**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Приглашенный преподаватель базовой кафедры ПАО Сбербанк  факультета компьютерных наук  А.И. Калинин  « » 2025 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы  «Программная инженерия» старший преподаватель департамента программной инженерии  Н.А. Павлочев  « » 2025 г. |

## Приложение для создания пешеходных маршрутов

|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* |  |

**Программа и методика испытаний  
ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.05.06-01 51 02-1-ЛУ**

Исполнитель:

Студент группы БПИ224

/ А. Евсюков/

« » 2025 г.

УТВЕРЖДЕНО RU.17701729.05.06-01 51 02-1-ЛУ

## Приложение для создания пешеходных маршрутов

**Программа и методика испытаний**

**RU.17701729.05.06-01 51 02-1**

|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* |  |

**Листов 22**

# АННОТАЦИЯ

Программа и методика испытаний — это документ, в котором содержится информация о программном продукте, а также полное описание приемочных испытаний для данного программного продукта.

Настоящая Программа и методика испытаний для «Приложение для создания пешеходных маршрутов» содержит следующие разделы: «Объект испытаний», «Цель испытаний», «Требования к программе», «Требования к программной документации», «Средства и порядок испытаний», «Методы испытаний», «Приложения».

В разделе «Объект испытаний» указано наименование, краткая характеристика и назначение программы.

В разделе «Цель испытаний» указана цель проведения испытаний.

Раздел «Требования к программе» содержит основные требования к программе, которые подлежат проверке во время испытаний (требования к функционалу и интерфейсу).

Раздел «Требования к программным документам» содержит состав программной документации, которая представляется на испытания.

Раздел «Средства и порядок испытаний» содержит информацию о технических и программных средствах, которые следует использовать во время испытаний, а также порядок этих испытаний.

Раздел «Методы испытаний» содержит информацию об используемых методах испытаний.

Настоящий документ разработан в соответствии с требованиями:

1. ГОСТ 19.101-77 Виды программ и программных документов;
2. ГОСТ 19.102-77 Стадии разработки;
3. ГОСТ 19.103-77 Обозначения программ и программных документов;
4. ГОСТ 19.105-78 Общие требования к программным документам;
5. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом;
6. ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению;

# СОДЕРЖАНИЕ

[1. Объект испытаний 5](#_Toc198002719)

[1.1. Наименование: 5](#_Toc198002720)

[1.2. Краткая характеристика области применения программы: 5](#_Toc198002721)

[2. Цель испытаний 6](#_Toc198002722)

[3. Требования к программе 7](#_Toc198002723)

[3.1. Требования к функциональным характеристикам 7](#_Toc198002724)

[3.1.1. Состав выполняемых функций 7](#_Toc198002725)

[3.1.2. Организация входных данных 7](#_Toc198002726)

[3.1.3. Организации выходных данных 7](#_Toc198002727)

[3.2. Требования к интерфейсу 8](#_Toc198002728)

[4. Требования к программной документации 9](#_Toc198002729)

[4.1. Состав программной документации 9](#_Toc198002730)

[5. Средства и порядок испытаний 10](#_Toc198002731)

[5.1. Технические средства 10](#_Toc198002732)

[5.2. Программные средства 10](#_Toc198002733)

[5.3. Порядок проведения испытаний 10](#_Toc198002734)

[5.4. Требования к персоналу 12](#_Toc198002735)

[6. Методы испытаний 14](#_Toc198002736)

[6.1. Подготовка к проведению испытаний 14](#_Toc198002737)

[6.2. Проверка требований к технической документации 14](#_Toc198002738)

[6.3. Проверка требований к функциональным характеристикам всех эндпоинтов 14](#_Toc198002739)

[6.3.1. Security Service 14](#_Toc198002740)

[6.3.2. Data Provider 15](#_Toc198002741)

[ПРИЛОЖЕНИЕ 1](#_Toc198002742) [ТЕРМИНОЛОГИЯ 18](#_Toc198002743)

[ПРИЛОЖЕНИЕ 2](#_Toc198002744) [СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 20](#_Toc198002745)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 23](#_Toc198002746)

# Объект испытаний

## Наименование:

**Наименование темы разработки**: «Приложение для создания пешеходных маршрутов».

**Наименование темы разработки на английском языке**: «Application for Creating Walking Routes».

## Краткая характеристика области применения программы:

Приложение для создания пешеходных маршрутов предоставит пользователям возможности для планирования прогулок и исследования новых мест.

Основная функция приложения — создание персонализированных маршрутов, где пользователи могут добавлять важные точки и сохранять маршрут в черновик для последующего редактирования. Кроме того, в приложении будет реализован поиск маршрутов с использованием фильтров и сортировки, что позволит находить маршруты, соответствующие предпочтениям пользователя. При прохождении маршрутов будет предусмотрена возможность ставить их на паузу и возвращаться к прогулке позже. Дополнительно будет реализована функция сохранения маршрутов в избранное, чтобы пользователи могли быстро вернуться к понравившимся вариантам.

В отличие от стандартных картографических сервисов, предлагающих маршруты для транспорта или спортивных приложений, ориентированных на поиск маршрутов для бега, наше приложение фокусируется именно на пеших маршрутах, позволяя пользователям создавать, сохранять и делиться маршрутами, что делает продукт инструментом для планирования прогулок.

# Цель испытаний

Целью испытаний является проверка корректности выполнения программой функций, изложенных в п. 4 «Требования к программе» документа «Техническое задание» из комплекта документации в соответствии с ЕСПД (Единой системой программной документации).

# Требования к программе

## Требования к функциональным характеристикам

## Состав выполняемых функций

1. Основные функции:
   * 1. Сервис взаимодействия с базой данных:

* Хранение информации о пользователях, маршрутах и связанных с ними данных.
* Предоставление хранимых данных по средством создания RESTful API
  + 1. Сервис авторизации и аутентификации:
* Механизмы аутентификации и авторизации пользователей.
* Создание и валидация JWT токенов пользователей.
  + 1. Сервис управления взаимодействия фронтенда и бэкенда:
* Обеспечение централизованной точки входа в приложение.
* Валидация запросов к приложению.

1. Основные хранимые данные:

* Пользователи.
* Маршруты.
* Категории.
* Избранные маршруты.
* Пройденные маршруты.
* Черновики маршрутов.
* Отзывы о маршрутах.

## Организация входных данных

Обработка POST запросов от фронтенд-части приложения по добавлению новых пользователей, маршрутов, оценок и отзывов.

## Организации выходных данных

Обработка GET запросов при выводе информации отображающей основной функционал из пункта 3.1.1.

## Требования к интерфейсу

Поскольку разрабатываемое приложение представляет собой бэкенд-часть мобильного приложения, к интерфейсу требования не предъявляются.

# Требования к программной документации

## Состав программной документации

На испытание должна быть представлена документация в следующем составе:

1. «Приложение для создания пешеходных маршрутов». Техническое задание (ГОСТ 19.201-78);
2. «Приложение для создания пешеходных маршрутов». Пояснительная записка (ГОСТ 19.404-79);
3. «Приложение для создания пешеходных маршрутов». Руководство оператора (ГОСТ 19.505-79);
4. «Приложение для создания пешеходных маршрутов». Программа и методика испытаний (ГОСТ 19.301-79);
5. «Приложение для создания пешеходных маршрутов». Текст программы. (ГОСТ 19.401-78);

**4.2. Специальные требования к программной документации**

Все документы к программе должны быть выполнены в соответствии с [ГОСТ 19.106-78](https://docs.google.com/document/d/11r44eTU7FLhUXV_17txmIn8BmtwUEnwuQ46Jne51a7Y/edit#heading=h.44sinio) и ГОСТ к этому виду документа (см. п. 4.1.);

Пояснительная записка должна быть загружена в систему Антиплагиат через SmartLMS «НИУ ВШЭ».

Техническое задание и пояснительная записка, титульные листы других документов должны быть подписаны руководителем разработки и исполнителем.

Документация и программа сдается в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar;

За две недели дня до начала защит комиссии все материалы курсового проекта: программная документация, программный проект, исполняемый файл, отзыв руководителя отчет системы Антиплагиат должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект» в личном кабинете в информационной образовательной среде SmartLMS НИУ ВШЭ.

# Средства и порядок испытаний

## Технические средства

Для работы программы необходим следующий состав технических средств:

1. Доступ к сети интернет.
2. Серверная инфраструктура (или персональный компьютер), которая будет отвечать следующим минимальным требованиям:

* Процессор: 2 ядра
* Оперативная память: 4 Гб
* Публичный IP-адрес (при тестировании на серверной инфраструктуре)
* Дисковое хранилище: 20 Гб

## Программные средства

Во время испытаний должны быть использованы следующие программные средства:

1. Установленная Java версии 21 или выше;
2. Установленная система автоматической сборки Gradle 8.11.1 или выше
3. Docker 24.0.0 или выше.

## Порядок проведения испытаний

Испытания должны проводиться в следующем порядке:

1. Ознакомиться с технической документацией, основным функционалом системы и требованиями к интерфейсу.
2. После загрузки всех сервисов открыть сервис Data Provider и, перейдя в директорию dev-env, выполнить запуск файла docker-compose.yaml.
3. После запуска убедиться в наличии следующих работающих сервисов в Docker:  
   walking\_routes\_app\_db, zookeeper, kafka, kafka-ui, prometheus, grafana.
4. Перейти в Kafka UI по адресу <http://localhost:28090/ui/clusters/local/all-topics> (при необходимости заменить localhost на адрес сервера) и убедиться в наличии нужного топика.  
   Также проверить, что Prometheus отображает 4 job'а со статусом "active" на портах 8080, 8081, 8082 и 8083.
5. Откройте сервис API Gateway и укажите следующие переменные окружения (environment variables):

* DEV\_ENV\_HOST\_PROVIDER — хост, на котором расположен сервис Data Provider (по умолчанию: localhost на персональном компьютере).
* DEV\_ENV\_HOST\_SECURITY — хост, на котором расположен сервис Security Service (по умолчанию: localhost).
* JWT\_SECRET — секретный ключ в формате Base64 для генерации и валидации JWT-токенов. Значение должно совпадать с аналогичной переменной в Security Service.
* SERVER\_PORT — порт, на котором запускается сервис API Gateway (по умолчанию: 8080).

1. Откройте сервис Data Provider и укажите следующие переменные окружения:

* DEV\_ENV\_HOST — хост, на котором развернута база данных (по умолчанию: localhost).
* DEV\_ENV\_DB\_USER — имя пользователя для подключения к базе данных (по умолчанию: admin).
* DEV\_ENV\_DB\_PASSWORD — пароль пользователя базы данных (по умолчанию: admin).
* SERVER\_PORT — порт запуска сервиса Data Provider (по умолчанию: 8081).
* DEV\_ENV\_S3\_BUCKET — имя S3-бакета, в котором будут храниться файлы.
* DEV\_ENV\_S3\_ENDPOINT — URL-адрес S3-совместимого хранилища.
* DEV\_ENV\_S3\_REGION — регион хранилища (например, us-east-1).
* DEV\_ENV\_S3\_ACCESS\_KEY — ключ доступа к S3-хранилищу.
* DEV\_ENV\_S3\_SECRET\_KEY — секретный ключ доступа к S3-хранилищу.

1. Откройте сервис Security Service и укажите следующие переменные окружения:

* DEV\_ENV\_HOST\_DATA — хост, на котором расположен сервис Data Provider.
* DEV\_ENV\_HOST\_NOTIFICATION — хост, на котором работает Notification Service.
* JWT\_SECRET — секрет для JWT-токенов (тот же, что и в API Gateway).
* JWT\_EXPIRATION — время жизни JWT-токенов (например, 3600000 миллисекунд для 1 часа).
* SERVER\_PORT — порт запуска сервиса Security Service (по умолчанию: 8082).

1. Откройте сервис Notification Service и укажите следующие переменные окружения:

* MAIL\_SENDER\_USERNAME — имя пользователя для почтового сервера (обычно почтовый адрес).
* MAIL\_SENDER\_PASSWORD — пароль или токен приложения для аутентификации почтового отправителя.
* SERVER\_PORT — порт запуска сервиса Notification Service (по умолчанию: 8083).

1. Запустите сервисы API Gateway, Security Service, Data Provider, Notification Service.
2. Перейти по адресу <http://localhost:8080/swagger-ui.html> и убедиться в наличии спецификаций API для Data Provider и Security Service.
3. Открыть спецификацию Security Service по адресу <http://localhost:8080/swagger-ui/index.html?urls.primaryName=security-service> и проверить работоспособность функций авторизации и аутентификации, а затем — других доступных функций.
4. Проверить появление новых записей в базе данных после выполнения запросов.
5. Открыть спецификацию Data Provider по адресу <http://localhost:8080/swagger-ui/index.html?urls.primaryName=data-provider> и протестировать работоспособность описанных в ней функций.
6. Проверить состояние основных хранимых данных, описанных в пункте 3.1.1, в базе данных.

## Требования к персоналу

Для корректной работы программы достаточно одного человека. Оператор должен иметь базовые представления о работе с Docker и контейнеризацией, а также обладать следующими навыками:

* Умение запускать и останавливать Docker-контейнеры с использованием docker-compose;
* Знание основ работы с REST API и инструментами вроде Swagger UI;
* Понимание принципов работы переменных окружения и умение их настраивать;
* Базовые навыки работы с командной строкой (терминалом);
* Желательно наличие общего понимания микросервисной архитектуры и взаимодействия сервисов через HTTP и Kafka.

# Методы испытаний

## Подготовка к проведению испытаний

Испытания представляют собой процесс установления следующих показателей:

1. Проверка документации и программы на соответствие установленным в техническом задании требованиям (в комплект поставки программы входит архив с документацией, готовой реализацией программного продукта и презентацией проекта);
2. Проверка корректности работы API, описанного в спецификации Swagger;
3. Проверка корректного хранения и обработки основных данных, предусмотренных системой.

## Проверка требований к технической документации

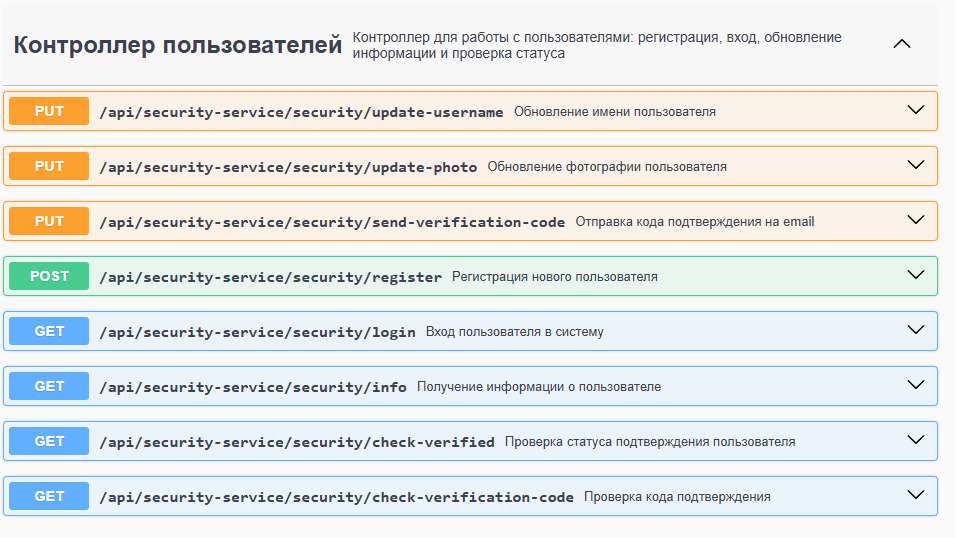
Состав программной документации проверяется наличием всех подписей и программной документации в системе LMS. Также проверяется соответствие документации требованиям ГОСТ.

Все документы удовлетворяют представленным требованиям.

## Проверка требований к функциональным характеристикам всех эндпоинтов

## Security Service

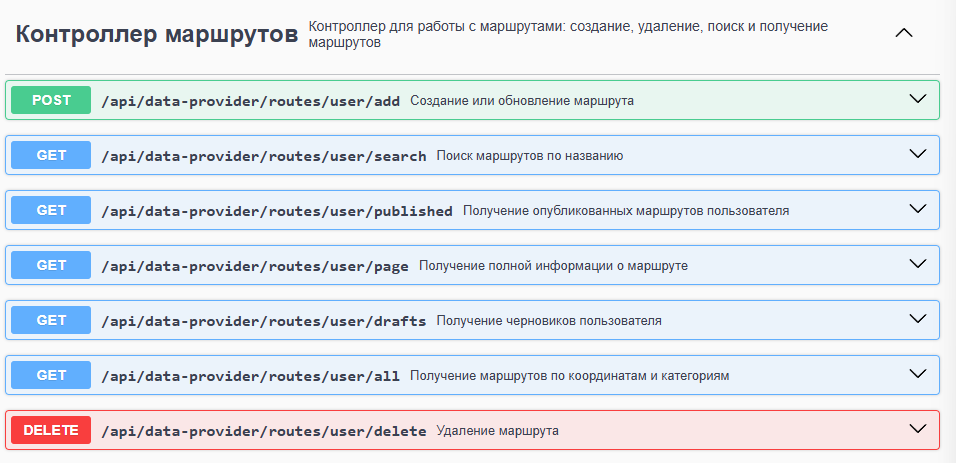
* Проверка корректной работы эндпоинтов, предоставляемых UserController (см. рис. 1), включая:
* регистрацию пользователя;
* авторизацию (вход);
* обновление пользовательской информации;
* получение информации о пользователе;
* проверка кода подтверждения;
* проверку статуса текущего пользователя.



*Рисунок 1 – эндпоинты контроллера пользователей*

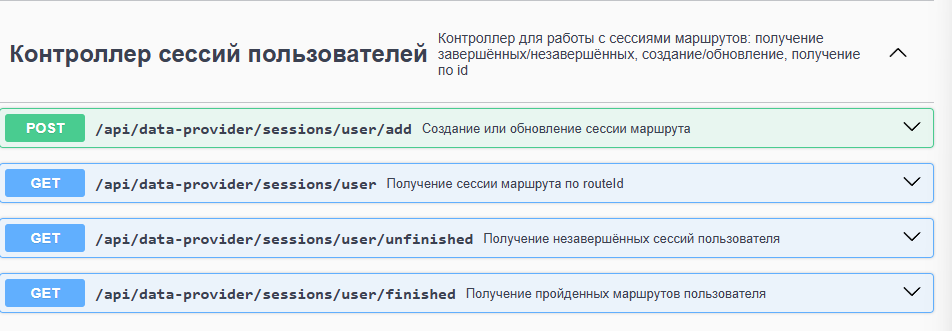
## Data Provider

* Проверка корректной работы эндпоинтов, предоставляемых RouteController (см. рис. 2), включая:
* создание нового маршрута;
* удаление маршрута;
* поиск маршрутов по различным параметрам;
* получение маршрутов пользователя (черновиков и опубликованных);
* получение подробной информации о маршруте.



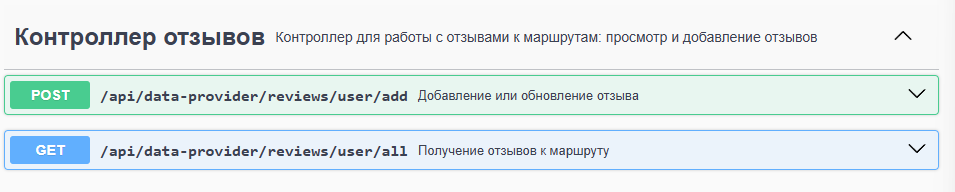
*Рисунок 2 – эндпоинты контроллера маршрутов*

* Проверка корректной работы эндпоинтов, предоставляемых RouteSessionController (см. рис. 3), включая:
* получение списка завершённых и незавершённых сессий маршрутов;
* создание новой сессии маршрута;
* обновление существующей сессии;
* получение сессии маршрута по идентификатору.

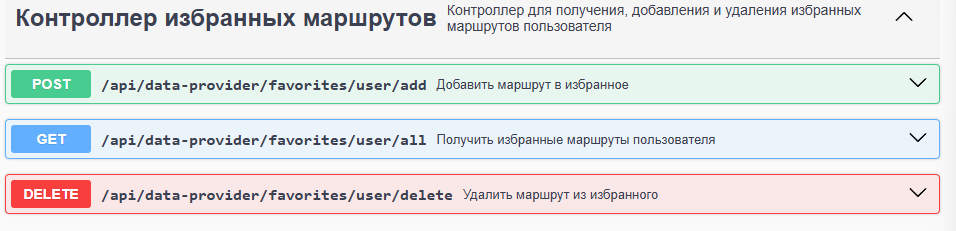


*Рисунок 3 – эндпоинты контроллера сессий пользователей*

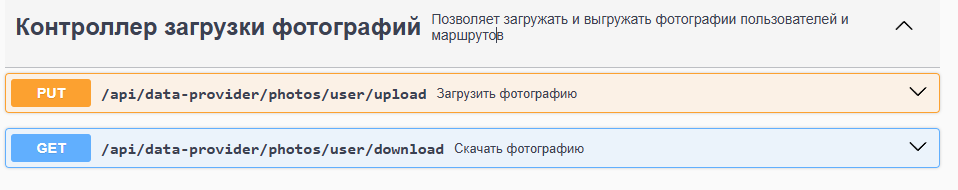
* Проверка корректной работы эндпоинтов, предоставляемых ReviewController (см. рис. 4), включая:
* добавление нового отзыва к маршруту;
* просмотр отзывов по маршрутам.

*Рисунок 4 – эндпоинты контроллера отзывов*

* Проверка корректной работы эндпоинтов, предоставляемых FavoriteController (см. рис. 5), включая:
* получение списка избранных маршрутов пользователя;
* добавление маршрута в избранное;
* удаление маршрута из избранного.

*Рисунок 5 – эндпоинты контроллера избранных маршрутов*

* Проверка корректной работы эндпоинтов, предоставляемых PhotoController (см. рис. 6), включая:
* загрузку фотографий пользователей и маршрутов;
* получение (скачивание) загруженных фотографий.

*Рисунок 6 – эндпоинты контроллера фотографий*

# ПРИЛОЖЕНИЕ 1

# ТЕРМИНОЛОГИЯ

Таблица 1

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| База данных | Совокупность данных, организованных в соответствии с концептуальной структурой, описывающей характеристики этих данных и взаимоотношения между ними, которая поддерживает одну или более областей применения |
| Бэкенд | Часть веб-приложения, отвечающая за обработку данных и бизнес-логику, которая скрыта от пользователя. |
| Программное обеспечение | Совокупность программных и документальных средств для  создания и эксплуатации систем обработки данных средствами  вычислительной техники. |
| RESTful API | Архитектурный стиль взаимодействия компонентов приложения с использованием стандартных HTTP-запросов для управления ресурсами. |
| Микросервис | Независимые компоненты приложения, каждый из которых выполняет определённую бизнес-задачу и взаимодействует с другими через API. |
| JWT-токен | Компактный формат передачи данных, используемый для аутентификации и авторизации, обеспечивающий безопасность взаимодействия между клиентом и сервером. |
| Программное обеспечение | Совокупность программных и документальных средств для  создания и эксплуатации систем обработки данных средствами  вычислительной техники. |
| RESTful API | Архитектурный стиль взаимодействия компонентов приложения с использованием стандартных HTTP-запросов для управления ресурсами. |
| Дашборд (панель мониторинга) | Интерактивная визуальная панель, отображающая ключевые метрики системы или приложения в реальном времени, используемая для анализа состояния и выявления аномалий. |
| JSON (JavaScript Object Notation) | Лёгкий текстовый формат обмена данными, используемый для передачи информации между клиентом и сервером. |
| Load Balancer | Механизм распределения входящего сетевого трафика между несколькими серверами для обеспечения отказоустойчивости и производительности. |
| Producer (производитель) | Компонент, отправляющий сообщения в Kafka-топик. |
| Consumer (потребитель) | Компонент, подписывающийся на Kafka-топик и обрабатывающий поступающие сообщения. |
| Топик (Topic) | Категория или канал в Apache Kafka, куда публикуются и из которого читаются сообщения. |
| Микросервисная архитектура | Архитектурный стиль, при котором приложение состоит из набора мелких, независимых сервисов, взаимодействующих друг с другом через API. |
| R-дерево | Структура данных для индексирования многомерной информации, такой как географические координаты, используемая в PostGIS. |
| Эндпоинт | Конечная точка веб-сервиса, к которой клиентское приложение обращается для выполнения определённых операций или получения данных. |

# ПРИЛОЖЕНИЕ 2

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

* + 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 19.201-78 Техническое задание. Требования к содержанию и оформлению.

//Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

* + 1. ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
    2. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
    3. ГОСТ 15150-69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.
    4. ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
    5. Docker Documentation. [Электронный ресурс], URL: https://docs.docker.com/ (дата обращения: 05.05.2025).
    6. Official PostgreSQL Documentation. [Электронный ресурс], URL: https://www.postgresql.org/docs/ (дата обращения: 05.05.2025).
    7. PostGIS Documentation [Электронный ресурс], URL: https://postgis.net/documentation/ (дата обращения: 05.05.2025).
    8. Spring Framework Reference Documentation. [Электронный ресурс], URL: https://docs.spring.io/spring-framework/docs/current/reference/html/ (дата обращения: 05.05.2025).
    9. Strava. [Мобильное приложение], URL: https://www.strava.com/ (дата обращения: 05.05.2025).
    10. AllTrails. [Мобильное приложение], URL: https://www.alltrails.com/ (дата обращения: 05.05.2025).
    11. Яндекс Карты. [Мобильное приложение], URL: https://yandex.ru/maps (дата обращения: 05.05.2025).
    12. Outdooractive. [Мобильное приложение], URL: https://www.outdooractive.ru/ (дата обращения: 05.05.2025).
    13. Wikiloc. [Мобильное приложение], URL: https://ru.wikiloc.com/ (дата обращения: 05.05.2025).
    14. Komoot. [Мобильное приложение], URL: https://www.komoot.com/ (дата обращения: 05.05.2025).
    15. Spring MVC – Framework. [Электронный ресурс], URL: <https://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm> (дата обращения: 05.05.2025)
    16. Spring Security. [Электронный ресурс], URL: <https://docs.spring.io/spring-security/reference/index.html> (дата обращения: 05.05.2025)
    17. Spring Data JDBC. [Электронный ресурс], URL: <https://docs.spring.io/spring-data/jdbc/reference/> (дата обращения: 05.05.2025)
    18. Spring Cloud Gateway. [Электронный ресурс], URL: <https://docs.spring.io/spring-cloud-gateway/reference/> (дата обращения: 05.05.2025)
    19. Spring for Apache Kafka. [Электронный ресурс], URL: <https://docs.spring.io/spring-kafka/reference/> (дата обращения: 05.05.2025)
    20. Prometheus для Spring Boot. [Электронный ресурс], URL: https://micrometer.io/docs/registry/prometheus (дата обращения: 05.05.2025)
    21. Grafana. [Электронный ресурс], URL: https://grafana.com/docs/grafana/latest/ (дата обращения: 05.05.2025)

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в  докум.) | №  документа | Входящий  № сопроводит ельного  докум. и дата | Подп. | Дата |
| Изм. | Изменен ных | Заменен ных | Новых | Аннули рованх |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |