FINAL ASSIGNMENT Part 2

|  |
| --- |
| BLOCKCHAIN TECHNOLOGY (INFBCT01) |

Educational Period 1 [2023-24]

|  |  |
| --- | --- |
| **GOODCHAIN** | A P2P Blockchain Wallet/Miner App |  |

The main objective of this assignment is to extend the functionality of [Final Assignment Part 1](https://hrnl-my.sharepoint.com/:w:/r/personal/bashb_hr_nl/Documents/Current%20Courses/BC/23-24-OP34/04_Final_Assignments/FA_1_BC_2023_OP34.docx?d=wee24c50be4d341ee955fd7db756cc800&csf=1&web=1&e=ljd5JY).

*Please note that you must first complete and get a PASS grade in Part 1. If you did not fulfil the FA Part 1, you should not submit Part 2 (It will not be graded).*

The description of the system and all previous requirements will mostly remain as they were, except the extra requirement described below:

* Nodes run in different machines.

At any specific time, some nodes are active and some nodes are inactive. A general overview of the system is shown in the Figure below. As depicted, each node has its own local copy of the ledger, the pool and user database. However, these data need to be synced, by a consensus mechanism.

Diagram

Description automatically generated

The main goal is to implement synchronization. For example:

1. Once a transaction is created, it should be sent to all nodes on the network. All active nodes should receive the transactions, validate them, and store them on the pool.
2. Once a node mined a block, the block should be sent to all nodes. All active nodes should receive the block, validate it and flag it as valid. In this case, the block should be added to the local copy of the ledger.

In general, all previous functions of the system should be working on a network of nodes.

|  |
| --- |
| Submission |

## 

## What to deliver?

The delivery to be handed in must consist of one zip-file, named as below:

**Studentnumber1\_ Studentnumber2\_firstname1\_lastname1\_ firstname2\_lastname2.zip**

For example:

**1019972\_0990112**\_**Geert\_Jan \_Lexmond\_Marco\_de\_Boer**.**zip**

**The zip-file must contain:**

A main directory called **goodchain** containing 2 sub-directories as below:

1. A sub-directory called **src**, containing all the **code files**, including one main file **goodchain.py**. Starting the system should be done by running **goodchain.py**.
2. A sub-directory called **data**, containing all the **data files**.

Please note that the code must read/write only from/on the goodchain directory and its subdirectory, if needed. Do not attempt to read or write on any other locations on the disk.

## How to deliver?

The submission will be through the assignment channel of the course team in MS Teams.

## Deadline

The deadline is announced in MS Teams.

You have a maximum of two chances for each part of the final assignment to get “sufficient” grade.

There are 4 chances in total for the submission of the final assignments, as given in the table below. Red cells show “insufficient”, and green cells show “sufficient”.

|  |  |  |  |
| --- | --- | --- | --- |
| **2023-2024** | | | |
| **OP1** | **OP2** | **OP3** | **OP4** |
| FA1 (1st) | FA2 (1st) |  |  |
| FA1 (1st) | FA1 (2nd) | FA2 (1st) |  |
| FA1 (1st) | FA2 (1st) | FA2 (2nd) |  |
| FA1 (1st) | FA1 (2nd) | FA2 (1st) | FA2 (2nd) |

If you could not get sufficient through any of the given paths, you need to retake the course, and resubmit all different components, again. The submission deadline will be announced in the course team on MS Teams.

|  |
| --- |
| ⚠ IMPORTANT NOTES  1. **Do not** include any **bulky** Python system files in the delivery. 2. In this assignment, in addition to the **standard library modules** of Python, you canuse **cryptography, pickle, sqlite3, socket, threading, select, random, re, sys, os,** andany third-party package for functions related to the user interface. If you need to use any other extra package, you may ask your teacher to check if it is allowed. 3. The code must run **error-free** (on a standard Windows or MAC PC). |