

# POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Gestionale

Tesi di Laurea Magistrale

## Blockchain for Digital Handshake



**Relatrice**

Dott. Gatteschi Valentina

**Correlatore**

Prof. Lamberti Fabrizio

**Tutor Aziendale**

Dott. Favenza Alfredo



**Candidata**

Infantino Claudia

Dicembre 2020

## Indice

1.	Introduzione .....	4
2.	Blockchain: la tecnologia dall'enorme potenziale .....	5
2.1	Principi di funzionamento della Blockchain .....	6
2.1.1	Dai sistemi centralizzati ai sistemi distribuiti .....	7
2.1.2	Il Funzionamento .....	9
2.1.3	La firma digitale e il sistema delle chiavi .....	12
2.1.4	Algoritmi e protocolli di consenso .....	13
2.2	Tipologie di Blockchain: pubblica, privata e consortium .....	15
2.2.1	Blockchain Privata .....	15
2.2.2	Blockchain Pubblica .....	16
2.2.3	Blockchain Consortium .....	16
2.2.4	Le tre tipologie di Blockchain a confronto .....	17
2.3	Smart Contract .....	18
2.3.1	Cenni Storici .....	18
2.3.2	Che cosa è uno Smart Contract .....	19
2.4	I Token .....	19
2.4.1	Le tipologie di Token .....	20
3.	E-Commerce B2B: I Trend del Settore, Analisi e Prospettive .....	21
3.1	Il "Digital Handshake" .....	21
3.2	La Reputazione Digitale .....	22
3.3	Mercato Business-to Business (B2B) .....	25
3.3.1	I Prodotti .....	25
3.3.2	Le Relazioni .....	26
3.3.3	Le Transazioni .....	26
3.3.4	Differenze tra B2B e B2C .....	27
3.3.5	TOE Framework .....	28
3.3.6	Trend del Mercato B2B .....	30
3.4	Individuazione del Target di Beneficiari .....	32
3.4.1	Caso d'uso: Fornitura di prodotti a chilometro zero .....	34
3.4.2	Caso d'uso: I freelancers e il mondo della Gig Economy .....	36
3.5	Analisi degli Stakeholders .....	40
3.5.1	Stakeholders Esterni .....	40
3.5.2	Stakeholders Interni .....	41
3.5.3	Matrice per l'Analisi degli Stakeholders .....	41

3.6	Analisi della Concorrenza .....	42
3.6.1	Tabella di Benchmarking .....	44
4.	<b>DLT-based Digital Handshaking: la nuova piattaforma per la stratta di mano digitale</b> .....	47
4.1	Criticità degli accordi non-digital .....	47
4.2	Soluzione Proposta.....	49
4.2.1	Perché EOS (EOS vs Ethereum).....	49
5.	<b>Sviluppo dell'Applicazione</b> .....	53
5.1	Architettura del sistema.....	53
5.2	Diagrammi UML.....	54
5.2.1	Use Case Diagram.....	54
5.2.2	Activity Diagram .....	58
5.2.3	Class Diagram .....	64
5.3	Smart Contract .....	65
5.3.1	Strumenti utilizzati .....	65
5.3.2	Sviluppo Smart Contract.....	67
5.4	Studio dei Mockup.....	72
5.4.1	Strumento utilizzato e analisi effettuate.....	72
5.4.2	Raffigurazione Mockup.....	73
5.5	Sviluppo Front-end .....	75
5.5.1	Tecnologie e Strumenti .....	75
5.5.2	Cenni di funzionamento di Redux e React.js.....	76
5.5.3	Struttura dell'applicazione.....	78
5.5.4	Interazione con la Blockchain EOS .....	80
5.5.5	Descrizione Interfacce .....	81
5.5.6	Class Diagram Web Application.....	91
5.5.7	DB Server .....	92
6.	<b>Conclusioni e Prospettive Future</b> .....	94
6.1	Conclusioni .....	94
6.2	Futuri Sviluppi: i Contratti Ricardiani .....	96
6.2.1	Le Caratteristiche del Contratto Ricardiano .....	97
6.2.2	Comparazione tra Contratto Ricardiano, Contratto Intelligente e Contratto Tradizionale .....	99
	<b>Sitografia</b> .....	101
	<b>Ringraziamenti</b> .....	104

# 1. Introduzione

Dopo poco più di dieci anni dalla sua nascita, la Blockchain si presenta come una tecnologia utilizzabile e sfruttabile in diversi ambiti. Ragion per cui, il presente lavoro di tesi mira ad applicare tale tecnologia alle transazioni del mercato business-to-business (B2B), in gran parte ancora radicate agli strumenti e alle metodologie del passato.

Il presente lavoro di tesi, svolto presso l'azienda Fondazione LINKS, si inquadra all'interno di un progetto che vede la collaborazione di Fondazione LINKS, del centro di ricerca milanese, CEFRIEL, e della multinazionale in ambito di trasformazione digitale, GFT, con l'obiettivo di fornire un nuovo modo sicuro, veloce e affidabile di gestire contratti e accordi tra piccoli fornitori/venditori ed i loro clienti mediante tecnologia Blockchain EOS.

Il presente lavoro di tesi è organizzato nel seguente modo:

- i) nel capitolo 2 vengono descritti il funzionamento ed i principi alla base della Blockchain, al fine di comprendere l'enorme potenziale di tale tecnologia;
- ii) il capitolo 3 illustra i risultati di analisi condotte in merito ai trend del mercato B2B, ai possibili casi d'uso, agli attori coinvolti e, per concludere, viene presentata un'analisi di benchmarking;
- iii) nel capitolo 4, si introduce il modello proposto e sviluppato nel presente lavoro di tesi, completo delle problematiche che mira a risolvere e della tecnologia utilizzata (Blockchain Eos);
- iv) nel capitolo 5, si descrivono dettagliatamente le diverse fasi di sviluppo della soluzione (DApp). Si partirà dalle analisi progettuali, seguite dalla scrittura di uno Smart Contract, fino allo studio dei mockup e allo sviluppo del front-end;
- v) nel capitolo 6, si analizzano le sfide e le criticità ancora aperte nella soluzione e si riportano le possibili applicazioni future.

## 2. Blockchain: la tecnologia dall'enorme potenziale

Con il termine Blockchain si intende quella tecnologia che consente la creazione e la gestione di un grande registro distribuito per la sincronizzazione di numerosi dati, condivisi tra più partecipanti, appartenenti alla stessa rete. Il concetto di Blockchain fu concepito e ideato nell'ottobre del 2008 da Satoshi Nakamoto, come fondamento del protocollo Bitcoin, al fine di automatizzare e decentralizzare in maniera sicura qualsiasi transazione digitale. Infatti, l'utilizzo di strumenti crittografici rende sicuri e immutabili i dati memorizzati su tale piattaforma.

La Blockchain, detta anche "tecnologia di codifica a blocchi", è una delle innovazioni tecnologiche più importanti avvenute in ambito informatico negli ultimi trent'anni. Essa, ha acquistato nel corso degli anni sempre più credibilità e si è proposta, in virtù del suo altissimo potenziale, come soluzione ed opportunità per svariati campi applicativi, quali ad esempio i servizi, la pubblica amministrazione, la finanza, la governance globale ecc.

La Blockchain è un Libro Mastro distribuito, ovvero un database di transazioni condiviso tra i nodi partecipanti al sistema. Tale libro non è tenuto da un'autorità centrale, garante della bontà di ciò che vi è scritto e delle transazioni memorizzate, ma è custodito, in modo paritario, da tutti i partecipanti (definiti "nodi") al sistema.

Nel 2015 l'Economist definisce tale tecnologia come "the trust machine" [1], poiché rappresenta la principale soluzione al ben noto problema del "trust gap", derivante dall'impossibilità di effettuare delle transazioni senza la presenza di una terza figura intermediaria (*middleman*), nella quale le parti riporranno la propria 'fiducia'. Il modello attualmente in uso risulta quindi essere inefficiente e costoso e pertanto destinato ad essere superato.

La blockchain non deve essere vista solo come una soluzione tecnologica, bensì come un nuovo approccio, decentralizzato, al concetto di fiducia (*trust*). Infatti, ogni utente connesso alla rete non ha bisogno di fidarsi della controparte, o di un intermediario centrale, per poter effettuare una transazione, ma dovrà affidarsi unicamente alla tecnologia, al protocollo su cui questa è strutturata ed alle logiche che regolano il funzionamento della piattaforma stessa. Fino ad oggi, il sistema centralizzato è stato il garante dell'univocità delle quantità di denaro spedite o ricevute tra gli utenti, ma adesso, il sistema distribuito, dà uguale potere a tutti i nodi della rete, che sono chiamati a concordare la veridicità e la correttezza delle informazioni trasferite. Queste ultime, se approvate, saranno pubbliche e immutabili [2]. Di conseguenza, se un gruppo di nodi

fosse corrotto o il messaggio fosse errato, la rete dovrebbe essere in grado di rilevare e bloccare tale anomalia.

Si prevede che la blockchain possa portare innovazioni sui sistemi di economie e società fondate su libri maestri centralizzati e rappresenterà nei prossimi anni un grande vantaggio competitivo, configurandosi come uno standard per diversi elementi, quali ad esempio la sicurezza e la trasparenza delle informazioni, l'efficienza e l'automazione dei processi e l'interoperabilità delle soluzioni e delle piattaforme.

## 2.1 Principi di funzionamento della Blockchain

Nel presente paragrafo viene illustrata l'infrastruttura alla base della tecnologia blockchain unitamente alle sue principali componenti.

L'infrastruttura elementare della blockchain consta di:

- Database distribuito: è una base di dati condivisa tra più computer connessi alla rete
- Protocollo: è l'insieme delle regole primarie di una blockchain
- Algoritmo di consenso: è il meccanismo attraverso il quale le regole presenti nel protocollo vengono fatte rispettare. Esso è distribuito su tutti i nodi che partecipano al processo di validazione delle transazioni
- Token: è un asset digitale il cui scambio può avvenire tra le parti senza la presenza di un intermediario

Le componenti necessarie al funzionamento della blockchain includono:

- Hash: funzione algoritmica informatica non invertibile, che mappa una stringa di lunghezza arbitraria in una stringa di lunghezza predefinita. Non è altro che una stringa alfanumerica che rappresenta le transazioni contenute in un blocco.
- Blocco: è il contenitore di un certo numero di transazioni ed è contrassegnato da un hash identificativo e da una sequenza di caratteri che rappresentano una data e un orario per accertare l'effettivo avvenimento di un dato evento (timestamp).
- Nodo: è uno dei partecipanti al network che contiene al suo interno le informazioni su tutte le transazioni avvenute all'interno del sistema
- Transazione: è il trasferimento di risorse digitali tra i nodi di una rete
- Ledger: è un libro maestro che contiene al suo interno lo stato condiviso dell'intera catena di blocchi e l'elenco di tutte le transazioni che sono state create.

- Miner: colui che convalida in modo volontario le transazioni e aggiunge i blocchi alla catena

### **2.1.1 Dai sistemi centralizzati ai sistemi distribuiti**

Come precedentemente descritto, le motivazioni che spingono verso lo sviluppo di tale tecnologia sono molteplici e tutte riconducibili ai problemi legati all'utilizzo dei sistemi centralizzati di seguito menzionati:

- problema della tendenza alla centralizzazione e all'accentramento di potere;
- vulnerabilità;
- neutralità della rete assente;
- elevati costi di manutenzione dei server centrali;
- scarsa sicurezza e resistenza ad attacchi informatici;

Nei sistemi centralizzati è presente una rete, caratterizzata dal rapporto Uno-A-Molti, nella quale tutti gli elementi sono connessi ad un unico nodo centrale, creando accentramento di potere. In questo tipo di sistemi, la fiducia risiede esclusivamente nell'autorità del soggetto, o del sistema, che occupa la posizione centrale, come rappresentato nel modello a sinistra di Figura 1. Quindi, il centro di comando è un punto di vulnerabilità che, in caso di malfunzionamento o manomissione, genererebbe effetti negativi sull'intera rete.

Similmente avviene nei sistemi decentralizzati (Figura 1, centro) in cui è riproposta la stessa logica vista in precedenza, ma a livello "locale", con la presenza di diversi nodi centrali connessi tra loro ed organizzati a loro volta come un sistema centralizzato. Anche in questo caso, la fiducia è delegata ai nodi centrali, tendendo a formare accentramento di potere. Il vero cambiamento è rappresentato invece dal sistema distribuito, in cui la struttura centralizzata è sostituita da un network peer-to-peer tra i nodi, con un rapporto Molti-A-Molti (Figura 1, destra). Infatti, il potere centrale è eliminato e ogni punto risulta connesso con un certo numero di punti, che a loro volta sono collegati con altri punti, fino a connettere l'intera rete. In tal modo ogni nodo è utile agli altri in quanto collabora nel replicare e condividere i dati che trafficano in rete, ma nessuno è indispensabile per il funzionamento della rete stessa.

La Figura 1, riporta il confronto grafico tra le strutture dei tre sistemi appena descritti.[3]

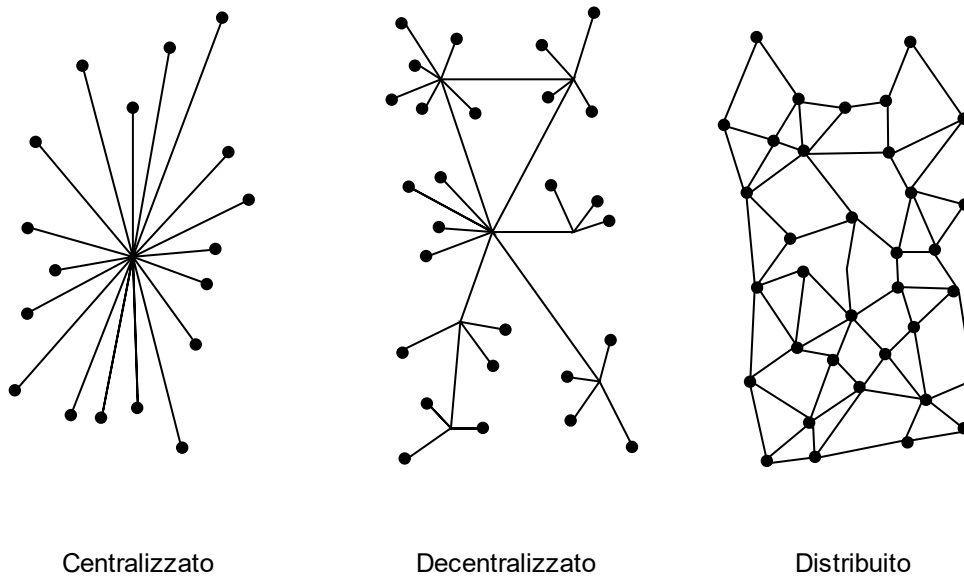


Figura 1: i diversi sistemi a confronto [3]

In una rete peer-to-peer ogni nodo è equivalente a tutti gli altri, e ha le stesse capacità e responsabilità degli altri. A differenza delle architetture client/server, in cui alcuni computer (server) sono destinati al solo soddisfacimento delle richieste dei clients, nel sistema peer-to-peer ciascun nodo ricopre sia il ruolo di client che quello di server, al fine di condividere le informazioni. In Figura 2, si riporta la differenza tra un sistema client/server e un sistema peer-to-peer.

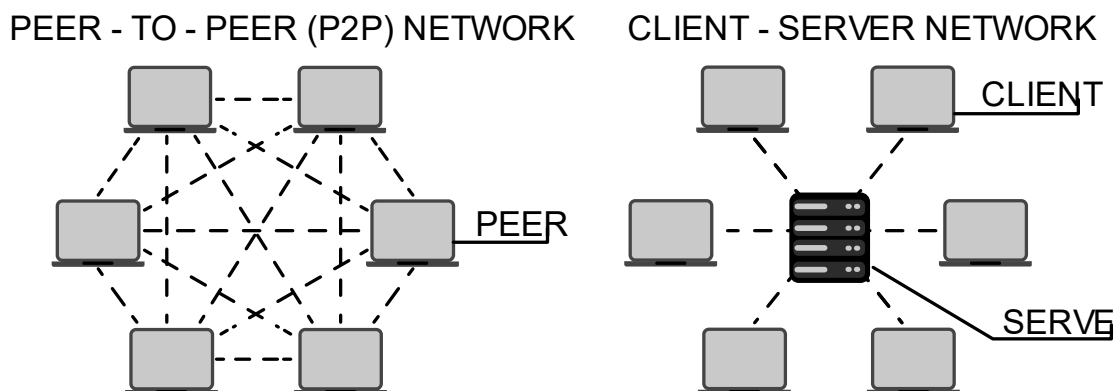


Figura 2: Confronto tra una struttura client/server e una peer-to-peer [4]



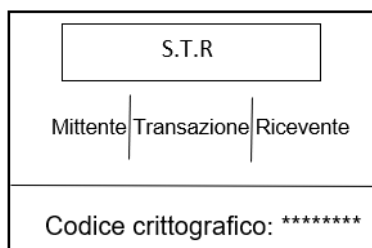
## 2.1.2 Il Funzionamento

L'obiettivo del presente paragrafo è la descrizione del funzionamento della tecnologia blockchain. A tal fine la blockchain pubblica "Bitcoin" viene utilizzata a titolo di esempio. È fondamentale tuttavia precisare che le funzionalità e i principi descritti potranno essere adattati ad altre tipologie di blockchain.

Gli steps che vengono eseguiti sulla rete sono sei e sono riportati di seguito [5]:

1. Le nuove transazioni vengono trasmesse a tutti i nodi
2. Ogni nodo raccoglie le nuove transazioni in un blocco
3. Ogni nodo lavora al fine di generare una complessa prova matematica (proof-of-work) per il proprio blocco
4. Nel momento in cui un nodo risolve il proof-of-work trasmette il blocco a tutti gli altri nodi
5. I nodi accettano il blocco solo se tutte le transazioni in esso sono validate
6. I nodi esprimono l'accettazione di un blocco lavorando alla creazione del blocco successivo.

Di seguito vengono illustrati i principali costituenti della struttura blockchain. La transazione, i cui componenti sono indicati in Figura 3, rappresenta l'elemento base. Ogni transazione, trasferimento di risorse tra i nodi di una rete, è caratterizzata dall'indirizzo pubblico del mittente e del ricevente, dalle caratteristiche della transazione e dalla firma crittografica.



**Figura 3:** componenti di una transazione [3]

Un insieme di transazioni formano invece un blocco, che possiede a sua volta, all'interno dell'intestazione (definita "header") l'hash identificativo, l'hash del blocco precedente e il timestamp. Infine, una catena di singoli blocchi costituisce l'intero sistema (Figura 4), come suggerisce il significato letterale della parola "blockchain".

## Blockchain

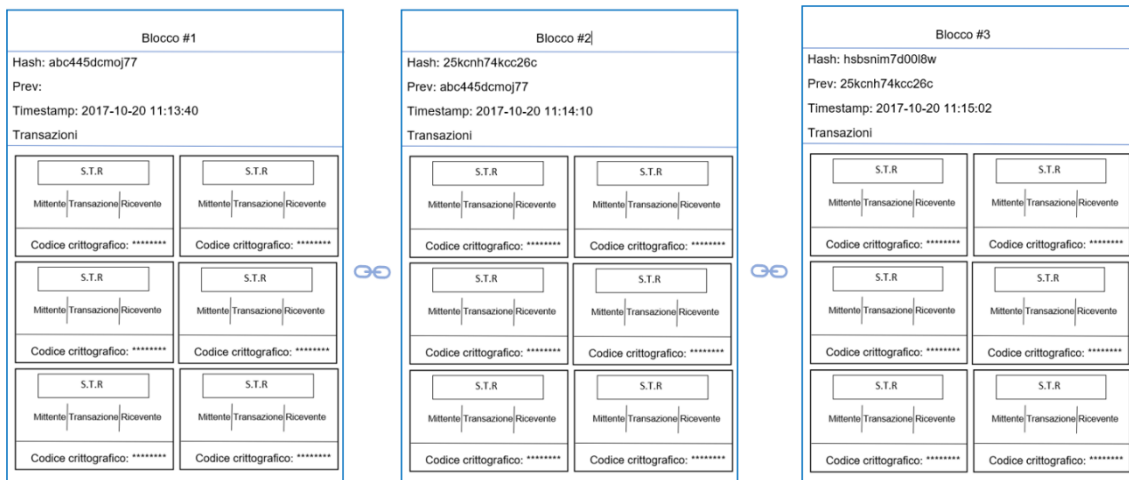


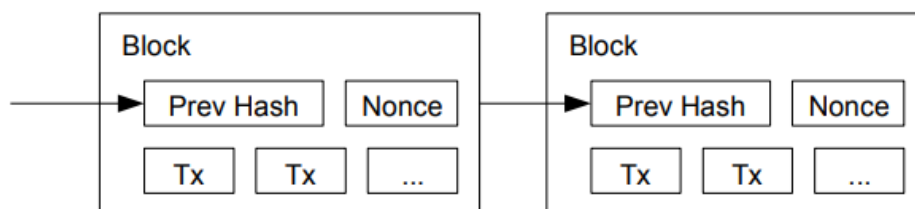
Figura 4: Blockchain vista come unione di più blocchi [3]

Lo stato condiviso dell'intera catena di blocchi e l'elenco di tutte le transazioni effettuate sono memorizzati all'interno di un registro, definito "ledger". Ecco perché spesso la blockchain viene anche chiamata "distributed ledger", ad indicare un database distribuito su ogni nodo della rete.

Infatti, la catena viene continuamente aggiornata e sincronizzata tra tutti i nodi della rete non appena questi si connettono. Questo è uno degli elementi fondanti della tecnologia blockchain e una delle motivazioni che rende tale sistema sicuro e immutabile.

Come spiegato precedentemente, prima della nascita della Blockchain non era possibile coordinare attività individuali su internet senza un ente che assicurasse la veridicità delle informazioni. Si tratta del così detto "problema dei generali bizantini", cioè del problema informatico su come raggiungere il consenso senza la presenza di un intermediario garante in situazioni in cui è possibile riscontrare errori o manomissione delle informazioni. La Blockchain risolve il problema con un approccio probabilistico. Le informazioni, cioè i dati che costituiscono le transazioni, viaggiano sulla rete e sono verificabili tramite dei problemi matematici che, per essere risolti, richiedono una potenza computazionale importante. In pratica, si immagini una transazione come l'invio nella rete di un problema matematico complesso inserito in un blocco; il primo nodo che riuscirà a risolvere l'enigma, convaliderà la transazione, aggiungendo il blocco alla Blockchain e aggiornando la catena su tutti gli altri nodi. I blocchi, essendo composti da svariate transazioni possiedono dei problemi crittografici molto complessi, la cui risoluzione avviene attraverso il sostenimento e il superamento della cosiddetta "*proof of work*" (prova di lavoro) di stampo probabilistico. Alcuni nodi, detti miners, metteranno

a disposizione del sistema la propria potenza computazionale per risolvere il gioco matematico-crittografico che terminerà con successo nel momento in cui verrà individuato un preciso numero, definito *Nonce* (Vedi figura 5 e 6). Tutti gli altri nodi invece, verificheranno che la soluzione trovata non corrisponda ad una transazione già eseguita.



**Figura 5:** Blocco che deve essere validato dal miner [4]

Nel momento in cui la prova matematica sarà risolta e il blocco sarà approvato, un timestamp verrà aggiunto al blocco, così da permettere la concatenazione in ordine cronologico dei diversi blocchi.



**Figura 6:** Il processo di mining

Dopo l’inserimento del nuovo blocco, esso non può essere cancellato o modificato e le transazioni presenti nello stesso sono verificabili da tutti i membri della rete.

A tal proposito è importante sottolineare un’altra caratteristica fondamentale della Blockchain che consiste proprio nell’immutabilità della registrazione: una volta che un blocco viene aggiunto alla catena, i dati ivi presenti diventano permanenti.

In realtà un modo per modificare un blocco validato ci sarebbe, se si riuscisse a mettere insieme il 50% più uno della potenza computazionale della rete. Per questo motivo, gli inventori del Bitcoin hanno predisposto una serie di incentivi per fare in modo che, per chi eventualmente fosse in possesso di tale potenza computazionale, sia più conveniente metterla a disposizione della rete per la soluzione degli enigmi crittografici, piuttosto che impiegarla per la manomissione di un blocco. Questo incentivo, finalizzato ad incoraggiare i nodi a comportarsi in modo onesto, si basa sulla teoria dei giochi, ed è rappresentato dal “premio”, sotto forma di bitcoin o commissioni, che la rete attribuisce a chi valida un blocco. Se un utente malintenzionato fosse in grado di mettere insieme una potenza computazionale pari al 50%+1 della potenza della rete, potrebbe scegliere

se usare tale potenza per commettere frodi, oppure utilizzarla per generare nuove monete. Con il sistema dei “premi” troverà più conveniente giocare secondo le regole, che permettono di guadagnare nuove monete, piuttosto che destabilizzare il sistema mettendo a rischio il proprio guadagno indebitamente acquisito.

Un altro aspetto da sottolineare è la correlazione tra la lunghezza della catena, ovvero il numero di blocchi aggiunti, e la sicurezza dei dati trasmessi. Infatti, ogni blocco possiede al suo interno l’hash del blocco precedente, creando così una connessione tra tutti i blocchi che costituiscono la rete. In tal modo, se un soggetto malintenzionato volesse modificare i dati presenti in un blocco, automaticamente verrebbe mutato anche il relativo hash, che è a sua volta memorizzato nel blocco successivo e che influenza gli hash di tutti i blocchi seguenti. In pratica, per alterare le informazioni presenti in un blocco occorrerebbe modificare anche tutti i blocchi successivi, con una potenza computazionale richiesta sempre maggiore.

Al fine di comprendere adesso in che modo verrebbe estromessa la figura dell’intermediario dalle transazioni, si prenda come esempio il sistema Bitcoin. L’algoritmo di validazione di un blocco infatti, nel momento in cui risolve la “proof-of-work” controlla che la chiave crittografica presente sulla transazione corrisponda alla chiave del soggetto che detiene la criptomoneta che si intende trasferire. Verifica quindi la reale disponibilità della “moneta” inviata, senza dover necessariamente ricorrere ad un *soggetto* che funga da garante di ciò.

### 2.1.3 La firma digitale e il sistema delle chiavi

Ogni transazione può essere iscritta in Blockchain solo dopo essere stata validata attraverso la firma digitale, mediante un algoritmo crittografico. Lo schema della firma digitale è costituito da tre componenti principali:

1. **L’algoritmo di generazione delle chiavi:** crea una chiave privata, utilizzata per firmare i “messaggi”, e una chiave pubblica, resa disponibile a tutti gli utenti del sistema per controllare l’attendibilità del messaggio.
2. **L’algoritmo che genera la firma digitale:** prende in input il messaggio (completo di tutti i dati) generato dall’utente e la sua chiave privata al fine di fornire come output i dati firmati digitalmente. Questi ultimi corrisponderanno ad una chiave pubblica (Vedi Figura 7).
3. **L’algoritmo di verifica della firma:** prende in input i dati firmati digitalmente (ottenuti dal meccanismo spiegato al punto 2) compresi di chiave pubblica dell’utente che ha effettuato la transazione e di firma digitale. Tale algoritmo calcolerà e confronterà i valori di due differenti hash, ovvero il primo ottenuto attraverso l’utilizzo della chiave

pubblica che decifrerà la firma e darà origine ad un hash, e il secondo ricavato attraverso il calcolo della funzione di hash dei dati ricevuti. L'algoritmo darà un risultato positivo (valore booleano "vero") se gli hash calcolati sono uguali, altrimenti segnalerà l'anomalia tramite il valore booleano "falso" (Vedi Figura 7).

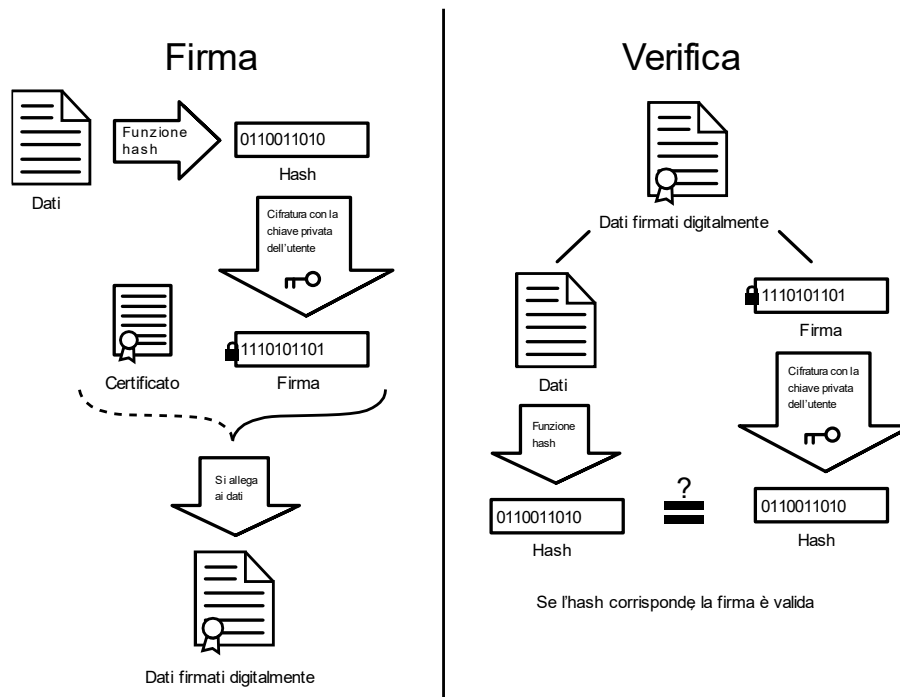


Figura 7: Generazione e verifica della firma [6]

In generale, l'algoritmo di firma presente sulla blockchain risulta essere ben definito e sicuro, e gode di due proprietà essenziali [7]:

1. La verificabilità delle firme
2. Le firme non sono falsificabili

### 2.1.4 Algoritmi e protocolli di consenso

Il protocollo di consenso è il processo nel quale la maggioranza dei nodi di una rete Blockchain, o in alcuni casi tutti, concordano sullo stato di un registro, garantendo l'integrità e l'affidabilità delle operazioni eseguite. In sostanza, il protocollo contiene un insieme di regole e procedure da eseguire per ottenere il consenso e permettere così la prosecuzione dell'operabilità della piattaforma. Esistono vari e differenti algoritmi di consenso, tra i quali i più importanti risultano essere:

- Proof-of-Work (PoW)
- Proof-of-Stake (PoS)
- Delegated-Proof-of-Stake (DPoS)

L'algoritmo **PoW** non è altro che una complessa prova matematica che i miners devono risolvere per poter verificare le transazioni sulla rete e ricevere un compenso in forma di criptomoneta. Tale procedura è chiamata mining e richiede un elevato costo. L'algoritmo PoW è sì di difficile esecuzione, ma, una volta completato, è molto semplice per gli altri nodi da verificare, rendendo così superflua la presenza di un qualsiasi middleman.

Differente è l'idea alla base dell'algoritmo **PoS**. Infatti, nel PoW, la probabilità di minare un blocco, e conseguentemente essere remunerato dal sistema, è strettamente legata alla potenza di calcolo posseduta dai "miners", invece nel PoS, la probabilità di creare un blocco è proporzionale alla ricchezza (wealth o stake) dell'utente nel sistema. La PoS infatti non prevede miners, bensì validatori che non hanno il compito di risolvere il complesso problema matematico del proof-of-work, ma dovranno congelare nel sistema una quantità di moneta (*Stake*) da loro decisa, per poter esser poi selezionati per validare il blocco. Maggiore stake bloccheranno, maggiore sarà la probabilità di diventare il nodo validatore ed essere ricompensati del lavoro svolto sulla base delle commissioni sulle transazioni.

L'algoritmo **D-PoS** invece, è fondato su un meccanismo di voto in cui i vari nodi esternalizzano il proprio lavoro a terze parti, denominate "nodi delegati", alle quali è affidato il controllo della rete. Ogni delegato è votato in base ad un sistema di votazioni pesate, ciò vuol dire che il peso dei voti dei singoli nodi è direttamente proporzionale alla ricchezza che ogni nodo possiede nel sistema.

Generalmente, le ricompense ottenute da un delegato in seguito alla convalida di nuovi blocchi sono divise proporzionalmente tra gli elettori che hanno votato a suo favore [8]. Per concludere, si può affermare che, tale algoritmo, genera un sistema di voto basato sulla reputazione dei delegati, infatti, se uno di essi dovesse intraprendere delle azioni in cattiva fede, verrebbe rapidamente espulso e sostituito da un nuovo delegato, eletto col sistema di voto descritto in precedenza.

La Tabella 1 riporta le caratteristiche, messe a confronto, dei principali algoritmi di consenso presenti sulla Blockchain.

Consenso	Efficienza	Scalabilità	Applicazioni	Tipo di Blockchain
PoW	Bassa	Povera	Bitcoin	Pubblica
PoS	Più alta	Buona	PPcoin	Pubblica
DPoS	Alta	Buona	EOS	Pubblica

Tabella 1: Algoritmi di consenso a confronto [7]

## 2.2 Tipologie di Blockchain: pubblica, privata e consortium

Un altro aspetto importante riguarda l'individuazione del modello Blockchain da adottare per il caso d'uso oggetto di studio. A tal proposito, è necessario aver presente le due conformazioni di blockchain, ovvero "*permissionless*" e "*permissioned*", che generano tre differenti tipologie di blockchain: pubblica, privata o consortium.

La soluzione proposta nei paragrafi precedenti si è strutturata intorno alle caratteristiche di una blockchain pubblica, alla luce del fatto che rimane la più rappresentativa e fedele al pensiero dell'ideatore. Per Nakamoto infatti, la tecnologia del database distribuito dovrebbe avere una natura decentralizzata e "permissionless". Alcuni studiosi (più tradizionalisti) ritengono che le soluzioni private manchino e siano essenziali in molti contesti, ragion per cui, negli ultimi tempi svariate versioni private di blockchain hanno iniziato a diffondersi, soprattutto a livello aziendale e governativo. Questi modelli consentono di dar vita a sistemi chiusi, in cui i permessi di accesso sono controllati da un'autorità centrale. A seguire, sono descritte le tre diverse tipologie di blockchain.

### 2.2.1 Blockchain Privata

Una blockchain privata ("*permissioned*") [8] pone dei vincoli alla partecipazione alla rete. Infatti, non tutti i nodi possono partecipare all'algoritmo di consenso, ma solo coloro che possiedono l'autorizzazione specifica. In generale, ogni nodo, in base alla tipologia di autorizzazione che possiede gode dei seguenti permessi:

- \* lettura dei blocchi;
- \* scrittura dei blocchi;
- \* esecuzione di codice (se prevista dalla blockchain);
- \* verifica dei nodi.

La blockchain privata è una blockchain in cui le autorizzazioni di scrittura e di modifica dei singoli blocchi sono completamente centralizzate. Per quanto riguarda invece le autorizzazioni in lettura, queste possono essere pubbliche o limitate ad un numero finito di utenti. La blockchain privata non è un sistema decentralizzato, poiché esiste una chiara gerarchia sul controllo. Può essere definita come un sistema centralizzato tradizionale e distribuito, poiché molti nodi mantengono ancora una copia della catena, con l'aggiunta di un grado di verificabilità crittografica. È chiaro quindi che, la blockchain privata è sicuramente più adatta ai modelli di business tradizionali in cui i dati e la privacy rimangono sotto il controllo aziendale.

### **2.2.2 Blockchain Pubblica**

La blockchain pubblica (“permissionless”) [8] è una rete in cui tutti i nodi possono partecipare al processo di verifica dei blocchi e possiedono tutte le proprietà elencate nel paragrafo precedente (Blockchain Privata).

In essa chiunque può verificare, visionare ed emettere nuove transazioni, svolgere l'attività di mining e prender parte al meccanismo di consenso.

Come già definito in precedenza, la blockchain pubblica è un sistema in grado di offrire un elevato livello di sicurezza, grazie alla presenza di un complesso meccanismo di consenso che spinge i miners ad investire pesantemente in potenza computazionale. Per le caratteristiche appena descritte, questa tipologia di blockchain viene definita come “completamente decentralizzata”.

### **2.2.3 Blockchain Consortium**

Le consortium blockchain [8] vengono concepite come un ibrido tra la tipologia pubblica e quella privata, poiché si basano su un meccanismo di consenso che è gestito da un insieme di nodi preselezionati. Per tale ragione è definita “parzialmente decentralizzata”.

In un sistema aperto chiunque può convalidare i blocchi, mentre, in uno chiuso, tale azione è concessa solamente ai nodi nominati da un'autorità centrale. In questo modello invece, sono presenti diversi nodi che godono del medesimo potere e che fungono da validatori.

Le autorizzazioni in lettura possono essere sia pubbliche, ovvero rilasciate a tutti i nodi della rete, sia private, vale a dire concesse ad alcuni partecipanti autorizzati, e sia rilasciate solamente ai singoli validatori. Questa tipologia contiene al suo interno



caratteristiche tipiche delle soluzioni pubbliche, ad esempio un minor controllo sulle attività del network, e caratteristiche intrinseche delle blockchain private, come la necessità di maggiore affidabilità sulle operazioni del sistema. Questa tipologia di blockchain potrebbe essere utile in tutti quei contesti in cui più figure o enti collaborino per un medesimo scopo e necessitano perciò di un luogo comune in cui operare e trasmettere transazioni ed informazioni.

## 2.2.4 Le tre tipologie di Blockchain a confronto

Sulla base delle considerazioni fatte, è dunque possibile riepilogare le principali caratteristiche delle blockchain pubbliche e private (Tabella 2) al fine di poter comprendere in modo chiaro le implicazioni che derivano dall'implementazione dell'una piuttosto che dell'altra soluzione, con i rispettivi vantaggi e svantaggi legati ad esse. [9]

- La Blockchain pubblica fornisce protezione sui dati attraverso un meccanismo che rende molto difficile, se non impossibile, effettuare modifiche sul “database”, mirando così a tutelare gli utenti da potenziali attacchi esterni. È un sistema aperto ed immutabile e chiunque può, in qualsiasi momento, divenire un nodo della rete, effettuare transazioni, verificare e visionare i contenuti del “database”, mantenendo sempre l’anonimato. Di contro però, vi è la lentezza nella validazione dei blocchi e il dispendioso spreco energetico derivato dal lavoro compiuto dai miner.
- La Blockchain privata invece, sia che si tratti di un consorzio che di una blockchain completamente “permissioned”, dà la possibilità di modificare le regole della piattaforma e ripristinare le transazioni. Il meccanismo di consenso si struttura intorno ad un numero di nodi limitato e noto a priori (godono di fiducia pregressa), e ciò rende più veloce il processo. Inoltre, i vantaggi derivanti dall’uso di tale tecnologia sono: i) la correzione di errori in tempi brevi, grazie ad interventi manuali, e ii) una maggiore privacy sui dati, possibile grazie alle limitazioni dei permessi in lettura.

	<b>Pubblica</b>	<b>Privata</b>	<b>Consortium</b>
<b>Determinazione del consenso</b>	Tutti i nodi partecipanti alla rete	Un insieme selezionato di nodi	Un insieme selezionato di nodi
<b>Chi può leggere</b>	Tutti	Solo gli utenti invitati	Dipende
<b>Chi può scrivere</b>	Tutti	I partecipanti approvati	I partecipanti approvati

<b>Proprietà</b>	Di nessuno	Di un'entità singola	Molteplici entità
<b>Conoscenza dei partecipanti</b>	No	Si	Si
<b>Velocità di transazione</b>	Lenta	Veloce	Veloce
<b>Immutabilità</b>	Elevate	Non chiara	Non chiara
<b>Centralizzazione</b>	No	Si	Parziale

**Tabella 2:** Confronto tra le diverse tipologie di Blockchain [8]

Alla luce delle differenze che sussistono tra i tre modelli è importante sottolineare che, ai fini dell'implementazione, le opportunità che tali sistemi possano offrire variano da settore a settore. In alcuni casi la blockchain pubblica può risultare la scelta ottimale, in altri invece, in cui è richiesto un maggiore controllo sul sistema, la soluzione più adeguata risulta essere una blockchain privata o consortium.

## 2.3 Smart Contract

### 2.3.1 Cenni Storici

Nick Szabo, un autore di riferimento nel panorama dei dati crittografati degli anni Novanta, pubblicò nel 1997 due documenti "Formalizing and Securing Relationships on Public Networks" [10] e "The Idea of Smart Contracts" [11], nei quali introdusse il concetto astratto di "contratti intelligenti". Nel 1998 invece, pubblicò un documento, "Secure Property Titles with Owner Authority" [12], in cui formalizzò i concetti delineati nei precedenti lavori. Il suo modello predispone un sistema di circolazione di diritti tramite contratti digitali e computabili, in cui la prestazione viene eseguita automaticamente in base ad un contratto scritto in codice matematico, basato su algoritmi, in grado di eseguirsi automaticamente al realizzarsi di alcune condizioni, senza l'intervento umano. Il tutto è paragonato a un distributore automatico, programmato per erogare un servizio sulla base di alcune condizioni presenti all'interno di un codice di programmazione. A seguire [13], grazie all'invenzione della Blockchain, nel 2014, il ventenne Vitalik Buterin delineò le caratteristiche di Ethereum, ovvero quella che è poi diventata la piattaforma di

riferimento per lo sviluppo e l'esecuzione degli smart contracts. Lo scopo di questa piattaforma è quello di fornire una Blockchain con un linguaggio di programmazione incorporato, utilizzato per costruire delle regole contrattuali che si auto eseguano in accordo con alcune regole preimpostate.

### **2.3.2 Che cosa è uno Smart Contract**

Quando si parla di contratti intelligenti si può far riferimento al contratto convenzionale che, dal punto di vista della forma e dei contenuti è pressoché simile.

Lo smart contract è la trascrizione di un contratto convenzionale in codice digitale, che rileva il verificarsi delle condizioni contrattuali e, in tal caso, esegue, sempre digitalmente, le azioni che il contratto prevede.

L'esecuzione della prestazione di uno Smart Contract non dipende dalla volontà delle parti. Infatti, nel momento in cui si verifica una condizione, automaticamente verrà eseguita l'azione corrispondente, senza l'intervento umano. Pertanto, qualora un soggetto non volesse adempiere ad una determinata prestazione, non avrebbe la possibilità di sottrarsi alla sua esecuzione, poiché il sistema non dipende né dalla volontà umana né da quella del legislatore, ma dal codice trascritto.

Lo scopo di questo sistema è quello di evitare gli inadempimenti delle parti e quindi di limitare le forme di litigiosità dovute a verificarsi di questo problema. In sostanza, l'obiettivo è quello di eliminare il contenzioso relativo all'attuazione del contratto ed eliminare la necessità di assistenza legale nella fase di esecuzione di un accordo.

## **2.4 I Token**

All'interno della tecnologia blockchain. il token è visto come un "gettone" virtuale che rappresenta un'unità di un bene digitale. In genere, essi vengono emessi e venduti al pubblico attraverso una ICO (Initial Coin Offering). L'ICO, è una forma di finanziamento utilizzata da startup o da soggetti che intendono realizzare un progetto su base blockchain. Tale finanziamento ha durata limitata ed emette token digitali in cambio di moneta tradizionale.

Il token è un oggetto che acquista un certo valore solo all'interno di un determinato contesto, al di fuori di esso invece, non ha alcuna rilevanza. Per rendere meglio l'idea, si è soliti in genere paragonare i token alle fiches utilizzate nel gioco d'azzardo. Esse infatti, acquistano il valore che un soggetto decide di attribuirgli e rappresentano qualcosa solamente all'interno delle mura del casinò. I token pertanto, hanno il valore

che il proprio creatore decide di attribuirgli, limitatamente al contesto per il quale vengono create.

### 2.4.1 Le tipologie di Token

L'acquisto del token conferisce una royalties su un determinato progetto oppure semplicemente la possibilità di usufruire di un servizio erogato dall'emittente. È necessario suddividere i token emessi da una ICO in tre gruppi (Figura 8):

1) **Token di pagamento**: sono delle valute vere e proprie e costituiscono uno strumento di pagamento per l'acquisto di beni e servizi. Possono essere scambiati con altre tipologie di token o con moneta tradizionale. In questa categoria rientrano ad esempio i bitcoin e gli ether.

2) **Token di investimento**: questa tipologia di token attribuisce ai possessori una serie di diritti, come ad esempio il diritto di voto o il diritto di ricevere dei pagamenti. Chi investe in questa tipologia di token effettua un investimento in quanto crede che il valore del token acquistato possa aumentare nel tempo.

3) **Token di utilizzo**: è una tipologia di token che permette di utilizzare beni e servizi erogati dal soggetto emittente. Il token in questo caso è una chiave di accesso per far parte della rete e per usufruire degli strumenti e dei servizi che essa offre.

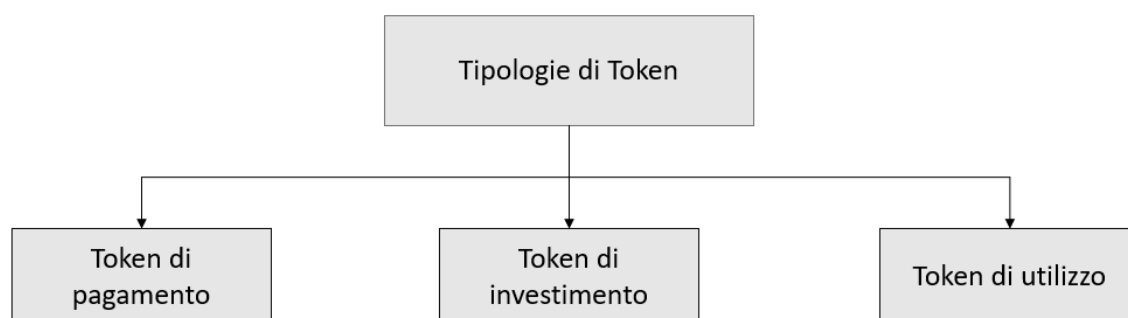


Figura 8: Tipologie di Token [20]

## 3. E-Commerce B2B: I Trend del Settore, Analisi e Prospettive

### 3.1 Il “Digital Handshake”

L'ordinamento italiano (articolo 1325 del codice civile) stabilisce che l'accordo è il primo requisito essenziale alla base di un contratto, seguito dalla causa, dall'oggetto e dalla forma. Giornalmente tutti noi stipuliamo contratti, basti pensare che le più banali attività quali la semplice azione di comprare il giornale, il biglietto del cinema, fare la spesa al supermercato o acquistare un caffè al distributore costituiscono la stipula di un contratto vero e proprio. Le due parti, possono concretizzare il proprio “agreement” espressamente oppure tacitamente. Nel primo caso, una dichiarazione in forma scritta funge da tutela per le due parti dell'accordo. In essa si riportano in maniera dettagliata i vincoli che i soggetti si impegnano a rispettare, andando a generare l'oggetto del contratto stesso. Contrariamente avviene invece nel caso dell'accordo tacito, in cui le due parti manifestano l'intenzione di stipulare un contratto tramite fatti concreti, come ad esempio un accordo verbale o una stretta di mano fisica.

La stretta di mano non è un prodotto della società moderna, e nemmeno esclusivo del mondo occidentale. La sua origine risale a più di 5000 anni fa ed è confermata da geroglifici egizi che raffigurano uomini e dei nell'atto di stringersi la mano in segno di accordo. Quindi la stretta di mano nella storia ha sempre rappresentato un segno di fiducia e di intesa, oltre che un segnale di saluto. Oggi, il valore della stretta di mano è analogo a quello della firma su un contratto scritto, purché le parti siano realmente intenzionate a portare avanti gli accordi presi. Questa tipologia di patto possiede un limite legato all'interpretazione soggettiva, ovvero all'importanza che i soggetti attribuiscono alla stretta di mano stessa. Infatti, il problema principale dei contratti orali è che essi risultano difficili da provare in caso di contestazioni sulla loro esistenza o su un dettaglio come il prezzo, qualità della prestazione etc. Così, a causa dell'assenza di un documento scritto nel quale sono riportate le specifiche dell'accordo preso, c'è la necessità di far ricorso a testimoni, tra l'altro molto difficili da rintracciare, per fornire le prove di quanto affermato. Quindi, sebbene una stretta di mano abbia valore simbolico, purtroppo significa poco dal punto di vista legale e, per tale motivo è vivamente sconsigliato concludere accordi facendo ricorso a tale gesto. I contratti scritti sono senza dubbio da privilegiare perché forniscono alle parti certezza sui termini esatti dell'accordo e aiutano ad impedire ai malintenzionati di affermare successivamente la diversità delle condizioni stabilite. [15]

Tuttavia, la complessità delle società ed economie odierne necessita di contratti sempre più difficili da gestire a causa dell'elevato numero di dettagli da includere. La redazione

di un contratto implica inoltre ingenti costi in termini di tempo e denaro, sia per le aziende che per i privati. Oneroso è infatti il prezzo per la scrittura, interpretazione e per l'applicazione di questi documenti che richiedono anche la presenza di un intermediario. Ecco che, per risolvere questi problemi, si sta sempre più puntando sulla così detta "Digital Handshake", ovvero sulla stretta di mano non più fisica, ma digitale che, affinché abbia esito positivo, richiede la presenza di tre fattori:

- Autenticità: deve essere genuina e rispettare il vero
- Omnicanalità: i clienti sono seguiti su più canali di comunicazione, così da trasmettere loro una maggiore fiducia
- Calore: il modo in cui avviene l'interazione è di fondamentale importanza; il tono, le immagini e l'intimità con cui si genera la contrattazione fanno parte della strategia alla base delle relazioni sui canali digitali.

La tecnologia blockchain è il campo perfetto su cui attuare la stretta di mano digitale. Infatti, grazie alle caratteristiche possedute da tale piattaforma, si trasmette all'utente un senso di trasparenza e sicurezza. Il sistema decentralizzato alla base della blockchain, assicura infatti uguale trattamento a tutti gli utenti, accesso protetto, garanzia contro la falsificazione dei dati e assenza di problemi tecnici. Per tale ragione, Distributed Ledger Technology (DLT) è il candidato ideale per diventare l'infrastruttura alla base del Digital Handshake e dei contratti digitali.

## 3.2 La Reputazione Digitale

A partire dagli anni '50 si iniziò a dare importanza a tutti i concetti legati all'immagine aziendale. Da quel momento in poi, molti studiosi hanno ripetutamente affermato che la reputazione, ovvero l'idea generata dai giudizi del pubblico riguardo qualità e affidabilità, per un'azienda riveste un forte ruolo strategico, in quanto è in grado di influenzare i risultati economici e di mercato dell'impresa stessa. Godere di una buona reputazione implica, generalmente, qualità nei prodotti e nei servizi offerti e ciò è un forte segnale per il mercato e per la concorrenza. Infatti, vi è una relazione fra reputazione e possibilità di attrarre migliori risorse, migliori investitori e una maggiore flessibilità nella formazione dei prezzi.

L'avvento e la diffusione di Internet hanno trasformato le modalità di svolgimento degli "affari", attribuendo una maggiore rilevanza alla reputazione del venditore. La compravendita infatti avviene su piattaforme digitali, su prodotti o servizi di cui il consumatore non possiede un'informazione completa e tra soggetti sconosciuti, che

difficilmente hanno avuto rapporti in precedenza. Sembra evidente, che tutto ciò richieda un certo livello di fiducia nel venditore, infatti tale mancanza riduce o elimina del tutto la propensione all'acquisto.

La diffusione di internet è stata accompagnata da un pubblico rinnovato, tecnologico e consapevole di non rivestire più il ruolo di consumatori passivi, ma di essere utenti che producono contenuti, come recensioni e giudizi, fondamentali per le dinamiche del mercato online.

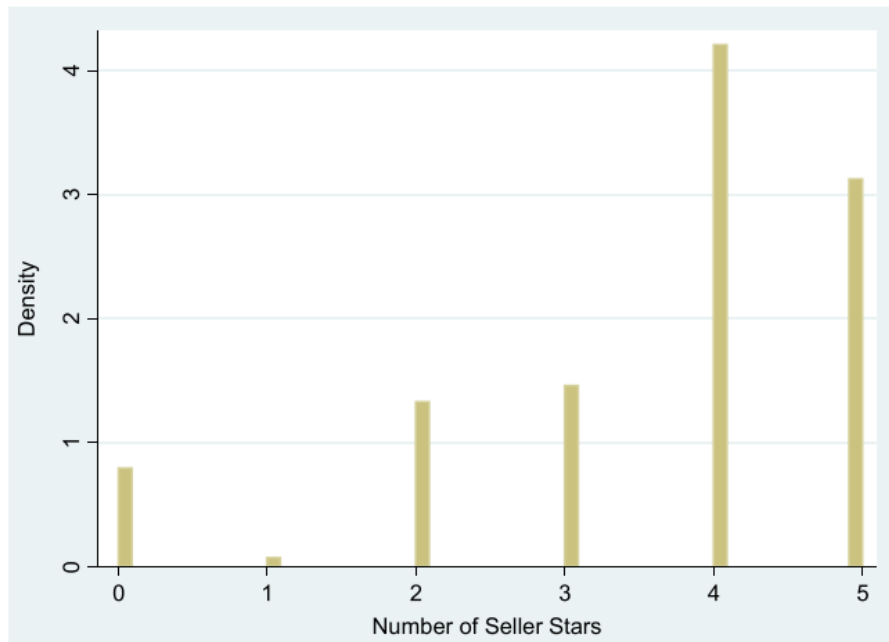
Sul mondo digitale si è creata una sorta di comunità in cui i soggetti, per svariati motivi, entrano in contatto, si scambiano informazioni, riportano fatti ed esperienze personali, tramite delle conversazioni "many to many". In questo modo si generano immagini sui vari prodotti e servizi offerti sul web. [16]

Uno studio, effettuato da Steve Thompson e Michelle Haynes [17], è stato incentrato sul valore della reputazione dei rivenditori al dettaglio che commerciano tramite un sito di confronto dei prezzi. Tale piattaforma affrontava il problema della qualità, incoraggiando il feedback generato dagli utenti in merito alle prestazioni del venditore, che aggregate, formavano una valutazione, da uno a cinque, sul venditore stesso.

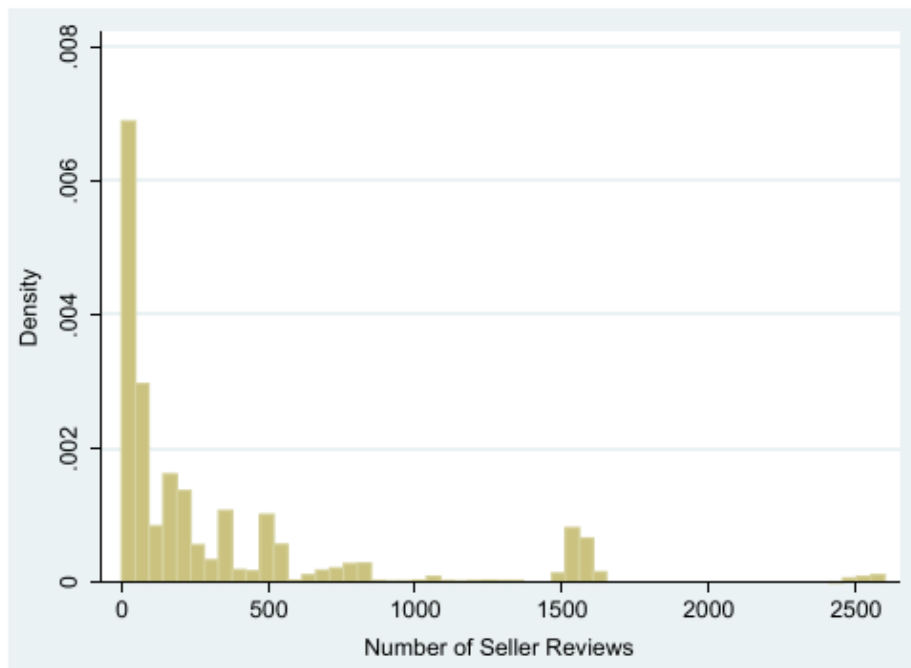
La Figura 9, tratta dallo studio [17] sopra citato, raffigura l'istogramma della distribuzione per stelle. Nonostante la valutazione da uno a cinque abbia una facile interpretazione sulla misura della qualità del servizio fornito dal venditore, essa possiede una pecca legata alla valutazione con zero stelle, che potrebbe indicare sia una mancanza di dati sufficienti per formare una valutazione, che un nuovo iscritto al sistema. Pertanto, la relazione tra 0 e 1-5 stelle crea una certa ambiguità e, in effetti, potrebbe addirittura essere preferibile possedere un punteggio con zero stelle piuttosto che una stella, indicatore chiaro di una scarsa prestazione.

Il grafico presente nella Figura 10 riporta invece pochi venditori con un elevato numero di recensioni [17]. La presenza di molte valutazioni, siano esse positive o negative, permette al cliente di costruire un'immagine dettagliata del venditore. Contrariamente, la scarsa presenza di valutazioni non dà la possibilità di avere una reale percezione in merito al comportamento della controparte.

Da questo studio è emerso anche che i venditori che hanno una scarsa reputazione usano strategie che si basano su prezzi bassi, paragonabili alle tattiche "mordi e fuggi", tattiche in cui un soggetto entra sul mercato per un tempo limitato con prezzi vantaggiosi, al fine di ottenere un imminente guadagno e poi uscire, ed optare per il successivo offuscamento e cambio nome. Invece, i venditori che hanno ricevuto molte stelle, a dimostrazione dell'elevata fiducia, tendono a mantenere i prezzi più alti, forti del fatto che saranno scelti grazie all'affidabilità trasmessa dalla valutazione visibile sul sistema.



**Figura 9:** Densità per numero di stelle



**Figura 10:** Numero di valutazioni per venditore

Lo studio condotto da Hema Yoganarasimhan [18] ha analizzato i siti web che danno ai lavoratori autonomi la possibilità di entrare in contatto con gli acquirenti di servizi erogabili elettronicamente, al fine della comprensione dei mercati online per liberi professionisti. Questo studio mostra anche le distribuzioni dei prezzi relativamente a tre tipi di venditori (Figura 11): (a) venditori che non hanno valutazioni precedenti, (b) venditori che hanno una media reputazione e (c) venditori che hanno un'ottima reputazione. In media, i venditori senza valutazioni precedenti hanno prezzi più bassi,



mentre quelli con reputazione media hanno prezzi leggermente più alti e quelli con la reputazione più alta hanno i prezzi migliori.

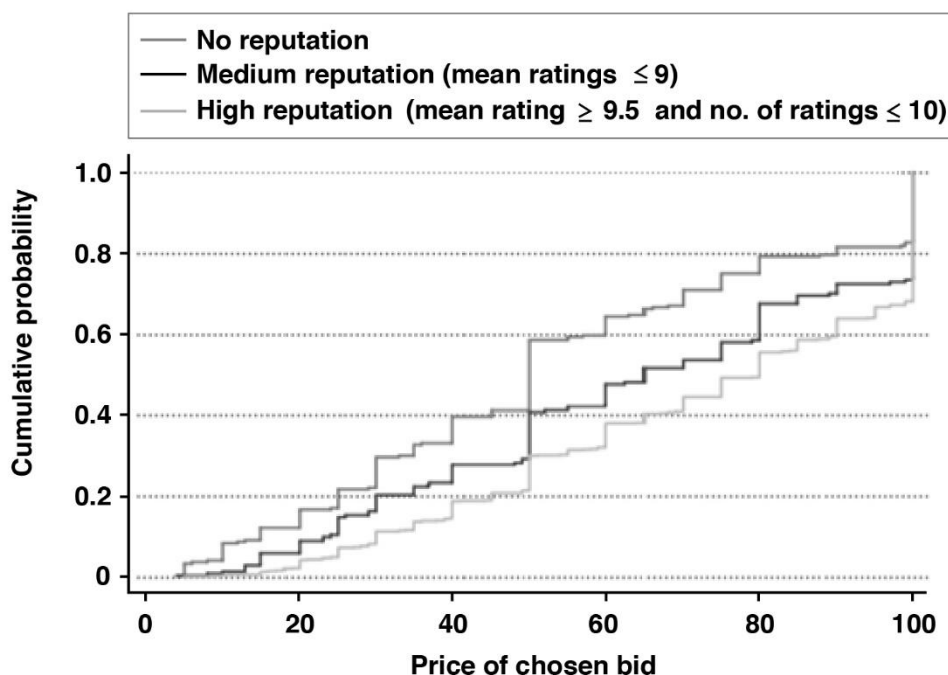


Figura 11: Prezzi delle offerte in base alla reputazione [37]

### 3.3 Mercato Business-to Business (B2B)

B2B, acronimo di “business-to-business”, viene utilizzato per descrivere il mercato che regola le transazioni commerciali ed ogni genere di rapporto lungo la catena, ovvero tra un’azienda e i suoi fornitori o tra un’azienda e un’altra. Con tale espressione si indicano quindi tutte le imprese, organizzazioni o amministrazioni che acquistano prodotti o servizi per la loro attività.

In generale, si ha a che fare con un settore che coinvolge un minor numero di potenziali clienti rispetto al mercato B2C (Business-to-Consumer).

#### 3.3.1 I Prodotti

Siamo in presenza di un mercato complesso in cui lo scambio di beni e servizi tra i soggetti può avvenire in differenti modi: dalla singola vendita alla fornitura periodica, fino

alla partnership tra aziende. I beni che generalmente vengono scambiati in questo mercato sono [19]:

1. Beni input: ogni materia prima o semilavorato che andrà a generare un prodotto finito da mettere sul mercato.
2. Beni strumentali: tutti i prodotti a supporto dei processi produttivi e logistici, come ad esempio impianti o macchinari
3. Beni facilitanti: servizi forniti ad un soggetto per il supporto di processi organizzativi o per il mantenimento e la riparazione di beni. Tra questi possono rientrare anche i servizi di consulenza e di formazione del personale.

### **3.3.2 Le Relazioni**

Aspetto caratterizzante del settore B2B è la collaborazione tipicamente continuativa fra i diversi stadi della filiera. È ammissibile la singola compravendita tra azienda e azienda, ma nella maggior parte dei casi è presente un rapporto periodico tra le due parti. Questo segmento di mercato è caratterizzato da un rapporto di continuità tra gli attori e, poiché generalmente gli accordi possono protrarsi nel tempo, per mesi o anni, anche le trattative il cui oggetto sono i dettagli dell'accordo possono a loro volta durare un lungo periodo di tempo e coinvolgere diversi attori.

### **3.3.3 Le Transazioni**

In genere, le transazioni che si concretizzano sul mercato B2B hanno un valore economico più rilevante e una durata maggiore, rispetto al settore B2C. Per tale motivo si è soliti procedere con la stipula di un contratto in cui le due controparti si impegnano a rispettare le condizioni stabilite.

Gli stakeholders business-to-business cercano in genere maggiori informazioni relative al prodotto o servizio che intendono acquistare rispetto ai clienti B2C. Questo atteggiamento deriva dal fatto che l'investimento in genere è ingente, e dal fatto che la qualità del bene o servizio acquistato influisce sulla bontà di ciò che si andrà ad offrire all'utente finale.

### 3.3.4 Differenze tra B2B e B2C

Le principali differenze tra il mercato business-to-business e quello business-to-consumer sono schematizzate in Tabella 3 [20].

Il mercato business-to-business, come detto in precedenza riguarda le società che vendono i propri prodotti o servizi esclusivamente ad altre imprese (ad esempio, un produttore tratta con un grossista o un grossista con un rivenditore) e non ai consumatori; mentre il mercato business-to-consumer riguarda le aziende che vendono i propri beni o servizi ai consumatori finali. La dimensione del mercato business-to-consumer interessa una vasta area nazionale e internazionale, invece, la dimensione del mercato business-to-business potrebbe essere circoscritta al numero di produttori specializzati nell'area geografica di riferimento. Inoltre, le vendite nel settore B2B hanno valori di ordini più elevati, cicli più lunghi e sono spesso più complesse di quelle del mercato B2C. In quest'ultimo il pagamento avviene all'atto dell'acquisto tramite contanti o carta, e la domanda che si vuole soddisfare ha alla base il desiderio immediato del consumatore. Invece il mercato B2B è caratterizzato da tempi più lunghi e da più complessi metodi di pagamento, con una domanda dettata dalla necessità e dalla ratio delle imprese di portare avanti i propri business.

	<b>Business-to-Consumer</b>	<b>Business-to-Business</b>
<b>Obiettivo</b>	Utente finale	Impresa
<b>Dimensioni del mercato</b>	Grande	Piu piccolo
<b>Volumi di vendita</b>	Basso	Alto
<b>Processo decisionale</b>	Individuale	Su commissione
<b>Rischio</b>	Basso	Alto
<b>Processo di acquisto</b>	Breve	Più lungo
<b>Pagamento</b>	Spesso istantaneo	Il pagamento immediato potrebbe non essere richiesto
<b>Transazione</b>	Carta o contanti	Richiede sistemi più complessi
<b>Decisione del consumatore</b>	Emotivo	Razionale
<b>Domanda</b>	Basata sul desiderio	Basata sul bisogno
<b>Utilizzo dei mass media</b>	Essenziale	Evitabile

**Tabella 3:** Differenze tra B2B E B2C [23]

### 3.3.5 TOE Framework

L'acronimo TOE sta per tecnologia-organizzazione-ambiente e rappresenta un framework che ha l'obiettivo di descrivere in che modo un'azienda adotta e implementa le innovazioni tecnologiche e come tale scelta sia influenzata da tre tipologie di contesto: tecnologico, organizzativo e ambientale. Il contesto tecnologico comprende diverse tecnologie, interne ed esterne rilevanti per l'azienda. Il contesto organizzativo raggruppa invece tutte le risorse e le caratteristiche dell'impresa, ad esempio la dimensione, il grado di centralizzazione, formalizzazione etc. Infine, il contesto ambientale include le caratteristiche del settore, ovvero dimensione, concorrenza, aspetto normativo etc.

In Figura 12 è rappresentato lo schema generale del framework TOE, nel quale si evidenzia come l'adozione del B2B EC (Electronic Commerce) da parte di un'azienda sia influenzata da fattori ambientali, tecnologici e organizzativi

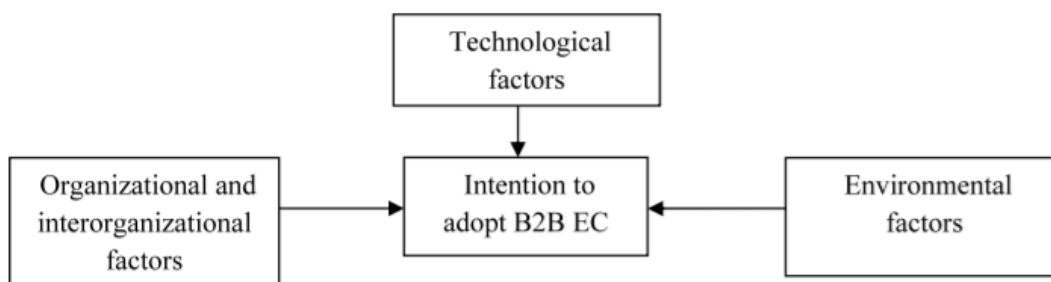


Figura 12: Categorie generali del quadro TOE [21]

Il contesto tecnologico è influenzato dai seguenti fattori [21]:

- **Costo:** Internet offre l'opportunità a chiunque di avere accesso ai mercati globali, così da poter effettuare un'analisi in termini di prezzo e ridurre i costi di inventario, approvvigionamento e coordinamento. Pertanto, l'utilizzo del digitale, in particolare l'adozione del B2B EC, potrebbe portare alle aziende un notevole vantaggio economico in termini di riduzione dei costi, incoraggiandole così a seguire la strada del digitale.
- **Complessità:** Il livello di utilizzo di una nuova tecnologia dipende anche dal grado in cui un'innovazione è percepita come semplice o difficile da comprendere e utilizzare. B2B EC viene vista come un'innovazione complessa, poiché richiede l'adeguamento tecnologico, amministrativo e organizzativo delle aziende lungo la catena di fornitura.

- *Affidabilità della rete:* La capacità dell'azienda di trasferire con successo applicazioni aziendali critiche da e verso i suoi partner nella catena di fornitura.
- *Sicurezza dei dati:* Si riferisce ai problemi di sicurezza associati alle transazioni su internet.
- *Scalabilità:* Riguarda le economie di scala fornite da Internet. Le aziende possono creare dei nuovi mercati ed espandere la portata dei propri prodotti grazie all'adozione del B2B EC.

Il contesto organizzativo risulta invece condizionato dai seguenti fattori [21]:

- *Il supporto dei manager:* L'atteggiamento positivo dei managers nei confronti del cambiamento è un requisito fondamentale per il primo passo verso l'innovazione. In tal modo si crea un ambiente organizzativo ricettivo alla modernizzazione.
- *Fiducia:* Il fattore critico nelle transazioni tra clienti e fornitori è proprio la fiducia. Nel mercato B2B la cooperazione duratura e l'investimento di elevate quantità di denaro per l'acquisto di beni o servizi sono alla base di questo settore. A tal proposito quindi, è di fondamentale importanza per i partner sviluppare un livello di fiducia reciproca.
- *Dimensione dell'azienda:* Spesso le grandi aziende possiedono alcuni vantaggi che consentono loro di utilizzare maggiormente il B2B EC rispetto alle PMI (piccole medie imprese). Tra questi, ad esempio, il maggior numero di risorse, le economie di scala, maggiore potere sui partner e quant'altro.
- *Tipologia di impresa:* Si è riscontrato come il tipo di azienda influisca sul grado di utilizzo del B2B EC, infatti le aziende manifatturiere hanno livelli di utilizzo di EC superiori a quelle non manifatturiere.
- *Livello di Management:* Alcuni studi hanno dimostrato che la presenza di forti risorse umane, e la presenza di managers con elevate capacità nell'e-business aumentano la probabilità di facilitare l'implementazione del B2B EC.

Per quanto riguarda invece il contesto ambientale si evidenzia come le aziende possano subire le pressioni dell'ambiente in cui operano, ovvero da parte dei clienti, dei fornitori o anche dei concorrenti. Inoltre, il paese di origine è un altro fattore chiave, in quanto le leggi e le misure che regolano gli stati non sono sempre uguali.

La Figura 13 mostra in maniera dettagliata i fattori che rientrano in ciascuna categoria del TOE Framework.

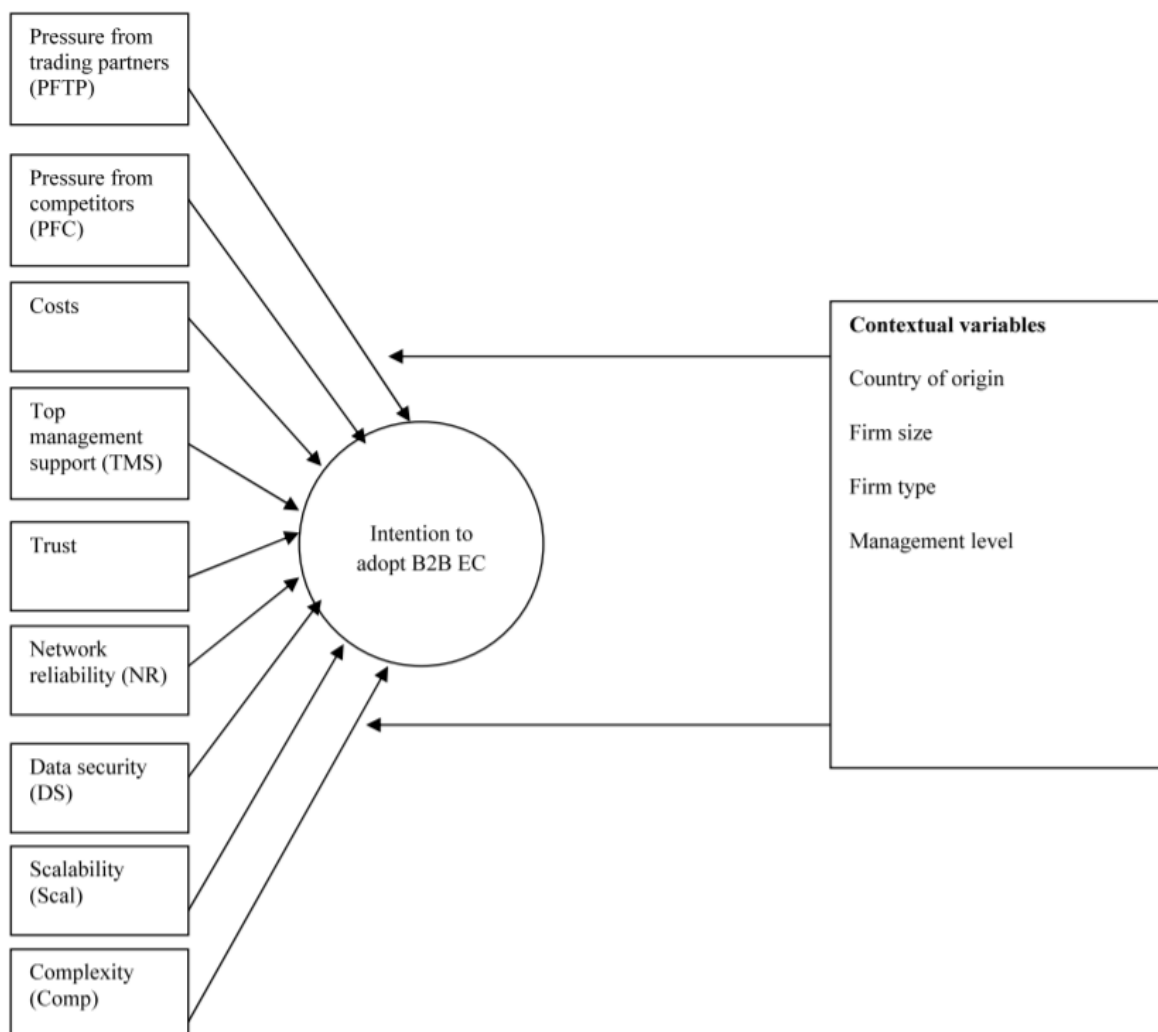


Figura 13: Fattori di adozione e variabili contestuali del B2B EC [21]

### 3.3.6 Trend del Mercato B2B

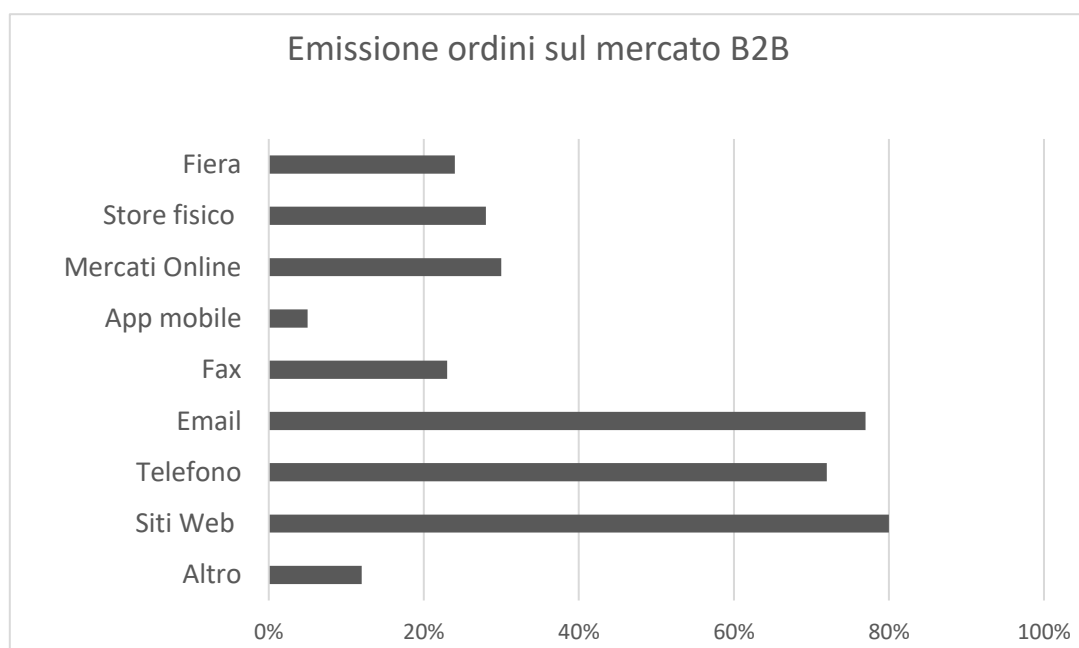
Il mondo dell'e-commerce B2B si sta rapidamente evolvendo e tale evoluzione è strettamente collegata ai significativi cambiamenti che interessano la figura dell'acquirente. Le nuove generazioni, che possiedono una maggiore istruzione e che sono cresciute nell'era tecnologica, hanno una maggiore propensione verso l'innovazione rispetto alle generazioni passate, che invece traevano le loro informazioni da annunci, schede tecniche, fiere, conferenze ecc. La consapevolezza del grande mercato online e dei suoi benefici ha fatto sì che il 55 % degli investimenti del marketing B2B si indirizzi all'innovazione digitale.

Si stima che l'industria dell'e-commerce B2B Statunitense raggiungerà 1.8 trilioni di dollari entro il 2023 e che nei prossimi cinque anni il suo tasso di crescita annuale (CAGR) sarà del 10%.

I grafici riportati in basso, realizzati sulla base di un sondaggio condotto da BigCommerce [22] su un campione di oltre 500 imprese internazionali, danno l'idea di come le imprese del settore B2B operano in merito al rilascio di nuovi ordini e al metodo di pagamento.

La Figura 14 riporta, in valori percentuali, i metodi con cui sono stati effettuati gli ordini nel mercato in questione durante l'anno 2019.

Si rileva che la maggioranza degli ordini emessi, cioè l'80%, avviene tramite siti web, seguiti dalle email (77%) e dalle chiamate telefoniche (72%). Tuttavia, sono ancora rilevanti le percentuali di ordini emessi direttamente in fiera o via fax, infatti entrambi superano la soglia del 20%. Infine, si riscontra un continuo incremento di ordini effettuati utilizzando le applicazioni mobile e i mercati online come Amazon, un chiaro segno che i rivenditori B2B ricercano approcci di vendita omnicanale.



**Figura 14:** Metodi di emissione ordini su mercato B2B a fine 2019 [22]

La Figura 15 [22] riporta i valori percentuali relativi ai metodi di pagamento utilizzati durante le transazioni B2B. È evidente come le carte di credito primeggino su tutte le altre tipologie di pagamento, con un 94%, nonostante continuino ad avere rilevanza gli assegni, i pagamenti a termine e gli ordini di acquisto, rispettivamente con un 53%, 51% e 50%.

Si vede anche la costante crescita della barra relativa ai portafogli mobili, come ad esempio Amazon Pay, ciò a dimostrazione della richiesta sempre maggiore di metodi di

pagamento veloci e sicuri. Al contrario invece, i dati mostrano un uso via via minore dei classici metodi di pagamento, come l'acconto e il saldo alla consegna.

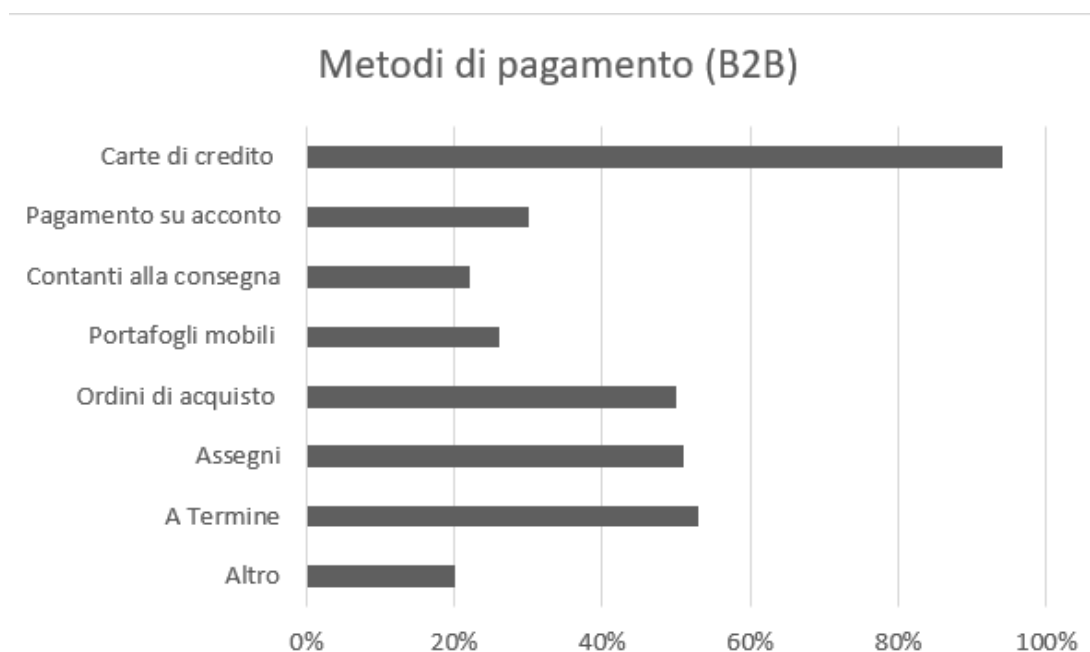


Figura 15: Metodi di pagamento nel settore B2B [22]

A conclusione di ciò, si può affermare che il mercato business-to-business è in gran parte radicato agli strumenti e alle metodologie del passato ma, non si possono non tenere in considerazione i rapidi cambiamenti in atto, perché non considerandoli si rischierebbe di compromettere la sopravvivenza stessa dell'attività. Da ciò l'importanza di concorrere nel mercato digitale, che si ritiene riesca, per il numero sempre maggiore di consensi, a ricoprire l'importante ruolo di protagonista.

La consapevolezza di ciò ha spinto le aziende del mercato B2B ad apportare delle innovazioni, le quali hanno determinato un aumento delle compravendite sulle piattaforme digitali e ad autorizzare i moderni metodi di pagamento.

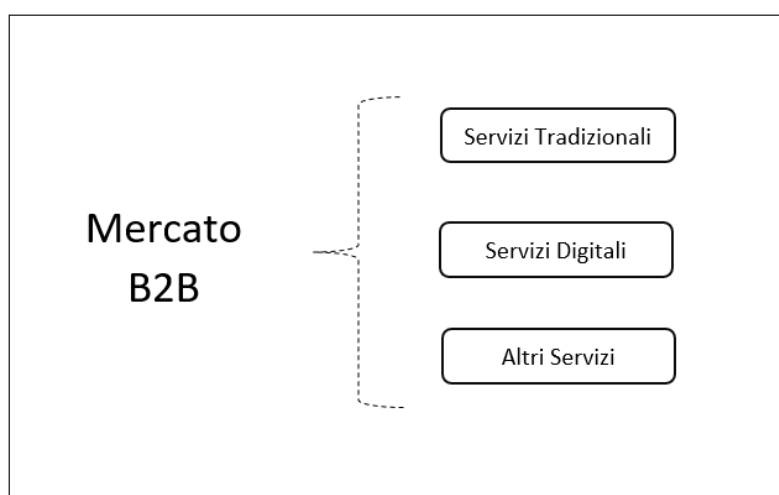
### 3.4 Individuazione del Target di Beneficiari

Dall'analisi generale sulle caratteristiche del mercato business-to-business si è passati quindi all'individuazione del giusto target di aziende clienti tra quelle che operano all'interno dell'intero mercato B2B e si è giunti alla conclusione che coloro che vanno indirizzati verso il digitale sono proprio le piccole attività, che operano nell'ambito della ristorazione, dell'industria agroalimentare, del turismo, dell'informatica, dell'istruzione, etc. In pratica tutti i piccoli imprenditori, microimprese e PMI che utilizzano la stretta di mano e non il contratto scritto ai fini della conclusione di un accordo.



Prima di analizzare dettagliatamente i diversi casi d'uso, gli attori sono stati suddivisi in base al servizio da loro fornito all'interno del mercato di riferimento. A tal proposito si osservi la Figura 16, che riporta la seguente partizione: servizi tradizionali, servizi digitali, altri servizi.

- *Servizi tradizionali:* Tale sottoinsieme raggruppa tutte quelle attività da sempre esistenti, che si praticano giornalmente in ogni parte del mondo. Basti pensare ad esempio al lavoro dell'agricoltore che fornisce prodotti a chilometro zero ai ristoranti locali, o all'attività del sarto che esegue piccole riparazioni per i titolari di negozi di abbigliamento locali o ancora al servizio fotografico prestato a una piccola-media impresa di abbigliamento del posto. Questi esempi ci aiutano a individuare le categorie su cui lo studio si focalizza e a comprendere quanto esse siano estese.
- *Servizi digitali:* Questa categoria comprende i servizi a pagamento forniti attraverso l'utilizzo di laptop e che risultano essenziali per molte attività. Si pensi ai servizi offerti negli studi professionali, come ad esempio l'assistenza software o la consulenza sul tema della sicurezza informatica. Oppure ancora, all'attività dei freelancers, ovvero dei liberi professionisti del web che operano in settori come la musica, il web design, la programmazione informatica, le traduzioni, le produzioni di film e video ecc.
- *Altri Servizi:* In quest'ultimo sottoinsieme rientrano quei servizi che non sono né tradizionali e neanche digitali, come ad esempio la collaborazione tra liberi professionisti: è questo il caso della cooperazione che viene effettuata tra un ingegnere e un architetto ai fini della partecipazione ad un bando pubblico.



**Figura 16:** Suddivisione dei casi d'uso per servizi

Nei paragrafi successivi sono riportati due casi d'uso, uno attinente al sottoinsieme dei servizi tradizionali e l'altro riguardante invece i servizi digitali.

### 3.4.1 Caso d'uso: Fornitura di prodotti a chilometro zero

Relativamente ai servizi tradizionali, lo studio si è focalizzato sull'attività di fornitura di prodotti a km 0 da parte di piccoli agricoltori per il settore della ristorazione. Tale scelta deriva dal fatto che oggi, una fetta sempre maggiore di popolazione, specialmente quella più giovane, presta molta attenzione a ciò che mangia, sia dentro che fuori casa. L'interesse verso la sostenibilità ambientale, la salute, la volontà di aiutare l'economia locale, e la freschezza dei prodotti sono criteri molto importanti che possono condizionare la scelta dei clienti. La fornitura di prodotti a km zero risulta essere senza dubbio un importante elemento che va ad avvantaggiare anche i ristoratori, i quali si trovano ad avere la disponibilità di prodotti appena raccolti, senza la necessità di far ricorso a intermediari per ottenerli. Inoltre, il ristoratore che si rifornisce direttamente dal produttore agricolo ne ricaverà un vantaggio economico, in quanto quest'ultimo riuscirà certamente a praticare prezzi più convenienti.

Dallo studio di una ricerca condotta da CHD Expert [23] relativa al mercato del consumo dei cibi fuori casa risulta come l'84% degli intervistati sia interessato ai prodotti locali, il 90% li consumi regolarmente, il 69% li consumi invece fuori casa e l'85% si dica disposto al consumo di piatti a km 0 nel ristorante.

L'indagine invece condotta da Coldiretti [24], in merito ai prodotti consumati dagli italiani in vacanza, evidenzia come, 3 soggetti su 4, ovvero il 72% degli intervistati, preferisca acquistare prodotti del posto a chilometro zero, per gustare le realtà enogastronomiche del luogo. Si è constatato inoltre come, il 19% dei vacanzieri scelga la meta in base alle pietanze culinarie del luogo, il 53% reputi il cibo uno dei criteri su cui basare la propria preferenza e solo il 5% dichiara invece di non considerarlo influente per le proprie scelte. Un altro dato che risalta dall'articolo Coldiretti, è proprio come il 42% degli intervistati dichiara di acquistare come souvenir un prodotto tipico, tra i quali risultano maggiormente graditi i formaggi, i salumi, l'olio extravergine di oliva e il vino.

Questi dati evidenziano come la ricerca dei prodotti a chilometro zero sia diventata per le famiglie italiane un elemento non più trascurabile, sia nella vita quotidiana che in vacanza.

A ciò si aggiunge l'attenzione che gli organi istituzionali hanno dimostrato di avere in merito alla vendita di prodotti a km zero. Infatti, a tal proposito si fa menzione della legge della regione Toscana che ha disposto nuovi incentivi per l'anno 2020, finalizzati al consumo di prodotti del territorio nelle mense scolastiche. Questa disposizione prevede l'utilizzo nella preparazione dei pasti di almeno il 50% di prodotti a km 0, ragion per cui è stato stanziato un budget di 500.000 euro per il 2020 e una cifra analoga da utilizzare invece nel 2021.[24]

I dati presenti nella Figura 17 sono stati tratti dal report annuale 2019 di Confcommercio [25], il quale mostra il parere dei ristoratori in merito alle forniture di prodotti a chilometro zero e dice come, l'86% dei soggetti intervistati esprima un giudizio positivo nei confronti dei prodotti a km 0, e come tale giudizio dipenda dalla qualità e dalla freschezza dei prodotti, dal rispetto ambientale e dalla fiducia nella controparte. Chi invece esprime un giudizio negativo, solamente il 6.2% lo fa per ragioni di scarsa reperibilità del prodotto, quantità limitate, prezzi eccessivi e bassi livelli di servizio come ad esempio ritardi di consegna, bassa trasparenza sui prezzi e gestione dei reclami inadeguata. Il report rileva anche come il 63.5% dei ristoratori sia solito richiedere forniture di prodotti locali in maniera frequente, il 34.2% invece effettui tali ordini occasionalmente e solo il 2.3% non li emetta mai.

Giudizio	%		Motivazioni	%		Frequenza di utilizzo	%
Molto positivo	40,7	→	Più qualità	45,2	→	Spesso	63,5
Positivo	46,2		Più freschezza	66,6			
Negativo	6,2	→	Maggiore fiducia nel produttore	70,9	→	Qualche volta	34,2
			Rispetto all'ambiente	37,5			
			Prezzi eccessivi	37,5		Mai	2,3
			Scarsa reperibilità	62,5			
Quantità limitate	37,5						
Indifferente	6,9		Basso livello di servizio	50			

**Figura 17:** I prodotti a chilometro zero nella ristorazione di qualità [25]

Questi dati confermano il giudizio positivo che i clienti e i ristoratori hanno nei confronti dei prodotti a chilometro zero e come questa vendita sia destinata a crescere nel tempo. C'è comunque da evidenziare come l'accordo tra il produttore e il ristoratore difficilmente sia tutelato attraverso un contratto scritto. Invece, è attraverso la forma verbale che le parti concretizzano il loro impegno, impegno che porta il piccolo agricoltore di zona a rifornire periodicamente il ristoratore dei prodotti a lui richiesti e il ristoratore a sua volta a corrispondergli quanto dovuto. Lo studio, il cui oggetto è la compravendita tra l'agricoltore e il ristoratore di prodotti agricoli, si propone di trovare soluzioni vantaggiose per entrambe le parti, assicurando loro una completa tutela. Tutela che non si riuscirà ad ottenere con l'accordo verbale. È trasferendo invece l'accordo sul digitale che tale tutela verrebbe a concretizzarsi. In tal modo le parti riuscirebbero ad ottenere una garanzia completa che altrimenti sarebbe loro negata.

### 3.4.2 Caso d'uso: I freelancers e il mondo della Gig Economy

Nell'ambito dei servizi digitali lo studio si è incentrato sui lavoratori freelancers. La parola freelance deriva dall'omonimo termine della lingua inglese free-lance, letteralmente lancia libera, utilizzata per indicare il ruolo del mercenario, ovvero un soldato professionista non legato emotivamente ad un comandante specifico, ma al servizio di chiunque lo paghi.

Come si è potuto intuire dal significato letterale del termine, il freelancer è un lavoratore indipendente che offre le sue prestazioni sotto pagamento a società, organizzazioni e privati.

I vantaggi legati a questa tipologia di lavoro sono: orari flessibili, minori spese quotidiane, ferie autogestite e guadagni molto alti se le cose dovessero andare bene.

Gli svantaggi invece sono: maggiori responsabilità, assenza di stipendio fisso, investimento iniziale e totale autogestione contabile e amministrativa.

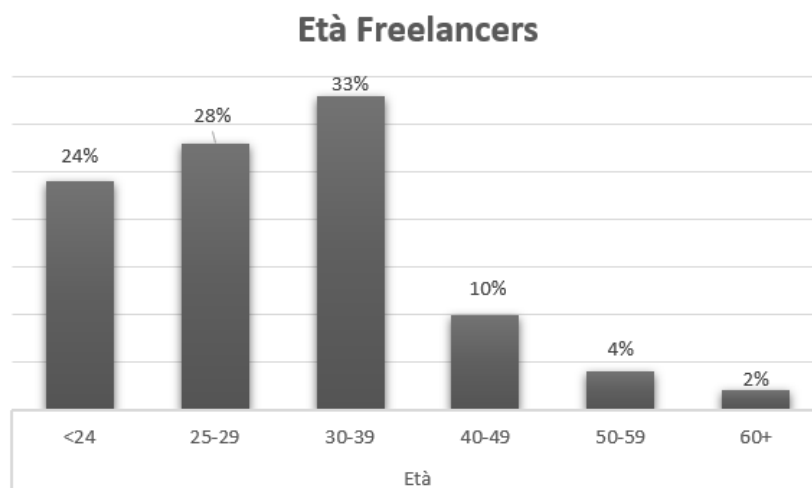
Il freelancer è una figura che si sta diffondendo molto, poiché le aziende sono solite ricercare liberi professionisti quando non possiedono determinate competenze interne o quando hanno bisogno di una competenza spot. Anche i privati preferiscono affidarsi ad un libero professionista piuttosto che ad un'azienda al fine di ottenere servizi più veloci e personalizzati. Pertanto, tale figura è in continuo aumento in tutto il mondo e l'Italia, in particolare, conta circa 3,6 milioni di lavoratori indipendenti, dato sopra la media europea del 9.2%. [26]

Il mercato europeo negli ultimi cinque anni ha visto un'impennata nella crescita dei lavoratori freelance che, insieme allo sviluppo di nuove tecnologie, ha generato organizzazioni e reti di persone che operano senza esser legate a un particolare luogo di lavoro e senza possedere una formale assunzione. È nata così la Gig Economy, ossia un modo di *“guadagnarsi da vivere, o integrare il proprio reddito, facendo lavori saltuari, senza contratto, solo quando viene richiesto o quando si può”* [27]. La frase appena citata fa capire cosa si intende per Gig Economy ed è riportata in un articolo del giornale “La Stampa”. In esso si dice che in questa nuova economia, la retribuzione è corrisposta per singola prestazione e può richiedere da pochi minuti ad un'intera giornata di lavoro. Alcuni esempi di lavori tipici della Gig Economy sono: le consegne a domicilio, il baby-sitting, le ripetizioni private, il noleggio di un'auto privata come taxi, le traduzioni ecc.

In sostanza, questi lavori, anche definiti “accessori”, sono in costante e rapida crescita. A confermarlo è un'indagine sul mercato del lavoro condotta dall'Istat nell'anno 2017 [27], che dimostra come nel giro di cinque anni si è quadruplicato il numero di datori di lavoro che ricercano soggetti che praticano questa tipologia di lavori.

Per tale motivo, lo studio si focalizza anche sui lavoratori autonomi del digitale, che in cambio di una prestazione lavorativa, generalmente di un servizio, ricevono una retribuzione. In particolar modo, l'interesse è rivolto a tutti coloro che svolgono traduzioni online, ripetizioni private, sviluppo di siti web, consulenze informatiche ecc. Sono queste attività che raramente vengono tutelate da un contratto e il cui pagamento avviene il più delle volte in nero.

Si riportano e si commentano di seguito i dati estratti dal report *"The Payoneer Freelancer Income Survey"*, condotto dall'azienda Payoneer in merito ad un'intervista fatta a 21.000 freelancers in 170 paesi differenti [28]. Da tale indagine è emerso come la maggioranza di questi lavoratori abbia meno di 30 anni (il 52%) e come invece solo il 6% abbia già compiuto il cinquantesimo compleanno (Figura 18). Si ha a che fare quindi con dei soggetti per lo più di giovane età.



**Figura 18:** Percentuale di freelancers in base all'età [28]

La figura 19 rappresenta un istogramma che indica la percentuale di freelancers che percepisce la tariffa oraria corrispondente a ciascuna barra.

È evidente come la maggioranza, ovvero il 24% del totale, chieda da 6 a 10 dollari l'ora per il lavoro svolto, e come il 70% di tutti i lavoratori non prenda più di 25 dollari l'ora. Dal report si rileva inoltre come la paga sia direttamente proporzionale agli anni di esperienza del lavoratore, infatti con l'aumentare dell'età è richiesta una retribuzione via via superiore.

## Percentuale di freelancers per tariffa oraria

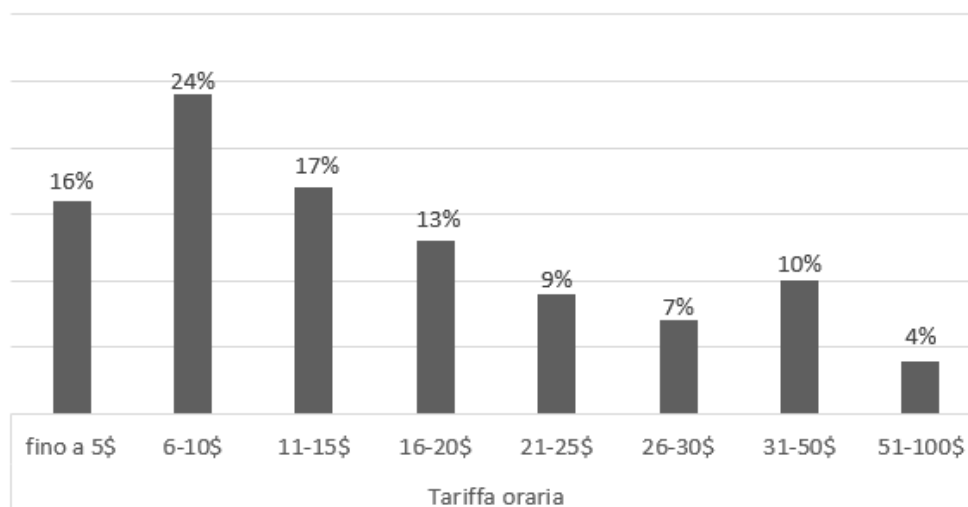


Figura 19: Percentuale di freelancers per tariffa oraria [28]

I dati presenti nella Figura 20 e nella Figura 21 sono senza dubbio quelli che meritano una maggiore attenzione.

Per la modellazione dei dati presenti all'interno della Figura 20, si è chiesto agli intervistati di classificare il livello di soddisfazione del loro reddito da freelancer su una scala da 1 a 10. Dal grafico risulta come la maggioranza di liberi professionisti (il 56%) vorrebbe idealmente guadagnare di più o addebitare tariffe orarie più elevate al cliente e, solo il 5% si reputi pienamente appagato dal reddito percepito tramite questo mestiere.

## Cosa i Freelancers vorrebbero migliorare



Figura 20: Livello di soddisfazione dei freelancers in base al reddito [28]

La Figura 21 riporta le risposte che i freelancers intervistati hanno dato in merito alla domanda che è stata posta loro su che cosa andrebbero a migliorare all'interno proprio

lavoro. È stata data loro la possibilità di scegliere fino a tre opzioni di una lista contenente più elementi. Non sorprende come la maggior parte dei liberi professionisti vorrebbe guadagnare di più (68%) e scoprire nuovi modi per acquisire nuovi clienti e ottenere più lavoro (52%). Al di là di questi obiettivi piuttosto ovvi, i liberi professionisti vorrebbero anche lavorare con più clienti internazionali (34%), migliorare l'efficienza del proprio lavoro (32%) ed apportare migliorie nel metodo di pagamento. Tra queste ultime richieste emerge la rapidità nel ricevere il saldo (16%) e la semplicità nel processo di pagamento per il lavoratore (11%) e per il cliente (9%).

### Grado di soddisfazione dei Freelancers

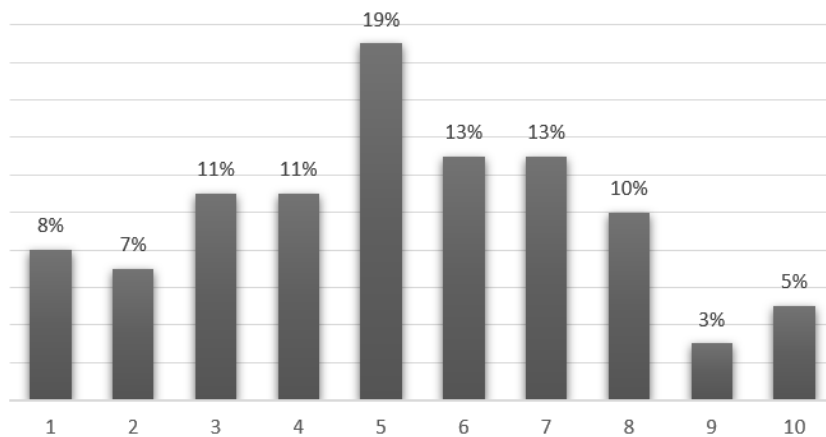


Figura 21: Aspetti da migliorare dal punto di vista dei freelancers [28]

L'intervista realizzata da Daniel Huffman e rivolta solo ai freelancers cartografici, ha dimostrato come la maggioranza delle prestazioni lavorative di questa tipologia di lavoratori non sia regolata da un contratto scritto. Dalla Figura 22, infatti risulta come solo il 41% degli accordi lavorativi in genere sia formalizzato per iscritto.[29]

### Stipula di contratti

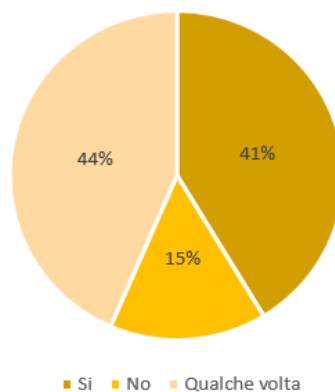


Figura 22: Stipula di contratti [29]

I valori precedentemente riportati, sono indicativi del fatto che i freelancers sono una categoria poco tutelata a causa del ridotto uso di contratti scritti.

## 3.5 Analisi degli Stakeholders

In economia, con il termine stakeholder si intende qualsiasi soggetto interessato e influente nei confronti di un'iniziativa economica, un'azienda o qualsiasi altro progetto. Si può riferire anche ad un gruppo di soggetti che ha un interesse legittimo nei confronti di un progetto e delle sue attività, passate, presenti e future, e il cui contributo, volontario o involontario, risulta essenziale al suo successo.

L'analisi degli Stakeholders permetterà di individuare gli attori coinvolti in questo studio, focalizzando l'attenzione sull'interesse che li lega al problema, sulla reale motivazione che li spinge verso il cambiamento e sulle possibili azioni da effettuare per soddisfare gli interessi degli stessi.

Si procederà con la suddivisione tra stakeholders interni e stakeholders esterni al progetto:

### 3.5.1 Stakeholders Esterni

All'interno di questa categoria di attori si inseriscono coloro che non partecipano direttamente allo studio e allo sviluppo del progetto, ma che ricaveranno dei benefici dall'utilizzo della soluzione.

- *Cliente*: Con questo termine si indica la categoria dei piccoli imprenditori interessati a soddisfare delle particolari domande, motivati ad ottenere una più vasta scelta di offerte e interessati ad usufruire di un modo semplice, rapido ed economico di gestione delle eventuali controversie.
- *Fornitore*: Raggruppa tutti soggetti che vendono beni e servizi e il cui desiderio è quello di incrementare il proprio business. La motivazione che li spinge ad accettare un cambiamento è la possibilità di ampliare il numero di clienti e di ricevere pagamenti istantanei.
- *Avvocato*: Rappresenta la categoria di professionisti esperti di diritto, che presta assistenza in favore di una parte nel giudizio di una controversia. Non possiede una concreta motivazione per partecipare al cambiamento, ma è interessata a ricevere nuovi incarichi per incrementare i guadagni.



### 3.5.2 Stakeholders Interni

Il progetto oggetto di tesi, prevede la collaborazione dei tre enti descritti di seguito:

- *Fondazione LINKS*: Fondata dal Politecnico di Torino e dalla Fondazione Compagnia di San Paolo, Fondazione LINKS si afferma come un centro di ricerca di riferimento per le organizzazioni che intendono aumentare la propria competitività e per gli studenti che desiderano crescere in un ambiente stimolante. Propone processi innovativi e tecnologie di ultima generazione per vincere le sfide poste dalla trasformazione digitale. *Cefriel*: Fondata dal Politecnico di Milano, Cefriel si presenta come un centro di innovazione che crea prodotti, servizi e processi digitali, attraverso la partecipazione a programmi di ricerca nazionali e internazionali.
- *GFT*: Fondata nel 1987, GFT (Gesellschaft für Technologietransfer) è una multinazionale tedesca, attiva nel settore della trasformazione digitale, principalmente in ambito finanziario, bancario e assicurativo.

La ricerca sul modo di portare innovazione nelle transazioni business-to-business è senza dubbio la motivazione che ha spinto questi enti a intraprendere una collaborazione. In esse, c'è l'interesse che si realizzi, nel mercato B2B, una nuova soluzione basata sulla tecnologia Blockchain, al fine di ampliarne la domanda e l'offerta, garantire pagamenti istantanei, transazioni sicure e un meccanismo di gestione delle dispute rapido ed economico nel quale la figura dell'avvocato non può certamente mancare. Al fine di soddisfare gli interessi di tutti gli stakeholders, si è sviluppata un'applicazione decentralizzata (DApp), che costituisce l'oggetto di tesi.

### 3.5.3 Matrice per l'Analisi degli Stakeholders

La matrice per l'analisi degli stakeholders, in Figura 23, è un modello che schematizza le informazioni relative ad ogni attore, suddivise in base ai loro interessi in merito al problema, alla motivazione che li spinge verso il cambiamento e alle azioni che si intende intraprendere durante il progetto per soddisfare gli interessi degli stessi.

STAKEHOLDERS ANALYSIS MATRIX	Stakeholder	Interesse	Motivazione a realizzare il cambiamento	Azioni effettuate per gli interessi dello Stakeholder
	<ul style="list-style-type: none"> <li>Fondazione LINKS</li> <li>CEFRIEL</li> <li>GFT</li> </ul>	<ul style="list-style-type: none"> <li>Fornire una nuova soluzione per il commercio B2B basata sulla Blockchain</li> <li>Offrire una piattaforma per ampliare la domanda e l'offerta per il mondo B2B</li> <li>Garantire pagamenti immediati</li> <li>Garantire transazioni sicure</li> <li>Assicurare una semplice ed economica gestione delle dispute</li> <li>Creare un nuovo portale per la professione dell'avvocato</li> </ul>	<ul style="list-style-type: none"> <li>Fare ricerca</li> <li>Portare innovazione</li> </ul>	<ul style="list-style-type: none"> <li>Sviluppo di una Dapp (Decentralized Application)</li> </ul>
	Cliente	<ul style="list-style-type: none"> <li>Cercare un lavoratore che soddisfi la propria richiesta</li> </ul>	<ul style="list-style-type: none"> <li>Ottenere una maggiore offerta</li> <li>Essere tutelati in caso di controversia</li> </ul>	<ul style="list-style-type: none"> <li>Meccanismo di gestione delle eventuali dispute rapido ed economico</li> <li>Nuova piattaforma per ricercare i fornitori ideali</li> </ul>
	Fornitore	<ul style="list-style-type: none"> <li>Aumentare il proprio business</li> </ul>	<ul style="list-style-type: none"> <li>Ricevere pagamenti istantanei</li> <li>Trovare nuovi clienti</li> </ul>	<ul style="list-style-type: none"> <li>Nuova piattaforma per aumentare la domanda</li> <li>Meccanismo basato sul congelamento di uno Stake destinato al pagamento istantaneo</li> <li>Meccanismo basato sul concetto di reputazione online</li> </ul>
	Avvocato	<ul style="list-style-type: none"> <li>Aumentare il numero di incarichi</li> </ul>	<ul style="list-style-type: none"> <li>Motivazione a realizzare il cambiamento assente</li> </ul>	<ul style="list-style-type: none"> <li>Nuova piattaforma per aumentare gli incarichi</li> <li>Meccanismo basato sul concetto di reputazione online</li> </ul>

Figura 23: Matrice degli Stakeholders

### 3.6 Analisi della Concorrenza

L'analisi della concorrenza, anche definita benchmarking, è un'attività necessaria per le aziende che oggi desiderano occupare una posizione strategica in ambito digitale. È uno studio approfondito sui concorrenti e sui prodotti che gli stessi offrono sul mercato, con l'obiettivo di individuare i rispettivi punti di forza e di debolezza e scoprire le lacune del settore in cui agiscono. Se fatta correttamente, l'analisi della concorrenza può portare numerose informazioni qualitative e quantitative, in grado di comunicare le tendenze di mercato e suscitare idee per nuovi "concept" di prodotto o servizio.

Nella prima fase si stabilisce chi sono i concorrenti, suddividendoli in competitor diretti, se vendono un prodotto o servizio simile e sostituibile a quello che si intende offrire con il presente progetto, e competitor indiretti, se invece vendono un bene o servizio differente, ma rispondente alle stesse esigenze.

Nonostante l'e-commerce B2B sia un settore ancora poco esplorato, attraverso un'accurata ricerca si è giunti all'individuazione di tre possibili prodotti concorrenti:

## Indiretti

- PactSafe

## Diretti

- Emanate
- Heymate

Dopo aver individuato i principali competitors, si procede con la seconda fase. Essa consiste nell'analisi dettagliata delle caratteristiche e delle funzionalità dei prodotti concorrenti, così da comprenderne i punti di forza e di debolezza. Di seguito si riportano le informazioni ricavate da tale ricerca.

- **PactSafe:** è una soluzione basata su cloud, rivolta alle aziende che desiderano modificare il processo di contrattazione e gestione dei contratti nei settori B2B e B2C. PactSafe offre la possibilità di attivare dei promemoria automatizzati, fornisce una dashboard per facilitare il monitoraggio delle attività e l'analisi di documenti e presenta dei modelli di contratto personalizzabili approvati legalmente. Inoltre, tale soluzione offre la possibilità di usufruire della classica firma elettronica (e-signature), dell'approvazione di un accordo tramite un semplice click (metodo click-to-sign) e consente la ricezione di contratti via SMS, sottoscrivibili con la risposta "Acconsento" (metodo text-to-sign). In sintesi, PactSafe grazie all'integrazione di svariate funzionalità mette a disposizione un modo semplice di creare, approvare e gestire i contratti che possiedono un elevato volume di termini e condizioni. [30]
- **Emanate:** è una soluzione blockchain che stravolge l'attuale modello di industria musicale. Tale prodotto dà la possibilità a chiunque di entrare a far parte del nuovo ecosistema, come artista o come ascoltatore, e mette anche in comunicazione musicisti di tutto il mondo, aprendo così la strada a nuove forme di cooperazione. Gli artisti possono così realizzare delle collaborazioni musicali senza necessariamente dover arrivare a conoscere le parti e instaurare una fiducia reciproca. Ciò è possibile grazie alla presenza di un contratto intelligente, che conserva la proprietà intellettuale dei brani e distribuisce istantaneamente il pagamento ai musicisti sulla base degli accordi stabiliti e tracciati sulla blockchain. Ciò significa che, nel momento in cui un utente ascolta un brano, il contratto intelligente suddivide e autorizza la distribuzione automatica del pagamento, in token *MXN*, sulla base delle percentuali concordate e memorizzate on chain. Tale piattaforma, oltre a garantire il vantaggio del pagamento istantaneo, apre un mercato per la classe media musicale, permettendole un guadagno dell'85% dei ricavi. Si ricorda infatti, che oggi, nel settore musicale, solamente il 12% degli incassi di un brano finiscono nelle tasche dell'artista. Inoltre,

anche gli ascoltatori sono interessati a partecipare attivamente al miglioramento della piattaforma, poiché sulla stessa è presente un meccanismo che dà la possibilità, a chiunque crei playlist e aumenti la visibilità dei musicisti, di vincere delle ricompense (in token). [31][32]

- **heymate:** è una soluzione basata su blockchain e rivolta al mondo della “Gig economy”. È il prodotto che più si avvicina alla soluzione proposta nella tesi, poiché crea un’alternativa alla stretta di mano fisica per tutti i piccoli imprenditori e i lavoratori autonomi. È un’applicazione mobile che permette agli utenti di concretizzare la “stretta di mano digitale” al fine di creare, promuovere, accettare e concludere dei contratti. È un sistema che redige dei template di offerte legalmente valide e include il controllo della reputazione di ogni partecipante. In sostanza, ogni lavoratore può preparare un’offerta, valida legalmente, e pubblicarla sui principali social network (LinkedIn, Facebook e Twitter), in modo che sia visibile a tutti gli utenti heymate. Questi ultimi possono verificare la reputazione del lavoratore ed eventualmente decidere se accettare o rifiutare con un semplice click. Inoltre, il pagamento (in token) è sicuro ed istantaneo, poiché è trattenuto in un deposito a garanzia fino al momento della conclusione dell’affare. [33]

### 3.6.1 Tabella di Benchmarking

La tabella di Benchmarking è un framework utile ad organizzare i dati raccolti sui diversi competitor. Tale strumento permette di delineare le caratteristiche della soluzione oggetto di tesi, e di metterle a confronto con le caratteristiche dei competitor. Sarà così possibile effettuare un rapido confronto tra le varie soluzioni, ed individuare i punti di forza e di debolezza di ciascuna (Tabella 4).

Dall’osservazione della Tabella 4 riportata in basso, si dimostra come il prodotto oggetto di tesi abbia la maggior parte delle caratteristiche ritenute importanti dall’utente:

- È orientato al mercato B2B e rivolto ad un ampio numero di utenti (Microimprenditori, piccole imprese, PMI, freelancers e lavoratori della Gig economy).
- Trasmette affidabilità e sicurezza, grazie all’utilizzo della tecnologia Blockchain
- Utilizza un contratto intelligente per gestire le diverse fasi dell’accordo
- Utilizza dei Token
- Incrementa la domanda e l’offerta di tale mercato
- Garantisce pagamenti sicuri ed istantanei
- Offre un meccanismo di gestione delle eventuali controversie
- Offre una schermata per il monitoraggio delle attività in corso

Inoltre, attraverso l'osservazione di ciò che i prodotti concorrenti offrono sul mercato, è emerso che la soluzione del presente progetto potrebbe essere oggetto di ulteriori sviluppi, al fine di soddisfare meglio i bisogni del consumatore. Si potrebbe infatti:

- Creare una versione mobile
- Aggiungere notifiche intelligenti per permettere dei promemoria personalizzati
- Integrare la piattaforma con i principali social network per conquistare un maggior numero di utenti
- Utilizzare modelli di contratti legalmente validi. La soluzione attuale non li contempla, ma ne prevede il soddisfacimento attraverso l'uso del Ricardian Contract. Con questo termine si intende un metodo che permette di registrare un documento come un contratto legale (questo aspetto sarà spiegato dettagliatamente all'interno del capitolo finale "Conclusioni e Prospettive future").

	PactSafe	Emanate	heymate	La soluzione del progetto
Orientamento al mercato B2B	Si	Si	Si	Si
Basata su Blockchain (Affidabilità e sicurezza)	No	Si	Si	Si
Smart Contract	No	Si	Si	Si
Token	No	Si	Si	Si
Promemoria personalizzati	Si	No	No	No
Dashboard per monitoraggio attività	Si	No	No	Si
Uso mobile	Si	No	Si	No
Incremento di domanda e offerta	No	Si	Si	Si
Template di contratti legalmente validi	Si	No	Si	Al momento no (Prospettiva futura)
Integrazione con i social network	No	No	Si	No
Pagamenti sicuri	No	Si	Si	Si

Pagamenti istantanei	No	Si	Si	Si
Risoluzione delle controversie	No	No	No	Si

**Tabella 4:** Tabella di Benchmarking

## **4. DLT-based Digital Handshaking: la nuova piattaforma per la stratta di mano digitale**

Nel presente capitolo si illustrerà la soluzione proposta in questo lavoro di tesi, successivamente implementata nel Capitolo 5, per far fronte alle criticità emerse dall'analisi di mercato descritta al Capitolo 3.

Si partirà dall'individuazione e dalla descrizione delle problematiche con la successiva motivazione del perché il progetto in esame fornirebbe una valida e competitiva risoluzione a tali questioni. Successivamente, nel Capitolo 5, verrà illustrata l'architettura del sistema completa di diagrammi UML, a chiarimento delle logiche interne discusse in fase di brainstorming, unitamente agli strumenti e ai linguaggi utilizzati durante le fasi di scrittura di back-end e di front-end.

### **4.1 Criticità degli accordi non-digital**

Nei primi due decenni del nuovo millennio si è verificato un progressivo aumento dell'uso delle tecnologie digitali in termini di tempo, funzioni e numero di utenti.

Strumenti come smartphone, laptop, applicazioni e, in generale, internet, vengono quotidianamente utilizzati dalla maggioranza della popolazione mondiale, sia nel lavoro che nel tempo libero, a dimostrazione del fatto che la digitalizzazione sta rivestendo un ruolo sempre più importante ed indispensabile nel panorama economico.

Ad inizio 2020, l'avvento della pandemia da Coronavirus ha portato, tra le varie conseguenze, ad una spinta verso uno stile di vita 'digitale'. Negli ultimi mesi si sta assistendo infatti ad un uso massiccio della rete e degli strumenti informatici per riempire il tempo libero, per comunicare con il resto del mondo, per rimanere aggiornati e soprattutto per studiare e lavorare. In questa drammatica realtà in cui un virus invisibile ci sta violentemente separando è proprio l'informatica che ci tiene uniti in un universo artificiale. Per tale motivo, oggi più che mai, è fondamentale per tutte le aziende disporre di adeguati strumenti digitali al fine di rimanere competitive sul mercato.

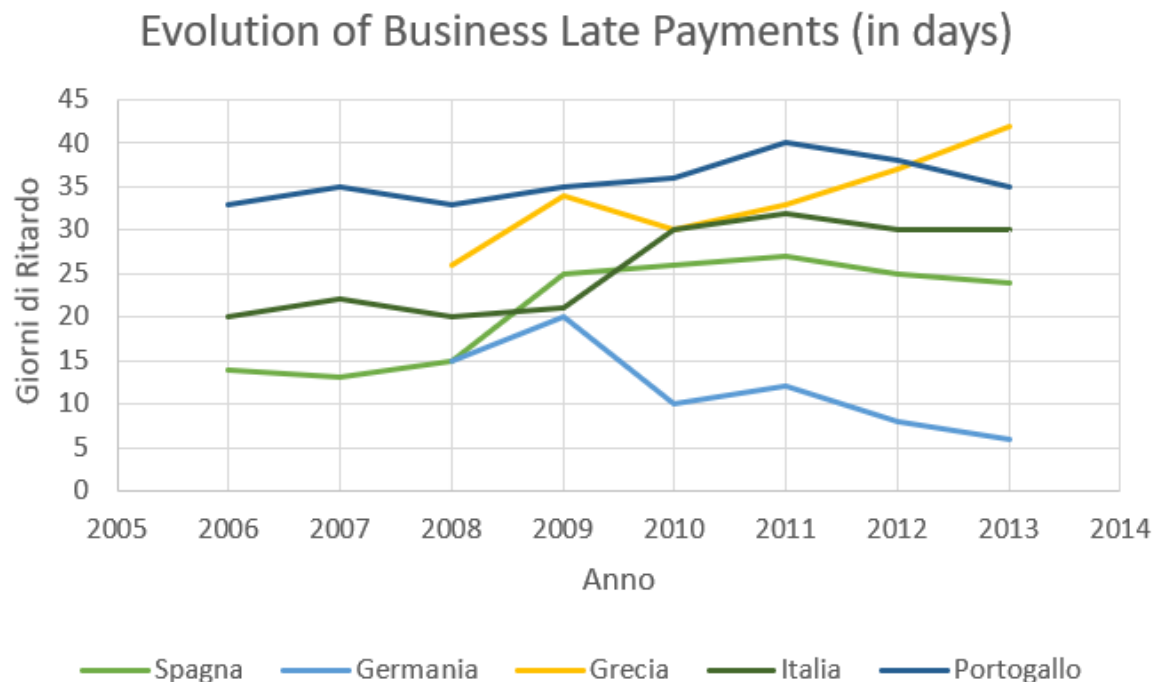
Pertanto, il principale obiettivo del progetto in cui rientra il presente lavoro di tesi è proprio quello di portare sul digitale tutte le attività che oggi operano attraverso degli accordi in presenza, con una stretta di mano fisica, e permettere loro di rimanere al passo in un'economia in continua evoluzione, i cui effetti saranno chiari soltanto con il tempo.

Potranno beneficiare di tale servizio sia le attività tradizionali, come ad esempio il contadino che vende i propri prodotti al mercato, che quelle digitali, come lo sviluppatore che crea siti web su commessa. Entrambe le tipologie di servizi, tradizionali e digitali, si

basano su accordi fisici tra clienti e fornitori, accordi che non danno nessuna garanzia in termini di pagamento, tempistiche e qualità. Infatti, non vi è nessun contratto scritto che tuteli i contraenti nel caso in cui una delle due parti non dovesse rispettare la parola data. A supporto di ciò il grafico di Figura 24, riporta i risultati dello studio di William Connell relativamente all’impatto negativo dei ritardi nei pagamenti nelle transazioni “Business to Business” [34]. Il grafico appena citato mostra come i cinque paesi con i peggiori risultati in termini di giorni di ritardo nei pagamenti sono, in ordine decrescente: Portogallo, Grecia, Italia, Spagna e Germania.

Questi valori elevati evidenziano l’esistenza di una reale problematica legata alla gestione dei pagamenti che andrebbe a danneggiare i fornitori di beni e servizi.

Un altro aspetto negativo che, questa volta, va contro gli interessi dei clienti è legato agli elevati costi che un soggetto dovrebbe affrontare nel caso in cui non fosse soddisfatto del servizio/bene ricevuto. L’apertura di una disputa richiede, infatti, delle ingenti spese in termini di tempi e costi, spesso anche salati, generati dalle imposte dovute e dal compenso dell’avvocato. In molti casi infatti, quando il prezzo del bene/servizio acquistato non è poi così elevato, i clienti tendono a non chiamare in causa il fornitore anche nel caso in cui non ritengano il bene conforme alle specifiche, e questo perché non gli converrebbe in base ad un’analisi di costi-benefici.



**Figura 24:** Giorni di ritardo nei pagamenti nel settore B2B in diversi paesi europei [34]



In breve, per riepilogare quanto discusso in questo paragrafo, quattro sono i problemi che si cercherà di risolvere, o per lo meno limitare, con la soluzione proposta in questa tesi:

1. Elevato numero di lavoratori che ancora esercitano il proprio mestiere senza possedere un'identità digitale
2. La maggior parte degli accordi tra clienti e fornitori avviene senza la stipula di un contratto che li tuteli
3. Ritardi nei pagamenti da parte dei clienti
4. Elevati costi per la gestione delle dispute

## **4.2 Soluzione Proposta**

Alla luce delle criticità e problematiche precedentemente discusse, nel presente paragrafo viene introdotta la soluzione elaborata in questo lavoro di tesi. La soluzione ivi proposta consiste nell'implementazione del Digital Handshake basato sulla tecnologia Blockchain, di tipo EOS, al fine di fornire un modo sicuro, veloce e affidabile per gestire i contratti e gli accordi tra piccoli fornitori/venditori e i loro clienti.

Il tutto è reso possibile dallo sviluppo di un'applicazione decentralizzata (o DApp) che goda di tre proprietà fondamentali quali scalabilità, sicurezza e decentralizzazione, e che permetta ai nostri venditori e clienti di operare sotto le regole di uno Smart Contract (SC). Quest'ultimo (SC) viene stipulato al fine di garantire entrambe le parti contraenti l'accordo. Difatti, con la soluzione messa in atto, da un lato i fornitori possono vendere i propri prodotti o servizi in modo sicuro ottenendo pagamenti istantanei, e, dall'altro, gli acquirenti insoddisfatti sono in grado di aprire una disputa senza dover investire ingenti quantità di tempo e denaro. Tali garanzie sono assicurate grazie all'implementazione di una logica di processo elaborata ma efficiente, che inserisce all'interno del nostro servizio la figura dell'avvocato. Quest'ultimo, tramite un sistema di votazione a favore del fornitore o del cliente, ha potere decisionale sulla sentenza finale della contesa. La spiegazione dettagliata dei flussi logici in cui si struttura tale approccio viene rimandata al paragrafo relativo ai diagrammi di flusso (Paragrafo 5.2).

### **4.2.1 Perché EOS (EOS vs Ethereum)**

La piattaforma utilizzata in questo lavoro, per la creazione di smart contract, è EOS, preferita alla più comunemente utilizzata "Ethereum" introdotta al Capitolo 2.

Ultimamente si discute molto della sfida tra EOS ed Ethereum, tanto da sentir parlare di EOS in termini di “Ethereum Killer”. Tale definizione rende chiara la rivalità tra questi due colossi.

Sebbene Ethereum sia la principale piattaforma per la creazione di smart contract e app decentralizzate ne sono altrettanto noti i suoi limiti. In particolar modo Ethereum è in grado di ottenere simultaneamente solamente due delle tre proprietà che dovrebbero caratterizzare le criptovalute: decentralizzazione e sicurezza a discapito della scalabilità. La sicurezza è garantita dal fatto che il Proof of Work richiede una potenza di calcolo, tempi e costi esorbitanti per mettere in atto un attacco del 51% (vedi Paragrafo 2.1.2), mentre la decentralizzazione è garantita dal fatto che chiunque può partecipare alla creazione dei blocchi.

Per quanto riguarda il limite della scalabilità la causa è da attribuire al fatto che le informazioni di ogni blocco devono essere trasmesse a tutti i nodi della rete e, al momento, l'ordine di grandezza del numero di nodi è circa  $10^4$ . Tale valore spiega perché il numero di transazioni al secondo (*throughout*) è limitato ad un valore di circa 15 (al massimo 18).

La validazione delle transazioni, realizzata attraverso il processo di *mining*, richiede un certo periodo di tempo, dipendente dal livello di congestione della rete in un dato momento e dal prezzo (*gas price*), che l'utente sarebbe disposto a pagare. Dal momento che la blockchain di Ethereum genera un nuovo blocco mediamente ogni 14 secondi, l'utente dovrà attendere circa 10-20 secondi per vedere la propria transazione approvata (sia essa un trasferimento di moneta, definita *Ether*, o l'invocazione di una funzione di uno smart contract che provoca un aggiornamento dello stato).

Oggigiorno gli utenti hanno aspettative molto elevate quando navigano sul web e si aspettano che le pagine e, in generale i servizi, vengano erogati in modo pressoché istantaneo. L'attesa necessaria per il mining, presente nelle applicazioni web decentralizzate basate su Ethereum, potrebbe quindi compromettere parzialmente l'esperienza dell'utente.

Altro limite di Ethereum è l'impossibilità di aggiornare facilmente gli smart contract: dal momento che il codice sorgente degli smart contract è memorizzato all'interno della blockchain, questo diventa immutabile e permanente, rendendo di fatto impossibile aggiornare direttamente il contenuto di uno smart contract già pubblicato. Questo non impedisce in maniera assoluta l'aggiornamento di un'applicazione decentralizzata, ma lo complica notevolmente, costringendo gli sviluppatori a ricorrere a diversi stratagemmi più articolati che comportano la creazione di un nuovo contratto che dovrà interagire in qualche modo con il precedente, divenendo di fatto una sorta di strato di secondo livello.

La scelta di utilizzare EOS risiede nel fatto che tale piattaforma sembrerebbe sopperire i limiti di Ethereum. Infatti, EOS, oltre a permettere una facile aggiornabilità degli SC, si basa sul Delegated Proof of Stake (DPoS), evoluzione del PoS, in cui viene coinvolta l'intera rete per la conferma di una transazione.

Nel sistema DPoS tale onere è a carico di un numero limitato di delegati, eletti dall'intera rete con un sistema di democrazia rappresentativa dei consensi pesati in base al numero di *token* posseduti dai votanti. Ogni nodo può diventare a turno un produttore di blocco, ma può essere anche rimosso da questo ruolo se non dovesse rispettare le regole della blockchain.

Quindi, in sostanza, il sistema di EOS possiede 21 *block producer* e non più l'intera rete, permettendo così di aumentare in modo netto il numero di transazioni processate ogni secondo. Ad oggi, infatti, si stima sia in grado di processare da 1000 a 8000 tps e si pensa possa raggiungere decine di migliaia di tps in futuro.

Le motivazioni che hanno avuto un peso rilevante al momento della scelta tra EOS e Ethereum sono legate alle performance, al codice e al costo. Per quanto riguarda le performances, oltre alla velocità di utilizzo, EOS cerca di creare una piattaforma user-friendly, con più funzionalità di base, come ad esempio la gestione del database e la gestione dell'account (account identificati da nomi leggibili dall'uomo di lunghezza compresa tra 1 e 12 caratteri che possono includere lettere dalla a alla z, numeri da 1 a 5 e punti opzionali) così da richiedere minor codifica e migliorare l'esperienza dell'utente. Relativamente al codice invece, a differenza di Ethereum che richiede un proprio linguaggio di programmazione (chiamato Solidity), EOS permette l'utilizzo del linguaggio C++, già ampiamente accettato e consolidato, e conta di aggiungerne altri in futuro. Parte di queste informazioni sono schematizzate nella Tabella 4, estratta da uno studio comparativo tra i differenti driver della blockchain, "*Blockchain and Cryptocurrency: A comparative framework of the main Architectural Drivers*" [35].

Ultima differenziazione, non di certo per grado di importanza, è quella legata al pagamento (*gas price*). In Ethereum, infatti, gli utenti che utilizzano i contratti intelligenti devono pagare per effettuare delle transazioni, viceversa, in EOS, quest'obbligo spetta agli sviluppatori. [36]

Avendo come obiettivo quello di offrire un servizio scalabile, decentralizzato e sicuro, si è deciso di puntare sulla facilità e sull'economicità di utilizzo di EOS, con l'obiettivo di offrire un prodotto attrattivo sul mercato.

	ETHEREUM	EOS
Tasso di produzione di un blocco	10-19 sec	0.5 sec
Tempo di configurazione	1 min	1 sec
TPS	15	1000+
Smart Contract	Si	Si
DApp	Si	Si
Linguaggi	Solidity	Qualunque
Consensus Algorithm	PoW	DPoS
Vulnerabilità	Attacco al 51%	66% produttori di blocchi
Ledger	Blockchain	Blockchain

**Tabella 4:** Valutazione qualitativa delle alternative blockchain più diffuse in base alle caratteristiche tecnologiche [35]

La Tabella 5, tratta dallo studio “*Blockchain and Cryptocurrency: A comparative framework of the main Architectural Drivers*” [35], conferma quanto detto fino ad ora. Oltre ad avere il miglior punteggio totale rispetto a tutte le altre soluzioni blockchain disponibili, EOS si classifica al primo posto anche in termini di costi, performance, funzionalità e consistenza (funzione del tempo di conferma, vale a dire numero di blocchi dopo i quali si può considerare una transazione saldamente collegata alla catena).

	BTC	ETH	XRP	BCH	EOS	XLM	LTC	ADA	USDT	MIOTA	TRX
<b>Popolarità</b>	1	2	3	4	5	6	7	8	9	10	11
<b>Costo</b>	1.33	2	4.66	1.66	5	4.66	2.66	4.33	5	5	5
<b>Consistenza</b>	1.33	2.33	4.33	1.33	5	4	2	3.66	1	4.66	4
<b>Funzionalità</b>	2	5	1.33	2	5	1.33	2	4.33	2	3.66	5
<b>Performace</b>	1.33	1.66	4.33	2	4.66	4	2.33	3	1	5	4.66
<b>Sicurezza</b>	4	4	2.33	4	3.33	4	4	4	3.33	3.66	3.33
<b>Decentralizzazione</b>	5	3.33	1	4.33	2.66	2.33	3.66	3.33	1.33	2.33	3.33
<b>Total</b>	14.99	18.32	17.98	15.32	25.65	20.32	16.65	22.65	13.66	24.31	25.32

**Tabella 5:** Valutazione quantitativa delle soluzioni blockchain in base al feedback degli esperti [35]

## 5. Sviluppo dell'Applicazione

### 5.1 Architettura del sistema

Questo paragrafo ha l'obiettivo di mostrare l'architettura del sistema completa di tutti i suoi componenti (Figura 25).

Come si evince dalla Figura, la DApp (Decentralized Application) è costituita da due macro blocchi: Front-end e Back-end.

Il Front-end, la parte del programma visibile all'utente con cui quest'ultimo può direttamente interagire, è responsabile dell'acquisizione dei dati in ingresso ed è stata sviluppata mediante **React**, una libreria Javascript opensource che consente la creazione di interfacce grafiche UI (User Interface) interattive e dinamiche in modo semplice.

Il back-end, invece, è la parte del sistema che elabora i dati generati dal Front-end. Nella struttura in esame il back-end è stato organizzato in due sotto blocchi (vedi Figura 25). Tale configurazione deriva dal fatto che si è deciso di salvare i dati rilevanti per la concretizzazione delle trattative clienti-fornitori "on-chain", mentre tutte le informazioni non essenziali sono state memorizzate su un database "off-chain" (Mongodb).

Per quanto riguarda il salvataggio "on-chain", il Front-end comunica con la Blockchain mediante **eosjs**, una libreria Javascript che consente la lettura ed il salvataggio dei dati sulla **blockchain di EOS**, in cui opera lo **smart contract**.

Inoltre, il Front-end interagisce indirettamente con il database 'off-chain' Mongodb mediante il modulo di intermediazione REST-API module (vedi Figura 25). In particolar modo la comunicazione tra il front-end e il database Mongodb si struttura nei seguenti passaggi: (i) il front-end, tramite la libreria Javascript **Axios**, inoltra al framework **Express**, contenuto in REST-API module, le richieste degli utenti esterni; (ii) per il soddisfacimento di tali richieste, il framework **Mongoose** accede in lettura e scrittura ai dati contenuti nel database **Mongodb**.

Quindi, in sintesi, l'implementazione della struttura sopra descritta si compone di 3 macro blocchi:

1. Scrittura di uno **Smart Contract** in C++ che regoli gli accordi tra clienti e fornitori
2. Sviluppo di una **Applicazione web** che interagisca con l'utente finale
3. Creazione di un **Server** e relativo collegamento al database **Mongodb** per il salvataggio dei dati off-chain.

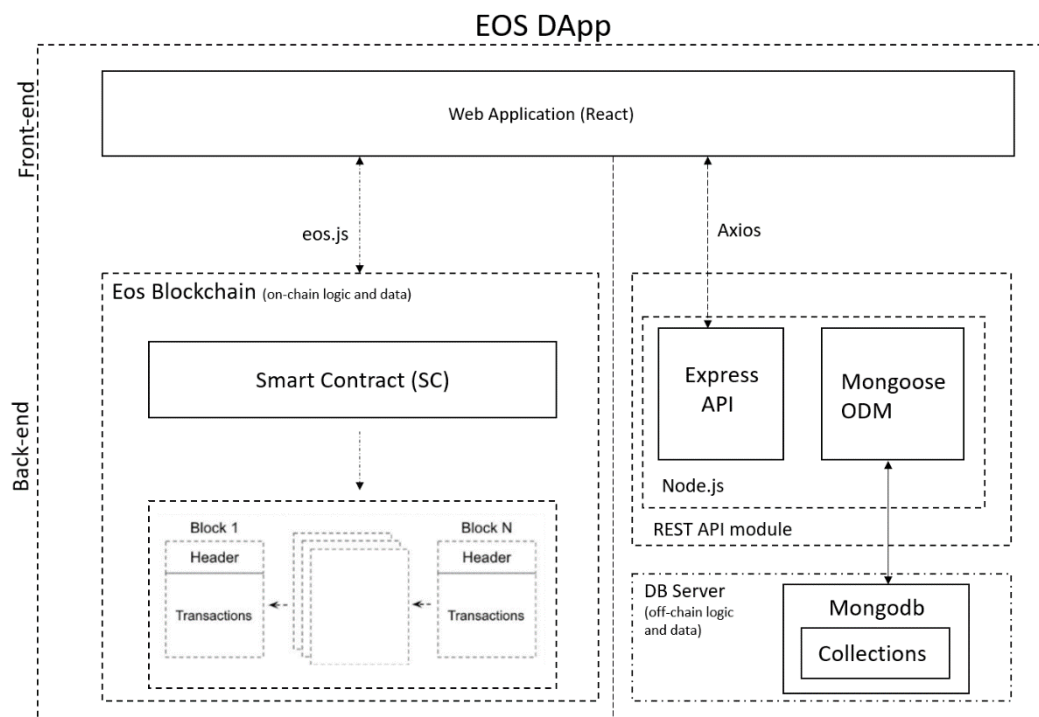


Figura 25: Architettura del sistema

## 5.2 Diagrammi UML

È di comune pratica in fase di progettazione, prima di sviluppare il codice vero e proprio, modellizzare l'intero processo (inclusi gli attori coinvolti e le principali funzioni) mediante appositi diagrammi, noti come diagrammi UML (Unified Modeling Language), al fine di rappresentare e documentare il prodotto finale (e.g. software, hardware).

L'obiettivo di ogni diagramma UML è quindi quello di fornire a sviluppatori e clienti una panoramica del prodotto finale da diverse prospettive e a vari gradi di astrazione, rendendo la grande mole di informazioni di immediata comprensione.

Fatta questa premessa, in questo capitolo verranno illustrati e descritti i diagrammi UML elaborati in questo lavoro di tesi al fine di chiarire: gli attori coinvolti unitamente alle funzioni che quest'ultimi sono autorizzati a fare (*use case diagram*, par. 5.2.1), i principali processi (*activity diagram*, par. 5.2.2) e in ultimo, le entità (*class diagram*, par. 5.2.3).

### 5.2.1 Use Case Diagram

Il primo step per definire la struttura passa dai diagrammi dei casi d'uso in cui viene fornita una panoramica della struttura di base e le principali funzioni (*casi d'uso*) che gli utenti possono svolgere.

Le unità fondamentali di un diagramma dei casi d'uso sono:



**Caso d'uso:** graficamente è costituito da un'ellisse contenente il nome del caso d'uso, rappresenta una funzione o servizio offerto dal sistema a uno o più attori. La funzione deve essere completa e significativa dal punto di vista degli attori che vi partecipano.



**Rettangolo vuoto:** definisce i contorni del sistema e mantiene separato tutto ciò che non ha un legame con il servizio



**Attori:** utenti che interagiscono direttamente con il sistema

I diagrammi dei casi d'uso varieranno in funzione della tipologia di attore. Poiché nel servizio qui considerato le tipologie di utente sono tre (cliente, fornitore e avvocato) di seguito verranno descritte le specifiche dei tre diagrammi-utente (Figure 26, 27, 28). Si noti come i differenti colori si riferiscano alle azioni che interagiscono con la blockchain (arancio), con database 'off-chain' (grigio) o con nessuno dei due (giallo).

- **Utente Cliente:** è il consumatore finale del servizio, una delle due parti contraenti l'accordo. È l'utente che desidera un determinato bene o servizio e utilizza la nostra piattaforma per gestire tutte le fasi della propria ricerca. Le funzioni che può effettuare l'acquirente (casi d'uso), mostrate in Figura 26, sono:
  1. Accesso/Uscita dal sistema.
  2. Sottoscrivere una nuova richiesta attraverso tre sotto-azioni: (i) descrizione del bene o servizio che desidera ricevere, (ii) prezzo disposto a spendere, (iii) data limite per la consegna del bene o servizio richiesto.
  3. Consultare l'offerta disponibile sulla piattaforma relativamente al bene/servizio desiderato.
  4. Contrattare con il fornitore. Questa funzione permette due opzioni: il cliente può accettare i termini contrattuali stabiliti dalla controparte oppure rifiutare e controproporre dei nuovi termini contrattuali. Quest'ultima opzione richiede l'inserimento dell'oggetto del contratto, del prezzo e della scadenza, dati ufficiali che saranno salvati sulla blockchain nel momento in cui si raggiunge l'accordo tra le due parti.

5. Confermare che il bene o servizio è conforme alle specifiche.
6. Aprire una nuova disputa inserendo la descrizione della motivazione del reclamo.
7. Osservare lo storico, ovvero tutti gli accordi conclusi in passato.

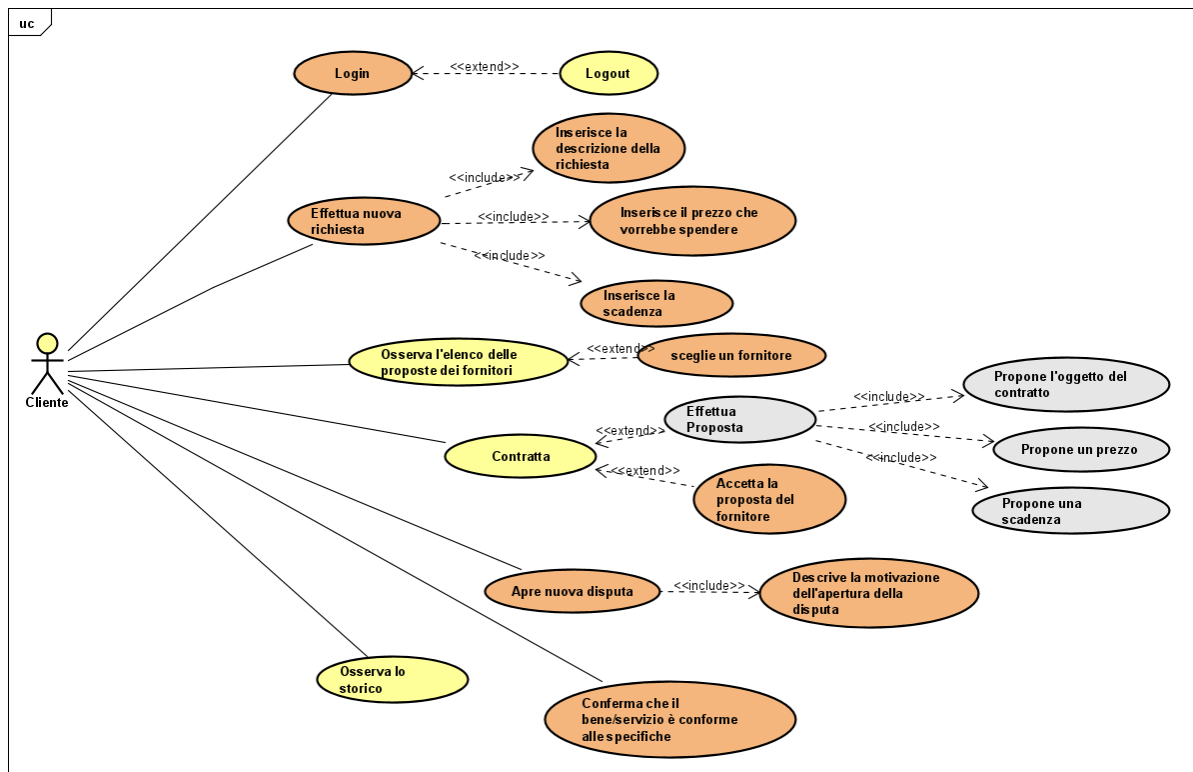


Figura 26: Diagramma dei casi d'uso relativo al cliente

- **Utente fornitore:** è il fornitore del bene o servizio a cui sono consentite le seguenti funzioni sulla piattaforma (Figura 27):
  1. Accesso/Uscita dal sistema.
  2. Consultare le richieste pubblicate dai clienti ed eventualmente, se di propria competenza, proporsi.
  3. Contrattare con il cliente. Le opzioni di tale azione sono le medesime di quelle descritte nell'azione dell'user cliente.
  4. Confermare la fine del lavoro. Nel momento in cui il fornitore spedisce il bene o termina un servizio per il cliente, segnala la conclusione del lavoro.
  5. Osservare lo storico, ovvero tutti gli accordi conclusi in passato.



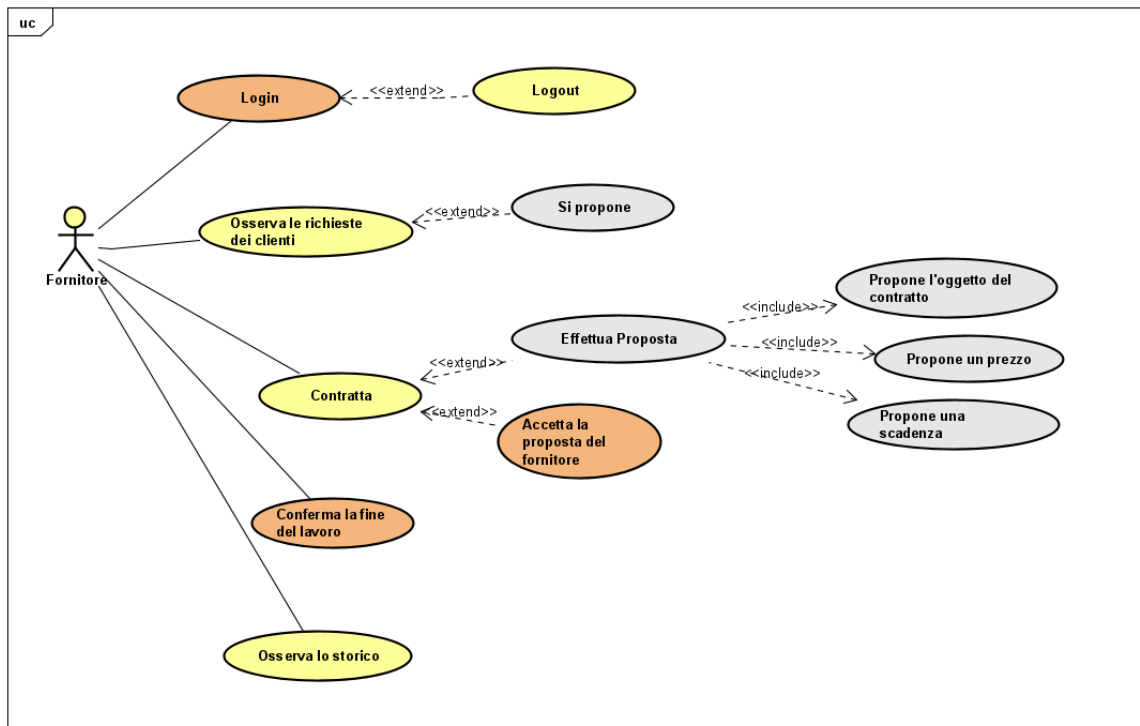


Figura 27: Diagramma dei casi d'uso relativo al fornitore

**Utente avvocato:** Questa categoria di attori viene inserita per risolvere un'ulteriore (eventuale) problematica legata agli elevati costi di gestione in caso di disputa. Questi soggetti sono chiamati ad esercitare la propria professione sulla piattaforma per garantire ai clienti la possibilità di avere "giustizia" nel caso in cui la controparte non dovesse fornire il bene o servizio conforme alle specifiche. Il tutto a un prezzo ragionevole e in maniera rapida.

Per questa categoria di attori il sistema offre un numero limitato di interazioni, come si può osservare dalla Figura 28, poiché il loro unico compito all'interno del servizio è quello di sentenziare.

Infatti, gli avvocati possono effettuare solo le tre azioni elencate di seguito:

1. Accesso/uscita dal sistema.
2. Sottomettere la sentenza. Dopo aver analizzato l'intero materiale relativo all'accordo, compreso di descrizione sulla motivazione dell'apertura della disputa, l'avvocato può svolgere il proprio ruolo prendendo la decisione di sentenziare a favore del fornitore oppure a favore del cliente.

- Controllare lo storico, ovvero tutte dispute per le quali è stato chiamato a sottomettere un giudizio

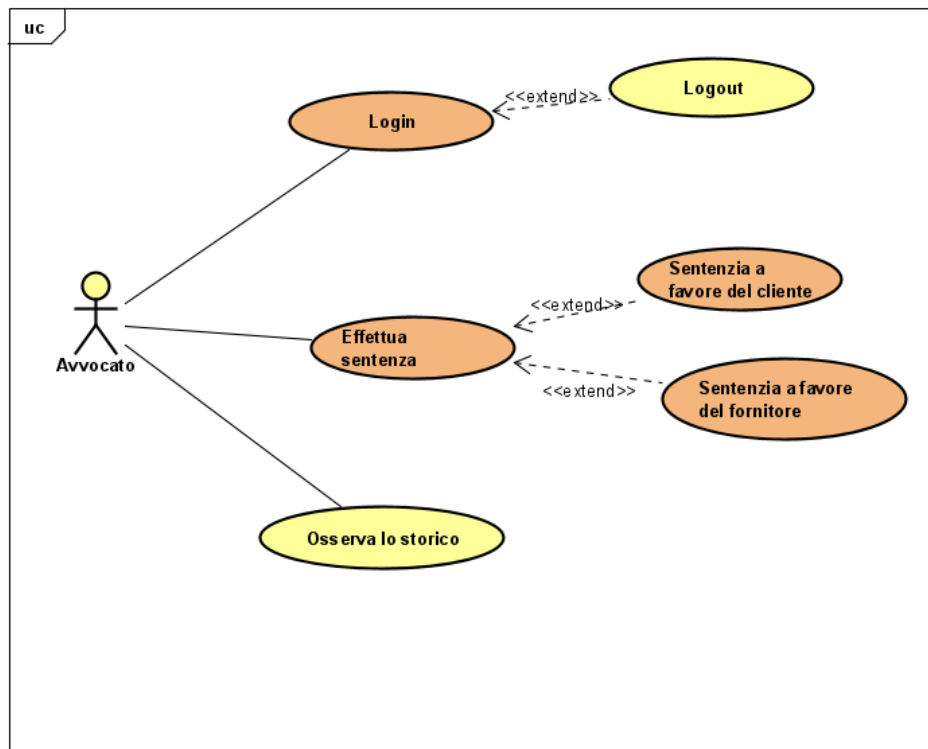

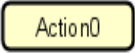




Figura 28: Diagramma dei casi d'uso relativo alla figura dell'avvocato

### 5.2.2 Activity Diagram

Un altro modello importante, facente parte degli UML, è quello relativo ai diagrammi delle attività.

Questi vengono usati per descrivere l'aspetto dinamico del sistema e i passaggi coinvolti nell'esecuzione di un caso d'uso. In pratica, si schematizzano i flussi di lavoro e le sequenze delle diverse azioni possibili. Le unità base che costituiscono i diagrammi (vedi Figure 29, 30, 31,32) delle attività sono:

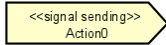
-  **Stato iniziale:** indica l'inizio di ogni attività
-  **Azione:** singola attività all'interno del diagramma
-  **Flussi di azione:** rappresentano la transazione da uno stato ad un altro
-  **Nodo decisionale:** si utilizza nel momento in cui il proseguimento del flusso di lavoro è condizionato dalla scelta dell'utente



**Clessidra:** per indicare il tempo a disposizione

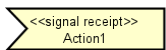


**Stato finale:** indica il termine di un particolare processo o attività



<<signal sending>>  
Action0

**Nodo di invio segnale:** invio messaggio



<<signal receipt>>  
Action1

**Nodo di ricezione segnale:** ricezione messaggio. L'esecuzione del processo è bloccata fino alla ricezione del messaggio.

Di seguito vengono descritte le attività che i vari utenti possono svolgere (vedi Figure 29, 30, 31, 32).

### **Attività di Registrazione e accesso:**

La prima attività è quella che l'utente deve compiere per loggarsi all'interno della piattaforma (Figura 29).

L'operazione è molto semplice, dopo l'avvio del sistema, l'utente (Cliente – Fornitore - Avvocato) si trova all'interno della pagina di Login in cui gli verrà richiesto l'inserimento delle credenziali (Username e Password) che gli consentiranno di effettuare l'accesso e visualizzare la propria Home page, nel caso si fosse precedentemente registrato al sistema. Contrariamente, se si tratta di un nuovo utente, quest'ultimo ha la possibilità di accedere al form di registrazione, compilare tutti i campi richiesti e sottomettere l'operazione. A questo punto sarà il sistema a memorizzare i dati del nuovo utente e lo rimanda alla Login Page.

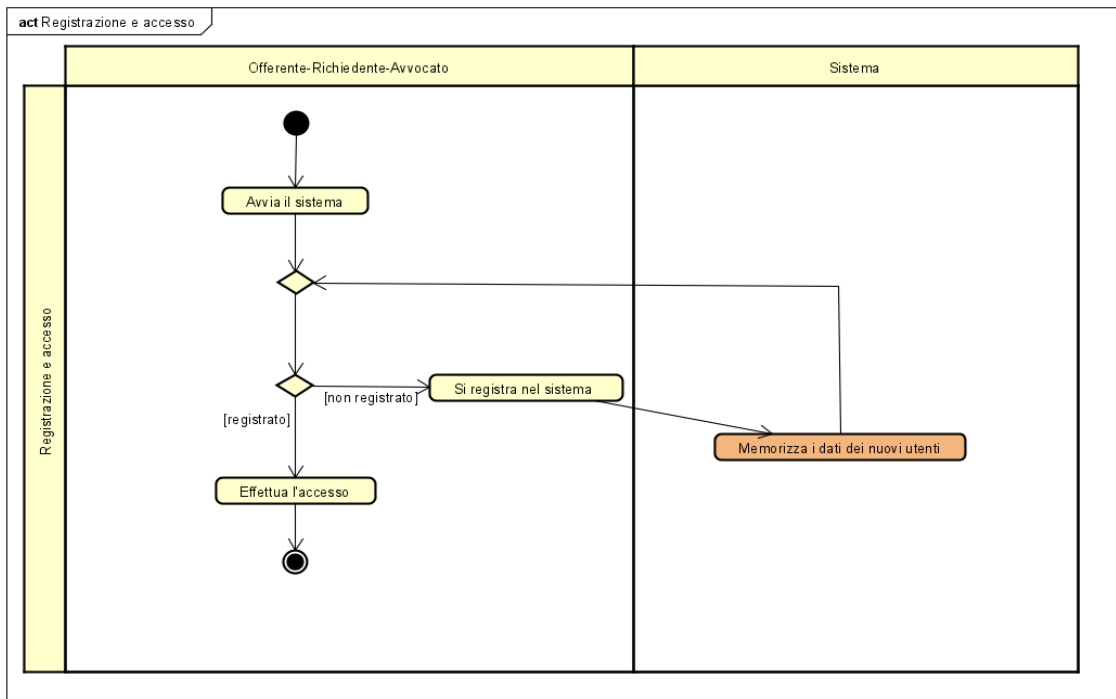


Figura 29: Diagramma delle attività di registrazione e accesso

### Attività di Contrattazione ed Esecuzione:

La modellizzazione di questo processo, schematizzato in Figura 30, vede in gioco due attori, ovvero il richiedente del servizio (cliente) e l'offerente, ovvero colui che si offre per la fornitura del bene/servizio richiesto (che verrà definito come fornitore per praticità).

Lo stato iniziale è localizzato sulla colonna del cliente (richiedente) che ha il compito di inizializzare il flusso del processo tramite la pubblicazione di una nuova richiesta, completa di tutte le informazioni necessarie.

A questo punto il sistema registra la nuova richiesta, salvando le informazioni sulla blockchain, e la rende visibile a tutti i possibili offerenti presenti sulla piattaforma. Questi ultimi, possono osservare la richiesta e, nel caso in cui ritenessero il task di propria competenza potrebbero proporsi al cliente. La "candidatura" dei fornitori viene così memorizzata dal sistema in un database esterno e mostrata sul portale del cliente che, a sua volta, ha la possibilità di osservare e analizzare le proposte dei diversi fornitori e scegliere quella che ritiene migliore. Il sistema registra il fornitore selezionato (sulla blockchain) e lo avvisa dell'avvenuto incarico. A questo punto, spetta al cliente il compito di inizializzare la fase di contrattazione proponendo un oggetto del contratto, un prezzo e una scadenza che può essere accettata oppure rifiutata dal fornitore. Nel caso di rigetto, l'offerente proseguirà con la sottomissione di una controproposta. Si crea così un ciclo di proposte, di volta in volta salvate su un database esterno, tra cliente e fornitore che terminerà nel momento in cui una delle due parti dovesse accettare la proposta della

controparte. Solo in questo caso il sistema preleverà e bloccherà dal portafoglio del cliente il compenso pattuito, congelerà un fondo prestabilito per la gestione dispute (denominato stake) dai portafogli delle due parti e registrerà i dati certi dell'accordo e l'inizio dei lavori sulla blockchain. Giunti a questo punto, è noterà la presenza di una clessidra, il cui scopo è quello di controllare il tempo e segnalare il momento in cui avviene la scadenza, così da permettere al fornitore di confermare la conclusione del proprio "lavoro".

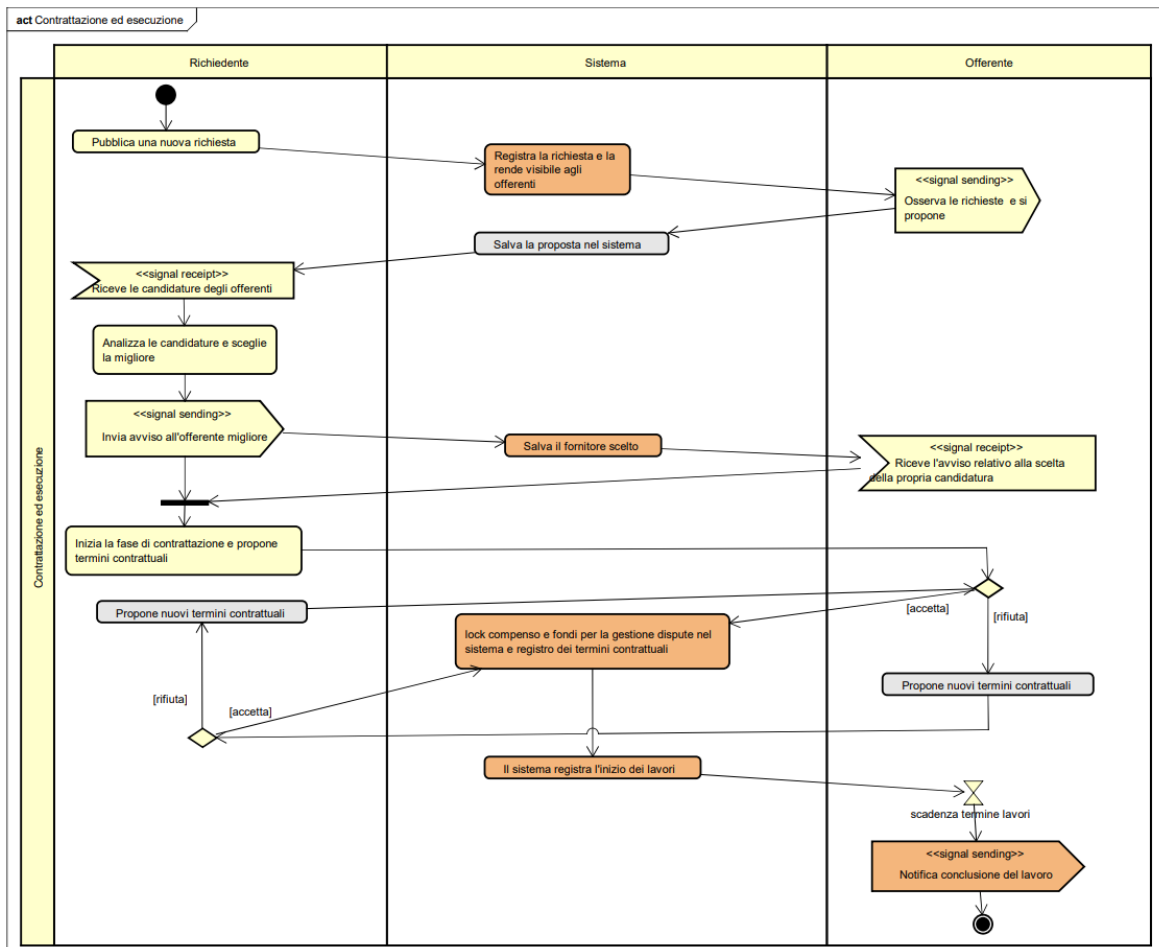


Figura 30: Diagramma della attività relative alla fase di contrattazione ed esecuzione

### Attività di Gestione Dispute:

Si passa ora alla descrizione dell'attività relativa alla gestione dispute, a cui partecipa la figura dell'avvocato. Tale attività è da pensarsi come la fase eventualmente successiva a quella precedentemente illustrata.

Infatti, nel momento in cui il fornitore del servizio conferma la fine dei lavori, il cliente riceve l'avviso di avvenuta conclusione e viene messo di fronte ad una scelta, ovvero confermare che il servizio o bene ricevuto è conforme alle specifiche, oppure avviare una procedura di reclamo. In quest'ultimo caso il sistema automaticamente sorteggia tre avvocati presenti nella piattaforma, attraverso un meccanismo di probabilità pesate in

base al rating, e rende loro disponibile il materiale relativo all'accordo tra i due contendenti. Gli avvocati visualizzano il nuovo incarico sul proprio portale e, dopo aver analizzato tutte le informazioni, sottometteranno una sentenza a favore del cliente o a favore del fornitore. A seguire, il sistema registrerà i voti dei tre avvocati e decreterà la sentenza finale in base alla maggioranza dei voti. Solo a seguire, il sistema dividerebbe e trasferirebbe il fondo precedentemente bloccato per la gestione dispute della parte perdente nei portafogli degli avvocati che hanno votato in maggioranza, aumentando così il loro rating di un'unità.

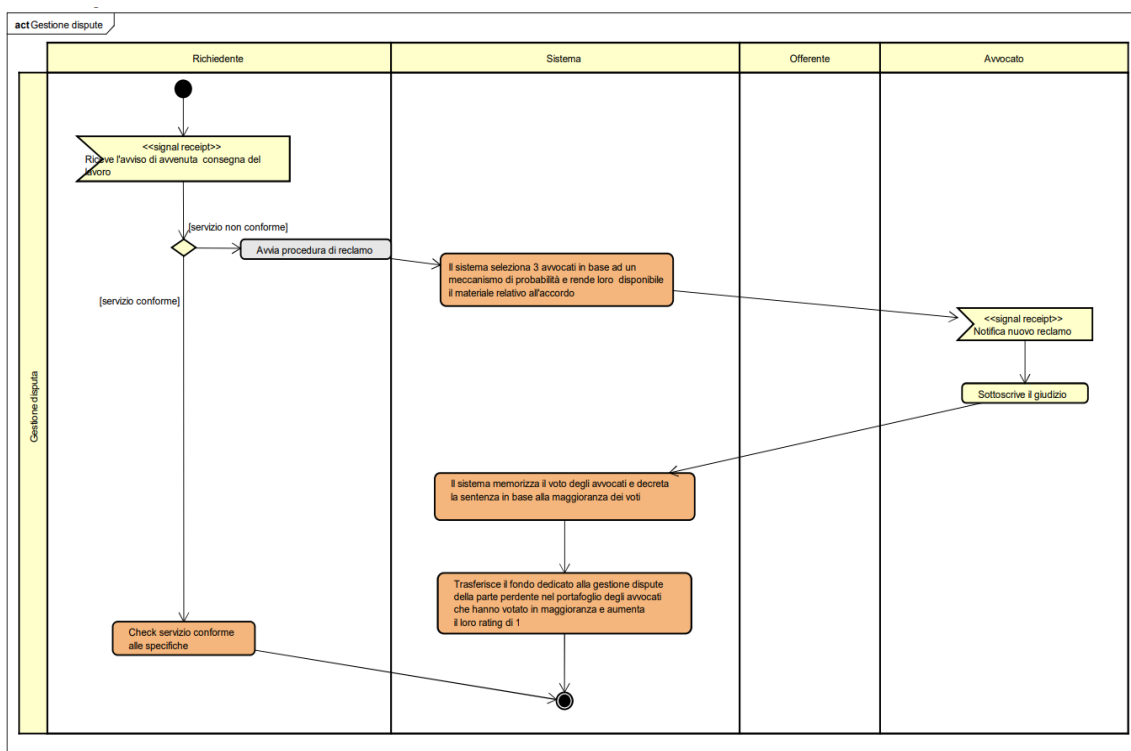


Figura 31: Diagramma delle attività relativo alla fase di gestione dispute

### Attività di Chiusura dell'accordo:

Tale attività viene svolta interamente dal sistema e risulta “invisibile” agli utenti esterni. La sequenza logica di questo processo, illustrata in Figura 32, è da ritenersi immediatamente successiva allo schema visto precedentemente.

Osservando la figura sottostante (Figura 32), si può notare come il flusso sia originato da un nodo decisionale condizionato dalla scelta del cliente. Le azioni messe in atto dal sistema dipendono infatti dalla volontà del cliente di confermare il servizio come conforme alle specifiche o di aprire una nuova disputa. Nel caso in cui l'acquirente dovesse confermare che il lavoro sia stato svolto secondo le condizioni pattuite, allora il sistema aumenterebbe il rating di entrambe le parti di un'unità, rimborserebbe loro la quota (stake) per la gestione dispute, precedentemente bloccata, e trasferirebbe istantaneamente il compenso al fornitore. Differentemente, avviene invece nel caso in cui l'acquirente dovesse aprire una disputa. In tale circostanza, osservando la Figura 32, si incontra un ulteriore nodo condizionale in base al quale il sistema gestirebbe il versamento del compenso bloccato a seconda dell'esito della sentenza. Infatti, se il fornitore, sulla base del giudizio dei tre avvocati, dovesse risultare il "vincitore" della disputa allora il sistema gli rimborserebbe la quota (stake) precedentemente bloccata, aumenterebbe il suo rating di una unità e trasferirebbe nel suo portafoglio la somma pattuita in fase di contrattazione. Viceversa, se le sentenze dei tre avvocati dovessero far primeggiare il cliente allora il sistema dividerebbe equamente il compenso bloccato, aumenterebbe il rating del cliente di un'unità e gli rimborserebbe lo stake.

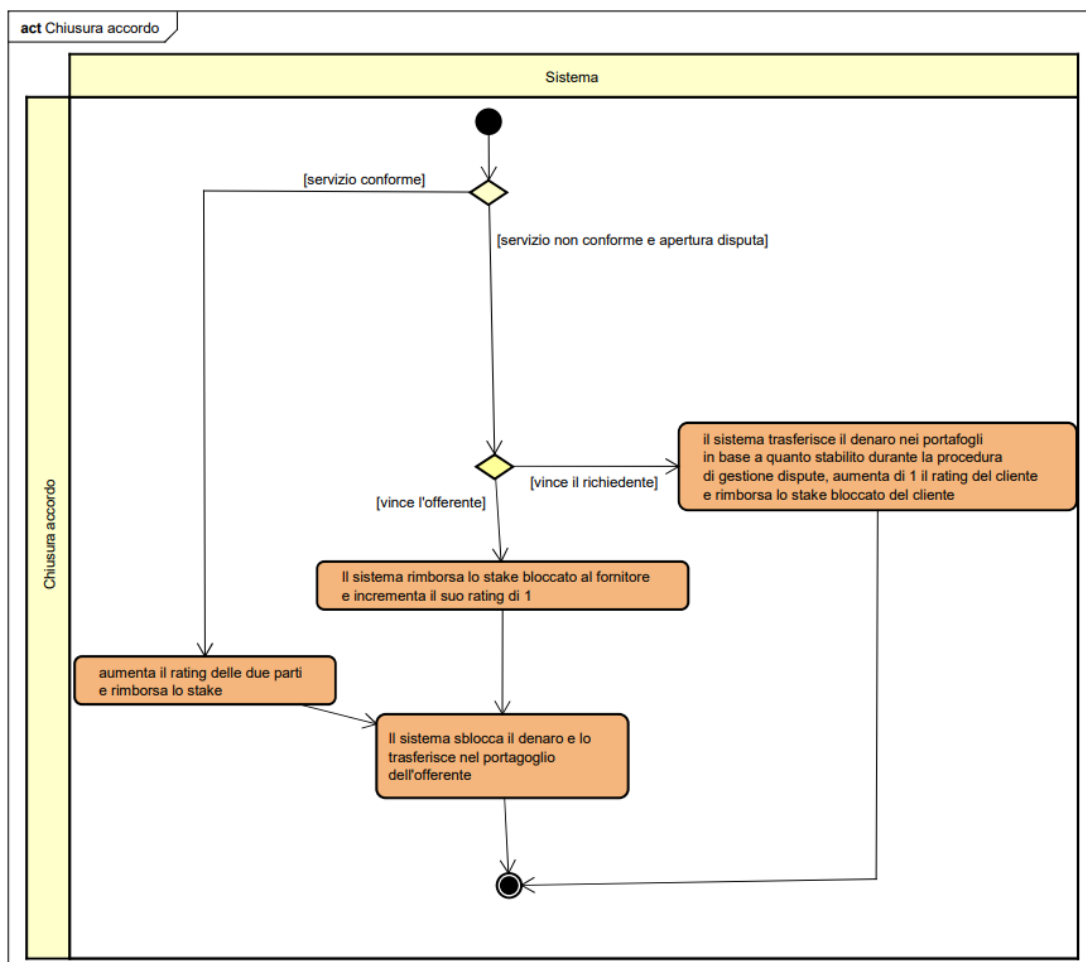
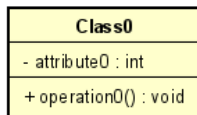


Figura 32: Diagramma delle attività relativo al processo di chiusura dell'accordo

### 5.2.3 Class Diagram

Il diagramma delle classi consente di descrivere le tipologie di entità presenti all'interno della soluzione proposta, con le loro caratteristiche e le eventuali relazioni con le altre entità esistenti.



Il simbolo grafico che rappresenta le classi UML è un rettangolo suddiviso in tre scomparti, rispettivamente dedicati al nome della classe, ai suoi attributi e alle operazioni. Due classi possono essere legate da relazioni che possiedono delle informazioni aggiuntive ovvero il legame che sussiste tra le due e la molteplicità.

In Figura 33 è rappresentato il diagramma delle classi relativo al servizio che si propone. È presente un'entità padre, vale a dire l'utente, completo di tutti i suoi attributi, da cui si generano tre entità figlie, ovvero il cliente, il fornitore e l'avvocato.

Ogni cliente può generare da zero a infinite richieste, può partecipare da zero a infinite contrattazioni e può aprire da zero a infinite dispute.

Ogni fornitore invece può partecipare da zero a infinite contrattazioni e può effettuare da zero a infinite proposte. Similmente avviene per l'avvocato che può sentenziare da zero a infinite dispute.

Ogni proposta è effettuata da un solo fornitore, è destinata ad una sola richiesta e viene scelta al massimo da un cliente.

Ciascuna disputa invece è aperta da un cliente, è relativa ad una richiesta ed è sentenziata da tre avvocati.

La classe relativa alla contrattazione ha un solo cliente e un solo fornitore, è legata ad una sola richiesta, però può subire da zero a infiniti aggiornamenti.

Infine, ogni richiesta è generata da un solo acquirente e può essere oggetto al massimo di una disputa, ma di diverse proposte (da zero a infinite).



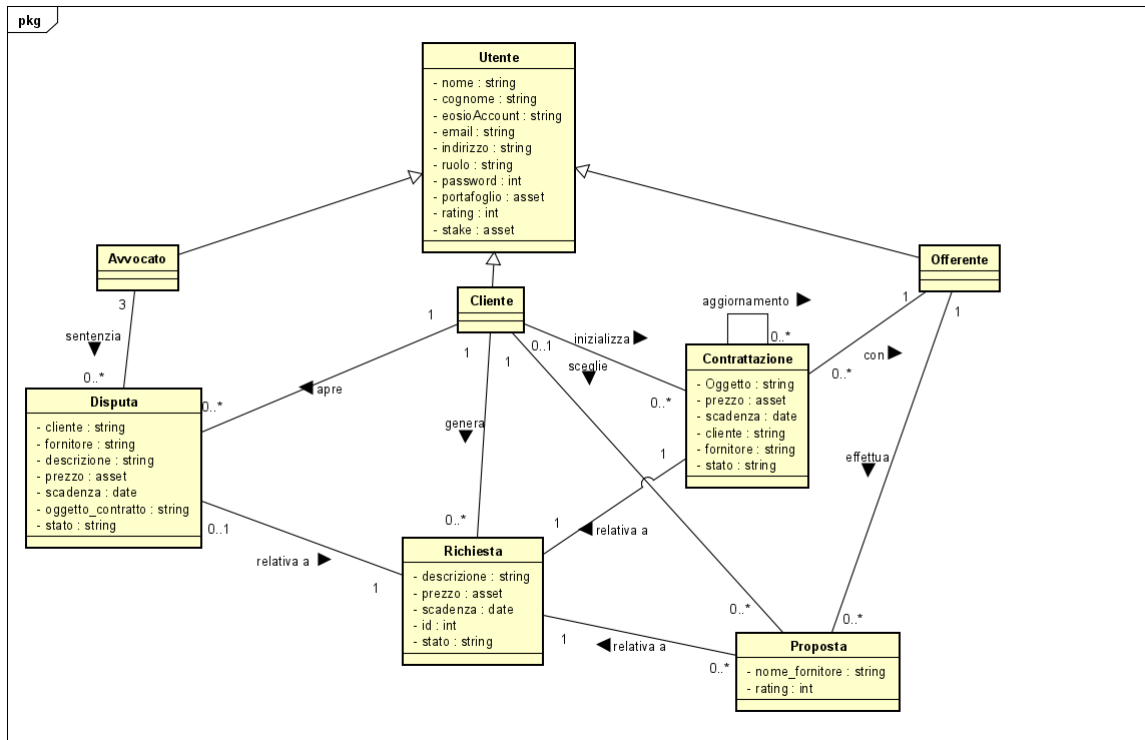


Figura 33: Diagramma delle classi

## 5.3 Smart Contract

### 5.3.1 Strumenti utilizzati

Per la fase di scrittura dello Smart Contract si è deciso di sfruttare un IDE grafico, denominato EOS Studio, lanciato per la prima volta nel febbraio 2019 e diventato poi lo strumento più popolare per EOSIO.

Tale IDE, integrando diversi strumenti necessari per EOSIO in un'applicazione unificata, mira a fornire uno strumento potente e facile da usare per lo sviluppo di DApp. Il tool utilizza un editor di codice C++ con integrazione della sintassi EOSIO. Per lo sviluppo del progetto in questione, è stata installata la versione Desktop autonoma e compatibile col sistema operativo di Windows.

#### Panoramica

Lo strumento utilizzato è dotato di un layout intuitivo e di facile utilizzo. L'interfaccia utente è divisa in quattro differenti sezioni:

- *Project Editor*: l'interfaccia principale in cui è possibile lavorare al progetto. Tale interfaccia comprende un terminale incorporato ed una toolbar con cui è possibile eseguire i comandi di "built", "deploy" e "test" del progetto con un semplice click.

- **Contract Inspector:** sezione che fornisce gli strumenti necessari per ispezionare ed eseguire il debug dei contratti intelligenti. Eos Studio Contract Inspector è costituito da due parti (vedi Figura 34):
  1. Il pannello di sinistra in cui vengono eseguite le azioni. Nell'immagine in basso (Figura 34) viene mostrata l'azione di login del nostro SC in cui l'utente dovrà inserire l'account, il "role", il "balance" e il "total stake" per poter memorizzare il proprio account nella tabella "users".
  2. Il pannello di destra con il quale poter interrogare i dati all'interno delle tabelle ("Data Table")

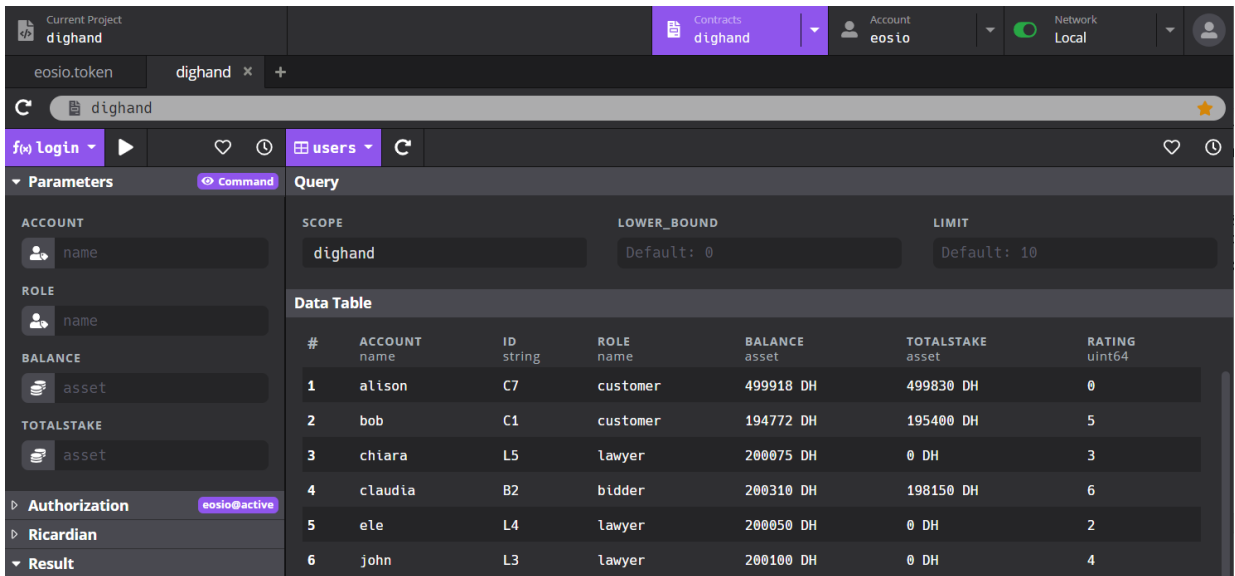


Figura 34: Contract Inspector di Eos Studio

- **Account Viewer:** sezione in cui è possibile creare, visualizzare gli account ed eseguire le operazioni relative agli stessi, il tutto in pochi e rapidi passaggi.
- **Network Manager:** interfaccia che permette di switchare da una rete locale ad altre reti di prova e mostrare le informazioni della rete selezionata (vedi Figura 35).

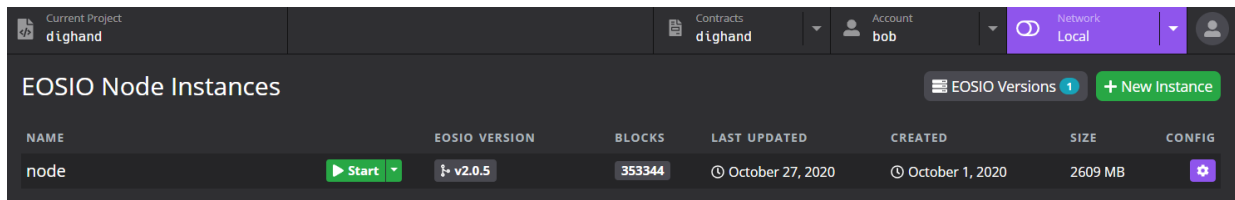


Figura 35: Network section

## Set up dell'ambiente di lavoro

Prima di programmare le logiche del processo è necessario configurare l'ambiente di lavoro. Lo sviluppo di una dApp EOSIO richiede:

- EOSIO, il principale software per lanciare la blockchain EOSIO-based, che a sua volta include:
  1. Nodeos: l'eseguibile principale per la produzione dei blocchi e fornisce endpoint Api.
  2. Cleos: strumento che viene utilizzato per interrogare la blockchain EOSIO
  3. Keosd: un portafoglio per gestire le coppie di chiavi e firmare le transazioni
- EOSIO.CDT (Contract Development Toolkit) che viene usato per compilare codici sorgenti C++ in WebAssembly

Grazie all'utilizzo di Docker sono stati installati ed utilizzati gli strumenti precedentemente descritti.

## Creazione di un nuovo progetto

Dopo aver completato le varie installazioni è stato possibile creare un nuovo progetto, semplicemente cliccando sul pulsante "Crea nuovo progetto". A seguire, premendo sul pulsante destinato alla costruzione del progetto, è stato eseguito EOSIO.CDT per compilare il contratto ed esportare due tipologie di file:

- Il file `wasm`: un file binario WebAssembly che viene eseguito sulla blockchain EOSIO
- Il file `abi`: un oggetto json che definisce le azioni del contratto e le tabelle di dati

### 5.3.2 Sviluppo Smart Contract

La scrittura del contratto intelligente su cui si fonda l'intero progetto vede l'utilizzo di due tipologie di file, ovvero l'Header file (`hpp`) che utilizza la sintassi `[[eosio::contract]]` e funge da definitore di tabelle e azioni, e il file sorgente (`cpp`), in cui si è implementata la logica di ogni azione.

#### Tabelle

Due tabelle sono state definite all'interno del file header, attraverso il codice `[eosio::table eosio::contract("project_name")]`, denominate rispettivamente **"users"** e **"handshake"**.

La prima (vedi Figura 36) è stata creata con l'intento di permettere all'utente di memorizzarsi sulla blockchain tramite l'azione di login, che richiede l'inserimento dei seguenti dati: l'account del fruitore del servizio (`name`), l'ID (`string`), il ruolo (`name`), il balance (`asset`), il proprio total stake (`asset`) e il rating (`uint64`).

#	ACCOUNT name	ID string	ROLE name	BALANCE asset	TOTALSTAKE asset	RATING uint64
1	alison	C7	customer	499918 DH	499830 DH	0
2	bob	C1	customer	194772 DH	195400 DH	5
3	chiara	L5	lawyer	200075 DH	0 DH	3
4	claudia	B2	bidder	200310 DH	198150 DH	6

Figura 36: Tabella “users” sulla blockchain

La seconda tabella, handshake table, nasce con l’obiettivo di salvare progressivamente on-chain i dati relativi agli accordi tra cliente e fornitore. Ogni riga corrisponde ad un accordo e possiede le seguenti informazioni (colonne della tabella):

- *id* (uint32 type): identificativo univoco dell’accordo
- *customer\_account* (name type): account della parte cliente dell’accordo
- *bidder\_account* (name type): account del fornitore del bene/servizio
- *request\_description* (string type): descrizione preliminare del prodotto/servizio richiesto
- *customer\_stake* (asset type): quantità di token che viene bloccata dal cliente per la gestione di un’eventuale disputa
- *bidder\_stake* (asset type): quantità di token che viene bloccata dal fornitore per la gestione di un’eventuale disputa
- *deadline* (time\_point type): termine entro il quale il servizio deve essere fornito
- *amount* (asset): prezzo del bene/servizio concordato dalle due parti
- *contract\_object* (string type): oggetto del contratto
- *handshake\_state* (string type): lo stato dell’accordo che varia da “richiesta” a “contrattazione” ad “esecuzione”, per poi divenire “finalizzato”, “disputa” ed infine “chiuso”.
- *lawyer\_account1* (name type): account del primo avvocato sorteggiato dal sistema in caso di disputa
- *lawyer\_account2* (name type): account del secondo avvocato sorteggiato dal sistema in caso di disputa
- *lawyer\_account3* (name type): account del terzo avvocato sorteggiato dal sistema in caso di disputa
- *lawyer\_vote1* (string type): sentenza del primo avvocato
- *lawyer\_vote2* (string type): sentenza del secondo avvocato
- *lawyer\_vote3* (string type): sentenza del terzo avvocato
- *hash\_description\_dispute* (string type): descrizione di un “hash” (informazione crittografata) legata alla motivazione dell’apertura del reclamo compilata dal cliente.

Questo hash viene generato al fine di poter controllare la veridicità di ciò che è salvato nel sistema. Infatti, qualora l'hash relativo alla disputa non dovesse corrispondere a quello salvato sulla blockchain ne risulterebbe un tentativo di manomissione del documento al fine di sabotare la sentenza finale.

Questi valori vengono gradualmente inseriti all'interno delle celle condizionatamente alle azioni intraprese lato cliente ed allo stato di avanzamento dell'accordo.

## Le Azioni

Le funzioni pubbliche che sono state maggiormente utilizzate all'interno del codice e che sono incorporate in EOSIO sono:

- “*Emplace function*”: aggiunge un nuovo oggetto (una riga) alla tabella (Figura 37).

```
template<typename Lambda>
const_iterator eosio::multi_index< TableName, T, Indices >::emplace(
    name payer,
    Lambda && constructor
)
```

Figura 37: Codice della funzione Emlace

- “*Modify function*”: modifica un oggetto esistente in una tabella (Figura 38).

```
template<typename Lambda>
void eosio::multi_index< TableName, T, Indices >::modify(
    const T & obj,
    name payer,
    Lambda && updater
)
```

Figura 38: Codice della funzione Modify

- “*Find function*”: cerca un oggetto esistente nella tabella in base alla chiave primaria passata come argomento (Figura 39).

```
const_iterator eosio::multi_index< TableName, T, Indices >::find(
    uint64_t primary
) const
```

Figura 39: Codice della funzione Find

- “*Lower\_bound function*”: cerca l'oggetto che possiede la più bassa chiave primaria, maggiore o uguale a quella passata come argomento (Figura 40).

```
const_iterator eosio::multi_index< TableName, T, Indices >::lower_bound(
    uint64_t primary
) const
```

Figura 40: Codice della funzione Lower\_bound

- “*Upper\_bound function*”: estrae l'oggetto che possiede la più alta chiave primaria, più elevata di quella passata come argomento (Figura 41).

```
const_iterator eosio::multi_index< TableName, T, Indices >::upper_bound(
    uint64_t primary
) const
```

Figura 41: Codice della funzione Upper\_bound

- “Require\_auth function”: controlla che l’account passato esista e gli attribuisce l’autorizzazione per effettuare l’azione in cui la funzione è inserita (Figura 42).

```
void eosio::require_auth(
    name n
)
```

Figura 42: Codice della funzione Require\_auth

All’interno del file sorgente, con circa 565 righe di codice, è stata sviluppata la logica di nove azioni messe a disposizione dei nostri utenti:

1. **login action**: aggiunge un nuovo utente alla tabella users.
2. **dighandshake action**: aggiunge un nuovo accordo alla tabella handshake, con le informazioni relative all’account del cliente, la descrizione della richiesta, il prezzo, la scadenza e lo stato di “richiesta”.
3. **choice\_bidder**: modifica il record aggiungendo l’account del fornitore e modificando lo stato da “richiesta” a “contrattazione”.
4. **agreement**: modifica il record aggiungendo i dati accettati dalle due parti durante la trattazione. Così, le informazioni sull’oggetto del contratto, sul prezzo e sulla scadenza definitivi vengono inseriti nel record in questione e viene modificato lo stato da “contrattazione” ad “esecuzione”. A seguire, il portafoglio del cliente verrà decrementato del valore del compenso concordato e verranno congelati i due stake per l’eventuale gestione disputa.
5. **job\_finalized**: modifica lo stato da “esecuzione” e “finito”.
6. **confirm**: modifica lo stato da “finito” a “chiuso”.
7. **claim**: azione corrispondente all’apertura di una disputa. In tale funzione si modifica lo stato del record in “disputa”, vengono sorteggiati i tre avvocati e inseriti nella tabella.
8. **description**: modifica il record memorizzando l’hash relativo alla descrizione della disputa.
9. **lawyersvote**: modifica il record inserendo la votazione dell’avvocato.

La tabella 6 indica quale stakeholder possiede l’autorizzazione a compiere una determinata azione.

Azione	Cliente	Fornitore	Avvocato
login	✓	✓	✓

dighandshake	✓	X	X
choice_bidder	✓	X	X
agreement	✓	✓	X
job_finalized	X	✓	X
confirm	✓	X	X
claim	✓	X	X
description	✓	X	X
lawyersvote	X	X	✓

**Tabella 6:** autorizzazioni per ogni action

La tabella 7 mostra come vengono memorizzate le informazioni relative ad un accordo al variare dello stato dell'accordo stesso. Infatti, quando lo stato dell'accordo corrisponde a "request", all'interno del record sono presenti solo i valori dell'id, dell'account del cliente, della descrizione della richiesta, dello stake di entrambe le parti, della scadenza e del prezzo. Tutti i restanti campi corrispondono a "null". Nel momento in cui il cliente sceglie il fornitore, lo stato passa a "bargaining" e viene inserito l'account relativo all'offerente (bidder account). A seguire, dopo il concretizzarsi della contrattazione, lo stato diventa "execution" e l'oggetto del contratto viene memorizzato. I campi riempiti rimangono invariati fino a quando il fornitore conferma la fine del lavoro e lo stato diventa "finalized". Nel caso in cui lo stato passasse a "dispute" tutte le celle della colonna in questione verrebbero compilate. Differente è il caso in cui lo stato prende il valore di "closed", ovvero quando termina il flusso logico dell'accordo, poiché i valori relativi alla disputa dipendono dal verificarsi o meno di tale evento.

	request	bargaining	execution	finalized	dispute	closed
ID	Si	Si	Si	Si	Si	Si
Customer account	Si	Si	Si	Si	Si	Si
Bidder account	No	Si	Si	Si	Si	Si
Request description	Si	Si	Si	Si	Si	Si
Customer HSK Stake	Si	Si	Si	Si	Si	Si
Bidder HSK Stake	Si	Si	Si	Si	Si	Si
Deadline	Si	Si	Si	Si	Si	Si

Amount	Si	Si	Si	Si	Si	Si
Contract Object	No	No	Si	Si	Si	Si
Handshake state	Si= 0	Si= 1	Si= 2	Si= 3	Si= 4	Si= 5
Lawyer account 1	No	No	No	No	Si	Si/No
Lawyer account 2	No	No	No	No	Si	Si/No
Lawyer account 3	No	No	No	No	Si	Si/No
Lawyer vote 1	No	No	No	No	Si	Si/No
Lawyer vote 2	No	No	No	No	Si	Si/No
Lawyer vote 3	No	No	No	No	Si	Si/No
Hash_description_dispute	No	No	No	No	Si	Si/No

**Tabella 7:** legame tra lo stato dell'accordo e le altre informazioni dell'accordo stesso

## 5.4 Studio dei Mockup

Il mockup non è altro che un modello, una rappresentazione fedele di come sarà il prodotto da un punto di vista grafico, con dettagli come colori, font o immagini. Sono però elementi statici, non interattivi e nati con l'idea di mostrare lo stile del prodotto e quelle che saranno le sue funzioni.

Consapevoli dell'importanza di questo modello si è deciso di dedicare un'intera settimana per lo studio dei mockup della soluzione, in modo tale da poter esplorare le decisioni visive prima di sviluppare il codice front-end.

### 5.4.1 Strumento utilizzato e analisi effettuate

Per lo sviluppo dei mockup si è deciso di utilizzare Adobe XD, un potente software di progettazione dell'esperienza utente per web app e mobile app, prodotto e distribuito da Adobe e disponibile su Windows. La scelta di utilizzare tale strumento è derivata dalla possibilità di effettuare delle microinterazioni tra i diversi mockup per rappresentare e visualizzare meglio il movimento del contenuto tra le diverse "tavole da lavoro".

Durante la settimana dedicata a tale studio si è diviso il lavoro principalmente in due attività:



- L'organizzazione dei dettagli: quante e quali schermate creare, il contenuto all'interno di ogni mockup e l'organizzazione dei dati e delle informazioni. Si è deciso inoltre di utilizzare il colore blu che suscita sensazioni di affidabilità, confort e consistenza agli occhi del fruitore del servizio e di strutturare il contenuto in modo semplice, ovvero con poche informazioni, ben distribuite e ben rappresentate. I dati e le diverse informazioni sono stati inseriti in tabelle chiare ed intuitive.
- L'implementazione del design: grazie alla funzionalità microinterattiva disponibile sul software utilizzato, si è testato ed approvato il design prima di procedere con la fase di sviluppo del codice vera e propria.

In questo modo si è giunti alla definizione del concept grafico del servizio oggetto di tesi.

#### 5.4.2 Raffigurazione Mockup

I mockup, frutto di un'intera settimana di lavoro in azienda, sono raffigurati di seguito (Figure 43, 44, 45, 46 e 47). Lo sfondo del mockup relativo alla pagina di Login, in Figura 43, è stato preso dall'articolo "*Match me if you can: Cryptographic Breakthrough helps spies to shake hands*" [37].

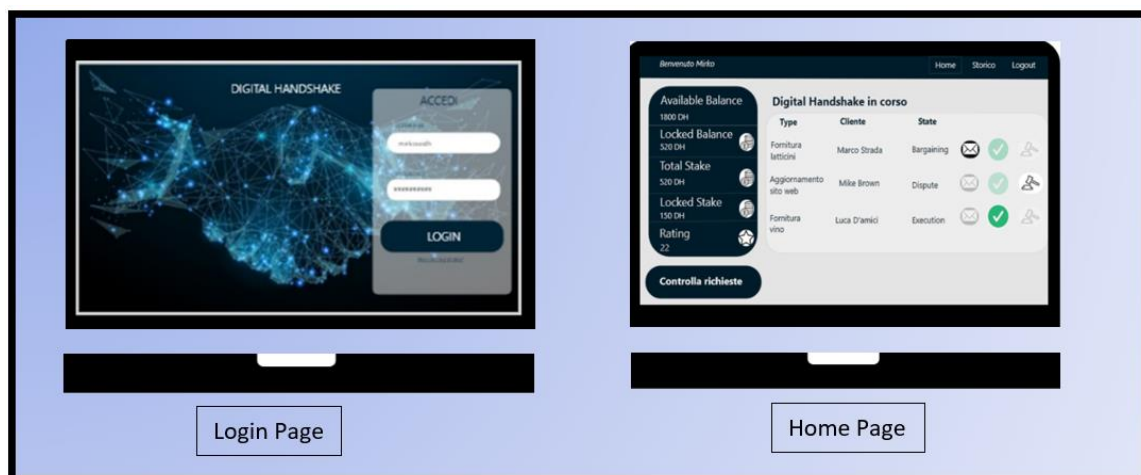


Figura 43: Mockup Login e Home Page

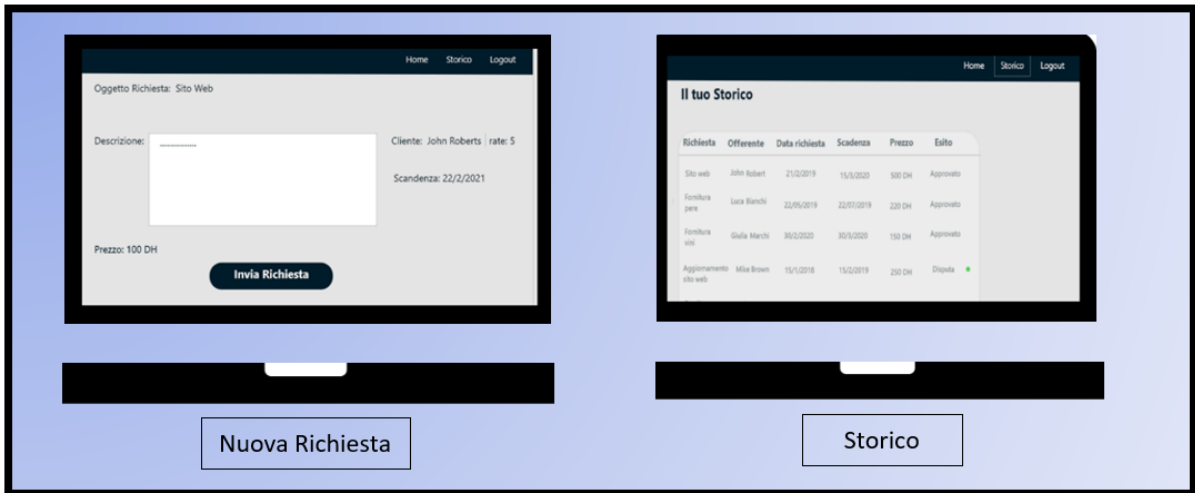


Figura 44: Mockup Nuova Richiesta e Storico

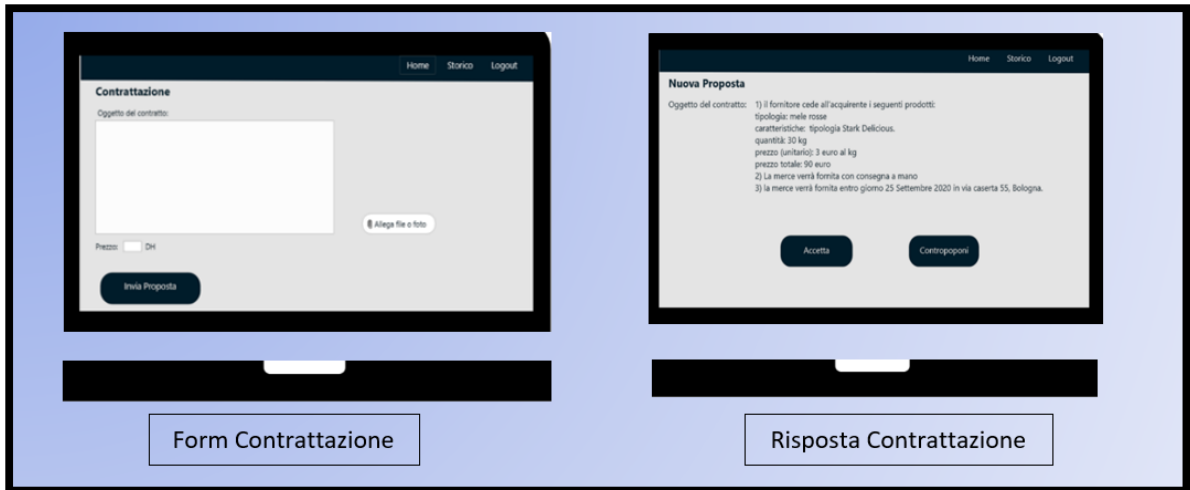


Figura 45: Mockup Form Contrattazione e Risposta Contrattazione



Figura 46: Mockup Proposte Fornitori e Richieste Clienti



Figura 47: Mockup Sentenza Avvocato

## 5.5 Sviluppo Front-end

Nel presente capitolo sono descritti gli strumenti utilizzati durante lo sviluppo e le scelte progettuali messe in opera nella versione finale dell'applicazione.

### 5.5.1 Tecnologie e Strumenti

Nella presente sezione vengono descritti gli strumenti di sviluppo utilizzati durante la fase di sviluppo front-end.

#### Ambiente di sviluppo

Lo sviluppo è avvenuto interamente sul sistema operativo Windows 10 pro e la versione di React.js adoperata è la 16.13.0.

#### Linguaggio utilizzato e principali dipendenze

React.js è una libreria del linguaggio JavaScript con varie funzionalità. Al fine di realizzare l'applicazione si è fatto uso dell'ampia collezione dei pacchetti installabili npm, che hanno semplificato lo sviluppo. NPM è un package manager (Node.js Package Manager), ovvero uno strumento che permette l'installazione di moduli sviluppati dalla comunità, consentendo di includere, rimuovere e aggiornare le librerie all'interno del progetto.

#### Redux.js

Si tratta di una libreria di state management creata appositamente per React.js. Permette l'utilizzo di uno Store globale che contiene idealmente l'intero stato dell'applicazione. Tramite l'utilizzo dei collegamenti con React, offerti dal pacchetto react-redux, è stato

possibile utilizzare lo Store e collegarlo ai componenti per gestire la visibilità di alcuni pulsanti.

### **React-router**

Questo pacchetto, di fondamentale importanza, permette di effettuare routing client side, caricando i componenti in base all'indirizzo della pagina.

### **React-bootstrap**

Dispone di una collezione di componenti per React.js che utilizzano le classi e lo stile di Bootstrap. Quest'ultimo non è altro che una libreria di componenti front-end per costruire pagine web responsive e accessibili da tutti.

### **React-axios**

Ha lo scopo di consentire il rendering delle richieste asincrone.

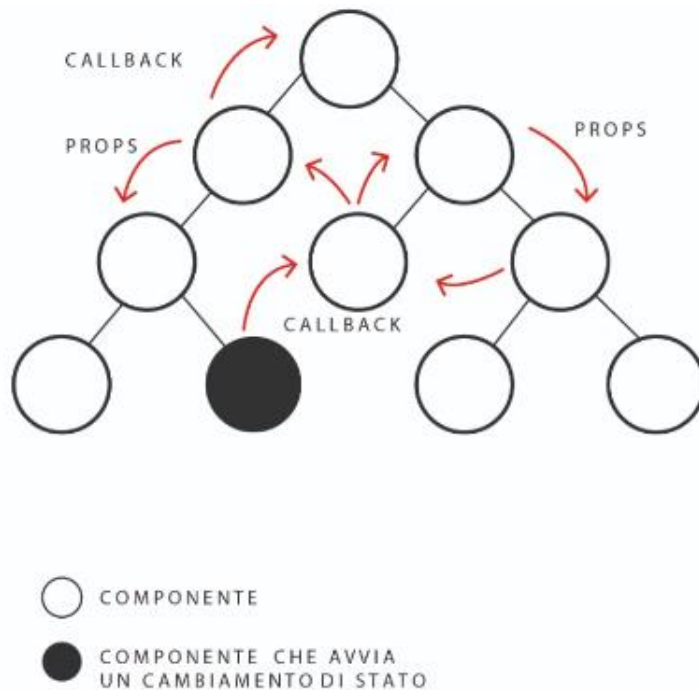
### **React-eosjs**

Consente l'integrazione delle API Javascript con la blockchain EOSIO. Questa parte sarà approfondita successivamente (Paragrafo 5.5.4).

## **5.5.2 Cenni di funzionamento di Redux e React.js**

**React.js** si basa sul concetto di componenti (o component) che rappresentano i tasselli indipendenti e riutilizzabili con i quali viene costruita una pagina web. Essi sono classi o funzioni Javascript che controllano l'aspetto e definiscono il comportamento di un certo frammento di codice HTML.

Ogni componente può possedere uno o più stati interni che vengono inizializzati ad ogni caricamento e spesso devono essere trasmessi a componenti figli, ovvero componenti annidati all'interno di un altro componente. Tale trasmissione avviene attraverso il passaggio di props (proprietà) lungo la gerarchia di componenti che formano un albero gerarchico e la direzione seguita va da padre in figlio, vedi Figura 48. È anche possibile passare ai componenti figli delle funzioni di callback, vale a dire a dei metodi che possono essere richiamati dal componente padre per chiedere la trasmissione di un'informazione dal componente figlio. [38]



**Figura 48:** Rappresentazione del passaggio di props e invocazione di callback nell'albero di componenti di React.js [38]

## Redux

Se diversi componenti non relazionati tra loro avessero bisogno di accedere alle stesse informazioni o volessero modificarle, il processo diventerebbe complesso, perché ciò richiederebbe un numero elevato di funzioni di callback, difficili da gestire. Al fine di eseguire un approccio più ordinato si è soliti utilizzare la libreria di Redux che offre un contenitore globale dello stato per applicazioni Javascript. Redux mette a disposizione uno *Store* centrale in cui è mantenuto lo stato dell'applicazione in forma di albero. Ogni componente è connesso a porzioni di Store e quando un valore all'interno dello Store viene modificato, tutti i componenti iscritti a quella porzione di Store, che possiede il dato modificato, verranno notificati ed automaticamente aggiornati.

Tutti i componenti quindi, ottengono il proprio stato dallo store ed inviano le relative modifiche allo store, senza doversi preoccupare di tutti gli altri componenti che richiedono il cambio di stato (Si osservi la Figura 49 in basso).

In tal modo Redux rende il flusso di dati più facile da gestire.

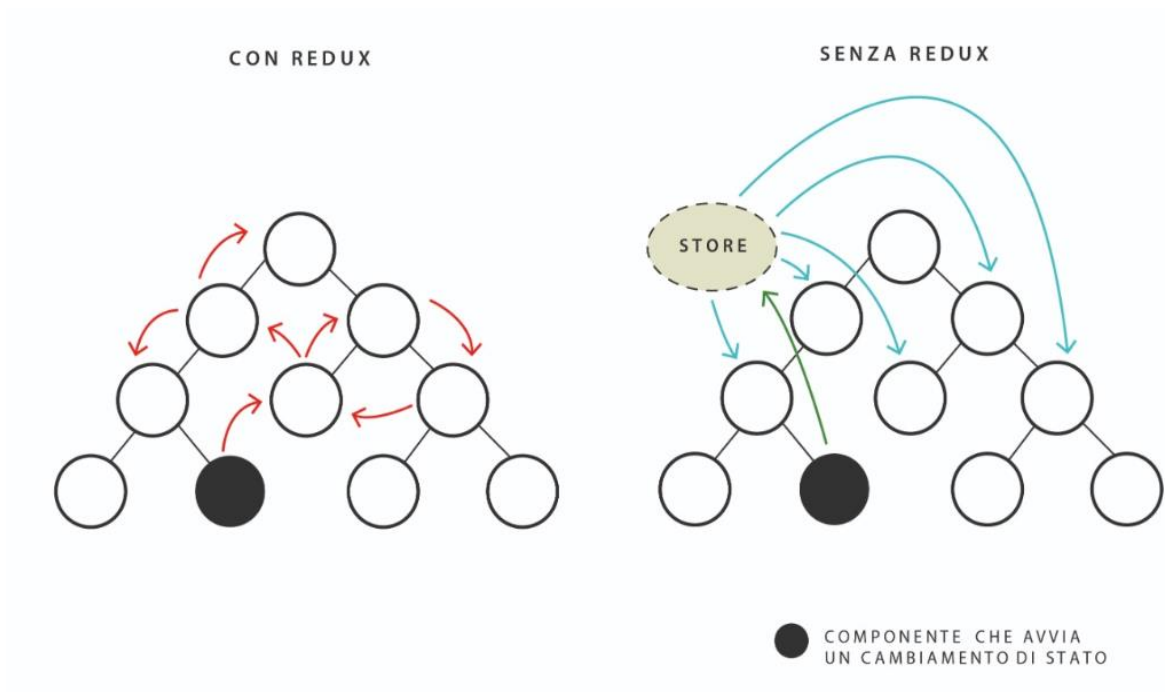


Figura 49: Differenza dei flussi di dati con e senza Redux [38]

### 5.5.3 Struttura dell'applicazione

Di seguito (Figura 50), si rappresenta l'organizzazione della cartella del progetto. Per ragioni di spazio non vengono riportati i contenuti completi, ma viene aggiunto qualche commento relativo alle cartelle principali.

Da una prima osservazione della struttura del progetto si nota che, all'interno del frontend, vi sono due macro cartelle: components e store. Dentro la prima vengono inseriti tutti i componenti che formano il progetto e, all'interno della seconda, vengono salvate le azioni che interagiscono col Redux Store.

Il file denominato "ApiService.js" contiene invece tutte le funzioni che permettono l'interazione, in lettura e in scrittura, con la Blockchain EOS.

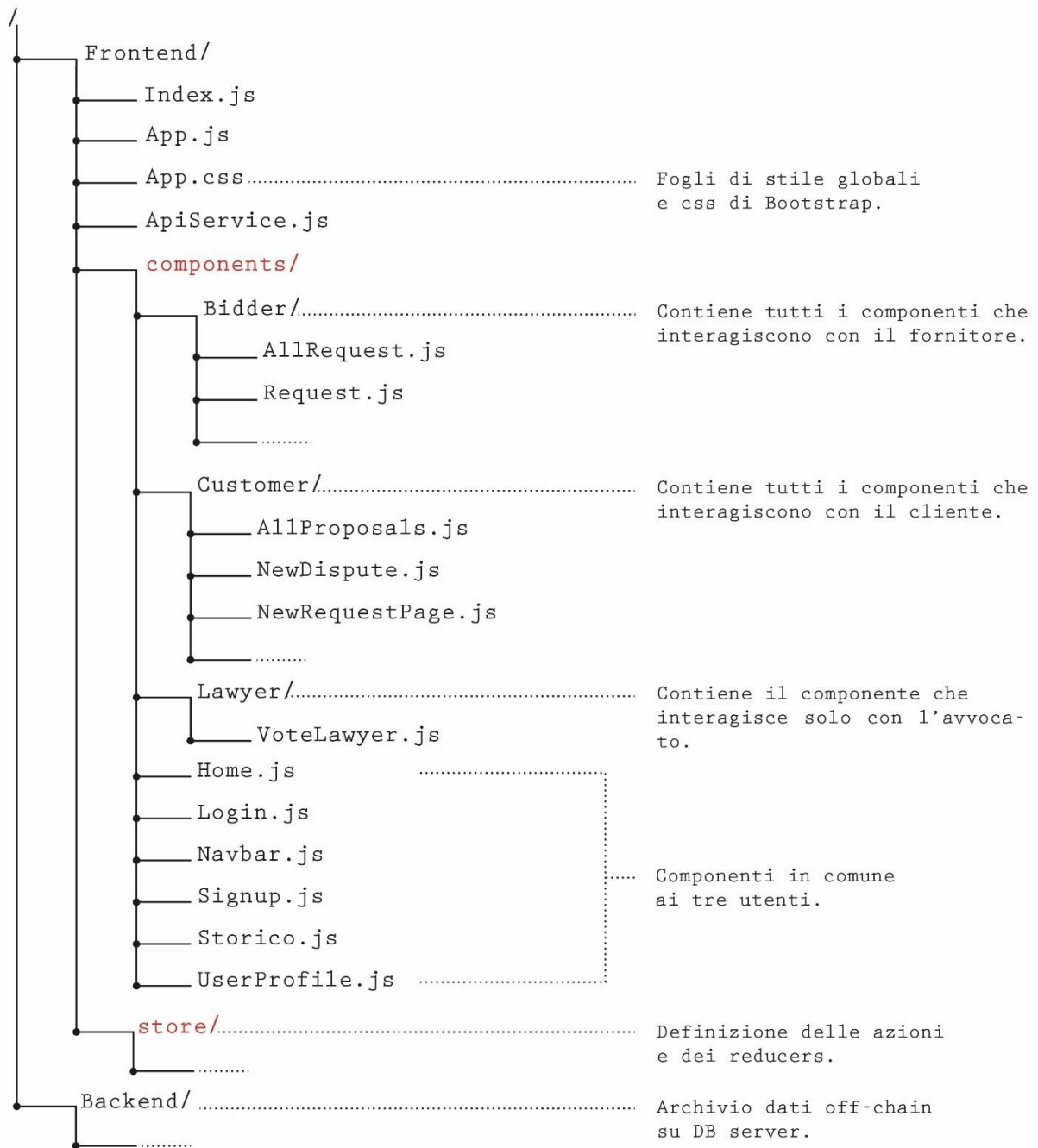


Figura 50: Organizzazione della cartella del progetto

## 5.5.4 Interazione con la Blockchain EOS

Lo sviluppo della Decentralized Application ha previsto l'uso della libreria *eosjs* al fine di effettuare le necessarie chiamate alla blockchain EOS. *Eosjs* è una libreria Javascript che supporta la firma e la trasmissione di transazioni, nonché l'interazione con altre parti dell'API EOSIO RPC.

Si è fatto uso della classe *eosjs.Api*, che possiede alcuni metodi di alto livello necessari per l'interazione con una blockchain EOSIO e, in particolare, si è utilizzato il metodo *Api.transact* per firmare una transazione e trasmetterla sulla catena. Tale metodo gestisce infatti la complessità della firma e della serializzazione delle transazioni.

Inoltre, si è fatto uso della classe *eosjs.JsonRpc* per interrogare direttamente i nodi API RPC ed, in particolare si è utilizzato il metodo *JsonRpc.get\_table\_rows* per ottenere un oggetto contenente le righe delle tabelle create.

Il codice sottostante, in Figura 51, è utilizzato nella classe *ApiService.js* per inizializzare *eosjs* e chiamare le actions del contratto intelligente, distribuito sull'account "dighand".

```
// Main action call to blockchain
async function takeAction(action, dataValue) {

  const privateKey = localStorage.getItem("dighand_key");
  const rpc = new JsonRpc('http://127.0.0.1:8888');
  const signatureProvider = new JSSignatureProvider([privateKey]);
  const api = new Api({ rpc, signatureProvider, textDecoder: new TextDecoder(), textEncoder: new TextEncoder() }); //mettere in una fnz

  // Main call to blockchain after setting action, account_name and data
  try {
    const resultWithConfig = await api.transact({
      actions: [{
        account: 'dighand',
        name: action,
        authorization: [{
          actor: localStorage.getItem("dighand_account"),
          permission: 'active',
        }],
        data: dataValue,
      }],
    }, {
      blocksBehind: 3,
      expireSeconds: 30,
    });
    return resultWithConfig;
  } catch (err) {
    throw(err)
  }
}
```

Figura 51: Inizializzazione di *eosjs* e chiamata delle actions

A seguire, in Figura 52 e 53, sono riportati due esempi di frammenti di codice scritti all'interno della classe *ApiService.js*. Tale classe nasce con l'obiettivo di permettere all'utente di interagire con la blockchain in scrittura e lettura.



```

static async dighandshake(req_description,deadline,price) {
  return takeAction("dighandshake", {
    customer_account:localStorage.getItem("dighand_account"),
    request_description: req_description,
    bidder_stake: "50 DH", //di default
    customer_stake:"50 DH",
    Deadline: deadline,
    amount: price+"DH"
  });
}

```

**Figura 52:** Codice in scrittura per l'invio e la memorizzazione di un nuovo accordo su blockchain

```

static async getTableRequests() {
  try {
    const rpc = new JsonRpc('http://127.0.0.1:8888');
    const result = await rpc.get_table_rows({
      "json": true,
      "code": 'dighand', // contract who owns the table
      "scope": 'dighand', // scope of the table
      "table": 'handshakes', // name of the table as specified by the contract abi
    });
    return result.rows;
  } catch (err) {
    console.error(err);
  }
}

```

**Figura 53:** Codice per la lettura della tabella degli accordi

### 5.5.5 Descrizione Interfacce

In questo paragrafo sono raffigurate le principali pagine dell'applicazione sviluppata in React.js, evidenziando gli utenti che vi hanno accesso e le considerazioni sulle scelte effettuate. In aggiunta, le principali interfacce saranno accompagnate dal relativo Sequence Diagram.

Il Sequence Diagram è infatti un diagramma del linguaggio UML che mette in relazione attori e classi di un sistema, evidenziando le interazioni dal punto di vista temporale. In poche parole, mostra la sequenza delle istruzioni da eseguire per caso d'uso.

#### Login

Una volta lanciata l'applicazione, l'utente è indirizzato alla pagina di Login (Figura 54) in cui gli verrà richiesto l'inserimento del proprio username, corrispondente all'account eosio, e la password, ovvero la chiave privata associata a quell'account.

Se i dati inseriti corrispondono a quelli presenti sulla blockchain l'utente verrà indirizzato alla propria Home Page, altrimenti l'accesso sarà negato e potrà registrarsi attraverso l'apposito link.











**Figura 54:** Pagina di Login della Web Application (l'immagine di sfondo è stata presa da [37])

## Home

Dopo aver effettuato il Login, l'utente viene reindirizzato nella propria Homepage. L'esempio in Figura 55 riporta l'Home page della parte cliente che fruisce del servizio. In alto è presente la barra di navigazione che consente di spostarsi tra lo Storico (Figura 56), l'Home (Figura 55) e la fuoriuscita dal sistema. Le linee guida che si è deciso di seguire per la fase di progettazione delle interfacce utente sono: chiarezza, semplicità, numero ridotto di voci, font leggibile e l'inserimento di pochi (ma fondamentali) collegamenti a cui rimandare l'utente. Fedeli a questa idea si è deciso di suddividere tale interfaccia in sole tre, ma essenziali sezioni:

1. Una carta utente per la consultazione dei dati personali. A sinistra di Figura 55 infatti, è possibile notare un riquadro, comune a tutti gli utenti, che mostra i dati personali di chi sta utilizzando la piattaforma. Le informazioni riportate, specie il "balance" e lo "stake", sono utili a chiunque stia effettuando delle operazioni di acquisto-vendita di prodotti o servizi, poiché permettono all'utente di tenere sotto controllo l'attuale disponibilità.
2. Un pulsante per effettuare una nuova richiesta accedendo al form in Figura 57 (nel caso del profilo fornitore avremmo avuto il pulsante per controllare le varie richieste in corso, Figura 59)
3. Una tabella contenente gli accordi in corso. La tabella in questione occupa quasi l'intera schermata ed offre la visualizzazione della descrizione della richiesta, del fornitore scelto, dello stato e di alcune icone o pulsanti utilizzati per interagire con il fornitore e la piattaforma.

Di seguito è riportato il significato di ogni bottone con il relativo collegamento.

-  Icona che rappresenta l'attesa del voto degli avvocati in stato di disputa.
-  Icona che mostra lo stato di esecuzione in attesa della conferma della conclusione dei lavori da parte del fornitore.
-  Pulsante che permette all'utente di concludere l'accordo senza alcun contenzioso.
-  Pulsante che permette all'utente di aprire una nuova disputa (Figura 65) qualora il prodotto ricevuto non fosse conforme alle specifiche.
-  Bottone che permette al cliente di osservare l'elenco dei fornitori che si stanno proponendo per soddisfare la richiesta dell'acquirente (rimando alla Figura 60)
-  Bottone che rimanda alla contrattazione tra cliente e fornitore (Figure 62 e 63)
-  Pulsante che consente al fornitore del bene o servizio di confermare la conclusione del lavoro
-  Pulsante che consente all'avvocato di accedere alla pagina indicata in Figura 67 per sottomettere la sentenza relativa ad una disputa



Richiesta	Fornitore	Stato	
sito web professionale	claudia	Dispute	
fornitura 30 kg di pere	claudia	Execution	
fornitura 20 kg di mele rosse	claudia	Bargaining	
fornitura 25 kg di pane integrale	simon	Finalized	 
consulenza arredo interni	null	Request	

Figura 55: Home page del profilo cliente

## Storico

La Figura 56 rappresenta la schermata relativa allo storico ed è caratterizzata da una semplice tabella contenente tutti gli accordi conclusi tra il cliente e i vari fornitori,

indifferentemente che si siano conclusi positivamente o negativamente (con una disputa).



The screenshot shows a web interface with a dark blue header containing 'Benvenuto' on the left and 'Home', 'Storico', and 'Logout' on the right. Below the header is a section titled 'Il tuo Storico'. Underneath this title is a table with five columns: 'Richiesta', 'Cliente', 'Fornitore', 'Prezzo', and 'Stato'. The table contains two rows of data.

Richiesta	Cliente	Fornitore	Prezzo	Stato
sito web professionale	bob	claudia	100 DH	Closed
fornitura 30 kg di pere	bob	claudia	200 DH	Closed

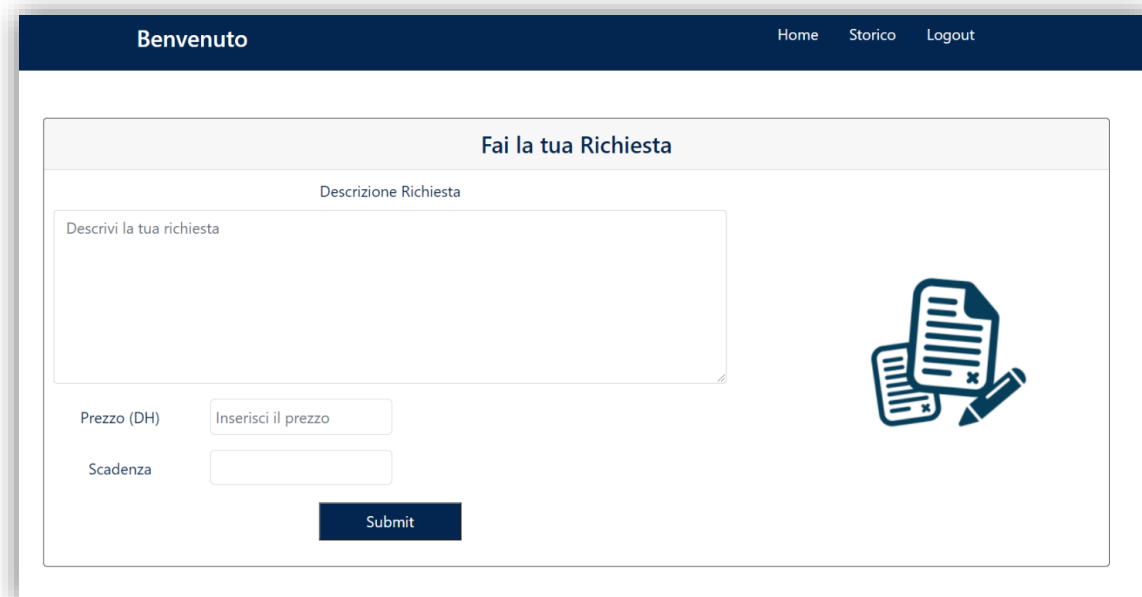
Figura 56: La pagina dedicata allo Storico

### Pagina di Nuova Richiesta

Qualora un cliente desiderasse effettuare una nuova richiesta verrebbe rimandato alla compilazione del Form sottostante (Figura 57), in cui dovrà inserire: la descrizione del servizio che desidera ricevere, il prezzo che intende spendere e il termine entro cui richiede l'adempimento del servizio.

Al momento della sottomissione i dati sono salvati on-chain.

A seguire è raffigurato il sequence diagram (Figura 58) di tale processo.



The screenshot shows a web interface with a dark blue header containing 'Benvenuto' on the left and 'Home', 'Storico', and 'Logout' on the right. Below the header is a section titled 'Fai la tua Richiesta'. Underneath this title is a form with the following elements:

- A text area labeled 'Descrizione Richiesta' with the placeholder text 'Descrivi la tua richiesta'.
- A text input field labeled 'Prezzo (DH)' with the placeholder text 'Inserisci il prezzo'.
- A text input field labeled 'Scadenza'.
- A dark blue 'Submit' button.
- An icon of a document with a pencil and a checkmark, indicating a request or submission.

Figura 57: Pagina in cui il cliente effettua una nuova Richiesta

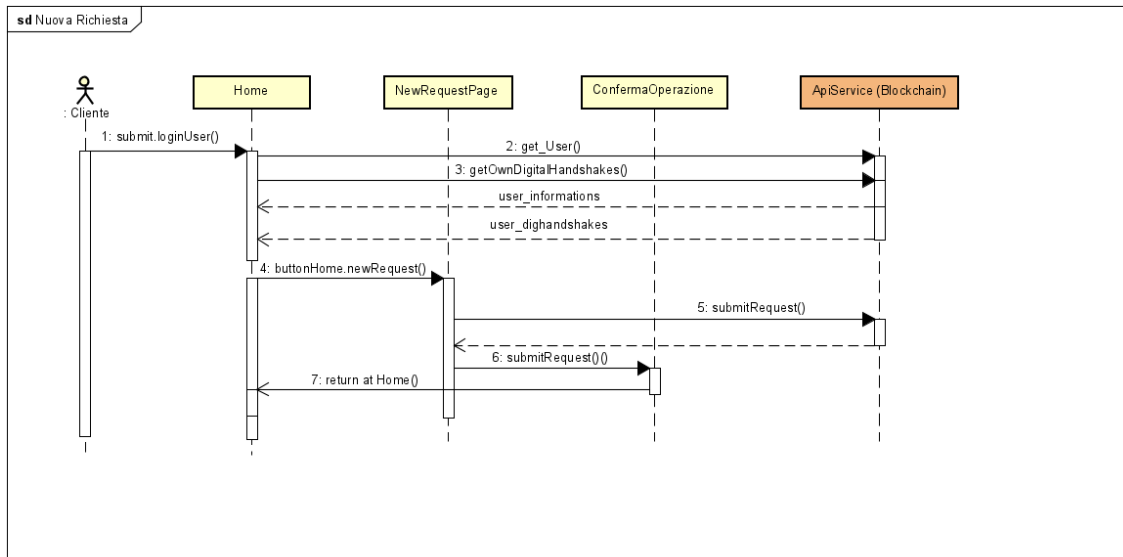


Figura 58: Sequence Diagram relativo ad una nuova Richiesta

## Elenco Richieste

Il fornitore che volesse osservare le richieste ancora aperte, potrebbe accedere alla schermata in basso (Figura 59), attraverso il pulsante “Controlla Richieste” presente nella propria Home page. Una tabella occuperà l’intero spazio della pagina e permetterà all’eventuale offerente di candidarsi per effettuare i servizi che riterrà di propria competenza. Le eventuali proposte sono salvate off-chain.

Benvenuto					Home	Storico	Logout
Tabella richieste							
Richiesta	Cliente	Prezzo	Scadenza	Proponiti			
sito web professionale	bob	200 DH	2020-11-27T00:00:00.000	➡			
fornitura 30 kg di pere	bob	150 DH	2020-11-04T00:00:00.000	➡			
fornitura 20 kg di mele rosse	bob	70 DH	2020-10-31T00:00:00.000	➡			
consulenza arredo interni	alison	50 DH	2020-11-10T00:00:00.000	➡			

Figura 59: Pagina in cui l’offerente controlla le richieste in corso e si propone

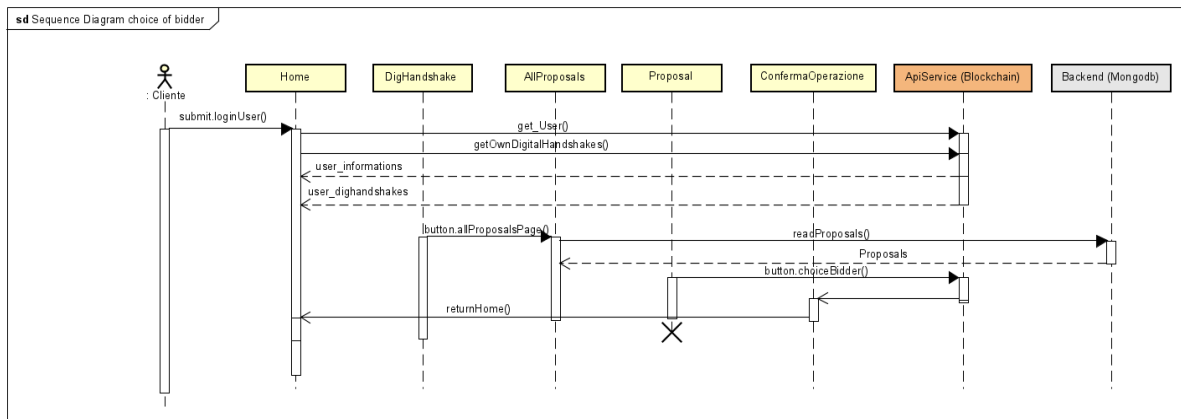
## Elenco Proposte

La schermata seguente (Figura 60) permette al cliente di controllare coloro che si propongono per soddisfare la domanda da lui effettuata e scegliere così l’offerente migliore, ovvero colui che possiede un maggiore livello di rating. Quest’ultimo verrà

memorizzato on chain. A seguire, in Figura 61, è riportato il sequence diagram relativo alla procedura di scelta del fornitore.

Benvenuto		Home	Storico	Logout
<b>Proposte relative alla richiesta n. 1</b>				
Offerente	Rating			
claudia	6			<b>Scegli</b>
simon	5			<b>Scegli</b>

**Figura 60:** Pagina in cui il cliente controlla le proposte dei fornitori e sceglie la migliore



**Figura 61:** Sequence Diagram relativo alla procedura di scelta del fornitore

### Schermate di Contrattazione

Il cliente, una volta scelto l’offerente migliore, potrà inizializzare la fase di contrattazione (Figura 62) nella quale propone un oggetto del contratto, un prezzo e una scadenza. A seguire, il fornitore osserverà la proposta del cliente (Figura 63) e potrà accettare oppure contro proporre. Il bottone “Controproponi” reindirizzerà nuovamente al form di Figura 62.

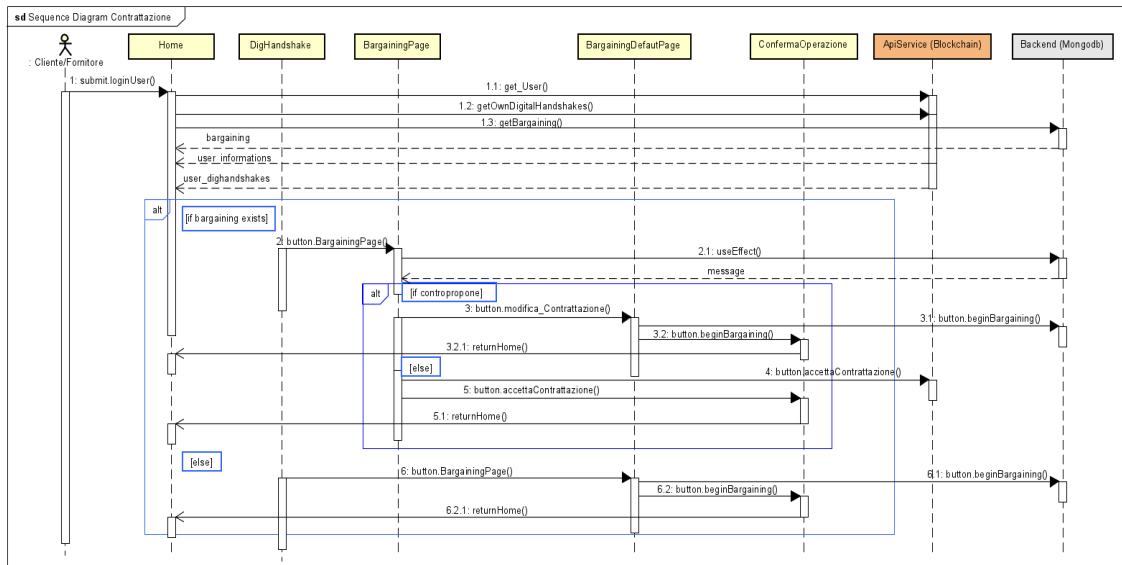
Questo scambio di “messaggi” tra cliente-fornitore è salvato off-chain, ma nel momento in cui una delle due parti dovesse cliccare su “Accetta”, l’ultima proposta sarà memorizzata on-chain.

**Figura 62:** Form di Contrattazione 1

Proposta (n. 1)	
Oggetto del contratto	Le quantità consegnate a seguito di ogni ordine di consegna saranno pari a quelle ordinate. Eventuali tolleranze sulla quantità consegnata rispetto a quella ordinata potranno essere richieste da [FORNITORE] purchè non superiori al +10% / -0%. In caso di eccedenze non concordate, questa avrà a propria scelta la facoltà di trattenere il materiale imputandolo a ordini futuri. Le carenze dovranno invece essere al più presto compensate entro 2 giorni di consegna; oltre tali termini saranno trattate come ritardi.
Deadline	2020-10-30T23:00:00.000Z
Prezzo	200
Risposta	<a href="#">Accetta</a> <a href="#">Controproponi</a>

**Figura 63:** Pagina di Contrattazione 2

A seguire, in Figura 64, il diagramma di sequenza relativo alla fase di contrattazione.



**Figura 64:** Sequence Diagram relativo alla fase di contrattazione

### Pagina di apertura nuova disputa

Di seguito, in Figura 65, si riportata il form che il cliente sarà chiamato a compilare per aprire una nuova disputa, nel quale verrà richiesto l'inserimento della motivazione dell'apertura del reclamo con eventuali foto o documenti da allegare. Queste informazioni saranno salvate on-chain insieme agli account dei tre avvocati nominati dal sistema per risolvere la disputa.

**Figura 65:** Form per aprire una nuova disputa

A seguire il sequence diagram relativo alla fase di apertura disputa (Figura 66).



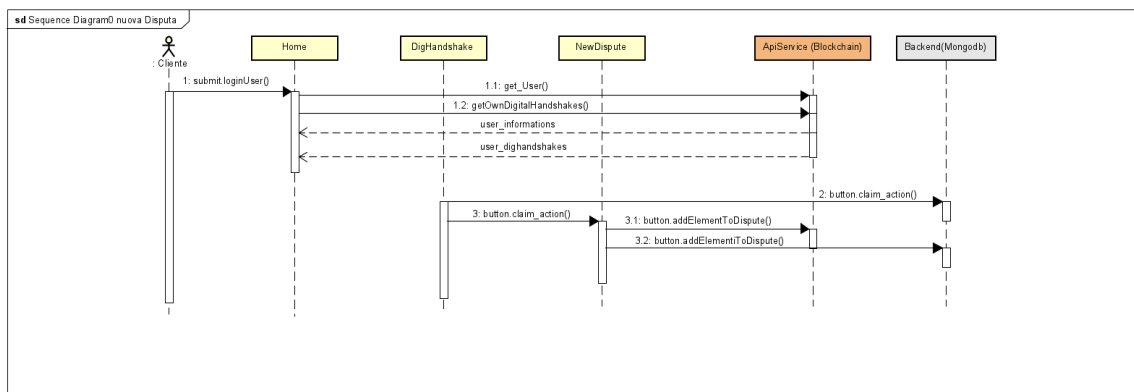


Figura 66: Sequence Diagram relativo alla fase di apertura disputa

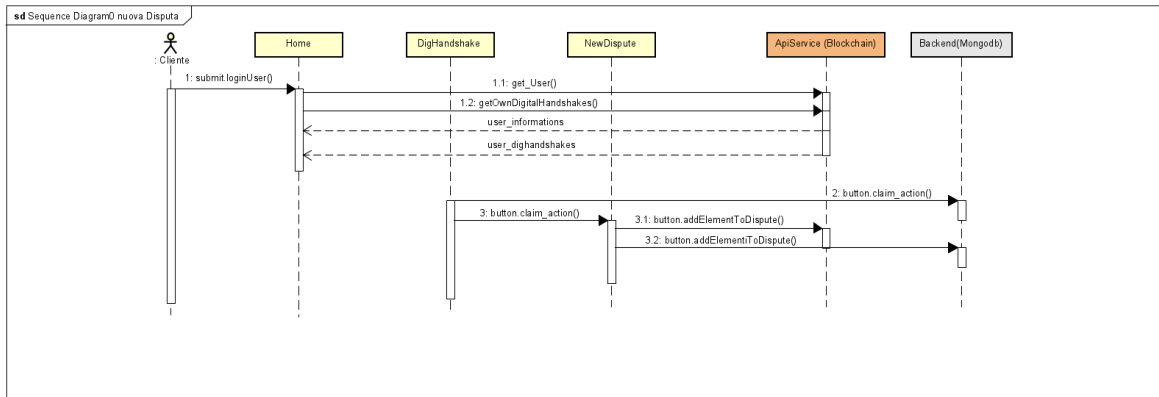
### Pagina per la votazione dell'avvocato

In basso, in Figura 67, è riportata l'interfaccia che trasmette all'avvocato le informazioni complete della disputa che è chiamato a sentenziare. Egli potrà sottomettere la "votazione" a favore del cliente o del fornitore e memorizzarla direttamente sulla blockchain.

Benvenuto		Home	Storico	Logout
<b>Disputa</b> transazione n: 1				
Commissa:	fornitura 30 kg di pere			
Oggetto del contratto:	Le quantità consegnate a seguito di ogni ordine di consegna saranno pari a quelle ordinate. Eventuali tolleranze sulla quantità consegnata rispetto a quella ordinata potranno essere richieste da [FORNITORE] purchè non superiori al +10% / -0%. In caso di eccedenze non concordate, questa avrà a propria scelta la facoltà di trattenere il materiale imputandolo a ordini futuri. Le carenze dovranno invece essere al più presto compensate entro 2 giorni di consegna; oltre tali termini saranno trattate come ritardi. job finished in time			
Prezzo:	200 DH			
Scadenza:	2020-10-31T00:00:00.000			
Descrizione Reclamo:	Frutta marcia e non confezionata nel modo adeguato. Foto in allegato. <a href="#">↓</a>			
Sentenza:	<input type="button" value="Cliente"/> <input type="button" value="Fornitore"/>			

Figura 67: Interfaccia che consente all'avvocato di sottomettere la propria sentenza

Il sequence diagram relativo alla sentenza dell'avvocato è riportato di seguito, in Figura 68.



**Figura 68:** Sequence Diagram relativo al voto dell'avvocato

## 5.5.6 Class Diagram Web Application

In questo paragrafo si rappresenta il class diagram della nostra Web Application, ottenuto attraverso il cosiddetto *reverse engineering*, ossia partendo dal codice completo e andando a ritroso fino al raggiungimento del modello sottostante (Figura 69).

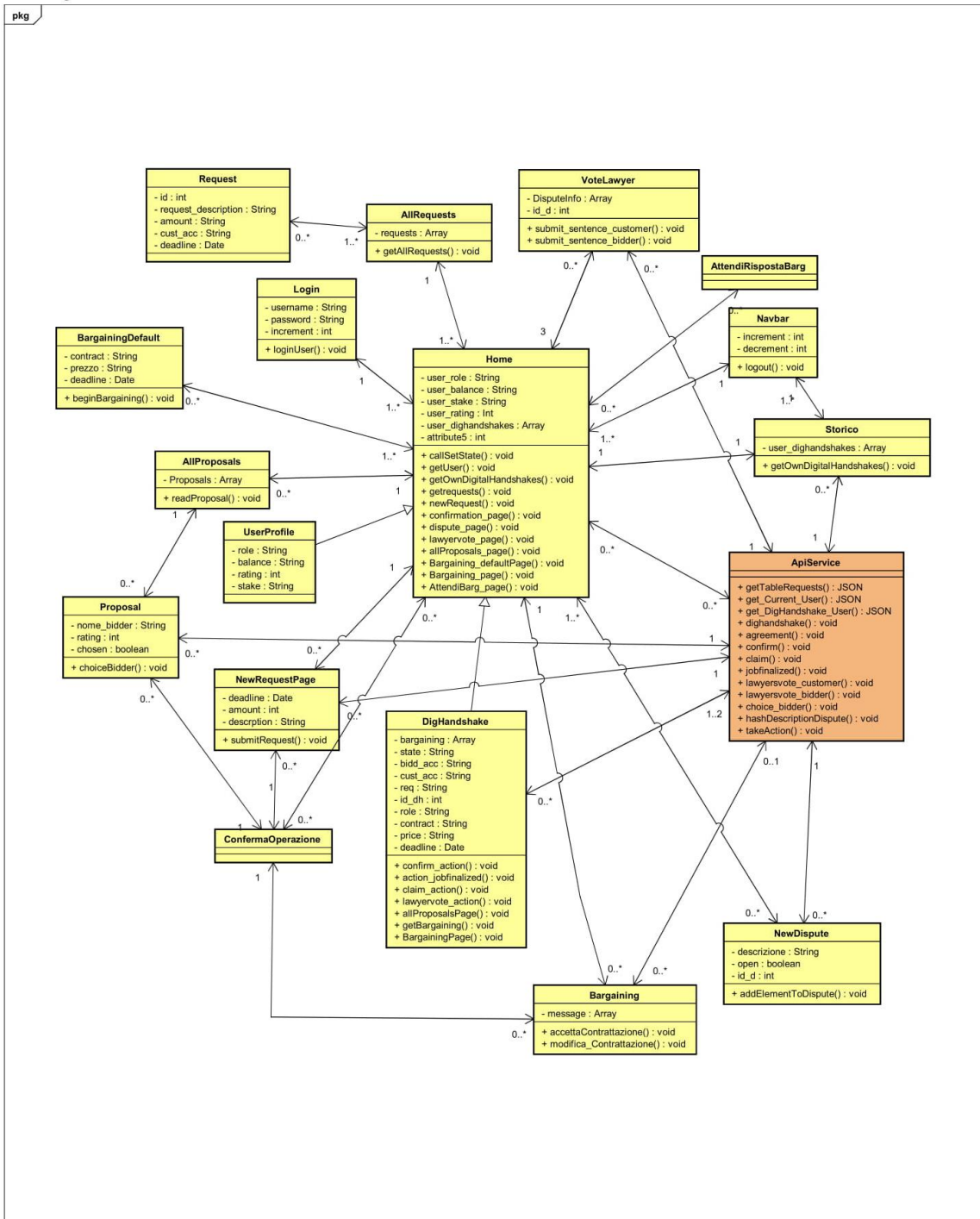


Figura 69: Class Diagram relativo alle classi della Web Application

### 5.5.7 DB Server

Durante le fasi di analisi del progetto si è deciso di effettuare il salvataggio off-chain dei dati non essenziali per la logica smart contract. A tal proposito si è preferito utilizzare un database non relazionale, MongoDB. In basso (Figura 70) è riportato il codice utilizzato per la connessione del server con il db esterno.

#### Mongo DB

MongoDB è un database di documenti gratuito che offre scalabilità e flessibilità. Esso permette la memorizzazione dei dati in documenti JSON flessibili, e ciò vuol dire che i campi dei dati possono variare da un documento all'altro.

Con l'utilizzo del tool MongoDBCompass è stato possibile creare il database denominato *DigitalHandshake\_db*, composto da tre sotto collezioni:

- *bargaining\_collection* in cui verranno salvati i messaggi scambiati tra il cliente e il fornitore in fase di contrattazione
- *dispute\_collection* in cui si memorizzeranno i dati relativi alle dispute
- *proposals\_collection* finalizzata al salvataggio dei fornitori che si propongono alle offerte dei clienti

Di seguito, in Tabella 8, sono riportate le strutture dati in formato JSON delle tre collections.

<b>bargaining_collection</b>	<b>proposals_collection</b>	<b>dispute_collection</b>
<pre>{   "_id": Int32,   "Contract_object": String,   "Price": String,   "User": String,   "Counter": Int32,   "Deadline": Date,   "__v": 0 }</pre>	<pre>{   "_id": Int32,   "Customer_account": String,   "Bidder_account": String,   "Amount": String,    "Request_description":String,   "Contract_object": String,   "Description": String   "Deadline": Date,   "__v": 0 }</pre>	<pre>{   "Proposal_array": [{     "_id": String,     "rating": Int32   }],   "__v": 0 }</pre>

**Tabella 8:** Strutture dati delle tre collezioni in MongoDB

A seguire è riportato il codice di connessione tra il server e il database di MongoDB (Figura 70) per permettere al client di effettuare le richieste http (metodi GET e POST).

```
app.use(express.json());
app.use(cors());
mongoose.connect("mongodb://localhost:27017/DigitalHandshake_db",
  {
    useNewUrlParser:true,
  });
```

**Figura 70:** Connessione a MongoDB

## 6. Conclusioni e Prospettive Future

L'ultimo capitolo della tesi, oltre a fornire una visione d'insieme del lavoro condotto, si concentrerà sull'analisi delle sfide e criticità ancora aperte relative alla soluzione proposta e sviluppata in questo lavoro di tesi e le possibili applicazioni future.

### 6.1 Conclusioni

Dalle analisi effettuate nei capitoli 3 e 4 è emerso che, le aziende che operano nel settore B2B (business to business), in particolar modo microimprese e PMI, sono ancora in gran parte radicate agli strumenti e alle metodologie del passato, sia in merito alle modalità di interazione tra le parti che alle procedure di pagamento utilizzate. Infatti, come ampiamente discusso, la maggior parte di questi lavoratori sono soliti concretizzare i propri affari tramite la classica "stretta di mano" fisica, con le conseguenti problematiche:

1. Mancanza di completa fiducia tra le parti
2. Ritardi nei pagamenti
3. Complessa e costosa gestione di eventuali dispute

L'intero progetto di tesi è stato guidato dall'obiettivo di risolvere tali problematiche, attraverso la progettazione di un nuovo prodotto in grado di fornire la stretta di mano digitale, anche detta "Digital Handshake", tra gli attori del mercato di riferimento.

A tal proposito, nel capitolo 5, è stato descritto il cuore del lavoro di tesi, ovvero lo sviluppo di una DApp (Decentralized Application) in grado di soddisfare l'obiettivo prefissato sfruttando la tecnologia blockchain, descritta nel capitolo 2, e i vantaggi che essa offre in termini di sicurezza e affidabilità.

Di seguito si riporta l'analisi SWOT, ovvero un framework utile per mostrare schematicamente le caratteristiche del prodotto oggetto di tesi e del mercato in cui si intende competere.

#### **Analisi SWOT**

L'analisi SWOT è l'acronimo di "strengths, weakness, opportunities, threats", e rappresenta uno strumento di pianificazione (Tabella 9) strategica in grado di valutare i punti di forza (Strengths), i punti di debolezza (Weaknesses) della soluzione realizzata unitamente alle opportunità (Opportunities) e le minacce (Threats) dell'ambiente esterno.

**Punti di forza:** la soluzione offre la possibilità di effettuare il Digital Handshake, sfruttando l'innovativa tecnologia blockchain (EOS) che garantisce performance migliori.

Inoltre, si propone un prodotto economico da utilizzare (non si deve pagare alcuna tassa per usufruire del servizio offerto), con pagamenti istantanei ed una rapida ed economica gestione delle controversie, affidata al giudizio di tre avvocati esterni.

**Punti di debolezza:** nella soluzione proposta il voto degli avvocati è pubblico e gli accordi presi tra le parti non hanno validità legale.

**Opportunità:** Relativamente all'ambiente esterno, la necessità di incrementare il commercio digitale all'interno del mercato B2B potrebbe rappresentare una valida opportunità per una maggiore diffusione della soluzione, eventualmente adattabile anche ad altri mercati, come ad esempio il C2C (consumer-to-consumer).

**Minacce:** La presenza di prodotti concorrenti già definiti e lanciati sul mercato è un chiaro segno di svantaggio competitivo in termini di tempo, dovuto principalmente all'economia di rete che vige ed influenza tale settore. Inoltre, il forte legame ancora esistente tra le aziende e i contratti tradizionali limiterebbe il numero di possibili utenti che utilizzerebbero il nuovo servizio offerto. Si ricorda infatti, che ogni cambiamento richiede dei costi da sostenere, anche definiti "switching costs" (ad esempio il costo che bisogna affrontare per acquisire le conoscenze e le competenze d'uso della nuova tecnologia o il costo per l'acquisto di beni complementari), che potrebbero costituire una barriera all'innovazione. Inoltre, la scarsa conoscenza e comprensione della nuova tecnologia Blockchain potrebbe costituire un ulteriore limite per la diffusione e per l'adozione della nuova soluzione.

<b>Forze</b>	<b>Debolezze</b>
<ul style="list-style-type: none"> <li>• Tecnologia all'avanguardia (EOS) che non richiede costi per l'emissione di transazioni</li> <li>• Piattaforma user-friendly</li> <li>• Pagamenti istantanei</li> <li>• Gestione delle controversie rapida, economica e affidata al un giudizio di avvocati esterni</li> </ul>	<ul style="list-style-type: none"> <li>• Mancanza di anonimato delle sentenze degli avvocati</li> <li>• Contratti non legalmente validi</li> </ul>
<b>Opportunità</b>	<b>Minacce</b>
<ul style="list-style-type: none"> <li>• Necessità di maggiori risorse digitali nel B2B</li> <li>• Espandere il mercato di riferimento (B2C o C2C)</li> </ul>	<ul style="list-style-type: none"> <li>• Presenza di concorrenti sul mercato</li> <li>• Prodotti competitors già definiti e lanciati sul mercato</li> <li>• Economia di rete</li> <li>• Aziende ancora dipendenti dai contratti tradizionali</li> <li>• Switching cost</li> <li>• Ridotta conoscenza e comprensione della tecnologia Blockchain</li> </ul>

**Tabella 9:** Analisi SWOT

In conclusione, il prodotto sviluppato non è ancora pronto per essere lanciato sul mercato e concorrere con i prodotti già esistenti, necessita infatti di future implementazioni che mirino a eliminare, o per lo meno ridurre, i punti di debolezza dello stesso.

## 6.2 Futuri Sviluppi: i Contratti Ricardiani

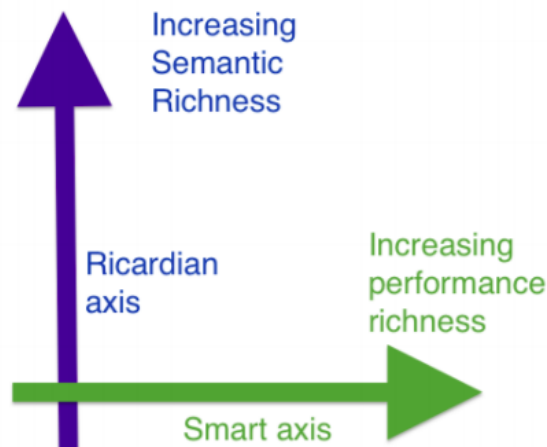
Dalle precedenti osservazioni è emerso come la mancanza di contratti legalmente validi sia uno dei punti della soluzione, ragion per cui in questo paragrafo si presenterà il “Contratto Ricardiano” (o Ricardian Contract), che sarà oggetto di una futura implementazione del progetto di tesi. Questo termine risale al 1995 e fa riferimento al contratto *Ricardo*, creato come parte del sistema di pagamento Ricardo dal programmatore Ian Grigg, uno dei primi pionieri della crittografia finanziaria. Grigg ha definito il “Ricardian Contract” come *“un unico documento che è a) un contratto offerto da un emittente ai titolari, b) un diritto prezioso detenuto dai titolari e gestito dall'emittente, c) facilmente leggibile da persone (come un contratto su carta), d) leggibile da programmi (analizzabile come un database), e) firmato digitalmente, f) contenente le chiavi e le informazioni del server, e g) associato a un identificatore unico e sicuro”*.<sup>[39]</sup> In sostanza, il contratto ricardiano è un documento leggibile sia dall'uomo sia dalla macchina, ed ha piena validità legale. Per tale motivo esso sarà integrato, in un prossimo sviluppo, in modo da trasformare l'attuale punto di debolezza in un punto di forza.

Poiché le imprese sono costantemente chiamate a rapportarsi con clienti e fornitori in un ambiente in continua evoluzione, da una parte è molto importante che queste siano in grado di comprendere a pieno il testo del contratto, di qualsiasi natura esso sia, e, dall'altra, occorrerebbe disporre di un contratto adattabile alle diverse circostanze. Attualmente però, l'applicazione del solo smart contract non garantisce tali fondamenti. Infatti, per quanto sia semplice per i programmatori e gli sviluppatori scrivere e leggere il codice di uno smart contract, non si può dire lo stesso per chi svolge il ruolo di avvocato o per chiunque non abbia competenze di programmazione. Ciò è la causa che porta le imprese ad essere ancora fortemente legate ai contratti redatti in linguaggio naturale, poiché sono comprensibili da tutti e più flessibili.

Pertanto, una soluzione “ibrida” che garantisca la chiarezza e la facilità interpretativa di un documento scritto in un linguaggio naturale, e al contempo che garantisca i benefici di un contratto intelligente, ovvero automatizzazione delle diverse fasi dell'accordo, trasparenza, immutabilità, accelerazione delle prestazioni e risparmio di tempo e risorse, potrebbe essere la risposta giusta per sfruttare a pieno l'enorme potenziale degli smart contract.



In Figura 71 si riporta il grafo proposto da Ian Grigg [40] rappresentante “Ricardian Contract” e “Smart Contract” su due assi perpendicolari. L’asse y, denominato “asse Ricardiano”, rappresenta la crescita della ricchezza semantica della prosa legale e l’asse x, chiamato “asse Smart”, esprime la crescita delle prestazioni in termini di esecuzione del codice del contratto intelligente. Entrambi hanno un ruolo rilevante e, con opportuni sviluppi, potrebbero muoversi lungo gli assi simultaneamente.



**Figura 71:** l'asse y rappresenta una capacità crescente di catturare la semantica di un contratto legale intelligente, mentre l'asse x rappresenta il aumentare le prestazioni del codice del contratto intelligente. [40]

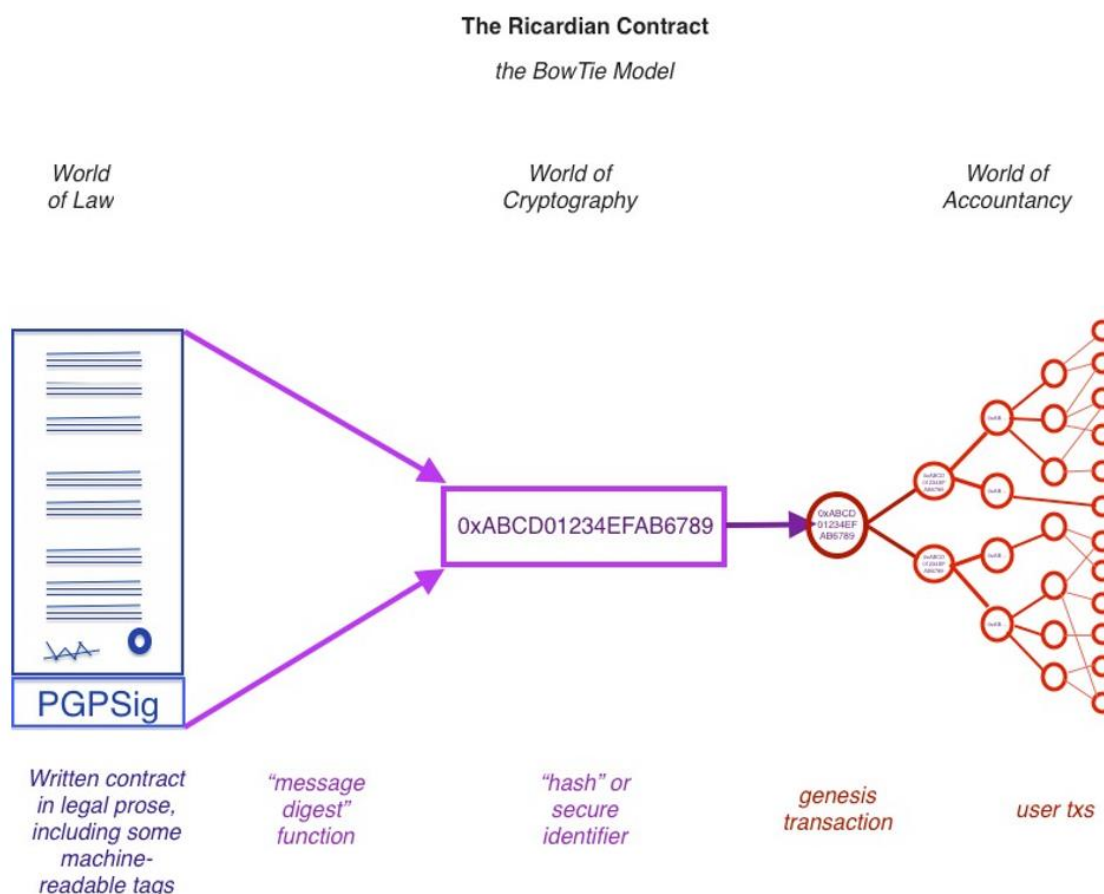
### 6.2.1 Le Caratteristiche del Contratto Ricardiano

Un Contratto Ricardiano può anche essere descritto come un documento in prosa che possiede un valore legale per l'emissione di azioni, obbligazioni, valute e per la vendita di beni e servizi via internet. Esso nasce per delineare le intenzioni e registrare l'accordo, a differenza del contratto intelligente che serve per eseguirlo.

Il contratto ricardiano è composto da tre elementi collegati tra loro: smart contract, parametri, e prosa. [41]

- Lo *smart contract*: il codice eseguibile dalla macchina che organizza e controlla l'arrivo di eventi e l'avvio di azioni.
- I *parametri*: sono le condizioni del contratto, come ad esempio i prezzi, le date, le quantità, le valute ecc. Tali parametri influenzano e determinano l'esecuzione dello smart contract.
- La *prosa*: il documento scritto in un linguaggio naturale che rende direttamente comprensibile il codice informatico.

Tale strumento è estremamente sicuro grazie alla firma crittografica e all'utilizzo di un hash identificativo. Ciò significa che ogni partecipante può vincolarsi e apportare la firma sul contratto utilizzando la propria chiave privata, in tal modo sarà possibile rintracciare le parti coinvolte e renderle responsabili. Inoltre, ogni "accordo" è identificabile in modo univoco dal proprio hash (vedi Figura 72) [42], che impedirà, a chiunque voglia modificare maliziosamente le informazioni presenti sul documento, di poterlo fare.



**Figura 72:** Il Contratto Ricardiano [42]

Di seguito sono riportate le caratteristiche che rendono un contratto ricardiano uno strumento di elevata qualità [42]:

1. È analizzabile dall'uomo
2. È un documento stampabile
3. È analizzabile dalla "macchina"
4. Le versioni nel punto 1,2 e 3 sono equivalenti
5. Firmato dall'emittente
6. Possiede tutte le informazioni in un unico documento, includendo le parti dell'accordo e la firma

7. Rappresenta un contratto legale
8. Gode di elevata sicurezza, infatti qualsiasi tentativo di eliminare un collegamento tra un riferimento e il contratto non è fattibile
9. È estensibile. Può interpretare obbligazioni, azioni, diritti ecc
10. Identifica l'emittente (firmatario del contratto) e il server di emissione
11. Non può essere modificato da nessuno, ad eccezione dell'emittente o delle altre parti contraenti
12. È verificabile in natura
13. Non necessita di autorizzazioni. Il contratto può essere creato e utilizzato da chiunque senza richiedere assegnazioni di spazi controllati

Di seguito, in Figura 73, è riportato l'esempio di un contratto ricardiano associato al trasferimento di una certa quantità di token.

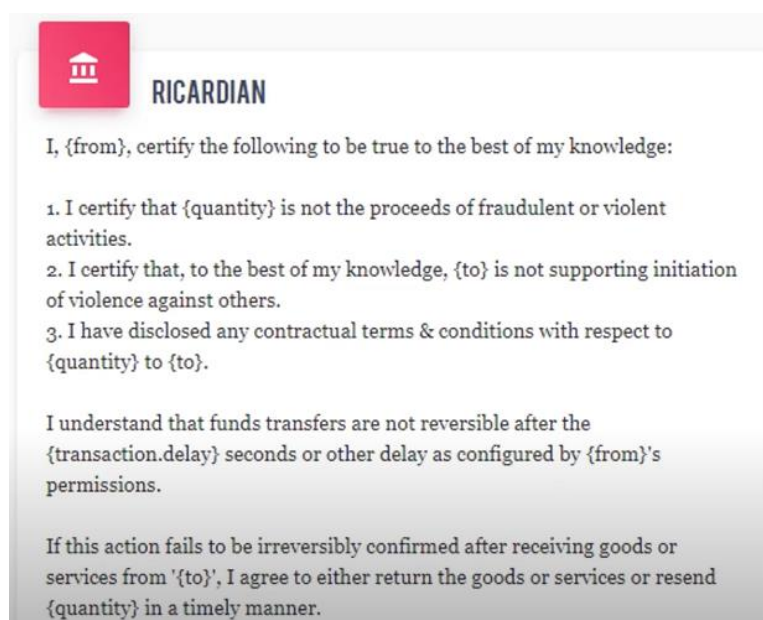


Figura 73: Template di esempio di un contratto ricardiano [43]

## 6.2.2 Comparazione tra Contratto Ricardiano, Contratto Intelligente e Contratto Tradizionale

Alla luce delle precedenti osservazioni è possibile adesso effettuare un'analisi comparativa delle tre forme di contratti esistenti, ovvero il contratto tradizionale, il contratto intelligente e il contratto ricardiano. Tale comparazione ha l'obiettivo di creare una migliore prospettiva delle soluzioni contrattuali esistenti in termini di fattibilità e vantaggi.

In sostanza, il contratto ricardiano è una soluzione ibrida che combina caratteristiche del contratto tradizionale con caratteristiche del contratto intelligente (vedi Tabella 10).

<b>Caratteristiche</b>	<b>Contratto Tradizionale</b>	<b>Contratto Ricardiano</b>	<b>Contratto Intelligente</b>
<b>Scopo</b>	Vincola legalmente le parti nell' adempimento di obblighi	Esegue e automatizza i termini di un accordo con vincolo legale	Esegue e automatizza i termini di un accordo
<b>Validità</b>	È giuridicamente vincolante	È un documento o un accordo giuridicamente vincolante	Non è giuridicamente vincolante
<b>Versatilità</b>	Può essere convertito in contratto Ricardiano	Qualsiasi contratto Ricardiano è anche un contratto intelligente	Non può essere un contratto Ricardiano
<b>Leggibilità</b>	Leggibile dall'uomo	Leggibile sia dalla macchina che dall'uomo	Leggibile dalla macchina e non necessariamente dall'uomo (solo da programmatori e sviluppatori)

**Tabella 10:** Confronto tra le tre tipologie di contratto

## Sitografia

- [1] The Economist, *The Trust Machine*, URL: <https://www.economist.com/leaders/2015/10/31/the-trust-machine>
- [2] Aeron Wright, Primavera De Filippi, *Decentralized Blockchain Technology and the Rise of Lex Cryptographia*, URL: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=2580664](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2580664)
- [3] Mauro Bellini, *Blockchain: cos'è, come funziona e gli ambiti applicativi in Italia*, URL: <https://www.blockchain4innovation.it/esperti/blockchain-perche-e-cosi-importante/>
- [4] *Blockchain. Cosa si nasconde dietro la famigerata tecnologia*, URL: <https://blog.yourtarget.ch/blockchain-cosa-ce-dietro>
- [5] *Bitcoin: A Peer-to-Peer Electronic Cash System*, URL: <https://bitcoin.org/bitcoin.pdf>
- [6] Stephen Evanczuk, *Creare sicurezza nelle applicazioni blockchain*, URL: <https://www.digikey.it/it/articles/build-security-into-blockchain-applications-part-1>
- [7] Rui Zhang, Rui Xue, Ling Liu, *Security and Privacy on Blockchain* URL: <https://arxiv.org/pdf/1903.07602.pdf>
- [8] Vitalik Buterin, *On Public and Private Blockchains*, URL: <https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/>
- [9] Binance Academy, *Private, Public, and Consortium Blockchain – What's the Difference?*, URL: <https://academy.binance.com/en/articles/private-public-and-consortium-blockchains-whats-the-difference>
- [10] Nick Szabo, *Formalizing and Securing Relationships on Public Networks*, URL: <https://firstmonday.org/ojs/index.php/fm/article/view/548/469>
- [11] Nick Szabo, *The Idea of Smart Contracts*, URL: <https://nakamotoinstitute.org/the-idea-of-smart-contracts/>
- [12] Nick Szabo, *Secure Property Titles with Owner Authority*, URL: <https://nakamotoinstitute.org/secure-property-titles/>
- [13] Maria Letizia Perugini, Paolo Dal Checco, *Introduzione Agli Smart Contract (Introduction to Smart Contract)*, URL: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=2729545](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2729545)
- [14] *Initial Coin Offering: una nuova modalità di finanziamento per le imprese*, <http://dspace.unive.it/bitstream/handle/10579/13619/848149-1220164.pdf?sequence=2>
- [15] La legge per tutti, *La stretta di mano vale come contratto?*, URL: [https://www.la leggepertutti.it/123552\\_la-stretta-di-mano-vale-come-contratto](https://www.la leggepertutti.it/123552_la-stretta-di-mano-vale-come-contratto)
- [16] Nicola Di Stefano, Federico Giannone, *Manuale sulla web reputation, dall'identità digitale all'economia della reputazione*, URL: <https://www.corecomplazio.it/images/manuale-web-reputation-defs.pdf>
- [17] Steve Thompson, Michelle Haynes, *The Value of Online Seller Reputation: Evidence from a Price Comparison Site*, URL: <https://onlinelibrary.wiley.com/doi/epdf/10.1002/mde.2777>

- [18] Hema Yoganarasimhan, *The value of Reputation in an Online Freelance Marketplace*, URL: <https://faculty.washington.edu/hemay/reputations.pdf>
- [19] Inside Marketing, *Cos'è il marketing B2B? E come si differenzia dal B2C*, URL: <https://www.insidemarketing.it/glossario/definizione/b2b/>
- [20] Kęstutis RĖKLAITIS, Lina PILELIENĖ, *Principle Differences between B2B and B2C Marketing Communication Processes*, URL: [https://www.researchgate.net/publication/336973845\\_Principle\\_Differences\\_between\\_B2B\\_and\\_B2C\\_Marketing\\_Communication\\_Processes](https://www.researchgate.net/publication/336973845_Principle_Differences_between_B2B_and_B2C_Marketing_Communication_Processes)
- [21] Ismail Sila, *Factors affecting the adoption of B2B e-commerce technologies*, URL: <https://link.springer.com/article/10.1007/s10660-013-9110-7>
- [22] BigCommerce, *B2B Ecommerce Trends*, URL: <https://www.bigcommerce.com/articles/b2b-ecommerce/b2b-ecommerce-trends/#b2b-ecommerce-payments-trends>
- [23] Irene Greguoli Venini, *Il cibo a chilometro zero piace agli italiani*, URL: <https://www.ristorantiweb.com/tendenze/cibo-km0-piace-italiani/>
- [24] Coldiretti, *Cucina a km 0 per 3 italiani su 4 in vacanza*, URL: <https://www.coldiretti.it/economia/cucina-km0-3-italiani-4-vacanza>
- [25] Confcommercio, *Ristorazione, Rapporto annuale 2019*, URL: <https://www.confcommercio.it/documents/20126/0/Rapporto+Ristorazione+2019+Fipe.pdf/27de185c-4d5d-9e60-abc4-7248d254846e?version=1.0&t=1579623545575>
- [26] Fattura in Cloud, *Cosa è il lavoro freelance, quali sono i vantaggi e gli ostacoli in Italia*, URL: <https://www.fattureincloud.it/guida-freelance/cosa-significa-freelance-in-italia/>
- [27] La Stampa, *Che cos'è la Gig Economy?*, URL: <https://www.lastampa.it/cultura/2018/02/24/news/che-cos-e-la-gig-economy-1.33984250>
- [28] Payoneer, *The Payoneer Freelancer Income Survey Global Benchmark Report for Hourly Ra*, URL: <https://www.payoneer.com/downloads/freelancer-income-report-2018.pdf>
- [29] Daniel Huffman, *Cartographic freelancer survey result*, URL: <https://somethingaboutmaps.wordpress.com/2018/03/25/freelancerresults/>
- [30] PactSafe, URL: <https://www.pactsafe.com/>
- [31] Martin Breuer, *Emanate| A decentralized approach to creating a fairer and more dynamic music industry* URL: <https://medium.com/@eosnationbp/emanate-a-decentralized-approach-to-creating-a-fairer-and-more-dynamic-music-industry-ab34b6e37b86>
- [32] Marcela Laskoski, *Emanate: La Revolucion de la Industria Musical*, URL: <https://www.somosinnovacion.lat/emanate-la-revolucion-de-la-industria-musical/>
- [33] AlphaX Digital Services, *Themenbereich smartphone apps*, URL: <https://www.alphax-ds.de/portofolio/heimate-mobile-app/?lang=en>
- [34] William Connel, *Economic Papers, The Economic Impact of Late Payment*, URL: [https://ec.europa.eu/economy\\_finance/publications/economic\\_paper/2014/pdf/ecp531\\_en.pdf](https://ec.europa.eu/economy_finance/publications/economic_paper/2014/pdf/ecp531_en.pdf)

- [35] Martin Garriga, Maxmiliano Arias, Alan De Renzis, *Blockchain and Cryptocurrency: A comparative framework of the main Architectural Drivers*, URL: <https://arxiv.org/abs/1812.08806>
- [36] Trading Education, *EOS vs Ethereum: Is EOS Better Than Ethereum*, URL: <https://trading-education.com/is-eos-better-than-ethereum>
- [37] Haboken, N.J., *Match me if you can: Breakthrough helps spies to shake hands*, URL: <https://www.stevens.edu/news/match-me-if-you-can-cryptographic-breakthrough-helps-spies-shake-hands>
- [38] Brad Westfall, *Leveling Up with React: Redux*, URL: <https://css-tricks.com/learning-react-redux/>
- [39] Ian Grigg, *The Ricardian Contract*, URL: [http://iang.org/papers/ricardian\\_contract.html](http://iang.org/papers/ricardian_contract.html)
- [40] Christopher D. Clack, Vikram A. Bakshi, Lee Braine, *Smart Contract Templates: foundations, design landscape and research directions*, URL: <https://arxiv.org/pdf/1608.00771.pdf>
- [41] Emanuel Regnath, Sebastian Steinhorst, *SmaCoNat: Smart Contract in Natural Language*, URL: [https://www.researchgate.net/publication/328815776\\_SmaCoNat\\_Smart\\_Contracts\\_in\\_Natural\\_Language](https://www.researchgate.net/publication/328815776_SmaCoNat_Smart_Contracts_in_Natural_Language)
- [42] Ian Grigg, *Guida ai contratti Ricardiani*, URL: <http://www.webfunds.org/guide/ricardian.html>
- [43] *What are EOS Ricardian Contracts?*, URL: <https://www.youtube.com/watch?v=qpUsxsYDoll>

## Ringraziamenti

*Giunta al termine di questo lavoro desidero ringraziare ed esprimere la mia riconoscenza nei confronti di tutte le persone che, in modi diversi, mi sono state vicine e hanno permesso e incoraggiato sia i miei studi che la realizzazione e la stesura di questa tesi.*

*I miei più sentiti ringraziamenti vanno al Prof. Fabrizio Lamberti e alla Dott.essa Valentina Gatteschi per avermi concesso l'opportunità di svolgere l'intero progetto di tesi presso Fondazione LINKS e di lavorare su un argomento che è stato di grande interesse per me.*

*Desidero inoltre ringraziare il Dott. Favenza Alfredo per la sua disponibilità e pazienza nel seguirmi e consigliarmi per l'intero svolgimento del lavoro di tesi in azienda.*

*Ringrazio tutti i miei "compagni di viaggio" e in generale tutte le persone che ho incontrato in questo percorso e che mi sono state vicine durante la vita Torinese. È grazie a loro che questi anni saranno per me indimenticabili!*

*Ringrazio le amiche di sempre, Silvia, Roberta e Carmen per aver condiviso tutti i più importanti successi e fallimenti. È confortante sapere che qualcosa di così puro duri nel tempo nonostante i profondi cambiamenti delle nostre vite.*

*Un enorme ringraziamento va alle mie sorelle, Maria e Patrizia, che da sempre sono in grado di capirmi e sostenermi in ogni mia scelta.*

*Infine, il ringraziamento più grande va ai miei genitori che mi hanno insegnato a credere in me stessa e a non mollare mai.*

*A mia Mamma che con enorme pazienza ha saputo guidarmi verso la strada giusta, dandomi consigli preziosi e supportandomi durante ogni sessione universitaria.*

*A mio Papà per avermi sempre sostenuto in tutte le scelte della mia vita ed essere stato un esempio di alta professionalità, dedizione e senso del dovere.*