

# REST. Табличный вид

REST API подход использует HTTP-методы (GET, POST, PUT, DELETE и т.д.) для управления сущностями с уникальными URL.

GET /getUpdates?offset={number\_last\_message}&timeout=30

Запрос новых сообщений пользователя с сервера ТГ

Метод: GET

Эндпоинт: /getUpdates

## Request

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
offset	int	query string	Идентификатор первого обновления, которое нужно вернуть. Должен быть на единицу больше самого большого update_id из предыдущего запроса.	да
timeout	int	query string	Таймаут в секундах для длинного опроса. Сервер будет держать соединение открытым до этого времени, ожидая новые данные. Как только появятся новые обновления или таймаут истечет, сервер вернет ответ.	да

## Response

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
ok	boolean	body	Флаг, указывающий на успешность запроса.	да
result	array	body	Массив объектов с обновлениями. Каждый объект в массиве представляет собой одно событие (входящее сообщение, callback от кнопки и т.д.).	да
↳ update_id	int	body	Уникальный идентификатор обновления.	да

↳ message	object	body	Объект, содержащий информацию о входящем сообщении. Присутствует только если обновление является сообщением.	нет
↳ edited_message	object	body	Объект, содержащий информацию об отредактированном сообщении.	нет
↳ callback_query	object	body	Объект, содержащий информацию о нажатии на inline-кнопку.	нет

Response code

200 - Success

400 Bad Request: Некорректный синтаксис запроса / неверные параметры

401 Unauthorized: Указан неверный или недействительный токен бота

429 - Too Many Requests: Превышены лимиты Telegram

5xx - Ошибки сервера Telegram

GET /users/{telegram\_id}

Получение информации о пользователе по ID

Метод: GET

Эндпоинт: /users

### Request

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
telegram_id	int	path	Уникальный идентификатор пользователя в ТГ	да

### Response

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
telegram_id	int	body	Уникальный идентификатор пользователя в ТГ	да
username	string	body	@username пользователя в ТГ	нет
first_name	string	body	Имя пользователя	да
last_name	string	body	Фамилия пользователя	нет
timezone	string	body	Часовой пояс пользователя	да

Response code

200 - Success

404 Not Found: Пользователь с указанным telegram\_id не найден в БД

500 Internal Server Error: Внутренняя ошибка БД

503 Service Unavailable: БД временно недоступна

POST /chat/completions

Создание расписания для пользователя

Метод: POST

Эндпоинт: /chat/completions

### Request

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
Authorization	string	headers	Обязательный заголовок для аутентификации (Bearer {your_api_key})	да
Content-Type	string	headers	Указывает, что тело запроса представлено в формате JSON (application/json)	да
model	string	body	Идентификатор модели LLM	да
messages	array	body	Массив сообщений (история диалога пользователя с LLM)	да
↳ role	string	body	Роль отправителя сообщения. (system (задает поведение ассистента), user (сообщение от пользователя), assistant (ответы модели))	да
↳ content	string	body	Текст сообщения (промпт + запрос пользователя)	да
temperature	int	body	Параметр, контролирующий случайность ответов	нет
max_tokens	int	body	Максимальное количество токенов для генерации в ответе. Ограничивает длину выходного текста	нет
stream	boolean	body	Указание, будет ли ответ приходить частями (потоково)	нет

## Response

Название параметра	Тип данных	Находится в	Описание	Обязательность параметра
id	string	body	Уникальный идентификатор ответа от API	да
model	string	body	Идентификатор модели, сгенерировавшей ответ	да
choices	array	body	Массив с результатами генерации	да
↳ message	object	body	Объект, содержащий сгенерированный ответ	да
↳ role	string	body	Роль создателя сообщения	да
↳ content	string	body	Ответ, сгенерированный LLM	да
↳ finish_reason	string	body	Причина завершения генерации (stop — модель естественно завершила ответ, length — достигнут лимит токенов)	да
usage	object	body	Объект с информацией об использованных токенах	да
↳ prompt_tokens	int	body	Количество токенов, потраченных на запрос	да
↳ completion_tokens	int	body	Количество токенов в сгенерированном ответе	да
↳ total_tokens	int	body	Суммарное количество	да

## Response code

200 - Success

400 - Bad Request: Неверный запрос (например, не указан model или messages)

401 - Unauthorized: Неверный, отсутствующий или просроченный API-ключ

429 - Too Many Requests: Превышены лимиты запросов или токенов

5xx - Ошибки сервера LLM