

ООО «Финтек Системс»

«Сервис рассылки сообщений»
Инструкция пользователя

2023 г.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОМ ОБЕСПЕЧЕНИИ	3
1.1. Назначение программного обеспечения	3
1.2. Область применения программного обеспечения	3
1.3. Функциональные характеристики	3
2. УПРАВЛЕНИЕ ДОСТУПОМ	4
2.1. Основные положения	4
3. ОТПРАВКА ЗАПРОСОВ	5
3.1. Примеры запросов	5

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

1.1. Назначение программного обеспечения

1.1.1. Программное обеспечение «Сервис рассылки сообщений» – программа, обеспечивающая регулярной автоматизированной рассылки сообщений по электронной почте и SMS по передаваемым контактам клиентов веб-сервисов из системы для автоматизации и контроля взаимодействия компании с клиентами (CRM системы).

1.2. Область применения программного обеспечения

1.2.1. Программное обеспечение обеспечивает рассылку информационных сообщений по электронной почте и SMS-сообщений с последующим получением статусов доставки и возможностью настройки расписания рассылки.

1.3. Функциональные характеристики

1.3.1. Программное обеспечение используется для реализации следующих функций:

- Рассылка информационных сообщений по электронной почте и SMS-сообщений;
- Получение статусов доставки отправленных сообщений в реальном времени;
- Настройка расписания рассылки.

2. УПРАВЛЕНИЕ ДОСТУПОМ

2.1. Основные положения

2.1.1. Для доступа к программному обеспечению авторизация не требуется.

3. ОТПРАВКА ЗАПРОСОВ

3.1. Примеры запросов

3.1.1. Пример запроса: добавление нового проекта:

POST/project/create
Content-type: application/json
<pre>"tags": [" Projects"], "summary": "Добавление нового проекта", "description": "Экшен для создания нового проекта", "operationId": "9e0d270c7c58e8066aee948ce0dc07fa", "requestBody": { "description": "Добавить новый проект", "required": true, "content": { " application/json": { "schema": { "required": [" name", "code"], "properties": { "name": { "description": "The project name", "type": "string", "example": "OCM" }, "code": { "description": " The project code", "type": "string", "example": "010" }, }, }, "type": "object" } } }</pre>

3.1.2. Код успешного ответа — 201. Структура ответа представлена в таблице 1.

Таблица 1

Параметр ответа	Тип данных	Описание
status	integer	Статус
message	string	Сообщение

Пример успешного ответа сервера

```
{
  "status": 201,
  "message": "Ok"
}
```

3.1.3. Код ответа 400: ошибка валидации. Структура ответа представлена в таблице 2.

Таблица 2

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 400: ошибка валидации

```
{
  "code": 400,
  "error": {
    "message": "Ошибка валидации",
    "status": 400
  }
}
```

3.1.4. Код ответа 403: доступ запрещен. Структура ответа представлена в таблице 3.

Таблица 3

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 403: доступ запрещен

```
{
  "code": 403,
  "error": {
    "message": "Доступ запрещен",
    "status": 403
  }
}
```

3.1.5. Код ответа 405: метод не поддерживается. Структура ответа представлена в таблице 4.

Таблица 4

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 405: метод не поддерживается

```
{
  "code": 405,
  "error": {
    "message": "Метод не поддерживается",
    "status": 405
  }
}
```

3.1.6. Код ответа 500: внутренняя ошибка сервера. Структура ответа представлена в таблице 5.

Таблица 5

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 500: внутренняя ошибка сервера
<pre>{ "code": 500, "error": { "message": "Внутренняя ошибка сервера", "status": 500 } }</pre>

3.1.7. Пример запроса: обновление данных проекта:

PUT/project/{id}/update
Content-type: application/json
<pre>"tags": [" Projects"], "summary": "Обновление данных проекта", "description": "Экшен для обновления данных проекта", "operationId": "9e0d270c7c58e8066aee948ce0dc07fa", "requestBody": { "description": "Обновить данные", "required": true, "content": { " application/json": { "schema": { "required": ["primary key", "name"], "properties": { "primary key": { "description": "Первичный ключ", "type": "integer", "example": "74892489" }, "name": { "description": "Наименование проекта", "type": "string", "example": "ОСМ" }, }, }, "type": "object" } } }</pre>

3.1.8. Код успешного ответа — 201. Структура ответа представлена в таблице 6.

Таблица 6

Параметр ответа	Тип данных	Описание
status	integer	Статус
message	string	Сообщение

Пример успешного ответа сервера

```
{  
  "status": 201,  
  "message": "Ok"  
}
```

3.1.9. Код ответа 400: ошибка валидации. Структура ответа представлена в таблице 7.

Таблица 7

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 400: ошибка валидации

```
{  
  "code": 400,  
  "error": {  
    "message": "Ошибка валидации",  
    "status": 400  
  }  
}
```

3.1.10. Код ответа 403: доступ запрещен. Структура ответа представлена в таблице 8.

Таблица 8

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 403: доступ запрещен

```
{
  "code": 403,
  "error": {
    "message": "Доступ запрещен",
    "status": 403
  }
}
```

3.1.11. Код ответа 404: страница не найдена. Структура ответа представлена в таблице 9.

Таблица 9

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 404: страница не найдена

```
{
  "code": 404,
  "error": {
    "message": "Страница не найдена",
    "status": 404
  }
}
```

3.1.12. Код ответа 405: метод не поддерживается. Структура ответа представлена в таблице 10.

Таблица 10

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 405: метод не поддерживается

```
{
  "code": 405,
  "error": {
    "message": "Метод не поддерживается",
    "status": 405
  }
}
```

3.1.13. Код ответа 500: внутренняя ошибка сервера. Структура ответа представлена в таблице 11.

Таблица 11

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 500: внутренняя ошибка сервера

```
{
  "code": 500,
  "error": {
    "message": "Внутренняя ошибка сервера",
    "status": 500
  }
}
```

3.1.14. Пример запроса: создание нового клиента:

POST/client/create
Content-type: application/json
<pre>"tags": ["Clients"], "summary": "Создание нового клиента", "description": "Экшен для создания нового клиента", "operationId": "9e0d270c7c58e806baee948ce0dc07fa", "requestBody": { "description": "Создать клиента", "required": true, "content": { " application/json": { "schema": { "required": ["project", "guid", "first_name", "last_name", "middle_name"], "properties": { "project": { "description": "Код проекта", "type": "string", "example": "ОСМ" }, "guid": { "description": "guid клиента", "type": "string", "example": "lisjrfengghb^&Tgfb4*Tasdf" } } } } }</pre>

```

        "first_name": {
            "description": "Имя клиента",
            "type": "string",
            "example": "Иван"
        },
        "last_name": {
            "description": "Фамилия клиента",
            "type": "string",
            "example": "Иванов"
        },
        "middle_name": {
            "description": "Отчество клиента",
            "type": "string",
            "example": "Иванович"
        },
    },
    "type": "object"
}
}
}

```

3.1.15. Код успешного ответа — 201. Структура ответа представлена в таблице 12.

Таблица 12

Параметр ответа	Тип данных	Описание
status	integer	Статус
message	string	Сообщение

Пример успешного ответа сервера

```

{
  "status": 201,
  "message": "Ok"
}

```

3.1.16. Код ответа 500: внутренняя ошибка сервера. Структура ответа представлена в таблице 13.

Таблица 13

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 500: внутренняя ошибка сервера

```
{  
  "code": 500,  
  "error": {  
    "message": "Внутренняя ошибка сервера",  
    "status": 500  
  }  
}
```

3.1.17. Пример запроса: создание рассылки:

POST/mailing/create
Content-type: application/json
<pre>"tags": ["Mailing"], "summary": "Создание рассылки", "description": "Создание рассылки", "operationId": "9e0d270c7c58e8066aee948ce0dc07fa", "requestBody": { "description": "Создание рассылки", "required": true, "content": { " application/json": { "schema": { "required": ["name", "project_id", "template_id", "status", "channels", "event",], "properties": { "name": { "description": "Наименование рассылки", "type": "string", "example": "SMSMailing" }, "project_id": { "description": "Идентификатор проекта", "type": "integer", "example": "0236" }, "template_id": { "description": "Идентификатор шаблона", "type": "integer", "example": "156" }, }, }, }, }, }</pre>


```

    "status": {
      "description": "Статус рассылки",
      "type": "integer",
      "example": "1"
    },
    "channels": {
      "description": "Каналы рассылки",
      "type": "array",
      "example": "1"
    },
    "event": {
      "description": "Дата рассылки",
      "type": "string",
      "example": "2020-08-02"
    },
  },
  "type": "object"
}
}
}

```

3.1.18. Код успешного ответа — 201. Структура ответа представлена в таблице 6.

Таблица 6

Параметр ответа	Тип данных	Описание
status	integer	Статус
message	string	Сообщение

Пример успешного ответа сервера
<pre> { "status": 201, "message": "Ok" } </pre>

3.1.19. Код ответа 500: внутренняя ошибка сервера. Структура ответа представлена в таблице 11.

Таблица 11

Параметр ответа	Тип данных	Описание
status	string	Код ошибки
message	string	Описание статуса запроса

Пример ответа 500: внутренняя ошибка сервера

```
{  
  "code": 500,  
  "error": {  
    "message": "Внутренняя ошибка сервера",  
    "status": 500  
  }  
}
```