

# Rubix API Calls

---

## DID Creation

---

Creates a unique Decentralized Identity

```
$ curl --location --request POST http://localhost:1898/create' --form 'image=@<"image path"
' --form 'data={ "data": "9876543333,user@rubix.network"}
```

**Request Type:** POST

**Port:** 1898

**Input:** data(String) and image(Multipart File)

**Returns:** DID (String)

## Initial Setup

---

Does the initial setup of IPFS and syncing the network nodes

Make sure to make this call before any other following calls

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/setup
```

**Request Type:** GET

**Port:** 1898

**Input:** null

**Returns:** Status(String), Message(String), Info(String)

**On Success:** {"Status":"Setup Complete!","Message":"User is Registered!","Info":""}

**On Failure:** {"Status":"Failed","Message":"User not registered, create your Decentralised Identity!","Info":"Config File Missing"}

## Transfer Tokens

---

Transfers token(s) from one wallet address to another

```
$ curl --header "Content-Type: application/json" --request POST
http://localhost:1898/initiateTransaction --data '{ "receiver":
"445f59c3d71c6769124470cf4b82ca0b9b1626aec4f14f50a8f1e6a13e1fc70d", "tokenCount":1,
"comment":"transaction comments"}
```

**Request Type:** POST

**Port:** 1898

**Input:** receiver (String), tokenCount (Integer), comment (String)

**Returns:** Transaction ID (String), Success / Failure (Boolean), DID (String)

## Account Information

---

Retrieves the user account details

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/getAccountInfo
```

**Request Type:** GET

**Port:** 1898

**Input:** null

**Returns:** DID (String), WalletID (String), TransactionAsSender (Integer), TransactionAsReceiver (Integer),

Balance (Integer)

{"wid":"QmQtLq2iu1uGdM5GcpRe6R9AoYqBxtPnWfzrnkkcVuAXD","balance":0,"receiverTxn":14,"did":"QmSCoxrydcVYDFuETCjyhbKqRQNDUYEcw6XhEzQD49v

## Account Balance

---

Check the balance in user's wallet

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/getBalance
```

**Request Type:** GET

**Port:** 1898

**Input:** null

**Returns:** Account Balance(Integer)

{"Balance":0}

## Get Transaction details with Transaction ID

---

Details of a particular transaction like Quorum involved, token transferred, receiver details, time and more

```
$ curl --header "Content-Type: application/json" --request POST http://localhost:1898/getTxnDetails
--data '{"transactionID":
"394caf2112586f73f1a6c90650960f3ad81eeb7da41566b0bcd9246b2c6475db"}'
```

**Request Type:** POST

**Port:** 1898

**Input:** transactionID (String)

**Returns:** senderDID(String), role(String), totalTime(Integer), tokens(ArrayList), comment(String), txn(String), receiverDID(String), Date(Date)

```
{ "senderDID": "QmRzipL1P9j2iaFqywVx1R4HNSvjThXmdUjGD3FLVHXMgu", "role": "Sender", "totalTime": 2050, "tokens":
["Qma19niardUMcHnfFtszmcZKQLjXqBEHBSZZTGBGaCMzG9"], "comment": "ReceiverFixed", "txn": "aa45765fe1f384be4d0bf03bf7d01f5e0b8f1d6c4be6f7
11-15"}*
```

## Get Transaction details with Date

Retrieves the details of all the transactions committed during the specified period

```
$ curl --header "Content-Type: application/json" --request POST http://localhost:1898/getTxnByDate
--data '{"eDate": "2020-04-21", "sDate": "2020-04-02"}'
```

**Request Type:** POST

**Port:** 1898

**Input:** sDate (String), eDate (String)

**Returns:** senderDID(String), role(String), totalTime(Integer), tokens(ArrayList), comment(String), txn(String), receiverDID(String), Date(Date)

## Get Transaction details with Comment

Retrieves the details of all the transactions committed with the specified comment

```
$ curl --header "Content-Type: application/json" --request POST
http://localhost:1898/getTxnByComment
--data '{"comment": "10,000Rs"}'
```

**Request Type:** POST

**Port:** 1898

**Input:** comment (String)

**Returns:** senderDID(String), role(String), totalTime(Integer), tokens(ArrayList), comment(String), txn(String), receiverDID(String), Date(Date)

## Get Transaction details with Count

Retrieves the last specified count of transactions committed

```
$ curl --header "Content-Type: application/json" --request POST
http://localhost:1898/getTxnByCount
--data '{"txnCount": 3}'
```

**Request Type:** POST

**Port:** 1898

**Input:** txnCount (Integer)

**Returns:** senderDID(String), role(String), totalTime(Integer), tokens(ArrayList), comment(String), txn(String), receiverDID(String), Date(Date)

## Get Transaction details with User's DID

Retrieves all the transactions made with the input DID

```
$ curl --header "Content-Type: application/json" --request POST http://localhost:1898/getTxnByDID
--data '{"did": "QmdkrLpyoGFrhsbeuZrXpcvr2QRsLuQnrbXVfJTe1yXqzy"}'
```

**Request Type:** POST

**Port:** 1898

**Input:** did (String)

**Returns:** List of all transactions and its details made with the did

## List of Tokens

Lists out all the tokens available in the user's wallet

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/viewTokens
```

**Request Type:** GET  
**Port:** 1898  
**Input:** nil  
**Returns:** List of Tokens (JSONObject)

## List of Peers Online

---

Lists all your contacts currently available online

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/getOnlinePeers
```

**Request Type:** GET  
**Port:** 1898  
**Input:** nil  
**Returns:** List of objects: did(String), online/offline

## View Contacts

---

Lists all your contacts

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/getContactsList
```

**Request Type:** GET  
**Port:** 1898  
**Input:** nil  
**Returns:** List of DIDs

## Details for Dashboard

---

Lists all required details of user's wallet

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/getDashboard
```

**Request Type:** GET  
**Port:** 1898  
**Input:** nil  
**Returns:** totalTxns(Integer), WalletID(String), balance(Integer), , receiverTxns(Integer), senderTxns(Integer), contactsCount(Integer), onlinePeers(Integer), transactionsPerDay(List), did(String)

## Working Directory

---

Checks if there exists the Rubix Working Directory for the user

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/check
```

**Request Type:** GET  
**Port:** 1898  
**Input:** nil  
**Returns:** Status(String - Success/Failed), Message(String)

*On success: {"Status": "Success", "Message": "User is Registered!"}*

*On Failure: {"Status": "Failed", "Message": "User not registered, create your Decentralised Identity!", "Info": "DID Folder Missing"}*

## Close Streams

---

To close all the streams open

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/p2pClose
```

**Request Type:** GET  
**Port:** 1898  
**Input:** nil  
**Returns:** Message(String)

*On Success: All Streams Closed*

## Synchronise Network Nodes

---

To synchronize the DIDs of the systems, so that the node will have an updated list of all the DIDs in the network.

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/sync
```

**Request Type:** GET

**Port:** 1898  
**Input:** nil  
**Returns:** Message(String)

*On Success: Successfully Synced*

## Shutdown

---

Closes all application processes and exits

```
$ curl --header "Content-Type: application/json" --request GET http://localhost:1898/shutdown
```

**Request Type:** GET

**Port:** 1898

**Input:** nil

**Returns:** nil