

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

СОГЛАСОВАНО

руководитель  
МТС.Тета

УТВЕРЖДАЮ

Академический руководитель  
образовательной программы  
«Программная инженерия»

\_\_\_\_\_ С. А. Демиденко  
«\_\_» \_\_\_\_\_ 2022 г.

\_\_\_\_\_ В. В. Шилов  
«\_\_» \_\_\_\_\_ 2022 г.

**Аналитическое хранилище для KION**

**Пояснительная записка**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.02.07-01 81 01-1-ЛУ**

Исполнитель:

студент группы БПИ206

\_\_\_\_\_ / К. А. Веселов /  
«\_\_» \_\_\_\_\_ 2022 г.

П о д п и с п и с а н и с т и т у т	
И н в е с т и ц и е н т и е н т	

<i>В</i>	
<i>з</i>	
<i>а</i>	
<i>м</i>	
<i>.</i>	
<i>и</i>	
<i>н</i>	
<i>в</i>	
<i>.</i>	
<i>№</i>	
<i>П</i>	
<i>о</i>	
<i>д</i>	
<i>н</i>	
<i>.</i>	
<i>и</i>	
<i>д</i>	
<i>а</i>	
<i>т</i>	
<i>а</i>	
<i>И</i>	R
<i>н</i>	U
<i>в</i>	.
<i>.</i>	1
<i>№</i>	7
<i>п</i>	7
<i>о</i>	0
<i>д</i>	1
<i>л</i>	7
	2
	9
	.
	0
	2
	.
	0
	7
	-
	0
	1
	8
	1

Москва 2022

## СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ</b>	<b>4</b>
Наименование программы	4
Документы, на основании которых ведется разработка	4
<b>НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ.</b>	<b>5</b>
Назначение программы	5
Функциональное назначение	5
<b>ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ</b>	<b>6</b>
Постановка задачи на разработку программы	6
Задачи работы	6
Описание архитектуры программы	6
Обоснование выбора архитектуры программы	6
Описание построения серверной части	7
Описание и обоснование выбора метода взаимодействия клиента и сервера	8
Описание и обоснование выбора состава технических и программных средств	8
<b>ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ</b>	<b>9</b>
Ориентировочная экономическая эффективность	9
Предполагаемая потребность	9
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>	<b>10</b>
<b>ПРИЛОЖЕНИЕ 1</b>	
<b>ТЕРМИНОЛОГИЯ</b>	<b>12</b>
<b>ПРИЛОЖЕНИЕ 2 СХЕМА АРХИТЕКТУРЫ</b>	<b>12</b>

## АННОТАЦИЯ

В данном программном документе приведена пояснительная записка к программе «Аналитического хранилища для KION».

В разделе «Введение» указано наименование программы, краткое наименование программы и документы, на основании которых ведется разработка.

В разделе «Назначение и область применения» указано функциональное назначение программы, эксплуатационное назначение программы и краткая характеристика области применения программы.

В разделе «Технические характеристики» содержатся следующие подразделы:

- 1) постановка задачи на разработку программы;
- 2) описание функционирования программы с обоснованием выбора схемы алгоритма решения задачи и возможные взаимодействия программы с другими программами;
- 3) описание и обоснование выбора метода организации входных и выходных данных;
- 4) описание и обоснование выбора состава технических и программных средств.

Настоящий документ разработан в соответствии с требованиями:

- 1) ГОСТ 19.101–77 Виды программ и программных документов [1];
- 2) ГОСТ 19.102–77 Стадии разработки [2];
- 3) ГОСТ 19.103–77 Обозначения программ и программных документов [3];
- 4) ГОСТ 19.104–78 Основные надписи [4];
- 5) ГОСТ 19.105–78 Общие требования к программным документам [5];
- 6) ГОСТ 19.106–78 Требования к программным документам, выполненным печатным способом [6];
- 7) ГОСТ 19.404–79 Пояснительная записка. Требования к содержанию и оформлению [7].

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Изменения к Пояснительной записке оформляются согласно ГОСТ 19.603–78 [8], ГОСТ 19.604–78 [9].

## 1. ВВЕДЕНИЕ

### 1.1. Наименование программы

Наименование: «Аналитическое хранилище для KION».

Наименование на английском языке: «Analytical repository for KION».

### 1.2. Документы, на основании которых ведется разработка

Основанием для разработки является учебный план подготовки бакалавров по направлению 09.03.04 "Программная инженерия" и утвержденная академическим руководителем тема курсового проекта.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ.

### 2.1. Назначение программы

#### 2.1.1. Функциональное назначение

Функциональным назначением программы является хранение и обработка данных о времени просмотра пользователем видео.

Основная цель приложения – предоставлять рекомендательному и, возможно, другим сервисам информацию о времени в просмотре фильма конкретным пользователем, для ввода этих данных в рекомендательный алгоритм.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### 3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

#### 3.1. Постановка задачи на разработку программы

##### 3.1.1. Задачи работы

1. Реализовать взаимодействие с приложением посредством GRPC-запросов;
2. Реализовать отправку данных из приложения в Kafka.
3. Организовать инфраструктуру, необходимую для запуска приложения: запустить Kafka, Clickhouse и прочее;
4. Реализовать балансировку загрузки приложения;
5. Реализовать отказоустойчивость приложения, даже в случаях неполадок на некоторых серверах в соответствии с ТЗ;
6. Протестировать приложение на соответствие заявленной производительности.
7. Реализовать сбор телеметрии о состоянии и загруженности серверов;

#### 3.2. Описание архитектуры программы

Программа представляет из себя:

- Скрипты для развертывания архитектуры: развертывания kubernetes-кластера на наборе машин используя k3sup, развертывания на кластере приложений Kafka, Clickhouse, Redis и приложений мониторинга, развертывания на кластере приложения сервера и настройки его масштабирования
- Серверное приложение на go, обрабатывающее запросы
- Приложение для нагрузочного тестирования

#### 3.3. Обоснование выбора архитектуры программы

В качестве БД был выбран Clickhouse из-за ряда преимуществ: Clickhouse столбцовая БД, что увеличивает его производительность при использовании его в аналитических запросах – то есть запросах, которые агрегируют данные по столбцам, Clickhouse open-source, и для него существует kubernetes-operator который упрощает запуск и работу с ним в kubernetes кластере.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

В качестве очереди сообщений была выбрана Kafka из-за её популярности, большой производительности и наличию kubernetes-operator

### 3.4. Описание построения серверной части

Схема архитектуры приложения будет приложена в приложении 2. Обработывая запросы, сервер обращается в кэш

set –

Действие пользователя когда он переходит на таймкод или смотрит видео до этого момента

В момент получения set запроса сервис:

- Отправляет в redis новую строчку user\_id-video\_id:video\_time
- Отправляет данные в Kafka

При получении get запроса сервис:

- Проверяет, есть ли в Redis нужная строчка
- Если она есть, то сервис её возвращает
- Если строчки нет, то сервис обращается в Clickhouse, и возвращает строчку оттуда

Также в сервисе реализован запрос на watchtime всех видео для конкретного пользователя: на такой запрос сервис всегда ходит в Clickhouse и считает суммарное просмотренное время по видео.

Выбор таких запросов оправдывается составом выполняемых функций и может быть несложно изменен в ходе эксплуатации и новых требований сервиса аналитики.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата



**3.5. Описание и обоснование выбора метода взаимодействия клиента и сервера**

Клиент отправляет запросы на сервер в формате GRPC. Был выбран такой формат, так как он распространен в контексте микросервисной архитектуры, в контексте которой был сделан проект. Также, GRPC бинарный формат, что делает его быстрее JSON

**3.6. Описание и обоснование выбора состава технических и программных средств**

Cloud сервера для kubernetes-кластера:

- SLA сервера 99.9%
- С операционной системой совместимой с kubernetes и docker-контейнером go приложения, например Ubuntu 20.04
- x86-64 процессор
- минимум 2ГБ оперативной памяти, так как при меньшем количестве, Kafka будет работать плохо

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

#### **4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ**

##### **4.1. Ориентировочная экономическая эффективность**

В рамках данной работы расчет экономической эффективности не предусмотрен.

##### **4.2. Предполагаемая потребность**

Данный сервис помогает сбору и хранению данных для аналитических сервисов, которые в свою очередь прямо влияют на пользователей.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 5. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 2) ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 4) ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 5) ГОСТ 19.105-78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 7) ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 8) ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 9) ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 10) Kleppmann, M. Designing Data-Intensive Applications / M. Kleppmann; A. Spencer, M. Beaugureau, K. Brown, R. Head, A. Kersey, E. Troutman-Zaig, D. Futato, K. Montgomery, R. Demarest. - First Edition - 1005 Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly Media, Inc, 2017. - 590 с.
- 11) Redis [Электронный ресурс] / Redis Ltd – Режим доступа: <https://redis.io/>, свободный (дата обращения 17.02.2022)
- 12) Clickhouse [Электронный ресурс] / ClickHouse, Inc. – Режим доступа: <https://clickhouse.com/docs/>, свободный (дата обращения 17.02.2022)
- 13) Apache Kafka [Электронный ресурс] / Apache Software Foundation — Режим доступа: <https://kafka.apache.org/>, свободный (дата обращения 17.02.2022)
- 14) The Go programming language [Электронный ресурс] / Google — Режим доступа: <https://go.dev/doc/>, свободный (дата обращения 17.02.2022)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

- 15) Elasticsearch, Logstash, Kibana [Электронный ресурс] / Elasticsearch B.V. - Режим доступа: <https://www.elastic.co/what-is/elk-stack>, свободный
- 16) Grafana [Электронный ресурс] / Grafana Labs - Режим доступа: <https://grafana.com/grafana/>, свободный (дата обращения 17.02.2022)
- 17) Документация gRPC [Электронный ресурс] / gRPC authors — Режим доступа: <https://grpc.io/docs/>, свободный. (дата обращения 17.02.2022)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**ПРИЛОЖЕНИЕ 1**  
**ТЕРМИНОЛОГИЯ**

1. **HTTP** (англ. *HyperText Transfer Protocol* — «протокол передачи гипертекста») — протокол прикладного уровня передачи данных, изначально — в виде гипертекстовых документов в формате HTML, в настоящее время используется для передачи произвольных данных.
2. **GRPC** (англ. Google Remote Procedure Call) — это система удалённого вызова процедур (RPC) с открытым исходным кодом, первоначально разработанная в Google в 2015 году. В качестве транспорта используется HTTP/2, в качестве языка описания интерфейса — Protocol Buffers.
3. **Таймкод** (англ. Time code) — Момент времени в видео, выраженный в количестве секунд с начала
4. **JSON** (англ. *JavaScript Object Notation*) — Текстовый формат обмена данными.
5. **База данных** (сокр. БД) — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных.
6. **Веб-сервер** — сервер, принимающий HTTP-запросы от клиентов и выдающий им HTTP-ответы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 2 СХЕМА АРХИТЕКТУРЫ

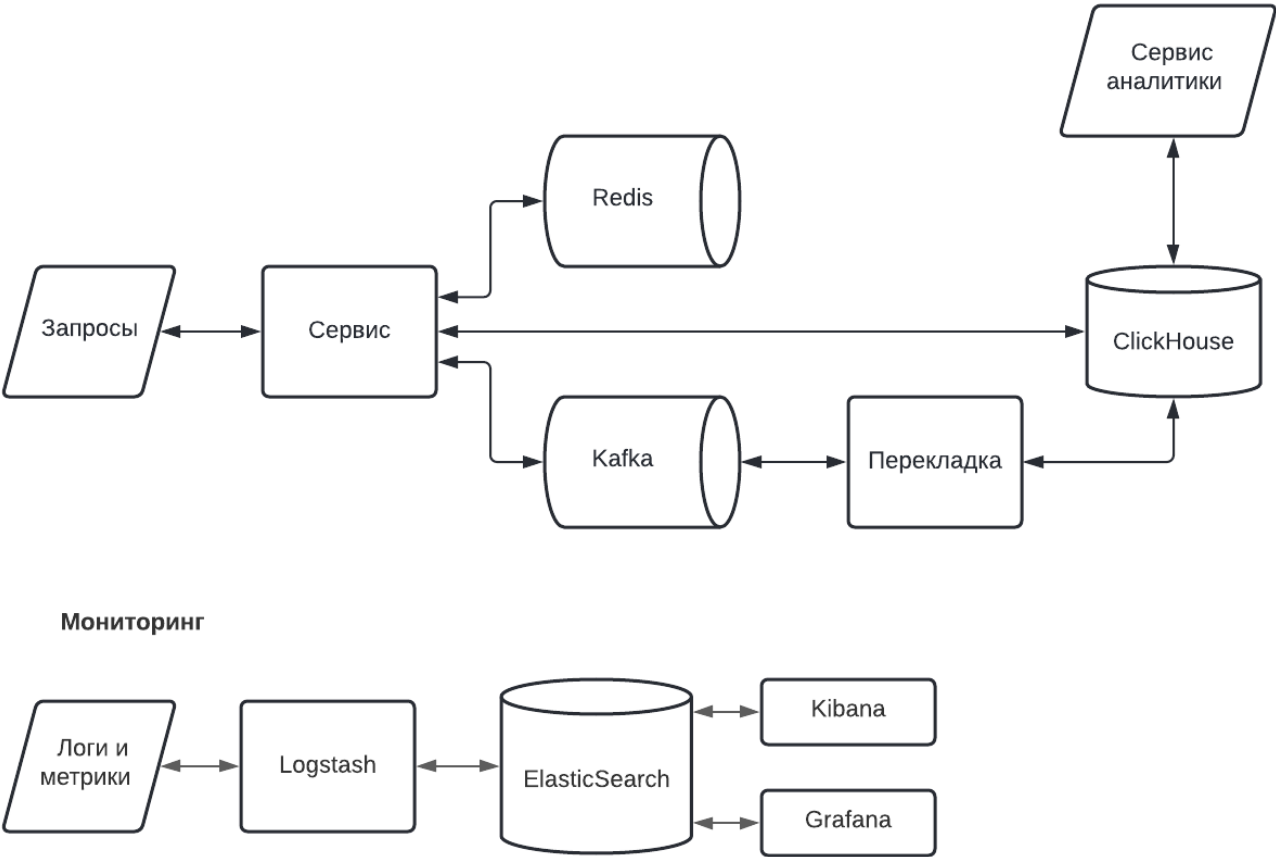


рис. 1

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.07 —01 81				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата