

# DAO

Realizzato da Samuele Furnari

Project

# DAO

01

Il progetto realizzato consisteva nella realizzazione di una Decentralized Autonomous Organization per DnA, un'azienda specializzata nella produzione di contenuti digitali. DnA ritiene che la community sia di fondamentale importanza per il raggiungimento dei propri obiettivi, per questo desidera coinvolgere ancora di più gli utenti, dando loro maggior voce e partecipazione attraverso strutture governative come le DAO.

# Hardhat & Solidity

Le tecnologie chiave utilizzate per la realizzazione di questo progetto sono: Hardhat e Solidity. Grazie a Hardhat, possiamo automatizzare i processi di sviluppo, testing e deployment, migliorando l'efficienza e la sicurezza del nostro progetto. Solidity, rappresenta il linguaggio chiave per la scrittura dei nostri contratti intelligenti, garantendoci la massima compatibilità e funzionalità nella blockchain su cui operiamo (Ethereum).

# Etherscan

03

Il contratto da me sviluppato è visibile su Etherscan, una piattaforma che fornisce informazioni approfondite sulla blockchain di Ethereum. Oltre a essere uno strumento di esplorazione della blockchain, Etherscan agevola la visualizzazione e la verifica degli smart contract, compresi quelli che sono stati distribuiti su reti di testnet. Questo fornisce ai programmatore uno strumento indispensabile nel ciclo di sviluppo degli smart contract.

# Smart Contract

Nel contesto gestionale inerente alla struttura governativa della DAO, ho sviluppato quattro contratti intelligenti che sono fondamentali per il funzionamento ottimale della gestione della struttura. Questi contratti intelligenti rappresentano l'ossatura della nostra DAO, garantendo trasparenza, sicurezza e conformità nelle operazioni governative. Approfondiremo l'importanza dei quattro contratti nelle prossime slide.

# GovernanceToken.sol

05

Questo contratto costituisce il cuore della DAO, consente l'acquisizione di GT (GovernanceToken) fondamentali per la partecipazione ai processi decisionali dell'organizzazione. Gli utenti (nella fase di vendita) potranno acquistare da 1 a 5 GT, che successivamente verranno utilizzati per creare proposte (5 GT) o votare le stesse (1 GT). L'owner detiene il potere di terminare la vendita dei token inizializzando così la fase successiva.

# Proposal.sol

06

Al termine della vendita dei token, l'attenzione si sposta su questo contratto. Gli utenti avranno la possibilità di presentare proposte a un costo di 5 GT, successivamente le proposte verranno rese disponibili per il voto. È importante dire che quando un utente invierà una proposta, l'account verrà automaticamente contrassegnato come "votante", indicando che ha già partecipato al processo decisionale della struttura governativa DAO.

# Vote.sol

07

Dopo la chiusura della fase di presentazione delle proposte, la DAO procede alla fase di voto in cui i detentori di GT esercitano i loro diritti di voto, esprimendosi pro, contro o astenendosi a proposte specifiche, identificate dai loro ID univoci. È importante sottolineare che l'astensione dal voto non è automatica. In caso di non partecipazione entro il termine delle votazioni, i GT verranno rimossi dall'account dell'utente non votante.

# Executive.sol

08

Dopo la conclusione delle votazioni, la DAO entra nella fase finale in cui vengono determinati ed eseguiti i risultati delle proposte. In questa fase, è obbligatorio eseguire tutte le proposte create per finalizzare il "ciclo" decisionale, infatti la vendita dei GT (e tutto ciò che ne deriva) non potrà riaprirsi se tutte le proposte non verranno eseguite. Tutti i membri hanno l'autorità di eseguire le proposte, verificandone l'assenso o il rifiuto.

Project Link

# GitHub Repository

<https://github.com/samuelefrni/DAO>

09

Il progetto da me realizzato costituisce la prova finale del corso "Smart Contract con Solidity Advanced". La sua natura pratica mi ha offerto un'occasione unica per mettere in pratica le conoscenze acquisite durante lo studio in un contesto concreto e tangibile, fornendomi una prospettiva pratica fondamentale per comprendere appieno la tecnologia e consolidarla, acquisendo sicurezza nella programmazione con Solidity.