

API ReMarked

Новое актуальное API

API для сервиса отзывы

- [Общая информация](#)
- [Ресурсы](#)
- [Добавление отзывов](#)
- [Получение списка отзывов](#)

API для сервиса по работе с депозитами

- [Общая информация](#)
- [Создание депозита](#)

API для вебхуков по событиям продаж

- [Общая информация](#)
- [Получение токена](#)
- [Создание события по заказу](#)

API для информации о гостях

- [Общая информация](#)
- [Общая структура тела запроса](#)
- [Поиск информации о госте](#)
- [Обновление информации о госте](#)
- [Создание гостя](#)
- [Обновление тегов гостя](#)
- [Дополнительное API работы с гостем](#)

API WiFi WebHooks

- [Общая информация](#)
- [Общая структура тела запроса](#)

- [Формат ответа](#)

API для сервиса по отправке сообщений в Whatsapp

- [Общая информация](#)
- [Тело запроса](#)
- [Формат ответа](#)

API для сервиса ОТЗЫВЫ

Общая информация

Адрес для запроса - <https://app.remarked.ru/api/v1/api>

Параметры запроса передаются только в формате json.

id - это необязательный параметр, он желателен, чтобы по логам отслеживать ошибки, лучше его передавать, но и без него отправка данных возможна

Ресурсы

```
{
  "restoclub.ru": "Restoclub.ru",
  "afisha.ru": "Afisha.ru",
  "allcafe.ru": "Allcafe.ru",
  "google": "Google",
  "twogis": "2ГИС",
  "tripadvisor": "Tripadvisor",
  "yandex": "Яндекс",
  "restoran.ru": "Restoran.ru",
  "foursquare": "foursquare",
  "deliveryclub": "Delivery Club",
  "vk": "VK",
  "instagram": "Instagram",
  "relaxby": "Relax.by",
  "tomesto": "ТоМесто",
  "flamp": "Flamp",
  "restorating": "Restorating",
  "llike": "Llike",
  "zoon": "Zoon",
  "yelp": "Yelp",
  "site": "Сайт",
  "app": "Приложение"
  "delivery_site" : "Доставка сайт"
}
```

Добавление ОТЗЫВОВ

Метод вызывается по ссылке: <https://app.remarked.ru/api/v1/api>

Данный метод обрабатывает только POST запрос. Для взаимодействия с ним необходимо

передать token. Он может быть передан как параметр в теле POST запроса, так и в виде заголовка **Authorization: Bearer <token>**. **Обязательны к передаче поля «date» и «source».**

Тело запроса

```
{
  "method": "ZumaReviewsApi.CreateReviewWithAllFields",
  "jsonrpc": "2.0",
  "params": {
    "token": <string> Токен указывается только в случае, если не передан в
заголовке
    "fields": {
      "date": <datetime> Дата написания отзыва, в формате YY-mm-dd
HH:ii:ss,
      "source": <string> Ресурс(укажите ключ ресурса из блока ресурсы),
... //перечисление полей
    }
  },
  "id": <string> Id операции //Необязательно
}
```

В ответ метод отдает данные в формате json

В случае успеха:

```
{
  "jsonrpc": "2.0",
  "result": {
    "status": "ok",
    "uid": <integer> Id добавленного отзыва
  },
  "id": <string> Id операции
}
```

В случае возникновения различного рода ошибок:

```
{
  "jsonrpc": "2.0",
  "error": {
    "code": <integer> Код ошибки,
    "message": <string> Сообщение с ошибкой
  },
  "id": <string> Id операции
}
```

Получение списка отзывов

Метод вызывается по ссылке <https://app.remarked.ru/api/v1/getRewievsList>

Метод обрабатывает как GET так и POST запросы. Для взаимодействия с ним необходимо передать token. Он может быть передан как параметр GET/POST запроса, так и в виде заголовка **Authorization: Bearer <token>**

В качестве фильтра вы можете передать следующие параметры:

1. **from**: дата сортировки **от**, в формате YYYY-mm-dd
2. **to**: дата сортировки **до**, в формате YYYY-mm-dd

В ответ метод отдает данные в формате json:

```
{
  "status": <string>,
  "data":<object> {
    "reviews":<array> [
      {
        "source":<string> Ресурс, с которого взят отзыв ,
        "datetime": <datetime> Дата размещения YYYY-mm-dd HH:ii:ss,
        "organization": <string> Организация/Заведение Nullable,
        "rating":<integer> Рейтинг/классы,
        "comment":<string> Текст отзыва
      }
    ]
  }
}
```

Запрос вида:

```
/getRewievsList?token=xxxxxxxxxxxxxxxxxxxx&from=2025-01-01&to=2025-05-31&page=1&per_page=100
```

Возвращает 100 записей, где:

- **per_page=100** - число записей, которые вернуться в ответе на запрос

API для сервиса по работе с сертификатами

Общая информация

Адрес для запросов к API - <https://app.remarked.ru/api/v1/api>

Параметры запроса передаются только в формате json.

id - это необязательный параметр, он желателен, чтобы по логам отслеживать ошибки, лучше его передавать, но и без него отправка данных возможна.

Для взаимодействия с методами API необходимо передать token. Он может быть передан как параметр в теле POST запроса, так и с помощью заголовка **Authorization: Bearer <token>**

Создание депозита

Метод вызывается по ссылке: <https://app.remarked.ru/api/v1/api>

Данный метод обрабатывает только POST запрос.

Тело запроса(только в формате json)

```
{
  "method": "IikoDepositApi.createDeposit",
  "jsonrpc": "2.0",
  "params": {
    "token": <string> Токен указывается только в случае, если не передан в заголовке,
    "fields": {
      "date": <datetime> Дата создания заявки, в формате YYYY-mm-dd HH:ii:ss,
      "name": <string> Имя гостя,
      "sum": <integer> Сумма бонусов для начисления(должно быть > 0),
      "surname": <string> фамилия гостя,
      "phone": <string> номер телефона гостя,
      "email": <string> электронный адрес(если есть),
      "position": <string> должность покупателя для сертификатов с типом(type) "Обычный" (не обязательно к заполнению),
      "use": <string> назначение для сертификатов с типом "Бартерный" (не обязательно к заполнению),
```

```

    "reason": <string> причина для сертификатов с типом "Бартерный" (не
    обязательно к заполнению),
    "certificates": <array> список сертификатов [
        {
            "nominal": <integer> сумма сертификата(принимает 1000, 3000,
            5000, 10000),
            "type": <string> тип сертификата (Обычный/Бартерный/Билет)
        },
        {
            "nominal": <integer> сумма сертификата(принимает 1000, 3000,
            5000, 10000),
            "type": <string> тип сертификата (Обычный/Бартерный/Билет)
        }
    ]
}
},
"id": <string> ID операции
}

```

В ответ метод отдает данные в формате json

В случае успеха:

```

{
  "jsonrpc": "2.0",
  "result": {
    "status": "ok",
    "uid": <integer> ID добавленного гостя
  },
  "id": <string> ID операции
}

```

В случае возникновения различного рода ошибок:

```

{
  "jsonrpc": "2.0",
  "error": {
    "code": <integer> Код ошибки,
    "message": <string> Сообщение с ошибкой
  },
  "id": <string> ID операции
}

```

API для вебхуков по событиям продаж

В настройках сотрудниками ReMarked создаётся интеграция с указанием домена и поинта (числового кода заведения), на основании проверки этих данных API разрешает или запрещает запросы с определенного домена.

Общая информация

Адрес для запроса - <https://app.remarked.ru/api/v1/ApiSalesWebhook>

Все запросы осуществляются методом post.

Все передается в json.

request_id - это необязательный параметр, он желателен, чтобы по логам отслеживать ошибки, лучше его передавать, но и без него отправка данных возможна

Получение токена

Запрос

```
{
  method: 'GetToken',
  point: POINT,
  request_id: IDEMPOTENCY KEY
}
```

Ответ

```
{
  "token": TOKEN
}
```

Создание события по заказу

```
{
  method: 'CreateSalesWebhook',
  token: TOKEN,
  purchase: {
```

```
        name: "TestClient",
        phone: "+79999999999",
        email: "test@mail.ru",
open_date: "2021-10-25",
open_time: "11:30",
close_date: "2021-10-25",
close_time: "11:30",
        order_id: "540ed7bf-eea2-4adc-9a3e-1131e907f5ed" (код заказа),
        order_number: "82806" (номер заказа),
order_type: "delivery" (delivery доставка, sales заказ в зале),
        guests_count: 4,
        utm: "utm_source=google",
        table_name: "305"(номер стола для заказа),
        coupon: "Название купона" (промо-код),
delivery_date: "2021-10-25" (дата, на которую заказана доставка),
delivery_time: "11:30" (время, на которое заказана доставка),
actual_delivery_date: "2021-10-25" (реальная дата, на которую доставлено),
actual_delivery_time: "11:30" (реальное время, на которое доставлено),
order_sum: "1100" (сумма заказа),
discount_sum: "0" (сумма скидки),
increase_sum: "0" (сумма наценки),
bonus_sum: "100" (сумма потраченных бонусов),
pay_sum: "1000" (сумма к оплате),
discount: "10%" (скидка)
    },
    request_id: IDEMPOTENCY KEY
}
```

API для информации о гостях

Общая информация

Адрес для совершения запросов - app.remarked.ru/api/v1/api

Все запросы передаются методом POST, с использованием протокола JSON-RPC.

Общая структура тела запроса

```
{
  "jsonrpc": "2.0",
  "id": REQUEST_ID,
  "method": METHOD,
  "params": {
    "PARAM_1": VALUE_1,
  }
}
```

где

- **REQUEST_ID** - id запроса, который определяется на запрашивающей данные стороне (сервер в ответе на запрос вернет его же)
- **METHOD** - метод запроса,
- **PARAM_1, VALUE_1** - пары параметр - значение

Поиск информации о госте

METHOD - GuestsApi.GetGuestsData

PARAMS:

Обязательные:

1. **token** - токен доступа
2. **phone** - телефон (в формате +79998887766) или
3. **id** - ID в базе remarked
или
4. **ext_uuid** - идентификатор внешней системы(пока не реализован)

Необязательные:

1. **point** - идентификатор дочерней точки, по которой нужно выгрузить данные. Если не указан, данные возвращаются по всем точкам.
- 2.

Структура ответа

```
{
  "jsonrpc": "2.0",
  "result": {
    "status": "ok",
    "data": [
      {
        "id": <integer> ID гостя Remarked,
        "ext_uuid": <integer> ID внешней системы,
        "phones" : [Массив номеров телефонов],
        "name": <string> Имя,
        "surname": <string> Фамилия,
        "patronymic" : <string> Отчество,
        "fio":<string> ФИО,
        "email": <string> Email,
        "birthday": <date> Дата рождения,
        "registration_date": <datetime> Дата регистрации,
        "is_subscriber": <bool> Согласен на рассылку,
        "cards": [Массив номеров карт],
        "comment": <string> Комментарий ,
        "bonuses":<decimal> Сумма бонусов,
        "photo": <string> URL фотографии
      },
    ]
  },
  "id": REQUEST_ID
}
```

Если передан и id и phone, поиск происходит по id

Обновление информации о госте

METHOD - GuestsApi.UpdateGuest

PARAMS:

Обязательные:

1. **token** - токен доступа
2. **id** - ID в базе remarked
или
3. **ext_uuid** - идентификатор внешней системы (если передано и то и другое, то идет поиск по id и обновление у найденного гостя ext_uuid)
(поиск по нему пока не реализован)
4. **fields** любые из полей на обновление (см запрос выше), кроме id, phone, cards, registration_date, bonuses

Структура ответа

```
{
  "jsonrpc": "2.0",
  "result": {
    "status": "ok",
  },
  "id": REQUEST_ID
}
```

или

```
{
  "jsonrpc": "2.0",
  "result": {
    "status": "error",
    "message": СООБЩЕНИЕ ОБ ОШИБКЕ
  },
  "id": REQUEST_ID
}
```

Создание гостя

METHOD - GuestsApi.CreateGuest

PARAMS:

Обязательные:

1. **token** - токен доступа
2. **fields** - данные о госте
Последние содержат обязательные ключи (phone) и необязательные (surname, name, patronymic, fio, photo (ссылка), birthday, gender, comment, is_subscriber, email, ext_uuid)

Формат полей:

- phone - Str - '+7xxxxxxxxxx'
- surname, name, patronymic, fio, comment - Str
- email - формат email адреса
- photo - url картинки
- birthday - DATE - YYYY-mm-dd
- gender - enum('male','female',")
- is_subscriber - enum(0,1)
- ext_uuid - Str

Структура ответа

```
{
  "jsonrpc": "2.0",
  "result": {
```

```
    "status": "ok",
    "gid": <integer>ID гостя Remarked
  },
  "id": REQUEST_ID
}
```

или

```
{
  "jsonrpc": "2.0",
  "result": {
    "status": "error",
    "message": СООБЩЕНИЕ ОБ ОШИБКЕ
  },
  "id": REQUEST_ID
}
```

Обновление тегов гостя

[Вернуть список тегов в заведении](#)

Request

POST <https://app.remarked.ru/api/v2/guest/tags/get>

payload

```
{
  "point": 0
}
```

Response samples

```
{
  "data": {
    "tags": [
      {
        "id": 7359,
        "name": "Непереносимость глютена",
        "color": "#C62828"
      },
      {
        "id": 7360,
        "name": "Орехи",

```

```
        "color": "#8BC34A"
      },
      {
        "id": 7364,
        "name": "Морепродукты",
        "color": "#AD1457"
      },
      {
        "id": 7365,
        "name": "Лук",
        "color": "#283593"
      }
    ]
  }
}
```

Назначить теги гостю

Request

POST <https://app.remarked.ru/api/v2/guest/tags/set>

payload

```
{
  "guest_id": 0,
  {
    "tags": [
      "id": 0,
      "id": 0
    ]
  }
}
```

Response samples

```
{
  "data": {
    "success": true
  }
}
```

Дополнительное API работы с гостем

<https://docs.google.com/document/d/1EY12GBOkOmMd6aiil69KNvL70hpTfD2BT5UOenZ6Rig/edit#heading=h.ikx6kkepubs4>

API WiFi WebHooks

Общая информация

Модуль предназначен для отправки в систему данных о WiFi авторизациях гостя по телефону.

Адрес для совершения запросов <https://app.remarked.ru/api/v1/ApiGuestWifi>

Общая структура тела запроса

```
{
  "phonenumber": <string>,
  "token": <string>,
  "mac": <string>,
  "date": <string>,
  "name": <string>,
  "email":<string>
}
```

Параметр	Описание	Обязательный
phonenumber	Номер телефона использованный при авторизации	Да
token	API токен	Да
mac	mac адрес устройства	Нет
date	Дата время авторизации в формате "YYYY-MM-DD HH:II:SS"	Нет
name	ФИО/Имя пользователя	Нет
email	email пользователя	Нет

Формат ответа

Успешно

```
{
  "status": "OK",
  "timestamp": 1680783455
}
```

Параметр	Описание
<code>status</code>	Статус выполнения запроса
<code>timestamp</code>	Временная метка

Ошибка

```
{
  "status": "error",
  "timestamp": 1680785988,
  "code": 10,
  "message": "invalid token"
}
```

Параметр	Описание
<code>status</code>	Статус выполнения запроса
<code>timestamp</code>	Временная метка
<code>code</code>	Код ошибки
<code>message</code>	Описание ошибки

Описание кодов ошибок

code	message	Описание ошибки
------	---------	-----------------

10	invalid token	Неправильный или несуществующий токен
20	invalid data	Неверно переданы данные
21	invalid phone	Неверно передан телефонный номер
30	internal error	Внутренняя ошибка

API для сервиса по отправке сообщений в Whatsapp

Общая информация

Адрес для запросов к API - <https://app.remarked.ru/api/v1/api>

Параметры запроса передаются только в формате json.

id - это необязательный параметр, он желателен, чтобы по логам отслеживать ошибки, лучше его передавать, но и без него отправка данных возможна.

Для взаимодействия с методами API необходимо передать token. Он может быть передан как параметр в теле POST запроса, так и с помощью заголовка Authorization: Bearer <token>

Тело запроса

(только в формате json)

```
{
  "method": "RemarkedWhatsappApi.SendMessage",
  "jsonrpc": "2.0",
  "params": {
    "token": <string>
    "phone": <string> Телефонный номер получателя в формате "+XXXXXXXXXX",
    "local_phone": <string> Телефонный номер отправителя в формате "+XXXXXXXXXX",
    "message": <string> Текст сообщения,
    "image": <string> Содержимое файла изображения, закодированное в base64.
    Обязательно наличие одного из параметров: "message" или "image"
  },
  "id": <string> ID операции
}
```

Формат ответа

В ответ метод отдает данные в формате json

В случае успеха:

```

{
    "jsonrpc": "2.0",
    "result": {
        "status": "ok",
        "data": {
            "success": true, <boolean>
            "details": {} <array> данные содержащие переданный запрос к
сервису whatsapp и ответ на запрос
        }
        "id": <string> ID операции
    }
}

```

В случае возникновения различного рода ошибок:

```

{
    "jsonrpc": "2.0",
    "result": {
        "status": "ok",
        "data": {
            "success": false, <boolean>
            "error": true, <boolean>
            "details": {} <array> данные содержащие сообщение об
ошибке
        }
        "id": <string> ID операции
    }
}

```

Получение сохранённых файлов изображений, полученных по whatsapp

(только в формате json)

```

{
    "method": "WhatsappMediaApi.GetImage",
    "jsonrpc": "2.0",
    "params": {
        "token": <string>
        "filename": <string> Имя файла",
    },
    "id": <string> ID операции
}

```

Формат ответа

В ответ метод отдает данные в формате json

В случае успеха:

```
{
    "jsonrpc": "2.0",
    "result": {
        "status": "ok",
        "data": {
            "success": true, <boolean>
            "media_info": {} <array> данные файла, полученные по
            getimagesize()
            "media_encoded": <string> Содержимое файла, закодированное в base64
        }
        "id": <string> ID операции
    }
}
```

В случае возникновения различного рода ошибок:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": <string> Код ошибки
        "message": <string> Сообщение об ошибке
    }
    "id": <string> ID операции
}
```