**Documentation for NTU-Block chain platform v1:**

Requirements:

OS: Linux, Windows, Mac

Language: Python 3

Libraries:

1. Sqlite3
2. Flask
3. Flask\_cors
4. Flask\_sqlalchemy
5. Crypto

List of files:

1. block.py
2. blockchain.py
3. make.db
4. Node.py
5. Transaction.py
6. Wallet.py
7. Utility - \_\_init\_\_.py
8. Utility/ hash\_util.py
9. Utility/ printable.py
10. Utility / verification.py

List of RestApis:

1. /user [Post]
2. /user [GET]
3. /user/<public key>[PUT]
4. /user/<public key>[DELETE]
5. /login[GET]
6. /balance [GET]
7. /transaction[POST]
8. /transactions[GET]
9. /createitem[POST]
10. /mine[POST]
11. /resolve-conflicts[POST]
12. /products[GET]
13. /chain[GET]
14. /node[POST]
15. /node/<node\_url>[DELETE]
16. /nodes[GET]

**User guide:**

All the apis can be called using Postman

**Starting the blockchain:**

To start the blockchain, run node.py in the command line.

**Logging in and calling APIs:**

To log, enter username and password in to the authorisation tab on postman.

‘/login’ [GET]

The response will be the public key of the account and a session token. To call some of the APIs, the token will be required. Each token last for **30mins**. To use the token, under Headers tab, **input x-access-token** under **key** and the **token value** under **values**.

**Creating a user account and wallet:**

Only admin accounts can create or add a new account to the database. To add new users,

‘/user’ [POST]

{ “name”: “elmo”,

“password”: “xxxx”

}

A new wallet will be created. A set of public key and private key is tied to the account. **New accounts does not have admin privileges**.

**Deleting a user account:**

Only admin accounts can delete an account to the database. To delete users,

‘/user/<public\_key>’ [DELETE]

**Querying all user accounts in the database:**

Only admin accounts can query accounts in the database. To query users,

‘/user [GET]

**Giving admin privileges:**

Only admin accounts can promote accounts in the database. To promote users,

‘/user/<public\_key>’ [PUT]

**Checking Wallet Balance:**

Return the balance of the current account.

/balance [GET]

**Creating item:**

Only admin accounts can add new product ID into the blockchain. The API adds a set quantity of a declared product into a given wallet.

‘/createitem’ [POST]

{ “recipient”: “public key”,

“productID”: “ xxxx…”,

“amount”: 0

}

**Making transactions:**

Users have to log in make transaction in the blockchain. User have to have the more or equal quantity of the product in their wallet to trade. To prevent double spending, the wallet will be consist of open transactions and transactions in the blockchain. For instance, if the user have 10 of ‘product001’ in the wallet, and he made a transaction, sending Qty 7 of ‘product001’ to another wallet, he cannot send any Qty more then 3 of ‘product001’, even though the previous transaction might not have been mined into a block.

‘/transaction’ [POST]

{ “recipient”: “public key”,

“productID”: “ xxxx…”,

“amount”: 0

}

**Reading open transactions:**

Append transactions that have not been added to the blockchain. The list is refresh after the transactions have been added to the block.

‘/transactions’ [GET]

Searching for transactions with a certain product ID:

Return a list of transactions with the required product ID.

‘/products’ [GET]

{“productID”: “xxxx”}

**Mining block:**

Only admin accounts can add new a new block. The new block will have a ‘signature transaction’ where the signature of the miner is recorded.

‘/mine’ [POST]

**Printing the blockchain:**

To print the blockchain,

‘/chain’ [GET]

**Adding nodes into the network:**

To add nodes,

‘/node’ [POST]

{“node”: “URL”}

**Deleting nodes in the network:**

‘/node/<node\_url>’ [DELETE]

**Listing nodes in the network:**

‘/nodes’ [GET]