Private API

Trademen also provides private HTTP APIs which allow read / write access to your private account using personal access token for following functionalities..

```
 - Open Order List
- Place Order
- Cancel Order
- Get System's Availale Payment Methods
- Get user's Bank Accounts
- Deposit List
- Deposit Fiat Currency
- Upload Fiat Deposit Receipt for Admin Review
- Deposit Cryptocurrency
- Withdrawal List
- Withdraw Wallet Balance
- Cancel Bank Withdrawal Request
```

Open Order List

Get order list for a given coinPair or market. Required GET method and Required URL parameter is currency symbol. If successful, the method will return a success message with 200 HTTP.

`Sample Request for Deposit List`

```
curl -v -X GET https://yourdomain.com/api/user/orders/BTC_USD/open \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

`Sample Successful Response` - status code - 200

```
{
    "success": true,
    "data": []
}
```

`Sample Error Response` - stataus code - 401

```
{
    "message": "Unauthenticated."
}
```

Place Order

Places a buy or sell order for a category (Limit, Stop Limit, Market) in a given market. Required POST method and the parameters are "order_type", "category", "trade_pair", "price", "amount", "total" and "stop". If successful, the method will return the order number.

- Place Order Endpoint: [POST] https://yourdomain.com/api/user/orders/place

Request Parameter	Description
order_type	`Required` . Available order types are buy, sell
category	`Required` . Available categories are limit, stop_limt, market
trade_pair	`Required` . A pair like BTC_USD
price	`Required if order category is not market` . Numerice between 0.000000001, 99999999999.99999999
amount	`Required if order category is not market and order type is not sell` . Numerice between 0.000000001, 99999999999.99999999
total	`Required if order category is not market and order type is not sell` . Numerice between 0.000000001, 99999999999.99999999

Request Parameter	Description
stop	`Required if order category is stop_limit .` Numerice between 0.000000001, 99999999999.99999999

Sample Limit Order Request

```
curl -v -X POST https://yourdomain.com/api/user/orders/place \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-d '{
  "order_type": "sell",
  "category": "limit",
  "trade_pair": "BTC_USD",
  "price": "7500.00000000",
  "amount": "0.5",
  "total": "3750.00000000",
}'
```

Sample Limit order Successful Response - status code - 201

```
{
  "success": true,
  "message": "Your order has been placed successfully.",
  "data": {
        "order_id": "624200fa-47f2-4e66-b02f-49a2d0f5db47",
        "order_type": "sell",
        "price": "7500.00000000",
        "amount": "0.50000000",
        "total": "3750.00000000",
        "open_amount": "0.50000000",
        "exchanged": "0",
        "stop_limit": null,
        "date": "2020-05-21 22:00:00",
        "category": "limit"
  }
}
```

Sample Market Order Request

```
curl -v -X POST https://yourdomain.com/api/user/orders/place \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-d '{
  "order_type": "buy",
  "category": "market",
  "trade_pair": "BTC_USD",
  "amount": "0.5",
}'
```

Sample Market Order Successful Response - status code - 201

```
{
  "success": true,
  "message": "Your order has been placed successfully.",
  "data": {
        "order_id": "624200fa-47f2-4e66-b02f-49a2d0f5db47",
        "order_type": "buy",
        "price": "0",
        "amount": "0.50000000",
        "total": "0",
        "open_amount": "0.50000000",
        "exchanged": "0",
        "stop_limit": null,
        "date": "2020-05-21 22:00:00",
        "category": "market"
  }
}
```

Sample Stop Limit Order Request

```
curl -v -X POST https://yourdomain.com/api/user/orders/place \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-d '{
  "order_type": "buy",
  "category": "stop_limit",
  "trade_pair": "BTC_USD",
  "price": "7500.00000000",
  "amount": "0.5",
  "total": "3750.00000000",
  "stop": "7600.00000000"
}'
```

Sample Successful Response - status code - 201

```
{
  "success": true,
  "message": "Your order has been placed successfully.",
  "data": {
        "order_id": "624200fa-47f2-4e66-b02f-49a2d0f5db47",
        "order_type": "buy",
        "price": "7500.00000000",
        "amount": "0.50000000",
        "total": "3750.00000000",
        "open_amount": "0.50000000",
        "exchanged": "0",
        "stop_limit": "7600.00000000",
        "date": "2020-05-21 22:00:00",
        "category": "stop_limit"
  }
}
```

Sample Error Response

```
{
  "success": false,
  "message": "Failed to create order."
}
```

Cancel Order

places an order (both buy and sell) cancel request for a any category (Limit, Stop Limit, Market) in a given market. Required DELETE method and the required

parameter is the order ID. If successful, the method will return a success message with 200 HTTP status.

Sample Cancel Order Request

```
curl -v -X DELETE https://yourdomain.com/api/user/orders/624200fa-47f2-4e66-b02f-49a2d0f5db47/destroy \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

Sample Successful Response - status code - 200

```
{
  "success": true,
  "message": "The order cancellation request has been placed successfully."
}
```

Sample Error Response - stataus code - 400

```
{
  "success": false,
  "message": "Failed to cancel the order."
}
```

Get System's Availale Payment Methods

Gets System's Availale Payment Methods for a given currency. Required GET method and url parameter is currency symbol . If successful, the method will return a success message and a list of available payment methods and bank information with 200 HTTP status.

Sample Get System Available Payment Method Request

```
curl -v -X GET https://yourdomain.com/api/coins/USD/payment-methods \
-H "Content-Type: application/json"
```

Sample Successful Response - status code - 200

```
{
    "success": true,
    "data": {
        "availablePaymentMethods": {
            "BankApi": "Bank"
        },
        "selectedSystemBanks": [
            {
                "id": "e062682a-6d4f-4809-bad3-0327fc09cc0b",
                "country": "Yemen",
                "bankName": "Graham and Sons",
                "iban": "SE2508297029259183706664",
                "swift": "BYNLEP58",
                "referenceNumber": "953987",
                "accountHolder": "Robbie Kunde V",
                "bankAddress": "226 Hildegard Gardens Suite 687\nGladysside, WI 89356-4682",
                "accountHolderAddress": "578 Tomasa Radial\nLake Gay, KS 22988",
                "isActive": 1
            }
        ]
    }
}
```

Sample Error Response - stataus code - 404

```
{
    "success": false,
    "message": "The relevant data is not found."
}
```

Get user's Bank Accounts

Gets user's bank accounts. Required GET method. If successful, the method will return a success message and a list of user's bank account information with 200 HTTP status.

`Sample Get User Bank Accounts Request`

```
curl -v -X GET https://yourdomain.com/api/user/bank-accounts \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

`Sample Successful Response` - status code - 200

```
{
    "success": true,
    "data": [
        {
            "id": "37d0e460-9813-4fc8-8e6c-7658e5dbd1ba",
            "name": "Pagac and Sons",
            "iban": "Pagac and Sons",
            "swift": "QOQLZUKP5JT",
            "bankAddress": "4463 Jones Burgs\nPort Lonniefort, OH 75119-9136",
            "referenceNumber": "49093775212",
            "accountHolderAddress": "64284 Annabel Course\nSouth Joan, MT 89057-3689",
            "isVerified": "Verified",
            "isActive": "Active",
            "createdAt": "2020-10-27T09:26:30.000000Z"
        },
        {
            "id": "9a751cd5-a0a4-407c-aa0b-8dbe9b19ebce",
            "name": "Haag Inc",
            "iban": "Haag Inc",
            "swift": "JKGRTU67TMA",
            "bankAddress": "1040 Joshua Springs\nWest Kiaraview, NJ 26021-2701",
            "referenceNumber": "7192044944693",
            "accountHolderAddress": "88515 Janie Fork\nWest Brenden, NM 46622",
            "isVerified": "Verified",
            "isActive": "Active",
            "createdAt": "2020-10-27T09:26:30.000000Z"
        }
    ]
}
```

`Sample Error Response` - stataus code - 401

```
{
    "message": "Unauthenticated."
}
```

Deposit List

Get deposit list for given wallet / currency. Required GET method and Required URL parameter is currency symbol. If successful, the method will return a success message with 200 HTTP.

`Sample Request for Deposit List`

```
curl -v -X GET https://yourdomain.com/api/user/wallets/USD/deposits \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

`Sample Successful Response` - status code - 200

```
{
    "success": true,
    "data": {
```

```json
        "current_page": 1,
        "data": [
            {
                "id": "a591a65c-6332-49a1-ae53-e5297726a0ff",
                "user_id": "658532d7-3aa2-4e3c-bb30-4d2b60a61440",
                "wallet_id": "ec048965-2206-4680-9b11-8aaba17ca9fc",
                "bank_account_id": "37d0e460-9813-4fc8-8e6c-7658e5dbd1ba",
                "symbol": "USD",
                "system_bank_account_id": null,
                "address": null,
                "amount": "200.00000000",
                "system_fee": "0.00000000",
                "txn_id": null,
                "api": "BankApi",
                "receipt": null,
                "status": "pending",
                "created_at": "2020-10-31T17:35:09.000000Z",
                "updated_at": "2020-10-31T17:35:09.000000Z",
                "bank_account": {
                    "id": "37d0e460-9813-4fc8-8e6c-7658e5dbd1ba",
                    "user_id": "658532d7-3aa2-4e3c-bb30-4d2b60a61440",
                    "country_id": 190,
                    "bank_name": "Pagac and Sons",
                    "iban": "LV55ZSQZZ682H91BID460",
                    "swift": "QOQLZUKP5JT",
                    "reference_number": "49093775212",
                    "account_holder": "Prof. Axel Ebert",
                    "bank_address": "4463 Jones Burgs\nPort Lonniefort, OH 75119-9136",
                    "account_holder_address": "64284 Annabel Course\nSouth Joan, MT 89057-3689",
                    "is_verified": 1,
                    "is_active": 1,
                    "created_at": "2020-10-27T09:26:30.000000Z",
                    "updated_at": "2020-10-27T09:26:30.000000Z"
                }
            },
            ...
        ],
        "first_page_url": "http://trademen.local/api/user/wallets/USD/deposits?page=1",
        "from": 1,
        "last_page": 1,
        "last_page_url": "http://trademen.local/api/user/wallets/USD/deposits?page=1",
        "links": [
            {
                "url": null,
                "label": "&laquo; Previous",
                "active": false
            },
            {
                "url": "http://trademen.local/api/user/wallets/USD/deposits?page=1",
                "label": 1,
                "active": true
            },
            {
                "url": null,
                "label": "Next &raquo;",
                "active": false
            }
        ],
        "next_page_url": null,
        "path": "http://trademen.local/api/user/wallets/USD/deposits",
        "per_page": 15,
        "prev_page_url": null,
        "to": 12,
        "total": 12
    }
`
```

```
}
```

Sample Error Response - stataus code - 401

```
{
    "message": "Unauthenticated."
}
```

Get Crypto Wallet Deposit Address

Gets wallet deposit address for cryptocurrency or crypto wallet. Required GET method and url parameter is cryptocurrency symbol. If successful, the method will return a wallet address and qrcode with 200 HTTP status.

Sample Get Crypto Wallet Deposit Address Request

```
curl -v -X GET https://yourdomain.com/api/user/wallets/BTC/get-deposit-address \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

Sample Successful Response - status code - 200

```
{
    "success": true,
    "data": {
        "walletAddress": "0x7d0e46098134fc88e6c7658e5dbd1ba",
        "qrCode": ...
    }
}
```

Sample Error Response - stataus code - 400

```
{
    "success": false,
    "message": "Network Error! Unable to generate address."
}
```

Deposit Fiat Currency

Deposits fiat currency for given wallet. Required POST method and parameters are "amount", "api", "bank_account_id" (If api/payment method is bank), deposit_policy. If successful, the method will return a success message with 200 HTTP status comprising deposit details, user bank detail and system bank detail for user to send the amount to that system bank. Sample Get Crypto Wallet Deposit Address Request

```
curl -v -X POST https://yourdomain.com/api/user/wallets/USD/deposits/store \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-d '{
  "amount": "200.00000000",
  "api": "BankApi",
  "bank_account_id": "724200fa-47f2-4e66-b02f-49a2d0f5db47",
  "deposit_policy": 1
}'
```

Request Parameter	Description
amount	`Required` . The amount user wants to deposit
api	`Required` . The available payment methods for the given currency. To get available payment methods see `Get System's Availale Payment Methods` API
bank_account_id	`Required if api/payment method is BANK` . The user bank account id by which he/she want to deposit. To get user bank account's id see `Get user's Bank Accounts` API
deposit_policy	`Required` . Accepted value is 1

Sample Successful Response - status code - 200

```json
{
    "success": true,
    "message": "Deposit has been created successfully.",
    "data": {
        "depositDetails": {
            "id": "a591a65c-6332-49a1-ae53-e5297726a0ff",
            "user": "Brett Jacobson",
            "wallet": "United States Dollar (USD)",
            "amount": "200",
            "bank": "Pagac and Sons",
            "txnId": null,
            "status": "pending"
        },
        "userBankDetail": {
            "id": "37d0e460-9813-4fc8-8e6c-7658e5dbd1ba",
            "bankName": "Pagac and Sons",
            "bankAddress": "4463 Jones Burgs\nPort Lonniefort, OH 75119-9136",
            "accountHolder": "Prof. Axel Ebert",
            "referenceNumber": "49093775212",
            "swift": "QOQLZUKP5JT",
            "iban": "LV55ZSQZZ682H91BID460",
            "country": "Sao Tome and Principe",
            "isActive": "Active",
            "isVerified": "Verified"
        },
        "depositWithBanks": [
            {
                "id": "e062682a-6d4f-4809-bad3-0327fc09cc0b",
                "country": "Yemen",
                "bankName": "Graham and Sons",
                "iban": "SE2508297029259183706664",
                "swift": "BYNLEP58",
                "referenceNumber": "953987",
                "accountHolder": "Robbie Kunde V",
                "bankAddress": "226 Hildegard Gardens Suite 687\nGladysside, WI 89356-4682",
                "accountHolderAddress": "578 Tomasa Radial\nLake Gay, KS 22988",
                "isActive": "Active"
            }
        ]
    }
}
```

Sample Error Response - stataus code - 400

```json
{
    "success": false,
    "message": "Invalid fiat deposit request."
}
```

Upload Bank Deposit Receipt

Uploads user bank deposit receipt in system bank. Required POST method and URL parameters are currency symbol and deposit id and request form data are "system_bank_id", "receipt" (image file). If successful, the method will return a success message with 200 HTTP status.

Request Parameter	Description
system_bank_id	`Required` . The relevant system bank account id
receipt	`Required` . The image file (JPEG,JPG,PNG) of deposit bank receipt

`Sample Get Crypto Wallet Deposit Address Request`

```
curl -v -X POST https://yourdomain.com/api/user/wallets/USD/deposits/824200fa-47f2-4e66-b02f-49a2d0f5db47/upload-bank-receipt \
-H "Content-Type: multipart/form-data" \
-H "Authorization: Bearer Access-Token" \
-F "system_bank_id=824200fa-47f2-4e66-b02f-49a2d0f5db47" \
-F "receipt=@/home/user1/Desktop/receipt.jpg"
```

Sample Successful Response - status code - 200

```
{
    "success": true,
    "message": "Receipt has been uploaded successfully."
}
```

Sample Error Response - stataus code - 400

```
{
    "success": false,
    "message": "Failed to upload receipt."
}
```

Withdrawal List

Get withdrawal list for given wallet / currency. Required GET method and Required URL parameter is currency symbol. If successful, the method will return a success message with 200 HTTP.

Sample Withdrawal Request for Withdrawal List

```
curl -v -X GET https://yourdomain.com/api/user/wallets/USD/withdrawals \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

Sample Successful Response - status code - 200

```
{
    "success": true,
    "data": {
        "current_page": 1,
        "data": [
            {
                "id": "086aa961-41aa-403e-a2de-5c5b9f78115e",
                "user_id": "658532d7-3aa2-4e3c-bb30-4d2b60a61440",
                "wallet_id": "ec048965-2206-4680-9b11-8aaba17ca9fc",
                "bank_account_id": "37d0e460-9813-4fc8-8e6c-7658e5dbd1ba",
                "symbol": "USD",
                "address": null,
                "amount": "1443.49232849",
                "system_fee": "28.86984656",
                "txn_id": "5d04d0ae-5f31-4d9e-b3a6-15499d72cfa8",
                "api": "BankApi",
                "status": "completed",
                "created_at": "2020-10-13T17:44:01.000000Z",
                "updated_at": "2020-10-04T00:01:52.000000Z",
                "bank_account": {
                    "id": "37d0e460-9813-4fc8-8e6c-7658e5dbd1ba",
                    "user_id": "658532d7-3aa2-4e3c-bb30-4d2b60a61440",
                    "country_id": 190,
                    "bank_name": "Pagac and Sons",
                    "iban": "LV55ZSQZZ682H91BID460",
                    "swift": "QOQLZUKP5JT",
                    "reference_number": "49093775212",
                    "account_holder": "Prof. Axel Ebert",
                    "bank_address": "4463 Jones Burgs\nPort Lonniefort, OH 75119-9136",
                    "account_holder_address": "64284 Annabel Course\nSouth Joan, MT 89057-3689",
                    "is_verified": 1,
```

```
                "is_active": 1,
                "created_at": "2020-10-27T09:26:30.000000Z",
                "updated_at": "2020-10-27T09:26:30.000000Z"
            }
        },
        ...
    ],
    "first_page_url": "http://trademen.local/api/user/wallets/USD/withdrawals?page=1",
    "from": 1,
    "last_page": 1,
    "last_page_url": "http://trademen.local/api/user/wallets/USD/withdrawals?page=1",
    "links": [
        {
            "url": null,
            "label": "&laquo; Previous",
            "active": false
        },
        {
            "url": "http://trademen.local/api/user/wallets/USD/withdrawals?page=1",
            "label": 1,
            "active": true
        },
        {
            "url": null,
            "label": "Next &raquo;",
            "active": false
        }
    ],
    "next_page_url": null,
    "path": "http://trademen.local/api/user/wallets/USD/withdrawals",
    "per_page": 15,
    "prev_page_url": null,
    "to": 4,
    "total": 4
    }
}
```

`Sample Error Response` - stataus code - 401

```
{
    "message": "Unauthenticated."
}
```

Withdraw Balance

Places withdraw request for given wallet / currency. Required POST method and Required URL parameter is currency symbol and request form data are "amount", "withdrawal_policy", "address" (required if the currency is crypto), "api" (required if the currency is fiat), "bank_account_id" (user bank account id is required if the api value is Bank) . If successful, the method will return a success message with 200 HTTP.

Request Parameter	Description
amount	`Required` . The amount user wants to withdraw
withdrawal_policy	`Required` . Accepted value is 1
address	`Required if currency is crypto` . Relevant crypto wallet address which receive the withdrawal amount
api	`Required if the currency is fiat` . The available payment methods for the given currency. To get available payment methods see `Get System's Availale Payment Methods` API
bank_account_id	`Required if api/payment method is BANK` . The user bank account id by which he/she want to receive the withdrawal amount. To get user bank account's id see `Get user's Bank Accounts` API

`Sample Withdrawal Request for Crypto Wallet`

```
curl -v -X POST https://yourdomain.com/api/user/wallets/BTC/withdrawals/store \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-d '{
  "amount": "0.50000000",
  "withdrawal_policy": 1,
  "address": "0x24200fa47f24e66b02f49a2d0f5db47"
}'
```

Sample Withdrawal Request for Fiat Wallet

```
curl -v -X POST https://yourdomain.com/api/user/wallets/USD/withdrawals/store \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-d '{
  "amount": "0.50000000",
  "withdrawal_policy": 1,
  "api": "BankApi",
  "bank_account_id": "724200fa-47f2-4e66-b02f-49a2d0f5db47"
}'
```

Sample Successful Response - status code - 200

```
{
    "success": true,
    "message": "Your withdrawal has been placed successfully."
}
```

Sample Error Response - stataus code - 400

```
{
    "success": false,
    "message": "Unable to withdraw amount."
}
```

Cancel Withdrawal

places a bank withdrawal cancel request for fiat currencies. Required DELETE method and the required url parameter is the currency symbol and withdrawal ID. If successful, the method will return a success message with 200 HTTP status.

Sample Cancel Withdrawal Request

```
curl -v -X DELETE https://yourdomain.com/api/user/wallets/USD/withdrawals/624200fa-47f2-4e66-b02f-49a2d0f5db47/destroy \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token"
```

Sample Successful Response - status code - 200

```
{
  "success": true,
  "message": "The withdrawal cancellation will be processed shortly."
}
```

Sample Error Response - stataus code - 400

```
{
  "success": false,
  "message": "Failed to cancel the withdrawal."
}
```