ТЕХНИЧЕСКАЯ СПЕЦИФИКАЦИЯ

**API-ИНТЕРФЕЙС**

**сервисА «Проверка чека онлайн»**

Оглавление

[1. Общие положения 3](#_Toc158671106)

[1.1. Цели и задачи документа 3](#_Toc158671107)

[1.2. Определения и сокращения 3](#_Toc158671108)

[2. Описание API-интерфейса сервиса 4](#_Toc158671109)

[2.1. Подключение 4](#_Toc158671110)

[2.2. Статистика запросов 4](#_Toc158671111)

[2.3. Форматы запросов 4](#_Toc158671112)

[2.3.1. Формат запроса 1 4](#_Toc158671113)

[2.3.2. Формат запроса 2 4](#_Toc158671114)

[2.3.3. Формат запроса 3 5](#_Toc158671115)

[2.3.4. Формат запроса 4 5](#_Toc158671116)

[2.4. Формат ответа 5](#_Toc158671117)

[2.5. Использование QR-кодов 6](#_Toc158671118)

[2.6. Пример запроса к сервису через curl 6](#_Toc158671119)

[2.7. Пример запроса к сервису на языке Python 7](#_Toc158671120)

[2.8. Пример запроса к сервису на языке PHP 7](#_Toc158671121)

[2.9. Пример запроса к сервису на языке JavaScript 8](#_Toc158671122)

[2.10. Параметр запроса API для идентификации акции 8](#_Toc158671123)

[2.11. Добавление произвольных параметров к запросу 9](#_Toc158671124)

[2.12. Виджет для запуска акции 9](#_Toc158671125)

[2.13. Использование программных библиотек и примеры подключения 9](#_Toc158671126)

1. Общие положения
   1. Цели и задачи документа

В документе представлена вся необходимая информация по использованию API-интерфейса сервиса «Проверка чека онлайн» для получения кассового чека по его реквизитам.

* 1. Определения и сокращения

Использованы термины и сокращения:

API – интерфейс - интерфейс программирования приложений, который позволяет отправлять и получать данные сервиса «Проверка чека онлайн» расположенный на веб-сайте.

БД – база данных.

Веб-сайт – источник информации, размещенный в сети Интернет.

Пользователь – лицо, заключившее договор на условиях настоящей Технической спецификации с поставщиком услуги API – интерфейса в интересах владельца Приложения Пользователя.

Приложение Пользователя — Интернет-площадка и прочие Веб-приложения (включая мобильные), принадлежащие или используемые Пользователем на законных основаниях под собственным товарным знаком или иным средством индивидуализации, с помощью которых Продукты показываются и/или доводятся до сведения Потребителей.

Потребитель — любое лицо, использующее Приложение Пользователя по функциональному назначению.

Ключ доступа — предоставляемая Пользователю уникальная комбинация знаков, ассоциированная с Идентификатором Приложения Пользователя и позволяющая обеспечить программное взаимодействие Приложения Пользователя к API – интерфейсу посредством Запросов.

Запрос — поступивший от сервера Пользователя к API – интерфейсу http-запрос, строго соответствующий указанным в Технической спецификации параметрам.

Успешный запрос – запрос на который был получен ответ от API – интерфейса.

1. Описание API-интерфейса сервиса
   1. Подключение

Для получения доступа к API-интерфейсу сервиса Пользователю необходимо направить заявку на подключение с указанием следующих сведений:

* Адрес Интернет-площадки или название Веб-приложения;
* Тематику Приложения Пользователя;
* Идентифицирующие Пользователя данные;
* Адрес электронной почты Пользователя для отправки Ключа доступа.
  1. Статистика запросов

Для получения статистики Запросов Пользователя к API-интерфейсу необходимо в личном кабинете на сайте проверка чека перейти в пункт меню «Сервис» - «Статистика». На странице выведется информация с разбивкой по месяцам о количестве успешных/неуспешных обращений к API.



* 1. Форматы запросов
     1. Формат запроса 1

POST https://proverkacheka.com/api/v1/check/get

Тип содержимого: application/json, application/x-www-form-urlencoded, multipart/form-data

Тело запроса:

token – Токен доступа

fn - Номер фискального накопителя

fd - Номер фискального документа

fp - Фискальный признак документа

t - Дата и время

n - Тип операции (Приход/Возврат прихода/Расход/Возврат расхода)

s - Сумма чека

qr – Признак сканирования qr кода (0/1)

* + 1. Формат запроса 2

POST https://proverkacheka.com/api/v1/check/get

Тип содержимого: application/json, application/x-www-form-urlencoded, multipart/form-data

Тело запроса:

qrraw - строка сырых данных QR-кода, например:

t=20200924T1837&s=349.93&fn=9282440300682838&i=46534&fp=1273019065&n=1

При передаче qrraw сервис обработает строку, закодированную в qr коде и

получит чек.

* + 1. Формат запроса 3

POST https://proverkacheka.com/api/v1/check/get

Тип содержимого: application/json, application/x-www-form-urlencoded, multipart/form-data

Тело запроса:

qrurl - ссылка на картинку QR-кода, например: https://domen.ru/image/qrimage.jpg

При передаче qrurl сервис скачает картинку, распознает по ней код и получит чек.

* + 1. Формат запроса 4

POST https://proverkacheka.com/api/v1/check/get

Тип содержимого: multipart/form-data

Тело запроса:

qrfile - содержимое файла картинки QR-кода

При передаче qrfile сервис получит картинку из запроса, распознает по

ней код и получит чек.

* 1. Формат ответа

|  |  |
| --- | --- |
| **Наименование ключа** | **Значение ключа** |
| JSON.code | КОД ОТВЕТА:  0 - чек некорректен,  1 - данные чека получены (успешный запрос),  2 - данные чека пока не получены,  3 - превышено кол-во запросов,  4 - ожидание перед повторным запросом,  5 - прочее (данные не получены) |
| JSON.data | ДАННЫЕ |
| JSON.data.user | Организация |
| JSON.data.retailPlaceAddres | Адрес |
| JSON.data.userInn | ИНН |
| JSON.data.ticketDate | Дата |
| JSON.data.requestNumber | Чек: № |
| JSON.data.shiftNumber | Смена |
| JSON.data.operator | Кассир |
| JSON.data.operationType | Тип операции:  1 - Приход  2 - Возврат прихода  3 - Расход  4 - Возврат расхода |
| JSON.data.items | ПОЗИЦИИ ТОВАРА/УСЛУГИ (массив) |
| JSON.data.items[i] | Позиция в чеке (элемент массива позиций) |
| JSON.data.items[i].name | Название товара/услуги |
| JSON.data.items[i].price | Цена |
| JSON.data.items[i].quantity | Количество |
| JSON.data.items[i].sum | Сумма (в коп.) |
| JSON.data.nds18 | НДС со ставкой 20% (сумма в коп.) |
| JSON.data.nds | НДС со ставкой 10% (сумма в коп.) |
| JSON.data.nds0 | НДС со ставкой 0% (сумма в коп.) |
| JSON.data.ndsNo | НДС не облагается (сумма в коп.) |
| JSON.data.totalSum | ИТОГО (сумма в коп.) |
| JSON.data.cashTotalSum | Наличные (сумма в коп.) |
| JSON.data.ecashTotalSum | Карта (сумма в коп.) |
| JSON.data.taxationType | ВИД НАЛОГООБЛОЖЕНИЯ:  1 - ОСН,  2 - УСН,  4 - УСН доход – расход,  8 - ЕНВД,  16 - ЕСХН,  32 - ПСН |
| JSON.data.kktRegId | РЕГ. НОМЕР ККТ |
| JSON.data.kktNumber | ЗАВОД. № |
| JSON.data.fiscalDriveNumber | ФН |
| JSON.data.fiscalDocumentNumber | ФД |
| JSON.data.fiscalSign | ФПД |

* 1. Использование QR-кодов

Содержимое QR кода кассового чека содержит строку следующего формата:

t=20190202T1044&s=476.00&fn=9289000100054082&i=16112&fp=399448105&n=1

Параметры:

t - Дата и время

s - Сумма чека

fn - Номер фискального накопителя

i - Номер фискального документа

fp - Фискальный признак документа

n - Тип операции (Приход/Возврат прихода/Расход/Возврат расхода)

* 1. Пример запроса к сервису через curl

**//**пример параметров формата запроса 1

**curl** https**://**proverkacheka.com**/**api**/**v1**/**check**/**get **--**data

"fn=9280440300770583&fd=33110&fp=4138469556&n=1&s=419.54&t=20210217T2028&qr=0&token=<здесь прописываем токен доступа>"

**//**пример параметров формата запроса 2

**curl** **--**location **--**request POST 'https://proverkacheka.com/api/v1/check/get' **\**

**--**form 'qrraw="t=20200924T1837&s=349.93&fn=9282440300682838&i=46534&fp=1273019065&n=1"' **\**

**--**form 'token="здесь прописываем токен доступа"'

**//**пример параметров формата запроса 3

**curl** **--**location **--**request POST 'https://proverkacheka.com/api/v1/check/get' **\**

**--**form 'qrurl ="https://domen.ru/image/qrimage.jpg"' **\**

**--**form 'token="здесь прописываем токен доступа"'

**//**пример параметров формата запроса 4

**curl** **--**location **--**request POST 'https://proverkacheka.com/api/v1/check/get' **\**

**--**form 'qrfile=@"путь до файла"' **\**

**--**form 'token="здесь прописываем токен доступа"'

* 1. Пример запроса к сервису на языке Python

**import** requests

url **=** 'https://proverkacheka.com/api/v1/check/get'

**//**пример параметров формата запроса 1

data**={** 'fn'**:** '9280440300770583'**,** **//**ФН

'fd' **:** '33110'**,** **//**ФД

'fp'**:** '4138469556'**,** **//**ФП

't' **:** '20210217T2028'**,** **//**время с чека

'n' **:** '1'**,** **//**вид кассового чека

's' **:** '419.54'**,** **//**сумма чека

'qr' **:** '0'**,** **//**признак сканирования QR**-**кода

'token'**:**'' **//**здесь прописываем токен доступа

**}**

**//**пример параметров формата запроса 2

data**={**'token'**:**''**,**'qrraw'**:** 't=20200924T1837&s=349.93&fn=9282440300682838&i=46534&fp=1273019065&n=1'**}**

**//**пример параметров формата запроса 3

data**={**'token'**:**''**,**'qrurl'**:** 'https://domen.ru/image/qrimage.jpg'**}**

**//**пример параметров формата запроса 4

data**={**'token'**:**''**}**

files **=** **{**'qrfile'**:** **open(**'файл-картинка'**,**'rb'**)}**

r **=** requests**.**post**(**url**,** data**=**data**,** files**=**files**)**

**print(**r**.**text**)**

* 1. Пример запроса к сервису на языке PHP

<?php

//пример параметров формата запроса 1

//параметры из формы запроса полученные от браузера клиента

$tPar=[

'fn' => '9280440300770583', //ФН

'fd' => '33110', //ФД

'fp' => '4138469556', //ФП

't' => '20210217T2028', //время с чека

'n' => '1', //вид кассового чека

's' => '419.54', //сумма чека

'qr' => '0', //признак сканирования QR-кода

'token' => '' //здесь прописываем токен доступа

];

//пример параметров формата запроса 2

$tPar=[

'qrraw' => 't=20200924T1837&s=349.93&fn=9282440300682838&i=46534&fp=1273019065&n=1', //строка сырых данных QR-кода

'token' => '' //здесь прописываем токен доступа

];

//пример параметров формата запроса 3

$tPar=[

'qrurl' => 'https://domen.ru/image/qrimage.jpg', //ссылка на картинку QR-кода

'token' => '' //здесь прописываем токен доступа

];

//пример параметров формата запроса 4

$localFile = $\_FILES["scan"]['tmp\_name'];

$filename = **basename**($localFile);

$tPar=[

'token' => '', //здесь прописываем токен доступа

'qrfile' => **curl\_file\_create**($localFile, $\_FILES['scan']['type'], $filename), //файл-картинка для обработки

];

//выполняем запрос на сервер Проверка чека используя API

$tСurl = **curl\_init**();

**curl\_setopt**($tСurl, **CURLOPT\_URL**, https://proverkacheka.com/api/v1/check/get');

**curl\_setopt**($tСurl, **CURLOPT\_RETURNTRANSFER**, **true**);

**curl\_setopt**($tСurl, **CURLOPT\_POST**, **true**);

**curl\_setopt**($tСurl, **CURLOPT\_POSTFIELDS**, $tPar);

$tRes = **curl\_exec**($tСurl);

**curl\_close**($tСurl);

?>

* 1. Пример запроса к сервису на языке JavaScript

//пример 1

**const** encodedParams **=** **new** URLSearchParams**();**

encodedParams**.**append**(**"qrraw"**,** "t=20200924T1837&s=349.93&fn=9282440300682838&i=46534&fp=1273019065&n=1"**);**

encodedParams**.**append**(**"token"**,** "<прописать токен>"**);**

**const** options **=** **{**

method**:** 'POST'**,**

headers**:** **{**

'content-type'**:** 'application/x-www-form-urlencoded'**,**

**},**

body**:** encodedParams

**};**

fetch**(**'https://proverkacheka.com/api/v1/check/get'**,** options**)**

**.**then**(**response **=>** response**.**json**())**

**.**then**(**response **=>** **{**

console**.**log**(**response**);**

**document.**querySelector**(**'.b-res'**).**innerHTML**=**response**.**data**.**html**;**

**})**

**.catch(**err **=>** console**.**error**(**err**));**

//пример 2

**const** encodedParams **=** **new** URLSearchParams**();**

encodedParams**.**append**(**"qrurl"**,** "https://i.imgur.com/IVX4DgO.jpg"**);**

encodedParams**.**append**(**"token"**,** "<прописать токен>"**);**

**const** options **=** **{**

method**:** 'POST'**,**

headers**:** **{**

'content-type'**:** 'application/x-www-form-urlencoded'**,**

**},**

body**:** encodedParams

**};**

fetch**(**'https://proverkacheka.com/api/v1/check/get'**,** options**)**

**.**then**(**response **=>** response**.**json**())**

**.**then**(**response **=>** **{**

console**.**log**(**response**);**

**document.**querySelector**(**'.b-res'**).**innerHTML**=**response**.**data**.**html**;**

**})**

**.catch(**err **=>** console**.**error**(**err**));**

//пример 3

**var** tFormData **=** **new** FormData**();**

tFormData**.**append**(**'qrfile'**,** **document.**querySelector**(**'.b-file'**).**files**[**0**]);**

tFormData**.**append**(**'token'**,** "<прописать токен>"**);**

**const** options **=** **{**

method**:** 'POST'**,**

body**:** tFormData

**};**

fetch**(**'https://proverkacheka.com/api/v1/check/get'**,** options**)**

**.**then**(**response **=>** response**.**json**())**

**.**then**(**response **=>** **{**

console**.**log**(**response**);**

**document.**querySelector**(**'.b-res'**).**innerHTML**=**response**.**data**.**html**;**

**})**

**.catch(**err **=>** console**.**error**(**err**));**

* 1. Параметр запроса API для идентификации акции

Параметр необходимо указывать, если требуется чтобы все запросы чеков через API попадали в соответствующую акцию в личном кабинете. Откуда уже можно будет выгрузить информацию или вести статистику по акции.

Спецификация по параметрам к POST запросу:

- 'promo\_id' //идентификатор акции из личного кабинета, тип - целое число

Подробнее узнать о функционале промо-акций можно написав на почту [webmaster@proverkacheka.com](mailto:webmaster@proverkacheka.com)

* 1. Добавление произвольных параметров к запросу

С отправкой чека через API возможно добавлять произвольные данные, связанные с данным чеком, для этого при запросе добавьте параметр по шаблону:

- userdata\_<param\_name> //строка данных, где <param\_name> - имя параметра, например: userdata\_name1="параметр1"

При выгрузке чеков в формате json в структуре содержится атрибут user\_data, который будет содержать добавленную информацию, для примера выше это поле "user\_data.name1".

Обратите внимание: при передаче данных userdata\_name1, а в выгрузке user\_data.name1

* 1. Виджет для запуска акции

С помощью виджета от нашего сервиса можно быстро запустить промо-акцию, для этого необходимо встроить виджет на свой сайт, построенный с помощью конструктора Tilda, Wix и других.

Все чеки, поступающие от виджета проверяются по условиям акции в режиме реального времени, т.е. после отправки чека результат получаем сразу.

После окончания акции все чеки Вы можете выгрузить в Excel и подвести итоги(розыгрыш).

Подробнее узнать о виджете можно написав нам на почту [webmaster@proverkacheka.com](mailto:webmaster@proverkacheka.com)

* 1. Использование программных библиотек и примеры подключения

По адресу: https://proverkacheka.com/example/index.php содержится:

- форма запроса реквизитов для встраивания на сайт,

- подключаемые библиотеки JS, CSS для обработки распознавания QR кода и направления параметров запроса на ваш сайт,

- пример использования интерфейса API на вашем сайте с помощью PHP CURL.