



POLITECNICO DI BARI

DIPARTIMENTO DI INGEGNERIA ELETTRICA E DELL'INFORMAZIONE
CORSO DI LAUREA TRIENNALE IN INGEGNERIA INFORMATICA E
DELL'AUTOMAZIONE

Chain4Good

*Progettazione e sviluppo di una piattaforma di
crowdfunding con tecnologia blockchain*

Candidati:

Angelica DE FEUDIS
Johnatan CAPUTO
Luca GENTILE

Docente:

Prof.ssa Marina
MONGIELLO

Academic Year: 2025/2026

Indice

Acronimi	iii
1 Introduzione	1
2 Background	2
2.1 Blockchain	2
2.1.1 Transazioni	2
2.1.2 Blocchi	2
2.1.3 Protocolli di consenso	3
2.1.4 Vantaggi della blockchain	4
2.2 Ethereum Blockchain	4
2.2.1 Smart Contract	5
2.2.2 DApps	6
2.2.3 DAO	6
3 Metodologia di progetto	7
3.1 Modello di processo	7
3.2 Pianificazione delle attività	7
3.3 Analisi dei rischi	7
3.4 Stima dei costi	7
4 Progettazione e implementazione	8
4.1 L'obiettivo di Chain4Good	8
4.2 Analisi dei requisiti	8
4.3 Analisi SWOT	8
4.4 Architettura del Software	8
5 Prototipo	9
5.1 Dashboard	9
5.2 Creazione progetto	9
5.3 Inserimento e valutazione spesa	9
6 Validazione e discussione	10
6.1 Valutazione dell'applicazione	10
6.2 Realizzazione dei requisiti	10
7 Conclusioni e sviluppi futuri	11
Riferimenti bibliografici	12

Elenco delle figure

Elenco delle tabelle

Acronimi

DApps Decentralized Applciations.

EOA Externally Owned Account.

EVM Ethereum Virtual Machine.

P2P Peer-to-Peer.

PoS Proof of Stake.

PoW Proof of Work.

1 Introduzione

2 Background

2.1 Blockchain

Una blockchain è una base di dati distribuita, condivisa e immutabile, mantenuta da una rete di nodi interconnessi secondo un'architettura [Peer-to-Peer \(P2P\)](#). Dal punto di vista dei sistemi distribuiti, essa si configura come un sistema che garantisce affidabilità e sicurezza replicando i dati e le operazioni su molteplici nodi indipendenti [1].

Tale paradigma architetturale è stato introdotto per la prima volta nel 2008 da Satoshi Nakamoto - pseudonimo utilizzato dall'autore o dal collettivo di persone dietro il *white paper* di Bitcoin - come infrastruttura di supporto per la nota criptovaluta.

Come suggerito dal termine stesso, la blockchain è una catena sequenziale di blocchi, ciascuno contenente un insieme di transazioni validate dalla rete. Ogni blocco è collegato al precedente attraverso un riferimento crittografico, detto *hash*. Questa caratteristica rende il registro intrinsecamente resistente alla manomissione: qualsiasi tentativo di alterare le informazioni contenute in un blocco comporterebbe, infatti, l'invalidazione a cascata di tutti i blocchi successivi [2].

2.1.1 Transazioni

La transazione rappresenta l'unità fondamentale di informazione all'interno della blockchain. Essa descrive un'operazione richiesta da un utente, come ad esempio il trasferimento di un asset digitale (*token*) o l'invocazione di uno *Smart Contract* [3]. Ogni transazione è creata da un nodo e viene firmata digitalmente utilizzando una coppia di chiavi crittografiche (crittografia asimmetrica).

Prima di essere registrata nella blockchain, ogni transazione viene validata dalla rete secondo regole condivise e, una volta confermata, viene inserita in un blocco.

2.1.2 Blocchi

Un blocco è una struttura dati che aggrega un insieme di transazioni validate in un determinato intervallo di tempo. Oltre alle transazioni, ogni blocco contiene un *header* con informazioni di gestione fondamentali, tra cui:

- l'*hash* del blocco precedente, che collega crittograficamente i blocchi tra loro formando la catena;
- un *timestamp*, che indica il momento di creazione del blocco;
- il *Merkle Root*, ovvero la radice di un *Merkle Tree*, una struttura ad albero binario che consente di riassumere tutte le transazioni del blocco in un unico *hash*;
- dati aggiuntivi richiesti dal protocollo di consenso adottato.

L'inclusione dell'*hash* del blocco precedente rende la blockchain una struttura immutabile per costruzione: anche una minima modifica a una singola transazione altererebbe la *Merkle Root* e di conseguenza l'*hash* del blocco, invalidando l'intera catena successiva. Questo meccanismo costituisce uno dei principali fattori di sicurezza della blockchain [2].

2.1.3 Protocolli di consenso

Come precedentemente affermato, la blockchain è un sistema intrinsecamente decentralizzato, per cui la validazione delle transazioni non dipende più da un'autorità centrale ma è subordinata all'approvazione collettiva da parte tutti i nodi della rete. Il processo di validazione di un blocco impone dunque l'implementazione di algoritmi di consenso, come:

- **Proof of Work (PoW)**: rappresenta il protocollo di consenso più noto e storicamente rilevante (introdotto da Bitcoin). In questo meccanismo, i nodi della rete, detti *miner*, competono per risolvere un problema crittografico computazionalmente complesso ma facilmente verificabile. Il *miner* che per primo individua una soluzione valida al problema, acquisisce il diritto di creare un nuovo blocco e propagarlo alla rete. Tuttavia, l'aggiunta effettiva del blocco avviene solo se questo viene correttamente validato e accettato dalla maggioranza dei nodi. Se queste operazioni vanno a buon fine, il *miner* riceve una ricompensa, generalmente erogata sotto forma di criptovaluta [4].
- **Proof of Stake (PoS)**: in questo protocollo, il processo di validazione è affidato ad un insieme di nodi scelti in funzione della quantità di criptovaluta vincolata come garanzia, in un processo noto come *staking*. I nodi selezionati come *validatori* sono responsabili della creazione dei blocchi e della verifica delle transazioni. La sicurezza della rete è garantita da un sistema di disincentivi economici: qualora un validatore agisca in modo fraudolento o tenti di compromettere il sistema, incorre nella perdita parziale o totale dei

fondi posti in *staking*. Una volta raggiunto il consenso, il blocco viene aggiunto immutabilmente alla blockchain, aggiornando lo stato del *ledger* distribuito.

2.1.4 Vantaggi della blockchain

L'adozione della tecnologia blockchain offre vantaggi significativi rispetto ai sistemi centralizzati tradizionali, derivanti dalla sua architettura distribuita e dall'uso della crittografia:

- **Decentralizzazione:** l'assenza di un'autorità centrale elimina i *single point of failure* [1];
- **Immutabilità:** ogni transazione è irreversibile, per cui una volta registrata nella blockchain, non può più essere cancellata o modificata, in quanto farlo altererebbe l'intera catena dei blocchi [2];
- **Trasparenza:** tutte le transazioni sono verificabili pubblicamente (nelle *blockchain permissionless*) o dai membri autorizzati (nelle *permissioned*) [3];
- **Sicurezza:** l'uso combinato di crittografia asimmetrica (per l'autenticazione) e protocolli di consenso distribuito (per l'integrità del registro) rende il sistema resistente ad attacchi e frodi;

2.2 Ethereum Blockchain

Ethereum è una piattaforma decentralizzata, *open-source* basata su blockchain, proposta da Vitalik Buterin nel 2013 e resa operativa con il primo rilascio stabile nel 2015 [5].

A differenza delle blockchain di prima generazione, concepite prevalentemente come registri distribuiti per il trasferimento di valore (come bitcoin), Ethereum nasce come una piattaforma *general-purpose* in grado di superare alcune limitazioni strutturali, come la mancanza di Turing-completezza, e supportare l'esecuzione di applicazioni decentralizzate, le [Decentralized Applications \(DApps\)](#).

A tal fine, Ethereum mette a disposizione l'[Ethereum Virtual Machine \(EVM\)](#), una macchina virtuale decentralizzata all'interno della quale vengono eseguiti gli *Smart Contract* utilizzando la potenza computazionale dei nodi che costituiscono la rete.

2.2.1 Smart Contract

Gli *smart contract* sono programmi immutabili memorizzati sulla blockchain di Ethereum. Essi consentono di definire ed applicare automaticamente regole e condizioni logiche, rendendo possibile la gestione di risorse digitali e l'esecuzione di accordi senza il ricorso a intermediari centralizzati. Come illustrato nel *whitepaper* che ha introdotto questa tecnologia, gli *smart contract* possono essere concepiti come entità crittografiche capaci di detenere e trasferire valore, il cui comportamento è rigidamente definito dal codice e la cui esecuzione è consentita esclusivamente al verificarsi delle condizioni prestabilite [6]. Per supportare la loro esecuzione, Ethereum adotta un modello basato su *account*, distinguendo tra:

- *Externally Owned Account (EOA)*: comunemente identificati come *wallet*, sono controllati direttamente dagli utenti tramite una coppia di chiavi crittografiche: una chiave privata, utilizzata per firmare digitalmente le transazioni, e una chiave pubblica, che funge da indirizzo dell'account sulla rete. Per effettuare una transazione è necessario che il mittente conosca l'indirizzo, ossia la chiave pubblica del *wallet* del destinatario, e che firmi digitalmente la transazione prima di inviarla alla rete Ethereum con la propria chiave privata.
- *Contract Accounts*: sono account associati a uno *smart contract* e non sono controllati da una chiave privata. Essi non possono avviare transazioni, ma vengono attivati esclusivamente in risposta a chiamate provenienti da un *EOA* o da un altro *smart contract*.

La validazione delle transazioni e l'esecuzione degli *smart contract* è affidata all'*EVM*.

Ethereum dispone di una criptovaluta nativa, l'Ether, che oltre a poter essere scambiato fra account, è generato dalla piattaforma stessa come ricompensa ai *miner* per il lavoro computazionale svolto. Ciascuna transazione generata sulla rete Ethereum, sia essa un semplice trasferimento di Ether, o l'invocazione di una funzione di uno *smart contract*, comporta un costo che è proporzionale alla complessità computazionale, alla banda utilizzata e/o alla quantità di *storage* necessario. Questo meccanismo interno adottato da Ethereum prende il nome di "gas".

Per ogni transazione l'utente può definire un *gas price* e un *gas limit*, ovvero rispettivamente l'ammontare che decide di pagare per ogni unità di gas consumata (valore espresso solitamente in Gwei, un sottomultiplo dell'Ether) e la quantità massima di gas consumabile. In questo modo il mittente della

transazione conosce a priori quale sarà il costo massimo dell'operazione, e garantisce al tempo stesso che ogni computazione giungerà sempre a termine. Qualora infatti si verificasse ad esempio un loop infinito all'interno del codice di uno *smart contract*, la computazione verrà interrotta non appena si sarà consumato tutto il gas messo a disposizione al momento della creazione della transazione. Il costo effettivo sarà determinato dall'effettiva quantità di gas utilizzato moltiplicato per il *gas price* indicato e quindi detratto al mittente dal saldo del suo wallet in Ether.

2.2.2 DApps

Le **DApps**, sono applicazioni decentralizzate che operano su una rete **P2P** anziché su server centralizzati. Una **DApps** è qualcosa di più rispetto ad un contratto intelligente. I vantaggi rispetto ad un'applicazione centralizzata sono [20]:

- Trasparenza: ogni nodo della rete può esplorare il codice ed essere più sicuro;
- Resistenza alla censura: finché un nodo è connesso alla rete Ethereum, potrà interagire con essa senza interferenze.
- Resilienza: una DApp non avrà tempi di inattività, verrà eseguita fino a quando la piattaforma sarà operativa

2.2.3 DAO

3 Metodologia di progetto

3.1 Modello di processo

3.2 Pianificazione delle attività

3.3 Analisi dei rischi

3.4 Stima dei costi

4 Progettazione e implementazione

4.1 L'obiettivo di Chain4Good

4.2 Analisi dei requisiti

4.3 Analisi SWOT

4.4 Architettura del Software

5 Prototipo

5.1 Dashboard

5.2 Creazione progetto

5.3 Inserimento e valutazione spesa

6 Validazione e discussione

6.1 Valutazione dell'applicazione

6.2 Realizzazione dei requisiti

7 Conclusioni e sviluppi futuri

Riferimenti bibliografici

- [1] R. Rodrigues e P. Druschel, «Peer-to-peer systems,» *Communications of the ACM*, vol. 53, n. 10, pp. 72–82, 2010.
- [2] A. A. Monrat, O. Schelén e K. Andersson, «A survey of blockchain from the perspectives of applications, challenges, and opportunities,» *Ieee Access*, vol. 7, pp. 117 134–117 151, 2019.
- [3] M. Crosby, P. Pattanayak, S. Verma, V. Kalyanaraman et al., «Blockchain technology: Beyond bitcoin,» *Applied innovation*, vol. 2, n. 6-10, p. 71, 2016.
- [4] F. Tschorsch e B. Scheuermann, «Bitcoin and beyond: A technical survey on decentralized digital currencies,» *IEEE Communications Surveys & Tutorials*, vol. 18, n. 3, pp. 2084–2123, 2016.
- [5] V. Buterin et al., «A next-generation smart contract and decentralized application platform,» *white paper*, vol. 3, n. 37, pp. 2–1, 2014.
- [6] V. Buterin et al., «Ethereum white paper,» *GitHub repository*, vol. 1, n. 22-23, pp. 5–7, 2013.