

# Bot API

## Техническое описание

Февраль 2025

# СОДЕРЖАНИЕ

<b>1 Администрирование Чат-ботов</b>	<b>4</b>
1.1 Право администрирования	4
1.2 Функции администрирования	5
1.3 Взаимодействие с Чат-ботами	8
1.4 Возможности и ограничения	11
<b>2 Общее описание функциональности API</b>	<b>12</b>
<b>3 Опциональное шифрование</b>	<b>14</b>
<b>4 SSE endpoints</b>	<b>16</b>
<b>5 Rest endpoints</b>	<b>18</b>
5.1 Подтверждение о прочитанных и обработанных сообщениях/командах	19
5.2 Отправка сообщений	21
5.2.1 Отправка текстовых сообщений	21
5.2.2 Отправка файловых сообщений	27
5.2.3 Отправка текстовых сообщений, содержащих изображение	29
5.2.4 Отправка текстовых сообщений, содержащих видео	36
5.2.5 Отправка текстовых сообщений, содержащих изображение и видео	42
5.3 Обновление сообщений	51
5.3.1 Обновление текстового сообщения	51
5.3.2 Обновление файлового сообщения	58
5.3.3 Обновление сообщения, содержащего изображение	67
5.3.4 Обновление сообщения, содержащего видео	73
5.3.5 Обновление сообщения, содержащего изображение и видео	80
5.4 Сообщение с последующей клавиатурой и разметка кнопок	88
5.4.1 Сообщение с клавиатурой	88
5.4.2 Типы кнопок	91
5.4.3 Разметка кнопок	95
5.6 Получение сообщений	99

5.7 Получение списка непрочитанных сообщений	102
5.8 Удаление сообщений	105
5.9 Удаление всех кнопок на сообщении	106
5.10 Сохранение файлов в s3 хранилище	107
5.11 Скачивание расшифрованных файлов	115
5.12 Получение всех групп	116
5.13 Получение состояния Чат-бота в определенной группе	121
5.14 Звонки	126
<b>6 Разработка Чат-ботов с использованием специальной библиотеки приложения</b>	<b>127</b>

# 1 Администрирование Чат-ботов

## 1.1 Право администрирования

Чат-боты — это программы для имитация общения с пользователями мессенджера по различным вопросам. Сценарии работы Чат-ботов создают и хранятся вне мессенджера.

Для администрирования Чат-ботов по решению руководства организации назначается специальный работник организации (далее – уполномоченный работник), обладающий необходимыми навыками и знаниями.

Уполномоченному работнику выдается доступ к специальному Чат-боту (**Bot Creator**), с помощью которого осуществляется администрирование Чат-ботов. После получения права администрирования Чат-ботов уполномоченный работник может создавать, редактировать и останавливать работу Чат-ботов.

Для работы с Чат-ботом создается специальный Токен, который присваивается каждому Чат-боту и действует определенное время (по умолчанию один год). Уполномоченный работник осуществляет передачу данного Токена для работы с Чат-ботами другим работникам организации.

Разработчик мессенджера не несет ответственность за сбои в работе приложения по причине некачественно проработанной логики программного кода (скрипта) Чат-ботов.

## 1.2 Функции администрирования

Администрирование Чат-ботов осуществляется в UI мессенджера путем отправки определенных команд специальному Чат-боту (**Bot Creator**). Информация о Bot Creator и порядке его использования доступна в FAQ приложения в разделе «API».

При наличии у пользователя доступа к Bot Creator, он может найти его с помощью глобального поиска.

Уполномоченному работнику организации доступны следующие команды:

`/start` – начало работы с Bot Creator. Пользователю отображается список доступных команд;

`/newbot` – создать нового Чат-бота. При создании Чат-бота необходимо определить его имя. Имя должно включать приставку “bot”, указывается латинскими буквами (например, mychat bot). После создания уполномоченному работнику отображается Токен доступа Чат-бота, который используется для его работы в мессенджере, а также ссылка на профиль Чат-бота;

`/mybots` – показать список ваших Чат-ботов. Данная команда отображает уполномоченному работнику список созданных им Чат-ботов. Имена Чат-ботов отображаются в виде ссылок на их профиль;

`/setname` – изменить имя Чат-бота. После ввода команды необходимо ввести имя Чат-бота для его изменения. Команда позволяет изменить имя существующего Чат-бота;

`/setjobtitle` – изменить описание в профиле Чат-бота. После ввода команды необходимо ввести имя Чат-бота для изменения его описания. Команда позволяет изменить описание профиля существующего Чат-бота;

`/setdescription` – изменить описание Чат-бота. После ввода команды необходимо ввести имя Чат-бота для изменения его описания. Команда позволяет изменить описание существующего Чат-бота;

`/setuserpic` – изменить фотографию профиля Чат-бота. После ввода команды необходимо ввести имя Чат-бота для изменения его изображения. Команда позволяет изменить изображение существующего Чат-бота;

`/activate` – активировать работу Чат-бота. После ввода команды необходимо ввести имя Чат-бота для его активации. Функция доступна только для деактивированных Чат-ботов;

`/stop` – остановить работу Чат-бота. После ввода команды необходимо ввести имя Чат-бота для остановки его работы. Функция доступна только для активных Чат-ботов;

`/token` – сгенерировать новый Токен доступа Чат-бота. После ввода команды необходимо ввести имя Чат-бота для создания нового Токена доступа. Предыдущий Токен доступа деактивируется;

`/extend` – продление срока действия Токена доступа Чат-бота на один год с момента выполнения команды;

`/revoke` – отозвать Токен доступа Чат-бота. После ввода команды необходимо ввести имя Чат-бота для прекращения действия Токена доступа. Токен доступа деактивируется.

Приветствую! Я могу помочь вам создать ботов и управлять ими.

Вы можете управлять мной, посылая эти команды:

**/newbot** - создать нового бота

**/mybots** - список ваших ботов

Редактирование бота

**/setname** - изменить имя бота

**/setjobtitle** - изменить описание в профиле бота

**/setdescription** - изменить описание бота

**/setuserpic** - изменить фотографию профиля бота

**/activate** - активировать работу бота

**/stop** - остановить работу бота

Настройки бота

**/token** - сгенерировать новый токен доступа бота

**/extend** - продлить срок жизни токена доступа бота

**/revoke** - отозвать токен доступа

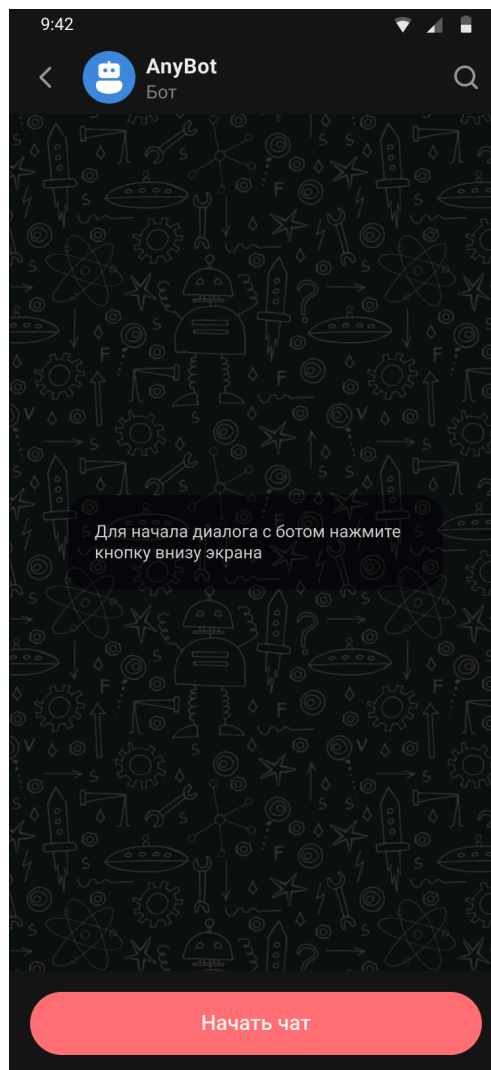


16:44

Пример отображения команд администрирования

## 1.3 Взаимодействие с Чат-ботами

Для начала диалога с Чат-ботом добавлена кнопка “Начать чат”, по нажатию на которую будет отправляться стартовая команда Чат-бота. Далее бот будет присылать сообщения и взаимодействовать с пользователем согласно заложенному сценарию. Типы сообщений и команды, которые может обрабатывать Чат-бот, определяются его разработчиком. Взаимодействие с Чат-ботом может осуществляться с помощью кликабельных команд и кнопок.

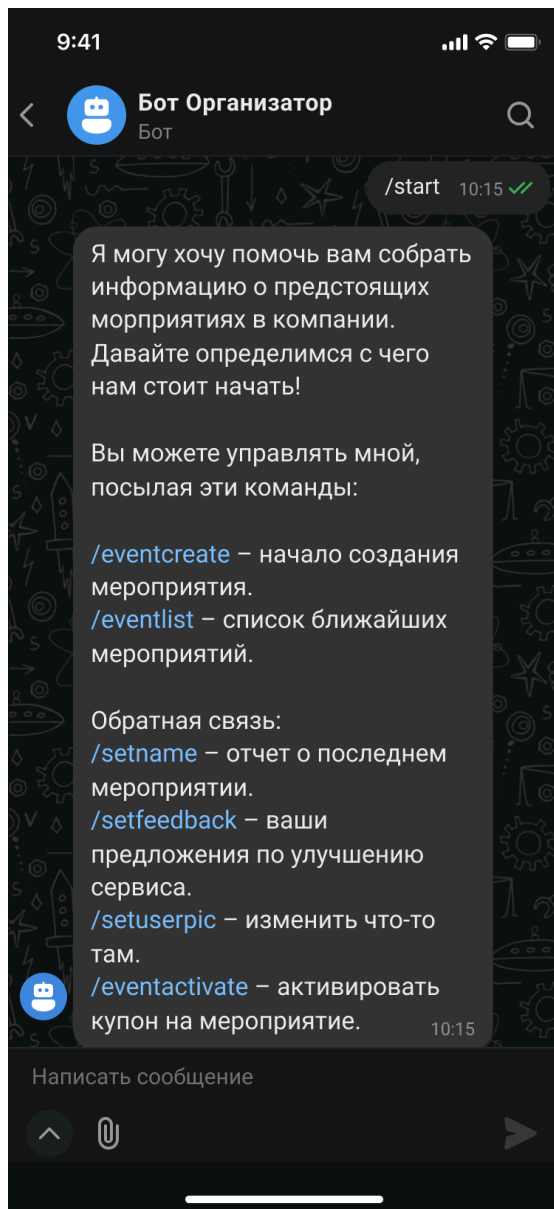


Пример отображения кнопки “Начать чат”



## Команды

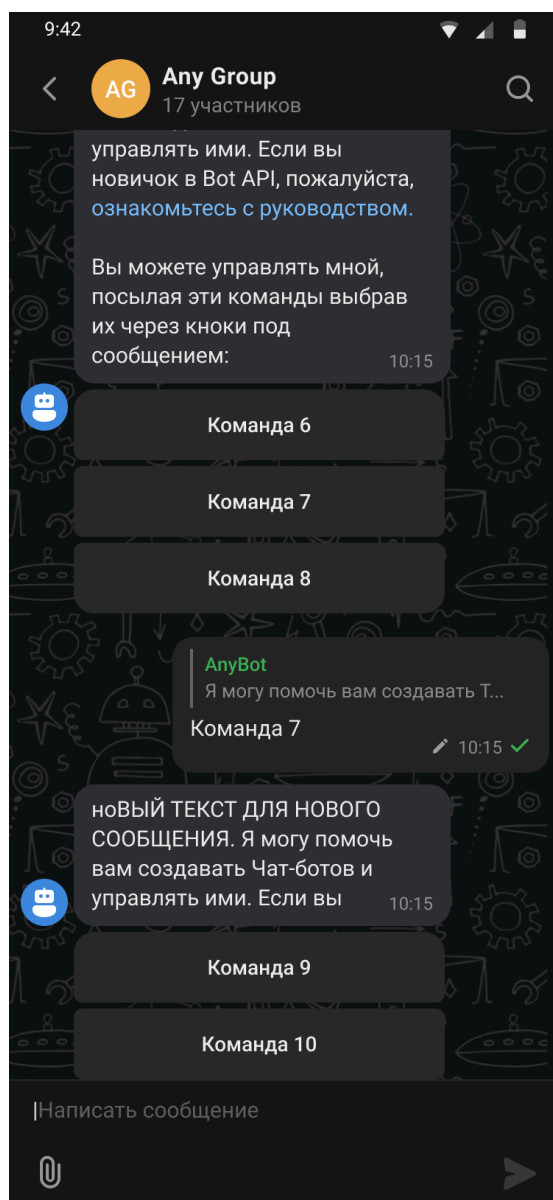
В сообщениях от Чат-бота могут быть заложены команды. Команды выделяются внешне голубым цветом и начинаются с символа “/”, они доступны для нажатия. При нажатии на команду, в чат от лица пользователя отправится сообщение с этой командой (если это групповой чат, то в сообщении добавится упоминание Чат-бота).



Пример взаимодействия с Чат-ботом с помощью команд

## Кнопки

Общение с Чат-ботом может выстраиваться путем нажатия кнопок под его сообщением. После нажатия на кнопку, от лица пользователя будет отправлено сообщение с текстом, заложенным разработчиком (в групповом чате будет в виде ответа на сообщение Чат-бота). Максимальное допустимое кол-во кнопок под сообщением – 20. Сообщения с кнопками НЕ поддерживаются в информационных каналах.



Пример взаимодействия с Чат-ботом с помощью кнопок

## 1.4 Возможности и ограничения

В приложении Чат-боты имеют права и возможности схожие с возможностями обычных пользователей, за исключением отдельных ограничений.

### Возможности работы с Чат-ботами:

- добавление в контакты;
- добавление как участников в пространства, группы и каналы;
- назначение администратором пространств, групп и каналов;
- упоминание в чатах;
- пересылка карточки контакта Чат-бота;
- пересылка Чат-боту и от него сообщений;
- осуществление поиска Чат-ботов.

### Ограничения работы с Чат-ботами:

- невозможность передачи прав на владение группами и каналами;
- отсутствие прав на создание групп, каналов и пространств;
- невозможность создания секретного чата.

## 2 Общее описание функциональности API

Функционал API представляет собой методы взаимодействия между системами посредством прикладного программного интерфейса по протоколу HTTPs. API позволяет облегчить использование в приложении данных и ресурсов других приложений и серверов, предоставляет сторонний доступ к ресурсам Мессенджера, поддерживает используемые стандарты для отправки HTTP-запросов к API, с выделением каждого URL в объекты.

**Функционал API поддерживает следующие типы HTTP-методов:**

- GET для запроса на предоставление данных, включающий параметры запроса для фильтрации результатов, полученных от API;
- POST для отправки информации к API с выделением в базе данных созданного ресурса.


**Функционал API поддерживает способы передачи параметров в запросах:**

- строка запросов URL;
- application/json (кроме загрузки файлов);
- multipart/form-data (используется для загрузки файлов).

Функционал API позволяет создать и настроить Чат-ботов, с возможностью присвоения уникального токена <token> в Мессенджере.

**Дополнительные запросы методов в формате JSON:**

- Метод возвращает описание методов API в формате JSON;
- Метод инициализирует исполнение прикладных методов.



Для работы с Bot API необходим специальный Токен, который присваивается каждому Чат-боту и действует определенное время. Для получения Токена необходимо обратиться к специально уполномоченному лицу организации, которое обладает правами администрирования Чат-ботов.

### 3 Опциональное шифрование

При работе с API необходимо учитывать, что протокол обмена данными с пользователем дополнительно шифруется с помощью алгоритма ГОСТ Р 34.12-2015 «Кузнечик».

Для внешней интеграции Чат-бота с API необходимо получить от Администратора Бэкофиса, который создавал Токен Чат-бота, уникальный симметричный ключ шифрования для алгоритма «Кузнечик».

После получения ключа нужно найти библиотеку, реализующую алгоритм «Кузнечик» на соответствующем языке программирования, на котором пишется логика самого Чат-бота. С помощью данной библиотеки осуществлять шифрование запросов на REST перед отправкой и расшифровывать ответы от REST после получения.

Примеры реализаций (библиотек) алгоритма «Кузнечик»:

Java – org.bouncycastle:bcprov-jdk15on;

Go – kuznec.go;

Python – PyGOST; gostcrypto.

Для внутренней интеграции с API есть возможность отключения шифрования алгоритмом «Кузнечик» взаимодействия бота с rest API. Включается шифрование для определенного Чат-бота и его токена через Бэкофис, где в результате генерируется уникальный секретный сессионный симметричный ключ, который автоматически прикрепляется на сервере к данному токenu Чат-бота. Впоследствии Чат-бот может вызывать rest API, шифруя запросы и расшифровывая ответы с помощью данного ключа. При таком режиме, ключ должен держаться в секрете, а сам токен становится безопасно показывать в хедере (он передается в хедере **Authorization**).

Например, для запроса на `/botapi/v1/messages/confirm/{workspaceId}/{groupId}` тело запроса при отключенном шифровании будет выглядеть следующим образом:

```
{
  "lastMessageId": 1188049
}
```

При включенном шифровании алгоритмом «Кузнечик» тело запроса будет представлено в виде base64 строки (**обернуто в json-конверт с “content”**):

```
{"content": "6Vs1HlrF1/FV/i37XIEv2SEUUDSbkRfFNVEbMU6J3rni6UaJV2Odi7/9V9atIRsFcfu65yiIRxu+qqGtq+RvKg=="}
```

## 4 SSE endpoints

Специально для Чат-бота выставляются **SSE HTTP endpoints**

- `/api/v2/events/bot` – Чат-бот может получать и обрабатывать sse события со следующими параметрами для подключения:
  - HTTP header **Authorization**, в который нужно передавать **authToken** Чат-бота;
  - HTTP header **Workspacelds**, в который нужно передать через запятую список ID пространств, на события которых Чат-бот подписывается. На публичное пространство (ID которого -1) подписка обязательна.
- `/api/v2/events/bot/group` – для получения событий по sse из виртуальных групп для комментариев в каналах (на сообщения администратора канала Чат-бот не отвечает), со следующими параметрами для подключения:
  - HTTP header **Authorization**, в который нужно передавать **authToken** Чат-бота;
  - HTTP header **groupid**, в который нужно передавать ID группы;
  - HTTP header **workspaceld**, в который нужно передать ID пространства, на события которого Чат-бот подписывается.

## События

Поддерживаются следующие SSE события:

- MESSAGE – событие на сообщение для Чат-бота;
- MESSAGE\_DELETE – когда происходит удаление определенного сообщения;



- `MESSAGES_DELETED` – удаление нескольких сообщений пользователем;
- `MESSAGES_DELETED_COMPLETELY` – автоматическое удаление сообщений в чатах;
- `MESSAGE_UPDATE` – при обновлении определенного сообщения;
- `PING` – утилитарное событие, не несет полезной нагрузки;
- `GROUP_USER_JOIN` – событие присоединения пользователя к группе;
- `GROUP_USERS_ADD` – событие добавления пользователей к группе;
- `GROUP_USER_LEAVE` – событие когда пользователь покидает группу;
- `GROUP_USER_REMOVE` – событие удаления пользователя из группы;
- `GROUP_ME_ADD` – событие добавления Чат-бота в группу;
- `GROUP_ME_REMOVE` – событие удаления Чат-бота из группы;
- `GROUP_HISTORY_CLEAR` – событие очистки истории для всех;
- `GROUP_DELETE` – событие удаления группы;
- `USER_SIGNED_UP_TO_WORKSPACE` – событие добавления пользователей в workspace;
- `USER_UPDATE` – событие обновление профиля пользователя.

## 5 Rest endpoints

Чат-бот, как определенный тип пользователя мессенджера, для своего функционирования может использовать определенный список REST методов.

Каждый метод нужно вызывать со следующим обязательным HTTP-хедером:

- Authorization: authToken – для аутентификации и авторизации бота, где authToken – его персональный токен.

При отправке запросов с включенным шифрованием следует учитывать, что тело запроса обертывается в контейнер **content**.

## 5.1 Подтверждение о прочитанных и обработанных сообщениях/командах

Цель Чат-бота – выполнение поступающих, в том числе и от конкретных пользователей, команд и сообщений. Этот процессинг выполняется в заданном порядке для фиксации, контроля, управления, аудита работы Чат-бота необходим механизм подтверждения выполненных команд и событий.

- Для подтверждения выполненных команд используется REST метод POST /botapi/v1/messages/confirm/{workspaceId}/{groupId}, который необходимо вызывать с пейлоадом запроса json вида:

```
{
  "lastMessageId": 1188049
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
lastMessageId	Long	ID сообщения, которое было обработано ботом на данный момент

В ответ приходит статус-код 200.

- Для подтверждения событий, таких как удаление, редактирование сообщения и т.д. используется REST метод POST/botapi/v1/groups/confirmEvents/{workspaceId}/{groupId}, который необходимо вызывать с пейлоадом запроса json вида:

```
{
  "events": [
    {
      "lastReceivedEventId": 2001359
    }
  ]
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
events	JSON-объект	Содержимое объекта events
lastReceivedEventId	Long	ID события

В ответ приходит статус-код 200.

## 5.2 Отправка сообщений

Чат-бот может общаться с клиентами посредством отправки текстовых и файловых сообщений в рамках определенной группы мессенджера. Например, это может быть р2р группа с каким-либо пользователем, либо это может быть общий чат/канал на несколько пользователей. Чат-бот принимает команды-сообщения от пользователей данной группы, обрабатывает их, и может отсылать результат в виде ответных сообщений, например, в эту группу, и не только.

### 5.2.1 Отправка текстовых сообщений

Для отправки текстовых сообщений Чат-ботом в мессенджер используется следующий REST метод:

- POST метод  
`/botapi/v1/messages/sendTextMessage/{workspaceId}/{groupId}` – для текстовых сообщений. Пейлоад запроса (json):

```
{
  "clientRandomId": 195482658,
  "message": "Текст сообщения от бота с результатом обработки команды",
  "replyMsgId": 345,
  "hidden": true,
  "entities": [
    {
      "mention": {
        "offset": 0,
        "length": 0,
        "userId": 0
      },
      "url": {
        "offset": 0,
        "length": 0
      },
      "email": {
        "offset": 0,
        "length": 0
      }
    }
  ]
}
```

```

    },
    "bold": {
      "offset": 0,
      "length": 0
    },
    "italic": {
      "offset": 0,
      "length": 0
    },
    },
    "phone": {
      "offset": 0,
      "length": 0
    },
    },
    "underline": {
      "offset": 0,
      "length": 0
    },
    },
    "textUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/",
      "disablePreview": false
    },
    "textFormattedUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/"
    },
    },
    "strike": {
      "offset": 0,
      "length": 0
    },
    },
    "monospace": {
      "offset": 0,
      "length": 0
    },
    },
    "channel": {
      "offset": 0,
      "length": 0,
      "channelId": 0
    },
    },
    "botCommand": {
      "offset": 0,
      "length": 0
    }
  }
},
],

```

```

"buttonsMarkup": {
  "inlineMessageButtons": [
    {
      "id": 0,
      "label": "Send",
      "callbackMessage": "callbackMessage_5b5a6683d638",
      "callbackData": "id = 1235"
    }
  ]
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
replyMsgId	Long	ID сообщения, на которое отвечает бот (опциональный параметр)
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; userId: ID пользователя.

url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длинна.
italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длинна.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка; disablePreview: отображать ли превью ссылки.



textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
monospace	JSON-объект	Положение текста с моноширифтом в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально): id: идентификатор кнопки (обязательное поле);

		<p>label: текст на кнопке (обязательное поле);</p> <p>callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).</p>
--	--	---

В ответ получаем:

```
{
  "messageId": 0
}
```

Описание объектов ответа:

Поле	Тип	Содержание
messageId	Long	ID сообщения

## 5.2.2 Отправка файловых сообщений

Для отправки файловых сообщений Чат-ботом в мессенджер используется следующий REST метод:

- POST метод /botapi/v1/messages/sendFile/{workspaceId}/{groupId} – для сообщений, содержащих данные о файле. Пейлоад запроса (json):

```
{
  "clientRandomId": 123,
  "file": {
    "fileName": "file_name.pdf",
    "length": 594359,
    "mimeType": "application/pdf",
    "resourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.pdf"
    }
  },
  "message": "Текст сообщения от бота с результатом обработки команды",
  "replyMsgId": 345,
  "hidden": true
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
clientRandomId	Long	Уникальный идентификатор на основе временных меток
file	JSON-объект	Содержимое объекта file
fileName	String	Имя файла
length	Long	Размер файла в байтах

Поле	Тип	Содержание
mimeType	String	Тип медиа
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
message	String	Текст сообщения
replyMsgId	Long	ID сообщения, на которое отвечает бот (опциональный параметр)
hidden	Boolean	При значении true сообщение будет скрыто под спойлером

В ответ получаем:

```
{
  "messageId": 0
}
```

Описание объектов ответа:

Поле	Тип	Содержание
messageId	Long	ID сообщения

### 5.2.3 Отправка текстовых сообщений, содержащих изображение

Для отправки текстовых сообщений, содержащих изображение, Чат-ботом в мессенджер используется следующий REST метод:

- POST метод `/botapi/v1/messages/sendImage/{workspaceId}/{groupId}` – для текстовых сообщений, содержащих изображение. Пейлоад запроса (json):

```
{
  "image": {
    "fileName": "fileName.jpg",
    "width": 800,
    "height": 800,
    "length": 594359,
    "resourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "mimeType": "image/jpeg"
  },
  "replyMsgId": 345,
  "hidden": true,
  "clientRandomId": 195482658,
  "message": "Текст сообщения от бота с результатом обработки команды",
  "entities": [
    {
      "mention": {
        "offset": 0,
        "length": 0,
        "userId": 0
      },
      "url": {
        "offset": 0,
        "length": 0
      },
      "email": {
        "offset": 0,
        "length": 0
      },
      "bold": {
```

```

        "offset": 0,
        "length": 0
    "italic": {
        "offset": 0,
        "length": 0
    },
    "phone": {
        "offset": 0,
        "length": 0
    },
    "underline": {
        "offset": 0,
        "length": 0
    },
    "textUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/",
        "disablePreview": false
    },
    "textFormattedUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/"
    },
    "strike": {
        "offset": 0,
        "length": 0
    },
    "monospace": {
        "offset": 0,
        "length": 0
    },
    channel": {
        "offset": 0,
        "length": 0,
        "channelId": 0
    },
    "botCommand": {
        "offset": 0,
        "length": 0
    }
}
],
"buttonsMarkup": {
    "inlineMessageButtons": [

```

```

    {
      "id": 0,
      "label": "Send",
      "callbackMessage": "callbackMessage_5b5a6683d638",
      "callbackData": "id = 1235"
    }
  ]
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
image	JSON-объект	Содержимое объекта image
fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа

replyMsgId	Long	ID сообщения, на которое отвечает бот (опциональный параметр)
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длина; userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.



italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длинна.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка; disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
monospace	JSON-объект	Положение текста с моноширифтом в тексте сообщения (опциональный параметр):

		offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально): id: идентификатор кнопки (обязательное поле); label: текст на кнопке (обязательное поле); callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки); callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).

В ответ получаем:

```
{  
  "messageId": 1002306  
}
```

Описание объектов ответа:

Поле	Тип	Содержание
messageId	Long	ID сообщения

## 5.2.4 Отправка текстовых сообщений, содержащих видео

Для отправки текстовых сообщений, содержащих данные о видео, Чат-ботом в мессенджер используется следующий REST метод:

- POST метод `/botapi/v1/messages/sendVideo/{workspaceId}/{groupId}` – для текстовых сообщений, содержащих данные о видео. Пейлоад запроса (json):

```
{
  "video": {
    "fileName": "fileName.mp4",
    "width": 800,
    "height": 800,
    "length": 594359,
    "thumbnailResourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "duration": 0,
    "resourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.mp4"
    },
    "mimeType": "image/mp4"
  },
  "replyMsgId": 345,
  "hidden": true,
  "clientRandomId": 195482658,
  "message": "Текст сообщения от бота с результатом обработки команды",
  "entities": [
    {
      "mention": {
        "offset": 0,
        "length": 0,
        "userId": 0
      },
      "url": {
        "offset": 0,
        "length": 0
      }
    }
  ]
}
```

```
    },
    "email": {
      "offset": 0,
      "length": 0
    },
    "bold": {
      "offset": 0,
      "length": 0
    },
    "italic": {
      "offset": 0,
      "length": 0
    },
    "phone": {
      "offset": 0,
      "length": 0
    },
    "underline": {
      "offset": 0,
      "length": 0
    },
    "textUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/",
      "disablePreview": false
    },
    "textFormattedUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/"
    },
    "strike": {
      "offset": 0,
      "length": 0
    },
    "monospace": {
      "offset": 0,
      "length": 0
    },
    "channel": {
      "offset": 0,
      "length": 0,
      "channelId": 0
    },
    "botCommand": {
      "offset": 0,
```

```

        "length": 0
    }
}
],
"buttonsMarkup": {
    "inlineMessageButtons": [
        {
            "id": 0,
            "label": "Send",
            "callbackMessage": "callbackMessage_5b5a6683d638",
            "callbackData": "id = 1235"
        }
    ]
}
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
video	JSON-объект	Содержимое объекта video
fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
thumbnailResourceRef	JSON-объект	Содержимое объекта thumbnailResourceRef. Изображение, которое отображается в сообщении для видео (опциональный параметр).
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования

url	String	Url, где лежит зашифрованный файл
duration	Long	Продолжительность видео (опциональный параметр)
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа
replyMsgId	Long	ID сообщения, на которое отвечает бот (опциональный параметр)
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длинна;

		userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.
italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длина.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка;



		disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
monospace	JSON-объект	Положение текста с моноширифтом в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально):

		<p>id: идентификатор кнопки (обязательное поле);</p> <p>label: текст на кнопке (обязательное поле);</p> <p>callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).</p>
--	--	--

В ответ получаем:

```
{
  "messageId": 1002306
}
```

Описание объектов ответа:

Поле	Тип	Содержание
messageId	Long	ID сообщения

## 5.2.5 Отправка текстовых сообщений, содержащих изображение и видео

Для отправки текстовых сообщений, содержащих изображение и видео, Чат-ботом в мессенджер используется следующий REST метод:

- POST метод /botapi/v1/messages/sendMultimedia/{workspaceId}/{groupId} – для текстовых сообщений, содержащих изображение и видео.

Пейлоад запроса (json):

```
{
  "multimedia": [
    {
      "image": {
        "fileName": "fileName.jpg",
        "width": 800,
        "height": 800,
        "length": 594359,
        "resourceRef": {
          "id": "80d30307-7122-4fbc-94c6-fa3691276453",
          "key": "XQh5NX=",
          "transformation": "GOST3412-2015/ECB/PKCS7Padding",
          "url": "https://s3.kchat.app/fileupload/5f136.jpg"
        },
        "mimeType": "image/jpeg"
      },
      "video": {
        "fileName": "fileName.mp4",
        "width": 800,
        "height": 800,
        "length": 594359,
        "thumbnailResourceRef": {
          "id": "80d30307-7122-4fbc-94c6-fa3691276453",
          "key": "XQh5NX=",
          "transformation": "GOST3412-2015/ECB/PKCS7Padding",
          "url": "https://s3.kchat.app/fileupload/5f136.jpg"
        },
        "duration": 0,
        "resourceRef": {
          "id": "80d30307-7122-4fbc-94c6-fa3691276453",
          "key": "XQh5NX=",
          "transformation": "GOST3412-2015/ECB/PKCS7Padding",
          "url": "https://s3.kchat.app/fileupload/5f136.mp4"
        },
        "mimeType": "image/mp4"
      }
    }
  ]
}
```

```

    }
  }
],
"replyMsgId": 345,
"hidden": true,
"clientRandomId": 195482658,
"message": "Текст сообщения от бота с результатом обработки
команды",
"entities": [
  {
    "mention": {
      "offset": 0,
      "length": 0,
      "userId": 0
    },
    "url": {
      "offset": 0,
      "length": 0
    },
    "email": {
      "offset": 0,
      "length": 0
    },
    "bold": {
      "offset": 0,
      "length": 0
    },
    "italic": {
      "offset": 0,
      "length": 0
    },
    "phone": {
      "offset": 0,
      "length": 0
    },
    "underline": {
      "offset": 0,
      "length": 0
    },
    "textUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/",
      "disablePreview": false
    },
    "textFormattedUrl": {
      "offset": 0,

```

```

        "length": 0,
        "url": "https://mail.google.com/"
    },
    "strike": {
        "offset": 0,
        "length": 0
    },
    "monospace": {
        "offset": 0,
        "length": 0
    },
    "channel": {
        "offset": 0,
        "length": 0,
        "channelId": 0
    },
    "botCommand": {
        "offset": 0,
        "length": 0
    }
},
],
"buttonsMarkup": {
    "inlineMessageButtons": [
        {
            "id": 0,
            "label": "Send",
            "callbackMessage": "callbackMessage_5b5a6683d638",
            "callbackData": "id = 1235"
        }
    ]
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
multimedia	Array of JSON-objects	Информация о мультимедиа
image	JSON-объект	Содержимое объекта image

fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа
video	JSON-объект	Содержимое объекта video
fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
thumbnailResourceRef	JSON-объект	Содержимое объекта thumbnailResourceRef. Изображение, которое отображается в сообщении для видео (опциональный параметр).
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования

url	String	Url, где лежит зашифрованный файл
duration	Long	Продолжительность видео (опциональный параметр)
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа
replyMsgId	Long	ID сообщения, на которое отвечает бот (опциональный параметр)
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длинна;

		userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.
italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длина.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка;



		disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
monospace	JSON-объект	Положение текста с моношрифтом в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально):

		<p>id: идентификатор кнопки (обязательное поле);</p> <p>label: текст на кнопке (обязательное поле);</p> <p>callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).</p>
--	--	--

В ответ получаем:

```
{
  "messageId": 1002306
}
```

Описание объектов ответа:

Поле	Тип	Содержание
messageId	Long	ID сообщения

## 5.3 Обновление сообщений

Чат-бот может обновлять текстовые и файловые сообщения в рамках определенной группы мессенджера.

### 5.3.1 Обновление текстового сообщения

Для обновления текстового сообщения используется REST метод POST:

- `/botapi/v1/messages/updateTextMessage/{workspaceId}/{groupId}/{messageId}`

Пейлоад запроса (json):

```
{  
  
  "message": "Текст сообщения",  
  "hidden": true,  
  "entities": [  
    {  
      "mention": {  
        "offset": 0,  
        "length": 0,  
        "userId": 0  
      },  
      "url": {  
        "offset": 0,  
        "length": 0  
      },  
      "email": {  
        "offset": 0,  
        "length": 0  
      },  
      "bold": {  
        "offset": 0,  
        "length": 0  
      },  
      "italic": {  
        "offset": 0,  
        "length": 0  
      },  
      "phone": {  
        "offset": 0,  
        "length": 0  
      }  
    }  
  ]  
}
```

```

    },
    "underline": {
      "offset": 0,
      "length": 0
    },
    "textUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/",
      "disablePreview": false
    },
    "textFormattedUrl": {
      "offset": 0,
      "length": 0,
      "url": "https://mail.google.com/"
    },
    "strike": {
      "offset": 0,
      "length": 0
    },
    "monospace": {
      "offset": 0,
      "length": 0
    },
    "channel": {
      "offset": 0,
      "length": 0,
      "channelId": 0
    },
    "botCommand": {
      "offset": 0,
      "length": 0
    }
  },
  "buttonsMarkup": {
    "inlineMessageButtons": [
      {
        "id": 0,
        "label": "Send",
        "callbackMessage": "callbackMessage_5b5a6683d638",
        "callbackData": "id = 1235"
      }
    ]
  }
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageId (path переменная)	Long	ID сообщения
message	String	Текст сообщения
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.

bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.
italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длина.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка; disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение;

		length: длина.
monospace	JSON-объект	Положение текста с моношрифтом в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально): id: идентификатор кнопки (обязательное поле); label: текст на кнопке (обязательное поле); callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки); callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля

		(опционально, не отображается пользователю).
--	--	--

В ответ объект обновленного сообщения.

```
{
  "groupId": 2214212496960365,
  "id": 1186920,
  "date": 1685647540245,
  "clientRandomId": 175157013,
  "senderId": 2222007305300254,
  "links": [],
  "message": "",
  "entities": [],
  "multimedia": [],
  "position": 2613,
  "deleted": false,
  "reactions": [],
  "stringId": "1186920",
  "senderType": "DEFAULT",
  "reactionGroups": []
}
```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
id	Long	ID сообщения
date	Long	Дата
clientRandomId	Long	Уникальный ID с учетом временных меток
senderId	Long	ID отправителя
links	Array	Ссылки в сообщении
message	String	Текст сообщения



Поле	Тип	Содержание
entities	Array of JSON-objects	Дополнительные объекты в сообщении
multimedia	Array	Информация о мультимедиа
position	Long	Позиция
deleted	Boolean	Статус удаления
reactions	Array	Информация о реакциях
stringId	Long	ID строки
senderType	Enum	Default или Group
reactionGroups	Array	Информация о сгруппированных реакциях

### 5.3.2 Обновление файлового сообщения

Для обновления файлового сообщения используется REST метод POST:

- /botapi/v1/messages/updateFile/{workspaceId}/{groupId}/{messageId}

Тело запроса:

```
{
  "file": {
    "fileName": "file_name.pdf",
    "length": 594359,
    "resourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "url": "https://s3.kchat.app/fileupload/5f136.pdf",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding"
    },
    "mimeType": "application/pdf"
  },
  "message": "message_7d6d7988067d",
  "hidden": true,
  "entities": [
    {
      "mention": {
        "offset": 0,
        "length": 0,
        "userId": 0
      },
      "url": {
        "offset": 0,
        "length": 0
      },
      "email": {
        "offset": 0,
        "length": 0
      },
      "bold": {
        "offset": 0,
        "length": 0
      },
      "italic": {
        "offset": 0,
        "length": 0
      },
      "code": {
```

```

        "offset": 0,
        "length": 0,
        "language": "language_a5c21d78e4ba"
    },
    "phone": {
        "offset": 0,
        "length": 0
    },
    "underline": {
        "offset": 0,
        "length": 0
    },
    "textUrl": {
        "offset": 0,
        "length": 0,
        "url": "url_f19e1aa9b6e5",
        "disablePreview": false
    },
    "textFormattedUrl": {
        "offset": 0,
        "length": 0,
        "url": "url_cc7aae829954"
    },
    "strike": {
        "offset": 0,
        "length": 0
    },
    "monospace": {
        "offset": 0,
        "length": 0
    },
    "hashtag": {
        "offset": 0,
        "length": 0
    },
    "channel": {
        "offset": 0,
        "length": 0,
        "channelId": 0
    },
    "botCommand": {
        "offset": 0,
        "length": 0
    }
}
],

```

```

"buttonsMarkup": {
  "inlineMessageButtons": [
    {
      "id": 0,
      "label": "label_edc9ba8060e5",
      "callbackMessage": "callbackMessage_57c1f2c85f11",
      "callbackData": "callbackData_492e1b8e711d"
    }
  ]
}
}

```

Описание объектов запроса:

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageId (path переменная)	Long	ID сообщения
file	JSON-объект	Содержимое объекта file
fileName	String	Имя файла
length	Long	Размер изображения в байтах
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл

contentType	String	Тип медиа
replyMsgId	Long	ID сообщения, на которое отвечаем
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длина; userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.

italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длинна.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка; disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
monospace	JSON-объект	Положение текста с моноширифтом в тексте сообщения (опциональный параметр):

		offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально): id: идентификатор кнопки (обязательное поле); label: текст на кнопке (обязательное поле); callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки); callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).

В ответ получаем:

```
{
  "groupId": 2214212496960365,
  "id": 1186920,
  "date": 1685647540245,
  "clientRandomId": 175157013,
  "senderId": 2222007305300254,
  "links": [],
  "message": "",
  "entities": [],
  "media": {
    "file": {
      "fileName": "Новая версия.pdf",
      "length": 8371515,
      "resourceRef": {
        "id": "5521bca0-d94f-41a9-bc95-485ff84cf8d0",
        "url": "https://s3.kchat.app/fileupload/1c9cEUpUV9%2F21060d79-e423-461f-833b-1d215f646855.pdf",
        "key": "0PYeGj/lWQGFQZw7IBywHa1FKj1MPu4CnxmSJhiX2jE=",
        "transformation": "GOST3412-2015/ECB/PKCS7Padding"
      },
      "mimeType": "application/pdf"
    }
  },
  "multimedia": [],
  "position": 2613,
  "deleted": false,
  "reactions": [],
  "stringId": "1186920",
  "senderType": "DEFAULT",
  "reactionGroups": []
}
```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
id	Long	ID сообщения
date	Long	Дата



Поле	Тип	Содержание
clientRandomId	Long	Уникальный ID с учетом временных меток
senderId	Long	ID отправителя
links	Array	Ссылки в сообщении
message	String	Текст сообщения
entities	Array of JSON-objects	Дополнительные объекты в сообщении
media	JSON-объект	Содержимое объекта media
file	JSON-объект	Содержимое объекта file
fileName	String	Имя файла
length	Long	Размер файла в байтах
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
url	String	Url, где лежит зашифрованный файл
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
mimeType	String	Тип медиа
multimedia	Array	Информация о мультимедиа
position	Long	Позиция
deleted	Boolean	Статус удаления
reactions	Array	Информация о реакциях
stringId	Long	ID строки

Поле	Тип	Содержание
senderType	Enum	Default или Group
reactionGroups	Array	Информация о сгруппированных реакциях

### 5.3.3 Обновление сообщения, содержащего изображение

Для отправки текстовых сообщений, содержащих изображение, Чат-ботом в мессенджер используется следующий REST метод:

- POST метод

/botapi/v1/messages/updateImage/{workspaceId}/{groupId}/{messageId} – для текстовых сообщений, содержащих изображение. Пейлоад запроса (json):

```
{
  "image": {
    "fileName": "fileName.jpg",
    "width": 800,
    "height": 800,
    "length": 594359,
    "resourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "mimeType": "image/jpeg"
  },
  "replyMsgId": 345,
  "clientRandomId": 195482658,
  "message": "Текст сообщения от бота с результатом обработки команды",
  "hidden": true,
  "entities": [
    {
      "mention": {
        "offset": 0,
        "length": 0,
        "userId": 0
      },
      "url": {
        "offset": 0,
        "length": 0
      },
      "email": {
        "offset": 0,
        "length": 0
      },
      "bold": {
```

```

        "offset": 0,
        "length": 0
    "italic": {
        "offset": 0,
        "length": 0
    },
    "phone": {
        "offset": 0,
        "length": 0
    },
    "underline": {
        "offset": 0,
        "length": 0
    },
    "textUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/",
        "disablePreview": false
    },
    "textFormattedUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/"
    },
    "strike": {
        "offset": 0,
        "length": 0
    },
    "monospace": {
        "offset": 0,
        "length": 0
    },
    channel": {
        "offset": 0,
        "length": 0,
        "channelId": 0
    },
    "botCommand": {
        "offset": 0,
        "length": 0
    }
}
],
"buttonsMarkup": {
    "inlineMessageButtons": [

```

```

    {
      "id": 0,
      "label": "Send",
      "callbackMessage": "callbackMessage_5b5a6683d638",
      "callbackData": "id = 1235"
    }
  ]
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageId (path переменная)	Long	ID сообщения
image	JSON-объект	Содержимое объекта image
fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл

contentType	String	Тип медиа
replyMsgId	Long	ID сообщения, на которое отвечаем
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение меншена в тексте сообщения (опциональный параметр): offset: смещение; length: длина; userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.

italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длинна.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка; disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длинна; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длинна.
monospace	JSON-объект	Положение текста с моноширифтом в тексте сообщения (опциональный параметр):

		offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.
buttonsMarkup	JSON-объект	Разметка кнопок в сообщении (опциональный параметр): inlineMessageButtons: Array (кнопки в сообщении, опционально): id: идентификатор кнопки (обязательное поле); label: текст на кнопке (обязательное поле); callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки); callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).

В ответ получаем объект обновленного сообщения.



### 5.3.4 Обновление сообщения, содержащего видео

Для отправки текстовых сообщений, содержащих данные о видео, Чат-ботом в мессенджер используется следующий REST метод:

- POST метод

/botapi/v1/messages/updateVideo/{workspaceId}/{groupId}/{messageId}

– для текстовых сообщений, содержащих данные о видео. Пейлоад запроса (json):

```
{
  "video": {
    "fileName": "fileName.mp4",
    "width": 800,
    "height": 800,
    "length": 594359,
    "thumbnailResourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "duration": 0,
    "resourceRef": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.mp4"
    },
    "mimeType": "image/mp4"
  },
  "replyMsgId": 345,
  "clientRandomId": 195482658,
  "message": "Текст сообщения от бота с результатом обработки команды",
  "hidden": true,
  "entities": [
    {
      "mention": {
        "offset": 0,
        "length": 0,
        "userId": 0
      },
      "url": {
        "offset": 0,
```

```
        "length": 0
    },
    "email": {
        "offset": 0,
        "length": 0
    },
    "bold": {
        "offset": 0,
        "length": 0
    },
    "italic": {
        "offset": 0,
        "length": 0
    },
    "phone": {
        "offset": 0,
        "length": 0
    },
    "underline": {
        "offset": 0,
        "length": 0
    },
    "textUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/",
        "disablePreview": false
    },
    "textFormattedUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/"
    },
    "strike": {
        "offset": 0,
        "length": 0
    },
    "monospace": {
        "offset": 0,
        "length": 0
    },
    "channel": {
        "offset": 0,
        "length": 0,
        "channelId": 0
    },
    "botCommand": {
```

```

        "offset": 0,
        "length": 0
    }
}
],
"buttonsMarkup": {
    "inlineMessageButtons": [
        {
            "id": 0,
            "label": "Send",
            "callbackMessage": "callbackMessage_5b5a6683d638",
            "callbackData": "id = 1235"
        }
    ]
}
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageId (path переменная)	Long	ID сообщения
video	JSON-объект	Содержимое объекта video
fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
thumbnailResourceRef	JSON-объект	Содержимое объекта thumbnailResourceRef. Изображение, которое отображается в сообщении для видео (опциональный параметр).
id	UUID	UUID файла

key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
duration	Long	Продолжительность видео (опциональный параметр)
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа
replyMsgId	Long	ID сообщения, на которое отвечаем
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание только для одного из указанных полей (опциональный параметр)

mention	JSON-объект	Положение mentions в тексте сообщения (опциональный параметр): offset: смещение; length: длина; userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.
italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длина.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.

textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка; disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
monospace	JSON-объект	Положение текста с моношрифтом в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.

buttonsMarkup	JSON-объект	<p>Разметка кнопок в сообщении (опциональный параметр):</p> <p>inlineMessageButtons: Array (кнопки в сообщении, опционально):</p> <p>id: идентификатор кнопки (обязательное поле);</p> <p>label: текст на кнопке (обязательное поле);</p> <p>callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).</p>
---------------	-------------	--

В ответ получаем объект обновленного сообщения.

### 5.3.5 Обновление сообщения, содержащего изображение и видео

Для отправки текстовых сообщений, содержащих данные об изображении и видео, Чат-ботом в мессенджер используется следующий REST метод:

- POST метод

/botapi/v1/messages/updateMultimedia/{workspaceId}/{groupId}/{messageId}

– для текстовых сообщений, содержащих данные об изображении и видео. Пейлоад запроса (json):

```
{
  "multimedia": [
    {
      "image": {
        "fileName": "fileName.jpg",
        "width": 800,
        "height": 800,
        "length": 594359,
        "resourceRef": {
          "id": "80d30307-7122-4fbc-94c6-fa3691276453",
          "key": "XQh5NX=",
          "transformation": "GOST3412-2015/ECB/PKCS7Padding",
          "url": "https://s3.kchat.app/fileupload/5f136.jpg"
        },
        "mimeType": "image/jpeg"
      },
      "video": {
        "fileName": "fileName.mp4",
        "width": 800,
        "height": 800,
        "length": 594359,
        "thumbnailResourceRef": {
          "id": "80d30307-7122-4fbc-94c6-fa3691276453",
          "key": "XQh5NX=",
          "transformation": "GOST3412-2015/ECB/PKCS7Padding",
          "url": "https://s3.kchat.app/fileupload/5f136.jpg"
        },
        "duration": 0,
        "resourceRef": {
          "id": "80d30307-7122-4fbc-94c6-fa3691276453",
          "key": "XQh5NX=",

```



```

        "transformation": "GOST3412-2015/ECB/PKCS7Padding",
        "url": "https://s3.kchat.app/fileupload/5f136.mp4"
    },
    "mimeType": "image/mp4"
}
],
"replyMsgId": 345,
"clientRandomId": 195482658,
"message": "Текст сообщения от бота с результатом обработки
команды",
"hidden": true,
"entities": [
    {
        "mention": {
            "offset": 0,
            "length": 0,
            "userId": 0
        },
        "url": {
            "offset": 0,
            "length": 0
        },
        "email": {
            "offset": 0,
            "length": 0
        },
        "bold": {
            "offset": 0,
            "length": 0
        },
        "italic": {
            "offset": 0,
            "length": 0
        },
        "phone": {
            "offset": 0,
            "length": 0
        },
        "underline": {
            "offset": 0,
            "length": 0
        },
        "textUrl": {
            "offset": 0,
            "length": 0,
            "url": "https://mail.google.com/",

```

```

        "disablePreview": false
    },
    "textFormattedUrl": {
        "offset": 0,
        "length": 0,
        "url": "https://mail.google.com/"
    },
    "strike": {
        "offset": 0,
        "length": 0
    },
    "monospace": {
        "offset": 0,
        "length": 0
    },
    "channel": {
        "offset": 0,
        "length": 0,
        "channelId": 0
    },
    "botCommand": {
        "offset": 0,
        "length": 0
    }
},
],
"buttonsMarkup": {
    "inlineMessageButtons": [
        {
            "id": 0,
            "label": "Send",
            "callbackMessage": "callbackMessage_5b5a6683d638",
            "callbackData": "id = 1235"
        }
    ]
}
]
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы

messageId (path переменная)	Long	ID сообщения
multimedia	Array of JSON-objects	Информация о мультимедиа
image	JSON-объект	Содержимое объекта image
fileName	String	Имя файла
width	Int	Ширина изображения
height	Int	Высота изображения
length	Long	Размер изображения в байтах
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа
video	JSON-объект	Содержимое объекта video
fileName	String	Имя файла
width	Long	Ширина изображения
height	Long	Высота изображения
length	Long	Размер изображения в байтах
thumbnailResourceRef	JSON-объект	Содержимое объекта thumbnailResourceRef. Изображение, которое отображается в сообщении для

		видео (опциональный параметр).
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
duration	Long	Продолжительность видео (опциональный параметр)
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл
mimeType	String	Тип медиа
replyMsgId	Long	ID сообщения, на которое отвечаем
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
hidden	Boolean	При значении true сообщение будет скрыто под спойлером
entities	Array of JSON-objects	Настройки для форматирования. каждый объект данного поля должен содержать описание

		только для одного из указанных полей (опциональный параметр)
mention	JSON-объект	Положение ментшена в тексте сообщения (опциональный параметр): offset: смещение; length: длина; userId: ID пользователя.
url	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
email	JSON-объект	Положение email в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
bold	JSON-объект	Положение текста выделенного жирным шрифтом в сообщении (опциональный параметр): offset: смещение; length: длина.
italic	JSON-объект	Положение текста выделенного курсивом в сообщении (опциональный параметр): offset: смещение; length: длина.
phone	JSON-объект	Положение телефона в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
underline	JSON-объект	Положение подчеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.

textUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка; disablePreview: отображать ли превью ссылки.
textFormattedUrl	JSON-объект	Положение ссылки в тексте сообщения (опциональный параметр): offset: смещение; length: длина; url: ссылка.
strike	JSON-объект	Положение зачеркнутого текста в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
monospace	JSON-объект	Положение текста с моноширифтом в тексте сообщения (опциональный параметр): offset: смещение; length: длина.
channel	JSON-объект	Содержимое объекта channel: offset: индекс символа с которого начинается текст канала в сообщении; length: длина текста с каналом; channelId: идентификатор канала.
botCommand	JSON-объект	Содержимое объекта botCommand: offset: индекс символа с которого начинается текст команды для бота в сообщении; length: длина текста с командой для бота.

buttonsMarkup	JSON-объект	<p>Разметка кнопок в сообщении (опциональный параметр):</p> <p>inlineMessageButtons: Array (кнопки в сообщении, опционально):</p> <p>id: идентификатор кнопки (обязательное поле);</p> <p>label: текст на кнопке (обязательное поле);</p> <p>callbackMessage: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).</p>
---------------	-------------	--

В ответ получаем объект обновленного сообщения.

## 5.4 Сообщение с последующей клавиатурой и разметка кнопок

Чат-бот может отправлять сообщения с клавиатурой для последующего сообщения-ответа. Дополнительно можно настраивать типы кнопок и новую разметку кнопок.

Описанные в данном разделе функции применяются для всех REST методов, раскрытых в разделах [5.2 Отправка сообщений](#) и [5.3 Обновление сообщений](#). Далее для примера будет использоваться только эндпоинт `sendTextMessage`, для остальных эндпоинтов все аналогично.

### 5.4.1 Сообщение с клавиатурой

Сообщение-вопрос отправленное с заполненным `followUpKeyboard` отображает кастомную клавиатуру указанного типа для последующего сообщения-ответа. Предусмотрены следующие виды клавиатуры:

- `tags` – тэги;
- `slider` – ползунок;
- `geolocation` – геолокация;
- `datePicker` – выбор даты.

В одном запросе может быть указан только один вида клавиатуры, иначе возвращается 400 Bad Request.

- POST метод  
`/botapi/v1/messages/sendTextMessage/{workspaceId}/{groupId}` – для текстовых сообщений.



Пейлоад запроса (json):

```
{
  "clientRandomId": 0,
  "message": "Text",
  ..., // Стандартные стандартные атрибуты запроса
  sendTextMessage
  "followUpKeyboard": {
    "tags": {
      "values": ["Мужчина", "Женщина"],
      "mutiple": false
    },
    "slider": {
      "minValue": 40,
      "maxValue": 150,
      "step": 1.0,
      "unitName": "кГ",
      "ranging": false
    },
    "geolocation": {},
    "datePicker": {
      "restrictToAdults": false
    }
  }
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
followUpKeyboard	JSON-объект	Настройки кастомной клавиатуры
tags	JSON-объект	Настройки тэгов: values: варианты ответа;

		multiple: разрешает множественный выбор при значении true.
slider	JSON-объект	Настройки ползунка: minValue: минимально возможное значение (обязательное); maxValue: максимальное возможное значение (обязательное); step: шаг одного деления. Дефолтное значение 1.0; unitName: единица измерения; ranging: разрешает настройку диапазона значений при значении true.
geolocation	JSON-объект	Активирует клавиатуру геолокации. Для корректной работы на сервере должна быть настроена поддержка геолокации.
datePicker	JSON-объект	Настройка для подтверждения совершеннолетия: restrictToAdults: при значении true содержит только даты совершеннолетних. Иначе разрешается выбор любой даты.

Тело ответа совпадает с sendMessage.

## 5.4.2 Типы кнопок

Предусмотрена настройка поведения кнопок при нажатии. В одном запросе может быть указан только один вид кнопки, иначе возвращается 400 Bad Request.

- POST метод

/botapi/v1/messages/sendTextMessage/{workspaceId}/{groupId} – для текстовых сообщений. Пейлоад запроса (json):

```
{
  "clientRandomId": 0,
  "message": "Text",
  ..., // Стандартные стандартные атрибуты запроса
  sendTextMessage
  "buttonsMarkup": {
    "inlineMessageButtons": [
      {
        "id": 0,
        "label": "label",
        "callbackMessage": "text",
        "callbackData": "data",
        "type": {
          "callbackMessage": {
            "text": "text",
            "data": "data"
          },
          "deeplink": {
            "url": "http://example.com"
          },
          "action": {
            "data": "data"
          }
        }
      }
    ]
  }
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
buttonsMarkup	JSON-объект	Разметка кнопок
inlineMessageButtons	Array	Выбранный тип разметки кнопок
id	Long	Идентификатор кнопки (обязательное поле)
label	String	Текст на кнопке (обязательное поле)
callbackMessage	String	Текст, которые отправится от имени пользователя после того как он нажмет кнопку (опционально, если не передан будет выведен текст кнопки)
callbackData	String	Дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).
type	JSON-объект	<p>Настройка поведения кнопки. Можно предусмотреть только один из типов ответного сообщения:</p> <p>1)callbackMessage (ответное сообщение в чат от имени пользователя):</p> <p>text: текст, который отправится от имени пользователя после</p>

		<p>нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData: дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю)</p> <p>2) deeplink (ссылка на ресурс внутри приложения или на внешний ресурс:</p> <p>url: адрес ссылки;</p> <p>3) action (при нажатии отправляет ивент боту, создавшему кнопку):</p> <p>data: данные которые будут отправлены боту по SSE внутри MESSAGE_BUTTON_CLICKED ивента.</p>
--	--	--

Пример MESSAGE\_BUTTON\_CLICKED ивента:

```
{
  "groupId":1234,
  "message":{
    "id":9876,
    ...// Остальные атрибуты сообщения
  },
  "buttonId":1,
  "data":"data",
  "userId":4321
}
```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
message	JSON-объект	
id	Long	ID сообщения
buttonId	Long	ID нажатой кнопки

Поле	Тип	Содержание
data	Long	Данные нажатой кнопки
userId	Long	ID пользователя, нажавшего кнопку

### 5.4.3 Разметка кнопок

Вместо стандартной разметки доступна настройка собственной разметки кнопок в сообщении Чат-бота. В `buttonsMarkup` предусмотрено два вида разметки кнопок в сообщении:

`inlineMessageButtons` – кнопка на одну линию;

`gridMessageButtons` – разметка сеткой. Максимальное количество колонок равно 4, максимальное количество всех кнопок в одном сообщении равно 20. Для корректной работы на сервере должна быть настроена поддержка этого функционала.

Одновременно может быть настроена только один вид разметки.

- POST метод

`/botapi/v1/messages/sendTextMessage/{workspaceId}/{groupId}` – для текстовых сообщений. Пейлоад запроса (json):

```
{
  "clientRandomId": 0,
  "message": "Text",
  ..., // Стандартные стандартные атрибуты запроса
  "sendTextMessage": {
    "buttonsMarkup": {
      "gridMessageButtons": [
        [ // первый ряд кнопок
          {
            "id": 1,
            "label": "label 1",
            "type": {
              "callbackMessage": {
                "text": "callbackMessage inline",
                "data": "callbackData inline"
              }
            }
          },
          {
            "id": 3,
            "label": "label inline",
            "type": {
              "callbackMessage": {
                "text": "callbackMessage inline",
                "data": "callbackData inline"
              }
            }
          }
        ]
      ]
    }
  }
}
```

```

    }
  }
},
[ // второй ряд кнопок
  {
    "id": 2,
    "label": "label 2",
    "type": {
      "callbackMessage": {
        "text": "callbackMessage inline",
        "data": "callbackData inline"
      }
    }
  }
]
]
}
}

```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
clientRandomId	Long	Уникальный идентификатор на основе временных меток
message	String	Текст сообщения
buttonsMarkup	JSON-объект	Разметка кнопок
gridMessageButtons	Array	Выбранный тип разметки кнопок
id	Long	Идентификатор кнопки (обязательное поле)
label	String	Текст на кнопке (обязательное поле)



type	JSON-объект	<p>Настройка типа поведения кнопки callbackMessage:</p> <p>text: текст, который отправится от имени пользователя после нажатия кнопки (опционально, если не передан, то будет выведен текст кнопки);</p> <p>callbackData:дополнительная информация, полученная от бота при создании кнопки из одноименного поля (опционально, не отображается пользователю).</p>
------	-------------	--

Пример MESSAGE\_BUTTON\_CLICKED ивента:

```
{
  "groupId":1234,
  "message":{
    "id":9876,
    ...// Остальные атрибуты сообщения
  },
  "buttonId":1,
  "data":"data",
  "userId":4321
}
```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
message	JSON-объект	
id	Long	ID сообщения
buttonId	Long	ID нажатой кнопки
data	Long	Данные нажатой кнопки
userId	Long	ID пользователя, нажавшего кнопку

## 5.5 Имитация набора текста сообщения

Для имитации набора текста сообщения Чат-ботом используется следующий REST метод:

- POST метод /botapi/v1/messages/typing/{workspaceId}.

Тело запроса (json):

```
{  
  "groupId":2204284738008927,  
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId	Long	ID группы

В ответ приходит статус-код 200.

## 5.6 Получение сообщений

Чат-бот может получать различные сообщения от других пользователей в рамках определенной группы мессенджера.

Для получения сообщений Чат-ботом в мессенджере можно использовать следующий REST метод:

- POST метод `/botapi/v1/messages/getMessages/{workspaceId}/{groupId}`.

Тело запроса:

```
{
  "messageIds": [ 1186920 ]
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageIds	Array of longs	ID сообщений

В ответ получаем:

```
{
  "groupId":2204284738008927,
  "id":1008435,
  "date":1660554032019,
  "clientRandomId":160768211,
  "senderId":2021713016764534,
  "links":[
    {
      "url":"https://web-backoffice.kchat.app/#/auth",
      "title":"Back Office",
      "icon":{
        "id":"88d15cf2-9380-490b-8824-a467e656886e",

```

```

        "url": "https://s3.kchat.app/fileupload/7NfrwAmU34%2F98b3
31f1-bd8e-4440-8631-9afd2a9ab31e",
        "key": "xd92H7R5CKpRb5GnA91oPulY5vhW2uMgxPV8k6OVSYU=",
        "transformation": "GOST3412-2015/ECB/PKCS7Padding"
    }
}
],
"message": "https://web-backoffice.kchat.app/#/auth",
"entities": [
    {
        "textUrl": {
            "offset": 0,
            "length": 46,
            "url": "https://web-backoffice.kchat.app/#/auth"
        }
    }
],
"multimedia": [

],
"position": 36,
"deleted": false,
"reactions": [

],
"stringId": "1008435",
"senderType": "DEFAULT",
"reactionGroups": [

]
}

```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
id	Long	ID сообщения
date	Long	Дата
clientRandomId	Long	Уникальный ID с учетом временных меток
senderId	Long	ID отправителя
links	Array	Информация о ссылках:

Поле	Тип	Содержание
		url: url ссылки; title: название; icon: JSON-объект в ссылке: id: ID объекта; url:url объекта; key:ключ шифрования; transformation: стандарт шифрования;
message	String	Текст сообщения
entities	Array	Информация о дополнительных объектах в сообщении: textUrl: JSON-объект: offset: смещение; length: длинна; url: url объекта.
multimedia	Array	Информация о мультимедиа
position	Long	Позиция
deleted	Boolean	Статус удаления
reactions	Array	Информация о реакциях
stringId	Long	ID строки
reactionGroups	Array	Информация о сгруппированных реакциях

## 5.7 Получение списка непрочитанных сообщений

Чат-бот может получать список всех непрочитанных сообщений в рамках определенной группы мессенджера.

Для получения списка всех непрочитанных сообщений в определенной группе пространства можно использовать REST метод POST:

- `/botapi/v1/messages/getAllUnreadMessages/{workspaceId}/{groupId}`.

Данный метод нужно вызывать без тела запроса.

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы

В ответ получаем:

```
{
  "groupId":2204284738008927,
  "id":1008435,
  "date":1660554032019,
  "clientRandomId":160768211,
  "senderId":2021713016764534,
  "links":[
    {
      "url":"https://web-backoffice.kchat.app/#/auth",
      "title":"Back Office",
      "icon":{
        "id":"88d15cf2-9380-490b-8824-a467e656886e",
        "url":"https://s3.kchat.app/fileupload/7NfrwAmU34%2F98b331f1-bd8e-4440-8631-9afd2a9ab31e",
        "key":"xd92H7R5CKpRb5GnA91oPulY5vhW2uMgxPV8k6OVSYU=",
        "transformation":"GOST3412-2015/ECB/PKCS7Padding"
      }
    }
  ],
  "message":"https://web-backoffice.kchat.app/#/auth",
```

```

"entities":[
  {
    "textUrl":{
      "offset":0,
      "length":46,
      "url":"https://web-backoffice.kchat.app/#/auth"
    }
  }
],
"multimedia":[

],
"position":36,
"deleted":false,
"reactions":[

],
"stringId":"1008435",
"senderType":"DEFAULT",
"reactionGroups":[

]
}

```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
id	Long	ID сообщения
date	Long	Дата
clientRandomId	Long	Уникальный ID с учетом временных меток
senderId	Long	ID отправителя
links	Array	Информация о ссылках: url: url ссылки; title: название; icon: JSON-объект в ссылке: id: ID объекта; url:url объекта; key:ключ шифрования; transformation: стандарт шифрования;

Поле	Тип	Содержание
message	String	Текст сообщения
entities	Array	Информация о дополнительных объектах в сообщении: textUrl: JSON-объект; offset: смещение; length: длинна; url: url объекта.
multimedia	Array	Информация о мультимедиа
position	Long	Позиция
deleted	Boolean	Статус удаления
reactions	Array	Информация о реакциях
stringId	Long	ID строки
reactionGroups	Array	Информация о сгруппированных реакциях



# 5.8 Удаление сообщений

Чат-бот может удалять сообщения в рамках определенной группы мессенджера в зависимости от его роли.

Для удаления Чат-ботом сообщений определенной группы в определенном пространстве можно использовать следующий REST метод:

- POST метод /botapi/v1/messages/deleteMessages/{workspaceId}/{groupId}

Тело запроса:

```
{
  "messageIds": [ 1186920 ]
}
```

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageIds	Array of longs	ID сообщений

В ответ приходит статус-код 200.

## 5.9 Удаление всех кнопок на сообщении

Для удаления Чат-ботом всех кнопок на сообщении можно использовать следующий REST метод:

- POST метод

/botapi/v1/messages/deleteMessageButtons/{workspaceId}/{groupId}/{messageId}

Данный метод нужно вызывать без тела запроса.

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы
messageId (path переменная)	Long	ID сообщения

В ответ приходит статус-код 200 и в теле ответа обновленный message.

## 5.10 Сохранение файлов в s3 хранилище

Передача любого типа файлов в мессенджере происходит следующим образом: файл сначала шифруется клиентом, затем в зашифрованном виде сохраняется (через fileupload REST) в s3 хранилище, и только потом ссылка на сохраненный ресурс прикрепляется к сообщению и пересылается адресату.

Для сохранения зашифрованного содержимого файла используется следующие REST методы:

- Для сохранения файла с прикрепленными ключами шифрования используется POST метод /api/v1/upload/ со следующими хедерами и телом:

```
Content-Type: multipart/form-data; boundary=...
Encryption-Key-Id: 345
Content-Disposition:          form-data;          name="file";
filename="some_file.pdf"
WorkspaceId: -1
Authorization: asdasd
В теле запроса сам файл
```

Описание объектов запроса:

Поле	Тип	Содержание
Content-Type	String	Тип формата передачи файла
Encryption-Key-Id	Long	ID ключа шифрования
Content-Disposition	String	Имя файла
WorkspaceId	Long	ID пространства
Authorization	String	authToken бота

Ответ будет:

```
{
  "resource": {
    "id": "c6fcf38e-62ea-430e-ad26-674fa8072572",
    "url": "https://s3.kchat.app/fileupload/5f136.pdf",
```

```

    "key": "DFUmZlEpXbzAT8AOSa6npKp06h9ijy4G9tS68l7qY2o=",
    "transformation": "GOST3412-2015/ECB/PKCS7Padding"
  }
}

```

Описание объектов ответа:

Поле	Тип	Содержание
resource	JSON-объект	Содержимое объекта resource
id	UUID	UUID файла
url	String	url файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования

- Для сохранения просто содержимого файла (шифрованного/нешифрованного) без ключей шифрования используется POST метод `/api/v1/upload/secret` со следующими хедерами и телом:

```

Content-Type: multipart/form-data; boundary=...
Content-Disposition: form-data; name="file";
filename="some_file.pdf"
WorkspaceId: -1
Authorization: asdasd
В теле запроса сам файл

```

Описание объектов запроса:

Поле	Тип	Содержание
Content-Type	String	Тип формата передачи файла
Content-Disposition	String	Имя файла
WorkspaceId	Long	ID пространства
Authorization	String	authToken бота

Ответ будет:

```
{
  "url": "https://s3.kchat.app/fileupload/5f136.pdf",
}
```

Описание объектов ответа:

Поле	Тип	Содержание
url	String	url файла

- Для сохранения файла или видео и шифрования его на сервере используется POST метод `/api/v1/upload/secret/encryptable` со следующими хедерами и телом:

```
Content-Type: multipart/form-data; boundary=...
Content-Length: 185,
Authorization: asdasd
В теле запроса сам файл
```

Описание объектов запроса:

Поле	Тип	Содержание
Content-Type	String	Тип формата передачи файла
Content-Length	Long	Длина запроса в байтах
Authorization	String	authToken бота

Ответ будет:

```
{
  "resource": {
    "id": "08e2ebde-21d9-48c8-9b97-a96d0ca90d7f",
    "url": "https://s3.kchat.app/fileupload/0YLXAfZ3kb%2F0845e362-b39a-4202-9031-ba30e74d8a2b.txt",
    "key": "8KBO3iFkdxhrzeBlLjHAh+QmQQrAxrWjuUYb9crDRfI=",
    "transformation": "GOST3412-2015/ECB/PKCS7Padding"
  }
}
```

Описание объектов ответа:

Поле	Тип	Содержание
resource	JSON-объект	Содержимое объекта resource
id	UUID	UUID файла
url	String	url файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования

- Для сохранения изображения и шифрования его на сервере используется POST метод `/api/v1/upload/image/encryptable` со следующими хедерами и телом:

```
Content-Type: multipart/form-data; boundary=...
Content-Length: 185,
Authorization: asdasd
В теле запроса сам файл
```

Описание объектов запроса:

Поле	Тип	Содержание
Content-Type	String	Тип формата передачи файла
Content-Length	Long	Длина запроса в байтах
Authorization	String	authToken бота

Ответ будет:

```
{
  "original": {
    "resource": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "width": 0,
    "height": 0
  }
}
```

```

},
"thumbnails": [
  {
    "resource": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "width": 0,
    "height": 0
  }
]
}

```

Описание объектов ответа:

Поле	Тип	Содержание
originals	JSON-объект	Содержимое объекта originals
resource	JSON-объект	Содержимое объекта resource
id	UUID	UUID файла
url	String	url файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
width	Int	Ширина изображения
height	Int	Высота изображения
thumbnails	Array of JSON-objects	Содержимое ресурса
resource	JSON-объект	Содержимое объекта resource
id	UUID	UUID файла
url	String	url файла
key	String	Ключ, которым зашифрован файл

transformation	String	Стандарт шифрования
width	Long	Ширина изображения
height	Long	Высота изображения

- Для сохранения изображения с прикрепленными ключами шифрования используется POST метод `/api/v1/upload/image` со следующими хедерами и пейлоадом

```
Content-Type: multipart/form-data; boundary=...
Encryption-Key-Id: 345
Content-Disposition:          form-data;          name="file";
filename="some_file.pdf"
WorkspaceId: -1
Authorization: asdasd
В теле запроса сам файл
```

Описание объектов запроса:

Поле	Тип	Содержание
Content-Type	String	Тип формата передачи файла
Encryption-Key-Id	Long	ID ключа шифрования
Content-Disposition	String	Имя файла
WorkspaceId	Long	ID пространства
Authorization	String	authToken бота

Ответ будет:

```
{
  "original": {
    "resource": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "width": 0,
    "height": 0
  }
}
```



```

},
"thumbnails": [
  {
    "resource": {
      "id": "80d30307-7122-4fbc-94c6-fa3691276453",
      "key": "XQh5NX=",
      "transformation": "GOST3412-2015/ECB/PKCS7Padding",
      "url": "https://s3.kchat.app/fileupload/5f136.jpg"
    },
    "width": 0,
    "height": 0
  }
]
}

```

Описание объектов ответа:

Поле	Тип	Содержание
originals	JSON-объект	Содержимое объекта originals
resource	JSON-объект	Содержимое объекта resource
id	UUID	UUID файла
url	String	url файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
width	Int	Ширина изображения
height	Int	Высота изображения
thumbnails	Array of JSON-objects	Содержимое ресурса
resource	JSON-объект	Содержимое объекта resource
id	UUID	UUID файла
url	String	url файла
key	String	Ключ, которым зашифрован файл

transformation	String	Стандарт шифрования
width	Int	Ширина изображения
height	Int	Высота изображения

## 5.11 Скачивание расшифрованных файлов

Для скачивания расшифрованных файлов следует использовать REST метод POST:

- /api/v1/download/secret/decryptable

Тело запроса:

```
"resourceRef": {  
  "id": "80d30307-7122-4fbc-94c6-fa3691276453",  
  "key": "XQh5NX=",  
  "transformation": "GOST3412-2015/ECB/PKCS7Padding",  
  "url": "https://s3.kchat.app/fileupload/5f136.pdf"  
}
```

Описание объектов запроса:

Поле	Тип	Содержание
resourceRef	JSON-объект	Содержимое объекта resourceRef. Сначала файл сохраняется зашифрованным в s3 хранилище, здесь данные файла из хранилища
id	UUID	UUID файла
key	String	Ключ, которым зашифрован файл
transformation	String	Стандарт шифрования
url	String	Url, где лежит зашифрованный файл

В ответе будет расшифрованное содержимое файла.

## 5.12 Получение всех групп

Чат-бот может получить список всех групп, в которых он является участником во всех пространствах, где он является участником. Для этого необходимо вызвать REST метод POST :

- /botapi/v1/groups/getAllUserGroupStates, передав туда следующий хедер **Authorization: authToken**, где **authToken** – токен Чат-бота.

В ответ получаем:

```
[
  {
    "groupId":2387045628266976,
    "counter":10,
    "lastMessageId":2003905,
    "lastSeenMessageId":2003878,
    "lastSeenMessagePosition":307,
    "opponentSeenId":2003878,
    "opponentSeenPosition":307,
    "mentionIds":[],
    "unseenDeletedPositions":[],
    "draftMessageId":0,
    "lastMessage":{
      "groupId":2387045628266976,
      "id":2003905,
      "date":1680698684970,
      "clientRandomId":178945799,
      "senderId":2367834071430918,
      "message":"10",
      "entities":[],
      "position":317,
      "deleted":false,
      "reactions":[],
      "stringId":"2003905",
      "senderType":"DEFAULT",
      "reactionGroups":[]
    },
    "group":{
      "workspaceId":-1,
      "id":2387045628266976,
      "type":"P2P",
      "visible":false,
      "autoJoin":false,
      "participantCount":2,

```

```

        "adminCount":2,
        "blockedCount":0,
        "archived":false,
        "creationDate":1679921805815,
        "createdBy":-2387015567499904,
        "pinnedMessages":[],
        "peer1Id":-2387015567499904,
        "peer2Id":2367834071430918
    },
    "role":"ADMIN",
    "notificationType":"OFF",
    "opponent":{
        "workspaceId":-1,
        "id":2367834071430918,
        "phone":74951010101,
        "firstName":"Ivan",
        "lastName":"Ivanov",
        "middleName":"",
        "inactive":false,
        "deleted":false,
        "bot":false,
        "userSystemGroupId":2367834071430918
    },
    "hidden":false,
    "pinned":false,
    "showTooltip":false,
    "blockedByMe":false,
    "blockedByOpponent":false,
    "historyStartPosition":0,
    "seenPinId":0
}
]

```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
counter	Long	Счетчик сообщений
lastMessageId	Long	ID последнего сообщения
lastSeenMessageId	Long	ID последнего просмотренного сообщения

Поле	Тип	Содержание
lastSeenMessagePosition	Long	Позиция последнего просмотренного сообщения
opponentSeenId	Long	ID последнего участника
opponentSeenPosition	Long	Позиция последнего участника
mentionIds	Array of longs	ID участника, отмеченного в сообщении
unseenDeletedPositions	Array of longs	Непросмотренные удаленные позиции
draftMessageId	Long	ID черновика сообщения
lastMessage	JSON-объект	Информация о последнем сообщении: groupId: ID группы id: ID сообщения date: дата clientRandomId: уникальный ID с учетом временных меток senderId: ID отправителя message: текст сообщения entities: дополнительные объекты в сообщении position: позиция deleted: статус удаления reactions: реакции stringId: ID строки senderType: тип отправителя reactionGroups: сгруппированные реакции
group	JSON-объект	Информация о группе: workspaceId: ID пространства id: ID группы type: тип группы visible: Доступность в поиске autoJoin: открытая или закрытая participantCount: счетчик участников

Поле	Тип	Содержание
		adminCount: счетчик администраторов blockedCount: Счетчик заблокированных Archived: Статус архивации creationDate: Дата создания createdBy: Кем создана pinnedMessages: закрепленные сообщения peer1Id: ID первого участника в p2p чате (опционально) peer2Id: ID второго участника в p2p чате (опционально)
role	String	Тип участника
notificationType	String	Статус нотификаций
opponent	JSON-объект	Информация об участнике (только для p2p чата): workspaceId: ID пространства id: ID участника phone: номер телефона firstName: Имя lastName: Фамилия middleName: Отчество Inactive: статус деактивации Deleted: статус удаления bot: статус бота userSystemGroupId: ID системной группы участника
hidden	Boolean	Статус скрытой группы
pinned	Boolean	Статус закрепленности группы
showTooltip	Boolean	Показ всплывающих подсказок
blockedByMe	Boolean	Заблокирован мной
blockedByOpponent	Boolean	Заблокирован другим участником

Поле	Тип	Содержание
historyStartPosition	Long	Стартовая позиция истории группы
seenPinId	Long	Видимый идентификатор контакта



## 5.13 Получение состояния Чат-бота в определенной группе

Для получения состояния Чат-бота в определенной группе используется следующий REST метод:

- POST метод `/botapi/v1/groups/getUserGroupState/{workspaceId}/{groupId}`

Данный метод нужно вызывать без тела запроса.

Описание объектов запроса:

Поле	Тип	Содержание
workspaceId (path переменная)	Long	ID пространства
groupId (path переменная)	Long	ID группы

Ответ будет:

```
{
  "groupId":2387045628266976,
  "counter":10,
  "lastMessageId":2003905,
  "lastSeenMessageId":2003878,
  "lastSeenMessagePosition":307,
  "opponentSeenId":2003878,
  "opponentSeenPosition":307,
  "mentionIds":[],
  "unseenDeletedPositions":[],
  "draftMessageId":0,
  "lastMessage":{
    "groupId":2387045628266976,
    "id":2003905,
    "date":1680698684970,
    "clientRandomId":178945799,
    "senderId":2367834071430918,
    "message":"10",
    "entities":[],
    "position":317,
    "deleted":false,
    "reactions":[],
    "stringId":"2003905",
    "senderType":"DEFAULT",
```

```

    "reactionGroups": []
  },
  "group": {
    "workspaceId": -1,
    "id": 2387045628266976,
    "type": "P2P",
    "visible": false,
    "autoJoin": false,
    "participantCount": 2,
    "adminCount": 2,
    "blockedCount": 0,
    "archived": false,
    "creationDate": 1679921805815,
    "createdBy": -2387015567499904,
    "pinnedMessages": [],
    "peer1Id": -2387015567499904,
    "peer2Id": 2367834071430918
  },
  "role": "ADMIN",
  "notificationType": "OFF",
  "opponent": {
    "workspaceId": -1,
    "id": 2367834071430918,
    "phone": 74951010101,
    "firstName": "Ivan",
    "lastName": "Ivanov",
    "middleName": "",
    "inactive": false,
    "deleted": false,
    "bot": false,
    "userSystemGroupId": 2367834071430918
  },
  "hidden": false,
  "pinned": false,
  "showTooltip": false,
  "blockedByMe": false,
  "blockedByOpponent": false,
  "historyStartPosition": 0,
  "seenPinId": 0
}

```

Описание объектов ответа:

Поле	Тип	Содержание
groupId	Long	ID группы
counter	Long	Счетчик сообщений
lastMessageId	Long	ID последнего сообщения
lastSeenMessageId	Long	ID последнего просмотренного сообщения
lastSeenMessagePosition	Long	Позиция последнего просмотренного сообщения
opponentSeenId	Long	ID последнего участника
opponentSeenPosition	Long	Позиция последнего участника
mentionIds	Array of longs	ID участника, отмеченного в сообщении
unseenDeletedPositions	Array of longs	Непросмотренные удаленные позиции
draftMessageId	Long	ID черновика сообщения
lastMessage	JSON-объект	Информация о последнем сообщении: groupId: ID группы id: ID сообщения date: дата clientRandomId: уникальный ID с учетом временных меток senderId: ID отправителя message: текст сообщения entities: дополнительные объекты в сообщении position: позиция deleted: статус удаления reactions: реакции stringId: ID строки senderType: тип отправителя reactionGroups: сгруппированные реакции

Поле	Тип	Содержание
group	JSON-объект	Информация о группе: workspaceId: ID пространства id: ID группы type: тип группы visible: Доступность в поиске autoJoin: открытая или закрытая participantCount: счетчик участников adminCount: счетчик администраторов blockedCount: Счетчик заблокированных Archived: Статус архивации creationDate: Дата создания createdBy: Кем создана pinnedMessages: закрепленные сообщения peer1Id: ID первого участника в р2р чате (опционально) peer2Id: ID второго участника в р2р чате (опционально)
role	String	Тип участника
notificationType	String	Статус нотификаций
opponent	JSON-объект	Информация об участнике (только для р2р чата): workspaceId: ID пространства id: ID участника phone: номер телефона firstName: Имя lastName: Фамилия middleName: Отчество Inactive: статус деактивации Deleted: статус удаления bot: статус бота userSystemGroupId: ID системной группы участника
hidden	Boolean	Статус скрытой группы

Поле	Тип	Содержание
pinned	Boolean	Статус закреплённости группы
showTooltip	Boolean	Показ всплывающих подсказок
blockedByMe	Boolean	Заблокирован мной
blockedByOpponent	Boolean	Заблокирован другим участником
historyStartPosition	Long	Стартовая позиция истории группы
seenPinId	Long	Видимый идентификатор контакта

## 5.14 Звонки

Для совершения звонков используется следующий REST метод:

- POST метод /botapi/v1/vcs/startCall – для начала звонка.

Тело запроса:

```
{
  "userId":195482658,
  "userPhone":"79101234567"
}
```

Необходимо обязательно заполнить один из параметров (userId либо userPhone). Если userId == null, то будет использоваться userPhone.

Описание объектов запроса:

Поле	Тип	Содержание
userId	Long	Идентификатор пользователя, которому звонят (необязательное поле)
userPhone	String	Телефон пользователя, которому звонят (необязательное поле)

Ответ будет:

```
{
  "config": {
    "callId": 2724580024848919,
    "workspaceId": -1
  }
}
```

Описание объектов ответа:

Поле	Тип	Содержание
config	JSON-объект	Содержимое объекта config
callId	Long	Идентификатор звонка
workspaceId	Long	Идентификатор пространства

## 6 Разработка Чат-ботов с использованием специальной библиотеки приложения

В мессенджере реализована библиотека для упрощения разработки скриптов Чат-ботов на языке программирования Python (начиная с версии 3.7).

Для получения доступа к данной библиотеке необходимо обратиться к команде разработки мессенджера, которая предоставит файл с архивом исходного кода для подключения библиотеки в проекте заказчика.

### Пример кода Чат-бота, реализованного с использованием библиотеки приложения

```
import logging
import sys

import requests
from requests import get

from messenger_bot_api import *

root = logging.getLogger()
root.setLevel(logging.DEBUG)

handler = logging.StreamHandler(sys.stdout)
handler.setLevel(logging.DEBUG)
formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
handler.setFormatter(formatter)
root.addHandler(handler)

token = 'BOT-a25a4807-6b50-445b-adf6-883e0fa111dc-f2917702-8ca5-4427-b923-3cac6464fd5d-1722230974460-c60fd54a-50a5-4fcb-9939-a29e366027cd'
rest = 'https://api.kchat.app'
sse = 'https://pusher.kchat.app'
file = 'https://s3.kchat.app'

logger = logging.getLogger('bot')

image_url = 'https://i.guim.co.uk/img/media/26392d05302e02f7bf4eb143bb84c8097d09144b/446_167_3683_2210/master/3683.jpg?width=1900&dpr=2&s=none'
image_response = requests.get(image_url)
```

```

video_url =
'https://media.istockphoto.com/id/1388629426/video/top-down-aerial-view-of-mangrove-f
orest-drone-zoom-out-above-the-tree.mp4?s=mp4-640x640-is&k=20&c=DOH2kXWpimBAFdc1SrUpa
ksT_TwA3ISDvfzMZNvqIjo='
video_response = requests.get(video_url)

def start_handler(event):
    event.reply_text(
        'Bot commands \n'
        ' /start - start command \n'
        ' /get_bot_ip - get bot ip \n'
        ' /buttons - get message with buttons \n'
        ' /send_multimedia - send multimedia \n'
        ' /send_image - send image \n'
        ' /send_video - send video \n'
        ' /send_file - send file\n'
        'If the message begins with "reply" the bot will send the bot forwarded this
message back'
    )

def message_with_buttons_handler(event: MessageBotEvent):
    message = MessageRequest('Message with buttons:', None,
                             [InlineMessageButton(1, "button1", "You pressed button
1", "callbackData"),
                             InlineMessageButton(2, "button2", "You pressed button
2",
                             "{ 'botCommand': { 'offset': 16, 'length': 6 } }"),
                             InlineMessageButton(3, "button3", "You pressed button
3"),
                             InlineMessageButton(4, "button4", "You pressed button
4")])
    event.reply_text_message(message)

def multimedia_handler(event: MessageBotEvent):
    event.reply_multimedia_message([Image('3683.jpg', image_response.content)],
                                   [Video('3683.mp4', video_response.content)],
                                   MessageRequest('Multimedia'))

def image_handler(event: MessageBotEvent):
    event.send_image_message(event.workspace_id,
                             event.group_id,
                             Image('3683.jpg', image_response.content),
                             MessageRequest('Image'))

def video_handler(event: MessageBotEvent):
    event.send_video_message(event.workspace_id,
                             event.group_id,
                             Video('3683.mp4', video_response.content),
                             MessageRequest('Video'))

```



```

def file_handler(event: MessageBotEvent):
    event.reply_file_message(File('3683.jpg', image_response.content),
MessageRequest('File'))

def ip_handler(event: MessageBotEvent):
    ip = get('https://api.ipify.org').content.decode('utf8')
    event.reply_text(f'Bot ip is {ip}')

def update_event_handler(event: MessageUpdateBotEvent):
    event.confirm_event_from_current_group(event.event_id)

def delete_event_handler(event: MessageDeleteBotEvent):
    event.confirm_event_from_current_group(event.event_id)

def messages_delete_event_handler(event: MessagesDeleteBotEvent):
    event.confirm_event_from_current_group(event.event_id)

def group_me_add_event_handler(event: GroupMeAddBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_me_remove_event_handler(event: GroupMeRemoveBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_history_clear_event_handler(event: HistoryClearGroupBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_delete_event_handler(event: DeleteGroupBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_user_remove_event_handler(event: GroupUserRemoveBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_user_add_event_handler(event: GroupUsersAddBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_user_join_event_handler(event: GroupUserJoinBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def group_user_leave_event_handler(event: GroupUserLeaveBotEvent):
    event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def user_signed_up_to_workspace_handler(event: UserSignedUpToWorkspaceBotEvent):

```

```

event.confirm_event(event.workspace_id, event.group_id, event.event_id)

def click_button_event_handler(event: MessageBotEvent):
    selected_button = event.selected_button
    message_text = generate_text(selected_button.id, selected_button.callback_data)
    if message_text is not None:
        event.reply_text(message_text)

def generate_text(button_id: int, value):
    if button_id == 1:
        return f'Confirm click by button with id: {button_id} and callbackData {value}'
    if button_id == 2:
        return f'Confirm click by button with id: {button_id} and callbackData {value}'
    else:
        return None

def get_ip_rest():
    try:
        return get('https://api.ipify.org').content.decode('utf8')
    except Exception as e:
        logger.exception('get ip failed', exc_info=e)
        return None

def message_handler(event: MessageBotEvent):
    txt: str = event.message_text.lower() if event.message_text is not None else None
    reply_message_id = None
    if txt == 'hello':
        resp = 'Hi, I am bot'
    elif txt == 'how are you':
        resp = 'I am fine'
    elif txt == 'bye':
        resp = 'Bye'
    elif txt is not None and txt.startswith('reply'):
        resp = 'I replied to the message'
        reply_message_id = event.message_id
    else:
        resp = event.message_text
    event.reply_text_message(MessageRequest(resp, reply_message_id))

def main():
    bot = Application(token, {'api_base_url': rest, 'sse_base_url': sse,
                              'file_upload_base_url': file})
    bot.add_handler(ClickButtonEventHandler(click_button_event_handler))
    bot.add_handler(CommandHandler('start', start_handler))
    bot.add_handler(CommandHandler('buttons', message_with_buttons_handler))
    bot.add_handler(CommandHandler('send_multimedia', multimedia_handler))
    bot.add_handler(CommandHandler('send_image', image_handler))
    bot.add_handler(CommandHandler('send_video', video_handler))
    bot.add_handler(CommandHandler('send_file', file_handler))

```

```

bot.add_handler(CommandHandler('get_bot_ip', ip_handler))
bot.add_handler(MessageHandler(message_handler))
bot.add_handler(UpdateMessageEventHandler(update_event_handler))
bot.add_handler>DeleteMessageEventHandler(delete_event_handler))
bot.add_handler>DeleteMessagesEventHandler(messages_delete_event_handler))
bot.add_handler(GroupMeAddHandler(group_me_add_event_handler))
bot.add_handler(GroupMeRemoveHandler(group_me_remove_event_handler))
bot.add_handler(GroupHistoryClearHandler(group_history_clear_event_handler))
bot.add_handler(GroupDeleteHandler(group_delete_event_handler))
bot.add_handler(GroupUserRemoveHandler(group_user_remove_event_handler))
bot.add_handler(GroupUsersAddHandler(group_user_add_event_handler))
bot.add_handler(GroupUserJoinHandler(group_user_join_event_handler))
bot.add_handler(GroupUserLeaveHandler(group_user_leave_event_handler))

bot.add_handler(UserSignedUpToWorkspaceHandler(user_signed_up_to_workspace_handler))

bot.start()

if __name__ == '__main__':
    main()

```

## Логика скрипта:

- создает Чат-бота;
- передает ему Токен и URL на ресурсы Bot API;
- регистрирует обработчики команд (start, get\_bot\_api, buttons, send\_multimedia, send\_image, send\_video, send\_file), общий обработчик для полученных сообщений (MessageHandler), а также обработчики для различных типов событий;
- задает логику обработчиков команд, сообщений и событий;
- стартует скрипт Чат-бота.