



SocialTrust

**Soluzione blockchain per la tracciabilità e
validità della diffusione dei contenuti
online**

Arto Manuel

Relatore: Dott. Luca Sciallo

Correlatore: Dott. Federico Montori

Alma Mater Studiorum - Università di Bologna

Diffusione della disinformazione

- Velocità di propagazione
- Fake News
- Deepfakes
- Agenzie di fact-checking limitate

Obiettivi della tesi

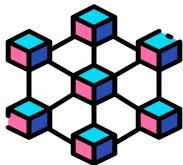


1. Design e implementazione di un framework basato su blockchain per tracciare la diffusione e l'origine dei contenuti
 - 1.1. Studio e realizzazione di un sistema di fact-checking decentralizzato
 - 1.2. Design di un meccanismo di ricompense e penalizzazioni
 - 1.3. Analisi della resistenza alle manipolazioni

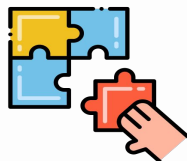


2. Validazione in un'applicazione di messaggistica
 - 2.1. Analisi del funzionamento del sistema

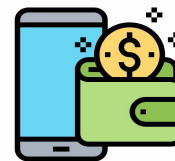
SocialTrustr



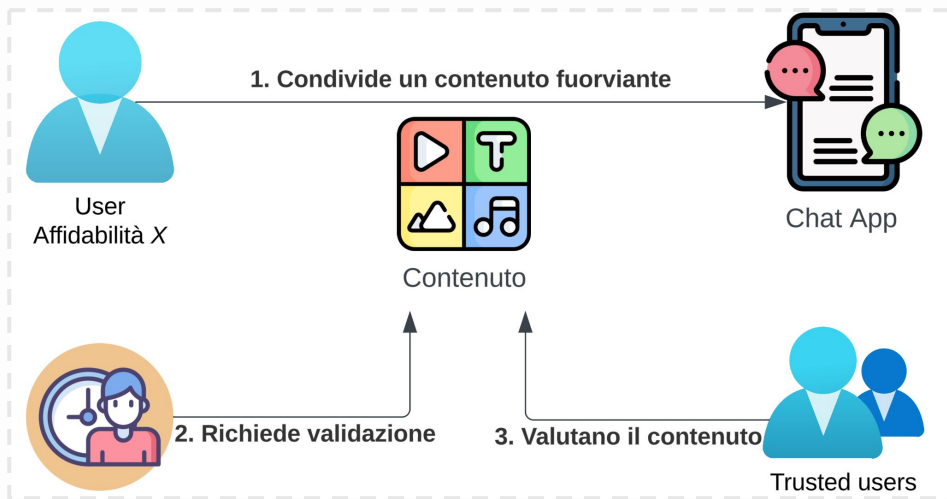
Realizzata su
blockchain



Ideata come
framework



Integrabile in
qualunque app social



Esempio d'uso

Componenti del sistema



Sistema di storage e
ottenimento dati



Sistema di voting



Modello predittivo
probabilistico

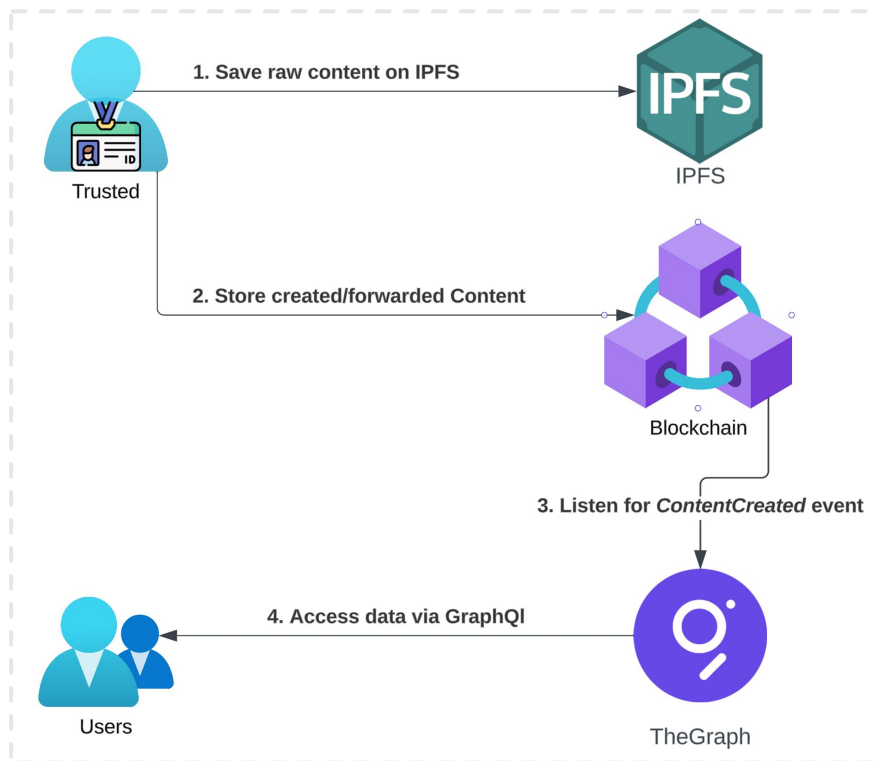
Storage e ottenimento dati



1. Immutabilità dei dati



2. Tracciamento della diffusione dei contenuti



Flusso storage e ottenimento dati

Sistema di voting



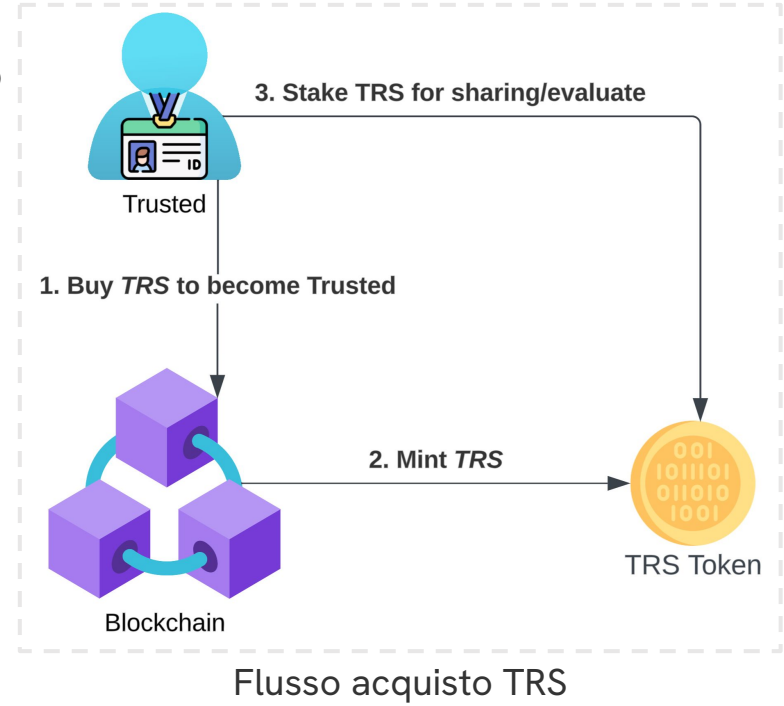
1. Realizzare un meccanismo di consenso resistente alle manipolazioni



2. Creare un Token come incentivo economico



3. Separare quantità di token e affidabilità di un utente



Resistenza alle manipolazioni



Sybil attack

Creazione di identità multiple per manipolare il sistema

Prevenzioni:

- L'iscrizione ha un costo iniziale
- Le operazioni richiedono lo staking di TRS
- La creazione di più utenti risulta costoso



51% Attack

Ottenere 51% del potere decisionale

Prevenzioni:

- Risulta difficile aumentare la propria affidabilità in autonomia
- Ricompense inversamente proporzionali alla affidabilità
- Penalizzazioni più drastiche delle ricompense

Modello predittivo probabilistico



1. Valutazioni tramite affidabilità e un confidence score



2. Redistribuzione dei token TRS in stake



3. Uso dell'entropia per regolare affidabilità degli utenti

$$Punishment = \sum_{i \in T_F} (Stake * Conf_i)$$

$$Reward_i = \frac{Conf_i}{Tot_{Conf}} * Punishment \forall T_T$$

Redistribuzione TRS

$$Punishment_i = AF_i * Conf_i * (1.0 - Entropy) \forall T_F$$

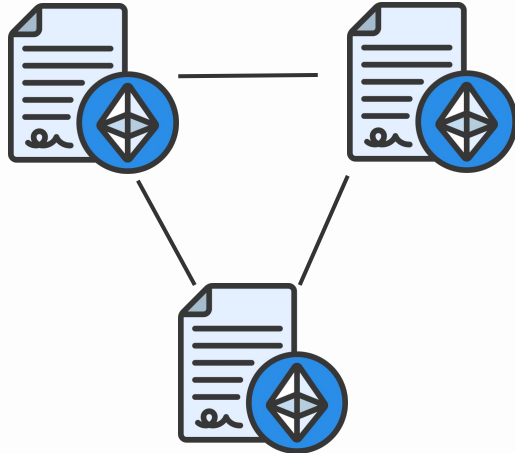
$$Reward_i = \frac{(100 - AF_i) * Conf_i * (1.0 - Entropy)}{M} \forall T_T$$

Regolazione affidabilità

Implementazione

Blockchain Application

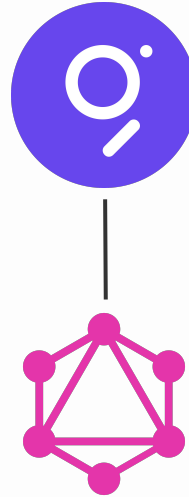
ContentSharing ContentEvaluation



TrustToken

Subgraph Application

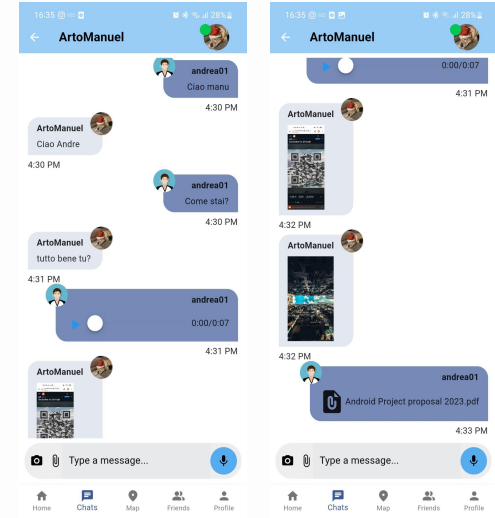
The Graph



GraphQL

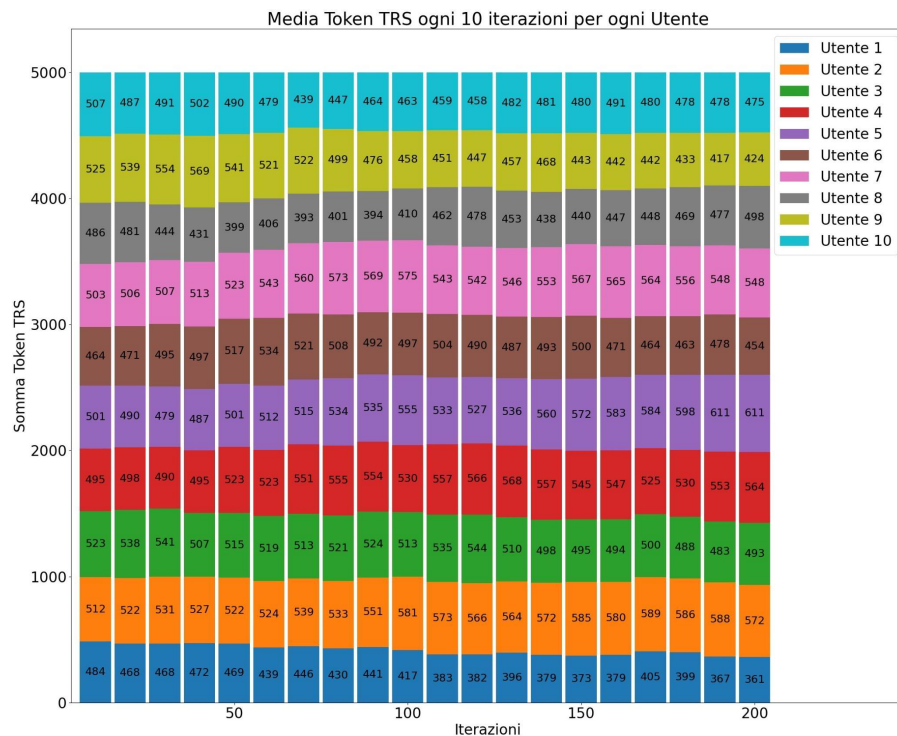
Mobile App Integration

Chat app

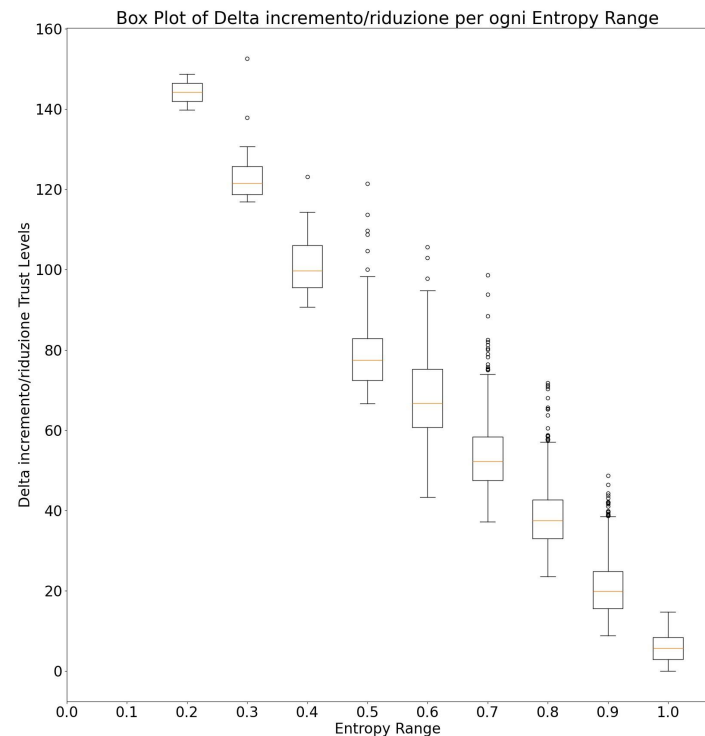


Validazione

Redistribuzione TRS su 10 utenti e 200 iterazioni



Entropia e delta incremento/riduzione affidabilità su 10 utenti e 10.000 ripetizioni



Conclusione

- Monitorare diffusione dei contenuti
- Immutabilità e trasparenza dei dati
- Potere in mano agli utenti
- Evitare la diffusione di fake news
- Tracciare affidabilità degli utenti

Sviluppi futuri



Verifica in scenari con utenti reali



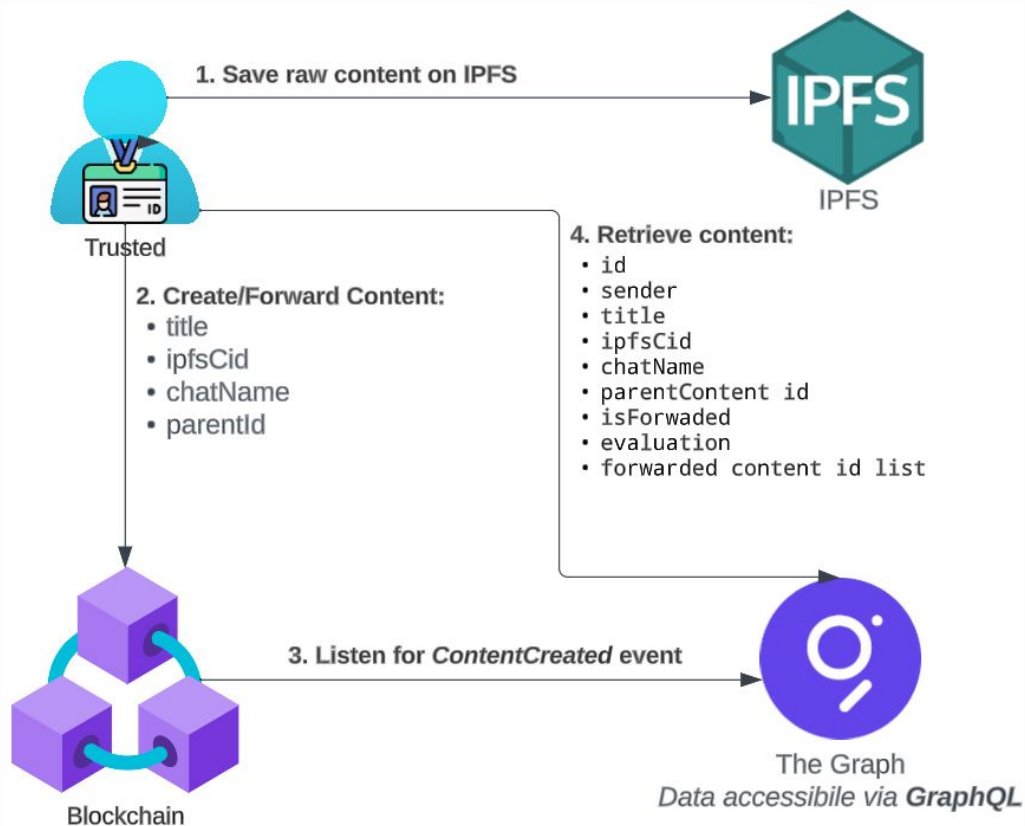
Interoperabilità su diverse blockchain



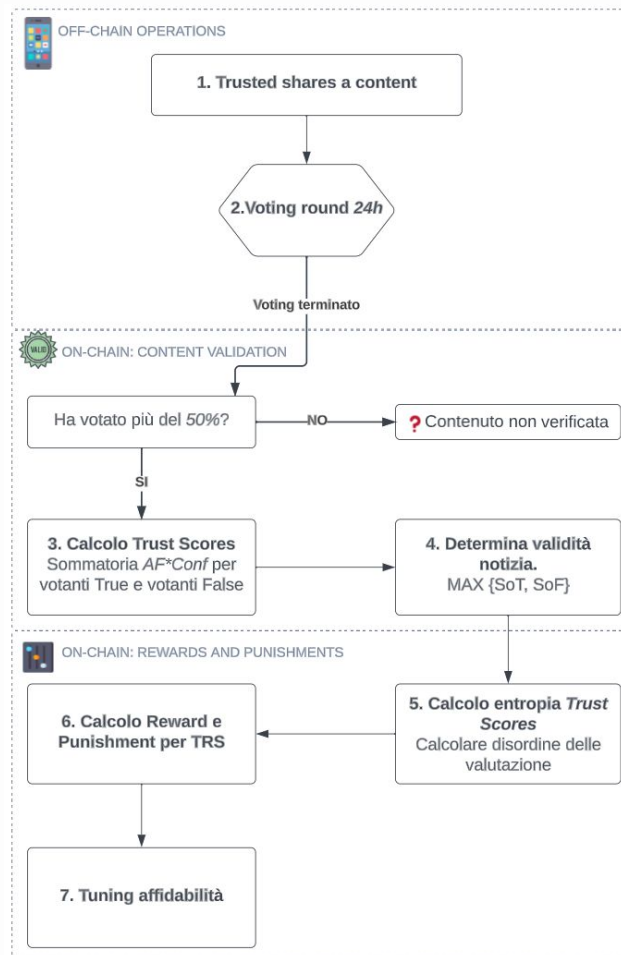
Estendere l'applicazione su più piattaforme social

Grazie dell'attenzione

Flusso condivisione di un contenuto

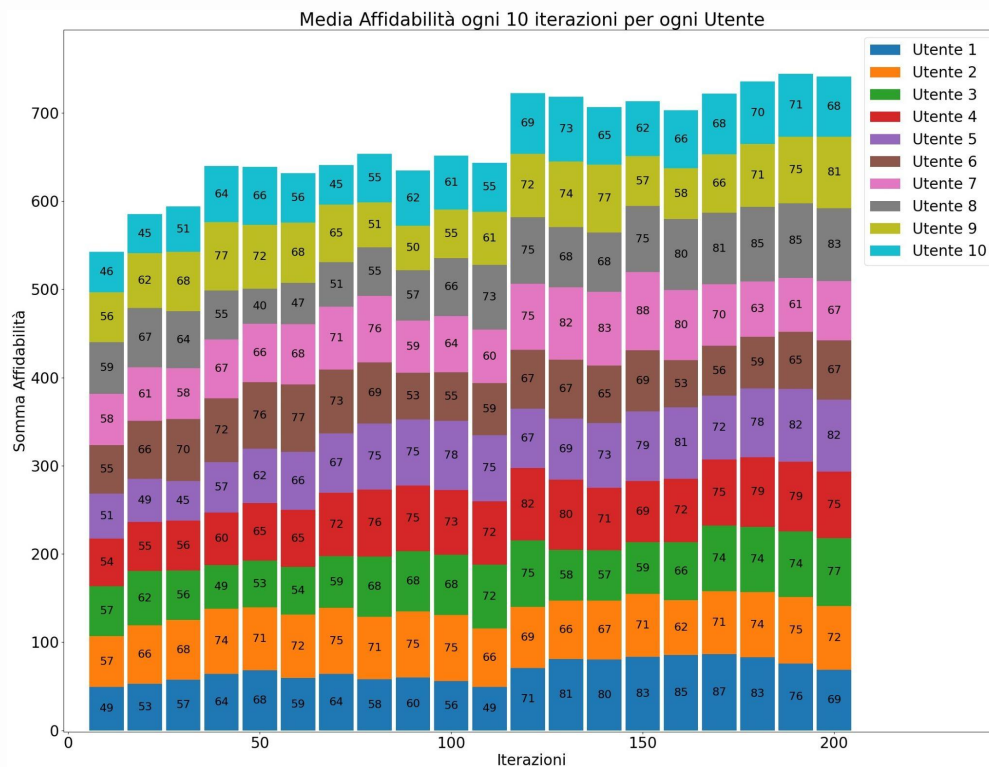


Flusso validazione di un contenuto



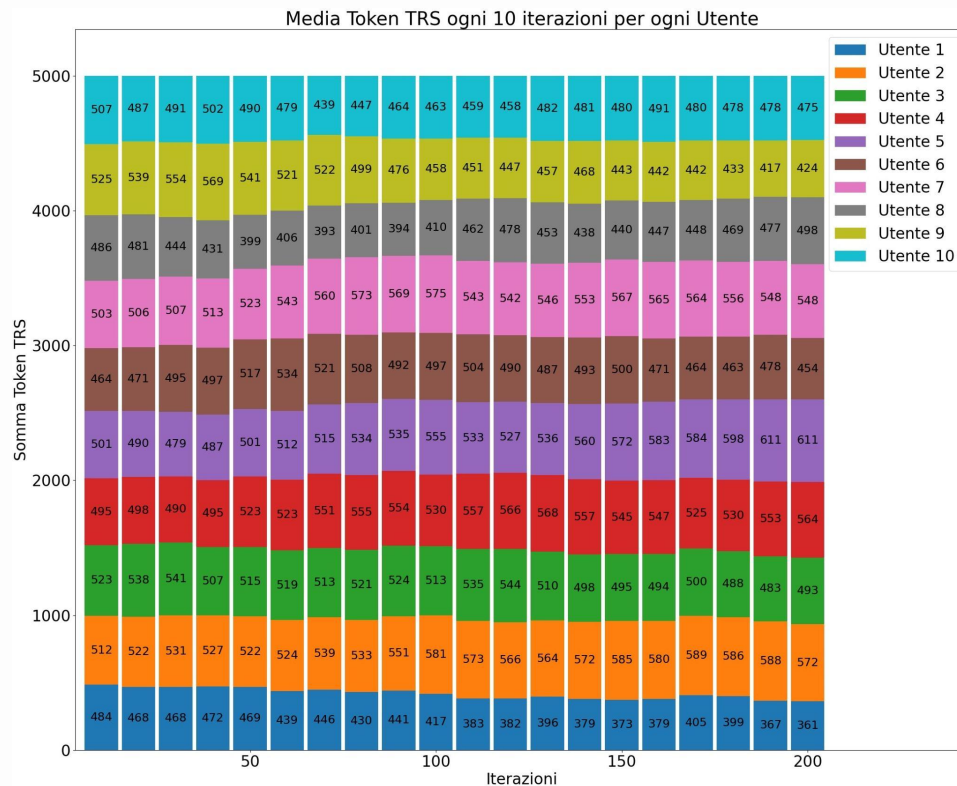
Simulazione Montecarlo

Affidabilità su 10 utenti e 200 iterazioni



Simulazione Montecarlo

Redistribuzione TRS su 10 utenti e 200 iterazioni



Simulazione Montecarlo

Entropia e delta incremento/riduzione affidabilit a su 10 utenti e 10.000 ripetizioni

