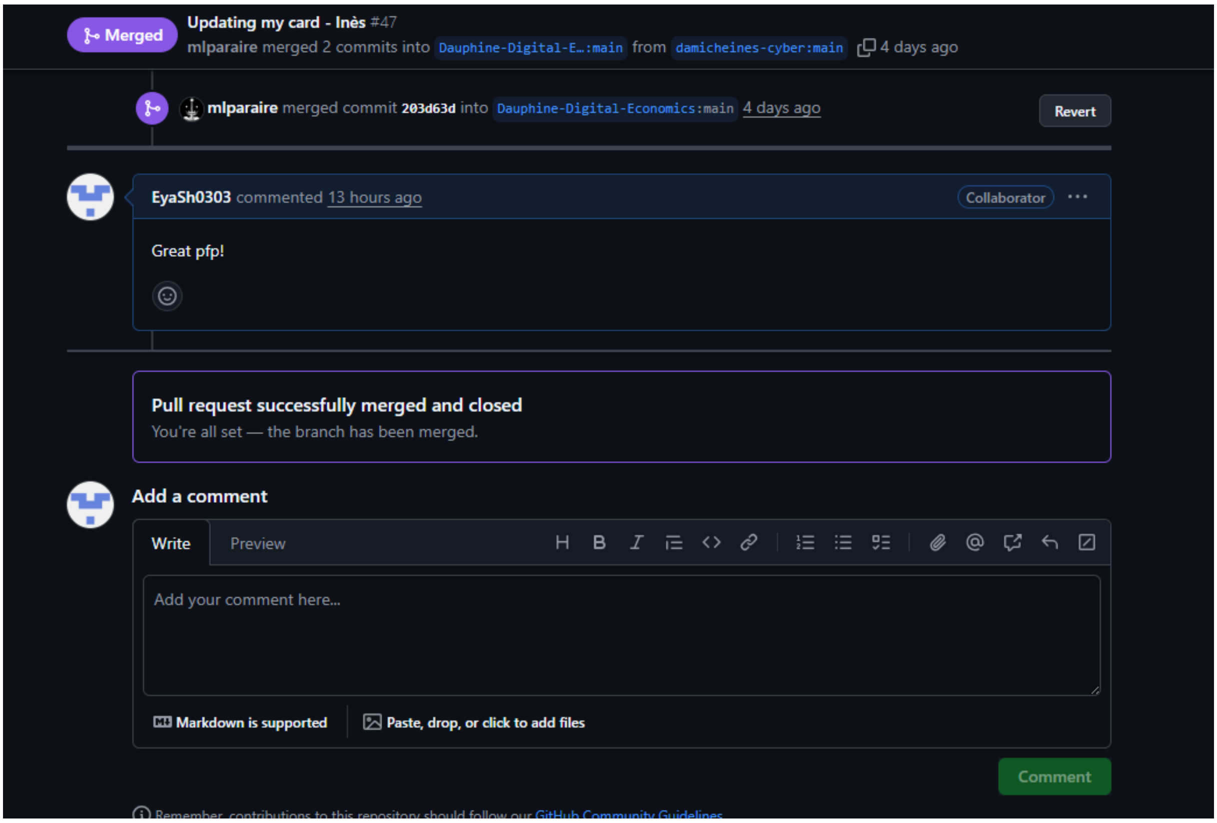
Exercise 1

4. As I was not assigned as a reviewer, I was able to leave a comment:

Updating my card - Inès by damicheines-cyber · Pull Request #47 · Dauphine-Digital-Economics/Class-Directory



Updating my card - Inès #47

mlparaire merged 2 commits into Dauphine-Digital-E…:main from damicheines-cyber:main 4 days ago

mlparaire merged commit 203d63d into Dauphine-Digital-Economics:main 4 days ago

EyaSh0303 commented 13 hours ago

Great pfp!

Pull request successfully merged and closed

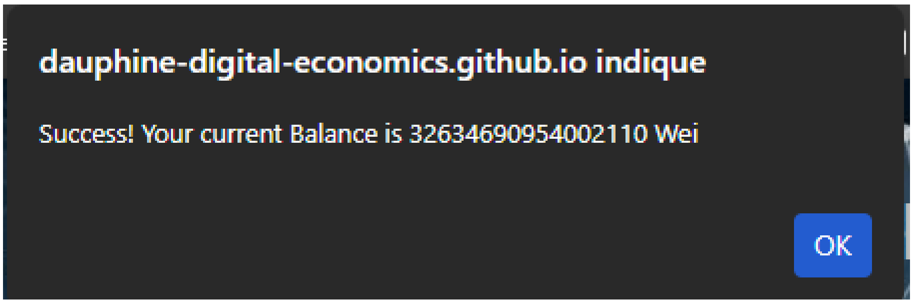
You're all set — the branch has been merged.

Add a comment

Add your comment here...

Remember, contributions to this repository should follow our GitHub Community Guidelines

5. When pressing the "check metamask balance" button, the balance displayed is 32634690954002110 Wei. This value is correct because MetaMask usually displays balances in Ether, while the smart contract returns the balance in Wei, the smallest unit of Ether (with 1 eth = 10^18 Wei). Converting the value gives around 0.0326 ETH, which matches the expected balance of the connected account on the selected network.



dauphine-digital-economics.github.io indique

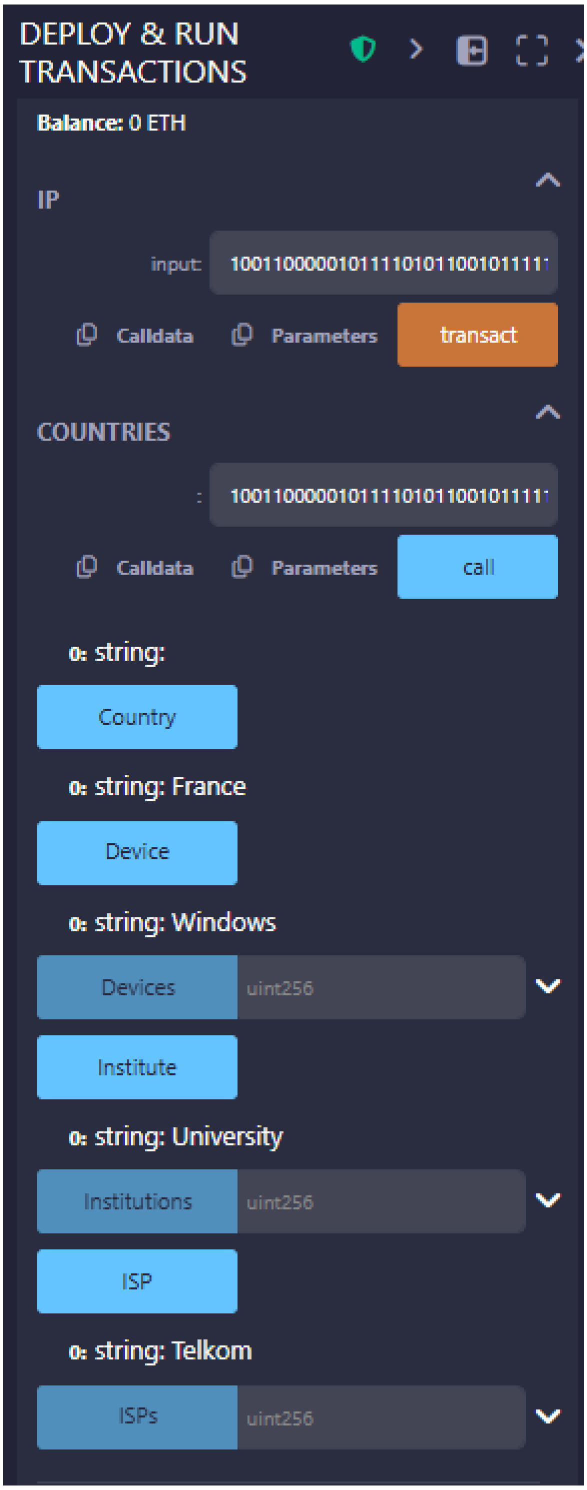
Success! Your current Balance is 32634690954002110 Wei

Exercise 2

Part A:

Github uses RSA (and SSH authentication) to authenticate a user by keeping the private key on the local machine and sharing only the public key with Github: during connection, it verifies the user's identity by checking a signature created with the private key. In blockchain systems, private keys are used to sign transactions that anyone can verify using the public key. SSH authentication is adapted for secure server access, while blockchain signatures are used for decentralized transaction verification.

Part B:



Balance: 0 ETH

input: 1001100000010111101011001011111

: 1001100000010111101011001011111

0: string:

0: string: France

0: string: Windows

0: string: University

0: string: Telkom

Exercise 3

1. During the simulation, the network was composed of multiple nodes acting as classmates. While all nodes followed the same general rules, I believe they didn't

behave in a coherent or reliable way. Some nodes experienced delays and failed to propagate information correctly or appeared inconsistent compared to the majority. From these observations that were transmitted to me from my peers, it seems like not all peers can be assumed to be honest or reliable at all times and that reliability is inferred by observing which nodes consistently shared the same information as others and responded within a reasonable time frame: blockchain systems should be designed to tolerate errors or slow nodes rather than rely on “trust”.

2. I understood, from what was reported that the simulation illustrated different topologies (centralized structures, decentralized peer-to-peer networks). In more centralized setups, information spreads quickly and consensus was reached faster, but the system depended heavily on a small number of nodes. However, decentralized systems are slower and require more communication, but they were far more resilient to individual node failures like a real blockchains, where decentralization improves security and error tolerance. The consensus mechanisms also reflects this trade-off: simpler rules allowed faster agreement, while more robust mechanisms increased reliability but required additional coordination.

3. Several parameters strongly influence network performance during a simulation. One could be latency (higher latency can slow consensus: larger networks require more time and messages to reach agreement), the number of nodes, and error tolerance.

   From a consultant’s perspective, the first questions to ask when designing a blockchain protocol would be
   - How decentralized does the system need to be?
   - How many malicious nodes must it tolerate?
   - What level of performance does it require?

   There needs to be a balance between security, decentralization and efficiency (performance).

Sources:

- Nakamoto, S. (2008) - Bitcoin: A Peer-to-Peer Electronic Cash System
- Buterin, V. (2014) - Ethereum Whitepaper