Plasma open test 1 API

Plasma REST API :

all APIs send key {“error”:true/false} in returned object signalling success or failure, and can send extra key {“reason”: “...”} for descriptions

Get all unspent transaction outputs for address
POST /utxos/:address
example:
POST /utxos/0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d (capitalization irrelevant)

returns a list of unspent transaction outputs for address

returned value example

```
{
        "error": false,
        "address": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
        "utxos": [
                {
                        "blockNumber": 2,
                        "txNumberInBlock": 0,
                        "outputNumberInTransaction": 0,
                        "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "value": "5000000000000000"
                },
                {
                        "blockNumber": 4,
                        "txNumberInBlock": 0,
                        "outputNumberInTransaction": 0,
                        "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "value": "10000000000000000"
                },
                {
                        "blockNumber": 5,
                        "txNumberInBlock": 0,
                        "outputNumberInTransaction": 0,
                        "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "value": "10000000000000000"
                },
                {
                        "blockNumber": 6,
                        "txNumberInBlock": 0,
                        "outputNumberInTransaction": 0,
                        "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "value": "10000000000000000"
                },
                {
                        "blockNumber": 7,
                        "txNumberInBlock": 0,
                        "outputNumberInTransaction": 0,
                        "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "value": "10000000000000000"
                }
        ]
}
```

“utxos” array can be empty

Create Plasma transaction to be signed by user

POST /createTX

```
{
   "txType" : 1,
   "inputs": [
                        {
        "blockNumber": 1,
        "txNumber": 0,
        "outputNumber" : 0
                        }
                ],
   "outputs": [{
        "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
        "amount": "50000000000000000"
        },
                                                        {
        "to": "0x6394b37Cf80A7358b38068f0CA4760ad49983a1B",
        "amount": "50000000000000000"
        }]
}
```

where txType - transaction type

```
const TxTypeSplit = 1;
const TxTypeMerge = 2;
const TxTypeWithdraw = 3;
const TxTypeFund = 4 - not allowed
const TxTypeTransfer = 5;
```

inputs array is never empty, outputs array must be empty for TxTypeWithdraw (filled automatically)

returns

```
{
    "error": false,
    "tx": {
        "header": {
            "txNumberInBlock": 0,
            "transactionType": 1,
            "v": "0x00",
            "r":
"0x0000000000000000000000000000000000000000000000000000000000000000",
            "s":
"0x0000000000000000000000000000000000000000000000000000000000000000"
        },
        "inputs": {
            "0": {
                "blockNumber": 5,
                "txNumberInBlock": 0,
                "outputNumberInTransaction": 0,
                "value": "10000000000000000"
            }
        },
        "outputs": {
            "0": {
                "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                "outputNumberInTransaction": 0,
                "value": "5000000000000000"
            },
            "1": {
                "to": "0x6394b37Cf80A7358b38068f0CA4760ad49983a1B",
                "outputNumberInTransaction": 1,
                "value": "5000000000000000"
            }
        }
    },
    "txHex":
"0x01000000050000000000000000000000000000000000000000000000000000000000000002
386f26fc10000e6877a4d8806e9a9f12eb2e8561ea6c1db19978d0000000000000000000000
00000000000000000000000000000011c37937e080006394b37cf80a7358b38068f0ca4760a
d49983a1b010000000000000000000000000000000000000000000000000011c37937e080
00",
    "txPersonalHash":
"0x11948e75cd88833950299853cd0c84acda4331a3d073c89891177a808a8156ae",
        "txPersonalMessage":
"0x19457468657265756d205369676e6564204d6573736167653a0a3134380100000005000
0000000000000000000000000000000000000000000000000000000002386f26fc10000e68
77a4d8806e9a9f12eb2e8561ea6c1db19978d0000000000000000000000000000000000000
00000000000000011c37937e080006394b37cf80a7358b38068f0ca4760ad49983a1b010000
000000000000000000000000000000000000000000000011c37937e08000"
}
```

all fields can be ignored for now except "txPersonalMessage", "txPersonalHash" and "txHex"

txHex is raw transaction hex encoded
txPersonalMessage is txHex with proper header appended
txPersonalHash is keccak256(txPersonalMessage)
txPersonalHash will be used further for transaction signature

Send signed Plasma transaction

POST /sendSignedTX

```
{
   "txType" : 1,
   "inputs": [
                       {
         "blockNumber": 2,
         "txNumber": 0,
         "outputNumber" : 0
                       }
               ],
   "outputs": [{
         "to": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
         "amount": "50000000000000000"
         },
                                                     {
         "to": "0x6394b37Cf80A7358b38068f0CA4760ad49983a1B",
         "amount": "50000000000000000"
         }],
         "signature" :
"0x832d1f037c1b556c18868cd860ab2011a9aa254901f89cc6984ae459252b178875a398d4
d20c7ec0839ef8707fb0bfb8675642867c1a64249b0aeb1c0caa96861b"
}
```

similar to /createTX, except signature field. Creation of signature field will be shown in WEB3 related functions

Prepare proof of withdraw transaction

Prepares proof for transaction being included in a block at certain number

POST /prepareProofForExpressWithdraw

```
{
        "blockNumber": 2,
        "txNumber": 0
}
```

request indicated block number and transaction number of particular Plasma transaction of type “TxTypeWithdraw”. Obtained proof can be sent to Plasma smart contract on main network and smart contract will send money to user that withdraws transaction.

```
{
        "error": false,
        "proof": {
                "blockNumber": 2,
                "txNumber": 0,
                "tx":
"0x00000000030000000100000000000000000000000000000000000000000000000000000
00000016345785d8a00000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000016345785d8a00001b9aac8a4239935d1d2
2b77619dcbe79faa0d65d256d6e366c683a20b9d86607e93c0a095e84832463b67a391dd85
77d9aa02d74c1458eca558a9bbf7e22867e22",
                "merkleProof": "0x00"
        }
}
```

WEB3 interactions

either functions similar to shown or pure API descriptions

Get user address

```
var account;
var localWeb3;
function initWeb3(){
  var provider = window.web3.currentProvider;
  localWeb3 = new window.Web3(provider);
  localWeb3.eth.getAccounts(function(error, res) {
    console.log(error);
    if (!error) {
      console.log(res);
      account=res[0];
    }});
}
```

Sign transaction

```
var account;
var localWeb3;
var signatureHex;
function sign() {
  var data = txPersonalHash;
  console.log(data);
  localWeb3.eth.sign(account, data, function(error, res) {
    console.log(error);
    if (!error){
      console.log(res);
      signatureHex = res;
    }
  })
}
```

Here txPersonalHash - is the hash obtained from /createTX Plasma API endpoint

signatureHex should be passed for “signature” in /sendSignedTX

Create Plasma parent contract reference for further interactions

```
DeployedPlasmaContract = new web3.eth.Contract(PlasmaContract.abi,
config.deployedPlasmaContract, {from: defaultCallerAddress});
```

plasmaContract.abi and deployedPlasmaContract are provided by Plasma operator

defaultCallerAddress is an address from which all further methods will be called. Usually it’s user address obtained from Metamask.

Send withdraw proof to make withdraw on Ethereum

```
const result = await
DeployedPlasmaContract.methods.finalizeWithdrawExpress(preparedProof.blockNumber,
                preparedProof.txNumberInBlock, preparedProof.tx, preparedProof.merkleProof
).send({from:from, gas: 3.6e6});

const withdrawFinalizedEvent = result.events.WithdrawFinalizedEvent;
console.log(withdrawFinalizedEvent);
```

preparedProof - is a “preparedProof” object from POST /prepareProofForExpressWithdraw

Proxified methods

There are WEB3 methonds that are not changing the Ethereum state and just proxified as REST for convenience

All deposit records for address
GET /plasmaParent/allDeposits/:address

```
{
        "error": false,
        "depositRecords": [
                {
                        "0": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "1": "1",
                        "2": "1000000000000000000",
                        "3": "5893395209781248",
                        "4": "0",
                        "from": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "status": "1",
                        "amount": "1000000000000000000",
                        "index": "5893395209781248",
                        "withdrawStartedTime": "0"
                },
                {
                        "0": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "1": "1",
                        "2": "500000000000000000",
                        "3": "5894675110035456",
                        "4": "0",
                        "from": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "status": "1",
                        "amount": "500000000000000000",
                        "index": "5894675110035456",
                        "withdrawStartedTime": "0"
                }
        ]
}
```

```
   enum DepositStatus {
      NoRecord,
      Deposited,
      WithdrawStarted,
      WithdrawChallenged,
      WithdrawCompleted,
      DepositConfirmed
   }
```

All withdraw records for address

GET /plasmaParent/allWithdraws/:address

```
{
        "error": false,
        "withdrawRecords": [
                {
                        "0": "281474976710656",
                        "1": "1",
                        "2": "0",
                        "3": "0",
                        "4": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "5": true,
                        "6": "3",
                        "7": "100000000000000000",
                        "8": "1512732635",
                        "9": "1512732635",
                        "index": "281474976710656",
                        "blockNumber": "1",
                        "txNumberInBlock": "0",
                        "outputNumberInTX": "0",
                        "beneficiary": "0xE6877A4d8806e9A9F12eB2e8561EA6c1db19978d",
                        "isExpress": true,
                        "status": "3",
                        "amount": "100000000000000000",
                        "timeStarted": "1512732635",
                        "timeEnded": "1512732635"
                }
        ]
}
```

```
enum WithdrawStatus {
        NoRecord,
        Started,
        Challenged,
        Completed,
        Rejected
    }
```