

Cryptocurrency Project

Prepared By Group 6

How to run the Program

The program is a java code that is built using the Netbeans IDE.

The jar files are located on the folder “jars”

The nodes are run as mentioned in the RUN.md

1. Run master-node.jar
2. Then run relay-node.jar (Relay 1)
3. To start another Relay Node (Relay 2): run another relay-node.jar
4. Run miner.jar
5. Get the miner ID: (For example the id here is 30)

```
The artifact org.apache.commons:commons-io:jar:1.3.2 has been relocated to commons-  
--- exec-maven-plugin:1.2.1:exec (default-cli) @ cryptocurrency ---  
this is the original Private key  
MIHHAgEAMIGoBgcqhkJ0OAQBMIGcAkEA/KaCzo4Syrom78z3EQ5SbbB4sF7ey80etKI864WF64B81uRpH9  
Dec 11, 2017 6:58:36 PM com.ulb.cryptography.network.MinerClient main  
INFO: my address is: adfe78c06a1cd5dde0a915e8cd9f6b4f743fccd1 my id is: 30  
Dec 11, 2017 6:58:36 PM com.ulb.cryptography.network.MinerClient main  
SEVERE: Now using host=localhost, portNumber=[I@2133c8f8
```

6. To create the first block of the chain and to mine it, from the miner options choose 1.
Request transactions...

```
Options:  
1.- Request transactions  
2.- Login  
3.- request blockchain  
0.- Exit
```

This will create a new block add the reward transaction to it and mine it. After done mining this new block is sent to the relay node which forwards it to the master node. Now the master node checks that the miner did the proof of work and that the reward transaction is valid... So

it validates the block and if it's valid it creates the block chain and sends it to relay nodes and thus to miners.

7. Now the miner is rewarded and it has the first created coins so it can do a transaction...

To test: Create new Wallet by running wallet.jar (Wallet 1)

8. From wallet options create new address

```
Options:
1.- Create new addresses
2.- Login
3.- Full copy of the blockchain
0.- Exit
```

Enter the password to create the new address. Keep wallet id and address.

```
3.- Full copy of the blockchain
0.- Exit
1
password:
123
this is the original Private key
MIHGAgEAMIGoBgcqhkjOOAQBMIgCkEA/KaCzo4Syrom78z3EQ5SbbB4sF7ey80etK
your idAccount is: 17,5455e919ddaec20c8fa5cf22643fbd677a9e59fb
```

(For example here the wallet id is 17 and the address is

5455e919ddaec20c8fa5cf22643fbd677a9e59fb)

9. Now, transfer some coins (that for the first time only exists in the miner) from miner to the wallet. To do that, on the miner options choose 2.Login then enter miner id and password (default password is "miner")

```
Options:
1.- Transfer
2.- Balance
0.- Logout
```

Now from these options choose 1. Transfer. Now enter the receiver address (Wallet address that we kept before) and the quantity to send.

```

1
Addressee to transfer
5455e919ddaec20c8fa5cf22643fbd677a9e59fb
Quantity
0.5|

```

10. Now, this block needs to be mined... Start new miner by running miner.jar (Miner 2) and choose the first option 1. Request Transactions...

From now on the remaining functionality can be tested by creating other wallets and creating new transactions.

From a wallet: to create a transaction after creating the address:

- Choose 1.Transfer
- Enter receiver address
- Enter the amount to be sent

This transaction (and others if exist) will wait now until a miner mines the new block. To mine the pending transactions, from the miner choose 1.Request Transaction.

We can check the balance for each wallet and miner (where the miner has a wallet) by logging in and choose the second option 2. Balance, to make sure that the transactions are done correctly.

We've implemented the following Functionality:

Functionalities	
Wallet	
create an address (and associated pairs of keys)	ok
create a transaction (and sign it)	ok
store encrypted keys (private key)	ok
decrypt a key with a password	ok
request a copy of the block chain	ok
Relay	
there are several nodes	ok
IP addresses are publicly known	ok
keep an updated copy of the blockchain	ok
send minded blocks	ok
forward a copy of the blockchain to the wallets	ok
receive transactions requests from wallets	ok
send transactions to the miners	ok

Miners	
create valid blocks	ok
request transactions from the relay nodes	ok
perform proof-of-work (validate transactions)	ok
update blockchain	ok
receive a reward	ok
Master	
one master node	ok
store the blockchain	ok
check new mined blocks	ok
add the blocks to the block chain	ok
send an updated copy of the block chain to relay nodes	ok