



Università degli Studi di Salerno

.DIEM

Dipartimento di Ingegneria dell'Informazione ed Elettrica e
Matematica Applicata

Corso di Laurea Magistrale in Ingegneria Informatica

Blockchain

Project Work

Gruppo 8

WP	Cognome e Nome	Matricola	e-mail
1	Lettieri Alessia	0622702324	a.lettieri36@studenti.unisa.it
3	Panico Marco	0622702416	m.panico20@studenti.unisa.it
4	Pasquale Messina	0622702323	p.messina1@studenti.unisa.it

Anno accademico 2024-2025

Sommario

WP1	4
1.1 Attori del sistema	4
Clienti.....	4
Hotel	5
Servizio di prenotazioni	5
Servizio di recensioni	5
1.2 Threat model	5
Impersonator	5
Hotel Reviewer Coercer	6
Hotel Saboteur	6
Reputational Manipulator	6
Platform Manipulator.....	7
Fake Reviewer	7
Feedback Abuser	7
NO-REVS	8
1.3 Proprietà di Sicurezza	8
Confidenzialità	8
Integrità	9
Trasparenza	9
Efficienza/Usabilità	9
1.4 Completeness	10
WP2	11
2.1 Panoramica Generale di Funzionamento	11
2.2 Policy	11
Policy di Accesso e Autenticazione	11
Policy Generali di Visualizzazione Recensioni:	12
Policy sui Contenuti delle Recensioni	12
Policy Antimanipolazione.....	12
2.3 Supposizioni	12
2.4 Prenotazione e soggiorno presso struttura ricettiva	13
2.5 Pubblicazione della Recensione da Parte del Cliente	14
Fase 1 – Autenticazione del Cliente	14
Fase 2 – Redazione della Recensione.....	15
Fase 3 – Archiviazione Decentralizzata e Registrazione On-chain	15
2.6 Modifica ed eliminazione della recezione	16
Fase 1 – Stato iniziale della recensione	16
Fase 2 – Modifica della recensione	16
Fase 3 – Eliminazione della recensione	16
2.8 Eventuale risposta al giudizio rilasciato dal cliente da parte dell’hotel	18
2.9 Incentivazione al rilascio delle recensioni	18
2.10 Specifiche utilizzate	19
Smart contract	19
Decentralized Identifier (DID)	19
Verifiable Data Registry.....	20
Verifiable Credential.....	21
Verifiable Presentation	22
Token.....	22

IPFS.....	22
WP3.....	23
3.1 Confidenzialità.....	23
3.2 Integrità	24
3.3 Trasparenza	25
3.4 Efficienza/Usabilità	26
Confidenzialità.....	28
Integrità	28
Trasparenza	28
Efficienza e usabilità.....	28
3.6 Contrasto degli avversari.....	29
Impersonator	29
Hotel Reviewer Coercer	29
Hotel Saboteur	30
Reputational Manipulator	31
Platform Manipulator.....	31
Fake Reviewer	32
Feedback Abuser	33
NO-REVS	33
WP4.....	35
4.1 Contratti.....	36
EthrDIDRegistry.sol	36
GestioneRecensioni.sol.....	36
MyToken.sol.....	36
4.2 File JavaScript.....	36
Create_vc_prenotazioni.js	36
Create_vc_hotel.js.....	37
Create_vp.js.....	37
Listening.js	38
Modify_rec.js.....	39
Delete_rec.js.....	39
Answer.js.....	39
Viewrec.js	40
Indice delle figure.....	42

WP1

Le recensioni online rappresentano oggi uno strumento fondamentale nel settore alberghiero: influenzano in modo diretto le scelte dei consumatori e l'immagine delle strutture ricettive. Tuttavia, la crescente diffusione di recensioni non autentiche, la possibilità di manipolazione da parte degli operatori del settore e la mancanza di trasparenza nei criteri di visibilità stanno erodendo progressivamente la fiducia degli utenti nei confronti di questi sistemi. Di conseguenza, tali problematiche minano non solo la fiducia degli utenti, ma anche l'efficacia delle piattaforme stesse nel promuovere contenuti autentici e utili.

Questo progetto nasce con l'obiettivo di progettare un **sistema decentralizzato per la gestione di recensioni affidabili**, applicato al contesto degli hotel, in cui la tecnologia blockchain sia impiegata per garantire maggiore trasparenza, verificabilità e resistenza alle manipolazioni, senza sacrificare la privacy degli utenti.

Nel presente Work Package ci proponiamo di definire in modo rigoroso il modello di riferimento su cui si baserà l'intero sistema. In particolare, saranno trattati i seguenti aspetti:

- identificazione degli **attori onesti** che partecipano al sistema e dei loro obiettivi;
- definizione della **funzionalità desiderata**;
- analisi dei **potenziali avversari**, con particolare attenzione alle loro **risorse, capacità e intenzioni malevoli**;
- individuazione delle **proprietà** che si desidera preservare anche in presenza di avversari.

Questa fase non prevede ancora la proposta di una soluzione tecnica, ma costituisce il fondamento teorico su cui si baserà la progettazione nei successivi work package. La qualità del modello qui sviluppato è cruciale per garantire coerenza, solidità e completezza all'intero progetto.

1.1 Attori del sistema

In questo scenario, sono coinvolti diversi attori nel sistema, ciascuno con obiettivi e funzionalità specifiche. Di seguito viene fornito un elenco dettagliato di tali attori e delle loro responsabilità.

Clienti

- **Ruolo:** Soggetto che ha usufruito di un servizio presso una struttura ricettiva e intende rilasciare una recensione.
- **Obiettivo:**
 1. Prenotare la struttura ricettiva attraverso il servizio di prenotazioni
 2. Usufruire del servizio
 3. Condividere, se desidera, la propria esperienza tramite il servizio di recensioni (veritiera) per contribuire alla qualità informativa del sistema.

Hotel

- **Ruolo:** Fornitore del servizio di pernottamento.
- **Obiettivo:**
 1. Fornire il servizio di pernottamento;
 2. Ricevere feedback onesti per migliorare la qualità del servizio e aumentare la fiducia nei clienti;
 3. Rispondere pubblicamente in modo tracciabile, senza possibilità di alterare i contenuti inseriti dagli utenti.

Servizio di prenotazioni

- **Ruolo:** Fornitore del servizio di prenotazioni dove i clienti prenotano il soggiorno in hotel

Servizio di recensioni

- **Ruolo:** Fornire servizio di recensioni
- **Obiettivo:**
 1. Permettere al cliente di recensire un hotel
 2. Permettere al cliente di revocare o modificare le proprie recensioni
 3. Adottare policy trasparenti per la gestione delle recensioni.

1.2 Threat model

Nel progettare un sistema di identificazione basato su recensioni, è importante considerare i diversi avversari che potrebbero tentare di attaccarlo.

Impersonator

- **Tipologia:** Attivo
- **Descrizione:** Individuo che cerca di impersonare un utente legittimo.
- **Risorse:** Possiede una discreta potenza di calcolo e accesso a informazioni personali dettagliate.
- **Attacchi:**
 - **Impersonificazione:** Utilizzare account falso per fingersi un cliente che ha realmente soggiornato.

Hotel Reviewer Coercer

- **Tipologia:** Attivo
- **Descrizione:** Struttura ricettiva che sollecita o esercita pressioni sui clienti affinché pubblichino recensioni positive o modifichino recensioni esistenti.
- **Risorse:** Accesso diretto ai clienti, strumenti di contatto, possibilità di offrire incentivi o minacciare conseguenze.
- **Attacchi:**
 - **Coercizione Post-Servizio:** Pressione psicologica o incentivi in cambio di recensioni positive.
 - **Modifica Forzata:** Richiesta esplicita di modifica di una recensione negativa già pubblicata.

Hotel Saboteur

- **Tipologia:** Attivo
- **Descrizione:** Hotel che pubblica recensioni false e negative contro i concorrenti.
- **Risorse:** Rete di account falsi o utenti pagati, tool di automazione.
- **Attacchi:**
 - **Attacchi alla Reputazione:** Recensioni negative non verificate e dettagliate contro hotel concorrenti.
 - **Flooding Coordinato:** Pubblicazione massiva di contenuti negativi in un breve lasso di tempo.

Reputational Manipulator

- **Tipologia:** Attivo
- **Descrizione:** Hotel che tenta di alterare artificiosamente la percezione pubblica del proprio profilo online, agendo sia per nascondere feedback negativi reali, sia per gonfiare la propria reputazione mediante recensioni positive fraudolente.
- **Risorse:**
 - Accesso a informazioni sulle policy della piattaforma.
 - Connessioni con reti di account falsi o broker di recensioni.
 - Budget economico per acquistare recensioni.
- **Attacchi:**
 - **Rimozione di Feedback Negativi:** Utilizzo delle procedure ufficiali della piattaforma (es. segnalazioni) per ottenere la cancellazione di recensioni autentiche ma negative.

- **Saturazione Positiva:** Generazione o acquisto massivo di recensioni positive con l'obiettivo di sovrastare e diluire l'effetto delle recensioni negative.
- **Oscuramento Algoritmico:** Sfruttamento dei meccanismi di ranking per rendere meno visibili recensioni critiche (es. creando molte recensioni nuove per spingere in basso quelle vecchie).

Platform Manipulator

- **Tipologia:** Attivo
- **Descrizione:** Operatore interno della piattaforma che altera recensioni valide.
- **Risorse:** Accesso ai sistemi di backend, autorizzazioni elevate, controllo sui contenuti pubblicati.
- **Attacchi:**
 - **Cancellazione Illegittima:** Rimozione di recensioni vere senza giustificazione o per conto di un hotel.
 - **Filtro Opaque:** Manipolazione del ranking o della visibilità delle recensioni selezionate.

Fake Reviewer

- **Tipologia:** Attivo
- **Descrizione:** Utente che pubblica recensioni su strutture che non ha realmente visitato.
- **Risorse:** Accesso alla piattaforma.
- **Attacchi:**
 - **Recensioni Fittizie:** Scrittura di contenuti ingannevoli per danneggiare o favorire un servizio.
 - **Recensioni su Commissione:** Attività incentivata economicamente da hotel.

Feedback Abuser

- **Tipologia:** Attivo
- **Descrizione:** Cliente che ha realmente soggiornato in una struttura, ma tenta di pubblicare più recensioni per la stessa prenotazione o modifica periodicamente la recensione effettuata, per influenzare sulla percezione dell'hotel.
- **Risorse:** Accesso alla piattaforma come cliente autenticato; presenza di una prenotazione valida e verificata.
- **Attacchi:**

- **Recensioni Multiple:** Inserimento di più recensioni per la stessa esperienza, per aumentare l'impatto sul punteggio dell'hotel.
- **Modifica Aggressiva:** Uso ripetuto della funzione di modifica per alterare continuamente il contenuto, inducendo confusione o manipolando la percezione nel tempo.

NO-REVS

- **Tipologia:** Passivo
- **Descrizione:** Opppositori convinti che qualsiasi tentativo di regolamentazione dei sistemi di feedback sia destinato al fallimento per via della corruzione intrinseca delle piattaforme centralizzate.
- **Risorse:** Spiccate abilità comunicative, capacità di influenzare l'opinione pubblica.
- **Attacchi:**
 - **Campagne di Disinformazione:** Diffusione di notizie negative per ostacolare l'adozione del sistema.

1.3 Proprietà di Sicurezza

Le principali proprietà di sicurezza che il sistema deve possedere per garantire un ambiente sicuro e affidabile sono analizzate in base ai quattro pilastri fondamentali: confidenzialità, integrità, trasparenza ed efficienza.

Confidenzialità:

- **C.1:** Le identità dei clienti devono poter essere mantenute riservate dal servizio di recensioni, impedendo che gli hotel o altri utenti possano risalirvi direttamente.
- **C.2:** Solo il cliente che ha effettivamente soggiornato presso una struttura deve poter inserire, modificare o revocare una recensione relativa a quel soggiorno. Il sistema deve impedire che utenti diversi possano impersonare il cliente.
- **C.3:** Il servizio di recensioni deve evitare di esporre dati non essenziali (es. dettagli identificativi) che possano ricondurre al cliente.
- **C.4:** L'elenco delle recensioni completo (anche di recensioni rimosse) deve essere accessibile solo alla struttura alberghiera interessata, non a utenti esterni o ad altri hotel.
- **C.5:** Un cliente che recensisce due strutture distinte non deve poter essere riconosciuto tramite la recensione come la stessa persona da soggetti terzi o dalle strutture stesse.

Integrità

- **I.1:** Solo i clienti che hanno effettuato e concluso una prenotazione reale devono essere abilitati a pubblicare recensioni. Il sistema deve garantire che tale diritto sia verificabile e non trasferibile ad altri utenti.
- **I.2:** Le recensioni devono essere immutabili una volta pubblicate: ogni eventuale modifica della recensione o risposta dell'hotel deve essere tracciata in modo indelebile, mantenendo accessibile l'elenco delle versioni.
- **I.3:** La rimozione o eliminazione di una recensione non deve comportare la perdita di informazioni: la cronologia completa degli eventi (inserimento, modifica, rimozione) deve essere conservata in modo sicuro e inalterabile.
- **I.4:** Il sistema deve impedire che una recensione possa essere modificata in modo arbitrario o illimitato. Devono esistere regole chiare (tempi, condizioni, limiti) che vincolano l'azione e garantiscono la coerenza con la versione originale.

Trasparenza:

- **T.1:** Il sistema deve rendere pubblici i criteri con cui le recensioni vengono moderate, rese visibili, ordinate e, eventualmente, rimosse.
- **T.2:** Le regole che stabiliscono chi può recensire, in quali casi, e con quali vincoli temporali, devono essere esplicite e consultabili dagli utenti.
- **T.3:** Gli algoritmi e i protocolli utilizzati per la gestione delle recensioni devono essere pubblicamente noti e verificabili.

Efficienza/Usabilità:

- **E.1:** Il sistema di inserimento della recensione deve essere rapido, semplice e fruibile senza richiedere competenze tecniche.
- **E.2:** Il processo di verifica dell'effettivo soggiorno del cliente deve essere eseguito in tempi brevi e non ostacolare l'esperienza utente.
- **E.3:** Il servizio deve permettere la pubblicazione, modifica o rimozione delle recensioni entro tempistiche che ne preservino l'utilità informativa e la qualità complessiva del sistema.
- **E.4:** Il sistema di recensioni deve prevedere policy chiare e accessibili che guidino l'utente nel processo di rilascio, indicando cosa è consentito fare, quando e in che modo. Tali policy devono agevolare la comprensione del funzionamento del sistema anche da parte di utenti non esperti.
- **E.5:** Il sistema deve incentivare attivamente il rilascio delle recensioni da parte degli utenti che hanno effettivamente soggiornato nella struttura.

1.4 Completeness

Il sistema proposto mira a garantire l'accesso al rilascio delle recensioni solo a coloro che hanno realmente soggiornato in una struttura ricettiva, seguendo determinate politiche di accesso.

All'istante T0:

Il cliente effettua una prenotazione tramite il Servizio di Prenotazioni, a seguito di una iscrizione al servizio, selezionando la struttura ricettiva e fornendo i dati necessari.

All'istante T1:

Il Servizio di Prenotazioni notifica l'avvenuta prenotazione all'hotel, rendendolo a conoscenza del cliente e del periodo di soggiorno.

All'istante T2:

Il cliente usufruisce del servizio di pernottamento presso l'hotel prenotato, completando il soggiorno.

All'istante T3:

Viene effettuato il controllo per verificare se cliente abbia effettivamente soggiornato e che quindi sia idoneo a recensire

All'istante T4:

Il cliente accede al Servizio di Recensioni e inserisce una recensione associata alla prenotazione verificata.

All'istante T5:

Il cliente può eventualmente modificare o rimuovere la recensione.

All'istante T6:

L'hotel può pubblicare una risposta pubblica alla recensione.

All'istante T7:

Il sistema di recensioni adotta le strategie per l'ordinamento delle recensioni opportuno.

WP2

In questo capitolo ci concentreremo sulla presentazione di una soluzione che risponde al modello identificato nel Work Package 1 (WP1). L'obiettivo è quello di proporre un sistema che riesca a raggiungere un ragionevole compromesso tra efficienza, trasparenza, confidenzialità e sicurezza.

Per la progettazione del sistema abbiamo utilizzato una blockchain permissionless in modo che ogni utente possa accedere al servizio di recensione senza dover effettuare l'accesso e di non far conoscere la sua identità ad altri utenti.

2.1 Panoramica Generale di Funzionamento

Le strutture che aderiscono al sistema sono convenzionate sia con la piattaforma di prenotazione sia con quella delle recensioni. Il cliente si iscrive alla piattaforma di prenotazione, effettua la prenotazione presso una struttura e usufruisce del soggiorno. Al termine dell'esperienza, ha la possibilità di lasciare una recensione sulla piattaforma dedicata. La recensione può essere rilasciata solo dopo almeno 12 ore dal checkout, per evitare pressioni o influenze dirette da parte della struttura.

Solo chi ha effettivamente prenotato e soggiornato può scrivere una recensione, che viene pubblicata in forma anonima per tutelarne l'identità. Dopo l'invio, la recensione può essere modificata entro 24 ore, mentre la cancellazione rimane sempre possibile. Al momento della compilazione, il cliente deve indicare chiaramente se l'esperienza è stata positiva o negativa, in modo da fornire un giudizio utile sia agli altri utenti sia alla struttura.

Per incentivare l'uso della piattaforma di recensioni, al termine della pubblicazione il sistema assegna un premio cumulabile (non cedibile ad altri utenti), che potrà essere convertito in un buono sconto sulla piattaforma di prenotazione una volta raggiunto un numero prestabilito di premi.

La consultazione delle recensioni è libera e accessibile a tutti, consentendo di esplorare le opinioni pubblicate e confrontare le esperienze di altri clienti. Le recensioni eventualmente eliminate dagli utenti potranno essere rese nuovamente visibili solo alla struttura oggetto della recensione, ed esclusivamente su esplicita richiesta. Inoltre, le strutture hanno la possibilità di rispondere pubblicamente alle recensioni ricevute, favorendo un dialogo trasparente con i clienti e consentendo chiarimenti o ringraziamenti ufficiali.

2.2 Policy

Policy di Accesso e Autenticazione

- *Recensioni solo da clienti verificati:* Solo clienti con prenotazione verificata e soggiorno concluso possono pubblicare recensioni.
- *Accesso individuale e non trasferibile:* L'accesso alla pubblicazione è legato all'identità crittografata dell'utente e non può essere delegato.
- *Validità temporale:* L'utente può lasciare una recensione dopo 12h dal rilascio della VC.

Policy Generali di Visualizzazione Recensioni:

- *Ordinamento dinamico*: Le recensioni vengono ordinate in base alla prevalenza qualitativa, se le recensioni positive superano numericamente le negative, vengono mostrate prima le positive. Viceversa, in caso di predominanza di recensioni negative. In caso di parità, viene data priorità alle recensioni positive per non penalizzare l'hotel.
- *Flag dell'esperienza non modificabile*: Una recensione è marcata come positiva o negativa all'inserimento e il flag non può essere cambiato, per garantire coerenza e prevenire coercizione o corruzione post-servizio. L'esperienza è una e non può cambiare.
- *Visualizzazione recensioni rimosse*: Solo l'hotel oggetto della recensione può vedere le recensioni rimosse se fa richiesta alla piattaforma di recensione.

Policy sui Contenuti delle Recensioni

- *Requisito minimo di lunghezza*: Le recensioni devono contenere almeno 20 caratteri e al massimo 200 per essere pubblicate.
- *Singolarità della recensione per prenotazione*: Ogni prenotazione consente l'inserimento di una sola recensione pubblicabile.
- *Modifica vincolata nel tempo*: Le recensioni possono essere modificate una sola volta entro 24h dalla pubblicazione.
- *Cancellazione possibile, ma tracciata*: La rimozione volontaria della recensione da parte del cliente è ammessa ma resta accessibile alla struttura interessata.

Policy Antimanipolazione

- *Nessuna cancellazione da parte della piattaforma o degli hotel*: Nessun attore (inclusi amministratori della piattaforma) può cancellare, modificare o nascondere contenuti validamente pubblicati.
- *Tracciabilità completa delle interazioni*: Ogni inserimento, modifica o risposta viene tracciato su blockchain con immutabilità e marcatura temporale.
- *Limitazioni alla frequenza di pubblicazione*: Un cliente può recensire solo una struttura per ogni prenotazione e non può pubblicare recensioni multiple sullo stesso soggiorno.

2.3 Supposizioni

Si assume che il servizio di prenotazione sia affidabile e che l'hotel sia convenzionato sia con la piattaforma di prenotazione che con quella di recensioni.

I clienti dispongano di un wallet digitale per conservare e presentare le proprie credenziali.

Si suppone che i clienti vogliano rimanere anonimi per preservare la propria identità ed evitare di essere riconosciuti dall'hotel dopo la pubblicazione della recensione, così da non temere possibili ripercussioni.

Il sistema di recensioni conserva in modo sicuro l'ID e il salt associato all'interno di un database.

Si suppone che le policy del sistema di recensioni siano pubbliche e consultabili, in modo da garantire agli utenti piena trasparenza sulle regole che disciplinano la pubblicazione, la visualizzazione e la gestione delle recensioni.

2.4 Prenotazione e soggiorno presso struttura ricettiva

Il cliente si iscrive a un servizio di prenotazione se non ha mai utilizzato quel servizio. Dopo essersi autenticato, effettua una prenotazione (Figura 1-1) presso una struttura ricettiva convenzionata al servizio di prenotazione (Figura 1-0).

Una volta completata la prenotazione, il servizio di prenotazione rilascia al cliente una Verifiable Credential (VC) (Figura 1-2), utilizzando il DID del cliente come identificativo del soggetto (subject). Questa credenziale è firmata digitalmente dal servizio di prenotazione, che agisce in qualità di issuer, e viene archiviata nel wallet del cliente. La VC di prenotazione contiene diversi claim significativi, tra cui:

- l'identificativo pubblico del cliente (DID),
- il periodo di inizio e fine prenotazione,
- l'indirizzo della struttura prenotata,
- il numero di notti prenotate,
- il numero delle persone nella stanza prenotata,
- Data di rilascio della credenziale.

Al termine del soggiorno (Figura 1-3), la struttura ricettiva rilascia una seconda Verifiable Credential al cliente (Figura 1-4). Anche questa viene firmata digitalmente e archiviata nel wallet dell'utente. La VC di soggiorno attesta l'effettiva permanenza presso la struttura e contiene:

- l'identificativo pubblico del hotel (DID),
- l'identificativo pubblico del cliente (DID),
- la data effettiva di check-in e check-out,
- l'indirizzo della struttura prenotata
- il numero di notti trascorse,
- il numero di persone nella stanza,
- Data di rilascio della credenziale.

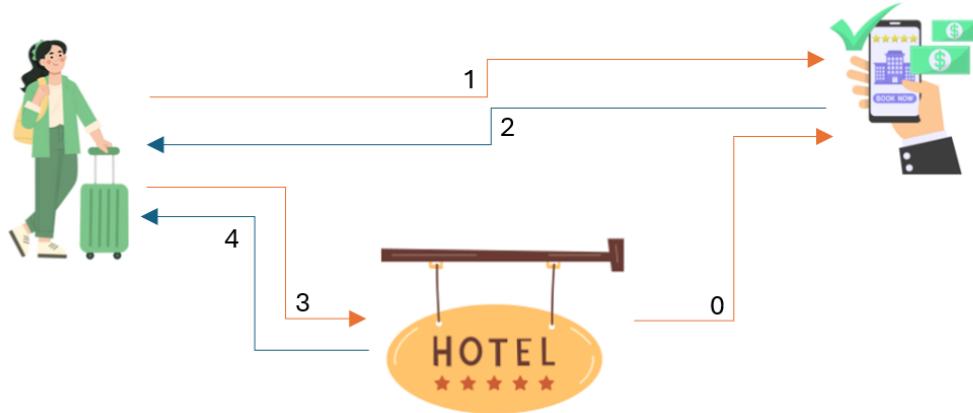


Figura 1: Prenotazione e soggiorno presso struttura ricettiva. In blu le interazioni da implementare

2.5 Pubblicazione della Recensione da Parte del Cliente

Al termine del soggiorno presso una struttura ricettiva, il cliente ha la possibilità di pubblicare una recensione attraverso una piattaforma digitale dedicata, accedendo mediante autenticazione con il proprio wallet digitale. Il wallet contiene le Verifiable Credential (VC), precedentemente rilasciate dal servizio di prenotazione e dalla struttura ospitante, che attestano l'effettiva esperienza di soggiorno del cliente.

Fase 1 – Autenticazione del Cliente

Il cliente accede alla piattaforma utilizzando il proprio wallet digitale. Durante questa fase, il cliente genera una Verifiable Presentation (VP) e la presenta al sistema di recensioni. Il sistema si occupa quindi di verificare la validità della VP e delle Verifiable Credential (VC) in essa contenute.

Il sistema di recensione quando gli viene presentata la VP del cliente, effettua un controllo sulla firma della VP (se issuer della VP è il colui che vuole rilasciare la recensione). Successivamente vengono estratte le VC e vengono effettuati i seguenti controlli:

- Sulla VC rilasciata dal servizio di prenotazione:
 - Verifica che l'issuer corrisponda alla piattaforma di prenotazione, garantendo l'autenticità della credenziale.
 - Controllo che il subject (cioè il cliente a cui è stata rilasciata la VC) coincida con l'issuer della VP.
- VC rilasciata dall'hotel:
 - Verifica che l'issuer corrisponda all'hotel, accertando l'autenticità della credenziale.
 - Controllo che il subject (cliente a cui viene rilasciata la VC) coincida con l'issuer della VP
 - Controllo della validità temporale della VC, 12h dal rilascio della VC, per assicurare l'idoneità del cliente alla pubblicazione della recensione.
 - Verifica dell'unicità: viene accertato che la stessa VC non sia già stata utilizzata per pubblicare un'altra recensione, impedendo duplicazioni e garantendo l'integrità del processo.

- Verifica di coerenza tra le informazioni delle VC:
 - Controllo della corrispondenza tra le date di check-in e check-out presenti nelle diverse VC, e altri campi.
 - Verifica che l'hotel dichiarato nella VC di prenotazione coincida con l'hotel per cui si sta effettuando la recensione.
 - Verifica che la data di rilascio della VC del sistema di prenotazioni sia antecedente alla data di rilascio della VC rilasciata dall'hotel.

Fase 2 – Redazione della Recensione

Una volta superati i controlli iniziali, il cliente procede alla scrittura della recensione. La piattaforma applica dei **vincoli di forma** per standardizzare i contenuti ed evitare spam:

- La recensione deve contenere un **numero minimo di 20 caratteri**.
- La lunghezza massima ammessa è di **200 caratteri**.

Solo le recensioni che rispettano tali vincoli vengono accettate per la fase successiva.

Fase 3 – Archiviazione Decentralizzata e Registrazione On-chain

Superati tutti i controlli, la recensione ed il rispettivo sentiment (esperienza positiva/negativa) vengono salvati su IPFS (InterPlanetary File System), un sistema di archiviazione distribuito che garantisce l'immutabilità e la tracciabilità del contenuto. L'operazione genera un Content Identifier (CID) univoco, che identifica in modo permanente il contenuto della recensione.

Il sistema, tramite l'attivazione di uno smart contract, provvede alla registrazione sulla blockchain delle informazioni chiave associate alla recensione pubblicata. In particolare, lo smart contract memorizza:

- Il CID IPFS associato alla recensione.
- Il sentiment dell'esperienza riportata (ad esempio, "positiva" o "negativa").
- Uno stato che rappresenta il ciclo di vita della recensione (INSERITA, MODIFICATA, ELIMINATA).
- L'hash di un identificativo univoco associato alla VC del cliente, ottenuto calcolando l'hash della concatenazione tra l'ID della VC e un salt (univoco).
- Timestamp con l'orario del rilascio della recensione.

Il **salt**, essendo un dato sensibile, non viene memorizzato sulla blockchain. Viene invece salvato in modo sicuro all'interno del **database** della piattaforma di recensioni. Questo accorgimento garantisce la protezione della privacy del cliente, impedendo il collegamento diretto tra l'hash registrato on-chain e la identità del cliente, pur mantenendo la possibilità di verifica in caso di audit.

2.6 Modifica ed eliminazione della recezione

Fase 1 – Stato iniziale della recensione

Una volta che la recensione è stata correttamente salvata su IPFS, il sistema, tramite smart contract, registra sulla blockchain lo stato corrispondente, impostandolo a **INSERITA**, che indica l'avvenuto inserimento della recensione. Questo stato rappresenta la condizione iniziale che consente eventualmente una futura modifica o eliminazione da parte del cliente.

Fase 2 – Modifica della recensione

Se il cliente intende modificare la propria recensione, può farlo una sola volta entro le 24 ore. Per avviare il processo, deve autenticarsi nuovamente presentando la stessa Verifiable Credential (VP) utilizzata per il primo inserimento. Il sistema di recensioni esegue nuovamente tutti i controlli sulla validità della VP, comprese le firme digitali degli emittenti delle VC. Una volta confermata la validità della credenziale, il sistema procede al recupero dell'identificativo contenuto nella VC e verifica nei propri database la presenza dell'ID e del salt associato. Attraverso la combinazione dell'ID della VC e del relativo salt viene ricalcolato l'hash, che consente l'interrogazione dello smart contract per ottenere il CID corrispondente alla recensione originale.

A questo punto, il sistema verifica lo stato attuale della recensione. Se esso risulta ancora **INSERITA** e non sono passate ancora 24 ore, la modifica è consentita. Il cliente può quindi scrivere una nuova versione della recensione, che sarà sottoposta agli stessi vincoli formali previsti per l'inserimento iniziale (ossia una lunghezza minima di 20 caratteri e massima di 200). La recensione modificata viene salvata nuovamente su IPFS, generando un nuovo CID. Il sistema aggiorna quindi lo stato sulla blockchain, impostandolo a **MODIFICATA**. A partire da questo momento, non saranno più consentite ulteriori modifiche alla recensione.

⚠️ Nota importante: Il campo sull'esperienza del cliente (positiva o negativa) non può essere modificato. L'esperienza è considerata un evento soggettivo e non alterabile, a meno di comportamenti illeciti. Per questo motivo, il campo è immutabile sin dal primo inserimento.

Fase 3 – Eliminazione della recensione

Nel caso in cui il cliente desideri eliminare la propria recensione, la procedura è analoga a quella prevista per la modifica. Il cliente presenta la VP al sistema, che ne verifica la validità e procede

come descritto sopra: recupera l'ID, cerca il salt associato, ricalcola l'hash e ottiene tramite smart contract il CID della recensione. Viene quindi controllato lo stato della recensione: se risulta INSERITA, il sistema fa riferimento al CID originale; se invece risulta MODIFICATA, viene considerato il CID aggiornato. In entrambi i casi, lo smart contract aggiorna lo stato a CANCELLATA, indicando che la recensione è stata eliminata.

La recensione eliminata non sarà più visualizzabile sulla piattaforma, ma resterà comunque registrata sulla blockchain. Tale scelta è coerente con la natura immutabile della blockchain, che garantisce la trasparenza e la tracciabilità dell'intero processo, anche in presenza di una cancellazione.

Le recensioni eliminate dagli utenti potranno essere rese nuovamente accessibili esclusivamente su esplicita richiesta della struttura alberghiera recensita. In fase di richiesta, viene eseguita una procedura di verifica (challenge) durante la quale l'hotel deve dimostrare il possesso del proprio DID (Decentralized Identifier). Solo a seguito del superamento positivo di tale verifica, la struttura potrà accedere alle recensioni eliminate.

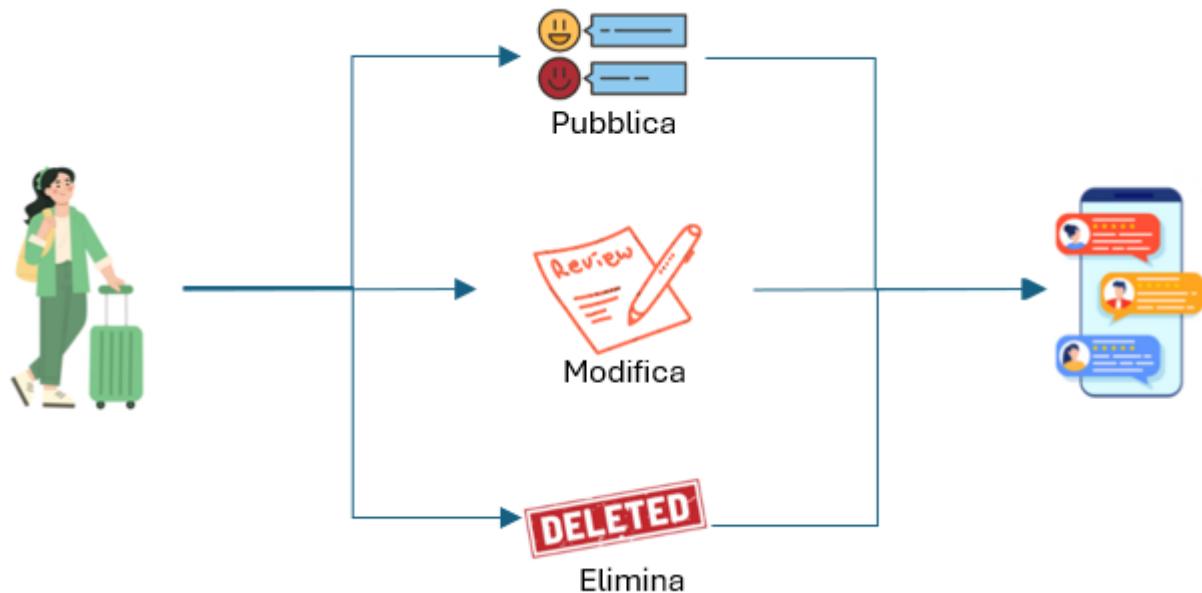


Figura 2: Operazioni sulle recensioni. Il cliente inserisce una recensione dopo aver superato i controlli di validità sul sito recensioni. Successivamente può eliminare o modificare la recensione. Dopo ogni operazione il sito aggiorna l'ordinamento delle recensioni.

2.7 Ordinamento delle recensioni

Il sistema adotta un criterio di ordinamento basato sull'esperienza complessiva verificata degli utenti, evitando meccanismi di rimozione manuale o segnalazione diretta. Ogni recensione viene valutata in base alla sua coerenza con le altre: se risulta estremamente discordante rispetto al comportamento aggregato (per esempio troppo positiva o negativa senza riscontro nei dati), viene automaticamente collocata in fondo alla lista, riducendone la visibilità.

Questo approccio tutela sia gli hotel onesti da eventuali giudizi distorti o ingiusti, sia l'affidabilità generale del sistema, impedendo manipolazioni da parte di clienti o strutture. Inoltre, l'ordinamento

riflette l'esperienza prevalente: se la maggior parte delle recensioni è positiva, quelle coerenti con questa esperienza vengono mostrate per prime, perché è più probabile che un nuovo cliente viva un soggiorno simile. Al contrario, se prevalgono le recensioni negative, queste guadagnano visibilità, segnalando un'alta probabilità di disservizi. In caso di parità, viene data priorità alle recensioni positive per non penalizzare l'hotel.

La differenza tra il numero di recensioni positive e negative diventa quindi un indicatore significativo della qualità percepita, permettendo al sistema di distribuire le recensioni in modo dinamico e informativo, offrendo agli utenti un quadro realistico e aggiornato della struttura.

2.8 Eventuale risposta al giudizio rilasciato dal cliente da parte dell'hotel

Dopo la pubblicazione di una recensione da parte del cliente, la struttura ricettiva oggetto del giudizio ha la possibilità di fornire una risposta pubblica. Questa funzionalità consente all'hotel di replicare in modo trasparente alla valutazione ricevuta, offrendo chiarimenti, ringraziamenti o approfondimenti utili a contestualizzare l'esperienza, contribuendo così a un confronto costruttivo con il cliente e alla trasparenza informativa per gli altri utenti della piattaforma.

Ogni recensione può ricevere una sola replica da parte dell'hotel, e tale risposta è associata in modo univoco al CID IPFS della recensione a cui si riferisce, sia che la recensione sia nello stato iniziale di inserimento, sia che sia stata successivamente modificata. Prima di accettare una risposta, il sistema effettua una serie di verifiche fondamentali: controlla che la recensione esista effettivamente, che sia correttamente collegata all'indirizzo dell'hotel che intende rispondere, e che non sia già stata registrata una risposta per quello specifico contenuto.

Una volta superati i controlli, la risposta viene salvata su IPFS e il suo CID viene registrato on-chain tramite lo smart contract, insieme all'indirizzo dell'hotel e al timestamp di pubblicazione. La risposta assume così valore pubblico e immutabile, non è prevista alcuna possibilità di modifica o cancellazione, nel rispetto dei principi di integrità, trasparenza e non ripudio. Nel caso in cui un cliente cancelli una recensione, la risposta ad essa collegata non sarà più visibile agli utenti.

In questo modo, la piattaforma tutela il diritto di replica della struttura.

2.9 Incentivazione al rilascio delle recensioni

Al fine di promuovere un utilizzo attivo e consapevole della piattaforma di recensioni, è stato previsto un meccanismo di incentivazione che premi i clienti per il rilascio di feedback. In particolare, al termine del processo di pubblicazione della recensione il sistema assegna automaticamente al cliente una frazione di token pari a un decimo (1/10).

Tali token (utilizziamo standard ERC-20), non scambiabili, vengono accumulati direttamente all'interno del wallet dell'utente e rappresentano un credito virtuale che certifica la partecipazione attiva del cliente alla piattaforma. Al raggiungimento di un token completo, quindi al rilascio di dieci recensioni valide, l'utente ha la possibilità di riscattare un buono sconto da utilizzare per una futura prenotazione presso una delle strutture convenzionate con la piattaforma.

Questo sistema di rewarding, basato su logiche di accumulo progressivo e legato a una soglia minima di attività, consente di stimolare la fidelizzazione dell’utente, promuovere l’emissione di recensioni trasparenti e verificabili, e rafforzare il valore reputazionale delle strutture ricettive presenti nel circuito.

2.10 Specifiche utilizzate

Smart contract

Uno smart contract è un programma auto-esecutivo memorizzato sulla blockchain, che applica automaticamente i termini di un accordo quando si verificano determinate condizioni. A differenza dei contratti tradizionali, gli smart contract non richiedono intermediari: il codice è pubblico, trasparente, immutabile e accessibile da tutti i partecipanti alla rete. Ogni nodo della blockchain esegue indipendentemente il contratto, garantendo coerenza e affidabilità.

Gli smart contract sono deterministicici e una volta distribuiti non possono essere modificati: per aggiornare il codice è necessario distribuire un nuovo contratto. Inoltre, il loro comportamento dipende completamente dalla logica programmata, esponendoli a rischi se il codice è scritto in modo errato.

Su Ethereum, gli smart contract vengono eseguiti in modo coerente e decentralizzato grazie alla Ethereum Virtual Machine. Scriviamo gli smart contract in Solidity, un linguaggio orientato agli oggetti pensato appositamente per la blockchain. In Solidity definiamo variabili di stato memorizzate sulla blockchain e funzioni con diversi livelli di visibilità che descrivono la logica dell'accordo, rendendolo automatico e verificabile da tutti. Per interagire con lo smart contract utilizziamo JavaScript e il test in locale usiamo Ganache, che simula una blockchain privata e permette di verificare facilmente il funzionamento del contratto.

Decentralized Identifier (DID)

Un Decentralized Identifier (DID) identifica univocamente un soggetto (non solo individui) ed è composto da:

- Uniform Resource Identifier
- Identificatore specifico per un metodo DID
- Identificatore specifico del metodo per il DID

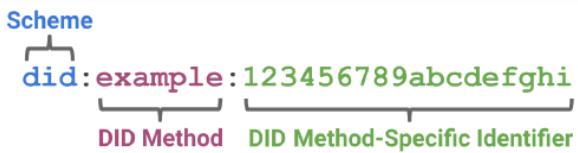
did:example:123456789abcdefghi
{ "@context": ["https://www.w3.org/ns/did/v1", "https://w3id.org/security/suites/ed25519-2020/v1"], "id": "did:example:123456789abcdefghi", "authentication": [{ "id": "did:example:123456789abcdefghi#keys-1", "type": "Ed25519VerificationKey2020", "controller": "did:example:123456789abcdefghi", "publicKeyMultibase": "zH3C2AVvLMv6gmMNam3uVAjZp..." }] }

Figura 3: Struttura DID

I componenti chiave sono:

- Namespace: Identifica il metodo utilizzato nel DID (es. ethr, sov, brcr, etc).

- Specifiche del metodo: Documento che definisce regole e operazioni per i DID che utilizzano questo metodo specificando come creare risolvere aggiornare o disattivare un DID.



Ogni DID risolve un DID document, un file JSON-LD che contiene varie informazioni sul soggetto del DID. Questo documento comprende la chiave pubblica, parametri di autenticazione, timestamp e metadati aggiuntivi.

Un soggetto può provare di possedere il DID sfruttando la chiave privata che corrisponde alla chiave pubblica inclusa nel documento. Per effettuare questo controllo il verifier accederà al documento condiviso tramite il Verifiable Data Registry (VDR).

Verifiable Data Registry

Nel contesto della nostra architettura, abbiamo scelto di implementare il Verifiable Data Registry (VDR) come una blockchain permissionless, in modo da garantire trasparenza, immutabilità e accesso aperto. Questo approccio consente a chiunque di generare e registrare il proprio Decentralized Identifier (DID) senza necessità di autorizzazione preventiva, nel pieno rispetto dei principi dell'identità decentralizzata.

Per ogni DID creato, è possibile associare un DID Document, contenente le informazioni necessarie per verificarne l'autenticità, come chiavi pubbliche, delegati e service endpoints. I DID Document sono gestiti direttamente all'interno della blockchain tramite un contratto intelligente Ethereum, che funge da registro on-chain per le operazioni di aggiornamento e gestione delle identità.

Come implementazione, abbiamo adottato il contratto ethr-did-registry, una soluzione matura e conforme allo standard W3C did:ethr. Questo smart contract consente al proprietario di un'identità ethr-did di aggiornare dinamicamente gli attributi associati al proprio DID Document. Inoltre, il contratto espone una API accessibile via JavaScript, facilitando l'integrazione con applicazioni decentralizzate (dApp) e agenti DID.

Questa scelta ci permette di garantire una gestione sicura, trasparente e interoperabile delle identità digitali, favorendo la portabilità e la verificabilità delle credenziali in contesti eterogenei.

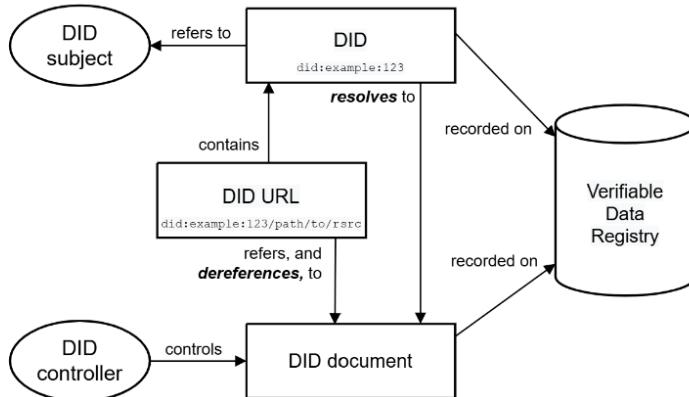


Figura 4: Funzionamento protocollo DID

Verifiable Credential

Una Verifiable Credential (VC) è una struttura dati interoperabile, in grado di rappresentare affermazioni e proprietà di un soggetto titolare di identità (DID Subject) in modo critto-verificabile e resistente alle manomissioni.

Soggetti della VC:

- Holder: Entità che può avere una o più VC e generare da esse le Verifiable Presentation. L'holder è quasi sempre il subject delle VC che possiede. L'holder salva le VC nel proprio wallet.
- Issuer: Entità che afferma uno o più claim riguardanti un subject, crea le VC a partire da questi claim, e trasmette queste VC all'holder.
- Subject: Entità su cui vengono fatte delle affermazioni.
- Verifier: Entità riceve una o più VC, anche all'interno di VP, per controllarle.
- Verifiable data registry: Un Verifiable Data Registry (VDR) è un sistema che svolge un ruolo centrale nella gestione di dati affidabili all'interno di un ecosistema decentralizzato. In particolare, un VDR media la creazione, la verifica e la gestione di identificatori, materiali di verifica, schemi di credenziali verificabili, registri di revoca e altre informazioni strutturali essenziali per il funzionamento delle credenziali verificabili. A seconda della configurazione, il VDR può supportare anche identificatori correlabili, qualora sia necessario associare più dati allo stesso soggetto. In alcuni casi, come nei registri di UUID, il VDR può funzionare semplicemente come spazio dei nomi (namespace) per identificatori.

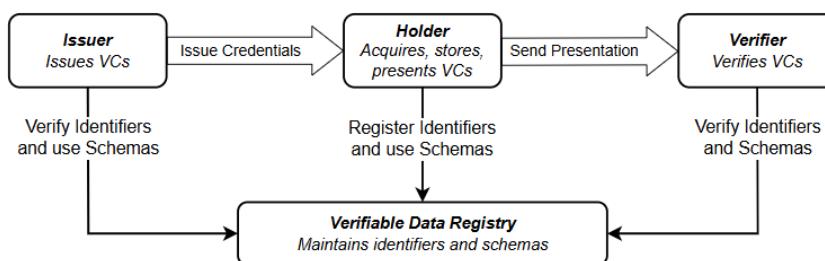


Figura 5: Funzionamento protocollo VC

Verifiable Presentation

La Verifiable Presentation (VP) specifica i metodi per firmare e presentare le VC da parte del titolare. Le VP possono essere utilizzate per aggregare informazioni provenienti da più credenziali verificabili.

Token

I token sono unità digitali di valore create e gestite su blockchain esistenti tramite smart contract. Sono programmabili, trustless e trasparenti, e possono rappresentare diversi tipi di asset o diritti d'uso. Esistono sia token fungibili (intercambiabili e divisibili) sia non fungibili (NFT), unici e non divisibili. I token vengono utilizzati in molti contesti, ad esempio per accedere a servizi, partecipare a ecosistemi decentralizzati, o per incentivare comportamenti desiderati all'interno di una piattaforma. Il valore di un token può dipendere dalla domanda e dall'offerta, dalla sua utilità all'interno di un sistema e dalle regole di emissione e distribuzione definite dal progetto.

IPFS

IPFS è una rete peer-to-peer decentralizzata per organizzare e trasferire dati indirizzati tramite contenuto (content-addressed). Non è un provider di storage o cloud.

I Content Identifier (CID) in IPFS sono etichette uniche che puntano ai dati basandosi sul contenuto stesso, non sulla loro posizione. I CIDs sono brevi, indipendente dalla dimensione del contenuto e sono generati calcolando un hash crittografico (di default SHA-256) del blocco di dati, il che significa che anche una minima modifica nel contenuto produce un CID diverso mentre lo stesso contenuto aggiunto a due nodi IPFS diversi con le stesse impostazioni produrrà lo stesso CID. Ogni CID contiene l'hash e informazioni sul codec, che indicano come interpretare i dati, e utilizza formati multibase, multicodec e multihash per garantire flessibilità e compatibilità.

Quando un file è più grande di qualche centinaio di KB, viene suddiviso in blocchi più piccoli, ognuno con il proprio CID, e questi blocchi vengono collegati in una struttura chiamata Merkle DAG. In questa struttura, i nodi foglia contengono i blocchi di dati, mentre i nodi intermedi contengono i riferimenti (CID) ai blocchi o ad altri nodi. Il CID del nodo radice rappresenta l'intero file.

WP3

Questo work package ha lo scopo di analizzare la soluzione presentata in WP2 rispetto al modello presentato in WP1. Verranno esaminate le principali proprietà di sicurezza: confidenzialità, integrità, trasparenza ed efficienza. Alla fine del documento, verrà esibito e giustificato un grafico radar che rappresenta le prestazioni del sistema in base a queste quattro proprietà.

3.1 Confidenzialità

La confidenzialità è una proprietà cruciale per garantire che le informazioni sensibili degli utenti siano protette e accessibili solo dalle parti autorizzate.

- **C.1: Le identità dei clienti devono poter essere mantenute riservate dal servizio di recensioni, impedendo che gli hotel o altri utenti possano risalirvi direttamente.**

L'hash dell'identificativo della VC, concatenato con un salt non pubblico, impedisce il collegamento diretto tra la recensione on-chain e l'identità del cliente. Il salt è custodito off-chain, in modo sicuro. Gli utenti non possono risalire ai clienti che hanno rilasciato una recensione. Quando viene rilasciato un token a un cliente, l'hotel, che conosce i DID dei suoi clienti, potrebbe confrontare l'indirizzo associato alla transazione con quelli a sua disposizione.

- **C.2: Solo il cliente che ha effettivamente soggiornato presso una struttura deve poter inserire, modificare o revocare una recensione relativa a quel soggiorno. Il sistema deve impedire che utenti diversi possano impersonare il cliente.**

La presenza della Verifiable Credential (VC) rilasciata dalla struttura, validata tramite firma digitale e unicità, impedisce impersonificazioni. Solo chi possiede la VC può agire su quella recensione.

Per garantire che un cliente non sia fittizio (non ha soggiornato ma ha una VC valida rilasciata in accordo dall'hotel) viene utilizzata anche una VC rilasciata dal sistema di prenotazione.

- **C.3: Il servizio di recensioni deve evitare di esporre dati non essenziali (es. dettagli identificativi) che possano ricondurre al cliente.**

Nessun dato personale del cliente viene scritto on-chain. Anche il CID IPFS non contiene informazioni personali.

Il sistema garantisce lo pseudoanonimato, in quanto effettuando la transazione per il rilascio del reward viene esposto l'indirizzo pubblico del cliente. Un utente semplice non può risalire all'identità degli altri utenti, ma un hotel potrebbe.

- **C.4: L'elenco delle recensioni completo (anche di recensioni rimosse) deve essere accessibile solo alla struttura alberghiera interessata, non a utenti esterni o ad altri hotel.**

Anche se una recensione viene eliminata, il suo CID (Content Identifier) rimane accessibile al servizio di recensioni. Tuttavia, solo l'hotel destinatario della recensione può richiederne la visualizzazione al servizio di recensioni. Questo consente alla struttura di consultare anche le recensioni rimosse relative a sé stessa, mantenendo però la riservatezza rispetto ad altri utenti o hotel, e assicurando così che l'elenco sia visibile esclusivamente al soggetto interessato.

- **C.5: Un cliente che recensisce due strutture distinte non deve poter essere riconosciuto tramite la recensione come la stessa persona da soggetti terzi o dalle strutture stesse.**

Ogni recensione è legata a una Verifiable Credential (VC) emessa specificamente per una singola esperienza presso una struttura. Il sistema di recensione conserva un identificatore univoco (ID) per la VC associato a un salt, ma questo dato non è legato ai dati personali dell'utente.

3.2 Integrità

L'integrità è fondamentale per garantire che i dati non siano stati alterati in modo non autorizzato e che le credenziali siano autentiche.

- **I.1: Solo i clienti che hanno effettuato e concluso una prenotazione reale devono essere abilitati a pubblicare recensioni. Il sistema deve garantire che tale diritto sia verificabile e non trasferibile ad altri utenti.**

La VC, emessa solo al termine del soggiorno, funge da prova crittografica dell'esperienza. La piattaforma verifica la firma dell'hotel e l'unicità d'uso della VC.

- **I.2: Le recensioni devono essere immutabili una volta pubblicate: ogni eventuale modifica della recensione o risposta dell'hotel deve essere tracciata in modo indelebile, mantenendo accessibile l'elenco delle versioni.**

La versione originale della recensione viene salvata su IPFS, garantendone l'immutabilità, mentre lo smart contract registra il relativo CID (Content Identifier) e lo stato associato. In caso di modifica da parte dell'utente, la nuova versione della recensione, viene registrata nuovamente su IPFS, viene restituito un nuovo CID e aggiornato lo stato.

Anche le risposte dell'hotel vengono salvate su IPFS e associate alla recensione di riferimento.

Grazie all'uso della blockchain, l'intera cronologia delle modifiche e delle risposte resta tracciabile e consultabile in modo trasparente e sicuro.

- **I.3: La rimozione o eliminazione di una recensione non deve comportare la perdita di informazioni: la cronologia completa degli eventi (inserimento, modifica, rimozione) deve essere conservata in modo sicuro e inalterabile.**

La recensione eliminata non viene cancellata da IPFS e il CID resta tracciabile sulla blockchain. Lo stato viene solo modificato, segnalando la cancellazione logica, non fisica. La tracciabilità è completa.

- **I.4: Il sistema deve impedire che una recensione possa essere modificata in modo arbitrario o illimitato. Devono esistere regole chiare (tempi, condizioni, limiti) che vincolano l'azione e garantiscono la coerenza con la versione originale.**

Le recensioni possono venire modificate una sola volta ed entro le 24h dalla pubblicazione come specificato nelle policy. Quando viene richiesta la modifica il sistema controlla lo stato della recensione e se stata già modificata ne impedisce un ulteriore modifica. Per garantire la coerenza con la versione originale il campo sull'esperienza del cliente (positiva o negativa) non può essere modificato. L'esperienza è considerata un evento soggettivo e non alterabile, a meno di comportamenti illeciti. Per questo motivo, il campo è immutabile sin dal primo inserimento.

3.3 Trasparenza

La trasparenza è essenziale per garantire che gli utenti e le autorità possano fidarsi del sistema.

- **T.1: Il sistema deve rendere pubblici i criteri con cui le recensioni vengono moderate, rese visibili, ordinate e, eventualmente, rimosse.**

Il sistema fa riferimento in modo esplicito alle seguenti policy:

- **Policy Antimanipolazione:**
 - "*Nessuna cancellazione da parte della piattaforma o degli hotel*": garantisce che non ci siano interventi discrezionali nella moderazione da parte di terzi.
 - "*Tracciabilità completa delle interazioni*": ogni modifica o cancellazione viene tracciata su blockchain, rendendo il processo trasparente e consultabile.
- **Policy Generali di Visualizzazione Recensioni:**
 - "*Ordinamento dinamico*": le recensioni sono ordinate in base alla prevalenza qualitativa (positive o negative), criterio esplicitamente dichiarato e accessibile agli utenti.
 - "*Visualizzazione recensioni rimosse*": Solo l'hotel oggetto della recensione può vedere le recensioni rimosse se fa richiesta alla piattaforma di recensione.

- **T.2: Le regole che stabiliscono chi può recensire, in quali casi, e con quali vincoli temporali, devono essere esplicate e consultabili dagli utenti.**

Il sistema di recensioni adotta policy rigorose e vincolanti che stabiliscono con precisione chi può pubblicare una recensione e a quali condizioni. In particolare, solo gli utenti che possono dimostrare di aver prenotato e completato un soggiorno presso la struttura, presentando due VC (una per la prenotazione e una per il soggiorno effettivo), hanno diritto a lasciare una

recensione. Ciascun utente può pubblicare una sola recensione per ogni soggiorno, e solo dopo almeno 12 ore dalla registrazione dell'ultima VC.

Le policy della piattaforma sono disponibili pubblicamente.

- **T.3: Gli algoritmi e i protocolli utilizzati per la gestione delle recensioni devono essere pubblicamente noti e verificabili.**

Le tecnologie adottate, come Smart Contract, Verifiable Credentials (VC), Verifiable Presentation (VP) e Decentralized identifiers (DID), si basano su standard aperti e verificabili, come quelli definiti dal W3C. Le recensioni sono archiviate su IPFS, un sistema distribuito open source che assicura integrità e accesso pubblico ai contenuti, pur non essendo uno standard formale.

3.4 Efficienza/Usabilità

L'efficienza e l'usabilità sono importanti per garantire che il sistema sia pratico e possa essere utilizzato senza problemi dagli utenti.

- **E.1: Il sistema di inserimento della recensione deve essere rapido, semplice e fruibile senza richiedere competenze tecniche.**

L'utente, una volta autenticato tramite wallet e in possesso di due Verifiable Credentials (VC) valide, una rilasciata dalla piattaforma di prenotazione e una dall'hotel al termine del soggiorno, può rilasciare un'unica recensione attraverso un'interfaccia semplice e intuitiva. Per farlo, presenta una Verifiable Presentation (VP) firmata, che incapsula le due VC e ne attesta la validità in modo sicuro. L'utente scrive la recensione e il contenuto viene salvato dalla piattaforma su IPFS, garantendo integrità e persistenza dei dati. Successivamente, viene registrato tramite smart contract sulla blockchain che memorizza il riferimento (CID) alla recensione su IPFS, senza richiedere all'utente un ulteriore interazione. Questo flusso semplificato assicura un'esperienza d'uso accessibile anche a utenti senza competenze tecniche.

- **E.2: Il processo di verifica dell'effettivo soggiorno del cliente deve essere eseguito in tempi brevi e non ostacolare l'esperienza utente.**

L'utente presenta una VP (Verifiable Presentation) tramite il proprio wallet digitale, contenente le VCs, che attesta in modo verificabile e sicuro la prenotazione e l'effettiva permanenza presso la struttura. La verifica viene eseguita automaticamente tramite controlli crittografici sulle firme digitali, la validità temporale e la coerenza dei dati, senza richiedere azioni aggiuntive da parte dell'utente. L'intero processo avviene in pochi secondi, garantendo rapidità e fluidità nell'esperienza utente, senza ostacoli o ritardi.

- **E.3: Il servizio deve permettere la pubblicazione, modifica o rimozione delle recensioni entro tempistiche che ne preservino l'utilità informativa e la qualità complessiva del sistema.**

Il sistema permette la pubblicazione e una sola modifica della recensione entro tempi stabiliti da policy che garantiscono pseudoanonimato e integrità del contenuto. Tali vincoli temporali (12 ore post-soggiorno per la pubblicazione e massimo 24 ore per la modifica) sono definiti per prevenire influenze esterne e comportamenti scorretti, preservando l'utilità informativa e la qualità complessiva del sistema. Gli algoritmi di verifica VP e VCs, scrittura su IPFS e interazione con smart contract non impattano significativamente sui tempi, mantenendo l'efficienza del processo. La rimozione non ha limiti temporali in quanto non intacca l'utilità informativa o la qualità del sistema.

- **E.4: Il sistema di recensioni deve prevedere policy chiare e accessibili che guidino l'utente nel processo di rilascio, indicando cosa è consentito fare, quando e in che modo. Tali policy devono agevolare la comprensione del funzionamento del sistema anche da parte di utenti non esperti.**

Le policy che regolano il rilascio delle recensioni sono pubbliche, esplicate e accessibili agli utenti. Indicano in modo trasparente:

- chi può recensire -> solo chi può dimostrare di aver prenotato e soggiornato nella struttura (VCs)
- cosa è possibile fare -> rilasciare una recensione, e successivamente modificarla/eliminarla. Ogni cliente deve dichiarare la qualità dell'esperienza al rilascio della recensione.
- quando -> almeno 12 ore dopo il rilascio della VC
- in che modalità -> una sola recensione per soggiorno. È possibile modificare la recensione entro le 24 ore dalla pubblicazione.

Queste regole aiutano l'utente a comprendere con chiarezza come comportarsi.

- **E.5: Il sistema deve incentivare attivamente il rilascio delle recensioni da parte degli utenti che hanno effettivamente soggiornato nella struttura.**

Ogni utente che pubblica una recensione valida, verificata tramite VC rilasciate da hotel e piattaforma di prenotazione, riceve automaticamente 1/10 di un token ERC-20 non scambiabile. Dopo 10 recensioni autentiche, può riscattare un buono sconto per prenotazioni future. Il sistema premia solo chi ha effettivamente soggiornato, incentivando feedback genuini e tracciabili.

3.5 Valutazione delle proprietà

L'analisi delle proprietà di confidenzialità, integrità, trasparenza ed efficienza/usabilità ha messo in luce i punti di forza e le aree di miglioramento del sistema esaminato. Di seguito, una valutazione approfondita di ciascuna proprietà, con punteggi assegnati da 1 a 100 per riflettere la qualità complessiva della soluzione.

Confidenzialità

Punteggio: 90/100

Le proprietà di confidenzialità garantiscono che le informazioni sensibili degli utenti siano accessibili solo alle parti autorizzate. Il sistema tutela il cliente garantendo lo pseudo-anonimato. Un punto di fallimento potrebbe presentarsi dopo il rilascio della reward, nessun dato sensibile viene esposto ma viene mostrato l'indirizzo pubblico a cui è indirizzata la reward. Nessun utente può risalire all'identità di chi ha rilasciato la recensione a differenza degli hotel che potrebbero salvare il DID dei clienti che hanno soggiornato presso essi e confrontarlo con l'indirizzo pubblico associato al rilascio della reward.

Tuttavia, una possibile soluzione a questo problema consiste nell'adozione di tecnologie come Mixer o Zero-Knowledge Proof.

Integrità

Punteggio: 100/100

Il sistema soddisfa pienamente i requisiti di integrità previsti: garantisce che solo clienti verificati possano recensire, assicura l'immutabilità e la tracciabilità completa delle versioni e delle eliminazioni, e applica regole chiare e vincolanti per le modifiche. L'uso combinato di Verifiable Credentials, IPFS e blockchain fornisce solide garanzie crittografiche e un controllo coerente e trasparente sul ciclo di vita delle recensioni, con un livello di protezione elevato.

Trasparenza

Punteggio: 100/100

Il sistema soddisfa pienamente i requisiti di trasparenza. Le policy di moderazione, visualizzazione e pubblicazione delle recensioni sono chiaramente definite, pubbliche e facilmente consultabili dagli utenti. L'adozione di standard aperti e tecnologie verificabili come Verifiable Credentials, Smart Contract e IPFS garantisce un controllo indipendente sui dati e sui processi, assicurando fiducia e comprensione completa del funzionamento del sistema.

Efficienza e usabilità

Puntaggio: 95/100

Il sistema proposto dimostra un elevato grado di soddisfacimento delle proprietà legate all'efficienza e all'usabilità, grazie a soluzioni ben progettate sia dal punto di vista tecnico sia dell'esperienza utente. Non richiede competenze tecniche elevate, ma questo aspetto potrebbe essere ulteriormente semplificato: l'uso di wallet, Verifiable Credentials (VC) e Verifiable Presentations (VP), pur reso il più intuitivo possibile, può comunque rappresentare una barriera per utenti meno esperti o non familiari con tecnologie decentralizzate.

Di seguito il grafico radar per mostrare visivamente i punti di forza e debolezza del progetto.

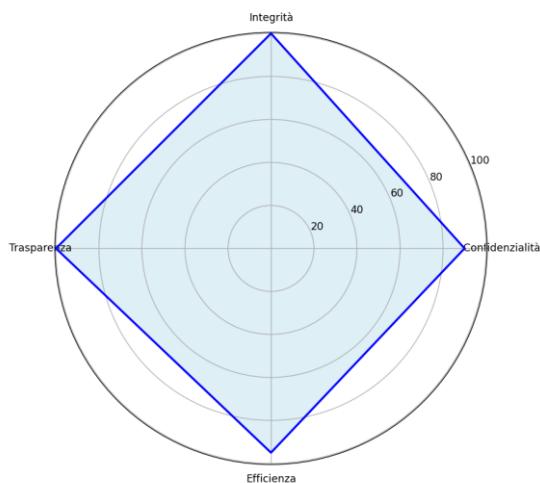


Figura 6: Grafico Radar

Il risultato comunica un sistema ben progettato, affidabile e trasparente, con piccole criticità.

3.6 Contrasto degli avversari

Impersonator

Descrizione: Individuo che cerca di impersonare un utente legittimo.

Attacchi:

- Impersonificazione: Utilizzare account falso per fingere di essere un cliente che ha realmente soggiornato.

Come lo sconfiggiamo:

- Accesso con wallet digitale e presentazione di una VP contenente VC verificate. Viene effettuata la verifica sulle firme digitali su VP e VC per garantire autenticità e identità del cliente e controlli sulle VC come la validità temporale e coerenza delle informazioni. Solo chi possiede le VC emesse direttamente dalla struttura ricettiva e dal sistema di prenotazione può pubblicare o modificare una recensione.

Risultato: un Impersonator che prova a fingere di essere un cliente legittimo non potrà pubblicare recensioni, poiché non dispone delle credenziali valide.

Hotel Reviewer Coercer

Descrizione: Struttura ricettiva che sollecita o esercita pressioni sui clienti affinché pubblichino recensioni positive o modifichino recensioni esistenti.

Attacchi:

- Coercizione Post-Servizio: Pressione psicologica o incentivi in cambio di recensioni positive.
- Modifica Forzata: Richiesta esplicita di modifica di una recensione negativa già pubblicata.

Come lo limitiamo:

- Pseudononimato garantito: Le recensioni sono pseudoanonime e non collegate all'identità del cliente, impedendo all'hotel di sapere chi ha scritto cosa.
- Temporizzazione protettiva: Il sistema consente all'utente di rilasciare la recensione solo dopo 12 ore dal rilascio della Verifiable Credential (VC).
Questo intervallo temporale garantisce che il cliente sia ormai libero da qualsiasi pressione da parte dell'hotel, senza più contatti diretti con la struttura.
- Esperienza non modificabile: Il campo relativo all'esperienza (positiva o negativa) è immutabile e impostato dall'utente, non può essere modificata dopo l'inserimento.
- È possibile modificare la recensione una sola volta e dopo 24 ore dalla pubblicazione non è più modificabile. Questo impedisce pressioni ritardate o strategie coercitive nel tempo.

Risultato: L'Hotel Reviewer Coercer è completamente sconfitto.

Grazie ad pseudoanonimato, vincoli temporali, immutabilità dell'esperienza generale e limitazioni sulle modifiche, ogni tentativo di manipolazione da parte dell'hotel è inefficace e bloccato dal sistema.

Hotel Saboteur

Descrizione: Hotel che pubblica recensioni false e negative contro i concorrenti.

Attacchi:

- Attacchi alla Reputazione: Recensioni negative non verificate e dettagliate contro hotel concorrenti.
- Flooding Coordinato: Pubblicazione massiva di contenuti negativi in un breve lasso di tempo.

Come lo sconfiggiamo:

- Le recensioni possono essere pubblicate solo se si possiede una VC (Verifiable Credential) rilasciata dalla struttura in cui si è soggiornato. Senza VC valide, non può pubblicare recensioni false, quindi non puo' effettuare attacchi alla reputazione e neanche eseguire flooding coordinato.

Risultato: L'hotel saboteur non può ottenere VC dai concorrenti; quindi, non può simulare soggiorni presso altre strutture.

Reputational Manipulator

Descrizione: Hotel che tenta di alterare artificiosamente la percezione pubblica del proprio profilo online, agendo sia per nascondere feedback negativi reali, sia per gonfiare la propria reputazione mediante recensioni positive fraudolente.

Attacchi:

- Rimozione di Feedback Negativi: Utilizzo delle procedure ufficiali della piattaforma (es. segnalazioni) per ottenere la cancellazione di recensioni autentiche ma negative.
- Saturazione Positiva: Generazione o acquisto massivo di recensioni positive con l'obiettivo di sovrastare e diluire l'effetto delle recensioni negative.
- Oscuramento Algoritmico: Sfruttamento dei meccanismi di ranking per rendere meno visibili recensioni critiche (es. creando molte recensioni nuove per spingere in basso quelle vecchie).

Come lo contrastiamo:

- Solo utenti che presentano una VP contenente VC di prenotazione e soggiorno possono recensire, rendendo di fatto impossibile per account fintizi o bot scrivere recensioni senza entrambe le prove.
- L'algoritmo di ordinamento penalizza le recensioni incoerenti con l'esperienza complessiva di tutti i clienti, si ha ridotta visibilità di contenuti anomali.
- Le policy limitano ogni utente alla pubblicazione di una sola recensione per ciascun soggiorno, impedendo la possibilità di aumentare il numero di recensioni.
- Il sistema non consente la segnalazione delle recensioni da parte delle strutture ricettive. Le recensioni autentiche, legate a soggiorni verificati, non possono essere cancellate o oscurate arbitrariamente.

Risultato: Il Reputational Manipulator è completamente sconfitto. L'assenza di meccanismi di segnalazione, il vincolo di una sola recensione per soggiorno, l'accesso limitato solo a chi possiede VC valide e l'ordinamento basato sulla coerenza con l'esperienza generale impediscono ogni tentativo di manipolazione. Le recensioni false perdono visibilità, quelle autentiche restano, e l'hotel non può alterare la propria reputazione.

Platform Manipulator

Descrizione: Operatore interno della piattaforma che altera recensioni valide.

Attacchi:

- **Cancellazione Illegittima:** Rimozione di recensioni vere senza giustificazione o per conto di un hotel.
- **Filtro Opaque:** Manipolazione del ranking o della visibilità delle recensioni selezionate.

Come lo limitiamo:

- L'operatore della piattaforma non può modificare il flag dell'esperienza, che viene impostato direttamente dall'utente e risulta immutabile. Questo flag determina l'ordinamento e la visibilità delle recensioni, secondo le Policy Generali di Visualizzazione.
- L'eliminazione di una recensione è riservata esclusivamente al cliente, che può farlo solo autenticandosi tramite wallet. L'operazione di cancellazione avviene attraverso uno smart contract, rendendo impossibile per l'operatore intervenire direttamente sulla recensione.

Risultato: Il Platform Manipulator è completamente sconfitto. L'impossibilità di modificare i flag che determinano la visibilità delle recensioni e l'uso di smart contract per la cancellazione, accessibili solo all'utente autenticato, impediscono qualsiasi intervento diretto da parte dell'operatore. Ogni tentativo di alterazione viene bloccato dal design stesso del sistema.

Fake Reviewer

Descrizione: Utente che pubblica recensioni su strutture che non ha realmente visitato.

Attacchi:

- Recensioni Fittizie: Scrittura di contenuti ingannevoli per danneggiare o favorire un servizio.
- Recensioni su Commissione: Attività incentivata economicamente da hotel.

Come lo sconfiggiamo:

- La pubblicazione di recensioni è riservata esclusivamente agli utenti in possesso di due Verifiable Credentials (VC): VC di prenotazione, rilasciata dal sistema di prenotazione ufficiale e VC di fine soggiorno, rilasciata dalla struttura ricettiva al termine della visita. Questo garantisce che solo chi ha effettivamente soggiornato possa lasciare una recensione, eliminando recensioni false o di persone mai state in struttura. Se l'hotel rilascia VC arbitrariamente per avere recensioni, l'utente senza una prenotazione valida non può rilasciare una recensione.
- Ogni recensione viene valutata in base alla coerenza con il comportamento degli altri clienti: recensioni sospette, cioè con valutazioni estreme o discordanti rispetto alla media, vengono automaticamente penalizzate nella visibilità, posizionate in fondo alla lista. Quindi i contenuti ingannevoli hanno un impatto marginale.

Risultato: Il Fake Reviewer è completamente sconfitto. Solo i clienti reali possono lasciare recensioni, e i contenuti sospetti hanno un impatto trascurabile. Il sistema impedisce sia l'inserimento che la valorizzazione di recensioni non idonee.

Feedback Abuser

Descrizione: Cliente che ha realmente soggiornato in una struttura, ma tenta di pubblicare più recensioni per la stessa prenotazione o modifica periodicamente la recensione effettuata, per influenzare sulla percezione dell'hotel.

Attacchi:

- Recensioni Multiple: Inserimento di più recensioni per la stessa esperienza, per aumentare l'impatto sul punteggio dell'hotel.
- Modifica Aggressiva: Uso ripetuto della funzione di modifica per alterare continuamente il contenuto, inducendo confusione o manipolando la percezione nel tempo

Come lo sconfiggiamo:

- Blocco delle recensioni multiple: Il sistema associa in modo univoco ogni recensione a una specifica esperienza (hash dell'id della VC dell'hotel + salt). Una volta pubblicata una recensione per un soggiorno non se ne possono effettuare più.
- Controllo sulle modifiche: È consentita una sola modifica alla recensione entro le 24 ore dalla pubblicazione. Lo stato garantisce la tracciabilità e blocca ulteriori cambiamenti.

Risultato: Il Feedback Abuser è completamente sconfitto. Il cliente non può abusare del sistema per:

- Pubblicare più recensioni per lo stesso soggiorno
- Cambiare in modo continuativo il contenuto per manipolare la percezione pubblica.

NO-REVS

Descrizione: Opppositori convinti che qualsiasi tentativo di regolamentazione dei sistemi di feedback sia destinato al fallimento per via della corruzione intrinseca delle piattaforme centralizzate.

Attacchi:

- **Campagne di Disinformazione:** Diffusione di notizie negative per ostacolare l'adozione del sistema.

Come li contrastiamo:

- Il sistema adotta un modello decentralizzato e tracciabile (VC + IPFS + blockchain), proprio per superare le debolezze delle piattaforme centralizzate.
- Tuttavia, il sistema di recensione conserva il salt: questo potrebbe essere usato come argomento per alimentare la sfiducia.
- Utilizzo di token come ricompensa per incentivare il rilascio di recensioni.

- Ordinamento in base all'esperienza complessiva di tutti gli utenti che hanno recensito. Se il maggior numero di utenti ha avuto un'esperienza negativa è maggiore la probabilità che anche un cliente futuro avrà un'esperienza negativa, e viceversa.
- L'utilizzo di policy mirate, trasparenti e ben definite contribuisce ad accrescere la fiducia degli utenti e a rafforzare la percezione positiva nei confronti del servizio

Risultato: NO-REVS parzialmente sconfitti: l'adozione di tecnologie decentralizzate e policy trasparenti contrasta le critiche verso le piattaforme centralizzate, ma la presenza del salt potrebbe ancora offrire margine alla disinformazione. Inoltre il meccanismo di ordinamento aiuta ad aumentare la fiducia, e il rilascio di token ad incentivare l'utilizzo della piattaforma.

WP4

L'obiettivo del WP4 è implementare in modo concreto le funzionalità progettate e descritte nel WP2, realizzando il sistema attraverso smart contract scritti in Solidity, utilizzando Ganache per il testing locale, file JavaScript per la logica applicativa e IPFS per la gestione e la condivisione dei dati.

Gli indirizzi utilizzati per i test sono illustrati nelle seguenti tabelle.

Indice	Account	Sistema	Address
0		Recensioni	0xbeA7DCBb8Bf2aCEA3A9caED9795FED0Eb74852dE
9		Prenotazioni	0xe669317f3b7fE8Cda46Aa8e0c99aE87D3Ee48640

Indice	Account	Ruolo	Nome	Address
1		Hotel	Hotel California	0x1294c7C392AACba5a4f5d133ed0DfA08aC519552
2		Hotel	Hotel Team 8	0xa8f0a4625d3D157a1a23cbE08b025D78D1af575F
3		Cliente	Marco	0x89b80914F85FBd9513028409fB68a991903B7d65
4		Cliente	Alessia	0x5D6536dE59DeED70a4a6D1c977b3d535eece3866
5		Cliente	Pasquale	0xB17Fd55900b5a1324E2cadaBB3df7262E693E259

L'ordine di esecuzione dei file è così definito:

1. Compilare e deployare tutti i contratti presenti nella cartella contract;
2. Creare VP e VC, `create_vc_prenotazione.js`, `create_vc_hotel.js` e `create_vp.js`. Saranno presenti nella cartella Wallet/utente e verranno utilizzate per l'identificazione dell'utente.
3. Sostituire correttamente gli address dei contratti deployati all'interno dei file javascript dove necessario;
4. Per simulare il sistema di recensioni:
 - o `insert_rec.js`
 - o `modify_rec.js`
 - o `answer.js`
 - o `viewRec.js`
 - o `delete_rec.js`

⚠ Note importanti:

- L'ordine dopo la `insert_rec` è puramente arbitrario;
- Per inserire recensioni con utenti diversi, bisogna modificare l'indirizzo utilizzato scegliendo opportunamente quello dell'utente desiderato

4.1 Contratti

La cartella contract contiene gli smart contract Solidity del progetto, organizzati in sottocartelle per argomenti.

EthrDIDRegistry.sol

Il contratto EthereumDIDRegistry gestisce l'identità decentralizzata (DID) sul sistema, fornisce la base per un sistema di identità trasparente e verificabile on-chain, supportando la gestione sicura di ruoli e permessi senza un'autorità centrale.

GestioneRecensioni.sol

Il contratto GestioneRecensioni consente di registrare recensioni su IPFS, modificarle entro 24 ore o cancellarle, permettendo a chiunque di visualizzare solo le recensioni attive e le relative risposte dell'hotel; solo l'hotel (tramite l'admin) può invece consultare tutte le recensioni, comprese quelle eliminate.

MyToken.sol

Il contratto MyToken definisce token ERC-20 con nome, simbolo, decimali e fornitura totale assegnata all'admin al momento del deploy; solo l'admin può trasferire i token tra indirizzi, aggiornando i bilanci e registrando i movimenti con eventi di trasferimento. Serve per ricompensare il cliente per aver utilizzato l'applicazione.

Il file listen.js può essere utilizzato per mettersi in ascolto degli eventi emessi.

4.2 File JavaScript

Create_vc_prenotazioni.js

Questo file serve per creare la vc relativa alla prenotazione per effettuare un soggiorno in una struttura. Questa vc viene rilasciata dal sistema di prenotazione e salvata sul wallet del cliente.

```
● PS C:\Users\marco\OneDrive\Desktop\ProjectWork_blochchain> node .\create_vc_prenotazioni.js
User DID is: did:ethr:0x539:0xB17Fd55900b5a1324E2cadaBB3df7262E693E259
Booking DID is: did:ethr:0x539:0xe669317f3b7fE8Cda46Aa8e0c99aE87D3Ee48640
{
  '@context': [ 'https://www.w3.org/2018/credentials/v1' ],
  id: 'http://Booking.example/credentials/0008',
  type: [ 'VerifiableCredential', 'Bookingx' ],
  issuer: 'did:ethr:0x539:0xe669317f3b7fE8Cda46Aa8e0c99aE87D3Ee48640',
  credentialSubject: {
    id: 'did:ethr:0x539:0xB17Fd55900b5a1324E2cadaBB3df7262E693E259',
    Book: {
      Num_person: 3,
      Num_notti: 1,
      CheckIn: 'Thu Jul 10 2025',
      CheckOut: 'Fri Jul 11 2025',
      Add_hotel: '0x1294c7C392AACba5a4f5d133ed0DfA08aC519552',
      Release: '2025-07-15T08:29:53.961Z'
    }
  }
}
VC saved in the wallet: Wallet/Pasquale/vc_Jwt_Booking.txt
```

Figura 7: VC rilasciata dal sistema di prenotazioni

Create_vc_hotel.js

Questo file serve per creare la vc rilasciata alla fine del soggiorno dalla struttura ricettiva e viene salvata sul wallet del cliente.

```
● PS C:\Users\marco\OneDrive\Desktop\ProjectWork_blochchain> node .\create_vc_hotel.js
User DID is: did:ethr:0x539:0xB17Fd55900b5a1324E2cadaBB3df7262E693E259
Hotel DID is: did:ethr:0x539:0x1294c7C392AACba5a4f5d133ed0DfA08aC519552
○ {
    vc: {
        '@context': [ 'https://www.w3.org/2018/credentials/v1' ],
        id: 'http://Hotel California.example/credentials/0008',
        type: [ 'VerifiableCredential', 'Hotel California' ],
        issuer: 'did:ethr:0x539:0x1294c7C392AACba5a4f5d133ed0DfA08aC519552',
        credentialSubject: {
            id: 'did:ethr:0x539:0xB17Fd55900b5a1324E2cadaBB3df7262E693E259',
            Stay: [Object]
        }
    },
    sub: 'did:ethr:0x539:0xB17Fd55900b5a1324E2cadaBB3df7262E693E259',
    nbf: 1752568266,
    iss: 'did:ethr:0x539:0x1294c7C392AACba5a4f5d133ed0DfA08aC519552'
}
VC saved in the wallet: Wallet/Pasquale/vc_Jwt_Hotel_California.txt
```

Figura 8: VC rilasciata dall'hotel

Create_vp.js

Questo file permette all'utente di generare la VP prendendo dal proprio wallet le due VC possedute. Questa VP verrà poi utilizzata per verificare l'identità del cliente.

Figura 9: VP del cliente

Listening.js

Questo file Node.js si occupa di ascoltare gli eventi generati dagli smart contract sulla blockchain locale (Ganache), come l'inserimento o la modifica di recensioni, l'invio di risposte e i trasferimenti di token.

⚠️ Nota importante: Per eseguire correttamente il file effettuare il deploy presente nelle cartelle in `contract/EthrDIDRegistry` e `contract/GestioneRecensioni` e inserire gli indirizzi all'interno del `listening.js`.

Insert_rec.js

Questo file serve per inserire la recensione:

- Controlla che la VP e le VC siano corrette, e con informazioni coerenti. In caso di esito positivo, genera un salt e calcola l'hash della VC rilasciata dall'hotel concatenata con il salt.
- Effettua controlli sulla recensione inserita (può essere pubblicata solo 12 ore dopo il rilascio della VC da parte dell'hotel) e successivamente carica la recensione su IPFS ottenendo il CID e chiama lo smart contract `GestioneRecensioni.sol` per inserire la recensione.
- Una volta pubblicata, la recensione viene emesso il reward al cliente tramite lo smart contract `Token.sol`.

⚠️ Nota importante: Per eseguire correttamente il file effettuare il deploy presente nelle cartelle in `contract` e inserire gli indirizzi all'interno del `insert_rec.js`.

```
PS C:\Users\Asus\Desktop\ProjectWork_blochchain> node .\insert_rec.js
User DID is: did:ethr:0x539:0xB17Fd55900b5a1324E2cadaBB3df7262E693E259
Hotel DID is: did:ethr:0x539:0x1294c7C392AACBaa5a4f5d133ed0DfA08aC519552
Booking DID is: did:ethr:0x539:0xe669317f3b7fE8Cda46Aa8e0c99aE87D3Ee48640
Checking the VP and VCs...
VP and VCs validated successfully!
Uploading review on IPFS...
New Cid: QmXzKC5yS1pJbr55EixwGrTg9y8WfwD4b7SsbJCZ6b6caU
Calling smart contract for insert the review...
Transfer the token as reward...
User balance:: 0.1
```

Figura 10: Inserimento recensione

Vedere figura 14 nel sottoparagrafo `Viewrec.js` per la visualizzazione del risultato

Modify_rec.js

Questo file consente all'utente di modificare una recensione, previa verifica dell'autenticità della Verifiable Presentation (VP) e della Verifiable Credential (VC) rilasciata dall'hotel. Viene controllata la presenza dell'ID della VC nei database e calcolato l'hash combinando l'ID_VC con il salt. La modifica è consentita solo se sono trascorse meno di 24 ore dalla pubblicazione della recensione. Il campo relativo all'esperienza generale, tuttavia, non è modificabile.

⚠️ Nota importante: Per eseguire correttamente il file effettuare il deploy presente nelle cartelle in `contract/EthrDIDRegistry` e `contract/GestioneRecensioni` e inserire gli indirizzi all'interno del `modify_rec.js`.

```
PS C:\Users\Asus\Desktop\ProjectWork_blochchain> node .\modify_rec.js
New CID: QmQvfDPfrCyHmV7DNsCSdV9R78rtoAraD5uppvpuNHvvyE
Recensione modificata con successo.
```

Figura 11: Modifica effettuata

Vedere figura 15 nel sottoparagrafo `Viewrec.js` per la visualizzazione del risultato

Delete_rec.js

Con `delete_rec.js` gestisce l'eliminazione di una recensione: verifica la validità della Verifiable Presentation (VP) e della Verifiable Credential (VC) rilasciata dall'hotel, controlla la presenza dell'ID della VC nel database, calcola l'hash (ID_VC + salt) e abilita l'eliminazione solo se tutti i controlli hanno esito positivo.

⚠️ Nota importante: Per eseguire correttamente il file effettuare il deploy presente nelle cartelle in `contract/EthrDIDRegistry` e `contract/GestioneRecensioni` e inserire gli indirizzi all'interno del `delete_rec.js`.

```
PS C:\Users\marco\Desktop\ProjectWork_blochchain> node .\delete_rec.js
Review successfully deleted.
```

Figura 12: Eliminazione di una recensione

Vedere figura 17 nel sottoparagrafo `Viewrec.js` per la visualizzazione del risultato

Answer.js

Questo file consente a un hotel di rispondere a una recensione lasciata da un cliente. La risposta è possibile perché, al momento della visualizzazione, sono noti i CID associati alle recensioni. Ogni

⚠️ Nota importante: Per eseguire correttamente il file effettuare il deploy presente nella cartella `contract/GestioneRecensioni` e inserire gli indirizzi all'interno del `answer.js`.

recensione può avere una sola risposta, che una volta inserita non è modificabile. L'operazione avviene tramite lo smart contract GestioneRecensioni.sol.

```
PS C:\Users\marco\Desktop\ProjectWork_blochchain> node .\answer.js  
Uploading answer on IPFS...  
Cid answer: QmYPrHS35PYMHBB9r2dUU4t8cTeGRhuLHYAzSyL8FDKD5y
```

Figura 13: Inserimento Risposta

Vedere figura 16 nel sottoparagrafo Viewrec.js per la visualizzazione del risultato

Viewrec.js

Questo file permette di visualizzare le recensioni attive di un hotel specifico. Lo smart contract GestioneRecensioni.sol restituisce i CID delle recensioni, che vengono poi recuperate e stampate tramite il file JavaScript. Ogni utente può richiedere la visualizzazione delle recensioni attive. Il sistema ordina le recensioni in base alla coerenza con l'esperienza verificata degli altri utenti: se la maggioranza ha valutato l'esperienza come positiva, le recensioni positive vengono mostrate per prime; se prevalgono giudizi negativi, queste recensioni ottengono maggiore visibilità, segnalando un rischio più alto ai futuri clienti. In caso di parità, viene data priorità alle recensioni positive per non penalizzare l'hotel (figura 17).

Per il test sono state inserite tre recensioni provenienti da clienti diversi.

⚠️ Nota importante: Per eseguire correttamente il file effettuare il deploy presente nella cartella contract/GestioneRecensioni e inserire gli indirizzi all'interno del answer.js.

```
PS C:\Users\marco\Desktop\ProjectWork_blochchain> node .\viewRec.js  
1. L'hotel non era il massimo!  
2. L'hotel è veramente sporco! Sconsigliato.  
1. L'hotel in cui ho soggiornato mi è sembrato molto accogliente. Il personale è stato molto cordiale, ed in generale un ottima esperienza! Raccomando tantissimo.
```

Figura 14: Ordinamento visualizzazione recensioni inserite. Maggiore priorità alle negative

```
PS C:\Users\marco\Desktop\ProjectWork_blochchain> node .\viewRec.js  
1. L'hotel non era il massimo!  
2. L'hotel è veramente sporco! E il personale davvero incompetente. Sconsigliato.  
1. L'hotel in cui ho soggiornato mi è sembrato molto accogliente. Il personale è stato molto cordiale, ed in generale un ottima esperienza! Raccomando tantissimo.
```

Figura 15: Visualizzazione recensione dopo la modifica della seconda recensione modificata

Solo l'hotel puo' visualizzare tutte le recensioni (attive ed eliminate), effettuandone la richiesta viene eseguita una procedura di verifica (challenge) durante la quale l'hotel deve dimostrare il

possesso del proprio DID. Solo a seguito del superamento positivo di tale verifica, la struttura potrà accedere alle recensioni eliminate.

```
PS C:\Users\marco\Desktop\ProjectWork_blochchain> node .\viewRec.js
1. L'hotel non era il massimo!
↳ Risposta: Non mi sono per niente offeso! Secondo me menti.
2. L'hotel è veramente sporco! E il personale davvero incompetente. Sconsigliato.
1. L'hotel in cui ho soggiornato mi è sembrato molto accogliente. Il personale è stato molto cordiale,
ed in generale un ottima esperienza! Raccomando tantissimo.
```

Figura 16: Visualizzazione recensioni dopo la risposta da parte dell'hotel interessato

```
PS C:\Users\marco\Desktop\ProjectWork_blochchain> node .\viewRec.js
1. L'hotel in cui ho soggiornato mi è sembrato molto accogliente. Il personale è stato molto
cordiale, ed in generale un ottima esperienza! Raccomando tantissimo.
1. L'hotel non era il massimo!
↳ Risposta: Non mi sono per niente offeso! Secondo me menti.
```

Figura 17: Visualizzazione recensioni attive dopo l'eliminazione

Indice delle figure

Figura 1: Prenotazione e soggiorno presso struttura ricettiva. In blu le interazioni da implementare	14
Figura 2: Operazioni sulle recensioni. Il cliente inserisce una recensione dopo aver superato i controlli di validità sul sito recensioni. Successivamente può eliminare o modificare la recensione. Dopo ogni operazione il sito aggiorna l'ordinamento delle recensioni.....	17
Figura 3: Struttura DID	19
Figura 4: Funzionamento protocollo DID.....	21
Figura 5: Funzionamento protocollo VC	21
Figura 6: Grafico Radar	29
Figura 7: VC rilasciata dal sistema di prenotazioni	36
Figura 8: VC rilasciata dall'hotel	37
Figura 9: VP del cliente	37
Figura 10: Inserimento recensione	38
Figura 11: Modifica effettuata	39
Figura 12: Eliminazione di una recensione	39
Figura 13: Inserimento Risposta	40
Figura 14: Ordinamento visualizzazione recensioni inserite. Maggiore priorità alle negative	40
Figura 15: Visualizzazione recensione dopo la modifica della seconda recensione modificata	40
Figura 16: Visualizzazione recensioni dopo la risposta da parte dell'hotel interessato.....	41
Figura 17: Visualizzazione recensioni attive dopo l'eliminazione	41