Data security course project

2022-23

ONE CLICK LOGIN

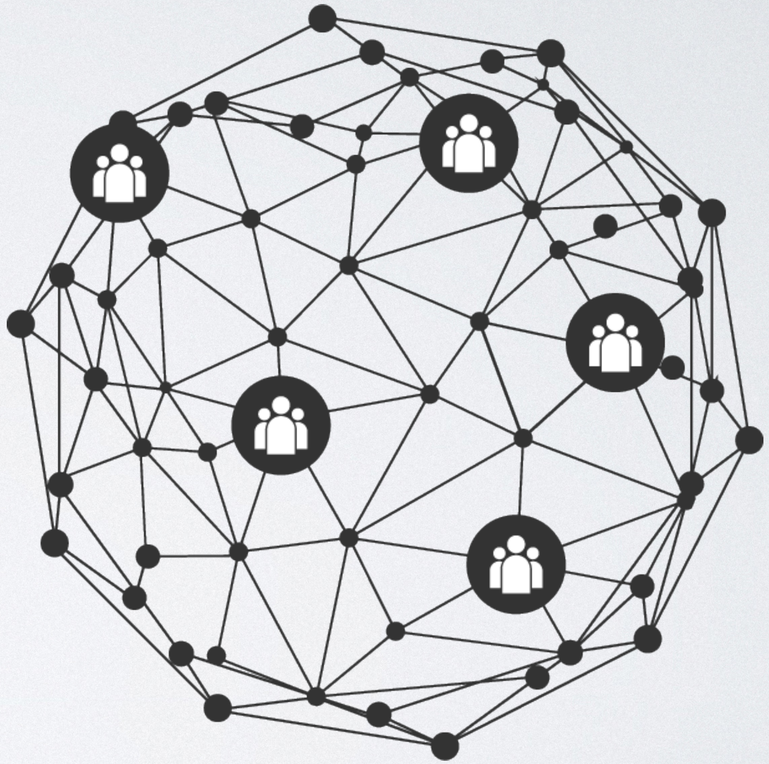
A

Decentralized application for authentication

Marco Amorosi

DECENTRALIZED APP

- Decentralizzate
- Sicurezza
- Trasparenza
- Censorship-resistance
- Interoperabilità



AUTH-APP

Registrazione e login tramite wallet metamask

Vantaggi:

- Non dover inserire/ricordare ID e Pass
- Velocizzare il processo singin/up

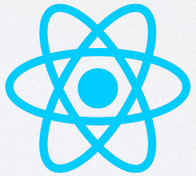
TECHNOLOGIES

Truffle

Ganache

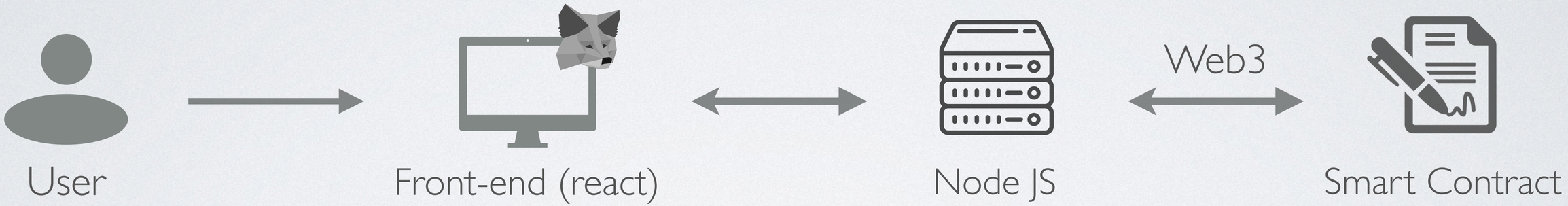
Metamask

Node JS

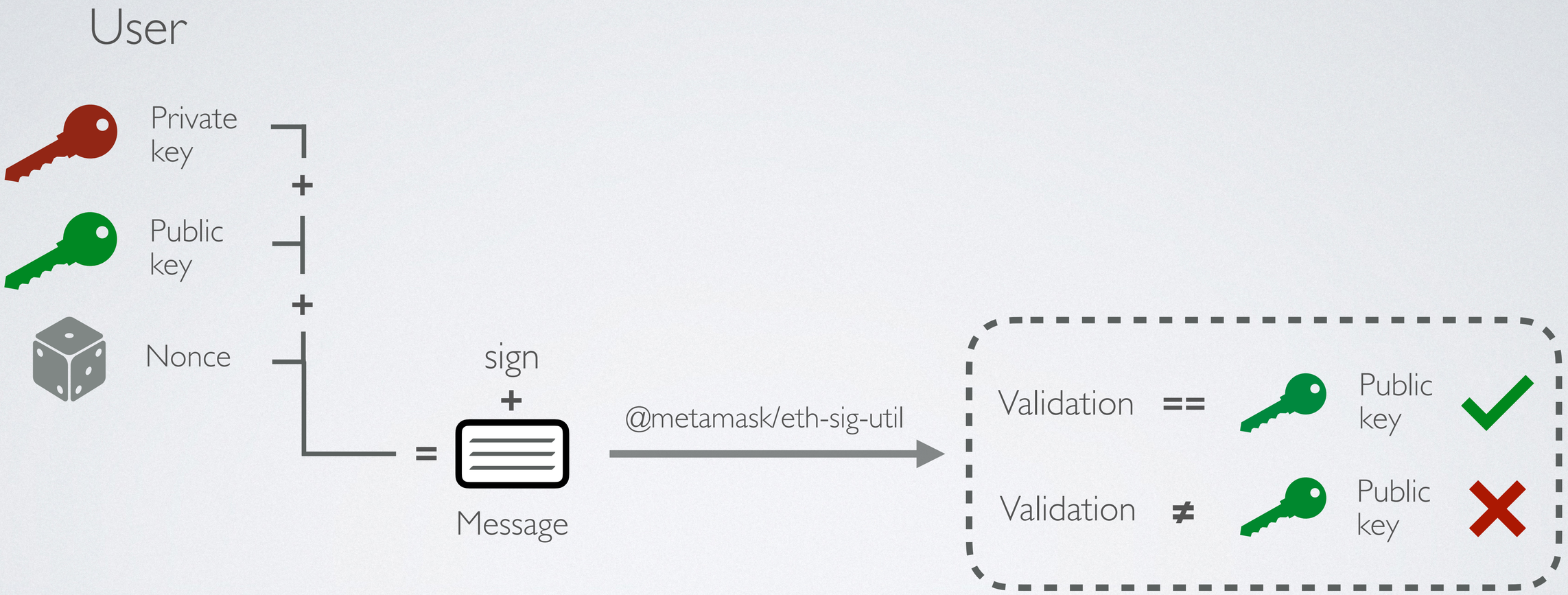


React

ARCHITECTURE



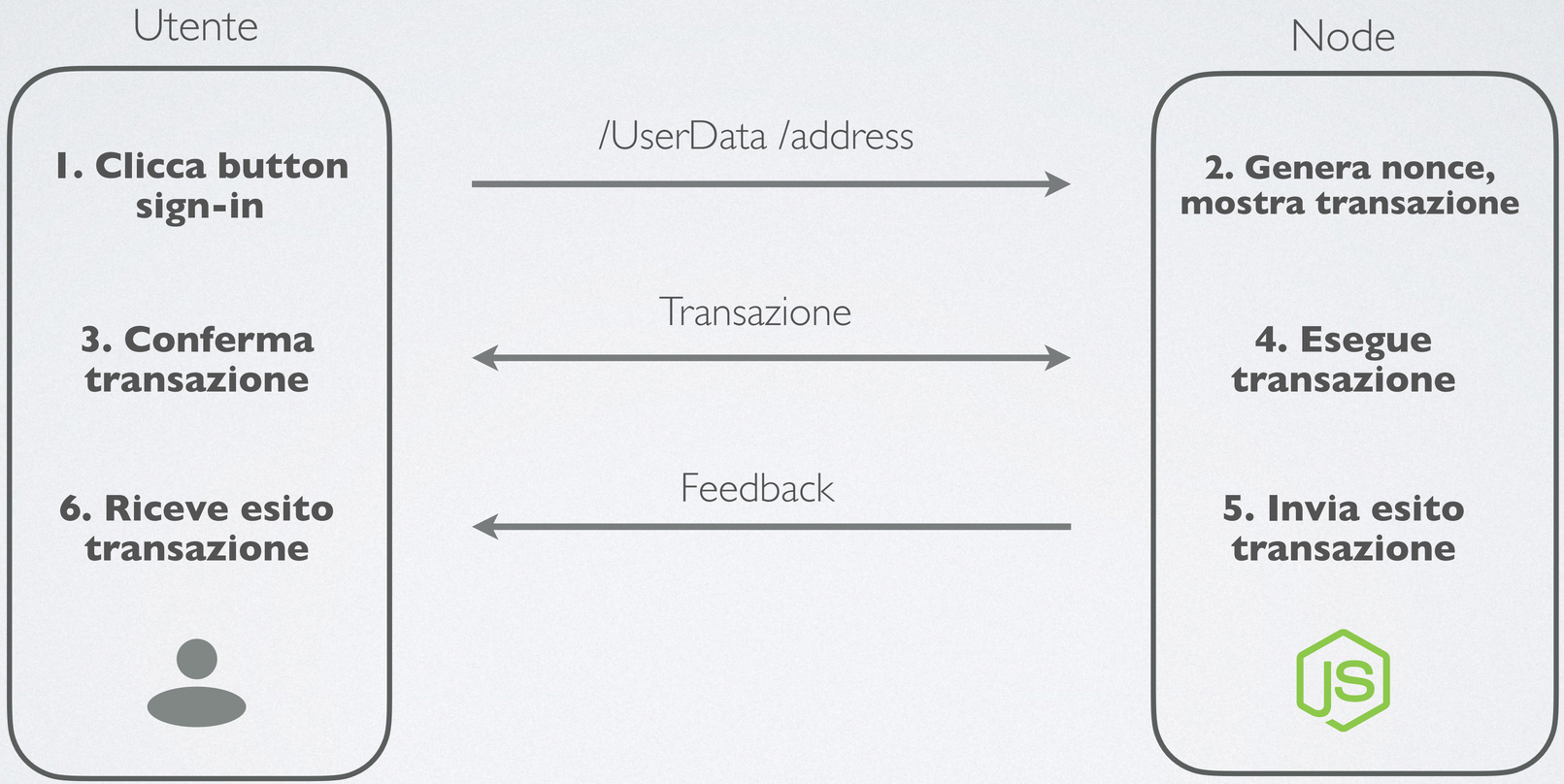
CRYPTOGRAPHY



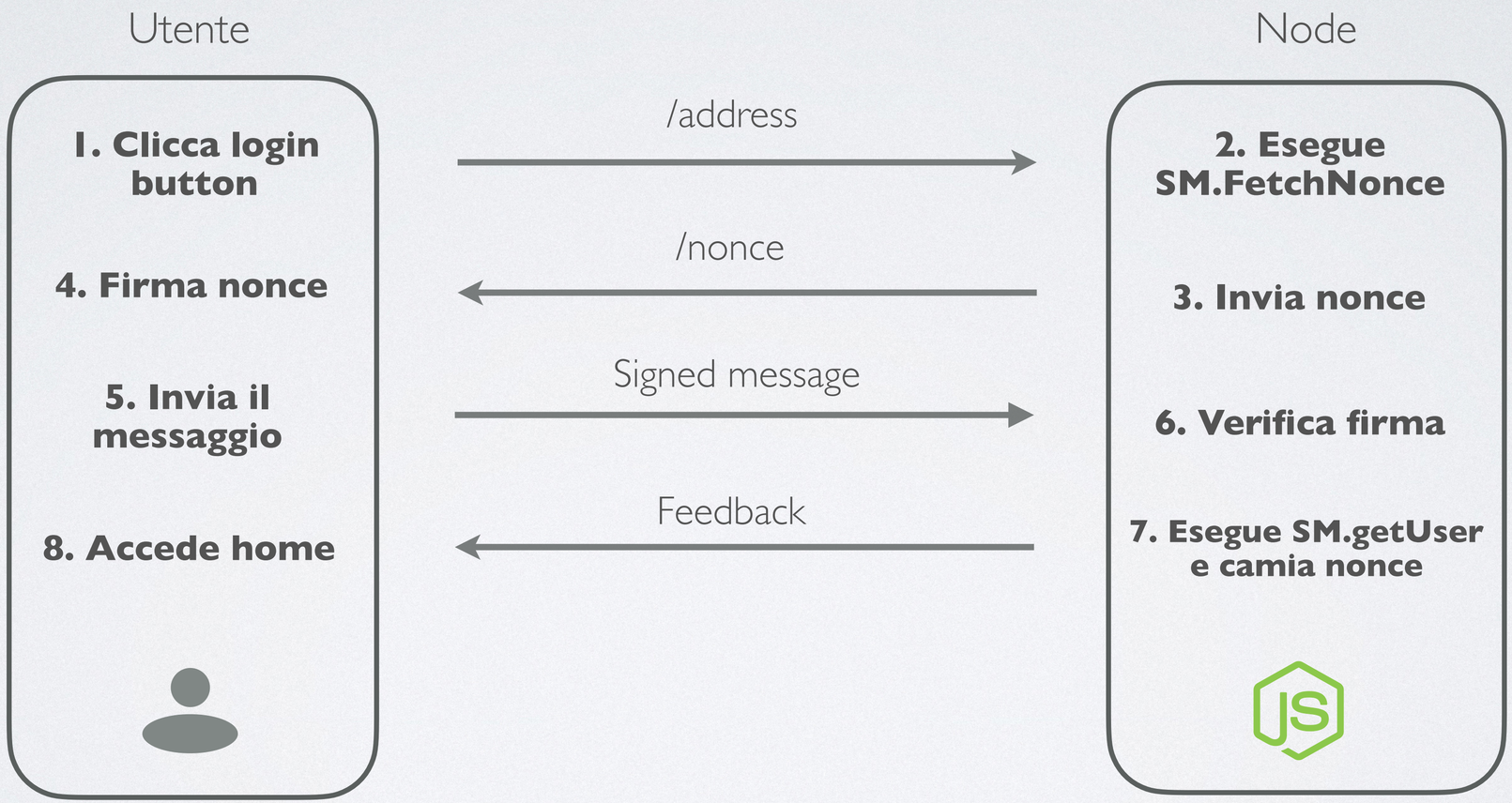
SMART CONTRACT

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.9;

contract Auth  {

    mapping(address => user) public usersList;

    struct user {
        string username;
        address addr;
        bytes32 nonce;
    }

    modifier onlyOwner(address _address)

    function getUserAndUpdateNonce(address _address) public onlyOwner(_address) returns (string memory)

    function generateRandomSequence() public view returns (bytes32)

    function createUser(string memory _username, address _address, string memory _nonce) public

    function fetchNonce(address _address) public view returns (bytes32)

    function deleteUser(address _address) public onlyOwner(_address)

}
```

SIGN-UP FLOW


Utente
1. Clicca button sign-in
3. Conferma transazione
6. Riceve esito transazione
/UserData /address
Transazione
Feedback
Node
2. Genera nonce, mostra transazione
4. Esegue transazione
5. Invia esito transazione
JS

SIGN-IN FLOW


Utente
Node
1. Clicca login button
4. Firma nonce
5. Invia il messaggio
8. Accede home
/address
/nonce
Signed message
Feedback
2. Esegue SM.FetchNonce
3. Invia nonce
6. Verifica firma
7. Esegue SM.getUser e camia nonce

PRODUCTION-READY

- Sicurezza: basata su proof of key encryption
- User-friendly: velocizza il processo in un paio di click
- Privacy: non utilizza e-mail, password, n°cellulare