

ООО «Финтек Системс»

«Сервис управления коммуникационными стратегиями»
Инструкция пользователя

2023 г.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОМ ОБЕСПЕЧЕНИИ	3
1.1. Назначение программного обеспечения	3
1.2. Область применения программного комплекса	3
1.3. Функциональные характеристики	3
2. УПРАВЛЕНИЕ ДОСТУПОМ	5
2.1. Основные положения	5
3. ОТПРАВКА СООБЩЕНИЙ МЕТОДОМ POST	6
3.1. Структура сообщений	6
3.2. Примеры запросов JSON	7
3.3. Ответы	10
4. ПОЛУЧЕНИЕ СТАТУСОВ СООБЩЕНИЙ МЕТОДОМ GET	12
4.1. Структура запроса	12
4.2. Примеры запросов JSON	12
4.3. Ответы	12

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

1.1. Назначение программного обеспечения

1.1.1. «Сервис управления коммуникационными стратегиями» — программное обеспечение (далее – ПО), предназначенное для обеспечения маршрутизации с интегрированными провайдерами рассылок, приоритезацию трафика и контроль статусов обработки сообщений.

1.2. Область применения программного комплекса

1.2.1. Программное обеспечение обеспечивает возможность маршрутизации рассылок между интегрированными провайдерами и веб – сервисом Пользователя посредством распределенной системы обмена сообщениями, в том числе пакетной отправки сообщений, в сети Интернет. Программный комплекс является связующим звеном между провайдером и веб – сервисом Пользователя.

1.3. Функциональные характеристики

1.3.1. Программное обеспечение используется для реализации следующих функций:

- Получение запросов на отправку отдельного сообщения или пакета сообщений формате JSON по REST API или Kafka;
- Маршрутизация сообщений между интегрированными провайдерами сообщений;
- Создание очереди сообщений с приоритезацией посредством интеграции с сервером структур данных Redis;
- Контроль времени отправки сообщения;
- Пакетная отправка сообщений провайдеру для снижения нагрузки на сеть;
- Запись и хранение входящих запросов на отправку сообщений и полученных ответов от провайдеров в базе данных MongoDB с последующей интеграцией с системой Prometheus для анализа данных мониторинга.

- Сбор системной информации о работе ПО (логирование взаимодействия) посредством системы управления логами GrayLog.
- Предоставление доступа к эндпоинтам для подключения систем мониторинга.

2. УПРАВЛЕНИЕ ДОСТУПОМ

2.1. Основные положения

2.1.1. Для доступа к программному обеспечению авторизация не требуется.

3. ОТПРАВКА СООБЩЕНИЙ МЕТОДОМ POST

3.1. Структура сообщений

3.1.1. Структура сообщения приведена в таблице 1.

Таблица 1

№ п.п	Параметр	Тип данных	Описание	Обязательность поля
1	send	object	Объект с массивом данных отправляемого сообщения	
1.1	text_female	string	Текст сообщения для адресата женского пола. Кодировка — UTF-8.	НЕТ
1.2	text_male	string	Текст сообщения для адресата мужского пола. Кодировка — UTF-8.	ДА
2	fields	object	Массив с данными адресата	НЕТ
2.1	fields => name	string	Имя адресата	НЕТ
3	users	object	Массив объектов с информацией об адресате	ДА
3.1	users => gender	string	Информация о половой принадлежности адресата	ДА, если данные указаны в "text female"
3.2	users => phone	string	Номер телефона адресата в формате: +79876543210	ДА
3.3	users => name	string	Имя адресата	ДА, если данные указаны в структуре "fields"
4	route	string	Наименование провайдера	ДА
5	ttl	number	Предельное значение периода времени актуальности отправляемого сообщения, проверяемое перед отправкой. При превышении предельного значения периода времени актуальности отправки сообщение не отправляется.	НЕТ
6	mode	number	Режим выполнения запроса	НЕТ
7	is_force_debug sensitive	null	Параметр для отладки	НЕТ
8	voice	number	Отправка голосового сообщения	НЕТ

3.2.Примеры запросов JSON

3.2.1. Пример запроса JSON: отправка СМС сообщения адресату (с указанием пола):

POST/api/v1
Content-type: application/json
<pre>{ "ttl": null, "is_force_debug_sensitive": null, "send": { "text_female": "Уважаемая {{name}}", "text_male": "Уважаемый {{name}}", "fields": ["name"], "users": [{ "gender": "male", "phone": "+79876543210", "name": "Alexey" }, { "gender": "female", "phone": "+79876543211", "name": "Mariia" }] }, "route": "route_name", "voice": null, "mode": null }</pre>

3.2.2. Пример запроса JSON: отправка СМС сообщения адресату (без указания пола):

POST/ api/v1
Content-type: application/json
<pre>{ "ttl": null, "is_force_debug_sensitive": null, "send": { "text_female": null, "text_male": "Добрый день, {{name}}", "fields": ["name"], "users": [{ "gender": null, "phone": "+79876543210", "name": "Alexey" }, { "gender": null, "phone": "+79876543211", "name": "Mariia" }] }, "route": "route_name", "voice": 0, "mode": null }</pre>

3.2.3. Пример запроса JSON: отправка голосового сообщения одному адресату:

POST/api/v1
Content-type: application/json
<pre>{ "timeLimit": "2023-05-15T07:22:31.898926+00:00", "is_force_debug_sensitive": null, "send": { "text_alternative": "Добрый день #1 {{name}}", "text_male": "Добрый день {{name}}", "fields": ["name"], "users": [{ "phone": "+79876543210", "name": "Alexey" }] }, "route": "route_name", "voice": 1, "mode": null }</pre>

3.2.4. Пример запроса JSON: отправка голосового сообщения двум адресатам:

POST/ api/v1
Content-type: application/json
<pre>{ "timeLimit": "2023-05-15T07:22:31.8991433+00:00", "is_force_debug_sensitive": null, "send": { "text_alternative": " Добрый день #1 {{name}}", "text_male": " Добрый день {{name}}", "fields": ["name"], "users": [{ "phone": "+79876543210", "name": "Alexey" }, { "phone": "+79876543211", "name": "Mariia" }] }, "route": "route_name", "voice": 1, "mode": null }</pre>

3.3. Ответы

3.3.1. Код успешного ответа — 200. Успешный ответ означает, что сообщение **принято платформой для обработки сообщения**. Код ответа 200 не связан со статусом сообщения 200 и не означает, что сообщение отправлено. Структура ответа представлена в таблице 2. Заголовки и их структура представлены в таблице 3.

Таблица 2

Параметр ответа	Тип данных	Описание
status	string	Статус отправки сообщения
message	string	Описание статуса отправки
key	string	Ключ запроса

Пример успешного ответа сервера

```
[
  {
    "send": {
      "status": "ok",
      "message": "Сообщения в очереди на отправку",
      "key": "ecefff5b-55c1-4089-b49d-6fd9555b5cda"
    }
  }
]
```

Таблица 3

Параметр заголовка	Тип данных	Описание
content-length	string	Количество символов сообщения
content-type	string	Тип передаваемого контента, формат кодирования символов сообщения
date	string	Дата
server	string	Наименование сервера
x-served-by	string	Сервер обрабатывающий запрос

Пример передаваемых заголовков

```
content-length: 142
content-type: application/json; charset=utf-8
date: Tue, 16 May 2023 08:44:21 GMT
server: openresty
x-served-by: comgateway.website.ru
```

4. ПОЛУЧЕНИЕ СТАТУСОВ СООБЩЕНИЙ МЕТОДОМ GET

4.1. Структура запроса

4.1.1. Структура запроса приведена в таблице 4.

Таблица 4

№ п.п	Параметр	Тип данных	Описание	Обязательность поля
1	status	object	Объект с массивом данных статуса сообщения	
1.1	key	string	Ключ запроса	ДА
1.2	route	string	Наименование провайдера	ДА

4.2. Примеры запросов JSON

4.2.1. Пример запроса JSON: запрос на получение статуса сообщения:

GET/api/v1
Content-type: application/json
<pre>{ "status": { "key": "2405d82a789e14155f484fabb1e68c65" }, "route": "route1" }</pre>

4.3. Ответы

4.3.1. Код успешного ответа — 200. Структура ответа представлена в таблице 5.

Таблица 2

Параметр ответа	Тип данных	Описание
status	string	Статус отправки сообщения
message	string	Описание статуса запроса
key	string	Ключ запроса

Пример успешного ответа сервера

```
[
  {
    "status": {
      "status": "ok",
      "message": "Статус запроса получен",
      "key": "2405d82a789e14155f484fabb1e68c65",
      "data": "debug"
    }
  }
]
```

Пример ответа сервера: сообщение не отправлено

```
[
  {
    "status": {
      "status": "err",
      "message": "Задание не выполнено",
      "key": "2405d82a789e14155f484fabb1e68c66"
    }
  }
]
```