UDPN All-in-One Digital Currency Sandbox OpenApi

Description：

Tokenized Deposit Management System Api

HOST:10.0.7.87:8000

Contact:

Version:1.0

api url： /v2/api-docs

APP Approve

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params：

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
```

```json
     "data": {
         "chainId": 0,
         "networkId": 0,
         "rawTransaction": {
             "data": "",
             "eip1559Transaction": true,
             "feeCap": 0,
             "gasLimit": 0,
             "gasPremium": 0,
             "gasPrice": 0,
             "legacyTransaction": true,
             "nonce": 0,
             "to": "",
             "value": 0
         }
     },
     "message": ""
 }
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x1111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```json
{
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
```

```
                "availableBalance": 0,
                "state": 0,
                "walletId": 0
            }
        ],
        "toWalletId": 0,
        "walletName": ""
    }
    ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoin	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoin	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Approve--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method: `POST`

consumes: ["application/json"]

produces: ["*/*"]

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Captcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		

Payment--Check payment password

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Setting--Forget payment password		

note:Forget payment password

url: `/v1/app/paymentSetup/forgetPayPassword/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params:

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

Tokenized Deposit/Stablecoin--Top up

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "bankCard": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params:

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
bankCard	bank card	body	true	string	
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	orderId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount：Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5：success 6：failed other：processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin-- Query wallet information		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method：`POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params：

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Encapsulated withdrawal transaction		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```
{
    "code": 0,
```

```json
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Send withdrawal transaction		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "bankCard": "42424242424242420000",
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
bankCard	bank card	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Query transaction by hash		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
        "txHash": "",
        "txTime": "",
        "type": 0,
        "walletAddress": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Transaction list		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
```

```
      "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Encapsulated transfer transaction		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method：`POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params：

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin-- Send transfer transaction		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method: `POST`

consumes: ["application/json"]

produces: ["*/*"]

Request Example:

```
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description	type	schema
createTime	created time	string(date-time)	
email	email	string	
expire	Tokenized Deposit/Stablecoin expire time	integer(int64)	
iconPath	icon path	string	
kycType	kyc type	integer(int32)	
token	token	string	
userName	user name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

Wallet--Cancel wallet

note:Cancel wallet

url: `/v1/app/userWallet/cancelWallet`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
cancelWalletReqVO	cancelWalletReqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--check result,legal currency		

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Create wallet		

note:Create wallet

url: `/v1/app/userWallet/create`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Wallet detail		

note:Wallet detail

url: `/v1/app/userwallet/detail`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
```

```
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Currency Symbol		

note:Currency Symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Check Tokenized Deposit/Stablecoin Price		

note:Check Tokenized Deposit/Stablecoin Price

url: `/v1/app/userWallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1
}
```

Request Params:

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example:

```
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	number	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Tokenized Deposit/Stablecoin--Stablecoin symbol		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userWallet/getStablecoinSymbol`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```json
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Import wallet		

note: Import wallet

url: `/v1/app/userwallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Wallet list		

note:Wallet list

url: `/v1/app/userwallet/list`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0：Anonymous 1：Regular 2：Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0：In Progress 1：Active 2：Frozen 3：Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Wallet balance		

note:Wallet balance

url: `/v1/app/userwallet/totalAmount`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```json
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Transaction detail		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankCard": 1,
            "bankCardName": "China bank",
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
```

```
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankCard	bank card	string	
bankCardName	bank card name	string	
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	number	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 3: Pending Submission 4: Submitting 5: Submitted 6: Failed to Submit	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 3: Pending Submission 4: Submitting 5: Submitted 6: Failed to Submit	integer(int32)	
transferType	transfer type 1：Transfer out 2: Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Wallet--Transaction record		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "bankCardName": "China Bank",
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
bankCardName	bank card name	string	
businessCode	bank card number if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 3: Pending Submission 4: Submitting 5:Submitted 6:Failed to Submit	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Bank Card		

Bank--Bank Card List

note:Bank card list

url: `/v1/bankCard/getUserBankAccountList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "balance": "",
            "bankCard": "",
            "bankUserName": "",
            "kycType": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	BankAccountRespVO
message	message	string	

schema Description

BankAccountRespVO

name	description	type	schema
balance	balance Unit: cent	string	
bankCard	bank card	string	
bankUserName	bank User Name	string	
kycType	kyc type 0：Anonymous 1：Regular 2：Business	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«BankAccountRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Bank--Bank card transactions list		

note:Bank card transactions list

url: `/v1/bankCard/query/bankFlowing`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params：

Response Example：

```
{
    "code": 0,
    "data": [
        {
            "currencySymbol": "USD",
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingResqVO
message	message	string	

schema Description

UserBankFlowingResqVO

name	description	type	schema
currencySymbol	currency Symbol	string	
payTime	payment time	string	
stablecoinAmount	Tokenized Deposit/Stablecoin count	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Bank--Query user's bank card information		

note:Query user's bank card information

url: `/v1/bankCard/query/user/bank`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```json
{
    "code": 0,
    "data": {
        "balance": 0,
        "bankCard": "",
        "bankUserName": "",
        "currencySymbol": "USD"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankRespVO	UserBankRespVO
message	message	string	

schema Description

UserBankRespVO

name	description	type	schema
balance	balance Unit: cents	number	
bankCard	bank card	string	
bankUserName	bank user name	string	
currencySymbol	currency Symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Message Notification		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params：

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submission (Processing) 5: Submission successful (Effective) 6: Submission failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

get token

note:get token

url: `/v1/openApi/getToken`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

Description:

Tokenized Deposit/Stablecoin Management System Api

HOST:10.0.7.87:9000

Contact:

Version:1.0

api url: /v2/api-docs

APP Account

**Account--Query account balance

note:Query account balance

url: `/v1/sp/payAccount/query/user/balance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "balance": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankSpRespVO	UserBankSpRespVO
message	message	string	

schema Description

UserBankSpRespVO

name	description	type	schema
balance	balance	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankSpRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**ccount--Query account transaction list		

note:Query account transaction list

url: `/v1/sp/payAccount/query/user/flowing`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingSpResqVO
message	message	string	

schema Description

UserBankFlowingSpResqVO

name	description	type	schema
payTime	Time of payment	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingSpResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Approve		

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x1111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "toWalletId": 0,
                "walletName": ""
            }
        ]
```

```
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoins	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: ["*/*"]

Request Example:

```
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoins	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
aptcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Message		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params:

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submitting (processing) 5: Submitting succeeded (taking effect) 6: submitting failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		
** Payment--Check payment password**		

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method： POST

consumes: ["application/json"]

produces: ["*/*"]

Request Params：

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Forget payment password		

note:Forget payment password

url: /v1/app/paymentSetup/forgetPayPassword/modifyPayPassword

method： POST

consumes: ["application/json"]

produces: ["*/*"]

Request Example：

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params:

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

** Tokenized Deposit/Stablecoin--Top up**

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]` http://localhost:8003/v1/home/getTrendGraph

Request Example:

```
{
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params:

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	busId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount：Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5：success 6：failed other：processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query wallet information**		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated withdrawal transaction**		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send withdrawal transaction**		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Query transaction by hash**		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
        "txHash": "",
        "txTime": "",
        "type": 0,
        "walletAddress": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Transaction list**		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
```

```
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated transfer transaction**		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method：`POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params：

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send transfer transaction**		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method : `POST`

consumes: ["application/json"]

produces: ["*/*"]

Request Example:

```json
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description	type	schema
createTime	created time	string(date-time)	
email	email	string	
expire	Tokenized Deposit/Stablecoin expire time	integer(int64)	
iconPath	icon path	string	
kycType	kyc type	integer(int32)	
token	token	string	
userName	user name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Password Check		

note:

url: `/v1/user/passwordCheck`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordCheckReqVO	password check	body	true	PasswordCheckReqVO	PasswordCheckReqVO

schema Description

PasswordCheckReqVO

name	description	in	require	data type	schema
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

** Wallet--check result,legal currency**

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet seal tx**		

note:Cancel wallet seal tx

url: `/v1/app/userWallet/cancelWalletSealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params :

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
```

```
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet send tx**		

note:Wallet--Cancel wallet send tx

url: `/v1/app/userWallet/cancelWalletSendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "signedMessage": "121324",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletSendTxReqVO	CancelWalletSendTxReqVO

schema Description

CancelWalletSendTxReqVO

name	description	in	require	data type	schema
signedMessage	sign message	body	true	string	
txHash	transaction hash	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Create wallet**		

note:Create wallet

url: `/v1/app/userwallet/create`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet detail**		

note:Wallet detail

url: `/v1/app/userWallet/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Stablecoin--Currency Symbol		

note:Currency symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Check Stablecoin Price**		

note:Check Tokenized Deposit/Stablecoin price

url: `/v1/app/userwallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "stablecoinCount": 1
}
```

Request Params:

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example:

```json
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Stablecoin symbol**		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userwallet/getStablecoinSymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Import wallet**		

note: Import wallet

url: `/v1/app/userWallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet list**		

note:Wallet list

url: `/v1/app/userWallet/list`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
```

```
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0: Anonymous 1: Regular 2: Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0: In Progress 1: Active 2: Frozen 3: Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet balance**		

note:Wallet balance

url: `/v1/app/userwallet/totalAmount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction detail**		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "payType": "Account balances",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00",
            "txType": "Tokenized Deposit/Stablecoin Top-up"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
payType	payment Type	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	
txType	tx Type	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
transferType	transfer type 1： Transfer out 2: Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction record**		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
businessCode	txType if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

** get token**

note:get token

url: `/v1/openApi/getToken`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

Description：

Tokenized Deposit/Stablecoin Management System Api

HOST:10.0.7.87:9000

Contact:

Version:1.0

api url：/v2/api-docs

APP Account

**Account--Query account balance

note:Query account balance

url: `/v1/sp/payAccount/query/user/balance`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "balance": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankSpRespVO	UserBankSpRespVO
message	message	string	

schema Description

UserBankSpRespVO

name	description	type	schema
balance	balance	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankSpRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**ccount--Query account transaction list		

note:Query account transaction list

url: `/v1/sp/payAccount/query/user/flowing`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingSpResqVO
message	message	string	

schema Description

UserBankFlowingSpResqVO

name	description	type	schema
payTime	Time of payment	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingSpResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Approve		

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method: `POST`

consumes: ["application/json"]

produces: ["*/*"]

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x111111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "walletId": 0
}
```

Request Params :

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "toWalletId": 0,
                "walletName": ""
            }
        ]
```

```
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoins	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoins	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method: `POST`

consumes: ["application/json"]

produces: ["*/*"]

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
aptcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Message		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method：`POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params：

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submitting (processing) 5: Submitting succeeded (taking effect) 6: submitting failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		
** Payment--Check payment password**		

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Forget payment password		

note:Forget payment password

url: `/v1/app/paymentSetup/forgetPayPassword/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params:

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

** Tokenized Deposit/Stablecoin--Top up**

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

http://localhost:8003/v1/home/getTrendGraph

Request Example:

```
{
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params:

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	busId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount： Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5： success 6： failed other： processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query wallet information**		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Encapsulated withdrawal transaction**		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
```

```
    }
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Send withdrawal transaction**		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query transaction by hash**		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
```

```
            "txHash": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        },
        "message": ""
    }
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Transaction list**		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated transfer transaction**		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send transfer transaction**		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description	type	schema
createTime	created time	string(date-time)	
email	email	string	
expire	Tokenized Deposit/Stablecoin expire time	integer(int64)	
iconPath	icon path	string	
kycType	kyc type	integer(int32)	
token	token	string	
userName	user name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Password Check		

note:

url: `/v1/user/passwordCheck`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordCheckReqVO	password check	body	true	PasswordCheckReqVO	PasswordCheckReqVO

schema Description

PasswordCheckReqVO

name	description	in	require	data type	schema
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

** Wallet--check result,legal currency**

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet seal tx**		

note:Cancel wallet seal tx

url: `/v1/app/userWallet/cancelWalletSealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
```

```
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet send tx**		

note:Wallet--Cancel wallet send tx

url: `/v1/app/userWallet/cancelWalletSendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "signedMessage": "121324",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletSendTxReqVO	CancelWalletSendTxReqVO

schema Description

CancelWalletSendTxReqVO

name	description	in	require	data type	schema
signedMessage	sign message	body	true	string	
txHash	transaction hash	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Create wallet**		

note:Create wallet

url: `/v1/app/userwallet/create`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet detail**		

note:Wallet detail

url: `/v1/app/userWallet/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Stablecoin--Currency Symbol		

note:Currency symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Check Stablecoin Price**		

note:Check Tokenized Deposit/Stablecoin price

url: `/v1/app/userwallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "stablecoinCount": 1
}
```

Request Params：

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example:

```
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Stablecoin symbol**		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userWallet/getStablecoinSymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Import wallet**		

note: Import wallet

url: `/v1/app/userWallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet list**		

note:Wallet list

url: `/v1/app/userWallet/list`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
```

```
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0：Anonymous 1：Regular 2：Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0：In Progress 1：Active 2：Frozen 3：Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet balance**		

note:Wallet balance

url: `/v1/app/userWallet/totalAmount`

method：`POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params：

Response Example:

```
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction detail**		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "payType": "Account balances",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00",
            "txType": "Tokenized Deposit/Stablecoin Top-up"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
payType	payment Type	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	
txType	tx Type	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
transferType	transfer type 1: Transfer out 2: Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction record**		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
businessCode	txType if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

** get token**

note:get token

url: `/v1/openApi/getToken`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

Description :

Tokenized Deposit/Stablecoin Management System Api

HOST:10.0.7.87:9000

Contact:

Version:1.0

api url : /v2/api-docs

APP Account

**Account--Query account balance

note:Query account balance

url: `/v1/sp/payAccount/query/user/balance`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "balance": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankSpRespVO	UserBankSpRespVO
message	message	string	

schema Description

UserBankSpRespVO

name	description	type	schema
balance	balance	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankSpRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**ccount--Query account transaction list		

note:Query account transaction list

url: `/v1/sp/payAccount/query/user/flowing`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingSpResqVO
message	message	string	

schema Description

UserBankFlowingSpResqVO

name	description	type	schema
payTime	Time of payment	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingSpResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Approve		

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x1111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "toWalletId": 0,
                "walletName": ""
            }
        ]
```

```
        },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoins	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: ["*/*"]

Request Example:

```
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoins	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params：

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
aptcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Message		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params:

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submitting (processing) 5: Submitting succeeded (taking effect) 6: submitting failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		
** Payment--Check payment password**		

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Forget payment password		

note:Forget payment password

url: `/v1/app/paymentSetup/forgetPayPassword/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params：

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params：

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example：

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params：

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status：

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

** Tokenized Deposit/Stablecoin--Top up**

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

http://localhost:8003/v1/home/getTrendGraph

Request Example：

```
{
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params：

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	busId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount：Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5：success 6：failed other：processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query wallet information**		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Encapsulated withdrawal transaction**		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
```

```
    }
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Send withdrawal transaction**		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query transaction by hash**		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
```

```
        "txHash": "",
        "txTime": "",
        "type": 0,
        "walletAddress": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Transaction list**		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params :

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated transfer transaction**		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send transfer transaction**		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description		type	schema
createTime	created time		string(date-time)	
email	email		string	
expire	Tokenized Deposit/Stablecoin expire time		integer(int64)	
iconPath	icon path		string	
kycType	kyc type		integer(int32)	
token	token		string	
userName	user name		string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Password Check		

note:

url: `/v1/user/passwordCheck`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordCheckReqVO	password check	body	true	PasswordCheckReqVO	PasswordCheckReqVO

schema Description

PasswordCheckReqVO

name	description	in	require	data type	schema
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params :

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

** Wallet--check result,legal currency**

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet seal tx**		

note:Cancel wallet seal tx

url: `/v1/app/userWallet/cancelWalletSealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params :

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
```

```
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet send tx**		

note:Wallet--Cancel wallet send tx

url: `/v1/app/userWallet/cancelWalletSendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "signedMessage": "121324",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletSendTxReqVO	CancelWalletSendTxReqVO

schema Description

CancelWalletSendTxReqVO

name	description	in	require	data type	schema
signedMessage	sign message	body	true	string	
txHash	transaction hash	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Create wallet**		

note:Create wallet

url: `/v1/app/userwallet/create`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet detail**		

note:Wallet detail

url: `/v1/app/userWallet/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Stablecoin--Currency Symbol		

note:Currency symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```json
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Check Stablecoin Price**		

note:Check Tokenized Deposit/Stablecoin price

url: `/v1/app/userwallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "stablecoinCount": 1
}
```

Request Params：

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example：

```
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params：

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Stablecoin symbol**		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userWallet/getStablecoinSymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Import wallet**		

note: Import wallet

url: `/v1/app/userWallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet list**		

note:Wallet list

url: `/v1/app/userWallet/list`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
```

```
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0: Anonymous 1: Regular 2: Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0: In Progress 1: Active 2: Frozen 3: Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet balance**		

note:Wallet balance

url: `/v1/app/userwallet/totalAmount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction detail**		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params :

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "payType": "Account balances",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00",
            "txType": "Tokenized Deposit/Stablecoin Top-up"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
payType	payment Type	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	
txType	tx Type	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
transferType	transfer type 1： Transfer out 2： Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction record**		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
businessCode	txType if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

** get token**

note:get token

url: `/v1/openApi/getToken`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

Description:

Tokenized Deposit/Stablecoin Management System Api

HOST:10.0.7.87:9000

Contact:

Version:1.0

api url: /v2/api-docs

APP Account

**Account--Query account balance

note:Query account balance

url: `/v1/sp/payAccount/query/user/balance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "balance": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankSpRespVO	UserBankSpRespVO
message	message	string	

schema Description

UserBankSpRespVO

name	description	type	schema
balance	balance	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankSpRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**ccount--Query account transaction list		

note:Query account transaction list

url: `/v1/sp/payAccount/query/user/flowing`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingSpResqVO
message	message	string	

schema Description

UserBankFlowingSpResqVO

name	description	type	schema
payTime	Time of payment	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingSpResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Approve		

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "toWalletId": 0,
                "walletName": ""
            }
        ]
```

```
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoins	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: ["*/*"]

Request Example:

```
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoins	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method: `POST`

consumes: ["application/json"]

produces: ["*/*"]

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
aptcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Message		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params:

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submitting (processing) 5: Submitting succeeded (taking effect) 6: submitting failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		
** Payment--Check payment password**		

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Forget payment password		

note:Forget payment password

url: `/v1/app/paymentSetup/forgetPayPassword/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params:

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

** Tokenized Deposit/Stablecoin--Top up**

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

http://localhost:8003/v1/home/getTrendGraph

Request Example:

```
{
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params:

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	busId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount：Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5：success 6：failed other：processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query wallet information**		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Encapsulated withdrawal transaction**		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
```

```
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send withdrawal transaction**		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Query transaction by hash**		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
```

```
        "txHash": "",
        "txTime": "",
        "type": 0,
        "walletAddress": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Transaction list**		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params :

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated transfer transaction**		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send transfer transaction**		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description	type	schema
createTime	created time	string(date-time)	
email	email	string	
expire	Tokenized Deposit/Stablecoin expire time	integer(int64)	
iconPath	icon path	string	
kycType	kyc type	integer(int32)	
token	token	string	
userName	user name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Password Check		

note:

url: `/v1/user/passwordCheck`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordCheckReqVO	password check	body	true	PasswordCheckReqVO	PasswordCheckReqVO

schema Description

PasswordCheckReqVO

name	description	in	require	data type	schema
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

** Wallet--check result,legal currency**

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet seal tx**		

note:Cancel wallet seal tx

url: `/v1/app/userWallet/cancelWalletSealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
```

```
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet send tx**		

note:Wallet--Cancel wallet send tx

url: `/v1/app/userWallet/cancelWalletSendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "signedMessage": "121324",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletSendTxReqVO	CancelWalletSendTxReqVO

schema Description

CancelWalletSendTxReqVO

name	description	in	require	data type	schema
signedMessage	sign message	body	true	string	
txHash	transaction hash	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Create wallet**		

note:Create wallet

url: `/v1/app/userwallet/create`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params：

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet detail**		

note:Wallet detail

url: `/v1/app/userwallet/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Stablecoin--Currency Symbol		

note:Currency symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Check Stablecoin Price**		

note:Check Tokenized Deposit/Stablecoin price

url: `/v1/app/userwallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```json
{
    "stablecoinCount": 1
}
```

Request Params：

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example:

```json
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Stablecoin symbol**		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userWallet/getStablecoinSymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Import wallet**		

note: Import wallet

url: `/v1/app/userWallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet list**		

note:Wallet list

url: `/v1/app/userWallet/list`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
```

```
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0：Anonymous 1：Regular 2：Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0：In Progress 1：Active 2：Frozen 3：Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet balance**		

note:Wallet balance

url: `/v1/app/userwallet/totalAmount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction detail**		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "payType": "Account balances",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00",
            "txType": "Tokenized Deposit/Stablecoin Top-up"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
payType	payment Type	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	
txType	tx Type	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
transferType	transfer type 1：Transfer out 2: Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction record**		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
businessCode	txType if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

** get token**

note:get token

url: `/v1/openApi/getToken`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

Description :

Tokenized Deposit/Stablecoin Management System Api

HOST:10.0.7.87:9000

Contact:

Version:1.0

api url : /v2/api-docs

APP Account

**Account--Query account balance

note:Query account balance

url: `/v1/sp/payAccount/query/user/balance`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "balance": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankSpRespVO	UserBankSpRespVO
message	message	string	

schema Description

UserBankSpRespVO

name	description	type	schema
balance	balance	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankSpRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**ccount--Query account transaction list		

note:Query account transaction list

url: `/v1/sp/payAccount/query/user/flowing`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```json
{
    "code": 0,
    "data": [
        {
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingSpResqVO
message	message	string	

schema Description

UserBankFlowingSpResqVO

name	description	type	schema
payTime	Time of payment	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingSpResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Approve		

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params :

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example :

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params :

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x11111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "toWalletId": 0,
                "walletName": ""
            }
        ]
```

```
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoins	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoins	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
aptcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Message		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params:

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submitting (processing) 5: Submitting succeeded (taking effect) 6: submitting failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		
** Payment--Check payment password**		

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Forget payment password		

note:Forget payment password

url: `/v1/app/paymentSetup/forgetPayPassword/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params:

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

** Tokenized Deposit/Stablecoin--Top up**

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

http://localhost:8003/v1/home/getTrendGraph

Request Example:

```
{
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params:

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	busId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount：Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5：success 6：failed other：processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query wallet information**		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated withdrawal transaction**		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
```

```
    }
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send withdrawal transaction**		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query transaction by hash**		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
```

```
        "txHash": "",
        "txTime": "",
        "type": 0,
        "walletAddress": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Transaction list**		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```json
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated transfer transaction**		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send transfer transaction**		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description	type	schema
createTime	created time	string(date-time)	
email	email	string	
expire	Tokenized Deposit/Stablecoin expire time	integer(int64)	
iconPath	icon path	string	
kycType	kyc type	integer(int32)	
token	token	string	
userName	user name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Password Check		

note:

url: `/v1/user/passwordCheck`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordCheckReqVO	password check	body	true	PasswordCheckReqVO	PasswordCheckReqVO

schema Description

PasswordCheckReqVO

name	description	in	require	data type	schema
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

** Wallet--check result,legal currency**

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet seal tx**		

note:Cancel wallet seal tx

url: `/v1/app/userWallet/cancelWalletSealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
```

```
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet send tx**		

note:Wallet--Cancel wallet send tx

url: `/v1/app/userWallet/cancelWalletSendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "signedMessage": "121324",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletSendTxReqVO	CancelWalletSendTxReqVO

schema Description

CancelWalletSendTxReqVO

name	description	in	require	data type	schema
signedMessage	sign message	body	true	string	
txHash	transaction hash	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Create wallet**		

note:Create wallet

url: `/v1/app/userwallet/create`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet detail**		

note:Wallet detail

url: `/v1/app/userwallet/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Stablecoin--Currency Symbol		

note:Currency symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Check Stablecoin Price**		

note:Check Tokenized Deposit/Stablecoin price

url: `/v1/app/userWallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```json
{
    "stablecoinCount": 1
}
```

Request Params：

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example:

```json
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Stablecoin symbol**		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userwallet/getStablecoinSymbol`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Import wallet**		

note: Import wallet

url: `/v1/app/userWallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet list**		

note:Wallet list

url: `/v1/app/userWallet/list`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
```

```
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0： Anonymous 1： Regular 2： Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0： In Progress 1： Active 2： Frozen 3： Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet balance**		

note:Wallet balance

url: `/v1/app/userwallet/totalAmount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction detail**		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "payType": "Account balances",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00",
            "txType": "Tokenized Deposit/Stablecoin Top-up"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
payType	payment Type	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	
txType	tx Type	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
transferType	transfer type 1: Transfer out 2: Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction record**		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
businessCode	txType if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

** get token**

note:get token

url: `/v1/openApi/getToken`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

Description:

Tokenized Deposit/Stablecoin Management System Api

HOST:10.0.7.87:9000

Contact:

Version:1.0

api url: /v2/api-docs

APP Account

**Account--Query account balance

note:Query account balance

url: `/v1/sp/payAccount/query/user/balance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "balance": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserBankSpRespVO	UserBankSpRespVO
message	message	string	

schema Description

UserBankSpRespVO

name	description	type	schema
balance	balance	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserBankSpRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**ccount--Query account transaction list		

note:Query account transaction list

url: `/v1/sp/payAccount/query/user/flowing`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params :

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "payTime": "",
            "stablecoinAmount": 0,
            "type": 0
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserBankFlowingSpResqVO
message	message	string	

schema Description

UserBankFlowingSpResqVO

name	description	type	schema
payTime	Time of payment	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount	number	
type	type 1：purchase 2：embody	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserBankFlowingSpResqVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Approve		

Approve--Deauthorize the sending transaction

note:Deauthorize the sending transaction

url: `/v1/app/approveTransfer/CancelApprove/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1,
    "payPassword": "123456",
    "signedMessage": "323443"
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSendTxReqVO	Send cancel authorization transaction	body	true	CancelApproveSendTxReqVO	CancelApproveSendTxReqVO

schema Description

CancelApproveSendTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization quota query		

note:Authorization quota query

url: `/v1/app/approveTransfer/allowance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
allowanceReqVO	Query authorized amount	body	true	AllowanceReqVO	AllowanceReqVO

schema Description

AllowanceReqVO

name	description	in	require	data type	schema
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Approve available balance		

note:Authorize available balance

url: `/v1/app/approveTransfer/approve/availableBalance`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
approveAvailableBalanceReqVO	approveAvailableBalanceReqVO	body	true	ApproveAvailableBalanceReqVO	ApproveAvailableBalanceReqVO

schema Description

ApproveAvailableBalanceReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	number	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«bigdecimal»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transaction encapsulation		

note:Authorize transaction encapsulation

url: `/v1/app/approveTransfer/approve/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveSealTxReqVO	ApproveSealTxReqVO

schema Description

ApproveSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the sending of transactions		

note:Authorize the sending of transactions

url: `/v1/app/approveTransfer/approve/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveSendTxReqVO	Send authorization transaction	body	true	ApproveSendTxReqVO	ApproveSendTxReqVO

schema Description

ApproveSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
to	to	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--List of authorized parties		

note:List of authorized parties

url: `/v1/app/approveTransfer/authorizedPartyList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
authorizedPartyListReqVO	authorizedPartyListReqVO	body	true	AuthorizedPartyListReqVO	AuthorizedPartyListReqVO

schema Description

AuthorizedPartyListReqVO

name	description	in	require	data type	schema
walletId	Sender Wallet Id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "surplusStablecoinCount": 1,
            "walletAddress": "0x11111111111111111111111111111111111111111"
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	AuthorizedPartyListRespVO
message	message	string	

schema Description

AuthorizedPartyListRespVO

name	description	type	schema
surplusStablecoinCount	Authorized balance available	number	
walletAddress	From	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«AuthorizedPartyListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Deauthorize the encapsulated transaction		

note:Deauthorize the encapsulated transaction

url: `/v1/app/approveTransfer/cancelApprove/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "authId": 1
}
```

Request Params:

name	description	in	require	data type	schema
cancelApproveSealTxReqVO	Encapsulate cancel authorization transaction	body	true	CancelApproveSealTxReqVO	CancelApproveSealTxReqVO

schema Description

CancelApproveSealTxReqVO

name	description	in	require	data type	schema
authId	auth_id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization management		

note:Authorization Transaction List

url: `/v1/app/approveTransfer/listPage`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferListPageReqVO	approveTransferListPageReqVO	body	true	ApproveTransferListPageReqVO	ApproveTransferListPageReqVO

schema Description

ApproveTransferListPageReqVO

name	description	in	require	data type	schema
walletId		body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "authorizationRespVOS": [
            {
                "authorizationAddress": "",
                "authorizationCount": 0,
                "authorizationRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "authorizationWalletCount": 0,
                "walletBalance": "",
                "walletId": 0,
                "walletName": ""
            }
        ],
        "beAuthorizedRespVOS": [
            {
                "beAuthorizationAddress": "",
                "beAuthorizationWalletCount": 0,
                "beAuthorizedRecordRespVOS": [
                    {
                        "authId": 0,
                        "authorizedAddress": "",
                        "authorizedBalance": 0,
                        "authorizedState": 0,
                        "authorizedTime": "",
                        "availableBalance": 0,
                        "state": 0,
                        "walletId": 0
                    }
                ],
                "toWalletId": 0,
                "walletName": ""
            }
        ]
```

```
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveTransferListPageRespVO	ApproveTransferListPageRespVO
message	message	string	

schema Description

ApproveTransferListPageRespVO

name	description	type	schema
authorizationRespVOS	authorization	array	AuthorizationRespVO
beAuthorizedRespVOS	Be authorized	array	BeAuthorizedRespVO

AuthorizationRespVO

name	description	type	schema
authorizationAddress	Authorized address	string	
authorizationCount	Total number of authorized Tokenized Deposit/Stablecoins	number	
authorizationRecordRespVOS	Authorization records	array	AuthorizationRecordRespVO
authorizationWalletCount	Number of authorized wallets	integer(int32)	
walletBalance	wallet Balance	string	
walletId	walletId	integer(int64)	
walletName	Wallet name	string	

AuthorizationRecordRespVO

name	description	type	schema
authId	authId	integer(int64)	
authorizedAddress	Authorized Address	string	
authorizedBalance	Authorized Balance	number	
authorizedState	Authorization Status 1: Authorization 2: Deauthorization	integer(int32)	
authorizedTime	Authorized Time	string	
availableBalance	Available Balance	number	
state	Status: 4: Submitting (Processing) 5: Submission successful (effective) 6: Submission failed	integer(int32)	
walletId	walletId	integer(int64)	

BeAuthorizedRespVO

name	description	type	schema
beAuthorizationAddress	Authorized address	string	
beAuthorizationWalletCount	Number of authorized wallets	integer(int32)	
beAuthorizedRecordRespVOS	Authorized records	array	AuthorizationRecordRespVO
toWalletId	toWalletId	integer(int64)	
walletName	Wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveTransferListPageRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorization records		

note:Authorization Transaction Details

url: `/v1/app/approveTransfer/record`

method: `POST`

consumes: `["application/json"]`

produces: ["*/*"]

Request Example:

```
{
    "toWalletId": 0,
    "walletId": 0
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferDetailReqVO	approveTransferDetailReqVO	body	true	ApproveTransferDetailReqVO	ApproveTransferDetailReqVO

schema Description

ApproveTransferDetailReqVO

name	description	in	require	data type	schema
toWalletId	toWalletId	body	false	integer(int64)	
walletId	walletId	body	false	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "stablecoinCount": "",
            "state": 0,
            "toWalletAddress": "",
            "txTime": "",
            "type": 0,
            "walletAddress": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	ApproveTransferDetailRespVO
message	message	string	

schema Description

ApproveTransferDetailRespVO

name	description	type	schema
stablecoinCount	Number of authorized Tokenized Deposit/Stablecoins	string	
state	Submission Status 3: Pending (Pending) 4: Submitting (Processing) 5: Submission Successful (Effective) 6: Submission Failed	integer(int32)	
toWalletAddress	Authorized wallet address	string	
txTime	txTime	string	
type	Type 1: Authorization 2: Deauthorization	integer(int32)	
walletAddress	Wallet address	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«ApproveTransferDetailRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize transfers to encapsulate transactions		

note:Authorize transfers to encapsulate transactions

url: `/v1/app/approveTransfer/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 1,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSealTxReqVO	Encapsulate authorization transaction	body	true	ApproveTransferSealTxReqVO	ApproveTransferSealTxReqVO

schema Description

ApproveTransferSealTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ApproveSealTxRespVO	ApproveSealTxRespVO
message	message	string	

schema Description

ApproveSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«ApproveSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**pprove--Authorize the transfer to send the transaction		

note:Authorize the transfer to send the transaction

url: `/v1/app/approveTransfer/transfer/sendTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```json
{
    "amount": 3,
    "from": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "payPassword": "123456",
    "signedMessage": "323443",
    "spender": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "to": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
approveTransferSendTxReqVO	Send authorization transaction	body	true	ApproveTransferSendTxReqVO	ApproveTransferSendTxReqVO

schema Description

ApproveTransferSendTxReqVO

name	description	in	require	data type	schema
amount	amount	body	true	number	
from	from	body	true	string	
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
spender	spender	body	true	string	
to	to	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Captcha		

Captcha--Check captcha

note:Check captcha

url: `/v1/captcha/checkCaptcha`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "00000",
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
checkCaptchaReqVO	Check captcha code	body	true	CheckCaptchaReqVO	CheckCaptchaReqVO

schema Description

CheckCaptchaReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
aptcha--Get captcha		

note:Get captcha

url: `/v1/captcha/getCaptcha`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "captchaType": "APP_REGISTER",
    "email": "123456@mail.com"
}
```

Request Params:

name	description	in	require	data type	schema
getCaptchaReqVO	Get captcha code	body	true	GetCaptchaReqVO	GetCaptchaReqVO

schema Description

GetCaptchaReqVO

name	description	in	require	data type	schema
captchaType	captcha type	body	true	string	
email	email	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {},
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	object	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Message		

App--Message Notification

note:Message Notification

url: `/v1/app/message/notification`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
[
    {
        "busId": 0,
        "type": 0
    }
]
```

Request Params：

name	description	in	require	data type	schema
notificationReqVOs	notificationReqVOs	body	true	array	NotificationReqVO

schema Description

NotificationReqVO

name	description	in	require	data type	schema
busId	Business ID	body	true	integer(int64)	
type	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "busId": 0,
            "busType": 0,
            "state": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	NotificationRespVO
message	message	string	

schema Description

NotificationRespVO

name	description	type	schema
busId	Business ID	integer(int64)	
busType	Business type 1: Wallet activation 2: Wallet freeze 3: Wallet unfreeze 4: Wallet cancellation 5: Wallet type change 6: Tokenized Deposit/Stablecoin freezing 7: Tokenized Deposit/Stablecoin unfreezing 8: General limit setting 9: Custom limit setting 10: Transfer 11: Authorization 12: Authorization transfer 13: Tokenized Deposit/Stablecoin issuance and additional issuance 14: Tokenized Deposit/Stablecoin burning 15: Tokenized Deposit/Stablecoin price setting 16: Contract deployment 17: Contract upgrade 18: Tokenized Deposit/Stablecoin purchase 19: Tokenized Deposit/Stablecoin withdrawal	integer(int32)	
state	Status 4: Submitting (processing) 5: Submitting succeeded (taking effect) 6: submitting failed	integer(int32)	
walletName	walletName	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«NotificationRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Payment Settings		
** Payment--Check payment password**		

note:Check payment password

url: `/v1/app/paymentSetup/checkPayPassword`

method : POST

consumes: ["application/json"]

produces: ["*/*"]

Request Params:

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Forget payment password		

note:Forget payment password

url: /v1/app/paymentSetup/forgetPayPassword/modifyPayPassword

method : POST

consumes: ["application/json"]

produces: ["*/*"]

Request Example :

```
{
    "captcha": "00000",
    "newPayPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
forgetPayPasswordReqVO	Forget payment password	body	true	ForgetPayPasswordReqVO	ForgetPayPasswordReqVO

schema Description

ForgetPayPasswordReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
newPayPassword	new payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Modify payment password		

note:Modify payment password

url: `/v1/app/paymentSetup/modifyPayPassword`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "newPayPassword": "123456",
    "oldPayPassword": "000000"
}
```

Request Params：

name	description	in	require	data type	schema
modifyPayPasswordReqVO	Modify payment password	body	true	ModifyPayPasswordReqVO	ModifyPayPasswordReqVO

schema Description

ModifyPayPasswordReqVO

name	description	in	require	data type	schema
newPayPassword	new payment password	body	true	string	
oldPayPassword	old payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Setting--Set payment password		

note:Set payment password

url: `/v1/app/paymentSetup/setPayPassword`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
setPayPasswordReqVO	Set payment password	body	true	SetPayPasswordReqVO	SetPayPasswordReqVO

schema Description

SetPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Payment--Verify payment password		

note:Verify payment password

url: `/v1/app/paymentSetup/verifyPayPassword`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "payPassword": "123456"
}
```

Request Params：

name	description	in	require	data type	schema
verifyPayPasswordReqVO	Verify payment password	body	true	VerifyPayPasswordReqVO	VerifyPayPasswordReqVO

schema Description

VerifyPayPasswordReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	

Response Example：

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params：

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status：

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Tokenized Deposit/Stablecoin		

** Tokenized Deposit/Stablecoin--Top up**

note:Top up Tokenized Deposit/Stablecoin

url: `/v1/stablecoin/order/buy`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

http://localhost:8003/v1/home/getTrendGraph

Request Example:

```
{
    "payPassword": "123456",
    "stablecoinCount": 1,
    "walletAddress": "0x813564570bc06898ca46deb56e3cda59f10eb554"
}
```

Request Params:

name	description	in	require	data type	schema
buyStablecoinReqVO	Top Up Tokenized Deposit/Stablecoin	body	true	BuyStablecoinReqVO	BuyStablecoinReqVO

schema Description

BuyStablecoinReqVO

name	description	in	require	data type	schema
payPassword	payment password aes	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "orderId": 0,
        "orderNumber": 123456,
        "payAmount": 2.22,
        "payState": 0,
        "payTime": "2023-04-15 17:00:00",
        "stablecoinCount": 1,
        "state": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyStablecoinRespVO	BuyStablecoinRespVO
message	message	string	

schema Description

BuyStablecoinRespVO

name	description	type	schema
orderId	busId	integer(int64)	
orderNumber	order number	string	
payAmount	payment amount：Unit: cents	number	
payState	payment state 0: paying 1: success 2: failed	integer(int32)	
payTime	payment time yyyy-MM-dd HH:mm:ss	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 5：success 6：failed other：processing	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyStablecoinRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query wallet information**		

note:Query wallet information

url: `/v1/stablecoin/order/getBuyAccount`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params :

name	description	in	require	data type	schema
buyAccountReqVO	Query the information of the wallet that buys Tokenized Deposit/Stablecoin	body	true	BuyAccountReqVO	BuyAccountReqVO

schema Description

BuyAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "balanceCount": 1000000,
        "isCancel": true,
        "stablecoinCount": 1000
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	BuyAccountRespVO	BuyAccountRespVO
message	message	string	

schema Description

BuyAccountRespVO

name	description	type	schema
balanceCount	balance	string	
isCancel	wallet is removed or not	boolean	
stablecoinCount	max available Tokenized Deposit/Stablecoin amount	string	

Response Status:

code	description	schema
200	OK	ResultInfo«BuyAccountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Encapsulated withdrawal transaction**		

note:Encapsulated withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
withdrawalTxSealReqVO	Encapsulate withdrawal transaction	body	true	WithdrawalTxSealReqVO	WithdrawalTxSealReqVO

schema Description

WithdrawalTxSealReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
walletAddress	from address	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
```

```
    }
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send withdrawal transaction**		

note:Send withdrawal transaction

url: `/v1/stablecoin/order/withdrawal/sendTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "",
    "signedMessage": "0xaf421d******be1d5664a",
    "stablecoinCount": 1,
    "txHash": "0xaf421d******be1d5664a",
    "walletAddress": "0xaf421d******be1d5664a",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
sendWithdrawalTxReqVO	Send transaction	body	true	SendWithdrawalTxReqVO	SendWithdrawalTxReqVO

schema Description

SendWithdrawalTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	signature	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
txHash	transaction hash	body	true	string	
walletAddress	wallet address	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Query transaction by hash**		

note:Query transaction by hash

url: `/v1/stablecoin/transfer/getTxInfoByHash`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
txInfoReqVO	Transfer transaction hash	body	true	TxInfoReqVO	TxInfoReqVO

schema Description

TxInfoReqVO

name	description	in	require	data type	schema
txHash	transaction hash	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "senderWalletAddress": "",
        "stablecoinCount": 0,
        "state": 0,
        "toWalletAddress": "",
```

```
        "txHash": "",
        "txTime": "",
        "type": 0,
        "walletAddress": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferInfoRespVO	TransferInfoRespVO
message	message	string	

schema Description

TransferInfoRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	transfer type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferInfoRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Transaction list**		

note:Transaction list

url: `/v1/stablecoin/transfer/queryTransferList`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "data": {
        "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«TransferListReqVO»	DataTable«TransferListReqVO»

schema Description

DataTable«TransferListReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	TransferListReqVO	TransferListReqVO
page	Paging Information	body	true	PageInfo	PageInfo

TransferListReqVO

name	description	in	require	data type	schema
walletAddress	from	body	true	string	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "senderWalletAddress": "",
                "stablecoinCount": 0,
                "state": 0,
                "toWalletAddress": "",
                "txHash": "",
                "txTime": "",
                "type": 0,
                "walletAddress": ""
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«TransferListRespVO»	ResultData«TransferListRespVO»
message	message	string	

schema Description

ResultData«TransferListRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	TransferListRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

TransferListRespVO

name	description	type	schema
senderWalletAddress	sender	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 3：Pending Submission 4：Submitting 5：Success 6：Failed	integer(int32)	
toWalletAddress	to	string	
txHash	transaction hash	string	
txTime	transaction time	string	
type	type	integer(int32)	
walletAddress	from	string	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«TransferListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Encapsulated transfer transaction**		

note:Encapsulated transfer transaction

url: `/v1/stablecoin/transfer/sealTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
transferSealTxReqVO	Encapsulated transfer transaction	body	true	TransferSealTxReqVO	TransferSealTxReqVO

schema Description

TransferSealTxReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
walletAddress	from	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Send transfer transaction**		

note:Send transfer transaction

url: `/v1/stablecoin/transfer/sendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "signedMessage": "121324",
    "stablecoinCount": 1,
    "toWalletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
transferSendTxReqVO	Send transfer transaction	body	true	TransferSendTxReqVO	TransferSendTxReqVO

schema Description

TransferSendTxReqVO

name	description	in	require	data type	schema
payPassword	payment password: crypto	body	true	string	
signedMessage	sign message	body	true	string	
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	
toWalletAddress	to	body	true	string	
txHash	transaction hash	body	true	string	
walletAddress	from	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "transferId": 0,
        "txHash": "",
        "txState": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferTxStateRespVO	TransferTxStateRespVO
message	message	string	

schema Description

TransferTxStateRespVO

name	description	type	schema
transferId	transferId	integer(int64)	
txHash	transaction hash	string	
txState	transaction state 3: Pending Submission 4: Submitting 5: Success 6: Failed	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferTxStateRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP User		

APP--Login

note:APP user Login

url: `/v1/user/login`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```json
{
    "code": 0,
    "data": {
        "createTime": "",
        "email": "",
        "expire": 0,
        "iconPath": "",
        "kycType": 0,
        "token": "",
        "userName": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	LoginRespVO	LoginRespVO
message	message	string	

schema Description

LoginRespVO

name	description	type	schema
createTime	created time	string(date-time)	
email	email	string	
expire	Tokenized Deposit/Stablecoin expire time	integer(int64)	
iconPath	icon path	string	
kycType	kyc type	integer(int32)	
token	token	string	
userName	user name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«LoginRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Update password		

note:APP user update password

url: `/v1/user/passwordChange`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "newPassword": "456789",
    "oldPassword": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordChangeReqVO	Update password	body	true	PasswordChangeReqVO	PasswordChangeReqVO

schema Description

PasswordChangeReqVO

name	description	in	require	data type	schema
newPassword	new password	body	true	string	
oldPassword	old password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Password Check		

note:

url: `/v1/user/passwordCheck`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
passwordCheckReqVO	password check	body	true	PasswordCheckReqVO	PasswordCheckReqVO

schema Description

PasswordCheckReqVO

name	description	in	require	data type	schema
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Reset password		

note:APP user reset password

url: `/v1/user/passwordReset`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params :

name	description	in	require	data type	schema
passwordResetReqVO	Reset password	body	true	PasswordResetReqVO	PasswordResetReqVO

schema Description

PasswordResetReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**PP--Register		

note:APP user register

url: `/v1/user/register`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "captcha": "000000",
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
userRegisterReqVO	User register	body	true	UserRegisterReqVO	UserRegisterReqVO

schema Description

UserRegisterReqVO

name	description	in	require	data type	schema
captcha	captcha	body	true	string	
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": true,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	boolean	
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«boolean»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
APP Wallet		

** Wallet--check result,legal currency**

note:check result,legal currency

url: `/v1/app/userWallet/cancelWalletCheckBalance`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example :

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	GetBalanceReqVO	GetBalanceReqVO

schema Description

GetBalanceReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "status": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletBalanceCheckRespVO	CancelWalletBalanceCheckRespVO
message	message	string	

schema Description

CancelWalletBalanceCheckRespVO

name	description	type	schema
status	0: Balance<0.01 1: Balance>=0.01 2: Balance=0 3: Freeze	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletBalanceCheckRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet seal tx**		

note:Cancel wallet seal tx

url: `/v1/app/userWallet/cancelWalletSealTx`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletReqVO	CancelWalletReqVO

schema Description

CancelWalletReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "chainId": 0,
        "networkId": 0,
        "rawTransaction": {
            "data": "",
```

```
            "eip1559Transaction": true,
            "feeCap": 0,
            "gasLimit": 0,
            "gasPremium": 0,
            "gasPrice": 0,
            "legacyTransaction": true,
            "nonce": 0,
            "to": "",
            "value": 0
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	TransferSealTxRespVO	TransferSealTxRespVO
message	message	string	

schema Description

TransferSealTxRespVO

name	description	type	schema
chainId		integer(int64)	
networkId		integer(int64)	
rawTransaction		RawTransactionRespVO	RawTransactionRespVO

RawTransactionRespVO

name	description	type	schema
data		string	
eip1559Transaction		boolean	
feeCap		integer	
gasLimit		integer	
gasPremium		integer	
gasPrice		integer	
legacyTransaction		boolean	
nonce		integer	
to		string	
value		integer	

Response Status:

code	description	schema
200	OK	ResultInfo«TransferSealTxRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Cancel wallet send tx**		

note:Wallet--Cancel wallet send tx

url: `/v1/app/userWallet/cancelWalletSendTx`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "signedMessage": "121324",
    "txHash": "0xfe4eb8ca4ed97363d27372ca9c27cbb5abe68",
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
reqVO	reqVO	body	true	CancelWalletSendTxReqVO	CancelWalletSendTxReqVO

schema Description

CancelWalletSendTxReqVO

name	description	in	require	data type	schema
signedMessage	sign message	body	true	string	
txHash	transaction hash	body	true	string	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CancelWalletRespVO	CancelWalletRespVO
message	message	string	

schema Description

CancelWalletRespVO

name	description	type	schema
recordId		integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CancelWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Create wallet**		

note:Create wallet

url: `/v1/app/userwallet/create`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "payPassword": "123456",
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68",
    "walletName": "xxx"
}
```

Request Params:

name	description	in	require	data type	schema
createAccountReqVO	Create wallet	body	true	CreateAccountReqVO	CreateAccountReqVO

schema Description

CreateAccountReqVO

name	description	in	require	data type	schema
payPassword	payment password	body	true	string	
walletAddress	wallet address: max 100 characters	body	true	string	
walletName	wallet name: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "recordId": 0,
        "walletId": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CreateWalletRespVO	CreateWalletRespVO
message	message	string	

schema Description

CreateWalletRespVO

name	description	type	schema
recordId	recordId	integer(int64)	
walletId	walletId	integer(int64)	

Response Status:

code	description	schema
200	OK	ResultInfo«CreateWalletRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet detail**		

note:Wallet detail

url: `/v1/app/userwallet/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
userWalletDetailReqVO	Wallet detail	body	true	UserWalletDetailReqVO	UserWalletDetailReqVO

schema Description

UserWalletDetailReqVO

name	description	in	require	data type	schema
walletId	wallet id	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "createTime": "",
        "freezeCount": 0,
        "kycType": 0,
        "ruleType": 0,
        "stablecoinCount": 0,
        "state": 0,
        "usableCount": 0,
        "walletAddress": "",
        "walletId": 0,
        "walletName": "",
        "withdrawTime": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	UserWalletDetailRespVO	UserWalletDetailRespVO
message	message	string	

schema Description

UserWalletDetailRespVO

name	description	type	schema
createTime	created time	string	
freezeCount	freeze amount	number	
kycType	kyc type 0: Anonymous 1: Standard 2: Business	integer(int32)	
ruleType	rule type 1: General 2: Custom	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin amount	number	
state	state 0: Unknown 1: Enabled 2: Disabled 3: Removed	integer(int32)	
usableCount	usable amount	number	
walletAddress	wallet address	string	
walletId	wallet ID	integer(int64)	
walletName	wallet name	string	
withdrawTime	withdraw time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«UserWalletDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
**Stablecoin--Currency Symbol		

note:Currency symbol

url: `/v1/app/userWallet/getCurrencySymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "currencySymbol": "USD",
        "unit": "$"
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	CurrencySymbolRespVO	CurrencySymbolRespVO
message	message	string	

schema Description

CurrencySymbolRespVO

name	description	type	schema
currencySymbol	currency Symbol	string	
unit	unit	string	

Response Status:

code	description	schema
200	OK	ResultInfo«CurrencySymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin--Check Stablecoin Price**		

note:Check Tokenized Deposit/Stablecoin price

url: `/v1/app/userwallet/getStablecoinConversion`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "stablecoinCount": 1
}
```

Request Params：

name	description	in	require	data type	schema
stablecoinConversionReqVO	Tokenized Deposit/Stablecoin price	body	true	StablecoinConversionReqVO	StablecoinConversionReqVO

schema Description

StablecoinConversionReqVO

name	description	in	require	data type	schema
stablecoinCount	Tokenized Deposit/Stablecoin count	body	true	number	

Response Example:

```
{
    "code": 0,
    "data": {
        "currencyCount": 1
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinConversionRespVO	StablecoinConversionRespVO
message	message	string	

schema Description

StablecoinConversionRespVO

name	description	type	schema
currencyCount	currency count	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinConversionRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Tokenized Deposit/Stablecoin-- Stablecoin symbol**		

note:Tokenized Deposit/Stablecoin symbol

url: `/v1/app/userWallet/getStablecoinSymbol`

method : `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "symbol": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	StablecoinSymbolRespVO	StablecoinSymbolRespVO
message	message	string	

schema Description

StablecoinSymbolRespVO

name	description	type	schema
symbol	Tokenized Deposit/Stablecoin symbol	string	

Response Status:

code	description	schema
200	OK	ResultInfo«StablecoinSymbolRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Import wallet**		

note: Import wallet

url: `/v1/app/userWallet/import`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "walletAddress": "0xfe2c8a084511ed97363d27372ca9c27cbb5abe68"
}
```

Request Params:

name	description	in	require	data type	schema
importAccountReqVO	Import wallet	body	true	ImportAccountReqVO	ImportAccountReqVO

schema Description

ImportAccountReqVO

name	description	in	require	data type	schema
walletAddress	wallet address: max 100 characters	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": 0,
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	integer(int64)	integer(int64)
message	message	string	

Response Status:

code	description	schema
200	OK	ResultInfo«long»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet list**		

note:Wallet list

url: `/v1/app/userWallet/list`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": [
        {
            "freezeCount": 0,
```

```
            "kycType": 0,
            "stablecoinCount": 0,
            "state": 0,
            "usableCount": 0,
            "walletAddress": "",
            "walletId": 0,
            "walletName": ""
        }
    ],
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	array	UserWalletListRespVO
message	message	string	

schema Description

UserWalletListRespVO

name	description	type	schema
freezeCount	freeze count	number	
kycType	kyc type 0：Anonymous 1：Regular 2：Business	integer(int32)	
stablecoinCount	Tokenized Deposit/Stablecoin count	number	
state	state 0：In Progress 1：Active 2：Frozen 3：Inactive	integer(int32)	
usableCount	usable count	number	
walletAddress	wallet address	string	
walletId	wallet id	integer(int64)	
walletName	wallet name	string	

Response Status:

code	description	schema
200	OK	ResultInfo«List«UserWalletListRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Wallet balance**		

note:Wallet balance

url: `/v1/app/userWallet/totalAmount`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Params:

Response Example:

```
{
    "code": 0,
    "data": {
        "availableAmount": 0,
        "totalAmount": 0
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	WalletTotalAmountRespVO	WalletTotalAmountRespVO
message	message	string	

schema Description

WalletTotalAmountRespVO

name	description	type	schema
availableAmount	Available Amount	number	
totalAmount	Total Amount	number	

Response Status:

code	description	schema
200	OK	ResultInfo«WalletTotalAmountRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction detail**		

note:Transaction detail

url: `/v1/app/userWallet/tx/detail`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "businessId": 1,
    "txType": 1,
    "walletId": 1
}
```

Request Params:

name	description	in	require	data type	schema
queryWalletRecordDetailReqVO	Wallet transaction detail	body	true	QueryWalletRecordDetailReqVO	QueryWalletRecordDetailReqVO

schema Description

QueryWalletRecordDetailReqVO

name	description	in	require	data type	schema
businessId	business ID	body	true	integer(int64)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

Response Example:

```
{
    "code": 0,
    "data": {
        "txType": 1,
        "walletOrderDetailRespVO": {
            "bankOrderNumber": 1,
            "createTime": "15/04/2023 17:00:00",
            "orderNumber": 1,
            "payState": 1,
            "payTime": "15/04/2023 17:00:00",
            "payType": "Account balances",
            "stablecoinAmount": 1,
            "stablecoinCount": 1,
            "state": 1,
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00",
            "txType": "Tokenized Deposit/Stablecoin Top-up"
        },
        "walletTransferDetailRespVO": {
            "createTime": "15/04/2023 17:00:00",
            "fromWalletName": "fromWalletName",
            "stablecoinCount": 1,
            "state": 1,
            "transferType": 1,
            "transferWalletAddress": "0xabcdecjhijklmn",
            "txHash": "txhash",
            "txTime": "15/04/2023 17:00:00"
        }
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	QueryWalletRecordDetailRespVO	QueryWalletRecordDetailRespVO
message	message	string	

schema Description

QueryWalletRecordDetailRespVO

name	description	type	schema
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	
walletOrderDetailRespVO	transaction type 3: Top-up 6: Withdrawa	WalletOrderDetailRespVO	WalletOrderDetailRespVO
walletTransferDetailRespVO	transaction type 4: Transfer 5: Authorized Transfer	WalletTransferDetailRespVO	WalletTransferDetailRespVO

WalletOrderDetailRespVO

name	description	type	schema
bankOrderNumber	bank order Number	string	
createTime	created time	string	
orderNumber	order number	string	
payState	payment state 0:paying 1:success 2:failed	integer(int32)	
payTime	payment time	string	
payType	payment Type	string	
stablecoinAmount	Tokenized Deposit/Stablecoin amount Unit: cent	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txHash	transaction hash	string	
txTime	transaction time	string	
txType	tx Type	string	

WalletTransferDetailRespVO

name	description	type	schema
createTime	created time	string	
fromWalletName	from wallet name	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
transferType	transfer type 1：Transfer out 2：Transfer in	integer(int32)	
transferWalletAddress	transfer wallet address	string	
txHash	transaction hash	string	
txTime	transaction time	string	

Response Status:

code	description	schema
200	OK	ResultInfo«QueryWalletRecordDetailRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
** Wallet--Transaction record**		

note:Transaction record

url: `/v1/app/userWallet/tx/query`

method： `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example：

```
{
    "data": {
        "createEndTime": "2023-04-05",
        "createStartTime": "2023-04-05",
        "txType": 1,
        "walletId": 1
    },
    "page": {
        "pageNum": 0,
        "pageSize": 0
    }
}
```

Request Params:

name	description	in	require	data type	schema
param	param	body	true	DataTable«QueryWalletRecordReqVO»	DataTable«QueryWalletRecordReqVO»

schema Description

DataTable«QueryWalletRecordReqVO»

name	description	in	require	data type	schema
data	Query Entity	body	true	QueryWalletRecordReqVO	QueryWalletRecordReqVO
page	Paging Information	body	true	PageInfo	PageInfo

QueryWalletRecordReqVO

name	description	in	require	data type	schema
createEndTime	create end time yyyy-MM-dd	body	true	string	
createStartTime	create start time yyyy-MM-dd	body	true	string	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	body	true	integer(int32)	
walletId	wallet ID	body	true	integer(int64)	

PageInfo

name	description	in	require	data type	schema
pageNum	pageNum	body	true	integer(int32)	
pageSize	pageSize	body	true	integer(int32)	

Response Example:

```
{
    "code": 0,
    "data": {
        "page": {
            "isFirstPage": true,
            "isLastPage": true,
            "pageNum": 0,
            "pageSize": 0,
            "pages": 0,
            "total": 0
        },
        "rows": [
            {
                "businessCode": 1,
                "businessId": 1,
                "createTime": "15/04/2023 17:00:00",
                "stablecoinCount": 1,
                "state": 1,
                "txType": 1
            }
        ]
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	ResultData«QueryWalletRecordRespVO»	ResultData«QueryWalletRecordRespVO»
message	message	string	

schema Description

ResultData«QueryWalletRecordRespVO»

name	description	type	schema
page	page	ResultPageInfo	ResultPageInfo
rows	return information	array	QueryWalletRecordRespVO

ResultPageInfo

name	description	type	schema
isFirstPage	isFirstPage	boolean	
isLastPage	isLastPage	boolean	
pageNum	pageNum	integer(int32)	
pageSize	pageSize	integer(int32)	
pages	pages	integer(int32)	
total	total	integer(int64)	

QueryWalletRecordRespVO

name	description	type	schema
businessCode	txType if the transaction state is 3 or 6; to address if the state is 4 or 5	string	
businessId	business ID	integer(int32)	
createTime	created time	string	
stablecoinCount	Tokenized Deposit/Stablecoin amount	string	
state	state 4: In Progress 5:Success 6:Error	integer(int32)	
txType	transaction type 3: Top-up 4: Transfer 5: Authorized Transfer 6: Withdrawal	integer(int32)	

Response Status:

code	description	schema
200	OK	ResultInfo«ResultData«QueryWalletRecordRespVO»»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	
Open Api		

** get token**

note:get token

url: `/v1/openApi/getToken`

method: `POST`

consumes: `["application/json"]`

produces: `["*/*"]`

Request Example:

```
{
    "email": "123456@mail.com",
    "password": "123456"
}
```

Request Params:

name	description	in	require	data type	schema
loginReqVO	App login	body	true	LoginReqVO	LoginReqVO

schema Description

LoginReqVO

name	description	in	require	data type	schema
email	email	body	true	string	
password	password	body	true	string	

Response Example:

```
{
    "code": 0,
    "data": {
        "expire": 0,
        "token": ""
    },
    "message": ""
}
```

Response Params:

name	description	type	schema
code	code	integer(int32)	integer(int32)
data	data	OpenApiGetTokenRespVO	OpenApiGetTokenRespVO
message	message	string	

schema Description

OpenApiGetTokenRespVO

name	description	type	schema
expire	token expire	integer(int64)	
token	token	string	

Response Status:

code	description	schema
200	OK	ResultInfo«OpenApiGetTokenRespVO»
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	