Документация по ресурсу **Fox Hustle**

Оглавление

[Общая информация: 4](#_Toc31200053)

[Права доступа 4](#_Toc31200054)

[Права доступа для токена пользователя 5](#_Toc31200055)

[Права доступа для токена бота 6](#_Toc31200056)

[Актуальные версии API 7](#_Toc31200057)

[Методы 7](#_Toc31200058)

[Аутентификация: 7](#_Toc31200059)

[Auth.Login 7](#_Toc31200060)

[Auth.Logout 8](#_Toc31200061)

[Auth.GetPermissions 9](#_Toc31200062)

[Профиль: 10](#_Toc31200063)

[Profile.Get 10](#_Toc31200064)

[Profile.GetSettings 11](#_Toc31200065)

[Profile.Subscribe 12](#_Toc31200066)

[Платформа ботов: 13](#_Toc31200067)

[Bot.getSettings 13](#_Toc31200068)

[Платежная система 14](#_Toc31200069)

[Payment.Get 14](#_Toc31200070)

[Payment.Create 15](#_Toc31200071)

[Payment.CreateCheck 16](#_Toc31200072)

[Payment.PayCheck 17](#_Toc31200073)

[Payment.AddMoney 18](#_Toc31200074)

[Список ошибок 19](#_Toc31200075)

[Описание структур объектов 19](#_Toc31200076)

[Profile 19](#_Toc31200077)

[Photo 19](#_Toc31200078)

[Message 20](#_Toc31200079)

[Wallet 20](#_Toc31200080)

[Wallet\_History 20](#_Toc31200081)

[Permission 20](#_Toc31200082)

[Check 21](#_Toc31200083)

[Ограничения 21](#_Toc31200084)

[Библиотеки 22](#_Toc31200085)

[Python 22](#_Toc31200086)

[JS 22](#_Toc31200087)

[React.js 22](#_Toc31200088)

[Java 22](#_Toc31200089)

[C++ 22](#_Toc31200090)

# Общая информация:

Три вида профиля:

* Неподтвержденный
* Расширенный
* Подтвержденный

Для каждого профиля есть свои ограничения (см. [Ограничения](#_Ограничения))

API представляет из себя набор поддерживаемых версий. Методы обращения к API могут отличаться из версии к версии.

Аутентификация происходит по протоколу, похожему на OAuth2.0. Перед авторизацией с помощью API нужно создать свое приложение в разделе *«Приложения»* на странице для разработчиков. Необходимо своему приложению придумать уникальное название. После создания приложения Вам выдастся уникальный ***id*** приложения, его ***code*** и ***service\_key***. По ***service\_key*** в дальнейшем будет осуществляться настройка приложения с помощью API.

После создания приложения, необходимо изучить настоящую документацию по методам API данного продукта.

Сервер работает по адресу <http://foxhustle.ru/>.

Все параметры должны передаваться на сервер с помощью post-запроса. Возвращающиеся значения будут в json-формате. API продукта находится по адресу <http://foxhustle.ru/api/>. Актуальную документацию к нему можно прочитать на сайте <http://foxhustle.ru/dev/>.

Все передаваемые параметры подразделяются на обязательные и необязательные, для каждого метода обращения к API свои передаваемые параметры. Обязательные должны передаваться к данному методу при каждом обращении к нему. Необязательные параметры могут не передаваться, в документации к ним добавлен символ \*.

# Права доступа

Права доступа определяют возможность использования токена для работы с тем или иным разделом данных. Так, например, для отправки личного сообщения от имени пользователя токен должен быть получен с правами ***messages***.

Получить разрешение можно только в случае, если такое разрешение возможно

## Права доступа для токена пользователя

|  |  |
| --- | --- |
| Название | Описание |
| profile  (+1)  (1 << 0) | Доступ к профилю |
| friends  (+2)  (1 << 1) | Доступ к друзьям |
| photos  (+4)  (1 << 2) | Доступ к фотографиям |
| menu  (+8)  (1 << 3) | Доступ к меню слева |
| status  (+16)  (1 << 4) | Доступ к статусу пользователя |
| notes  (+32)  (1 << 5) | Доступ к заметкам пользователя |
| messages  (+64)  (1 << 6) | Доступ к сообщениям пользователя |
| apps  (+128)  (1 << 7) | Доступ к приложениям |
| groups  (+256)  (1 << 8) | Доступ к группам |
| notifications  (+512)  (1 << 9) | Доступ к оповещениям пользователя |
| email  (+1024)  (1 << 10) | Доступ к email пользователя |
| market  (+2048)  (1 << 11) | Доступ к товарам |
| experience  (+4096)  (1 << 12) | Доступ к опыту пользователя |
| payment  (+8192)  (1 << 13) | Доступ к платежным методам |
| admin  (+16384)  (1 << 14) | Доступ к админпанели (секретно) |
| bots  (+32768)  (1 << 15) | Доступ к ботам |

## Права доступа для токена бота

|  |  |
| --- | --- |
| Название | Описание |
| profile  (+1)  (1 << 0) | Доступ к профилям |
| photos  (+4)  (1 << 2) | Доступ к фотографиям |
| notes  (+32)  (1 << 5) | Доступ к заметкам пользователя |
| messages  (+64)  (1 << 6) | Доступ к сообщениям пользователя |
| groups  (+256)  (1 << 8) | Доступ к группам |
| notifications  (+512)  (1 << 9) | Доступ к оповещениям пользователя |
| email  (+1024)  (1 << 10) | Доступ к email пользователя |
| market  (+2048)  (1 << 11) | Доступ к товарам |
| experience  (+4096)  (1 << 12) | Доступ к опыту пользователя |
| payment  (+8192)  (1 << 13) | Доступ к платежным методам |
| bots  (+32768)  (1 << 15) | Доступ к ботам |

# Актуальные версии API

Здесь появляются актуальные версии FoxHustle API, а также даты создания и прекращения поддержки. Приложения должны поддерживать актуальную версию API.

|  |  |  |
| --- | --- | --- |
| Версия | Дата создания | Дата прекращения поддержки |
| 1.0 | 1 февраля 2020 | 1 февраля 2021 |
| 1.1 | 1 июня 2020 | 1 июня 2021 |

# Методы

## Аутентификация:

### Auth.Login

С помощью данного метода можно авторизоваться в приложении, используя логин с паролем от пользователя. При успешной аутентификации вернутся 2 параметра: access\_token и permissions, а при неудачном входе – error и description\_error. Access\_token при долгом неиспользовании будет обнулен.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| code | Код приложения, через которое осуществляется аутентификация, Str |
| login | Имя пользователя, Str |
| password | Пароль пользователя, Str |
| scope | Права доступа, необходимые для работы приложения (см. [Права доступа](#_Права_доступа)), Int |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| access\_token | Уникальный токен пользователя, Str |
| permissions | Доступы токена, Permission |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘access\_code’: ‘asdgu1igv3u1y2e’,  ‘permissions’: {  ‘profile’: 1,  ‘friends’: 1,  ‘apps’: 1  }  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Auth.Logout

С помощью данного метода можно деавторизоваться в приложении. При успешной деаутентификации вернется 1, а при неудачной попытке – error и description\_error. Access\_token в этом случае будет обнулен.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| code | Код приложения, через которое осуществляется аутентификация , Str |
| access\_token | Уникальный токен пользователя, Str |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: 1  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Auth.GetPermissions

С помощью данного метода можно получить список разрешений для токена.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, Str |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| permissions | Доступы токена, Permission |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘permissions’: {  ‘profile’: 1,  ‘friends’: 1,  ‘apps’: 1  }  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Auth.Register

С помощью данного метода можно зарегистрироваться.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| Code | Код приложения, Str |
| Nickname\* | Короткое имя пользователя, Str |
| Surname\* | Фамилия пользователя, Str |
| Name\* | Имя пользователя, Str |
| Email\* | Почта пользователя, Str |
| VK\_TOKEN\* | Токен пользователя ВКонтакте, происходит импортирование необходимых для регистрации данных из ВКонтакте. Должен иметь разрешение на получение информации о профиле, email |
| TELEGRAM\_TOKEN\* | Токен пользователя Telegram, происходит импортирование необходимых для регистрации данных из Telegram. Должен иметь разрешение на получение информации о профиле. Необходимо указать email пользователя |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘permissions’: {  ‘profile’: 1,  ‘friends’: 1,  ‘apps’: 1  }  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

## Профиль:

### Profile.Get

С помощью данного метода можно получить список разрешений для токена.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, Str |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| permissions\* | Доступы токена, Permission |
| error\* | Номер ошибки, Int |
| description\_error\* | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘permissions’: {  ‘profile’: 1,  ‘friends’: 1,  ‘apps’: 1  }  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Profile.GetSettings

С помощью данного метода можно получить список настроек для профиля.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, Str |
| v | Версия API, Str |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| settings | Доступы токена, Settings |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘settings: {  ‘privacy’: {  ‘sendMessage’: 0,  ‘subscribe’: 1,  ‘getWall’: 1,  ‘sendWall’: 0,  ‘private’: 0,  ‘photos’: 1,  ‘seeFriends’: 1,  ‘seeCommentsWall’: 0,  ‘sendCommentsWall’: 0,  ‘seeCommentsPhoto’: 0,  ‘sendCommentsWall’: 0,  ‘seeProfile’: 1,  ‘sendAdd’: 0  }  }  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Profile.Subscribe

С помощью данного метода можно подписаться на профиль.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, Str |
| v | Версия API, Str |
| id | Id пользователя, на новости которого надо подписаться, Int |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: 1  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

## Платформа ботов:

### Bot.getSettings

С помощью данного метода можно получить список настроек для профиля.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, Str |
| v | Версия API, Str |
| id | Id бота, Int |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| settings | Доступы токена, Settings |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘settings: {  ‘privacy’: {  ‘sendMessage’: 0,  ‘subscribe’: 1,  ‘getWall’: 1,  ‘sendWall’: 0,  ‘private’: 0,  ‘photos’: 1,  ‘seeFriends’: 1,  ‘seeCommentsWall’: 0,  ‘sendCommentsWall’: 0,  ‘seeCommentsPhoto’: 0,  ‘sendCommentsWall’: 0,  ‘seeProfile’: 1,  ‘sendAdd’: 0  }  }  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

## Платежная система

### Payment.Get

С помощью данного метода можно получить список кошельков и баланс на них.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, Str |
| v | Версия API, Str |
| Code | Код приложения, дается при создании приложения (см. [Общая информация](#_Общая_информация:)), Str |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| Wallets | Список кошельков, Wallet |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: [  {  ‘id’: 73418,  ‘currently’: ‘RUB’,  ‘value’: 3451.4,  ‘default: 0  }  ]  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Payment.Create

С помощью данного метода можно создать кошелек.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, String |
| v | Версия API, String |
| Code | Код приложения, дается при создании приложения (см. [Общая информация](#_Общая_информация:)) , String |
| Currently\* | Валюта создаваемого кошелька. Если этот параметр не передается, берется значение ‘KUS’, String |
| Default\* | Параметр, указывающий делать кошелек стандартным для списания. Если этот параметр не передается, берется значение False, Boolean |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, Тип данных |
| Wallet | Информация о кошельке, Wallet |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘id’: 73418,  ‘currently’: ‘RUB’,  ‘value’: 3451.4,  ‘default: 0  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Payment.CreateCheck

С помощью данного метода можно отправить чек на оплату.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, String |
| v | Версия API, String |
| Code | Код приложения, дается при создании приложения (см. [Общая информация](#_Общая_информация:)) , String |
| Currently\* | Валюта создаваемого чека. Если этот параметр не передается, берется значение ‘KUS’, String |
| CHECK\_HOLDER | Чекодержатель, Str |
| VALUE | Сумма, Float |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, Тип данных |
| Check | Информация о чеке, Check |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: {  ‘id’: 73418,  ‘currently’: ‘RUB’,  ‘value’: 3451.4,  ‘default: 0  }  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Payment.PayCheck

С помощью данного метода можно оплатить чек.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, String |
| v | Версия API, String |
| Code | Код приложения, дается при создании приложения (см. [Общая информация](#_Общая_информация:)) , String |
| CHECK\_HOLDER | Чекодержатель, Str |
| WALLET\_ID\* | Id кошелька(если не передается, оплата происходит с кошелька, указанного по умолчанию), Int |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, Тип данных |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: 1  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

### Payment.AddMoney

С помощью данного метода можно добавить деньги на счет.

Входящие параметры:

|  |  |
| --- | --- |
| Параметры | Описание, тип данных |
| app\_id | Идентификатор приложения, через которое осуществляется аутентификация, Int |
| access\_token | Уникальный токен пользователя, String |
| v | Версия API, String |
| Code | Код приложения, дается при создании приложения (см. [Общая информация](#_Общая_информация:)) , String |
| Currently\* | Валюта создаваемого чека. Если этот параметр не передается, берется значение ‘KUS’, String |
| CHECK\_HOLDER | Чекодержатель, Str |
| WALLET | Id кошелька, Int |

Возвращаемые параметры:

|  |  |
| --- | --- |
| Параметры | Описание, Тип данных |
| error | Номер ошибки, Int |
| description\_error | Описание ошибки, Str |

Пример возвращаемых параметров:

|  |
| --- |
| {  ‘response’: 1  } |
| {  ‘error’: 4,  ‘description\_error’: ‘System error. Please, repeat later’  } |

# Список ошибок

Здесь показаны все виды ошибок в API. Этот список всегда будет изменяться с выходом новых и прекращением поддержки версий API.

|  |  |
| --- | --- |
| Номер ошибки | Описание ошибки |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 | System error |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |

# Описание структур объектов

## Profile

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| Id | Id профиля, Int |
| Surname | Фамилия, Str |
| Name | Имя, Str |
| Middle\_name | Отчество, Str |
| Photo | Главное фото, Photo |
| Type | Тип профиля, Int |
| Verifery | Верификация, Boolean |

## Photo

|  |  |
| --- | --- |
| Название | Описание, тип данных |
| Id | Номер фотографии, Int |
| From\_id | Id автора фотографии, Int |
| Url | URL фотографии, Str |
| Key | Key фотографии (секретный ключ фотографии), Str |

## Message

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| id | Номер сообщения, Int |
| From\_id | Id автора сообщения, Int |
| Peer\_id | Номер диалога, Int |
| Title | Заголовок сообщения (доступен от 100 символов поля ‘Text’), Str |
| Text | Текст сообщения, Str |
| Attachments\* | Вложения (макс. 3), Str |

## Wallet

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| Id | Номер кошелька, уникальный для каждого кошелька, Int |
| Currency | Название валюты, Str |
| Value | Значение кошелька, Float |
| Default | Стандартный кошелек для валюты, Boolean |

## Wallet\_History

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| Id | Номер кошелька, уникальный для каждого кошелька, Int |
| Currency | Название валюты, Str |
| Value | Значение изменения, Float |
| Date | Дата изменения, Date |

## Permission

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| profile | Доступ к профилю, Boolean |
| friends | Доступ к друзьям, Boolean |
| photos | Доступ к фотографиям, Boolean |
| menu | Доступ к меню слева, Boolean |
| status | Доступ к статусу пользователя, Boolean |
| notes | Доступ к заметкам пользователя, Boolean |
| messages | Доступ к сообщениям пользователя, Boolean |
| apps | Доступ к приложениям, Boolean |
| groups | Доступ к группам, Boolean |
| notifications | Доступ к оповещениям пользователя, Boolean |
| email | Доступ к email пользователя, Boolean |
| market | Доступ к товарам, Boolean |
| experience | Доступ к опыту пользователя, Boolean |
| payment | Доступ к платежным методам, Boolean |
| admin | Доступ к админпанели (секретно) , Boolean |
| bots | Доступ к ботам, Boolean |

## Check

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| ID | Id чека, Int |
| DRAWER | Чекодатель, Str |
| CHECK\_HOLDER | Чекодержатель, Str |
| VALUE | Сумма, Float |
| CREATE\_DATE | Дата выставления счета, Date |

## Chat

|  |  |
| --- | --- |
| Название | Описание, Тип данных |
| ID | Id чата, Int |
| USER\_ID | Получатель, User |
| UNREAD\_COUNT | Количество непрочитанных сообщений, Int |
| TIME | Время отправления последнего сообщения в диалоге/беседе, Date |
| MESSAGES | Последнее сообщение, Message |

# Ограничения

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр \Тип профиля | Неподтвержденный | Расширенный | Подтвержденный |
| Записи *(в час)* | 2 | 4 | ∞ |
| Комментарии *(в час)* | 2 | 10 | ∞ |
| Кошельки | 2 | 3 | 10 |
| Опыт *(в день)* | 100 | 200 | 1000 |
| Заработок *(в час)* | 0.01$ | 0.1$ | 10$ |
| Переводы *(в день)* | 1$ | 10$ | 1000$ |

Ограничения по создаваемым кошелькам:

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр\Тип профиля | Неподтвержденный | Расширенный | Подтвержденный |
| Максимальное значение | 10$ | 100$ | 1000$ |
| Валюты | RUB, KUS | RUB, USD, KUS | RUB, EUR, USD, KUS |

# Библиотеки

## Python

Установить актуальную версию библиотеки FoxHustle API можно с помощью команды «pip install FoxHustleAPI».

Пример:

## JS

Пример:

## React.js

Пример:

## Java

Пример:

## C++

Пример: