегрузить основные сервера. Использование CDN улучшает производительность, снижает задержки и обеспечивает быстрое воспроизведение видео.

#### **4.3.3.5 Микросервисная архитектура**

Если система не разделена на микросервисы, узкие места могут возникнуть из-за монолитной архитектуры. Разделение системы на микросервисы позволяет лучше масштабировать, изолировать проблемы и оптимизировать каждый компонент независимо.

#### **4.3.3.6 Отказоустойчивость и отказобезопасность**

Возникновение сбоев в одной части системы может вызвать узкие места или неработоспособность системы в целом. Разработка стратегий резервного копирования, репликации и автоматического переключения на резервные узлы обеспечивает непрерывную работу системы в случае сбоев.

#### **4.3.3.7 Балансировка нагрузки**

Неравномерное распределение нагрузки между серверами может привести к перегрузке отдельных узлов и снижению производительности системы. Балансировщики нагрузки помогают равномерно распределять трафик между серверами и обеспечивать эффективное использование ресурсов.

#### **4.3.3.8 Оптимизация алгоритмов**

Неэффективные или медленные алгоритмы могут стать узким местом в системе, замедлять обработку запросов и воспроизведение видео. Непрерывная оптимизация алгоритмов и использование асинхронной обработки данных может снизить нагрузку на систему и улучшить производительность.

#### **4.3.3.9 Мониторинг и анализ производительности**

Непрерывный мониторинг и анализ производительности системы позволяют оперативно выявлять и устранять узкие места. Использование средств мониторинга, логирования и анализа данных помогает идентифицировать проблемные области и оптимизировать систему.

#### 4.3.3.10 Безопасность

Атаки на систему, такие как DDoS-атаки или попытки взлома, могут стать узким местом и снизить производительность или доступность системы. Обеспечение безопасности системы, использование методов защиты от DDoS-атак и мониторинг нарушений безопасности являются важными элементами проектирования системы.

Учитывая все эти потенциальные узкие места и применяя соответствующие решения и стратегии, можно создать надежную и высокопроизводительную систему, способную справляться с большим количеством пользователей и их запросов.

#### 4.4 Licensing requirements (Лицензионные требования)

4.4.1 Система должна использовать закрытую лицензию, то есть не разрешается использование исходного кода программы.

#### 4.5 Safety requirements (Требования безопасности)

4.5.1 Если значительная часть базы данных повреждена из-за катастрофического сбоя, такого как сбой сервера, метод восстановления восстанавливает предыдущую копию базы данных, резервная копия которой была сохранена в архивном хранилище (обычно на магнитной ленте), и восстанавливает текущее состояние, повторно применяя или повторяя операции зафиксированных транзакций из резервной копии журнала, вплоть до момента сбоя, тем временем сервер переключается на резервные серверы, чтобы веб-сайт продолжал работать.

#### 4.6 Security requirements

4.6.1 Безопасное проведение оплаты каждым из способов, безопасная передача ключей доступа, адресов электронной почты и номеров телефонов пользователей (как минимум: использование HTTPS).

4.6.2 Безопасность: защитить данные пользователей и системы от несанкционированного доступа, а также предотвратить атаки и утечки данных.

4.6.3 Пользователь также может сообщить, если обнаружит какую-либо подозрительную активность на RuTube, чтобы отдел безопасности мог позаботиться об этом.

### 5. External interface requirements (Требования к внешнему интерфейсу)

#### 5.1 User Interfaces (Пользовательские интерфейсы)

Пользовательский интерфейс предоставляется в виде HTML страницы со стилями CSS и языком клиентских сценариев JavaScript. Рендер полученного гипертекста осуществляется браузером.



## 5.2 Software Interfaces (Программные интерфейсы)

Для обеспечения работы программных интерфейсов на сайте необходимо использовать соответствующие технологии, такие как RESTful API, JSON и другие. API должен поддерживать стандартные методы HTTP, такие как GET, POST, PUT и DELETE, а также обеспечивать возможность передачи параметров и данных в формате JSON или XML.

## 5.3 Communications Interfaces (Коммуникационные интерфейсы)

### 5.3.1 Сетевой интерфейс TCP/IP.

### 5.3.2 HTTPS protocol.

## 6. Requirements Attributes (Атрибуты требований)

### Функциональные требования

№ требования	Приоритетность	Трудоемкость человеко-час	Стабильность
3.1.1	10	500	Низкая
3.1.2	8	200	Средняя
3.1.3	9	20	Высокая
3.1.4	10	30	Высокая
3.1.5	10	100	Средняя
3.1.6	8	20	Средняя
3.1.7	9	20	Высокая
3.1.8	9	10	Высокая
3.1.9	10	10	Высокая
3.1.10	10	20	Средняя
3.1.11	8	5	Высокая
3.1.12	5	50	Низкая
3.1.13	4	5	Средняя
3.1.14	7	5	Высокая
3.1.15	7	20	Средняя
3.1.16	10	5	Высокая
3.1.17	8	60	Высокая
3.1.18	6	10	Средняя
3.1.19	4	5	Высокая
3.1.20	8	20	Средняя
3.1.21	8	5	Высокая
3.1.22	9	200	Низкая

3.1.23	10	40	Средняя
3.1.24	6	500	Низкая
3.1.25	10	100	Низкая
3.1.26	8	20	Средняя
3.1.27	3	10	Высокая
3.1.28	10	200	Низкая
3.1.29	4	50	Средняя
3.1.30	6	200	Низкая
3.1.31	7	20	Высокая
3.1.32	6	200	Средняя
3.1.33	3	60	Низкая
3.1.34	9	30	Высокая
3.1.35	8	300	Низкая
3.1.36	4	80	Средняя
3.1.37	8	800	Низкая
3.1.38	5	200	Средняя
3.1.39	2	800	Низкая
3.1.40	5	600	Средняя
3.1.41	2	800	Низкая
3.1.42	7	400	Низкая
3.1.43	1	800	Низкая
3.1.44	8	200	Средняя
3.1.45	5	500	Средняя
3.1.46	9	400	Средняя
3.1.47	10	80	Высокая
3.1.48	10	250	Низкая
3.1.49	9	50	Высокая
3.1.50	8	300	Средняя
3.1.51	10	20	Высокая
3.1.52	8	50	Высокая

### Нефункциональные требования

Требование	Приоритетность	Стабильность
4.1.1	9	Средняя

4.1.2	10	Средняя
4.1.3	8	Высокая
4.1.4	8	Средняя
4.1.5	5	Высокая
4.1.6	9	Высокая
4.1.7	4	Низкая
4.1.8	10	Средняя
4.1.9	5	Низкая
4.2.1	10	Низкая
4.2.2	9	Средняя
4.2.3	8	Низкая
4.3.3.1	6	Низкая
4.3.3.2	7	Средняя
4.3.3.3	7	Средняя
4.3.3.4	9	Низкая
4.3.3.5	3	Низкая
4.3.3.6	8	Низкая
4.3.3.7	9	Средняя
4.3.3.8	9	Низкая
4.3.3.9	7	Высокая
4.3.3.10	8	Средняя
4.4.1	10	Высокая
4.5.1	9	Средняя
4.6.1	10	Высокая
4.6.2	9	Высокая
4.6.3	8	Средняя

### Риски

№	Риск	Тип	Вероятность	Масштаб потерь
1	Критический сбой в работоспособности системы	Технический	Низкая	Значительные
2	Критический сбой при доступе к базе данных	Технический	Низкая	Значительные
3	Малое количество посещений сайта из-за	Бизнес-риск	Низкая	Средние

	неудобного и устаревшего интерфейса и дизайна сайта			
4	Задержка зарплат разработчикам	Ресурсный	Очень низкая	Средние
6	Изменение законодательной базы, связанной с деятельностью системы.	Политический	Средняя	Значительные
7	Проигрыш в конкуренции	Бизнес-риск	Очень низкая	Средние
8	Нехватка денег из-за длительности разработки	Ресурсный	Очень низкая	Значительные
9	Недостаточная компетентность разработчиков	Технический	Низкая	Низкие
10	Нехватка людей на разработку/поддержку проекта	Ресурсный	Очень низкая	Средние

## 7. Precedents (Прецеденты)

Прецедент: Просмотр видео
ID: 1
Краткое описание: Пользователь просматривает рекомендованный контент на RuTube
Главный актер: Незарегистрированный пользователь, Неавторизованный пользователь, Авторизованный пользователь
Предусловия: Пользователь находится на главной странице RuTube
Основной поток: <ol style="list-style-type: none"><li>1. Пользователь выбирает видео, нажимая на его превью или заголовок.</li><li>2. Система открывает страницу видео.</li><li>3. Видео автоматически начинает воспроизводиться.</li><li>4. Пользователь может управлять воспроизведением (пауза, перемотка, изменение громкости).</li></ol>
Альтернативный поток: Если интернет-соединение прерывается, воспроизведение приостанавливается, и пользователю предлагается проверить соединение. Если по техническим причинам видео не может быть загружено (например, из-за ошибки на сервере), пользователю отображается сообщение об ошибке с предложением попробовать посмотреть видео позже.
Постусловия: Пользователь успешно просмотрел видео или столкнулся с технической проблемой, предотвращающей просмотр.

Прецедент: Авторизация пользователя
ID: 2
Краткое описание: Пользователь авторизуется на платформе для доступа к дополнительным функциям
Главный актер: Неавторизованный пользователь
Предусловия: Пользователь находится на главной странице RuTube или на странице с видео
Основной поток: <ol style="list-style-type: none"><li>1. Пользователь нажимает на кнопку входа в систему.</li><li>2. Система предлагает ввести электронную почту и пароль от RuTube аккаунта.</li><li>3. Пользователь вводит данные для входа.</li><li>4. Система авторизует пользователя и предоставляет доступ к дополнительным функциям (например, подписке на каналы, комментированию видео).</li></ol>
Альтернативный поток: Если данные введены некорректно, система информирует пользователя и предлагает ввести данные заново. Если пользователь не помнит свой пароль, он может воспользоваться функцией восстановления пароля, следуя инструкциям системы.
Постусловия: Пользователь успешно авторизован на платформе или начал процесс восстановления пароля.

Прецедент: Модерация контента
ID: 3
Краткое описание: Модератор проверяет видео на соответствие правилам платформы
Главный актер: Модератор
Предусловия: Видео загружено на платформу.
<p>Основной поток:</p> <ol style="list-style-type: none"> <li>1. Модератор получает уведомление о необходимости проверки нового видео.</li> <li>2. Модератор просматривает видео и описание к нему.</li> <li>3. Если видео соответствует правилам платформы, модератор одобряет его публикацию.</li> <li>4. Если видео нарушает правила, модератор блокирует видео и отправляет уведомление автору с указанием причины блокировки.</li> </ol>
<p>Альтернативный поток: Если модератор сомневается в решении, он может запросить дополнительную проверку у других модераторов.</p> <p>Если модератору требуются дополнительные разъяснения по контенту, он может связаться с автором видео перед принятием решения о его блокировке.</p>
Постусловия: Видео оставлено без изменений, удалено или заблокировано с уведомлением автору о причинах такого решения.

Прецедент: Комментирование видео
ID: 4
Краткое описание: Пользователь оставляет комментарий под видео.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован и просматривает видео.
<p>Основной поток:</p> <ol style="list-style-type: none"> <li>1. Пользователь прокручивает страницу до раздела комментариев.</li> <li>2. В поле для ввода пользователь набирает свой комментарий.</li> <li>3. Пользователь нажимает кнопку "Оставить комментарий".</li> <li>4. Система публикует комментарий под видео.</li> </ol>
<p>Альтернативный поток: Если комментарий содержит запрещенную информацию, система отклоняет его публикацию и уведомляет пользователя.</p> <p>Если у пользователя ограничен доступ к комментированию (например, из-за предыдущих нарушений), он получает уведомление об этом ограничении.</p>
Постусловия: Комментарий пользователя опубликован под видео или пользователь информирован о невозможности оставить комментарий.

Прецедент: Подписка на канал
ID: 5
Краткое описание: Пользователь подписывается на канал для получения уведомлений о новых видео.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован и находится на странице канала или просматривает видео этого канала.
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь нажимает кнопку "Подписаться".</li> <li>2. Система подписывает пользователя на канал.</li> <li>3. Пользователю предлагается включить уведомления о новых видео.</li> <li>4. При появлении новых видео пользователь получает уведомления.</li> </ol>
Альтернативный поток: -
Постусловия: Пользователь подписан на канал и, возможно, активировал уведомления о новых видео.

Прецедент: Загрузка видео
ID: 6
Краткое описание: Авторизованный пользователь загружает видео на свой канал.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован и имеет канал на RuTube .
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь переходит в раздел "Загрузить видео".</li> <li>2. Выбирает видеофайл на своем устройстве.</li> <li>3. Заполняет информацию о видео (заголовок, описание, теги).</li> <li>4. Выбирает настройки конфиденциальности и загружает видео.</li> <li>5. Система обрабатывает и публикует видео на канале пользователя.</li> </ol>
Альтернативный поток: Если загружаемое видео нарушает правила платформы, загрузка отменяется, и пользователь получает уведомление. Если размер загружаемого файла превышает установленный лимит, пользователю предлагается сократить видео или снизить его качество.
Постусловия: Видео опубликовано на канале пользователя или пользователь информирован о необходимости изменения файла для соответствия лимитам и правилам платформы.

Прецедент: Изменение настроек учетной записи
ID: 7
Краткое описание: Пользователь изменяет настройки своей учетной записи на RuTube.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован на платформе.
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь заходит в раздел "Настройки" в своем профиле.</li> <li>2. Пользователь выбирает раздел настроек, который хочет изменить (например, конфиденциальность, уведомления, язык).</li> <li>3. Пользователь вносит желаемые изменения и сохраняет их.</li> </ol>
Альтернативный поток: Если система не может сохранить изменения (например, из-за сетевой ошибки), пользователю предлагается попробовать снова позже. Если изменения настроек не могут быть сохранены из-за ошибки системы, пользователю предлагается повторить попытку позже.
Постусловия: Настройки учетной записи обновлены или пользователь уведомлен о временной невозможности их сохранения.

Прецедент: Управление плейлистами
ID: 8
Краткое описание: Пользователь создает, редактирует и удаляет плейлисты.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован и находится на странице своего профиля или в разделе плейлистов.
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь выбирает опцию создания нового плейлиста.</li> <li>2. Вводит название плейлиста и, при желании, описание.</li> <li>3. Добавляет в плейлист выбранные видео.</li> <li>4. Сохраняет плейлист.</li> </ol>
Альтернативный поток: Пользователь может редактировать или удалять существующие плейлисты, добавляя или удаляя видео, изменяя название и описание. Если во время создания или редактирования плейлиста происходит ошибка, например, из-за проблем с сетью, пользователю предлагается попробовать сохранить плейлист снова.
Постусловия: Плейлист успешно создан, отредактирован или пользователь уведомлен о необходимости повторить попытку сохранения.



Прецедент: Просмотр аналитики канала
ID: 9
Краткое описание: Авторизованный пользователь просматривает статистику и аналитику своего канала.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован и имеет свой канал на RuTube .
<p>Основной поток:</p> <ol style="list-style-type: none"> <li>1. Пользователь переходит в раздел "Аналитика" в настройках своего канала.</li> <li>2. Просматривает различные показатели: количество просмотров, подписчиков, среднее время просмотра, источники трафика и т. д.</li> <li>3. Использует доступные фильтры для уточнения данных (например, период времени, тип контента).</li> </ol>
<p>Альтернативный поток: Если данные аналитики временно недоступны из-за технических работ, пользователю предлагается проверить информацию позже. Если для выбранного периода времени отсутствуют аналитические данные, пользователю отображается соответствующее уведомление с предложением выбрать другой временной интервал.</p>
Постусловия: Пользователь получил доступ к аналитическим данным за доступный период или уведомлен о их отсутствии за выбранный период.

Прецедент: Репортинг неподобающего контента
ID: 10
Краткое описание: Пользователь сообщает о видео или комментарии, нарушающих правила платформы.
Главный актер: Незарегистрированный или неавторизованный пользователь, Авторизованный пользователь
Предусловия: Пользователь находит контент, который, по его мнению, нарушает правила RuTube.
<p>Основной поток:</p> <ol style="list-style-type: none"> <li>1. Пользователь нажимает на кнопку "Пожаловаться" рядом с видео или комментарием.</li> <li>2. Выбирает причину жалобы из предложенного списка.</li> <li>3. При необходимости, добавляет комментарий к своей жалобе.</li> <li>4. Отправляет жалобу.</li> </ol>
<p>Альтернативный поток: Если пользователь случайно отправил жалобу, он не может отменить своё действие, но может связаться с поддержкой для уточнения.</p>
Постусловия: Жалоба рассмотрена модераторами, а пользователь либо отозвал свою жалобу через поддержку, либо она была обработана согласно стандартному процессу.

Прецедент: Изменение качества воспроизведения видео
ID: 11
Краткое описание: Пользователь изменяет разрешение видео для оптимизации воспроизведения.
Главный актер: Незарегистрированный или неавторизованный пользователь, Авторизованный пользователь
Предусловия: Пользователь просматривает видео.
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь кликает на иконку настроек в плеере RuTube .</li> <li>2. Выбирает пункт "Качество".</li> <li>3. Выбирает предпочитаемое разрешение видео из доступных опций.</li> </ol>
Альтернативный поток: Если интернет-соединение пользователя медленное, RuTube может автоматически уменьшить качество воспроизведения для обеспечения бесперебойного просмотра. Если качество воспроизведения установлено на автоматическое и качество сети изменяется, система автоматически адаптирует качество воспроизведения видео без действий пользователя.
Постусловия: Качество воспроизводимого видео изменено в соответствии с выбором пользователя или автоматически, в зависимости от условий сети.

Прецедент: Просмотр истории просмотров
ID: 12
Краткое описание: Пользователь просматривает свою историю просмотров на RuTube.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован на RuTube .
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь переходит в раздел "История просмотров".</li> <li>2. Просматривает список просмотренных ранее видео.</li> <li>3. При желании, удаляет отдельные видео из истории или очищает всю историю целиком.</li> </ol>
Альтернативный поток: Если история просмотров отключена, пользователь видит соответствующее уведомление и может включить её в настройках. Если попытка очистить историю просмотров не удаётся из-за технической ошибки, пользователю предлагается повторить попытку позже.
Постусловия: История просмотров пользователя очищена или пользователь информирован о временной невозможности выполнить действие.

Прецедент: Управление уведомлениями
ID: 13
Краткое описание: Пользователь настраивает уведомления от каналов и платформы.
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован на RuTube и подписан на один или несколько каналов.
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь переходит в раздел настроек уведомлений.</li> <li>2. Выбирает типы уведомлений, которые хочет получать, и способы их доставки (на email, в мобильном приложении).</li> <li>3. Сохраняет изменения.</li> </ol>
Альтернативный поток: Если пользователь решает отписаться от всех уведомлений, он может отключить их полностью в настройках. Если система не может обработать запрос на подписку или отписку от уведомлений из-за технической ошибки, пользователю предлагается попробовать изменить настройки уведомлений позже.
Постусловия: Настройки уведомлений пользователя обновлены в соответствии с его выбором или пользователь уведомлен о временной проблеме с изменением настроек.

Прецедент: Изменение темы интерфейса
ID: 14
Краткое описание: Пользователь переключает между светлой и тёмной темами интерфейса RuTube .
Главный актер: Авторизованный пользователь
Предусловия: Пользователь авторизован и находится в настройках своего аккаунта.
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь переходит в раздел "Внешний вид" в настройках аккаунта.</li> <li>2. Выбирает "Светлая" или "Тёмная" тема интерфейса из доступных опций.</li> <li>3. Изменения применяются автоматически без перезагрузки страницы.</li> </ol>
Альтернативный поток: -
Постусловия: Тема интерфейса изменена согласно выбору пользователя.

Прецедент: Обновление правил платформы
ID: 15
Краткое описание: Модератор участвует в обновлении и публикации новых правил платформы.
Главный актер: Модератор
Предусловия: Необходимость внесения изменений или дополнений в правила платформы.
Основной поток: <ol style="list-style-type: none"> <li>1. Новые положения вносят в существующие правила.</li> <li>2. Оформляют изменения в официальном документе и публикуют обновлённые правила на платформе.</li> <li>3. Информируют пользователей о нововведениях через уведомления и новостные обновления.</li> </ol>
Альтернативный поток: -
Постусловия: Правила платформы обновлены и доведены до сведения пользователей, обеспечивая их соблюдение.

Прецедент: Поиск видеороликов
ID: 17
Краткое описание: Пользователь ищет видеоролик на RuTube, по ключевым словам
Главный актер: Незарегистрированный пользователь, Неавторизованный пользователь, Авторизованный пользователь
Предусловия: Пользователь находится на главной странице RuTube
Основной поток: <ol style="list-style-type: none"> <li>1. Пользователь нажимает на иконку поиска и переходит в окно поиска.</li> <li>2. Пользователь вводит текст поиска – искомое название видеоролика.</li> <li>3. Система предугадывает возможные продолжения текста поиска.</li> <li>4. Пользователь выбирает из возможных продолжений или оставляет свой текст поиска и нажимает «Enter».</li> <li>5. Система выдает пользователю наиболее релевантные видеоролики.</li> <li>6. Пользователь переходит к просмотру видеороликов.</li> </ol>
Альтернативный поток: Если не найдено точного совпадения по результатам поиска, система выдает наиболее близкие по жанру и теме поиска видеоролики.
Постусловия: Пользователь успешно нашел искомый видеоролик или нашел близкие по теме видеоролики.
Прецедент: Публикация ссылки на видеоролик в социальных сетях
ID: 18
Краткое описание: Пользователь делится видеороликом в социальных сетях.
Главный актер: Незарегистрированный пользователь, Неавторизованный пользователь, Авторизованный пользователь

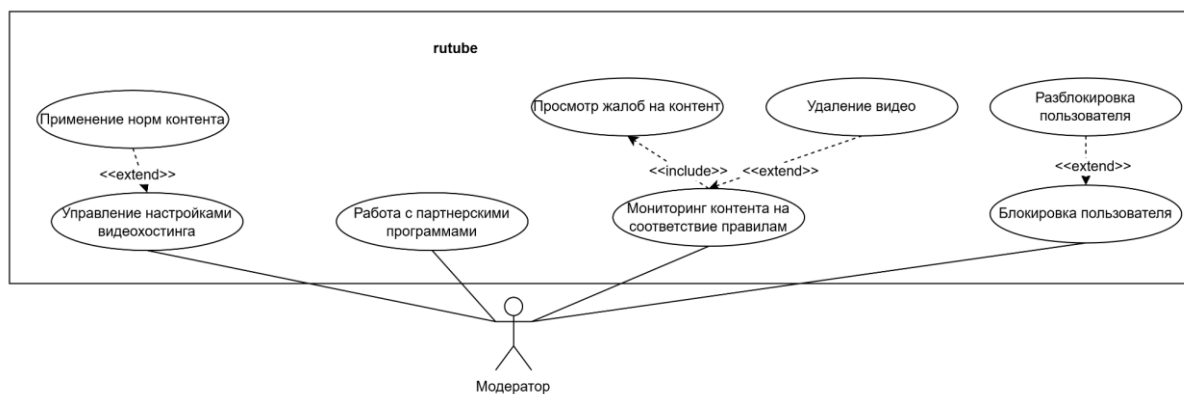
Предусловия: Пользователь находится на странице воспроизведения видеоролика.
<p>Основной поток:</p> <ol style="list-style-type: none"> <li>1. Пользователь нажимает на кнопку «Поделиться».</li> <li>2. Пользователь выбирает социальную сеть, в которой хочет поделиться.</li> <li>3. Опционально, пользователь добавляет временную метку.</li> <li>4. Пользователь нажимает «Отправить» и переходит в социальную сеть для уточнения объекта отправки.</li> <li>5. После отправки пользователь возвращается на страницу воспроизведения видеоролика.</li> </ol>
<p>Альтернативный поток: Если пользователь не нашел нужной социальной сети, от нажимает «Назад» и возвращается к просмотру видеоролика.</p> <p>Если отправка не удалась из-за ошибки сторонней социальной сети, пользователь получает уведомление об этом.</p>
Постусловия: Ссылка на видеоролик опубликована в сторонней социальной сети, либо пользователь получил уведомление о невозможности отправки.

Прецедент: Выход из аккаунта
ID: 21
Краткое описание: Пользователь выходит из аккаунта.
Главный актер: Авторизированный пользователь
Предусловия: Пользователь открыл окно «Мой канал»
<p>Основной поток:</p> <ol style="list-style-type: none"> <li>1. Пользователь находит кнопку «Выйти» и нажимает на нее.</li> <li>2. Пользователь выходит из аккаунта RuTube.</li> </ol>
Альтернативный поток: Если пользователю не удастся выйти из аккаунта по техническим причинам, он получает сообщение о причинах неполадок.
Постусловия: Авторизированный пользователь стал неавторизованным пользователем или получил сообщение об ошибке.

## APPENDIX A: ANALYSIS MODELS

1. Use-Case-диаграммы прецедентов использования, реализующих функциональные требования.





## ВЫВОД

В ходе выполнения работы я узнала про структуру SRS-документа, а также ознакомилась с методологией RUP. Создала список требований, предъявляемых к веб-сайту, для функциональных требований составил UML Use-Case - диаграммы, описывающие реализующие их прецеденты использования. Попрактиковалась в расчете времени и рисков для различных требований.